Oracle® Solaris Administration: Devices and File Systems
Ce logiciel et la documentation qui l'accompagne sont protégés par les lois sur la propriété intellectuelle. Ils sont concédés sous licence et soumis à des restrictions d'utilisation et de divulgation. Sauf disposition de votre contrat de licence ou de la loi, vous ne pouvez pas copier, reproduire, traduire, diffuser, modifier, brever, transmettre, distribuer, exposer, exécuter, publier ou afficher le logiciel, même partiellement, sous quelque forme et par quelque procédé que ce soit. Par ailleurs, il est interdit de procéder à toute ingénierie inverse du logiciel, de le désassembler ou de le décompiler, excepté à des fins d'interopérabilité avec des logiciels tiers ou tel que prescrit par la loi.

Les informations fournies dans ce document sont susceptibles de modification sans préavis. Par ailleurs, Oracle Corporation ne garantit pas qu'elles soient exemptes d'erreurs et vous invite, le cas échéant, à lui en faire part par écrit.

Si ce logiciel, ou la documentation qui l'accompagne, est concédé sous licence au Gouvernement des Etats-Unis, ou à toute entité qui délèvera la licence de ce logiciel ou l'utilise pour le compte du Gouvernement des Etats-Unis, la notice suivante s'applique :

U.S. GOVERNMENT RIGHTS. Programs, software, databases, and related documentation and technical data delivered to U.S. Government customers are "commercial computer software" or "commercial technical data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, the use, duplication, disclosure, modification, and adaptation shall be subject to the restrictions and license terms set forth in the applicable Government contract, and, to the extent applicable by the terms of the Government contract, the additional rights set forth in FAR 52.227-19, Commercial Computer Software License (December 2007). Oracle America, Inc., 500 Oracle Parkway, Redwood City, CA 94065.

Ce logiciel ou matériel a été développé pour un usage général dans le cadre d'applications de gestion des informations. Ce logiciel ou matériel n'est pas conçu ni n'est destiné à être utilisé dans des applications à risque, notamment dans des applications pouvant causer des dommages corporels. Si vous utilisez ce logiciel ou matériel dans le cadre d'applications dangereuses, il est de votre responsabilité de prendre toutes les mesures de secours, de sauvegarde, de redondance et autres mesures nécessaires à son utilisation dans des conditions optimales de sécurité. Oracle Corporation et ses affiliés déclinent toute responsabilité quant aux dommages causés par l'utilisation de ce logiciel ou matériel pour ce type d'applications.

Oracle et Java sont des marques déposées d'Oracle Corporation et/ou de ses affiliés. Tout autre nom mentionné peut correspondre à des marques appartenant à d'autres propriétaires qu'Oracle.

Intel et Intel Xeon sont des marques ou des marques déposées d'Intel Corporation. Toutes les marques SPARC sont utilisées sous licence et sont des marques ou des marques déposées d'Advanced Micro Devices. UNIX est une marque déposée de The Open Group.

Ce logiciel ou matériel et la documentation qui l'accompagne peuvent fournir des informations ou des liens donnant accès à des contenus, des produits et des services émanant de tiers. Oracle Corporation et ses affiliés déclinent toute responsabilité ou garantie expresse quant aux contenus, produits ou services émanant de tiers. En aucun cas, Oracle Corporation et ses affiliés ne sauraient être tenus pour responsables des pertes subies, des coûts occasionnés ou des dommages causés par l'accès à des contenus, produits ou services tiers, ou à leur utilisation.
Contents

Preface ...................................................................................................................................................17

1 Managing Removable Media (Overview) ........................................................................................21
  What’s New in Removable Media? ....................................................................................................21
  Changes and Improvements to Removable Media Management ..................................................21
  Where to Find Managing Removable Media Tasks .......................................................................24
  Removable Media Features and Benefits .....................................................................................25
  Comparison of Manual and Automatic Mounting .........................................................................25
  Overview of Accessing Removable Media .....................................................................................26

2 Managing Removable Media (Tasks) ..............................................................................................27
  Managing Removable Media (Task Map) .......................................................................................27
  Preparing Removable Media ..............................................................................................................28
    Removable Media Considerations ..................................................................................................28
  ▼ How to Load Removable Media ..................................................................................................29
  ▼ How to Format a Diskette (rmformat) ..........................................................................................30
  ▼ How to Create a File System on Removable Media ....................................................................31
  ▼ How to Create a File System on a DVD-RAM ............................................................................32
  ▼ How to Check a File System on Removable Media .....................................................................32
  ▼ How to Repair Bad Blocks on Removable Media .......................................................................33
    Applying Read or Write Protection and Password Protection to Removable Media .................33
  ▼ How to Enable or Disable Write Protection on Removable Media ............................................33
  ▼ How to Enable or Disable Read or Write Protection and Set a Password on Removable Media ..................................................................................................................34

3 Accessing Removable Media (Tasks) .............................................................................................35
  Accessing Removable Media (Task Map) ........................................................................................35
Accessing Removable Media ................................................................. 36
    Using Removable Media Names ...................................................... 36
    Guidelines for Accessing Removable Media Data .............................. 36
▼ How to Add a New Removable Media Drive ........................................ 36
▼ How to Disable or Enable Removable Media Services ................................ 37
▼ How to Access Information on Removable Media ................................. 38
▼ How to Copy Information From Removable Media ................................. 38
▼ How to Determine If Removable Media Is Still in Use ............................ 39
▼ How to Eject Removable Media ....................................................... 39
Accessing Removable Media on a Remote System (Task Map) .................... 40
▼ How to Make Local Media Available to Other Systems .......................... 40
▼ How to Access Removable Media on Remote Systems ......................... 41

4 Writing CDs and DVDs (Tasks) ......................................................... 43
Working With Audio CDs and Data CDs and DVDs ................................. 43
    CD/DVD Media Commonly Used Terms ........................................... 44
Writing CD and DVD Data and Audio CDs ........................................... 45
    Restricting User Access to Removable Media With RBAC .................. 46
▼ How to Restrict User Access to Removable Media With RBAC ............... 46
▼ How to Identify a CD or DVD Writer .................................................. 47
▼ How to Check the CD or DVD Media .................................................. 47
    Creating a Data CD or DVD ............................................................ 48
▼ How to Create an ISO 9660 File System for a Data CD or DVD ............... 48
▼ How to Create a Multi-Session Data CD .............................................. 49
    Creating an Audio CD ................................................................. 51
▼ How to Create an Audio CD ............................................................. 51
▼ How to Extract an Audio Track on a CD .............................................. 52
▼ How to Copy a CD .......................................................................... 53
▼ How to Erase CD-RW Media ........................................................... 54

5 Managing Devices (Overview/Tasks) ............................................... 55
What’s New in Device Management? .................................................... 56
    Customizing Driver Configuration ................................................. 56
    Solaris PCI Resource Manager ...................................................... 56
    New InfiniBand Administration Features ........................................... 57
New InfiniBand Diagnostic Tools and Commands ........................................................................... 57
New Ethernet Over InfiniBand Devices .......................................................................................... 57
New Hot Plugging Features .......................................................................................................... 57
Device Naming Enhancements ...................................................................................................... 58
Support for PCI Express (PCIe) .................................................................................................. 58
Where to Find Additional Device Management Tasks .............................................................. 59
Managing Devices in the Oracle Solaris OS .................................................................................. 60
Identifying Device Support ......................................................................................................... 60
About Device Drivers .................................................................................................................. 60
Automatic Configuration of Devices .......................................................................................... 61
Displaying Device Configuration Information ............................................................................. 63
Resolving Faulty Devices .......................................................................................................... 67
Adding a Peripheral Device to a System .................................................................................... 69
▼ How to Add a Peripheral Device .......................................................................................... 69
▼ How to Add a Device Driver ................................................................................................ 70
Accessing Devices ...................................................................................................................... 71
How Device Information Is Created .......................................................................................... 71
How Devices Are Managed ........................................................................................................ 72
Device Naming Conventions ..................................................................................................... 72
Logical Disk Device Names ........................................................................................................ 73
Logical Tape Device Names ..................................................................................................... 75
Logical Removable Media Device Names .................................................................................. 76

6 Dynamically Configuring Devices (Tasks) .................................................................................. 77
Dynamic Reconfiguration and Hot-Plugging ............................................................................... 77
Attachment Points ..................................................................................................................... 79
Detaching PCI or PCIe Adapter Cards ....................................................................................... 80
Attaching PCI or PCIe Adapter Cards ....................................................................................... 81
PCIe Hot-Plugging With the (hotplug) Command ..................................................................... 81
SCSI Hot-Plugging With the cfgadm Command (Task Map) ...................................................... 83
SCSI Hot-Plugging With the cfgadm Command ......................................................................... 84
▼ How to Display Information About SCSI Devices ............................................................... 84
▼ How to Unconfigure a SCSI Controller ................................................................................. 85
▼ How to Configure a SCSI Controller ................................................................................... 85
▼ How to Configure a SCSI Device ........................................................................................ 86
Contents

How to Disconnect a SCSI Controller .......................................................... 86
SPARC: How to Connect a SCSI Controller .................................................. 87
SPARC: How to Add a SCSI Device to a SCSI Bus ...................................... 88
SPARC: How to Replace an Identical Device on a SCSI Controller .............. 89
SPARC: How to Remove a SCSI Device ..................................................... 90
   Troubleshooting SCSI Configuration Problems ...................................... 91
How to Resolve a Failed SCSI Unconfigure Operation .............................. 93
PCI or PCIe Hot-Plugging With the \cfgadm\ Command (Task Map) .............. 93
PCI or PCIe Hot-Plugging With the \cfgadm\ Command .................................. 94
   PCIe LED Indicator Behavior ................................................................. 94
   How to Display PCI Slot Configuration Information .............................. 94
   How to Remove a PCI Adapter Card .................................................... 96
   How to Add a PCI Adapter Card .......................................................... 97
   Troubleshooting PCI Configuration Problems ..................................... 99
SATA Hot-Plugging With the \cfgadm\ Command ........................................ 100
   How to Unconfigure a SATA Device .................................................. 100
   How to Configure a SATA Device ...................................................... 101
Reconfiguration Coordination Manager (RCM) Script Overview ................ 101
   What Is an RCM Script? ................................................................. 102
   What Can an RCM Script Do? .......................................................... 102
   How Does the RCM Script Process Work? ........................................ 102
RCM Script Tasks ...................................................................................... 103
   Application Developer RCM Script (Task Map) .................................. 103
   System Administrator RCM Script (Task Map) .................................. 104
   Naming an RCM Script ...................................................................... 104
   Installing or Removing an RCM Script .............................................. 105
   How to Install an RCM Script .......................................................... 105
   How to Remove an RCM Script ........................................................ 105
   How to Test an RCM Script ............................................................. 106
   Tape Backup RCM Script Example .................................................... 106

7 Using USB Devices (Overview) ............................................................... 111
   What’s New in USB Devices? ............................................................ 112
      USB Interface Association Descriptor Support ................................ 112
      EHCI Isochronous Transfer Support .............................................. 112
USB Device Hotpluggable Behavior ........................................................................................112
x86: Support for USB CDs and DVDs in GRUB-Based Booting ..........................................113
USB Virtual Keyboard and Mouse Support ............................................................................113
Oracle Solaris Support for USB Devices .........................................................................................114
Overview of USB Devices .................................................................................................................115
Commonly Used USB Acronyms ............................................................................................115
USB Bus Description .................................................................................................................115
About USB in the Oracle Solaris OS ................................................................................................120
USB 2.0 Features .................................................................................................................120
USB Keyboards and Mouse Devices ........................................................................................122
USB Host Controller and Hubs ................................................................................................123
Guidelines for USB Cables ........................................................................................................124

8 Using USB Devices (Tasks) ........................................................................................................127
Managing USB Devices in the Oracle Solaris OS (Roadmap) ......................................................127
Using USB Mass Storage Devices (Task Map) ........................................................................128
Using USB Mass Storage Devices ........................................................................................129
Using USB Diskette Devices ........................................................................................................130
Using Non-Compliant USB Mass Storage Devices ..................................................................131
Hot-Plugging USB Mass Storage Devices .................................................................................131
Preparing to Use a USB Mass Storage Device .......................................................................135
▼ How to Display USB Device Information ............................................................................135
▼ How to Create a File System on a USB Mass Storage Device .............................................136
▼ How to Modify Partitions and Create a PCFS File System on a USB Mass Storage Device ..138
▼ How to Create a Solaris Partition and Modify the Slices on a USB Mass Storage Device ..141
▼ How to Mount or Unmount a USB Mass Storage Device .....................................................143
Troubleshooting Tips for USB Mass Storage Devices .................................................................144
Disabling Specific USB Drivers ................................................................................................145
▼ How to Disable Specific USB Drivers ..................................................................................145
▼ How to Remove Unused USB Device Links .......................................................................146
Using USB Audio Devices (Task Map) .....................................................................................146
Using USB Audio Devices ........................................................................................................147
Hot-Plugging Multiple USB Audio Devices .............................................................................148
▼ How to Add USB Audio Devices ..........................................................................................148
10 Managing Disks (Overview) ................................................................. 179
What’s New in Disk Management? .......................................................... 179
  Identifying Devices by Physical Locations ............................................. 180
Multiple Disk Sector Size Support ........................................................ 183
Two-Terabyte Disk Support for Installing and Booting the Oracle Solaris OS ................................................................. 183
iSNS Support in the Solaris iSCSI Target and Initiator ......................... 184
Solaris COMSTAR iSCSI Support .......................................................... 184
x86: Disk Management in the GRUB Boot Environment ...................... 184
Support for SCSI Disks That are Larger Than 2 Terabytes .................... 185
Where to Find Disk Management Tasks ............................................... 185
Overview of Disk Management ............................................................. 186
  Disk Terminology ............................................................................... 186
  About Disk Labels ............................................................................ 186
  EFI Disk Label ................................................................................. 187
  About Disk Slices ............................................................................ 190
  format Utility .................................................................................. 191
Partitioning a Disk ............................................................................... 194
  Partition Table Terminology .............................................................. 194
Displaying Partition Table Information ............................................... 195
  Using the Free Hog Slice ................................................................. 197

11 Administering Disks (Tasks) .............................................................. 199
Administering Disks (Task Map) ......................................................... 199
Identifying Disks on a System ............................................................. 200
  ▼ How to Identify the Disks on a System ........................................... 200
Formatting a Disk ................................................................................ 202
  ▼ How to Determine if a Disk Is Formatted ...................................... 202
Contents

▼ How to Enable the STMF Service ................................................................. 247
▼ How to Back Up and Restore a COMSTAR Configuration ........................................ 248
▼ How to Create an iSCSI LUN .............................................................. 248
▼ How to Create the iSCSI Target ............................................................. 250
▼ How to Configure an IB HCA for iSER ................................................... 250
▼ How to Configure an iSCSI Initiator ....................................................... 251
▼ How to Remove Discovered iSCSI Targets .................................................. 253
  Creating iSCSI Target Portal Groups ...................................................... 254
▼ How to Access iSCSI Disks ........................................................................ 256
  Making SCSI Logical Units Available ...................................................... 256
▼ How to Make a Logical Unit Available to All Systems ................................... 257
▼ How to Restrict LUN Access to Selected Systems ......................................... 257
Configuring Fibre Channel Devices With COMSTAR ............................................ 259
  Configuring Fibre Channel Ports For COMSTAR ......................................... 259
  Making Logical Units Available for FC and FCoE ....................................... 261
Configuring FCoE Devices With COMSTAR ..................................................... 263
  Configuring FCoE Ports ........................................................................ 263
  Enabling 802.3x PAUSE and Jumbo Frames on the Ethernet Interface ............ 264
▼ How to Create FCoE Target Ports ............................................................ 264
▼ How to Verify That an FCoE Target Port Is Working ...................................... 264
▼ How to Delete FCoE Target Ports ................................................................ 265
Configuring SRP Devices With COMSTAR ..................................................... 266
  Using COMSTAR Views With SRP .......................................................... 266
▼ How to Enable the SRP Target Service .................................................... 267
▼ How to Verify SRP Target Status ............................................................. 267
Configuring Authentication in Your iSCSI-Based Storage Network ..................... 267
▼ How to Configure CHAP Authentication for Your iSCSI Initiator .................. 268
▼ How to Configure CHAP Authentication for Your iSCSI Target .................... 270
  Using a Third-Party RADIUS Server to Simplify CHAP Management in Your iSCSI
  Configuration ........................................................................................ 270
Setting Up iSCSI Multipathed Devices in Oracle Solaris ........................................ 273
▼ How to Enable Multiple iSCSI Sessions for a Target ..................................... 274
Monitoring Your iSCSI Configuration ............................................................ 276
▼ How to Display iSCSI Configuration Information ......................................... 276
Modifying iSCSI Initiator and Target Parameters ............................................... 278
  Tuning iSCSI Parameters ..................................................................... 278
Contents

15 Configuring and Managing the Oracle Solaris Internet Storage Name Service (iSNS) ........291
   The iSNS Technology (Overview) ............................................................................................................. 291
   Configuring the iSNS Server ....................................................................................................................... 293
      Setting Up the iSNS Administrative Settings ......................................................................................... 294
      Using the Command Line Interface to Configure iSNS ........................................................................ 296
   Managing the iSNS Server and Clients .................................................................................................... 299
      ▼ How to Display the Status of a Discovery Domain Set ....................................................................... 299
      ▼ How to Display the Status of a Discovery Domain ............................................................................ 300
      ▼ How to Display the Status of Clients .............................................................................................. 300
      ▼ How to Remove a Client from a Discovery Domain .......................................................................... 300
      ▼ How to Remove a Discovery Domain from a Discovery Domain Set .............................................. 301
      ▼ How to Disable a Discovery Domain Set .......................................................................................... 301
      ▼ How to Remove a Discovery Domain Set .......................................................................................... 301

16 The format Utility (Reference) .................................................................................................................. 303
   Recommendations and Requirements for Using the format Utility .................................................... 303
   format Menu and Command Descriptions ................................................................................................. 304
      partition Menu .................................................................................................................................. 306
      x86: fdisk Menu .............................................................................................................................. 306
      analyze Menu .................................................................................................................................. 307
      defect Menu ...................................................................................................................................... 309
   Rules for Input to format Commands ...................................................................................................... 310
      Specifying Numbers to format Commands ......................................................................................... 310
      Specifying format Command Names ................................................................................................. 310
      Specifying Disk Names to format Commands .................................................................................... 311
   Getting Help on the format Utility ........................................................................................................... 311
17 Managing File Systems (Overview) ................................................................................................313

What's New in Oracle Solaris File Systems? ...................................................................................313

File System Monitoring Tool (fsstat) ...........................................................................................313

Oracle Solaris ZFS File System ................................................................................................314

Where to Find File System Management Tasks ...........................................................................314

Overview of File Systems ..................................................................................................................315

Types of Oracle Solaris File Systems ........................................................................................315

Default Oracle Solaris File Systems ........................................................................................320

Overview of Mounting and Unmounting File Systems ................................................................321

The Mounted File System Table ...................................................................................................322

The Virtual File System Table .......................................................................................................322

The NFS Environment .....................................................................................................................323

Automounting (autofs) ..................................................................................................................324

The Oracle Solaris SMB Service ...................................................................................................325

Determining a FileSystem's Type ...................................................................................................325

How to Determine a FileSystem's Type .......................................................................................325

18 Creating and Mounting File Systems (Tasks) ..........................................................................327

Creating Oracle Solaris File Systems ...............................................................................................327

Creating ZFS File Systems .............................................................................................................327

Creating a Temporary File System .................................................................................................328

Creating a LOFS File System ..........................................................................................................328

Mounting and Unmounting Oracle Solaris File Systems .............................................................328

Field Descriptions for the /etc/vfstab File ..................................................................................330

Prerequisites for Unmounting Oracle Solaris File Systems ..........................................................331

Creating and Mounting Oracle Solaris File Systems ...................................................................332

▼ How to Create an ZFS File System ..........................................................................................332

▼ How to Create and Mount a Legacy UFS File System ...........................................................333

▼ How to Create and Mount a TMPFS File System ..................................................................334

▼ How to Create and Mount an LOFS File System ..................................................................335

▼ How to Add an Entry to the /etc/vfstab File .........................................................................336

▼ How to Mount a File System (/etc/vfstab File) ......................................................................337

▼ How to Mount an NFS File System (mount Command) .........................................................338

▼ x86: How to Mount a PCFS (DOS) File System From a Hard Disk (mount Command) ... 339

▼ How to Stop All Processes Accessing a File System .................................................................340
19 Configuring Additional Swap Space (Tasks) .............................................................. 343
   About Swap Space ........................................................................................................... 343
   Swap Space and Virtual Memory .................................................................................... 343
   Swap Space and the TMPFS File System ................................................................. 344
   Swap Space and Dump Device Configuration ....................................................... 344
   How Do I Know If I Need More Swap Space? .......................................................... 345
   How Swap Space Is Allocated ......................................................................................... 346
   Planning for Swap Space ............................................................................................... 347
   Allocating Swap Space for ZFS-Based Systems .................................................... 348
   Monitoring Swap Resources ....................................................................................... 349
   Adding or Changing Swap Space in an Oracle Solaris ZFS Root Environment .............. 350
   How to Add Swap Space in an Oracle Solaris ZFS Root Environment ....................... 350

20 Copying Files and File Systems (Tasks) .................................................................... 353
   Commands for Copying File Systems ....................................................................... 353
   Copying Directories Between File Systems (cpio Command) .................................... 356
   Copying Files and File Systems to Tape ...................................................................... 357
   Copying Files to Tape (tar Command) ....................................................................... 358
   Copying Files to Tape With the pax Command ......................................................... 360
   Copying Files to Tape With the cpio Command ......................................................... 361
   Copying All Files in a Directory to a Tape (cpio) ...................................................... 361
   Copying All Files in a Directory to a Tape With the cpio Command ....................... 361
   Copying All Files in a Directory to a Tape With the cpio Command .......................... 361
   Copying All Files in a Directory to a Tape With the pax Command ....................... 361
   Copying All Files in a Directory to a Tape With the pax Command ....................... 361
   Copying All Files in a Directory to a Tape With the pax Command ....................... 361

Oracle Solaris Administration: Devices and File Systems • February 2012
Contents

21 Managing Tape Drives (Tasks) ................................................................. 367

Choosing Which Media to Use ................................................................. 367
Backup Device Names .................................................................................. 368
  Specifying the Rewind Option for a Tape Drive ....................................... 369
  Specifying Different Densities for a Tape Drive ......................................... 369
Displaying Tape Drive Status ..................................................................... 370
  ▼ How to Display Tape Drive Status ............................................................ 370
Handling Magnetic Tape Cartridges .......................................................... 371
  Retensioning a Magnetic Tape Cartridge .................................................... 371
  Rewinding a Magnetic Tape Cartridge ....................................................... 371
Guidelines for Drive Maintenance and Media Handling .......................... 371

Index ............................................................................................................. 373
Preface

System Administration Guide: Devices and File Systems is part of a set that includes a significant part of the Oracle Solaris system administration information. This guide contains information for both SPARC based and x86 based systems.

This book assumes you have completed the following tasks:

■ Installed the Oracle Solaris 11 Operating System
■ Set up all the networking software that you plan to use

For the Oracle Solaris 11 releases, new features of interest to system administrators are covered in sections called What's New in ... ? in the appropriate chapters.

Note – This Oracle Solaris release supports systems that use the SPARC and x86 families of processor architectures. The supported systems appear in the Oracle Solaris Hardware Compatibility List at http://www.oracle.com/webfolder/technetwork/hcl/index.html. This document cites any implementation differences between the platform types.

Who Should Use This Book

This book is intended for anyone responsible for administering one or more systems running the Oracle Solaris 11 release. To use this book, you should have 1–2 years of UNIX system administration experience. Attending UNIX system administration training courses might be helpful.

How the System Administration Guides Are Organized

Here is a list of the topics that are covered by the System Administration Guides.

<table>
<thead>
<tr>
<th>Book Title</th>
<th>Topics</th>
</tr>
</thead>
<tbody>
<tr>
<td>Booting and Shutting Down Oracle Solaris on SPARC Platforms</td>
<td>Booting and shutting down a system, managing boot services, modifying boot behavior, booting from ZFS, managing the boot archive, and troubleshooting booting on SPARC platforms</td>
</tr>
<tr>
<td>Book Title</td>
<td>Topics</td>
</tr>
<tr>
<td>------------</td>
<td>--------</td>
</tr>
<tr>
<td><strong>Booting and Shutting Down Oracle Solaris on x86 Platforms</strong></td>
<td>Booting and shutting down a system, managing boot services, modifying boot behavior, booting from ZFS, managing the boot archive, and troubleshooting booting on x86 platforms</td>
</tr>
<tr>
<td><strong>Oracle Solaris Administration: Common Tasks</strong></td>
<td>Using Oracle Solaris commands, booting and shutting down a system, managing user accounts and groups, managing services, hardware faults, system information, system resources, and system performance, managing software, printing, the console and terminals, and troubleshooting system and software problems</td>
</tr>
<tr>
<td><strong>Oracle Solaris Administration: Devices and File Systems</strong></td>
<td>Removable media, disks and devices, file systems, and backing up and restoring data</td>
</tr>
<tr>
<td><strong>Oracle Solaris Administration: IP Services</strong></td>
<td>TCP/IP network administration, IPv4 and IPv6 address administration, DHCP, IPsec, IKE, IP Filter, and IPQoS</td>
</tr>
<tr>
<td><strong>Oracle Solaris Administration: Naming and Directory Services</strong></td>
<td>DNS, NIS, and LDAP naming and directory services, including transitioning from NIS to LDAP</td>
</tr>
<tr>
<td><strong>Oracle Solaris Administration: Network Interfaces and Network Virtualization</strong></td>
<td>Automatic and manual IP interface configuration including WiFi wireless; administration of bridges, VLANs, aggregations, LLDP, and IPMP; virtual NICs and resource management.</td>
</tr>
<tr>
<td><strong>Oracle Solaris Administration: Network Services</strong></td>
<td>Web cache servers, time-related services, network file systems (NFS and autosfs), mail, SLP, and PPP</td>
</tr>
<tr>
<td><strong>Oracle Solaris Administration: Oracle Solaris Zones, Oracle Solaris 10 Zones, and Resource Management</strong></td>
<td>Resource management features, which enable you to control how applications use available system resources; Oracle Solaris Zones software partitioning technology, which virtualizes operating system services to create an isolated environment for running applications; and Oracle Solaris 10 Zones, which host Oracle Solaris 10 environments running on the Oracle Solaris 11 kernel</td>
</tr>
<tr>
<td><strong>Oracle Solaris Administration: Security Services</strong></td>
<td>Auditing, device management, file security, BART, Kerberos services, PAM, Cryptographic Framework, Key Management, privileges, RBAC, SASL, Secure Shell, and virus scanning</td>
</tr>
<tr>
<td><strong>Oracle Solaris Administration: SMB and Windows Interoperability</strong></td>
<td>SMB service, which enables you to configure an Oracle Solaris system to make SMB shares available to SMB clients; SMB client, which enables you to access SMB shares; and native identity mapping services, which enables you to map user and group identities between Oracle Solaris systems and Windows systems</td>
</tr>
<tr>
<td><strong>Oracle Solaris Administration: ZFS File Systems</strong></td>
<td>ZFS storage pool and file system creation and management, snapshots, clones, backups, using access control lists (ACLs) to protect ZFS files, using ZFS on a Solaris system with zones installed, emulated volumes, and troubleshooting and data recovery</td>
</tr>
<tr>
<td>Book Title</td>
<td>Topics</td>
</tr>
<tr>
<td>------------------------------------------------------------</td>
<td>----------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td><strong>Trusted Extensions Configuration and Administration</strong></td>
<td>System installation, configuration, and administration that is specific to Trusted Extensions</td>
</tr>
<tr>
<td><strong>Oracle Solaris 11 Security Guidelines</strong></td>
<td>Securing an Oracle Solaris system, as well as usage scenarios for its security features, such as zones, ZFS, and Trusted Extensions</td>
</tr>
<tr>
<td><strong>Transitioning From Oracle Solaris 10 to Oracle Solaris 11</strong></td>
<td>Provides system administration information and examples for transitioning from Oracle Solaris 10 to Oracle Solaris 11 in the areas of installation, device, disk, and file system management, software management, networking, system management, security, virtualization, desktop features, user account management, and user environments</td>
</tr>
</tbody>
</table>

## Access to Oracle Support


## What Typographic Conventions Mean

The following table describes the typographic conventions used in this book.

### TABLE P–1 Typographic Conventions

<table>
<thead>
<tr>
<th>Typeface or Symbol</th>
<th>Meaning</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>AaBbCc123</code></td>
<td>The names of commands, files, and directories; on screen computer output</td>
<td>Edit your <code>.login</code> file. Use <code>ls -a</code> to list all files. <code>machine_name%</code> you have mail.</td>
</tr>
<tr>
<td><code>AaBbCc123</code></td>
<td>What you type, contrasted with on screen computer output</td>
<td><code>machine_name% su</code> Password:</td>
</tr>
<tr>
<td><code>AaBbCc123</code></td>
<td>Command-line placeholder: replace with a real name or value</td>
<td>To delete a file, type <code>rm filename</code>.</td>
</tr>
<tr>
<td><code>AaBbCc123</code></td>
<td>Book titles, new words or terms, or words to be emphasized</td>
<td>Read Chapter 6 in <em>User’s Guide</em>. These are called <code>class</code> options. Do <em>not</em> save changes yet.</td>
</tr>
</tbody>
</table>
Shell Prompts in Command Examples

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

<table>
<thead>
<tr>
<th>Shell</th>
<th>Prompt</th>
</tr>
</thead>
<tbody>
<tr>
<td>C shell prompt</td>
<td>machine_name%</td>
</tr>
<tr>
<td>C shell superuser prompt</td>
<td>machine_name#</td>
</tr>
<tr>
<td>Bourne shell and Korn shell prompt</td>
<td>$</td>
</tr>
<tr>
<td>Bourne shell and Korn shell superuser prompt</td>
<td>#</td>
</tr>
</tbody>
</table>
Managing Removable Media (Overview)

This chapter provides general guidelines for managing removable media in the Oracle Solaris OS.

This is a list of the overview information in this chapter.

- "What's New in Removable Media?" on page 21
- "Where to Find Managing Removable Media Tasks" on page 24
- "Removable Media Features and Benefits" on page 25
- "Comparison of Manual and Automatic Mounting" on page 25
- "Overview of Accessing Removable Media" on page 26

What's New in Removable Media?

Changes and Improvements to Removable Media Management

Oracle Solaris 11: Previous features for managing removable media have been removed and replaced with services and methods that provide better removable media management.

The following new features are available:

- New removable media services are managed by SMF.
  
  online 12:17:54 svc:/system/hal:default
  online 12:17:56 svc:/system/filesystem/rmvolmgr:default
  online 12:17:26 svc:/system/dbus:default

- Removable media is now mounted automatically in the /media directory. However, symbolic links to /media are provided from previous media mount points, /cdrom and /rmdisk, for compatibility purposes.
  
  For example, a compact flash memory card (/dev/dsk/c4d0p0:1) is mounted as follows:
$ ls /media/NIKON

For example, a USB memory stick (/dev/dsk/c3t0d0s0) is mounted as follows:

$ ls /media/U3

- The default removable media volume manager, rmmgr, is responsible for following activities:
  - Mounting and unmounting volumes.
  - The root instance of rmmgr starts at system boot. However, you can configure your session’s configuration files to start an instance of rmmgr when you log in. When run within a user session, rmmgr only mounts devices owned by the current user or session and does not conflict with the root instance.
  - When rmmgr exits, it unmounts all media that it mounted.
  - For compatibility purposes, rmmgr creates symbolic links under the /cdrom or /rmdisk directories to the actual mount points under /media.
  - A special rmmgr run mode is available for CDE compatibility.
  - The hardware abstraction layer (HAL) daemon, hald, provides a view of the device attached to a system. This view is updated automatically as hardware configuration changes, by hotplugging or other mechanisms.

HAL represents a piece of hardware as a device object. A device object is identified by a unique device identifier (UDI) and carries a set of key-value pairs referred to as device properties. Some properties are derived from the actual hardware, some are merged from device information files (.fdi files) and some are related to the actual device configuration.

The following features are removed:

- Only USB diskettes are supported in this release.
- The vold daemon, the volfs file system, and the volfs service have been removed.
  
  svc:/system/filesystem/volfs
- Logical device names for removable media under the /vol directory, such as /vol/dev/rdsk/... or /vol/dev/aliases/..., are no longer provided.

To access removable media by its logical device name, the /dev device should be used. For example:

/dev/rdsk/c0t6d0s2

- Some vold device nicknames are no longer available. The following eject -l output identified the available device nicknames for each device and in this example, the mounted media pathname (/media/SOL_11_X86_4):

  $ eject -l /dev/dsk/c2t0d0s2
cdrom,cdrom0,cd,cd0,sr,sr0,SOL_11_X86_4,/media/SOL_11_X86_4

  The comma-separated list shows the nicknames that can be used to eject each device.
Customizations that were made in `volatile.conf` and `rmmount.conf` are no longer available because these configuration files no longer exist. For information about managing media customizations, see “Customizing Removable Media Management” on page 24.

Commands that begin with `vol` commands except for `volcheck` and `volrmmount`.

**Backward Compatibility**

The following features provide backward compatibility with previous Oracle Solaris removable media features:

- Removable media mount points have moved to the `/media` directory, which is used to mount removable media, such as CD-ROMs and USB devices. Symbolic links to `/media` from previous media mounts points, such as `/cdrom` and `/rdisk`, are provided for compatibility purposes.

- The `rmformat` command is still available. The output of this command is identical to what it looks in previous Solaris releases with `volatile` disabled.

  For example:
  ```
  # rmformat
  Looking for devices...
  1. Logical Node: /dev/rdsk/c0t6d0s2
     Physical Node: /pci@1f,4000/scsi@3/sd@6,0
     Connected Device: TOSHIBA DVD-ROM SD-M1401 1009
     Device Type: DVD Reader
     Bus: SCSI
     Size: 2.9 GB
     Label: <None>
     Access permissions: <Unknown>
  ```

- The `eject` command is available but has been enhanced. For more information, see “Ejecting Removable Media” on page 24.

**Mounting and Unmounting Removable Media**

Most commands that begin with `vol` are removed in this release. A modified version of `rmmount` and a new `rmumount` command are available to mount and unmount removable media.

These commands can be used to mount by device name, label, or mount point. For example, to mount an iPod:

```
% rmmount ipod
```

For example, to unmount the file systems on a DVD:

```
# rmumount cdrom
```

`cdrom /dev/dsk/c0t6d0s5 unmounted`

`cdrom /dev/dsk/c0t6d0s0 unmounted`

For more information, see `rmmount(1M)`.
**Ejecting Removable Media**

As in previous Solaris releases, use the `eject` command to unmount and eject removable media. However, the following `eject` options are available:

- `-f` Forces the device to eject even if the device is busy.
- `-l` Displays paths and nicknames of devices that can be ejected.
- `-t` A CD-ROM tray close command is provided to the device. Not all devices support this option.

For example, to eject by its volume label:

```
% eject mypictures
```

For more information, see `eject(1)`.

**Customizing Removable Media Management**

For most customizations that were available in the `vold.conf` and `rmmount.conf` files, you will need to either use Desktop Volume Manager preferences or modify the `.fdt` files.

- For `rmmount.conf` actions, you will need to use either Desktop Volume Manager actions, `gconf`, or HAL callouts.
- Previously, `rmmount.conf` actions could be run as root on behalf of ordinary users. Now, this is done by installing callout executables in the `/usr/lib/hal` directory.

**Disabling Removable Media Features**

To prevent volumes from mounting outside of user sessions, disable the `rmvolmgr` service. For example:

```
# svcadm disable rmvolmgr
```

**Where to Find Managing Removable Media Tasks**

Use these references to find step-by-step instructions for managing removable media.

<table>
<thead>
<tr>
<th>Removable Media Management Task</th>
<th>For More Information</th>
</tr>
</thead>
</table>
| Format removable media          | Chapter 2, "Managing Removable Media (Tasks)"  
| Access removable media          | Chapter 3, "Accessing Removable Media (Tasks)"  
| Write data CDs and DVDs and music CDs | Chapter 4, "Writing CDs and DVDs (Tasks)"  

Removable Media Features and Benefits

The Oracle Solaris release gives users and software developers a standard interface for dealing with removable media. Removable media services provide the following benefits:

- Automatically mounts removable media. For a comparison of manual and automatic mounting, see the following section.
- Enables you to access removable media without having to become an administrator.
- Allows you to give other systems on the network automatic access to any removable media on your local system. For more information, see Chapter 3, “Accessing Removable Media (Tasks).”

Comparison of Manual and Automatic Mounting

The following table compares the steps involved in manual mounting (without removable media services) and automatic mounting (with removable media management) of removable media.

<table>
<thead>
<tr>
<th>Steps</th>
<th>Manual Mounting</th>
<th>Automatic Mounting</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Insert media.</td>
<td>Insert media.</td>
</tr>
<tr>
<td>2</td>
<td>Become an administrator.</td>
<td>For USB diskettes, use the <code>volcheck</code> command.</td>
</tr>
<tr>
<td>3</td>
<td>Determine the location of the media device.</td>
<td>Removable media services automatically perform many of the tasks that are required to manually mount and work with removable media.</td>
</tr>
<tr>
<td>4</td>
<td>Create a mount point.</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>Make sure you are not in the mount point directory.</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>Mount the device and use the proper mount options.</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>Exit the administrator account.</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>Work with files on media.</td>
<td>Work with files on media.</td>
</tr>
<tr>
<td>9</td>
<td>Become an administrator.</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>Unmount the media device.</td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>Eject media.</td>
<td>Eject media.</td>
</tr>
</tbody>
</table>
Overview of Accessing Removable Media

Essentially, removable media services enable you to access removable media just as manual mounting does, but more easily and without the need for administrative access.

If the media contains a file system and a label, the media label name is used to name the /media/pathname mount point. If a label is not present, the disk model name is used to name the media, such as /media/cdrom. A generic nickname is used only for legacy symbolic links. For example, /rmdisk/rmdisk0.

If your system has more than one type of removable device, see the following table for their access points.

<table>
<thead>
<tr>
<th>Access</th>
<th>Insert</th>
<th>Unlabeled Media Pathnames</th>
<th>Labeled Media Pathname Examples</th>
</tr>
</thead>
<tbody>
<tr>
<td>Files on a removable hard disk</td>
<td>The removable hard disk and type volcheck on the command line</td>
<td>/media/usb-disk or the legacy path /rmdisk/rmdisk0</td>
<td>/media/00JB-00CRA0</td>
</tr>
<tr>
<td>Files on a DVD</td>
<td>The DVD and wait for a few seconds</td>
<td>/media/cdrom</td>
<td>/media/sol_10_sparc</td>
</tr>
<tr>
<td>Files on a DVD</td>
<td>The DVD and wait for a few seconds</td>
<td>/media/cdrom</td>
<td>/media/SOL_11_X86</td>
</tr>
</tbody>
</table>

You can use the rmmount -l command to identify mounted media on your system. For example:

```
# rmmount -l
/dev/dsk/c5t0d0p0  rmdisk6, /media/FD-05PUB
/dev/dsk/c4t0d3p0  rmdisk5, /media/223UHS-SD-MMC
/dev/dsk/c2t0d0s2  cdrom1, cd1, srl, SOL_11_X86, /media/SOL_11_X86
/dev/dsk/c3t0d0p0  rmdisk2, /media/00JB-00CRA0
```

In the above output, the mounted devices are as follows:

- /dev/dsk/c5t0d0p0  USB diskette
- /dev/dsk/c4t0d3p0  CF card in a USB card reader
- /dev/dsk/c2t0d0s2  DVD-ROM
- /dev/dsk/c3t0d0p0  Removable USB disk
Managing Removable Media (Tasks)

This chapter describes how to manage removable media from the command line in the Oracle Solaris OS.

For information on the procedures associated with managing removable media, see “Managing Removable Media (Task Map)” on page 27. For background information on removable media, see Chapter 1, “Managing Removable Media (Overview).”

Managing Removable Media (Task Map)

The following task map describes the tasks for managing removable media.

<table>
<thead>
<tr>
<th>Task</th>
<th>Description</th>
<th>For Instructions</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Load media.</td>
<td>Insert the diskette into the drive and type the <code>volcheck</code> command.</td>
<td>“How to Load Removable Media” on page 29</td>
</tr>
<tr>
<td>2. (Optional) Format the diskette.</td>
<td>Format diskette.</td>
<td>“How to Format a Diskette (<code>rmformat</code>)” on page 30</td>
</tr>
<tr>
<td>3. (Optional) Add a PCFS file system.</td>
<td>Add a PCFS file system to use the media for transferring files.</td>
<td>“How to Create a File System on Removable Media” on page 31</td>
</tr>
<tr>
<td>4. (Optional) Check the media.</td>
<td>Verify the integrity of the file system on the media.</td>
<td>“How to Check a File System on Removable Media” on page 32</td>
</tr>
<tr>
<td>5. (Optional) Repair bad blocks on the media.</td>
<td>Repair any bad blocks on the media, if necessary.</td>
<td>“How to Repair Bad Blocks on Removable Media” on page 33</td>
</tr>
</tbody>
</table>
Preparing Removable Media

The following sections describe how to prepare removable media for use.

Removable Media Considerations

Keep the following considerations in mind when working with diskettes:

- SunOS file system formats consist of the basic “bit” formatting in addition to the structure to support a SunOS file system. A complete format for a DOS file system consists of the basic “bit” formatting in addition the structure to support either an MS-DOS or an NEC-DOS file system. The procedures required to prepare a media for each type of file system are different. Therefore, before you format a diskette, consider which procedure to follow. For more information, see “Managing Removable Media (Task Map)” on page 27.
- For information on removable media names, see “Using Removable Media Names” on page 36.
- Diskettes that are not named (that is, they have no “label”) are assigned the default name of unnamed_floppy.
- Diskettes that are not named (that is, they have no “label”) are assigned the default name of floppy.

An Oracle Solaris system can format the following file system types:

- ZFS or UFS
- MS-DOS or NEC-DOS (PCFS)
- UDFS

On an Oracle Solaris system (either SPARC or x86), you can format diskettes with the following densities.

<table>
<thead>
<tr>
<th>Diskette Size</th>
<th>Diskette Density</th>
<th>Capacity</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.5&quot;</td>
<td>High density (HD)</td>
<td>1.44 MB</td>
</tr>
<tr>
<td>3.5&quot;</td>
<td>Double density (DD)</td>
<td>720 KB</td>
</tr>
</tbody>
</table>

By default, the diskette drive formats a diskette to a like density. This default means that a 1.44 MB drive attempts to format a diskette for 1.44 MB, regardless of whether the diskette
is, in fact, a 1.44 MB diskette, unless you instruct it otherwise. In other words, a diskette can be formatted to its capacity or lower, and a drive can format to its capacity or lower.

**Formatting Diskettes**

You can use the `rmformat` command to format and protect rewritable diskettes. File systems are mounted automatically. So, you might have to unmount media before you can format it, if the media contains an existing file system.

The `rmformat` command has three formatting options:

- **quick** – This option formats diskettes without certification or with limited certification of certain tracks on the media.
- **long** – This option completely formats diskettes. For some devices, the use of this option might include the certification of the whole media by the drive.
- **force** – This option formats completely without user confirmation. For diskettes with a password-protection mechanism, this option clears the password before formatting. This feature is useful when a password is forgotten. On diskettes without password protection, this option forces a long format.

**How to Load Removable Media**

For information about removable media hardware considerations, see “Removable Media Considerations” on page 28.

1. **Insert the media.**

2. **Ensure that the media is formatted.**
   
   If you aren’t sure, insert the media and check the status messages in the system console window, as described in Step 3. If you need to format the media, go to “How to Format a Diskette (rmformat)” on page 30.

3. **(Optional) Notify volume management if you are using a legacy, non-USB diskette device.**

   
   ```bash
   $ volcheck -v
   ```

   Two status messages are possible:

   ```plaintext
   media was found   Volume management detected the media and will attempt to mount it in the directory described in “Using Removable Media Names” on page 36.
   ```

   If the media is formatted properly, no error messages appear in the console.
If the media is not formatted, the “media was found” message is still displayed. However, error messages similar to the following appear in the system console window:

fd0: unformatted diskette or no diskette in the drive

fd0: read failed (40 1 0)

fd0: bad format

You must format the media before volume management can mount it. For more information, see Chapter 2, “Managing Removable Media (Tasks).”

no media was found

Volume management did not detect the media. Ensure that the media is inserted properly, and run volcheck again. If unsuccessful, check the media, which could be damaged. You can also try to mount the media manually.

4 Verify that the media was mounted by listing its contents.
   For example, do the following for a diskette:
   
   $ ls /media/floppy
   lost+found myfiles

▼ How to Format a Diskette (rmformat)

You can use the rmformat command to format a diskette. By default, this command creates two partitions on the media: partition 0 and partition 2 (the whole media).

1 Verify that removable media service is running. If so, you can use the shorter nickname for the device name.

   # svcs hal dbus rmvolmgr
   STATE STIME FMRI
   online Apr_09 svc:/system/dbus:default
   online Apr_09 svc:/system/hal:default
   online Apr_09 svc:/system/filesystem/rmvolmgr:default

   For information on restarting removable media services, see “How to Disable or Enable Removable Media Services” on page 37. For information on identifying media device names, see “Using Removable Media Names” on page 36.

2 Format the diskette.

   $ rmformat -F [ quick | long | force ] device-name

   See “Formatting Diskettes” on page 29 for more information on rmformat formatting options.
If the `rmformat` output indicates bad blocks, see “How to Repair Bad Blocks on Removable Media” on page 33.

3 (Optional) Label the diskette with an 8-character label.

$ rmformat -b label device-name

For information on creating a DOS label, see `mkfs_pcfs(1M)`.

Example 2–1 Formatting a Diskette

This example shows how to format a diskette.

$ rmformat -F quick /dev/rdiskette

Formatting will erase all the data on disk.
Do you want to continue? (y/n) y

How to Create a File System on Removable Media

1 (Optional) Format a diskette, if necessary.

To format a USB diskette, use syntax similar to the following:

$ rmformat -F long /dev/rdsk/c11t0d0p0

2 (Optional) Create an alternate Solaris partition table.

$ rmformat -s slice-file device-name

A sample slice file appears similar to the following:

slices: 0 = 0, 30MB, "wm", "home" :
  1 = 30MB, 51MB :
  2 = 0, 94MB, "wm", "backup" :
  6 = 81MB, 13MB

3 Become an administrator.

For more information, see “How to Use Your Assigned Administrative Rights” in Oracle Solaris Administration: Security Services.

4 Determine the appropriate file system type and select one of the following:

- Create a PCFS file system. For example:
  
  # mkfs -F pcfs -o nodisk, size=9800 /dev/rdsk/c11t0d0p0

- Create a UDFS file system. For example:

  # mkfs -F udfs /dev/rdsk/c0t1d0p0
How to Create a File System on a DVD-RAM

Use this procedure to create a file system on a DVD-RAM.

1 Become an administrator.
For more information, see “How to Use Your Assigned Administrative Rights” in Oracle Solaris Administration: Security Services.

2 Create a file system on the DVD-RAM device.
   ■ Create a UDFS file system. For example:
     
     ```
     # mkfs -F udfs /dev/rdsk/c0t0d0s2
     ```

3 Mount the file system.
   ■ Mount a UDFS file system. For example:
     
     ```
     # mount -F udfs /dev/dsk/c0t0d0s2 /mnt
     ```

4 Verify that you can read or write to the file system.

5 When finished, eject the DVD-RAM.

How to Check a File System on Removable Media

1 Become an administrator.
For more information, see “How to Use Your Assigned Administrative Rights” in Oracle Solaris Administration: Security Services.

2 Identify the file system type and select one of the following:
   ■ Check a UDFS file system.
     
     ```
     # fsck -F udfs device-name
     ```
   ■ Check a PCFS file system.
     
     ```
     # fsck -F pcfs device-name
     ```

Example 2–2 Checking a PCFS File System on Removable Media

The following example shows how check the consistency of a PCFS file system on media.

```bash
# fsck -F pcfs /dev/rdsk/c0t4d0s2
** /dev/rdsk/c0t4d0s2
** Scanning file system meta-data
```
** Correcting any meta-data discrepancies
1457664 bytes.
0 bytes in bad sectors.
0 bytes in 0 directories.
0 bytes in 0 files.
1457664 bytes free.
512 bytes per allocation unit.
2847 total allocation units.
2847 available allocation units.

▼ How to Repair Bad Blocks on Removable Media

You can only use the `rmformat` command to verify, analyze, and repair bad sectors that are found during verification if the drive supports bad block management. Most USB memory sticks do not support bad block management.

If the drive supports bad block management, a best effort is made to rectify the bad block. If the bad block cannot be rectified despite the best effort mechanism, a message indicates the failure to repair the media.

1 Repair bad blocks on removable media.

   $ rmformat -c block-numbers device-name

   Supply the block number in decimal, octal, or hexadecimal format from a previous `rmformat` session.

2 Verify the media.

   $ rmformat -V read device-name

Applying Read or Write Protection and Password Protection to Removable Media

You can apply read protection or write protection, and set a password, on removable media that support this feature.

▼ How to Enable or Disable Write Protection on Removable Media

1 Determine whether you want to enable or disable write protection and select one of the following:
   ▪ Enable write protection.
How to Enable or Disable Read or Write Protection and Set a Password on Removable Media

You can apply a password with a maximum of 32 characters for removable media that support this feature.

You will receive a warning message if you attempt to apply a password on media that does not support this feature.

1. Determine whether you want to enable or disable read protection or write protection and set a password. Select one of the following:
   - Enable read protection or write protection.
     
     $ rmformat -W enable device-name
     Please enter password (32 chars maximum): xxx
     Please reenter password:
     $ rmformat -R enable device-name
     Please enter password (32 chars maximum): xxx
     Please reenter password:
   - Disable read protection or write protection and remove the password.
     
     $ rmformat -W disable device-name
     Please enter password (32 chars maximum): xxx
     $ rmformat -R disable device-name
     Please enter password (32 chars maximum): xxx

2. Verify whether the media’s read protection or write protection is enabled or disabled.

   $ rmformat -p device-name
This chapter describes how to access removable media from the command line in the Oracle Solaris OS.

For information on the procedures associated with accessing removable media, see the following:

- “Accessing Removable Media (Task Map)” on page 35
- “Accessing Removable Media on a Remote System (Task Map)” on page 40

For background information on removable media, see Chapter 1, “Managing Removable Media (Overview).”

### Accessing Removable Media (Task Map)

The following task map describes the tasks for accessing removable media.

<table>
<thead>
<tr>
<th>Task</th>
<th>Description</th>
<th>For Instructions</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. (Optional) Add the removable media drive.</td>
<td>Add the removable media drive to your system, if necessary.</td>
<td>“How to Add a New Removable Media Drive” on page 36</td>
</tr>
<tr>
<td>2. (Optional) Decide whether you want to use removable media with or without volume management (vold).</td>
<td>Volume management (vold) runs by default. Decide whether you want to use removable media with or without volume management.</td>
<td>“How to Disable or Enable Removable Media Services” on page 37</td>
</tr>
<tr>
<td>3. Access removable media.</td>
<td>Access different kinds of removable media with or without volume management running.</td>
<td>“How to Access Information on Removable Media” on page 38</td>
</tr>
<tr>
<td>4. (Optional) Copy files or directories.</td>
<td>Copy files or directories from the media as you would from any other location in the file system.</td>
<td>“How to Copy Information From Removable Media” on page 38</td>
</tr>
</tbody>
</table>
Accessing Removable Media

You can access information on removable media with or without using volume management. For information on accessing information on removable media with GNOME’s File Manager, see the GNOME desktop documentation.

Using Removable Media Names

Removable media is now mounted automatically in the /media directory. However, symbolic links to /media are provided from previous media mount points, /cdrom and /rmdisk, for compatibility purposes.

For example, a compact flash memory card (/dev/dsk/c4d0p0:1) is mounted as follows:

$ ls /media/NIKON

For example, a USB memory stick (/dev/dsk/c3t0d0s0) is mounted as follows:

$ ls /media/U3

Guidelines for Accessing Removable Media Data

Most DVDs are formatted to the ISO 9660 standard, which is portable. So, DVDs can be mounted by volume management.

To accommodate possible different formats, a DVD is split into slices. Slices are similar in effect to partitions on hard disks. The 9660 portion is portable. If you are having trouble mounting a DVD, particularly if it is an installation DVD, make sure that its file system is appropriate for your system’s architecture. For example, you can check the label on DVD.

How to Add a New Removable Media Drive

Generally, most modern bus types support hot-plugging. This means you can insert a disk in an empty slot and the system recognizes it.
For more information about hot-plugging devices, see Chapter 6, “Dynamically Configuring Devices (Tasks).”

1 Become an administrator.

2 Connect the new media drive.
   See your hardware handbook for specific instructions.

3 Confirm that the system sees the new media drive.
   
   ```
   # rmformat
   Looking for devices...
   ```

## How to Disable or Enable Removable Media Services

Occasionally, you might want to manage media without using removable media services. This section describes how to disable and enable removable media services.

Disabling these services means that you would have to mount all media manually by using the `mount` command.

1 **Ensure that the media is not being used.**
   If you are not sure whether you have found all users of the media, use the `fuser` command, see “How to Determine If Removable Media Is Still in Use” on page 39.

2 Become an administrator.

3 Select one of the following:
   - You can disable some or all removable media features in this release:
     - To prevent volumes from mounting outside of user sessions, disable the `rmvolmgr` service. For example:
       ```bash
       # svcadm disable rmvolmgr
       ```
     - To prevent any volume management, disable the `dbus`, `hal`, and `rmvolmgr` services.
       ```bash
       # svcadm disable rmvolmgr
       # svcadm disable dbus
       # svcadm disable hal
       ```
     Disabling these services means that you would have to mount all media manually by using the `mount` command.
   - Enable removable media services.
     ```bash
     # svcadm enable rmvolmgr
     # svcadm enable dbus
     # svcadm enable hal
     ```
How to Access Information on Removable Media

1. **Insert the media.**
   The media is mounted after a few seconds.

2. **List the contents of the media.**
   
   ```
   % ls /media
   ```

**Example 3–1** Accessing Information on Removable Media

This example shows how to access information on a USB memory stick.

```
$ ls /media/usb-name
```

This example shows how to access information on a DVD.

```
$ ls /media
sol_10.811_sparc cdrom
```

How to Copy Information From Removable Media

You can access files and directories on removable media as with any other file system. The only significant restrictions are related to ownership and permissions.

For instance, if you copy a file from a DVD into your file system, you are the owner. However, you won’t have write permissions because the file on the DVD never had them. You must change the permissions yourself.

1. **Ensure that the media is mounted.**

   ```
   $ ls /media
   ```

   The `ls` command displays the contents of a mounted media. If no contents are displayed, see “How to Access Information on Removable Media” on page 38.

2. **(Optional) Copy the files or directories.**

   For example, for a DVD, you would do the following:

   ```
   $ cp /media/sol_sparc/Solaris_11/file .
   $ ls -l
   -rwxr-xr-x 1 pmorph gelfs 64065 Aug 2 2010 file
   ```
How to Determine If Removable Media Is Still in Use

1. Become an administrator.

2. Identify the processes that are accessing the media.
   
   # fuser -u /media
   
   The -u displays the user of the media.
   
   For more information, see fuser(1M).

3. (Optional) Kill the process accessing the media.
   
   # fuser -u -k /media
   
   The -k kills the processes accessing the media.

Caution – Killing the processes that are accessing the media should only be used in emergency situations.

4. Verify that the process is gone.
   
   # pgrep process-ID

Example 3–2  Determining If the Media Is Still in Use

The following example shows that the user pmorph, is accessing the /media/sol_10_1008_sparc/Solaris_10/Tools directory.

# fuser -u /media/sol_10_1008_sparc/Solaris_10/Tools
/media/sol_10_1008_sparc/Solaris_10/Tools: 723c(pmorph) 316c(pmorph)

How to Eject Removable Media

1. Ensure that the media is not being used.
   
   Remember, media is “being used” if a shell or an application is accessing any of its files or directories. If you are not sure whether you have found all users of a DVD (for example, a shell hidden behind a desktop tool might be accessing it), use the fuser command. See “How to Determine If Removable Media Is Still in Use” on page 39.

2. Eject the media.
   
   # eject media
   
   For example, for a DVD, you would do the following:

   # eject cdrom
For example, for a USB memory stick, you would do the following:

```
# eject rmdisk0
```

**Tip**—You can view the removable device name with the `eject -l` command.

---

### Accessing Removable Media on a Remote System (Task Map)

The following task map describes the tasks need to access removable media on a remote system.

<table>
<thead>
<tr>
<th>Task</th>
<th>Description</th>
<th>For Instructions</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Make local media available to remote systems.</td>
<td>configure your system to share its media drives to make any media in those drives available to other systems.</td>
<td>&quot;How to Make Local Media Available to Other Systems&quot; on page 40</td>
</tr>
<tr>
<td>2. Access removable media on remote systems.</td>
<td>Access the remote media on the local system.</td>
<td>&quot;How to Access Information on Removable Media&quot; on page 38</td>
</tr>
</tbody>
</table>

---

#### ▼ How to Make Local Media Available to Other Systems

You can configure your system to share its media drives to make any media in those drives available to other systems. One exception is musical CDs. Once your media drives are shared, other systems can access the media they contain simply by mounting them. For instructions, see "How to Access Removable Media on Remote Systems" on page 41.

1. **Become an administrator.**

2. **Confirm that the media is loaded.**

3. **Share the media.**
   
   For example:
   
   ```
   # share -F nfs -o ro /media/sol_10_811_sparc
   ```

4. **Determine whether the NFS server service is running.**
   
   ```
   # svcs *nfs*
   ```
   
   The following output is returned from the `svcs` command if NFS server service is running:
   
   ```
   online 14:28:43 svc:/network/nfs/server:default
   ```

5. **Identify the NFS server status, and select one of the following:**

---
If the NFS server service is running, go to Step 7.
If the NFS server service is not running, go to the next step.

6 Start the NFS server service.

   # svcadm enable network/nfs/server

Verify that the NFS daemons are running.

   For example:

   # svcs -p svc:/network/nfs/server:default

   STATE STIME FMRI
   online Aug_30 svc:/network/nfs/server:default
       Aug_30 319 mountd
       Aug_30 323 nfsd

7 Verify that the media is indeed available to other systems.
   If the media is available, its share configuration is displayed.

   # share
   - /media/sol_10_811_sparc sec=sys,ro ""

Example 3–3 Making Local DVDs Available to Other Systems

The following example shows how to make any local DVD available to other systems on the network.

   # share -F nfs -o ro /media
   # svcs "nfs*"
   # svcadm enable network/nfs/server
   # svcs -p svc:/network/nfs/server:default
   # share
   - /media/sol_10_811_sparc ro ""

How to Access Removable Media on Remote Systems

You can access media on a remote system by manually mounting the media into your file system. Also, the remote system must have shared its media according to the instructions in "How to Make Local Media Available to Other Systems" on page 40.

1 Select an existing directory to serve as the mount point. Or create a mount point.
   $ mkdir /directory

   where /directory is the name of the directory that you create to serve as a mount point for the remote system’s DVD.

2 Find the name of the media you want to mount.
   $ showmount -e system-name
3 As an administrator, mount the media.
   # mount -F nfs -o ro system-name:/media/media-name local-mount-point
   
   system-name: Is the name of the system whose media you will mount.
   media-name: Is the name of the media you want to mount.
   local-mount-point: Is the local directory onto which you will mount the remote media.

4 Log out as an administrator.

5 Verify that the media has been mounted.
   $ ls /media

Example 3–4 Accessing DVDs or CDs on Remote Systems

The following example shows how to automatically access the remote DVD named sol_10_811_sparc from the remote system starbug using autofs.

   $ showmount -e starbug
   export list for starbug:
   /media/sol_10_811_sparc (everyone)
   $ ls /net/starbug/media/
   sol_10_811_sparc
This chapter provides step-by-step instructions for writing and copying data CDs and DVDs and audio CDs with the `cdrw` command.

- “How to Restrict User Access to Removable Media With RBAC” on page 46
- “How to Identify a CD or DVD Writer” on page 47
- “How to Check the CD or DVD Media” on page 47
- “How to Create an ISO 9660 File System for a Data CD or DVD” on page 48
- “How to Create a Multi-Session Data CD” on page 49
- “How to Create an Audio CD” on page 51
- “How to Extract an Audio Track on a CD” on page 52
- “How to Copy a CD” on page 53
- “How to Erase CD-RW Media” on page 54

**Working With Audio CDs and Data CDs and DVDs**

You can use the `cdrw` command to write file systems for CDs and DVDs in ISO 9660 format with Rock Ridge or Joliet extensions on CD-R, CD-RW, DVD-RW, or DVD+RW media devices.

You can use the `cdrw` command to perform the following tasks:

- Create data CDs and DVDs.
- Create audio CDs.
- Extract audio data from an audio CD.
- Copy CDs and DVDs.
- Erase CD-RW media.

The `cdrw` command is available in the following releases:

- Oracle Solaris 10 releases
- Oracle Solaris 11 release, `media/cdrw` package
For information on recommended CD-R or CD-RW devices, go to:


CD/DVD Media Commonly Used Terms

This section defines commonly used terms related to CD/DVD media.

<table>
<thead>
<tr>
<th>Term</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CD-R</td>
<td>CD read media that can be written once and after that, can only be read from.</td>
</tr>
<tr>
<td>CD-RW</td>
<td>CD rewritable media that can be written to and erased. CD-RW media can only be read by CD-RW devices.</td>
</tr>
<tr>
<td>DVD-R</td>
<td>Digital video disk (recordable) that can be written once and after that, can only be read from. These devices have much larger capacity than CD-R media.</td>
</tr>
<tr>
<td>DVD+R</td>
<td>Digital video disk (recordable) that can be written once and after that, can only be read from. DVD+R devices have more complete error management system than DVD-R, which allows for more accurate burning to media, independent of the quality of the media.</td>
</tr>
<tr>
<td>DVD-RW</td>
<td>Digital video disk (rewritable) with storage capacity equal to a DVD-R. This media can be re-recorded by first erasing the entire disk.</td>
</tr>
<tr>
<td>DVD+RW</td>
<td>Digital video disk (random-access rewritable) with storage capacity equal to a DVD+R. This medium allows overwriting of individual blocks without erasing the entire disk.</td>
</tr>
<tr>
<td>DVD-RAM</td>
<td>Digital video disk (random access memory, rewritable) with circular rather than spiral tracks and hard sectoring.</td>
</tr>
<tr>
<td>ISO 9660</td>
<td>ISO, an acronym for Industry Standards Organization, is an organization that sets standards for computer storage formats. An ISO 9660 file system is a standard CD or DVD file system that enables you to read the same CD or DVD on any major computer platform. The standard, issued in 1988, was written by an industry group named High Sierra, named after the High Sierra Hotel in Nevada. Almost all computers with CD or DVD drives can read files from an ISO 9660 file system.</td>
</tr>
</tbody>
</table>
### Joliet extensions
Adds Windows file system information.

### Rock Ridge extensions
Adds UNIX file system information. (Rock Ridge is named after the town in the movie Blazing Saddles.)

**Note** – These extensions are not exclusive. You can specify both mkisofs -R and -j options for compatibility with both systems. (See mkisofs(1M) for details.)

### MMC-compliant recorder
Acronym for Multi Media Command, which means these recorders comply with a common command set. Programs that can write to one MMC-compliant recorder should be able to write to all other recorders.

### Red Book CDDA
Acronym for Compact Disc Digital Audio, which is an industry standard method for storing digital audio on compact discs. Also known by the term “Red Book” format. The official industry specification calls for one or more audio files sampled in 16-bit stereo sound at a sampling rate of 44.1 kilohertz (kHz).

Commonly used terms when writing to CD media are:

<table>
<thead>
<tr>
<th>Term</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>blanking</td>
<td>The process of erasing data from the CD-RW media.</td>
</tr>
<tr>
<td>mkisofs</td>
<td>The command to create ISO file system on a CD.</td>
</tr>
<tr>
<td>session</td>
<td>A complete track with lead-in and lead-out information.</td>
</tr>
<tr>
<td>track</td>
<td>A complete data or audio unit.</td>
</tr>
</tbody>
</table>

### Writing CD and DVD Data and Audio CDs

The process of writing to a CD or DVD cannot be interrupted and needs a constant stream of data. Consider using the cdrw - S option to simulate writing to the media to verify that the system can provide data at a sufficient rate for writing to the CD or DVD.

Write errors can be caused by one of the following problems:

- The media cannot handle the drive speed. For example, some media are only certified for 2x or 4x speeds.
- The system is running too many heavy processes that are starving the writing process.
- Network congestion is causing delays in reading the image, and the image is on a remote system.
- The source drive is slower than the destination drive.
If any of these problems occur, you can lower the writing speed of the device by using the \texttt{cddw -p} option.

For example, the following command shows how to simulate writing at 4x speed:

$ \texttt{cddw -iS -p 4 image.iso}$

\textbf{Note} – CD-R, CD-RW (not MRW formatted), DVD-R, and DVD-RW media support simulation mode (-S), but DVD-RAM, DVD+R, DVD+RW, and any MRW-formatted media and some others do not support simulation mode. The following message is displayed if simulation mode is not supported:

\texttt{Media does not support simulated writing}

For more information about media types, see ”CD/DVD Media Commonly Used Terms” on page 44.

For more information, see \texttt{cddw(1)}.

\section*{Restricting User Access to Removable Media With RBAC}

By default, all users can access removable media. However, you can restrict user access to removable media by setting up a role through role-based access control (RBAC). Access to removable media is restricted by assigning the role to a limited set of users.

For a discussion of using roles, see “Role-Based Access Control (Overview)” in \textit{Oracle Solaris Administration: Security Services}.

\subsection*{How to Restrict User Access to Removable Media With RBAC}

\begin{enumerate}
\item \textbf{Become an administrator.}

For more information, see ”How to Use Your Assigned Administrative Rights” in \textit{Oracle Solaris Administration: Security Services}.

\item \textbf{Set up a role that includes the Device Management rights.}

For more information, see Chapter 9, ”Using Role-Based Access Control (Tasks),” in \textit{Oracle Solaris Administration: Security Services}.

\texttt{# roleadd -m -d /export/home/muser -c "mediauser role" -A solaris.device.cddw -P All muser}
\end{enumerate}
3 Add users who need to use the `cdrw` command to the newly created role.
   # usermod -R muser joe

4 Comment the following line in the `/etc/security/policy.conf` file:
   AUTHS_GRANTED=solaris.device.cdrw
   If you do not do this step, all users still have access to the `cdrw` command, not just the members of the device management role.
   After this file is modified, the device management role members are the only users who can use the `cdrw` command. Everyone else is denied access with the following message:
   Authorization failed, Cannot access disks.

▼ How to Identify a CD or DVD Writer

1 Identify the CD or DVD writers on the system.
   For example:
   $ cdrw -l
   Looking for CD devices...
<table>
<thead>
<tr>
<th>Node</th>
<th>Connected Device</th>
<th>Device type</th>
</tr>
</thead>
<tbody>
<tr>
<td>cdrom0</td>
<td>YAMAHA CRW8824S 1.0d</td>
<td>CD Reader/Writer</td>
</tr>
</tbody>
</table>

2 Identify a specific CD or DVD writer.
   For example:
   $ cdrw -a filename.wav -d cdrom2

3 Identify whether the media is blank or whether a table of contents exists on the media.
   For example:
   $ cdrw -M
   Device : YAMAHA CRW8824S
   Firmware : Rev. 1.00 (26/04/00)
   Media is blank

▼ How to Check the CD or DVD Media

The `cdrw` command works with or without removable media services running. For more information about disabling or enabling removable media services, see “How to Disable or Enable Removable Media Services” on page 37.

1 Insert a CD or DVD into the drive.
   The CD or DVD can be any CD or DVD that the drive can read.
Check that the drive is connected properly by listing the drive.

```bash
$ cdrw -l
Looking for CD devices...
```

<table>
<thead>
<tr>
<th>Node</th>
<th>Connected Device</th>
<th>Device type</th>
</tr>
</thead>
<tbody>
<tr>
<td>cdrom1</td>
<td>YAMAHA CRW8824S 1.0d</td>
<td>CD Reader/Writer</td>
</tr>
</tbody>
</table>

(Optional) If you do not see the drive in the list, select one of the following so that the system recognizes the drive.

- Add the drive without rebooting the system

```bash
# drvconfig
# disks
```

Then restart removable media services.

**Creating a Data CD or DVD**

Prepare the data first by using the `mkisofs` command to convert the file and file information into the High Sierra format used on CDs or DVDs.

**How to Create an ISO 9660 File System for a Data CD or DVD**

1. Insert a blank CD or DVD into the drive.

2. Create the ISO 9660 file system on the new CD or DVD.

   ```bash
   $ mkisofs -r /pathname > cd-file-system
   ```

   - `-r`     Creates Rock Ridge information and resets file ownerships to zero.
   - `/pathname` Identifies the path name used to create the ISO 9660 file system.
   - `> cd-file-system` Identifies the name of the file system to be put on the CD or DVD.

3. Copy the file system onto the CD or DVD.

   ```bash
   $ cdrw -i cd-file-system
   ```

   The `-i cd-file-system` specifies the image file for creating a data CD or DVD.

**Example 4–1 Creating an ISO 9660 File System for a Data CD or DVD**

The following example shows how to create an ISO 9660 file system for a data CD or DVD.
How to Create a Multi-Session Data CD

This procedure describes how to put more than one session on a CD. This procedure includes an example of copying the infoA and infoB directories onto the CD.

1. Create the file system for the first CD session.

   ```bash
   $ mkisofs -o infoA -r -V my_infoA /data/infoA
   Total translation table size: 0
   Total rockridge attributes bytes: 24507
   Total directory bytes: 34816
   Path table size(bytes): 98
   Max brk space used 2e000
   8929 extents written (17 Mb)
   -o infoA Identiﬁes the name of the ISO ﬁle system.
   -r Creates Rock Ridge information and resets ﬁle ownerships to zero.
   -V my_infoA Identiﬁes a volume label to be used as the mount point by removable media services.
   /data/infoA Identiﬁes the ISO image directory to create.
   ```

2. Copy the ISO ﬁle system for the ﬁrst session onto the CD.

   ```bash
   $ cdrw -iO infoA
   Initializing device...done.
   Writing track 1...done.
   Finalizing (Can take several minutes)...done.
   -i infoA Identifies the name of the image file to write to the CD.
   -O Keeps the CD open for writing.
   ```

3. Re-insert the CD after it is ejected.
4 Identify the path name of the CD media to include in the next write session.

```bash
$ eject -n
```

```
cdrom0 -> /vol/dev/rdsk/c2t4d0/my_infoA
```

Note the `/vol/dev/...` path name.

5 Identify the next writeable address on the CD to write the next session.

```bash
% cdrw -M /cdrom
```

Device: YAMAHA CRW8424S

Firmware: Rev. 1.0d (06/10/99)

<table>
<thead>
<tr>
<th>Track No.</th>
<th>Type</th>
<th>Start address</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Audio</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>Audio</td>
<td>33057</td>
</tr>
<tr>
<td>3</td>
<td>Data</td>
<td>60887</td>
</tr>
<tr>
<td>4</td>
<td>Data</td>
<td>68087</td>
</tr>
<tr>
<td>5</td>
<td>Data</td>
<td>75287</td>
</tr>
<tr>
<td>Leadout</td>
<td>Data</td>
<td>84218</td>
</tr>
</tbody>
</table>

Last session start address: 75287
Next writable address: 91118

Note the address in the *Next writable address* output so that you can provide this address when you write the next session.

6 Create the next ISO file system for the next CD session, and write it onto the CD.

```bash
$ mkisofs -o infoB -r -C 0,91118 -M /vol/dev/rdsk/c2t4d0/my_infoA /data/infoB
```

Total translation table size: 0
Total rockridge attributes bytes: 16602
Total directory bytes: 22528
Path table size(bytes): 86
Max brk space used 20000
97196 extents written (189 Mb)

-o infoB Identifies the name of the ISO file system.
-r Creates Rock Ridge information and resets file ownerships to zero.
-C 0,91118 Identifies the starting address of the first session and the next writable address.
-M /vol/dev/rdsk/c2t4d0/my_infoA Specifies the path of the existing ISO image to be merged.
/data/infoB Identifies the ISO image directory to create.
Creating an Audio CD

You can use the `cdrw` command to create audio CDs from individual audio tracks or from `.au` and `.wav` files.

The supported audio formats are described in the following table:

<table>
<thead>
<tr>
<th>Format</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>sun</td>
<td>Oracle .au file with data in Red Book CDDA format</td>
</tr>
<tr>
<td>wav</td>
<td>RIFF (.wav) file with data in Red Book CDDA format</td>
</tr>
<tr>
<td>cda</td>
<td>.cda file with raw CD audio data, which is 16-bit PCM stereo at 44.1 kHz sample rate in little-endian byte order</td>
</tr>
<tr>
<td>aur</td>
<td>.aur files with raw CD data in big-endian byte order</td>
</tr>
</tbody>
</table>

If no audio format is specified, the `cdrw` command tries to determine the audio file format based on the file extension. The case of the characters in the extension is ignored.

**How to Create an Audio CD**

This procedure describes how to copy audio files onto a CD.

1. **Insert a blank CD into the CD-RW drive.**
2. **Change to the directory that contains the audio files.**
   ```bash
   $ cd /myaudiodir
   ```
3. **Copy the audio files onto the CD.**
   ```bash
   $ cdrw -a track1.wav track2.wav track3.wav
   ```
   The `-a` option creates an audio CD.

**Example 4–2 Creating an Audio CD**

The following example shows how to create an audio CD.

```bash
$ cdrw -a bark.wav chirp.au meow.wav
Initializing device...done.
Writing track 1...done.
done.
Writing track 2...done.
Writing track 3...done.
done.
Finalizing (Can take several minutes)...done.
```
The following example shows how to create a multisession audio CD. The CD is ejected after the first session is written. You would need to re-insert the CD before the next writing session.

```
$ cdrw -aO groucho.wav chico.au harpo.wav
Initializing device...done.
Writing track 1...done.
done.
Writing track 2...done.
Writing track 3...done.
done.
Finalizing (Can take several minutes)...done.
<Re-insert CD>
$ cdrw -a zeppo.au
Initializing device...done.
Writing track 1...done.
done.
Finalizing (Can take several minutes)...done.
```

### How to Extract an Audio Track on a CD

Use the following procedure to extract an audio track from a CD and copy the audio track to a new CD.

If you don’t use the `cdrw -T` option to specify the audio file type, the `cdrw` command uses the filename extension to determine the audio file type. For example, the `cdrw` command detects that this file is a `.wav` file.

```
$ cdrw -x 1 testme.wav
```

1. **Insert an audio CD into the CD-RW drive.**

2. **Extract an audio track.**
   - `$ cdrw -x -T audio-type 1 audio-file`
   - `-x` Extracts audio data from an audio CD.
   - `audio-type` Identifies the type of audio file to be extracted. Supported audio types are `sun`, `wav`, `cda`, or `aur`.
   - `audio-file` Identifies the audio track to be extracted.

3. **Copy the track to a new CD.**
   - `$ cdrw -a audio-file`

**Example 4–3 Extracting and Creating Audio CDs**

The following example shows how to extract the first track from an audio CD and name the file `song1.wav`. 
This example shows how to copy a track to an audio CD.

```
cdrw -x -T wav 1 song1.wav
```

Extracting audio from track 1...done.

This example shows how to copy a track to an audio CD.

```
cdrw -a song1.wav
```

Initializing device...done.
Writing track 1...done.
Finalizing (Can take several minutes)...done.

## How to Copy a CD

This procedure describes how to extract all the tracks from an audio CD into a directory and then copy all of them onto a blank CD.

**Note** – By default, the `cdrw` command copies the CD into the `/tmp` directory. The copying might require up to 700 MB of free space. If there is insufficient space in the `/tmp` directory for copying the CD, use the `-m` option to specify an alternate directory.

1. **Insert an audio CD into a CD-RW drive.**

2. **Create a directory for the audio files.**
   ```bash
   mkdir /music_dir
   ```

3. **Extract the tracks from the audio CD.**
   ```bash
   cdrw -c -m music_dir
   ```
   An Extracting audio ... message is display for each track.
   The CD is ejected when all the tracks are extracted.

4. **Insert a blank CD and press Return.**
   After the tracks are extracted, the audio CD is ejected. You are prompted to insert a blank CD.

**Example 4–4**  
**Copying a CD**

This example shows how to copy one CD to another CD. You must have two CD-RW devices to do this task.

```
cdrw -c -s cdrom0 -d cdrom1
```
How to Erase CD-RW Media

You have to erase existing CD-RW data before the CD can be rewritten.

- Erase the entire media or just the last session on the CD by selecting one of the following:
  - Erase the last session only.
    $ cdrw -d cdrom0 -b session
    Erasing just the last session with the `-b session` option is faster than erasing the entire media with the `-b all` option. You can use the `-b session` option even if you used the `cdrw` command to create a data or audio CD in just one session.
  - Erase the entire media.
    $ cdrw -d cdrom0 -b all
Managing Devices (Overview/Tasks)

This chapter provides overview information and step-by-step instructions for managing peripheral devices, such as disks, DVD drives, and tape devices, in the Oracle Solaris release.

This is a list of the overview information in this chapter.

- "What’s New in Device Management?” on page 56
- “Where to Find Additional Device Management Tasks” on page 59
- “About Device Drivers” on page 60
- “Automatic Configuration of Devices” on page 61
- “Displaying Device Configuration Information” on page 63
- “Resolving Faulty Devices” on page 67
- “Accessing Devices” on page 71

This is a list of the step-by-step instructions in this chapter.

- "How to Customize a Driver Configuration” on page 61
- “How to Display System Configuration Information” on page 64
- “How to Resolve a Faulty Device” on page 68
- “How to Add a Device Driver” on page 70
- “How to Add a Peripheral Device” on page 69

Device management in the Oracle Solaris release usually involves adding and removing peripheral devices from systems, possibly adding a third-party device driver to support a device, and displaying system configuration information.
What's New in Device Management?

This section provides information about new device management features in the Oracle Solaris release.

- "Customizing Driver Configuration" on page 56
- "Solaris PCI Resource Manager" on page 56
- "New InfiniBand Administration Features" on page 57
- "New InfiniBand Diagnostic Tools and Commands" on page 57
- "New Ethernet Over InfiniBand Devices" on page 57
- "New Hot Plugging Features" on page 57
- "Device Naming Enhancements" on page 58
- "Support for PCI Express (PCIe)" on page 58

Customizing Driver Configuration

Oracle Solaris 11: In this Solaris release, you can provide a supplemental driver configuration file, driver.conf, in the /etc/driver/driv directory. In previous Solaris release, you had to modify the vendor’s driver.conf files directly.

In this release, the system automatically merges the driver vendor’s /kernel or /platform driver.conf files with the customized /etc/driver/driv driver.conf files so that the driver sees both sets of property values.

You can use the prtconf -u command to display the original vendor and customized property values. Separating vendor driver.conf files and locally customized driver.conf files allows the system to be upgraded with new vendor driver.conf files without overwriting your administrative customizations.

If you have existing modifications in a driver configuration file in the /kernel/driv directory from a previous Solaris release or system, you should copy them over to the /etc/driver/driv directory.

For information about creating an /etc/driver/driv/driver.conf file, see “How to Customize a Driver Configuration” on page 61, driver.conf(4), and driver(4).

Solaris PCI Resource Manager

Oracle Solaris 11: In this release, a PCI resource manager (PCIRM) is provided to rebalance PCI resources to ensure that when a systems boot, enough resources are assigned to support PCI Express (PCIe) devices. For example, Single Root I/O Virtualization (SR-IOV) devices require more PCI memory resources for their virtual functions. Most of the firmware today was designed prior to SR-IOV standards, and so, does not recognize the requirement to reserve resources for virtual devices. The resource rebalance process happens automatically at boot time and requires no administration.
New InfiniBand Administration Features

**Oracle Solaris 11:** Administration of IPoIB has changed from the earlier releases. In this release, you can create, delete, and view IPoIB datalinks information by using `dladm` sub-commands `create-part`, `delete-part`, `view-part`, and `show-ib`. These new `dladm` sub-commands provide an easier way to manage IPoIB datalinks and also helps to view and troubleshoot some IB issues. For example, the `show-ib` sub-command shows how many HCAs are present in the system and the corresponding HCA GUIDs. In the new administration model, there are two types of IP over IB datalinks.

- Physical data link - similar to a regular network interface card (NIC) data link
- IB partition data link - similar to a virtual NIC (VNIC)

On a newly installed system, physical links are created automatically by default. Then, you can use the `dladm` command to create an IB partition link over the IB physical link. Unlike a NIC data link, an IB physical link cannot be configured, such as plumbed or assigned an IP address for using the link.

For more information on using the `dladm` subcommands for managing IB components, see “Administering IPoIB Devices (dladm)” on page 173.

New InfiniBand Diagnostic Tools and Commands

**Oracle Solaris 11:** The `system/io/infiniband/open-fabrics` package provides a subset of the commands and utilities from the Open Fabrics Alliance (OFA) Open Fabrics Enterprise Distribution (OFED) version 1.5.3. The collection of tools provide the ability to list and query IB devices, diagnose and troubleshoot an IB Fabric, and measure IB performance with a collection of IB user verb micro-benchmarks. For more information, see “Monitoring and Troubleshooting IB Devices” on page 175.

New Ethernet Over InfiniBand Devices

**Oracle Solaris 11:** The Ethernet over InfiniBand (eoib) driver is a multi-threaded, loadable, cloneable, GLD-based STREAMS driver supporting the Data Link Provider Interface, (DLPI) and overall IB ports on a system that are connected to a Oracle Network QDR InfiniBand Gateway switch.

New Hot Plugging Features

**Oracle Solaris 11:** In this release, the `hotplug` command is available to manage hot pluggable connections on PCI Express (PCIe) and PCI SHPC (Standard Hot Plug Controller) devices. This feature is not supported on other bus types, such as USB and SCSI.
You would still use the `cfgadm` to manage hot pluggable USB and SCSI devices as in previous Solaris releases. The benefit of using the hotplug features in this release is that in addition to enable and disable operations, the `hotplug` command provides offline and online capabilities for your supported PCI devices.

For more information, see “PCle Hot-Plugging With the (hotplug) Command” on page 81.

Device Naming Enhancements

**Oracle Solaris 11**: The `/dev` name space supports multiple file system instances as needed. A global instance of the `/dev` file system is created automatically when the system is booted. Subsequent `/dev` instances are created and mounted when needed, such as when devices are added to a non-global zone. When a non-global zone is shutdown, the available `/dev` instance is unmounted and unavailable.

In addition, device configuration is improved in the following ways:

- **Reconfiguration boot is eliminated** – In previous Solaris releases, a reconfiguration boot was needed if you connected a device to a system that is powered off.
  
  In this release, performing a reconfiguration boot is unnecessary when attaching devices to a system that is powered off. Newly attached devices are automatically recognized and the appropriate device links are created when the system is rebooted.
  
  For more information, see `dev(7FS)`.

- **Zone device support is simplified** – As described above, device support for Oracle Solaris zones is enhanced by providing specific instances of the `/dev` directory for non-global zones. In addition, zones are no longer dependent upon the `devfsadm` daemon for reconfiguration of devices within a zone.

- **Pseudo device creation is improved** – In this release, the content of the `/dev/pts` directory is created on demand in the global `/dev` name space as well as a `/dev` instance when needed in a non-global zone. In addition, the pts links are only visible in the global zone or the non-global zone from which they are allocated.
  
  For more information, see `grantpt(3C)`.

For more information about device configuration, see “Managing Devices in the Oracle Solaris OS” on page 60.

Support for PCI Express (PCle)

**Oracle Solaris 11**: This Oracle Solaris release provides support for the PCI Express (PCle) interconnect, which is designed to connect peripheral devices to desktop, enterprise, mobile, communication, and embedded applications, on both SPARC and x86 systems.
The PCIe interconnect is an industry-standard, high-performance, serial I/O bus. For details on PCIe technology, go to the following site:

http://www.pcisig.com/home

The PCIe software provides the following features in this Oracle Solaris release:

- Support for extended PCIe configuration space
- Support for PCIe baseline error handling and MSI interrupts
- Modified IEEE-1275 properties for PCIe devices
- PCIe hotplug support (both native and ACPI-based) by enhancing the `cfgadm_pci` component of the `cfgadm` command
- ATTN Button usage based PCIe peripheral autoconfiguration

The administrative model for hotplugging PCIe peripherals is the same as for PCI peripherals, which uses the `cfgadm` command.

Check your hardware platform guide to ensure that PCIe and PCIe hotplug support is provided on your system. In addition, carefully review the instructions for physically inserting or removing adapters on your system and the semantics of device auto-configuration, if applicable.

For information about using the `cfgadm` command with PCIe peripherals, see "PCI or PCIe Hot-Plugging With the `cfgadm` Command (Task Map)" on page 93.

Where to Find Additional Device Management Tasks

The following table describes where to find step-by-step instructions for hot-plugging devices and adding serial devices, such as printers and modems.

<table>
<thead>
<tr>
<th>Device Management Task</th>
<th>For More Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Add a disk that is not hot-pluggable</td>
<td>Chapter 12, &quot;SPARC: Setting Up Disks (Tasks),&quot; or Chapter 13, &quot;x86: Setting Up Disks (Tasks)&quot;</td>
</tr>
<tr>
<td>Hot-plug a SCSI or PCI device.</td>
<td>&quot;SCSI Hot-Plugging With the <code>cfgadm</code> Command&quot; on page 84 or</td>
</tr>
<tr>
<td></td>
<td>&quot;PCI or PCIe Hot-Plugging With the <code>cfgadm</code> Command&quot; on page 94</td>
</tr>
<tr>
<td>Hot-plug a USB device.</td>
<td>&quot;Using USB Mass Storage Devices (Task Map)&quot; on page 128</td>
</tr>
<tr>
<td>Add a modem.</td>
<td>Chapter 16, &quot;Managing the System Console, Terminal Devices, and Power Services (Tasks),&quot; in Oracle Solaris Administration: Common Tasks</td>
</tr>
</tbody>
</table>
Managing Devices in the Oracle Solaris OS

The following sections provide overview information about features that manage devices in the Oracle Solaris OS. For information about accessing devices, see "Accessing Devices" on page 71.

Identifying Device Support

You can use the device detection tool to identify whether your x86 hardware is supported in this Oracle Solaris release. For more information, go to the following site:

http://www.oracle.com/webfolder/technetwork/hcl/hcts/device_detect.jsp

About Device Drivers

A computer typically uses a wide range of peripheral devices and mass-storage devices. Your system, for example, probably has a disk drive, a keyboard and a mouse, and some kind of magnetic backup medium. Other commonly used devices include the following:

- DVD drives
- Printers and plotters
- Light pens
- Touch-sensitive screens
- Digitizers
- Tablet-and-stylus pairs

The Oracle Solaris software does not directly communicate with all these devices. Each type of device requires different data formats, protocols, and transmission rates.

A *device driver* is a low-level program that allows the operating system to communicate with a specific piece of hardware. The driver serves as the operating system’s “interpreter” for that piece of hardware.
How to Customize a Driver Configuration

In the Oracle Solaris 11 release, driver customizations are made in the /etc/driver/drv directory rather than in the /kernel directory as in previous releases. This improvement means that your driver customizations are not overwritten when the system is upgraded. The files in the /etc/driver/drv directory are preserved during the upgrade.

Customizing a driver configuration usually means that a per-device parameter or global property that impacts all devices is added or modified.

1 Become an administrator.

2 Copy the original vendor supplied driver.conf file to the /etc/driver/drv directory. For example:

   # cp /kernel/drv/sd.conf /etc/driver/drv/sd.conf

3 Modify the parameter entry and save the file.

   For example, the sd.conf includes the following entry for sd device at target 0, lun 0:

   name="sd" class="scsi" target=0 lun=0;

   To add the retries parameter for this device, modify the existing entry as follows:

   name="sd" class="scsi" target=0 lun=0 retries=4;

4 Display the customized property value. For example:

   # prtconf -u
   sd, instance #1
   Admin properties:
   name='retries' type=int items=1
   value=00000004

Automatic Configuration of Devices

The kernel consists of a small generic core with a platform-specific component and a set of modules. The kernel is configured automatically in the Oracle Solaris release.

A kernel module is a hardware or software component that is used to perform a specific task on the system. An example of a loadable kernel module is a device driver that is loaded when the device is accessed.

The platform-independent kernel is /kernel/genunix. The platform-specific component is /platform/'uname -m'/kernel/unix.

The kernel modules are described in the following table.
The system determines what devices are attached to it at boot time. Then, the kernel configures itself dynamically, loading needed modules into memory. At this time, device drivers are loaded when devices, such as disk devices and tape devices, are accessed. This process is called autoconfiguration because all kernel modules are loaded automatically when they are needed.

You can customize the way in which kernel modules are loaded by modifying the /etc/system file. For instructions on modifying this file, see system(4).

**Features and Benefits of Autoconfiguration**

The benefits of autoconfiguration are as follows:

- Main memory is used more efficiently because modules are loaded when needed.
- There is no need to reconfigure the kernel when new devices are added to the system.
- Drivers can be loaded and tested without having to rebuild the kernel and reboot the system.

Autoconfiguration is used when you add a new device (and driver) to the system. In previous releases, it was necessary to perform a reconfiguration boot if you added a device to a system that is shutdown. Device configuration enhancements make a reconfiguration boot unnecessary when a device is added to a system that is shutdown.

You can add, remove, or replace devices in the Oracle Solaris OS while the system is still running, if the system components support hot-plugging. For information about hot-plugging devices, see Chapter 6, “Dynamically Configuring Devices (Tasks).”

**What You Need for Unsupported Devices**

Device drivers needed to support a wide range of standard devices are included in the Oracle Solaris release. These drivers can be found in the /kernel/drv and /platform/uname -m/kernel/drv directories.

However, if you have purchased an unsupported device, the manufacturer should provide the software that is needed for the device to be properly installed, maintained, and administered.
At a minimum, this software includes a device driver and its associated configuration (.conf) file. The .conf files reside in the drv directories. This software might also include custom maintenance and administrative utilities because the device might be incompatible with Oracle Solaris utilities.

For more information about what you need for unsupported devices, contact your device manufacturer.

**Displaying Device Configuration Information**

Three commands are used to display system and device configuration information.

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
<th>Man Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>prtconf</td>
<td>Displays system configuration information, including the total amount of memory and the device configuration as described by the system's device hierarchy. The output displayed by this command depends upon the type of system.</td>
<td>prtconf(1M)</td>
</tr>
<tr>
<td>sysdef</td>
<td>Displays device configuration information, including system hardware, pseudo devices, loadable modules, and selected kernel parameters.</td>
<td>sysdef(1M)</td>
</tr>
<tr>
<td>dmesg</td>
<td>Displays system diagnostic messages as well as a list of devices attached to the system since the last reboot.</td>
<td>dmesg(1M)</td>
</tr>
</tbody>
</table>

For information on the device names that are used to identify devices on the system, see “Device Naming Conventions” on page 72.

**Driver not attached Message**  
The following driver-related message might be displayed by the prtconf and sysdef commands:

`device, instance #number (driver not attached)`

This message does not always mean that a driver is unavailable for this device. This message means that no driver is currently attached to the device instance because no device exists at this node or the device is not in use. Drivers are loaded automatically when the device is accessed. They are unloaded when the device is not in use.

**In-Use Device Error Checking**

The following utilities detect when a specified device is in use:

- dumpadm
- format
These enhancements mean that the above utilities might detect some of the following usage scenarios:

- Device is part of a ZFS storage pool
- Device is a dump or swap device
- Mounted file system or an entry for the device exists in the /etc/vfstab file

For example, if you attempt to use the `format` utility to access an active device, you will see a message similar to the following:

```
# format
Searching for disks...done
AVAILABLE DISK SELECTIONS:
   0. c1t0d0 <FUJITSU-MAY2073RCSUN72G-0401 cyl 8921 alt 2 hd 255 sec 63>
       /pci@0,0/pci1022,7450@2/pci1000,30603(sd0,0]
       /dev/chassis/SYS/HD0/disk
   1. c1t1d0 <FUJITSU-MAY2073RCSUN72G-0401-68.37GB>
       /pci@0,0/pci1022,745002/pci1000,30603(sd0,0]
       /dev/chassis/SYS/HD1/disk
Specify disk (enter its number): 0
selecting c1t0d0
[disk formatted]
/dev/dsk/c1t0d0s0 is part of active ZFS pool rpool. Please see zpool(1M).
```

**How to Display System Configuration Information**

Use the output of the `prtconf` and `sysdef` commands to identify which disk, tape, and DVD devices are connected to the system. The output of these commands displays the `driver not attached` messages next to the device instances. Because these devices are always being monitored by some system process, the `driver not attached` message is usually a good indication that no device exists at that device instance.

Use the `sysdef` command to display system configuration information that include pseudo devices, loadable modules, and selected kernel parameters.

- **Display system and device configuration information.**
  
  Display all the devices connected to a system.

  For example, the following `prtconf -v` output on a SPARC system identifies the disk devices connected to the system. The detailed disk information is described in the Device Minor Nodes section.
Displaying Information about One Specific Device Connected to the System

For example, the following `prtconf` output on a SPARC system displays the `sd` instance number for `/dev/dsk/c4t2d0s0`.

```
# prtconf -v /dev/dsk/c4t2d0s0 | grep instance
ds, instance #5
```

Display only the devices that are attached to the system.

```
# prtconf | grep -v not
```

Display device usage information.

For example, the following `fuser` command displays which processes are accessing the `/dev/console` device.

```
# fuser -d /dev/console
/dev/console: 5742o 2269o 20322o 858o
```

### Example 5–1 Displaying System Configuration Information

The following `prtconf` output is displayed on an x86 based system.

```
# prtconf
System Configuration: Oracle Corporation i86pc
Memory size: 8192 Megabytes
System Peripherals (Software Nodes):
```

The following sysdef output is displayed from an x86 based system.

```
# sysdef
* Hostid
  * 29f10b4d
  *
  * i86pc Configuration
  *
  *
  * Devices
  *
  +boot (driver not attached)
```
Resolving Faulty Devices

A device retirement mechanism isolates a device as faulty by the fault management framework (FMA). This feature allows faulty devices to be safely and automatically inactivated to avoid data loss, data corruption, or panics and system downtime. The retirement process is done safely, taking into account the stability of the system after the device has been retired.

Critical devices are never retired. If you need to manually replace a retired device, use the fmadm repair command after the device replacement so that system knows that the device is replaced, in addition to the manual replacement steps.

For more information, see fmadm(1M).

A general message regarding device retirement is displayed on the console and written to the /var/adm/messages file so that you aware of a retired device. For example:

Aug 9 18:14 starbug genunix: [ID 751201 kern.notice] NOTICE: One or more I/O devices have been retired

You can use the prtconf command to identify specific retired devices. For example:

# prtconf

pci, instance #2
  scsi, instance #0
    disk (driver not attached)
How to Resolve a Faulty Device

Use the steps below to resolve a faulty device or a device that has been retired.

You can also review ZFS device problem or failure information by using the `zpool status` or the `fmdump` command. For ZFS device problem or failure information, see Chapter 11, “Oracle Solaris ZFS Troubleshooting and Pool Recovery,” in Oracle Solaris Administration: ZFS File Systems.

1 Identify the faulted device with the `fmadm faulty` command.

```
# fmadm faulty
```

<table>
<thead>
<tr>
<th>TIME</th>
<th>EVENT-ID</th>
<th>MSG-ID</th>
<th>SEVERITY</th>
</tr>
</thead>
<tbody>
<tr>
<td>May 06 03:38:06</td>
<td>0376b4b6-fce7-c0f0-ffd9-a0a685376284</td>
<td>ZFS-8000-D3</td>
<td>Major</td>
</tr>
</tbody>
</table>

**Host**: neo-2  
**Platform**: Sun-Fire-X4140  
**Chassis_id**: 0904QAD02C  
**Product_sn**:  

**Fault class**: fault.fs.zfs.device  

**Affects**: zfs://pool=tank/vdev=c26c72a8ffcf889  

**Problem in**: zfs://pool=tank/vdev=c26c72a8ffcf889  

**Description**: A ZFS device failed. Refer to http://sun.com/msg/ZFS-8000-D3 for more information.

**Response**: No automated response will occur.

**Impact**: Fault tolerance of the pool may be compromised.

**Action**: Run ‘zpool status -x’ and replace the bad device.

2 Replace the faulty or retired device or clear the device error.

After a faulty device is replaced in a ZFS storage pool, clear the device error. For example:

```
# zpool clear rpool c4t0d0s0
```
If an intermittent device error occurred but the device was not replaced, you can attempt to clear the error as shown above.

3 Clear the FMA fault.
   # fmadm repair 0376b4b6-fce7-c0f0-ffd9-a0a685376284

4 Confirm that the fault is cleared.
   # fmadm faulty

Adding a Peripheral Device to a System

Adding a new peripheral device that is not-pluggable usually involves the following:

- Shutting down the system
- Connecting the device to the system
- Rebooting the system

Use "How to Add a Peripheral Device" on page 69 to add the following devices that are not hot-pluggable to a system:

- DVD drive
- Secondary disk drive
- Tape drive
- SBUS card

In some cases, you might have to add a third-party device driver to support the new device.

For information on hot-plugging devices, see Chapter 6, “Dynamically Configuring Devices (Tasks).”

▼ How to Add a Peripheral Device

1 Become an administrator.
   For more information, see "How to Use Your Assigned Administrative Rights" in Oracle Solaris Administration: Security Services.

2 (Optional) If you need to add a device driver to support the device, complete the procedure "How to Add a Device Driver" on page 70.

3 Shut down the system.
   # shutdown -i0 -g30 -y
   -i0 Brings the system to the 0 init state, which is the appropriate state for turning the system power off for adding and removing devices.
Shut the system down in 30 seconds. The default is 60 seconds.

Continues the system shutdown without user intervention. Otherwise, you are prompted to continue the shutdown process.

4 Select one of the following to turn off power to the system after it is shut down:
   - For SPARC platforms, it is safe to turn off power if the ok prompt is displayed.
   - For x86 platforms, it is safe to turn off power if the type any key to continue prompt is displayed.

5 Turn off power to all peripheral devices.
For the location of power switches on any peripheral devices, refer to the hardware installation guides that accompany your peripheral devices.

6 Install the peripheral device, making sure that the device you are adding has a different target number than the other devices on the system.
Often, a small switch is located at the back of the disk for selecting the target number.
Refer to the hardware installation guide that accompanies the peripheral device for information on installing and connecting the device.

7 Turn on the power to the system.
The system boots to multiuser mode, and the login prompt is displayed.

8 Verify that the peripheral device has been added by attempting to access the device.
For information on accessing the device, see “Accessing Devices” on page 71.

How to Add a Device Driver

In the Oracle Solaris 11 release, the pkg commands are used to add packages to the system. Device driver packages might still be in SRV4 package format so the steps below use the pkgadd command instead of the pkg install command.

This procedure assumes that the device has already been added to the system. If not, see “What You Need for Unsupported Devices” on page 62.

1 Become an administrator.
For more information, see “How to Use Your Assigned Administrative Rights” in Oracle Solaris Administration: Security Services.

2 Place the media into the drive.
3 Install the driver.
   
   # pkgadd [-d] device package-name

4 Verify that the package has been added correctly.
   
   # pkgchk package-name
   
   The system prompt returns with no response if the package is installed correctly.

Example 5–2 Adding a Device Driver

The following example shows how to install and verify a package called XYZdrv.

   # pkgadd XYZdrv
   (licensing messages displayed)
   
   Installing XYZ Company driver as <XYZdrv>
   
   Installation of <XYZdrv> was successful.
   # pkgchk XYZdrv
   #

Accessing Devices

You need to know how to specify device names when using commands to manage disks, file systems, and other devices. In most cases, you can use logical device names to represent devices that are connected to the system. Both logical and physical device names are represented on the system by logical and physical device files.

How Device Information Is Created

When a system is booted for the first time, a device hierarchy is created to represent all the devices connected to the system. The kernel uses the device hierarchy information to associate drivers with their appropriate devices. The kernel also provides a set of pointers to the drivers that perform specific operations.
How Devices Are Managed

The devfs file system manages the /devices directory, which is the name space of all devices on the system. This directory represents the physical devices that consists of actual bus and device addresses.

The dev file system manages the /dev directory, which is the name space of logical device names.

By default, the devfsadm command attempts to load every driver in the system and attach to all possible device instances. Then, devfsadm creates the device files in the /devices directory and the logical links in the /dev directory. The devfsadm command also maintains the path_to_inst instance database.

Updates to the /dev and /devices directories in response to dynamic reconfiguration events or file system accesses are handled by devfsadm, the daemon version of the devfsadm command. This daemon is started by the service management facility when a system is booted.

Because the devfsadm daemon automatically detects device configuration changes generated by any reconfiguration event, there is no need to run this command interactively.

For more information, see the following references:
- devfsadm(1M)
- dev(7FS)
- devfs(7FS)
- path_to_inst(4)

Device Naming Conventions

Devices are referenced in three ways in the Oracle Solaris OS.

- **Physical device name** – Represents the full device path name in the device information hierarchy. The physical device name is created by when the device is first added to the system. Physical device files are found in the /devices directory.

- **Instance name** – Represents the kernel’s abbreviation name for every possible device on the system. For example, sd0 and sd1 represent the instance names of two disk devices. Instance names are mapped in the /etc/path_to_inst file.

- **Logical device name** – The logical device name is created by when the device is first added to the system. Logical device names are used with most file system commands to refer to devices. For a list of file commands that use logical device names, see Table 5–3. Logical device files in the /dev directory are symbolically linked to physical device files in the /devices directory.
The preceding device name information is displayed with the following commands:

- dmesg
- format
- sysdef
- prtconf

Logical Disk Device Names

Logical device names are used to access disk devices when you perform the following tasks:

- Add a new disk to the system.
- Move a disk from one system to another system.
- Access or mount a file system residing on a local disk.
- Back up a local file system.

Many administration commands take arguments that refer to a disk slice or file system.

Refer to a disk device by specifying the subdirectory to which it is symbolically linked, either /dev/dsk or /dev/rdsk, followed by a string identifying the particular controller, disk, and slice.

FIGURE 5-1  Description of Logical Device Names

`/dev/[r]dsk/cvtwdxs[ypz]`

- Slice number (s0 to s7) or fdisk partition number (p0 to p4)
- Drive number
- Physical bus target number
- Logical controller number
- Raw disk device subdirectory
- Devices directory

Specifying the Disk Subdirectory

Some disk and file administration commands require the use of either a raw (or character) device interface, or a block device interface. The distinction is made by how data is read from the device.

Raw device interfaces transfer only small amounts of data at a time. Block device interfaces include a buffer from which large blocks of data are read at once.
Different commands require different interfaces:

- When a command requires the raw device interface, specify the /dev/rdsk subdirectory. (The "r" in rdsk stands for "raw.")
- When a command requires the block device interface, specify the /dev/dsk subdirectory.
- When you are not sure whether a command requires use of /dev/dsk or /dev/rdsk, check the man page for that command.

The following table shows which interface is required for some commonly used disk and file system commands.

<table>
<thead>
<tr>
<th>Command Reference</th>
<th>Interface Type</th>
<th>Example of Use</th>
</tr>
</thead>
<tbody>
<tr>
<td>dumpadm(1M)</td>
<td>Block</td>
<td>dumpadm -d /dev/zvol/dsk/rpool/dump</td>
</tr>
<tr>
<td>prtvtoc(1M)</td>
<td>Raw</td>
<td>prtvtoc /dev/rdsk/c0t0d0s0</td>
</tr>
<tr>
<td>swap(1M)</td>
<td>Block</td>
<td>swap -a /dev/zvol/dsk/rpool/swap</td>
</tr>
</tbody>
</table>

**Direct and Bus-Oriented Controllers**

You might access disk partitions or slices differently depending upon whether the disk device is connected to a direct or bus-oriented controller. Generally, direct controllers do not include a target identifier in the logical device name.

The conventions for both types of controllers are explained in the following subsections.

**Note** – Controller numbers are assigned automatically during system initialization. The numbers are strictly logical and imply no direct mapping to physical controllers.

**x86: Disks With Direct Controllers**

To specify a slice on a disk with an IDE controller on an x86 based system, follow the naming convention shown in the following figure.
To indicate the entire Solaris fdisk partition, specify slice 2 (s2).

If you have only one controller on your system, w is usually 0.

**Disks With Bus-Oriented Controllers**

To specify a slice on a disk with a bus-oriented controller, SCSI for instance, follow the naming convention shown in the following figure.

![Diagram of Disks With Bus-Oriented Controllers](image)

On a SPARC based system with directly connected disks such as the IDE disks on an UltraSPARC system, the naming convention is the same as that for systems with bus-oriented controllers.

If you have only one controller on your system, w is usually 0.

For SCSI controllers, x is the target address set by the switch on the back of the unit, and y is the logical unit number (LUN) of the drive attached to the target. If the disk has an embedded controller, y is usually 0.

To indicate the whole disk, specify slice 2 (s2).

**Logical Tape Device Names**

Logical tape device files are found in the /dev/rmt/* directory as symbolic links from the /devices directory.
The first tape device connected to the system is 0 (/dev/rmt/0). Tape density values (l, m, h, c, and u) are described in Chapter 21, “Managing Tape Drives (Tasks).”

Logical Removable Media Device Names

Since removable media is managed by removable media management services, the logical device name is usually not used unless you want to mount the media manually.

The logical device name that represents the removable media devices on a system are described in Chapter 3, “Accessing Removable Media (Tasks).”
Dynamically Configuring Devices (Tasks)

This chapter provides instructions for dynamically configuring devices in the Oracle Solaris OS. You can add, remove, or replace devices in the Oracle Solaris OS while the system is still running, if the system components support hot-plugging. If the system components do not support hot-plugging, you can reboot the system to reconfigure the devices.

For information on the procedures associated with dynamically configuring devices, see the following:

- “SCSI Hot-Plugging With the \texttt{cfgadm} Command (Task Map)” on page 83
- “PCI or PCIe Hot-Plugging With the \texttt{cfgadm} Command (Task Map)” on page 93
- “SATA Hot-Plugging With the \texttt{cfgadm} Command” on page 100
- "Application Developer RCM Script (Task Map)” on page 103
- “System Administrator RCM Script (Task Map)” on page 104

For information on hot-plugging USB devices with the \texttt{cfgadm} command, see “Hot-Plugging USB Devices With the \texttt{cfgadm} Command” on page 151.

For information on hot-plugging InfiniBand devices with \texttt{cfgadm} command, see Chapter 9, “Using InfiniBand Devices (Overview/Tasks).”

For information about accessing devices, see “Accessing Devices” on page 71.

Dynamic Reconfiguration and Hot-Plugging

Hot-plugging is the ability to physically add, remove, or replace system components while the system is running. Dynamic reconfiguration refers to the ability to hot-plug system components. This term also refers to the general ability to move system resources (both hardware and software) around in the system or to disable them in some way without physically removing them from the system.

Generally, you can hot-plug the following bus types:
In addition, you can hot-plug the following devices with the `cfgadm` command:

- USB devices on SPARC and x86 platforms
- SCSI or SAS devices on SPARC and x86 platforms
- PCI devices on SPARC and x86 platforms
- PCIe devices on SPARC or x86 platforms
- SATA devices on SPARC and x86 platforms
- InfiniBand devices on SPARC and x86 platforms

Features of the `cfgadm` command include the following:

- Displaying system component status
- Testing system components
- Changing component configurations
- Displaying configuration help messages

The benefit of using the `cfgadm` command to reconfigure systems components is that you can add, remove, or replace components while the system is running. An added benefit is that the `cfgadm` command guides you through the steps needed to add, remove, or replace system components.

For step-by-step instructions on hot-plugging components, see the following:

- "SCSI Hot-Plugging With the `cfgadm` Command" on page 84
- "PCI or PCIe Hot-Plugging With the `cfgadm` Command" on page 94
- "SATA Hot-Plugging With the `cfgadm` Command" on page 100
- `cfgadm(1M)`

**Note** – Not all SCSI and PCI controllers support hot-plugging with the `cfgadm` command.

As part of Oracle's high availability strategy, dynamic reconfiguration is expected to be used in conjunction with additional layered products, such as alternate pathing or fail over software. Both products provide fault tolerance in the event of a device failure.

Without any high availability software, you can replace a failed device by manually stopping the appropriate applications, unmounting noncritical file systems, and then proceeding with the add or remove operations.
Note – Some systems have slots that hot-pluggable and slots that are not hot-pluggable. For information about hot-plugging devices on your specific hardware configuration, such as on enterprise-level systems, refer to your hardware configuration documentation.

**Attachment Points**

The `cfgadm` command displays information about *attachment points*, which are locations in the system where dynamic reconfiguration operations can occur.

An attachment point consists of the following:

- An *occupant*, which represents a hardware component that can be configured into the system
- A *receptacle*, which is the location that accepts the occupant

Attachment points are represented by logical and physical attachment point IDs (Ap_Ids). The physical Ap_Id is the physical path name of the attachment point. The logical Ap_Id is a user-friendly alternative for the physical Ap_Id. For more information on Ap_Ids, refer to `cfgadm(1M)`.

The logical Ap_Id for a SCSI Host Bus Adapter (HBA), or SCSI controller, is usually represented by the controller number, such as c0.

In cases where no controller number has been assigned to a SCSI HBA, then an internally generated unique identifier is provided. An example of a unique identifier for a SCSI controller is the following:

```
fas1:scsi
```

The logical Ap_Id for a SCSI device usually has this format:

```
HBA-logical-apid::device-identifier
```

In the following example, c0 is the logical Ap_Id for the SCSI HBA:

```
c0::dsk/c0t3d0
```

The device identifier is typically derived from the logical device name for the device in the `/dev` directory. For example, a tape device with logical device name, `/dev/rmt/1`, has the following logical Ap_Id:

```
c0::rmt/1
```

If a logical Ap_Id of a SCSI device cannot be derived from the logical name in the `/dev` directory, then an internally generated unique identifier is provided. An example of an identifier for the `/dev/rmt/1` tape device is the following:
c0::st4

For more information on SCSI Ap_Ids, refer to `cfgadm_scsi(1M)`.

The `cfgadm` command represents all resources and dynamic reconfiguration operations in terms of a common set of states (such as configured and unconfigured) and operations (such as connect, configure, unconfigure, and so on). For more information on these common states and operations, see `cfgadm(1M)`.

The following table shows the receptacle and occupant states for the SCSI HBA attachment points.

<table>
<thead>
<tr>
<th>Receptacle State</th>
<th>Description</th>
<th>Occupant State</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>empty</td>
<td>N/A for SCSI HBA</td>
<td>configured</td>
<td>One or more devices is configured on the bus</td>
</tr>
<tr>
<td>disconnected</td>
<td>Bus quiesced</td>
<td>unconfigured</td>
<td>No devices are configured</td>
</tr>
<tr>
<td>connected</td>
<td>Bus active</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

The following table shows the receptacle and occupant states for SCSI device attachment points.

<table>
<thead>
<tr>
<th>Receptacle State</th>
<th>Description</th>
<th>Occupant State</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>empty</td>
<td>N/A for SCSI devices</td>
<td>configured</td>
<td>Device is configured</td>
</tr>
<tr>
<td>disconnected</td>
<td>Bus quiesced</td>
<td>unconfigured</td>
<td>Device is not configured</td>
</tr>
<tr>
<td>connected</td>
<td>Bus active</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

The state of SCSI attachment points is unknown unless special hardware indicates otherwise.

For instructions on displaying SCSI component information, see “How to Display Information About SCSI Devices” on page 84.

**Detaching PCI or PCIe Adapter Cards**

A PCI adapter card that is hosting nonvital system resources can be removed if the device driver supports hot-plugging. A PCI adapter card is not detachable if it is a vital system resource. For a PCI adapter card to be detachable, the following conditions must be met:

- The device driver must support hot-plugging.
- Critical resources must be accessible through an alternate pathway.

For example, if a system has only one Ethernet card installed in it, the Ethernet card cannot be detached without losing the network connection. This detachment requires additional layered software support to keep the network connection active.
Attaching PCI or PCIe Adapter Cards

A PCI adapter card can be added to the system as long as the following conditions are met:

- There are slots available.
- The device driver supports hot-plugging for this adapter card.

For step-by-step instructions on adding or removing a PCI adapter card, see "PCI or PCIe Hot-Plugging With the cfgadm Command” on page 94.

PCle Hot-Plugging With the (hotplug) Command

You can use the hotplug command to manage hot pluggable connections, where a connection can be a connector or port, on PCI Express (PCIe) and PCI SHPC devices only. A hotplug connector is a physical representation in the system where a component is inserted or removed. A hotplug port is a logical representation in the system device tree where the connection of a device to the system is managed.

You can use the hotplug features to online or offline a device, even an on-board device, without physically adding or removing the device from the system.

You must enable the hotplug service to manage devices with the hotplug command.

```
# svcadm enable svc:/system/hotplug:default
```

The following examples describe how to use the hotplug command.

Display all the PCI/PCIe hot pluggable connectors/ports (virtual and physical) in the system as follows:

```
# hotplug list -lv
```

Configure an Ethernet card in a PCIe slot. For example:

```
# hotplug enable /pci0,0 pcie0
```

Unconfigure an Ethernet card in a PCIe slot. For example:

```
# hotplug disable /pci0,0 pcie0
```

Offline a PCI device node, which means detach the device driver for that node. For example:

```
# hotplug offline /pci0,0/pci1 pci.0,2
```

Online a PCI device node, which means attach the driver for that node. For example:

```
# hotplug online /pci0,0/pci1 pci.0,2
```
Install dependent ports of an IOV physical function. For example:

```
# hotplug install /pci@400/pci@1/pci@0/pci@4 pci.0,1
```

Then, display the resulting IOV virtual functions that were probed. For example:

```
# hotplug list -v /pci@400/pci@1/pci@0/pci@4 pci.0,1
<pci.0,1> (ONLINE)
  { IOV physical function }
  { IOV virtual function 'pci.0,81' }
  { IOV virtual function 'pci.0,83' }
  { IOV virtual function 'pci.0,85' }
  { IOV virtual function 'pci.0,87' }
<pci.0,81> (OFFLINE)
ethernet@0,81
<pci.0,83> (OFFLINE)
ethernet@0,83
<pci.0,85> (OFFLINE)
ethernet@0,85
<pci.0,87> (OFFLINE)
ethernet@0,87
```

Uninstall the dependent ports of an IOV physical function. For example:

```
# hotplug uninstall /pci@400/pci@1/pci@0/pci@4 pci.0,0
```

This operation fails if a dependent IOV virtual function is busy. For example:

```
# hotplug uninstall /pci@400/pci@1/pci@0/pci@4 pci.0,0
ERROR: devices or resources are busy.
ethernet@0,81:
  { Network interface igbvf1 }
  { igbvf1: hosts IP addresses: 10.0.0.1 }
  { Plumbed IP Address }
```

### Troubleshooting PCI Hot Plug Operations (hotplug)

You might see the following maintenance states for an attached device in a hotplug port.

```
/pci@0,0/pci@0de,5d@e <pci.a,1> (MAINTENANCE)
```

```
/pci@0,0/pci@0de,534a@d <pci.0,0> (MAINTENANCE-SUSPENDED)
```

The above messages indicate that a fault event or a maintenance operation occurred. The **MAINTENANCE** states mean that a device is in use, but it is not fully operational. The **MAINTENANCE-SUSPENDED** state means that the device is live suspended, due to a maintenance operation. For example, reconfiguring the device hardware.

The following service must be running to use the `hotplug` command.

```
svc:/system/hotplug:default
```

Otherwise, you will see the following message:
ERROR: hotplug service is not available.

The following error message is displayed on systems that do not have any supported I/O buses:

ERROR: there are no connections to display.
(See hotplug(1m) for more information.)

The above message could mean that the system might have other hot-pluggable I/O devices, but you need to use the cfgadm command rather than the hotplug command to manage these devices.

### SCSI Hot-Plugging With the cfgadm Command (Task Map)

<table>
<thead>
<tr>
<th>Task</th>
<th>Description</th>
<th>For Instructions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Display information about SCSI devices.</td>
<td>Display information about SCSI controllers and devices.</td>
<td>“How to Display Information About SCSI Devices” on page 84</td>
</tr>
<tr>
<td>Unconfigure a SCSI controller.</td>
<td>Unconfigure a SCSI controller.</td>
<td>“How to Unconfigure a SCSI Controller” on page 85</td>
</tr>
<tr>
<td>Configure a SCSI controller.</td>
<td>Configure a SCSI controller that was previously unconfigured.</td>
<td>“How to Configure a SCSI Controller” on page 85</td>
</tr>
<tr>
<td>Configure a SCSI device.</td>
<td>Configure a specific SCSI device.</td>
<td>“How to Configure a SCSI Device” on page 86</td>
</tr>
<tr>
<td>Disconnect a SCSI controller.</td>
<td>Disconnect a specific SCSI controller.</td>
<td>“How to Disconnect a SCSI Controller” on page 86</td>
</tr>
<tr>
<td>Connect a SCSI controller.</td>
<td>Connect a specific SCSI controller that was previously disconnected.</td>
<td>“SPARC: How to Connect a SCSI Controller” on page 87</td>
</tr>
<tr>
<td>Add a SCSI device to a SCSI bus.</td>
<td>Add a specific SCSI device to a SCSI bus.</td>
<td>“SPARC: How to Add a SCSI Device to a SCSI Bus” on page 88</td>
</tr>
<tr>
<td>Replace an identical device on a SCSI controller.</td>
<td>Replace a device on the SCSI bus with another device of the same type.</td>
<td>“SPARC: How to Replace an Identical Device on a SCSI Controller” on page 89</td>
</tr>
<tr>
<td>Remove a SCSI device.</td>
<td>Remove a SCSI device from the system.</td>
<td>“SPARC: How to Remove a SCSI Device” on page 90</td>
</tr>
<tr>
<td>Troubleshoot SCSI configuration problems.</td>
<td>Resolve a failed SCSI unconfigure operation.</td>
<td>“How to Resolve a Failed SCSI Unconfigure Operation” on page 93</td>
</tr>
</tbody>
</table>
This section describes various SCSI hot-plugging procedures that you can perform with the `cfgadm` command.

**Note** – The SCSI framework generally supports hot-plugging of SCSI devices. However, you should consult your hardware documentation to confirm whether hot-plugging is supported for your SCSI devices.

These procedures use specific devices as examples to illustrate how to use the `cfgadm` command to hot-plug SCSI components. The device information that you supply, and that the `cfgadm` command displays, depends on your system configuration.

All of the procedures in this section require administrative privileges that are not generally granted to user accounts. For more information, see “How to Use Your Assigned Administrative Rights” in Oracle Solaris Administration: Security Services.

### How to Display Information About SCSI Devices

The following procedure uses SCSI controllers `c0` and `c1` and the devices that are attached to them in the examples of the type of device configuration information that you can display with the `cfgadm` command.

**Note** – If the SCSI device is not supported by the `cfgadm` command, the device does not display in the `cfgadm` command output.

1. **Become an administrator.**

2. **Display information about attachment points on the system.**

   ```shell
   # cfgadm -l
   Ap_Id Type  Receptacle Occupant Condition
   c0         scsi-bus  connected  configured unknown
   c1         scsi-bus  connected  configured unknown
   ```

   In this example, `c0` and `c1` represent two SCSI controllers.

3. **Display information about a system’s SCSI controllers and their attached devices.**

   ```shell
   # cfgadm -al
   Ap_Id Type  Receptacle Occupant Condition
   c0         scsi-bus  connected  configured unknown
   c0::dsk/c0t0d0 disk connected configured unknown
   c0::rmt/0   tape connected configured unknown
   c1         scsi-bus  connected  configured unknown
   c1::dsk/c1t3d0 disk connected configured unknown
   c1::dsk/c1t4d0 unavailable connected unconfigured unknown
   ```
How to Unconfigure a SCSI Controller

The following procedure uses SCSI controller c1 in the example of unconfiguring a SCSI controller.

1 Become an administrator.

2 Unconfigure a SCSI controller.

```
# cfgadm -c unconfigure c1
```

3 Verify that the SCSI controller is unconfigured.

```
# cfgadm -al

<table>
<thead>
<tr>
<th>Ap_Id</th>
<th>Type</th>
<th>Receptacle</th>
<th>Occupant</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>c0</td>
<td>scsi-bus</td>
<td>connected</td>
<td>configured</td>
<td>unknown</td>
</tr>
<tr>
<td>c0::dsk/c0t0d0</td>
<td>disk</td>
<td>connected</td>
<td>configured</td>
<td>unknown</td>
</tr>
<tr>
<td>c0::rmt/0</td>
<td>tape</td>
<td>connected</td>
<td>configured</td>
<td>unknown</td>
</tr>
<tr>
<td>c1</td>
<td>scsi-bus</td>
<td>connected</td>
<td>unconfigured</td>
<td>unknown</td>
</tr>
</tbody>
</table>
```

Notice that the Occupant column for c1 specifies unconfigured, indicating that the SCSI bus has no configured occupants.

If the unconfigure operation fails, see “How to Resolve a Failed SCSI Unconfigure Operation” on page 93.

How to Configure a SCSI Controller

The following procedure uses SCSI controller c1 in the example of configuring a SCSI controller.

1 Become an administrator.

2 Configure a SCSI controller.

```
# cfgadm -c configure c1
```

3 Verify that the SCSI controller is configured.

```
# cfgadm -al

<table>
<thead>
<tr>
<th>Ap_Id</th>
<th>Type</th>
<th>Receptacle</th>
<th>Occupant</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>c0</td>
<td>scsi-bus</td>
<td>connected</td>
<td>configured</td>
<td>unknown</td>
</tr>
<tr>
<td>c0::dsk/c0t0d0</td>
<td>disk</td>
<td>connected</td>
<td>configured</td>
<td>unknown</td>
</tr>
<tr>
<td>c0::rmt/0</td>
<td>tape</td>
<td>connected</td>
<td>configured</td>
<td>unknown</td>
</tr>
</tbody>
</table>
```
The previous unconfigure procedure removed all devices on the SCSI bus. Now all the devices are configured back into the system.

▼ How to Configure a SCSI Device

The following procedure uses SCSI disk c1t4d0 in the example of configuring a SCSI device.

1 Become an administrator.

2 Identify the device to be configured.

```bash
# cfgadm -al
Ap_Id  Type     Receptacle Occupant  Condition
c0     scsi-bus connected configured unknown
c0::dsk/c0t0d0 disk      connected configured unknown
c0::rmt/0 tape    connected configured unknown
c1     scsi-bus connected configured unknown
c1::dsk/c1t3d0 disk      unavailable connected configured unknown
c1::dsk/c1t4d0 disk      connected configured unknown
```

3 Configure the SCSI device.

```bash
# cfgadm -c configure c1::dsk/c1t4d0
```

4 Verify that the SCSI device is configured.

```bash
# cfgadm -al
Ap_Id  Type     Receptacle Occupant  Condition
c0     scsi-bus connected configured unknown
c0::dsk/c0t0d0 disk      connected configured unknown
c0::rmt/0 tape    connected configured unknown
c1     scsi-bus connected configured unknown
c1::dsk/c1t3d0 disk      connected configured unknown
c1::dsk/c1t4d0 disk      connected configured unknown
```

▼ How to Disconnect a SCSI Controller

**Caution** – Disconnecting a SCSI device must be done with caution, particularly when you are dealing with controllers for disks that contain critical file systems such as root (/), usr, var, and the swap partition. The dynamic reconfiguration software cannot detect all cases where a system hang might result. Use this procedure with caution.

The following procedure uses SCSI controller c1 in the example of disconnecting a SCSI device.

1 Become an administrator.
2 Verify that the device is connected before you disconnect it.

```
# cfgadm -al
```

<table>
<thead>
<tr>
<th>Ap_Id</th>
<th>Type</th>
<th>Receptacle</th>
<th>Occupant</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>c0</td>
<td>scsi-bus</td>
<td>connected</td>
<td>configured</td>
<td>unknown</td>
</tr>
<tr>
<td>c0::dsk/c0t0d0</td>
<td>disk</td>
<td>connected</td>
<td>configured</td>
<td>unknown</td>
</tr>
<tr>
<td>c0::rmt/0</td>
<td>tape</td>
<td>connected</td>
<td>configured</td>
<td>unknown</td>
</tr>
<tr>
<td>c1::dsk/c1t3d0</td>
<td>disk</td>
<td>connected</td>
<td>configured</td>
<td>unknown</td>
</tr>
<tr>
<td>c1::dsk/c1t4d0</td>
<td>disk</td>
<td>connected</td>
<td>configured</td>
<td>unknown</td>
</tr>
</tbody>
</table>

3 Disconnect the SCSI controller.

```
# cfgadm -c disconnect c1
```

WARNING: Disconnecting critical partitions may cause system hang. Continue (yes/no)? y

Caution – This command suspends all I/O activity on the SCSI bus until the `cfgadm -c connect` command is used. The `cfgadm` command does some basic checking to prevent critical partitions from being disconnected, but it cannot detect all cases. Inappropriate use of this command can result in a system hang and could require a system reboot.

4 Verify that the SCSI bus is disconnected.

```
# cfgadm -al
```

<table>
<thead>
<tr>
<th>Ap_Id</th>
<th>Type</th>
<th>Receptacle</th>
<th>Occupant</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>c0</td>
<td>scsi-bus</td>
<td>connected</td>
<td>configured</td>
<td>unknown</td>
</tr>
<tr>
<td>c0::dsk/c0t0d0</td>
<td>disk</td>
<td>connected</td>
<td>configured</td>
<td>unknown</td>
</tr>
<tr>
<td>c0::rmt/0</td>
<td>tape</td>
<td>connected</td>
<td>configured</td>
<td>unknown</td>
</tr>
<tr>
<td>c1::dsk/c1t3d0</td>
<td>unavailable</td>
<td>disconnected</td>
<td>configured</td>
<td>unknown</td>
</tr>
<tr>
<td>c1::dsk/c1t4d0</td>
<td>unavailable</td>
<td>disconnected</td>
<td>configured</td>
<td>unknown</td>
</tr>
</tbody>
</table>

The controller and all the devices that are attached to it are disconnected from the system.

▼ **SPARC: How to Connect a SCSI Controller**

The following procedure uses SCSI controller c1 in the example of connecting a SCSI controller.

1 Become an administrator.

2 Verify that the device is disconnected before you connect it.

```
# cfgadm -al
```

<table>
<thead>
<tr>
<th>Ap_Id</th>
<th>Type</th>
<th>Receptacle</th>
<th>Occupant</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>c0</td>
<td>scsi-bus</td>
<td>connected</td>
<td>configured</td>
<td>unknown</td>
</tr>
<tr>
<td>c0::dsk/c0t0d0</td>
<td>disk</td>
<td>connected</td>
<td>configured</td>
<td>unknown</td>
</tr>
<tr>
<td>c0::rmt/0</td>
<td>tape</td>
<td>connected</td>
<td>configured</td>
<td>unknown</td>
</tr>
<tr>
<td>c1::dsk/c1t3d0</td>
<td>unavailable</td>
<td>disconnected</td>
<td>configured</td>
<td>unknown</td>
</tr>
<tr>
<td>c1::dsk/c1t4d0</td>
<td>unavailable</td>
<td>disconnected</td>
<td>configured</td>
<td>unknown</td>
</tr>
</tbody>
</table>
3 Connect the SCSI controller.

# cfgadm -c connect c1

4 Verify that the SCSI controller is connected.

# cfgadm -al

<table>
<thead>
<tr>
<th>Ap_Id</th>
<th>Type</th>
<th>Receptacle</th>
<th>Occupant</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>c0</td>
<td>scsi-bus</td>
<td>connected</td>
<td>configured</td>
<td>unknown</td>
</tr>
<tr>
<td>c0::dsk/c0t0d0</td>
<td>disk</td>
<td>connected</td>
<td>configured</td>
<td>unknown</td>
</tr>
<tr>
<td>c0::rmt/0</td>
<td>tape</td>
<td>connected</td>
<td>configured</td>
<td>unknown</td>
</tr>
<tr>
<td>c1</td>
<td>scsi-bus</td>
<td>connected</td>
<td>configured</td>
<td>unknown</td>
</tr>
<tr>
<td>c1::dsk/c1t3d0</td>
<td>disk</td>
<td>connected</td>
<td>configured</td>
<td>unknown</td>
</tr>
<tr>
<td>c1::dsk/c1t4d0</td>
<td>disk</td>
<td>connected</td>
<td>configured</td>
<td>unknown</td>
</tr>
</tbody>
</table>

▼ SPARC: How to Add a SCSI Device to a SCSI Bus

SCSI controller c1 is used in the example of how to add a SCSI device to a SCSI bus.

- **Note** – When you add devices, you specify the Ap_Id of the SCSI HBA (controller) to which the device is attached, not the Ap_Id of the device itself.

1 Become an administrator.

2 Identify the current SCSI configuration.

# cfgadm -al

<table>
<thead>
<tr>
<th>Ap_Id</th>
<th>Type</th>
<th>Receptacle</th>
<th>Occupant</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>c0</td>
<td>scsi-bus</td>
<td>connected</td>
<td>configured</td>
<td>unknown</td>
</tr>
<tr>
<td>c0::dsk/c0t0d0</td>
<td>disk</td>
<td>connected</td>
<td>configured</td>
<td>unknown</td>
</tr>
<tr>
<td>c0::rmt/0</td>
<td>tape</td>
<td>connected</td>
<td>configured</td>
<td>unknown</td>
</tr>
<tr>
<td>c1</td>
<td>scsi-bus</td>
<td>connected</td>
<td>configured</td>
<td>unknown</td>
</tr>
<tr>
<td>c1::dsk/c1t3d0</td>
<td>disk</td>
<td>connected</td>
<td>configured</td>
<td>unknown</td>
</tr>
<tr>
<td>c1::dsk/c1t4d0</td>
<td>disk</td>
<td>connected</td>
<td>configured</td>
<td>unknown</td>
</tr>
</tbody>
</table>

3 Add the SCSI device to the SCSI bus.

a. Type the following `cfgadm` command.

For example:

```
# cfgadm -x insert_device c1
Adding device to SCSI HBA: /devices/sbus@1f,0/SUNW,fas@1,0000000
This operation will suspend activity on SCSI bus: c1
```

b. Type `y` at the Continue (yes/no) prompt to proceed.

```
Continue (yes/no)? y
SCSI bus quiesced successfully.
It is now safe to proceed with hotplug operation.
```

I/O activity on the SCSI bus is suspended while the hot-plug operation is in progress.

c. Connect the device and then power it on.
d. Type \textit{y} at the \texttt{Enter y if operation is complete or n to abort (yes/no)?} prompt.

\begin{verbatim}
Enter y if operation is complete or n to abort (yes/no)? y
\end{verbatim}

4 Verify that the device has been added.

\begin{verbatim}
# cfgadm -al
Ap_Id Type Receptacle Occupant Condition
c0 scsi-bus connected configured unknown
c0::dsk/c0t0d0 disk connected configured unknown
c0::rmt/0 tape connected configured unknown
c1 scsi-bus connected configured unknown
c1::dsk/c1t3d0 disk connected configured unknown
c1::dsk/c1t4d0 disk connected configured unknown
\end{verbatim}

A new disk has been added to controller c1.

\begin{flushleft}
\textbf{▼ SPARC: How to Replace an Identical Device on a SCSI Controller}
\end{flushleft}

The following procedure uses SCSI disk c1t4d0 in the example of replacing an identical device on a SCSI controller.

Review the following conditions when attempting to replace an identical device on a SCSI controller:

- If you are replacing a mirrored or unmirrored boot device that is part of a ZFS root pool, see the following reference:
  \url{http://www.oracle.com/technetwork/articles/servers-storage-admin/o11-091-sol-dis-recovery-489183.html}

- If the device is controlled by legacy volume management software, consult your volume management documentation for the specific steps to replace a device. If you have an active Oracle support plan, see the following references:
  - If you are using Solaris Volume Manager (SVM), see MOS document 1010753.1.
  - If you are using Veritas Volume Manager (VxVM), see MOS document 1003122.1, 1011782.1, and 1002285.1.

1 Become an administrator.

2 Identify the current SCSI configuration.

\begin{verbatim}
# cfgadm -al
Ap_Id Type Receptacle Occupant Condition
c0 scsi-bus connected configured unknown
c0::dsk/c0t0d0 disk connected configured unknown
c0::rmt/0 tape connected configured unknown
c1 scsi-bus connected configured unknown
\end{verbatim}
Replace a device on the SCSI bus with another device of the same type.

a. Type the following `cfgadm` command.

For example:

```bash
# cfgadm -x replace_device c1::dsk/c1t4d0
Replacing SCSI device: /devices/sbus@1f,0/SUNW,fas@1,8800000/sd@4,0
This operation will suspend activity on SCSI bus: c1
```

b. Type `y` at the `Continue (yes/no) ?` prompt to proceed.

I/O activity on the SCSI bus is suspended while the hot-plug operation is in progress.

Continue (yes/no)? y
SCSI bus quiesced successfully.
It is now safe to proceed with hotplug operation.

C. Power off the device to be removed and remove it.

d. Add the replacement device. Then, power it on.

The replacement device should be of the same type and at the same address (target and LUN) as the device to be removed.

e. Type `y` at the `Enter y if operation is complete or n to abort (yes/no)?` prompt.

Enter y if operation is complete or n to abort (yes/no)? y

Verify that the device has been replaced.

```bash
# cfgadm -al
Ap_Id Type Receptacle Occupant Condition
```

**SPARC: How to Remove a SCSI Device**

The following procedure uses SCSI disk `c1t4d0` in the example of removing a device on a SCSI controller.

1 Become an administrator.

2 Identify the current SCSI configuration.

```bash
# cfgadm -al
Ap_Id Type Receptacle Occupant Condition
```
3 Remove the SCSI device from the system.

a. Type the following `cfgadm` command.

   For example:
   
   ```
   # cfgadm -x remove_device c1::dsk/c1t4d0
   Removing SCSI device: /devices/sbus@1f,0/SUNW,fas@1,8800000/sd@4,0
   This operation will suspend activity on SCSI bus: c1
   ```

b. Type `y` at the Continue (yes/no)? prompt to proceed.

   ```
   Continue (yes/no)? y
   SCSI bus quiesced successfully.
   It is now safe to proceed with hotplug operation.
   ```

   I/O activity on the SCSI bus is suspended while the hot-plug operation is in progress.

c. Power off the device to be removed and remove it.

d. Type `y` at the Enter y if operation is complete or n to abort (yes/no)? prompt.

   ```
   Enter y if operation is complete or n to abort (yes/no)? y
   ```

**Note** — This step must be performed if you are removing a SCSI RAID device from a SCSI RAID array.

4 Verify that the device has been removed from the system.

```
# cfgadm -al
```

<table>
<thead>
<tr>
<th>Ap_Id</th>
<th>Type</th>
<th>Receptacle</th>
<th>Occupant</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>c0</td>
<td>scsi-bus</td>
<td>connected</td>
<td>configured</td>
<td>unknown</td>
</tr>
<tr>
<td>c0::dsk/c0t0d0</td>
<td>disk</td>
<td>connected</td>
<td>configured</td>
<td>unknown</td>
</tr>
<tr>
<td>c0::rmt/0</td>
<td>tape</td>
<td>connected</td>
<td>configured</td>
<td>unknown</td>
</tr>
<tr>
<td>c1</td>
<td>scsi-bus</td>
<td>connected</td>
<td>configured</td>
<td>unknown</td>
</tr>
<tr>
<td>c1::dsk/c1t3d0</td>
<td>disk</td>
<td>connected</td>
<td>configured</td>
<td>unknown</td>
</tr>
<tr>
<td>c1::dsk/c1t4d0</td>
<td>disk</td>
<td>connected</td>
<td>configured</td>
<td>unknown</td>
</tr>
</tbody>
</table>

**Troubleshooting SCSI Configuration Problems**

This section provides error messages and possible solutions for troubleshooting SCSI configuration problems. For more information on troubleshooting SCSI configuration problems, see `cfgadm(1M)`.
Error Message

cfgadm: Component system is busy, try again: failed to offline:

Resource   Information
------------------ --------------------------
/dev/dsk/c1t0d0s0 mounted filesystem "/file-system"

Cause
You attempted to remove or replace a device with a mounted file system.

Solution
Unmount the file system that is listed in the error message and retry the cfgadm operation.

Error Message

cfgadm: Component system is busy, try again: failed to offline:

Resource   Information
------------------ --------------------------
/dev/dsk/device-name swap area

Cause
If you use the cfgadm command to remove a system resource, such as a swap device or a dedicated dump device, a similar error message is displayed if the system resource is still active.

Solution
Unconfigure the swap areas on the device that is specified and retry the cfgadm operation.

Error Message

cfgadm: Component system is busy, try again: failed to offline:

Resource   Information
------------------ --------------------------
/dev/dsk/device-name dump device (swap)

Cause
You attempted to remove or replace a dump device that is configured on a swap area.

Solution
Unconfigure the dump device that is configured on the swap area and retry the cfgadm operation.

Error Message

cfgadm: Component system is busy, try again: failed to offline:

Resource   Information
------------------ --------------------------
/dev/dsk/device-name dump device (dedicated)

Cause
You attempted to remove or replace a dedicated dump device.
Solution
Unconfigure the dedicate dump device and retry the \texttt{cfgadm} operation.

\section*{How to Resolve a Failed SCSI Unconfigure Operation}

Use this procedure if one or more target devices are busy and the SCSI unconfigure operation fails. Otherwise, future dynamic reconfiguration operations on this controller and target devices will fail with a \texttt{dr in progress} message.

1. Become an administrator.
2. Reconfigure the controller.
   \texttt{# cfgadm -c configure device-name}

\section*{PCI or PCIe Hot-Plugging With the \texttt{cfgadm} Command (Task Map)}

The following task map describes the tasks for managing PCI or PCIe devices on your system.

<table>
<thead>
<tr>
<th>Task</th>
<th>Description</th>
<th>For Instructions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Display PCI slot configuration information.</td>
<td>Display the status of PCI hot-pluggable devices and slots on the system.</td>
<td>“How to Display PCI Slot Configuration Information” on page 94</td>
</tr>
<tr>
<td>Remove a PCI adapter card.</td>
<td>Unconfigure the card, disconnect power from the slot, and remove the card from the system.</td>
<td>“How to Remove a PCI Adapter Card” on page 96</td>
</tr>
<tr>
<td>Add a PCI adapter card.</td>
<td>Insert the adapter card into a hot-pluggable slot, connect power to the slot, and configure the card.</td>
<td>“How to Add a PCI Adapter Card” on page 97</td>
</tr>
<tr>
<td>Troubleshoot PCI configuration problems.</td>
<td>Identify error message and possible solutions to resolve PCI configuration problems.</td>
<td>“Troubleshooting PCI Configuration Problems” on page 99</td>
</tr>
</tbody>
</table>
PCI or PCIe Hot-Plugging With the cfgadm Command

This section provides step-by-step instructions for hot-plugging PCI or PCIe adapter cards on SPARC and x86 systems.

In addition to the `cfgadm` command, the `prtconf` command is helpful during hot-plug operations. The `prtconf` command displays additional configuration information that pertains to the hardware.

After adding hardware, you can use the `prtconf` command to verify that the hardware is configured correctly. For example, after a configure operation, use the `prtconf -D` command to verify that the driver is attached to the newly installed hardware device. If the device driver has not been added to the system prior to hardware configuration, it may be manually added by using the `add_drv` command.

For more information, see `prtconf(1M)` and `add_drv(1M)`.

All of the procedures in this section require administrative privileges that are not generally granted to user accounts. For more information, see "How to Use Your Assigned Administrative Rights" in Oracle Solaris Administration: Security Services.

In the examples, only PCI attachment points are listed, for brevity. The attachment points that are displayed on your system depend on your system configuration.

PCIe LED Indicator Behavior

You might observe the LED indicators on the system to get a visual indication about the status of the slot’s hot-plug operation. The LED behavior, in case of PCI Express, matches that defined in the PCI Express specification or the behavior might otherwise be platform dependent.

Please refer to your platform guide for specific details. In case of PCI Express, when the Attention Button is pressed, the power indicator blinks, which indicates the beginning of a state transition. The blinking ends when the state transition has ended.

How to Display PCI Slot Configuration Information

This procedure has been updated to include PCIe configuration information.

The `cfgadm` command displays the status of PCI hot-pluggable devices and slots on a system. For more information, see `cfgadm(1M)`.

1. Become an administrator.
2. Display PCI configuration information.
- Display PCI slot configuration information.

For example:

```
# cfgadm
Ap_Id       Type   Receptacle  Occupant  Condition
pci1:hpc0_slot0 unknown empty unconfigured unknown
pci1:hpc0_slot1 unknown empty unconfigured unknown
pci1:hpc0_slot2 unknown empty unconfigured unknown
pci1:hpc0_slot3 ethernet/hp connected configured ok
pci1:hpc0_slot4 unknown empty unconfigured unknown
```

- Display specific PCI device information.

For example:

```
# cfgadm -s "cols=ap_id:type:info" pci
Ap_Id       Type   Information
pci1:hpc0_slot0 unknown Slot 7
pci1:hpc0_slot1 unknown Slot 8
pci1:hpc0_slot2 unknown Slot 9
pci1:hpc0_slot3 ethernet/hp Slot 10
pci1:hpc0_slot4 unknown Slot 11
```

The logical Ap_Id, pci1:hpc0_slot0, is the logical Ap_Id for hot-pluggable slot, Slot 7. The component hpc0 indicates the hot-pluggable adapter card for this slot, and pci1 indicates the PCI bus instance. The Type field indicates the type of PCI adapter card that is present in the slot.

- Display PCIe slot configuration information.

For example:

```
# cfgadm pci
Ap_Id       Type   Receptacle  Occupant  Condition
pcie1       unknown empty unconfigured unknown
pcie2       unknown empty unconfigured unknown
pcie3       unknown empty unconfigured unknown
pcie4       ethernet/hp connected configured ok
pcie5       pci-pci/hp connected configured ok
pcie6       unknown disconnected unconfigured unknown
```

- Display specific PCIe device information.

For example:

```
# cfgadm -s "cols=ap_id:busy:o_state" pci
Ap_Id       Busy   Occupant
pcie1       n      unconfigured
pcie2       n      unconfigured
pcie3       n      unconfigured
pcie4       n      configured
pcie5       n      configured
pcie6       n      configured
```
Note – The logical Ap_Id in most cases should match the slot label that is silk-screened on the system chassis. Refer to your platform guide for the `cfgadm` output of the hot-pluggable slots. The Busy field can be displayed to ensure that the Ap_Id is not transitioning to another state before a hot-plug operation is attempted.

▼ How to Remove a PCI Adapter Card

The following procedure has been updated for removing a PCIe adapter card. However, the procedure to remove an adapter card is the same whether you are using PCI or PCIe.

1 Become an administrator.

2 Determine which slot the PCI adapter card is in.

   For example:
   
   ```
   # cfgadm pci
   Ap_Id | Type     | Receptacle | Occupant | Condition
   pcie1 | unknown  | empty      | unconfigured | unknown
   pcie2 | unknown  | empty      | unconfigured | unknown
   pcie3 | unknown  | empty      | unconfigured | unknown
   pcie4 | etherne/hp | connected | configured | ok
   pcie5 | pci-pci/hp | connected | configured | ok
   pcie6 | unknown  | disconnected | unconfigured | unknown
   ```

3 Stop the application that has the device open.

   For example, if the device is an Ethernet card, use the `ipadm` command to bring down the interface and unplug the interface. For example:
   
   ```
   # ipadm delete-ip bge1
   ```

4 Unconfigure the device manually by using the `cfgadm` command as shown below. Or, if you have a PCIe adapter card, use the auto-configuration method, such as pressing the slot’s Attention Button as defined by your platform guide.

   ```
   # cfgadm -c unconfigure pcie4
   ```

5 Confirm that the device has been unconfigured.

   For example:
   
   ```
   # cfgadm pci
   Ap_Id | Type     | Receptacle | Occupant | Condition
   pcie1 | unknown  | empty      | unconfigured | unknown
   pcie2 | unknown  | empty      | unconfigured | unknown
   pcie3 | unknown  | empty      | unconfigured | unknown
   pcie4 | unknown  | connected | unconfigured | unknown
   pcie5 | pci-pci/hp | connected | configured | ok
   pcie6 | unknown  | disconnected | unconfigured | unknown
   ```
Note – The Type and Condition also become unknown when the device is unconfigured.

6 Disconnect the power to the slot manually. If the auto-configuration method is used, this step is not necessary. Refer to your platform guide for more information.

```
# cfgadm -c disconnect pcie4
```

7 Confirm that the device has been disconnected.

For example:

```
# cfgadm pci

<table>
<thead>
<tr>
<th>Ap_Id</th>
<th>Type</th>
<th>Receptacle</th>
<th>Occupant</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>pcie1</td>
<td>unknown</td>
<td>empty</td>
<td>unconfigured</td>
<td>unknown</td>
</tr>
<tr>
<td>pcie2</td>
<td>unknown</td>
<td>empty</td>
<td>unconfigured</td>
<td>unknown</td>
</tr>
<tr>
<td>pcie3</td>
<td>unknown</td>
<td>empty</td>
<td>unconfigured</td>
<td>unknown</td>
</tr>
<tr>
<td>pcie4</td>
<td>unknown</td>
<td>disconnected</td>
<td>unconfigured</td>
<td>unknown</td>
</tr>
<tr>
<td>pcie5</td>
<td>pcie-pci/hp</td>
<td>connected</td>
<td>configured</td>
<td>ok</td>
</tr>
<tr>
<td>pcie6</td>
<td>unknown</td>
<td>disconnected</td>
<td>unconfigured</td>
<td>unknown</td>
</tr>
</tbody>
</table>
```

8 Follow appropriate instructions in your platform guide to remove the PCI adapter card. After the card is removed, the Receptacle state is empty.

For example:

```
# cfgadm pci

<table>
<thead>
<tr>
<th>Ap_Id</th>
<th>Type</th>
<th>Receptacle</th>
<th>Occupant</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>pcie1</td>
<td>unknown</td>
<td>empty</td>
<td>unconfigured</td>
<td>unknown</td>
</tr>
<tr>
<td>pcie2</td>
<td>unknown</td>
<td>empty</td>
<td>unconfigured</td>
<td>unknown</td>
</tr>
<tr>
<td>pcie3</td>
<td>unknown</td>
<td>empty</td>
<td>unconfigured</td>
<td>unknown</td>
</tr>
<tr>
<td>pcie4</td>
<td>unknown</td>
<td>empty</td>
<td>unconfigured</td>
<td>unknown</td>
</tr>
<tr>
<td>pcie5</td>
<td>pcie-pci/hp</td>
<td>connected</td>
<td>configured</td>
<td>ok</td>
</tr>
<tr>
<td>pcie6</td>
<td>unknown</td>
<td>disconnected</td>
<td>unconfigured</td>
<td>unknown</td>
</tr>
</tbody>
</table>
```

Note – The auto-configuration method can be enabled or disabled at boot, depending on the platform implementation. Set the auto-configuration method as appropriate for your environment.

▼ How to Add a PCI Adapter Card

The following procedure has been updated for adding a PCIe adapter card. However, the procedure to add an adapter card is the same whether you are using PCI or PCIe.

1 Become an administrator.

2 Identify the hot-pluggable slot and open latches.

For example, pcie3.

```
# cfgadm pci

<table>
<thead>
<tr>
<th>Ap_Id</th>
<th>Type</th>
<th>Receptacle</th>
<th>Occupant</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>pcie1</td>
<td>unknown</td>
<td>empty</td>
<td>unconfigured</td>
<td>unknown</td>
</tr>
</tbody>
</table>
```

Chapter 6 • Dynamically Configuring Devices (Tasks) 97
Follow the appropriate instructions in your platform guide to insert a PCI adapter card into the slot.

Determine which slot the PCI adapter card is in after it is inserted.

Connect the power to the slot manually using the \texttt{cfgadm} command. Or, if you have a PCIe adapter card, use the auto-configuration method, such as pressing the slot’s Attention Button as defined by your platform guide.

Confirm that the attachment point is connected.

Configure the PCI adapter card manually by using the \texttt{cfgadm} command as shown below. If using the auto-configuration method, this step should not be necessary. Refer to your platform guide for more information.

Verify the configuration of the PCI adapter card in the slot.
Configure any supporting software if this device is a new device.

For example, if this device is an Ethernet card, use the `ipadm` command to set up the interface.
For example:

```
# ipadm create-addr bge1
```

**Note** – The auto-configuration method can be enabled or disabled at boot, depending on the platform implementation. Set the auto-configuration method as appropriate for your environment.

## Troubleshooting PCI Configuration Problems

**Error Message**

```
cfgadm: Configuration operation invalid: invalid transition
```

**Cause**
An invalid transition was attempted.

**Solution**
Check whether the `cfgadm -c` command was issued appropriately. Use the `cfgadm` command to check the current receptacle and occupant state and to make sure that the `Ap_Id` is correct.

**Error Message**

```
cfgadm: Attachment point not found
```

**Cause**
The specified attachment point was not found.

**Solution**
Check whether the attachment point is correct. Use the `cfgadm` command to display a list of available attachment points. Also check the physical path to see if the attachment point is still there.
SATA Hot-Plugging With the cfgadm Command

SATA controller and port multiplier device ports are represented by attachment points in the device tree. SATA devices that are connected and configured on the system are shown as attachment point name extension. The terms attachment point and SATA port can be used interchangeably.

The `cfgadm` syntax that is used with SATA devices is slightly different than `cfgadm` syntax for SCSI or PCI devices.

You can display SATA device information as follows:

```
% cfgadm -al
Ap_Id Type Receptacle Occupant  Condition
sata0/0::dsk/c7t0d0 disk  connected configured ok
sata0/1::dsk/c7t1d0 disk  connected configured ok
sata0/2::dsk/c7t2d0 disk  connected configured ok
sata0/3::dsk/c7t3d0 disk  connected configured ok
```

How to Unconfigure a SATA Device

In general, SATA devices must be unconfigured before they can be removed and replaced. If you attempt to unconfigure a device that is part of an active ZFS storage pool, you will see an error message similar to the following:

```
# cfgadm -c unconfigure sata5/7
Unconfigure the device at: /devices/pci@2,0/pci1022...
This operation will suspend activity on the SATA device
Continue (yes/no)? y
```

cfgadm: Hardware specific failure: Failed to unconfig device at ap_id: /devices/pci@2,0/pci10...

1 Become an administrator.

2 Identify the device to be unconfigured.

```
# cfgadm -al | grep c7t0d0
sata0/0::dsk/c7t0d0 disk  connected configured ok
```

3 Unconfigure the device.

```
# cfgadm -c unconfigure sata0/0
```

If you attempt to unconfigure the device by specifying the individual device, you will see a message similar to the following:

```
# cfgadm -c unconfigure sata0/0::dsk/c7t0d0
do_control_ioctl: open failed: errno:2
```
cfgadm: Library error: Cannot open ap_id: /devices/pci@0,0/pci0...  
No such file or directory

4 Confirm that the device is unconfigured.

```
# cfgadm | grep sata0/0
sata0/0  disk  connected  unconfigured  ok
```

▼ How to Configure a SATA Device

After a disk is physically removed or replaced, it can be configured.

1 Become an administrator.

2 Configure the device.

```
# cfgadm -c configure sata0/0
```

3 Confirm that the device is configured.

```
# cfgadm | grep sata0/0
sata0/0::dsk/c7t0d0  disk  connected  configured  ok
```

Reconfiguration Coordination Manager (RCM) Script Overview

The Reconfiguration Coordination Manager (RCM) is the framework that manages the dynamic removal of system components. By using RCM, you can register and release system resources in an orderly manner.

You can use the new RCM script feature to write your own scripts to shut down your applications, or to cleanly release the devices from your applications during dynamic reconfiguration. The RCM framework launches a script automatically in response to a reconfiguration request, if the request impacts the resources that are registered by the script.

You can also release resources from applications manually before you dynamically remove the resource. Or, you can use the `cfgadm` command with the `-f` option to force a reconfiguration operation. However, this option might leave your applications in an unknown state. Also, the manual release of resources from applications commonly causes errors.

The RCM script feature simplifies and better controls the dynamic reconfiguration process. By creating an RCM script, you can do the following:

- Automatically release a device when you dynamically remove a device. This process also closes the device if the device is opened by an application.
- Run site-specific tasks when you dynamically remove a device from the system.
What Is an RCM Script?

- An executable shell script (Perl, sh, csh, or ksh) or binary program that the RCM daemon runs. Perl is the recommended language.
- A script that runs in its own address space by using the user ID of the script file owner.
- A script that is run by the RCM daemon when you use the `cfgadm` command to dynamically reconfigure a system resource.

What Can an RCM Script Do?

You can use an RCM script to release a device from an application when you dynamically remove a device. If the device is currently open, the RCM script also closes the device.

For example, an RCM script for a tape backup application can inform the tape backup application to close the tape drive or shut down the tape backup application.

How Does the RCM Script Process Work?

You can invoke an RCM script as follows:

```
$ script-name command [args ...]
```

An RCM script performs the following basic steps:

1. Takes the RCM command from command-line arguments.
2. Executes the command.
3. Writes the results to `stdout` as name-value pairs.
4. Exits with the appropriate exit status.

The RCM daemon runs one instance of a script at a time. For example, if a script is running, the RCM daemon does not run the same script until the first script exits.

RCM Script Commands

You must include the following RCM commands in an RCM script:

- `scriptinfo` – Gathers script information
- `register` – Registers interest in resources
- `resourceinfo` – Gathers resource information

You might include some or all of the following RCM commands:

- `queryremove` – Queries whether the resource can be released
- `preremove` – Releases the resource
- **post remove** – Provides post-resource removal notification
- **undo remove** – Undoes the actions done in **preremove**

For a complete description of these RCM commands, see `rcmscript(4)`.

**RCM Script Processing Environment**

When you dynamically remove a device, the RCM daemon runs the following:

- The script’s **register** command to gather the list of resources (device names) that are identified in the script.
- The script’s **query remove** and **preremove** commands prior to removing the resource if the script’s registered resources are affected by the dynamic remove operation.
- The script’s **post remove** command if the remove operation succeeds. However, if the remove operation fails, the RCM daemon runs the script’s **undo remove** command.

**RCM Script Tasks**

The following sections describe the RCM script tasks for application developers and system administrators.

**Application Developer RCM Script (Task Map)**

The following task map describes the tasks for an application developer who is creating an RCM script.

<table>
<thead>
<tr>
<th>Task Description</th>
<th>For Instructions</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Identify the resources your application uses.</td>
<td>Identify the resources (device names) your application uses that you could potentially dynamically remove.</td>
</tr>
<tr>
<td>2. Identify the commands to release the resource.</td>
<td>Identify the commands for notifying the application to cleanly release the resource from the application.</td>
</tr>
<tr>
<td>3. Identify the commands for post-removal of the resource.</td>
<td>Include the commands for notifying the application of the resource removal.</td>
</tr>
<tr>
<td>4. Identify the commands if the resource removal fails.</td>
<td>Include the commands for notifying the application of the available resource.</td>
</tr>
</tbody>
</table>
Task Description For Instructions

5. Write the RCM script. Write the RCM script based on the information identified in tasks 1-4. "Tape Backup RCM Script Example" on page 106

6. Install the RCM script. Add the script to the appropriate script directory. "How to Install an RCM Script" on page 105

7. Test the RCM script Test the script by running the script commands manually and by initiating a dynamic reconfiguration operation. "How to Test an RCM Script" on page 106

System Administrator RCM Script (Task Map)
The following task map describes the tasks for a system administrator who is creating an RCM script to do site customization.

<table>
<thead>
<tr>
<th>Task</th>
<th>Description</th>
<th>For Instructions</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>Identify the resources to be dynamically removed. Identify the resources (device names) to be potentially removed by using the <code>cfgadm -l</code> command.</td>
<td><code>cfgadm(1M)</code></td>
</tr>
<tr>
<td>2.</td>
<td>Identify the applications to be stopped. Identify the commands for stopping the applications cleanly.</td>
<td>Application documentation</td>
</tr>
<tr>
<td>3.</td>
<td>Identify the commands for pre-removal and post-removal of the resource. Identify the actions to be taken before and after the resource is removed.</td>
<td><code>rcmscript(4)</code></td>
</tr>
<tr>
<td>4.</td>
<td>Write the RCM script. Write the RCM script based on the information identified in tasks 1-3.</td>
<td>&quot;Tape Backup RCM Script Example&quot; on page 106</td>
</tr>
<tr>
<td>5.</td>
<td>Install the RCM script. Add the script to the appropriate script directory.</td>
<td>&quot;How to Install an RCM Script&quot; on page 105</td>
</tr>
<tr>
<td>6.</td>
<td>Test the RCM script. Test the script by running the script commands manually and by initiating a dynamic reconfiguration operation.</td>
<td>&quot;How to Test an RCM Script&quot; on page 106</td>
</tr>
</tbody>
</table>

Naming an RCM Script
A script must be named as `vendor,service` where the following applies:

*vendor* is the stock symbol of the vendor that provides the script, or any distinct name that identifies the vendor.
service  Is the name of the service that the script represents.

## Installing or Removing an RCM Script

You must be an administrator to install or remove an RCM script. Use this table to determine where you should install your RCM script.

<table>
<thead>
<tr>
<th>Directory Location</th>
<th>Script Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>/etc/rcm/scripts</td>
<td>Scripts for specific systems</td>
</tr>
<tr>
<td>/usr/platform/<code>uname -i</code>/lib/rcm/scripts</td>
<td>Scripts for a specific hardware implementation</td>
</tr>
<tr>
<td>/usr/platform/<code>uname -m</code>/lib/rcm/scripts</td>
<td>Scripts for a specific hardware class</td>
</tr>
<tr>
<td>/usr/lib/rcm/scripts</td>
<td>Scripts for any hardware</td>
</tr>
</tbody>
</table>

### How to Install an RCM Script

1. Become an administrator.

2. Copy the script to the appropriate directory.
   
   See Table 6–1.
   
   For example:
   ```
   # cp ABC,sample.pl /usr/lib/rcm/scripts
   ```

3. Change the user ID and the group ID of the script to the desired values.
   ```
   # chown user:group /usr/lib/rcm/scripts/ABC,sample.pl
   ```

4. Send SIGHUP to the RCM daemon.
   ```
   # pkill -HUP -x -u root rcm_daemon
   ```

### How to Remove an RCM Script

1. Become an administrator.

2. Remove the script from the RCM script directory.
   
   For example:
   ```
   # rm /usr/lib/rcm/scripts/ABC,sample.pl
   ```
3 Send SIGHUP to the RCM daemon.
   # pkill -HUP -x -u root rcm_daemon

▼ How to Test an RCM Script

1 Set environment variables, such as RCM_ENV_FORCE, in the command-line shell before running your script.
   For example, in the Korn shell, use the following:
   $ export RCM_ENV_FORCE=TRUE

2 Test the script by running the script commands manually from the command line.
   For example:
   $ script-name scriptinfo
   $ script-name register
   $ script-name preremove resource-name
   $ script-name postremove resource-name

3 Make sure that each RCM script command in your script prints appropriate output to stdout.

4 Install the script in the appropriate script directory.
   For more information, see “How to Install an RCM Script” on page 105.

5 Test the script by initiating a dynamic remove operation.
   For example, assume your script registers the device, /dev/dsk/c1t0d0s0. Try these commands.
   $ cfgadm -c unconfigure cl::dsk/c1t0d0
   $ cfgadm -f -c unconfigure cl::dsk/c1t0d0
   $ cfgadm -c configure cl::dsk/c1t0d0

   Caution – Make sure that you are familiar with these commands because they can alter the state of the system and cause system failures.

Tape Backup RCM Script Example

This example illustrates how to use an RCM script for tape backups.

What the Tape Backup RCM Script Does

The tape backup RCM script performs the following steps:
1. Sets up a dispatch table of RCM commands.
2. Calls the dispatch routine that corresponds to the specified RCM command and exits with status 2 for unimplemented RCM commands.

3. Sets up the scriptinfo section.

   ```
   rcm_script_func_info=Tape backup appl script for DR
   ```

4. Registers all tape drives in the system by printing all tape drive device names to stdout.

   ```
   rcm_resource_name=/dev/rmt/$f
   ```

   If an error occurs, the script prints the error information to stdout.

   ```
   rcm_failure_reason=$errmsg
   ```

5. Sets up the resource information for the tape device.

   ```
   rcm_resource_usage_info=Backup Tape Unit Number $unit
   ```

6. Sets up the preremove information by checking if the backup application is using the device. If the backup application is not using the device, the dynamic reconfiguration operation continues. If the backup application is using the device, the script checks `RCM_ENV_FORCE`. If `RCM_ENV_FORCE` is set to `FALSE`, the script denies the dynamic reconfiguration operation and prints the following message:

   ```
   rcm_failure_reason=tape backup in progress pid=...
   ```

   If `RCM_ENV_FORCE` is set to `TRUE`, the backup application is stopped, and the reconfiguration operation proceeds.

### Outcomes of the Tape Backup Reconfiguration Scenarios

Here are the various outcomes if you use the `cfgadm` command to remove a tape device without the RCM script.

- If you use the `cfgadm` command and the backup application is not using the tape device, the operation succeeds.
- If you use the `cfgadm` command and the backup application is using the tape device, the operation fails.

Here are the various outcomes if you use the `cfgadm` command to remove a tape device with the RCM script.

- If you use the `cfgadm` command and the backup application is not using the tape device, the operation succeeds.
- If you use the `cfgadm` command without the `-f` option and the backup application is using the tape device, the operation fails with an error message similar to the following:

   ```
   tape backup in progress pid=...
   ```

- If you use the `cfgadm -f` command and the backup application is using the tape device, the script stops the backup application and the `cfgadm` operation succeeds.
Example—Tape Backup RCM Script

#!/usr/bin/perl -w
#
# A sample site customization RCM script.
#
# When RCM_ENV_FORCE is FALSE this script indicates to RCM that it cannot
# release the tape drive when the tape drive is being used for backup.
# When RCM_ENV_FORCE is TRUE this script allows DR removing a tape drive
# when the tape drive is being used for backup by killing the tape
# backup application.
#
use strict;
my ($cmd, %dispatch);
$cmd = shift(@ARGV);

# dispatch table for RCM commands
%dispatch = {
    "scriptinfo" => \&do_scriptinfo,
    "register" => \&do_register,
    "resourceinfo" => \&do_resourceinfo,
    "queryremove" => \&do_preremove,
    "preremove" => \&do_preremove
};

if (defined($dispatch{$cmd})) {
    \$dispatch{\$cmd};
} else {
    exit (2);
}

sub do_scriptinfo {
    print "rcm_script_version=1\n";
    print "rcm_script_func_info=Tape backup appl script for DR\n";
    exit (0);
}

sub do_register {
    my ($dir, $f, $errmsg);
    $dir = opendir(RMT, "/dev/rmt");
    if (!$dir) {
        $errmsg = "Unable to open /dev/rmt directory: $!";
        print "rcm_failure_reason=$errmsg\n";
        exit (1);
    }

    while ($f = readdir(RMT)) {
        # ignore hidden files and multiple names for the same device
        if (!(\$f =~ /^\./) &\& (\$f =~ /^[\-0-9]+$/)) {
            print "rcm_resource_name=/dev/rmt/$f\n";
        }
    }
}
closedir(RMT);
exit (0);
}

sub do_resourceinfo
{
    my ($rsrc, $unit);
    $rsrc = shift(@ARGV);
    if ($rsrc =~ /^/dev/rmt/\([0-9]+\)/) {
        $unit = $1;
        print "rcm_resource_usage_info=Backup Tape Unit Number $unit\n";
        exit (0);
    } else {
        print "rcm_failure_reason=Unknown tape device!\n";
        exit (1);
    }
}

sub do_preremove
{
    my ($rsrc);
    $rsrc = shift(@ARGV);
    # check if backup application is using this resource
    #if (the backup application is not running on $rsrc) {
    #    # allow the DR to continue
    #    exit (0);
    #}
    #
    # If RCM_ENV_FORCE is FALSE deny the operation.
    # If RCM_ENV_FORCE is TRUE kill the backup application in order
    # to allow the DR operation to proceed
    #
    if ($ENV{RCM_ENV_FORCE} eq 'TRUE') {
        if ($cmd eq 'preremove') {
            # kill the tape backup application
            exit (0);
        } else {
            # indicate that the tape drive can not be released
            # since the device is being used for backup by the
            # tape backup application
            print "rcm_failure_reason=tape backup in progress pid=...\n"
        }
    } else {
        #
        exit (3);
    }
}
This chapter provides an overview of Universal Serial Bus (USB) devices in the Oracle Solaris OS.

This is a list of the overview information in this chapter.

- “What's New in USB Devices?” on page 112
- “Overview of USB Devices” on page 115
- “About USB in the Oracle Solaris OS” on page 120

For recent information about USB devices, go to the following site:


For general information about USB devices, go to the following site:

http://developers.sun.com/solaris/developer/support/driver/usb.html

For step-by-step instructions on using USB devices in the Oracle Solaris OS, see Chapter 8, “Using USB Devices (Tasks).”

For general information about dynamic reconfiguration and hot-plugging, see Chapter 6, “Dynamically Configuring Devices (Tasks).”

For information on configuring USB printers, see Chapter 15, “Setting Up and Administering Printers by Using CUPS (Tasks),” in Oracle Solaris Administration: Common Tasks.
What's New in USB Devices?

The following section describes new USB features in the Oracle Solaris release.

- "USB Interface Association Descriptor Support" on page 112
- "EHCI Isochronous Transfer Support" on page 112
- "USB Device Hotpluggable Behavior" on page 112
- "x86: Support for USB CDs and DVDs in GRUB-Based Booting" on page 113
- "USB Virtual Keyboard and Mouse Support" on page 113

**USB Interface Association Descriptor Support**

**Oracle Solaris 11**: A USB device node type, IA node, is created for Interface Association Descriptor (IAD) support. This feature means that a driver might support multiple interfaces for the same device, such as the video and audio interfaces of a webcam. If no driver is found for an IA node, a nexus driver, `usb_ia`, is bound to the IA node to create the interface nodes. For more information, see `usb_ia(7D)`.

**EHCI Isochronous Transfer Support**

**Oracle Solaris 11**: USB EHCl host controller driver provides isochronous transfer support for USB 2.0 or high-speed isochronous devices. For more information, see `usb_isoc_request(9S)`.

**USB Device Hotpluggable Behavior**

**Oracle Solaris 11**: This Solaris release introduces a new device attribute, `hotpluggable`, to identify those devices that can be connected or disconnected without rebooting the system and configured or unconfigured automatically without user intervention. All USB and 1394 devices are identified as hotpluggable devices to gain those benefits described in "Using USB Mass Storage Devices" on page 129. In addition, non-removable media USB and 1394 devices are no longer identified as removable-media devices and no longer have a `removable-media` attribute.

The changes are primarily made at the kernel level to improve support for non-removable media USB and 1394 devices, and improve the performance for those devices. However, these changes do not impact the use of these devices. For example, the responsibility of mounting and unmounting these devices is controlled by `rmvolmgr`. From a user's perspective, the only visible changes are the hotpluggable and removable-media attributes of a device.

**USB and 1394 (FireWire) Support**

Both non-removable USB storage devices and 1394 mass storage devices are identified as hotpluggable devices at the driver level. This behavior means that these devices can be connected or disconnected without rebooting the system and configured or unconfigured...
automatically without intervention. These changes are made at the kernel level and do not impact the use of these devices. For example, the responsibility of mounting and unmounting these devices is controlled by the removable media management services.

In addition, non-removable USB devices and 1394 mass storage devices can be accessed and labeled by using the `format` utility. However, you can override the new hotpluggable behavior of these devices by setting the `remvalue` to true in the `/etc/driver/drv/scsa2usb.conf` file. Setting this parameter to true means that the device is treated as a removable media device at the driver level, if that behavior is preferred.

For more information on using these devices, see `scsa1394(7D)` and “Using USB Mass Storage Devices (Task Map)” on page 128.

**x86: Support for USB CDs and DVDs in GRUB-Based Booting**

**Oracle Solaris 11**: You can use the following USB features in the GRUB-based booting environment:

- Installing from USB CD or DVD drives
- Booting from USB storage devices. You must install the Solaris release on the USB drive before you can boot from it.

For more information about booting, see [Booting and Shutting Down Oracle Solaris on SPARC Platforms](#) or [Booting and Shutting Down Oracle Solaris on x86 Platforms](#).

**USB Virtual Keyboard and Mouse Support**

**Oracle Solaris 11**: USB virtual keyboard and mouse support enables you to hook up multiple keyboards and multiple mice, where the set of keyboards or mice behave as one virtual keyboard or mouse. This means that the input of each physical device is coalesced into a single input stream. For example, if you type SHIFT on one keyboard and A on another, the character echoed is an uppercase A.

Also supported is the ability to add a USB keyboard or mouse to a laptop and have these devices work as one device with the laptop's PS/2 keyboard and pad.

In addition, support for barcode readers is provided by the virtual keyboard and mouse feature.

For more information, refer to `virtualkm(7D)`.
Oracle Solaris Support for USB Devices

Support for the following USB 1.1 and USB 2.0 devices is provided on both SPARC and x86 based systems, except where indicated:

- **Audio devices** – Only USB 1.x devices are supported on a USB 2.0 hub.
  - No USB 2.0 audio devices are supported.
  - A USB 1.x audio device that is connected to a USB 2.0 hub, which is connected to a USB 2.0 port, can be used in the Oracle Solaris 11 releases only. For more information, see `usb_ac(7D)` and `usb_as(7D)`.
  - Devices that are not supported by a USB driver might have `libusb` applications such as `gphoto2`, `gtkam`, and `pilotlink`. For more information, refer to `/usr/sfw/share/doc/libusb/libusb.txt`.
- **Generic USB driver** (`ugen(7D)`)
- **HID devices** (keyboard and mouse devices, `hid(7D)`)
- **Hubs** (`hubd(7D)`)
- **Printers**
- **Serial devices** (Edgeport (`usbser_edge(7D)`), Prolific (`usbsprl(7D)`), Keyspan (`usbsksp(7D)`))
- **Storage devices** (`scsa2usb(7D)`)
- **User-space USB device management library** (`libusb(3LIB)`)

Use the following table to identify Solaris support information for specific USB 1.1 and USB 2.0 devices.

**Notes:**

- Only USB 1.x audio devices are supported. No USB 2.0 audio devices are supported.
- A USB 1.x audio device that is connected to a USB 2.0 hub, which is connected to a USB 2.0 port, can be used. For more information, see `usb_ac(7D)` and `usb_as(7D)`.
- Devices that are not supported by a USB driver might have `libusb` applications such as `gphoto2`, `gtkam`, and `pilotlink`. For more information, refer to `/usr/sfw/share/doc/libusb/libusb.txt`.

For task information associated with mass storage devices, see Chapter 8, “Using USB Devices (Tasks).”

For more information about `ugen`, see “USB Driver Enhancements” on page 117.
Overview of USB Devices

Universal Serial Bus (USB) was developed by the PC industry to provide a low-cost solution for attaching peripheral devices, such as keyboards, mouse devices, and printers, to a system.

USB connectors are designed to fit only one type of cable, in one way. The primary design motivation for USB was to alleviate the need for multiple connector types for different devices. This design reduces the clutter on the back panel of a system.

Devices connect to USB ports on external USB hubs, or on a root hub that is located on the computer itself. Since hubs have several ports, several branches of a device tree can stem from a hub.

For more information, see usba(7D) or go to the following site:

http://www.usb.org/home

Commonly Used USB Acronyms

The following table describes the USB acronyms that are used in the Oracle Solaris OS. For a complete description of USB components and acronyms, go to:

http://www.usb.org/home

<table>
<thead>
<tr>
<th>Acronym</th>
<th>Definition</th>
<th>For More Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>UGEN</td>
<td>USB generic driver</td>
<td>ugen(7D)</td>
</tr>
<tr>
<td>USB</td>
<td>Universal Serial Bus</td>
<td>usb(7D)</td>
</tr>
<tr>
<td>USBA</td>
<td>Universal Serial Bus Architecture (Solaris)</td>
<td>usba(7D)</td>
</tr>
<tr>
<td>USBAI</td>
<td>USBA Client Driver Interface (Solaris)</td>
<td>N/A</td>
</tr>
<tr>
<td>HCD</td>
<td>USB host controller driver</td>
<td>N/A</td>
</tr>
<tr>
<td>EHCI</td>
<td>Enhanced Host Controller Interface</td>
<td>ehci(7D)</td>
</tr>
<tr>
<td>OHCI</td>
<td>Open Host Controller Interface</td>
<td>ohci(7D)</td>
</tr>
<tr>
<td>UHCI</td>
<td>Universal Host Controller Interface</td>
<td>uhci(7D)</td>
</tr>
</tbody>
</table>

USB Bus Description

The USB specification is openly available and free of royalties. The specification defines the electrical and mechanical interfaces of the bus and the connectors.
USB employs a topology in which hubs provide attachment points for USB devices. The host controller contains the root hub, which is the origin of all USB ports in the system. For more information about hubs, see “USB Host Controller and Hubs” on page 123.

**Figure 7–1** shows a system with three active USB ports. The first USB port connects a USB memory stick. The second USB port connects an external hub, which in turn, connects a cd rw device and a composite keyboard/mouse device. As a *composite device*, this keyboard contains a USB controller, which operates both the keyboard and an attached mouse. The keyboard and the mouse share a common USB bus address because they are directed by the same USB controller.

**Figure 7–1** also shows an example of a hub and a printer as a *compound device*. The hub is an external hub that is enclosed in the same casing as the printer. The printer is permanently connected to the hub. The hub and printer have separate USB bus addresses.

The device tree path name for some of the devices that are displayed in **Figure 7–1** are listed here.

- Memory stick: `/pci@1f,4000/usb@5/storage@1`
- Keyboard: `/pci@1f,4000/usb@5/hub@2/device@1/keyboard@0`
- Mouse: `/pci@1f,4000/usb@5/hub@2/device@1/mouse@1`
USB Devices and Drivers

USB devices with similar attributes and services are grouped into device classes. Each device class has a corresponding driver. Devices within a class are managed by the same device driver pair. However, the USB specification also allows for vendor-specific devices that are not part of a specific class.

The Human Interface Device (HID) class contains devices that are user-controlled such as the following devices:

- Keyboards
- Mouse devices
- Joysticks

The Communication Device class includes the following devices:

- Modems
- Ethernet adapters

Other device classes include the following classes:

- Audio
- Monitor
- Printer
- Storage Device

Each USB device contains descriptors that reflect the class of the device. A device class specifies how its members should behave in configuration and data transfer. You can obtain additional class information from:

http://www.usb.org/home

For more information about USB devices supported in the Oracle Solaris release, see usb(7D).

USB Driver Enhancements

The following USB driver enhancements are included.

- **USB CDC ACM device support** – The acm driver can work with devices that are compliant with the USB Communication Class Device specification’s Abstract Control Model and some PCMCIA cards that have modem capabilities.

  The pppd daemon can access these devices through the /dev/term/[0-9]* entries. For more information, see pppd(1M).
For more information, see `usbsacm(7D)`.

- **Generic USB driver** – USB devices can now be accessed and manipulated by applications using standard UNIX `read(2)` and `write(2)` system calls, and without writing a special kernel driver. Additional features include:
  - Applications have access to raw device data and device status.
  - The driver supports control, bulk, and interrupt (in and out) transfers.

The `ugen` driver no longer needs to bind explicitly to a device. By default, `usb_midi` binds to devices that lack a class driver and exports a `ugen` interface that works with `libusb`. For example, you can plug in a USB camera that is not a mass-storage device and use a `libusb` application to access it. In addition, both `scsa2usb` and `usbprn` drivers export `ugen` interfaces and `libusb` applications can be used on these classes of devices directly.

For more information, refer to `ugen(7D)`.

- **USB serial driver support**
  - Digi Edgeport USB support – The Edgeport USB driver only works with Edgeport devices and not with other USB serial devices.
    - New devices are accessed as `/dev/term/[0-9]*` and `/dev/cua/[0-9]*`.
    - USB serial ports are usable as any other serial port would be, except that they cannot serve as a local serial console. The fact that their data is run through a USB port is transparent to the user.

  For more information, see `usbser_edge(7D)`, or go to the following sites:
  - [http://www.digi.com](http://www.digi.com)

  - Keyspan – The Keyspan USB serial driver only works with Keyspan devices, which currently supports the USA-19HS and USA-49WLC models.
    For more information, see `usbsksp(7D)`.
  
  - Prolific – The Prolific USB serial driver only works with devices based on the PL2303 chipset.
    For more information, see `usbsprl(7D)`.

  For more information about the USB to serial devices support, go to the following site:


- **Documentation and binary support for user-written kernel and userland drivers** – For up-to-date information on USB driver development, go to:
  - Chapter 20, “USB Drivers,” in *Writing Device Drivers*
  - Appendix C, “Making a Device Driver 64-Bit Ready,” in *Writing Device Drivers*
The EHCI, OHCI, and UHCI Drivers

Features of the EHCI driver include:

- Complies with enhanced host controller interface that supports USB 2.0.
- Supports high-speed control, bulk, interrupt, and isochronous transfers.
- The USB 2.0 chip has one EHCI controller and one or more OHCI or UHCI controllers.
- A USB 1.1 device is dynamically assigned to the OHCI or UHCI controller when it is plugged in. A USB 2.0 device is dynamically assigned to the EHCI controller when it is plugged in.

Use the `prtconf` command output to identify whether your system supports USB 1.1 or USB 2.0 devices. For example:

```
# prtconf -D | egrep "ehci|ohci|uhci"
```

If your `prtconf` output identifies an EHCI controller, your system supports USB 2.0 devices.

If your `prtconf` output identifies an OHCI or UHCI controller, your system supports USB 1.1 devices.

Oracle Solaris USB Architecture (USBA)

USB devices can be represented as two levels of device tree nodes. A device node represents the entire USB device. One or more child interface nodes represent the individual USB interfaces on the device.

Driver binding is achieved by using the compatible name properties. For more information, refer to 3.2.2.1 of the IEEE 1275 USB binding and Writing Device Drivers. A driver can either bind to the entire device and control all the interfaces, or can bind to just one interface. If no vendor or class driver claims the entire device, a generic USB multi-interface driver is bound to the device-level node. This driver attempts to bind drivers to each interface by using compatible names properties, as defined in section 3.3.2.1 of the IEEE 1275 binding specification.

The Oracle Solaris USB Architecture (USBA) adheres to the USB 1.1 and USB 2.0 specifications and is part of the Oracle Solaris Device Driver Interface (DDI). The USBA model is similar to Oracle Common SCSI Architecture (SCSA). As the following figure shows, the USBA is a thin layer that provides a generic USB transport-layer abstraction to client drivers, providing them with services that implement core generic USB functionality.
About USB in the Oracle Solaris OS

This section describes information you should know about USB in the Oracle Solaris OS.

USB 2.0 Features

The following USB 2.0 features are included:

- **Better performance** - Increased data throughput for devices connected to USB 2.0 controllers, up to 40 times faster than USB 1.1 devices.
  
  You can take advantage of the high-speed USB protocol when accessing high-speed USB devices, such as DVDs and hard disks.

- **Backward Compatibility** - Compatibility with 1.0 and 1.1 devices and drivers so that you can use the same cables, connectors, and software interfaces.

For a description of USB devices and terminology, see “Overview of USB Devices” on page 115.

USB 2.0 Device Features and Compatibility Issues

USB 2.0 devices are defined as high-speed devices that follow the USB 2.0 specification. You can refer to the USB 2.0 specification at [http://www.usb.org/home](http://www.usb.org/home).

To identify the speed of your USB device, check the `/var/adm/messages` file for messages similar to the following:

```
Dec 13 17:05:57 mysystem usba: [ID 912658 kern.info] USB 2.0 device (usb50d,249) operating at hi speed (USB 2.x) on USB 2.0 external hub: storage@4, scsi2usb0 at bus address 4
```

Here are some of the USB devices that are supported in this Oracle Solaris release:
Mass storage devices, such as CD-RWs, hard disks, DVDs, digital cameras, diskettes, tape drives, memory sticks, and multi-format card readers
- Keyboards and mouse devices
- Audio devices, such as speakers and microphones

For a full listing of USB devices that have been verified on the Oracle Solaris release, go to:


Additional storage devices might work by modifying the scsa2usb.conf file. For more information, see scsa2usb(7D).

Oracle Solaris USB 2.0 device support includes the following features:
- Increased USB bus speed from 12 MB/sec to 480 MB/sec. This increase means devices that support the USB 2.0 specification can run significantly faster than their USB 1.1 counterparts, when they are connected to a USB 2.0 port.
  - A USB 2.0 port might be one of the following possibilities:
    - A port on a USB 2.0 PCI card
    - A port on a USB 2.0 hub that is connected to USB 2.0 port
    - A port on a SPARC or x86 computer motherboard
  - A USB 2.0 PCI card might be needed for older SPARC platforms.
- For a list of USB 2.0 PCI cards that have been verified for the Oracle Solaris release, go to:
- USB 1.1 devices work as they have in the past, even if you have both USB 1.1 and USB 2.0 devices on the same system.
- While USB 2.0 devices operate on a USB 1.x port, their performance is significantly better when they are connected to a USB 2.0 port.
- A USB 2.0 host controller has one high-speed Enhanced Host Controller Interface (EHCI) and one or more OpenHCI Host Controller Interface (OHCI) or Universal Host Controller Interface (UHCI) embedded controllers. Devices connected to a USB 2.0 port are dynamically assigned to either an EHCI or OHCI controller, depending on whether they support USB 2.0.
Note – USB 2.0 storage devices that are connected to a port on a USB 2.0 PCI card, and that were used with a prior Solaris release in the same hardware configuration, can change device names after upgrading to this release. This change occurs because these devices are now seen as USB 2.0 devices and are taken over by the EHCI controller. The controller number, \( w \) in \(/dev/[r]dsk/cwtxysz\), is changed for these devices.

Also note that the speed of a USB device is limited to what the parent port can support. For example, if a USB 2.0 external hub is followed by a USB 1.x hub and a USB 2.0 device downstream, devices that are connected to the USB 2.0 external hub run at full speed and not high speed.

For more information on USB 2.0 device support, see \texttt{ehci(7D)} and \texttt{usba(7D)}.

**Bus-Powered Devices**

Bus-powered hubs use power from the USB bus to which they are connected, to power devices connected to them. Special care must be taken to not overload these hubs, because the power these hubs offer to their downstream devices is limited.

Power budgeting is implemented for USB devices. This feature has the following limitations:

- Cascading two bus-powered hubs is prohibited.
- Each bus-powered hub is allowed a maximum of 100 mA only for each port.
- Only self-powered or low bus-powered devices are allowed to connect to a bus-powered hub. High bus-powered devices are denied the connection. Some hubs or devices can report a false power source, such that the connection might be unpredictable.

**USB Keyboards and Mouse Devices**

Keep the following issues in mind when using USB keyboards and mouse devices:

- Do not move the keyboard and mouse during a reboot or at the ok prompt on a SPARC system. You can move the keyboard and mouse to another hub at any time after a system reboot. After you plug in a keyboard and mouse, they are fully functional again.
- The keys just to the left of the keypad might not function on some third-party USB keyboards.
- **SPARC** – Keep the following issues in mind when using USB keyboards and mouse devices on SPARC systems:
The power key on a USB keyboard behaves differently than the power key on the type 5 keyboard. On a USB keyboard, you can suspend or shut down the system by using the SUSPEND/SHUTDOWN key. However, you cannot use that key to power up the system.

Before the boot process finishes, the OpenBoot PROM (OBP) limits keyboard and mouse devices to the motherboard root hub ports only.

USB keyboard and mouse devices cannot be used simultaneously with Type 3, 4, or 5 keyboards on legacy SPARC systems.

For information about multiple keyboard and mouse device support, see \texttt{virtualkm(7D)}.

**USB Wheel Mouse Support**

The following wheel mouse features are supported:

- Support for more than 3 buttons is available on USB or PS/2 mouse devices.
- Wheel mouse scrolling is available on a USB or PS/2 mouse device. This support means that rolling the wheel on a USB or a PS/2 mouse results in a scroll in the application or window under mouse focus. StarOffice, Firefox, and GNOME applications support wheel mouse scrolling. However, other applications might not support wheel mouse scrolling.

**USB Host Controller and Hubs**

A USB hub is responsible for the following:

- Monitoring the insertion or removal of a device on its ports
- Power managing individual devices on its ports
- Controlling power to its ports

The USB host controller has an embedded hub called the *root hub*. The ports that are visible at the system’s back panel are the ports of the root hub. The USB host controller is responsible for the following:

- Directing the USB bus. Individual devices cannot arbitrate for the bus.
- Polling the devices by using a polling interval that is determined by the device. The device is assumed to have sufficient buffering to account for the time between the polls.
- Sending data between the USB host controller and its attached devices. Peer-to-peer communication is not supported.
USB Hub Devices

- Do not cascade hubs beyond four levels on either SPARC based systems or x86 based systems. On SPARC systems, the OpenBoot PROM cannot reliably probe beyond four levels of devices.
- Do not plug a bus-powered hub into another bus-powered hub in a cascading style. A bus-powered hub does not have its own power supply.
- Do not connect a device that requires a large amount of power to a bus-powered hub. These devices might be denied connection to bus-powered hubs or might drain the hub of power for other devices. An example of such a device is a USB diskette device.

SPARC: USB Power Management

Suspending and resuming USB devices is fully supported on SPARC systems. However, do not suspend a device that is busy and never remove a device when the system is powered off under a suspend shutdown.

The USB framework makes a best effort to power manage all devices on SPARC based systems with power management enabled. Power managing a USB device means that the hub driver suspends the port to which the device is connected. Devices that support remote wake up can notify the system to wake up everything in the device’s path so that the device can be used. The host system could also wake up the device if an application sends an I/O to the device.

All HID devices (keyboard, mouse, hub, and storage devices), hub devices, and storage devices are power managed by default if they support remote wake-up capability. A USB printer is power managed only between two print jobs. Devices that are managed by the generic USB driver (UGEN) are power managed only when they are closed.

When power management is running to reduce power consumption, USB leaf devices are powered down first. After all devices that are connected to a hub’s ports are powered down, the hub is powered down after some delay. To achieve the most efficient power management, do not cascade many hubs.

For information about using the SUSPEND/SHUTDOWN key on SPARC systems, see “USB Keyboards and Mouse Devices” on page 122.

Guidelines for USB Cables

Keep the following guidelines in mind when connecting USB cables:

- Always use USB 2.0 compliant, fully rated (480 Mbit/sec) 20/28 AWG cables for connecting USB 2.0 devices.
- The maximum cable length that is supported is 5 meters.
• Do not use cable extenders. For best results, use a self-powered hub to extend cable length.

For more information, go to:

http://www.usb.org/about/faq/ans5
This chapter provides step-by-step instructions for using USB devices in the Oracle Solaris OS. This chapter specifically covers how to use USB devices with HAL services in the Oracle Solaris 11 release.

For information on the procedures associated with using USB devices, see the following:
- “Managing USB Devices in the Oracle Solaris OS (Roadmap)” on page 127
- “Using USB Mass Storage Devices (Task Map)” on page 128
- “Using USB Audio Devices (Task Map)” on page 146
- “Hot-Plugging USB Devices With the cfgadm Command (Task Map)” on page 150

For recent information about USB devices, go to the following site:

For overview information about using USB devices, see Chapter 7, “Using USB Devices (Overview).”

**Managing USB Devices in the Oracle Solaris OS (Roadmap)**

Use this road map to identify all the tasks for managing USB devices in the Oracle Solaris OS. Each task points to a series of additional tasks such as using USB devices, hot-plugging USB devices, and adding USB audio devices.

For information about using USB components in the Oracle Solaris OS, see “About USB in the Oracle Solaris OS” on page 120.
### Using USB Mass Storage Devices (Task Map)

<table>
<thead>
<tr>
<th>Task</th>
<th>Description</th>
<th>For Instructions</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Use USB mass storage devices.</strong></td>
<td>A USB mass storage device must contain a file system before you can add data to it. In addition, a USB diskette must be formatted before file systems can be created and mounted on them. This section also describes how to physically add or remove USB devices from your system.</td>
<td>&quot;Using USB Mass Storage Devices (Task Map)&quot; on page 128</td>
</tr>
<tr>
<td><strong>Add USB audio devices.</strong></td>
<td>Use this task map to identify tasks associated with adding USB audio devices.</td>
<td>&quot;Using USB Audio Devices (Task Map)&quot; on page 146</td>
</tr>
<tr>
<td><strong>Add or remove USB devices to and from your system with the cfgadm command.</strong></td>
<td>You can logically add or remove USB devices to and from your system with the cfgadm command.</td>
<td>&quot;Hot-Plugging USB Devices With the cfgadm Command (Task Map)&quot; on page 150</td>
</tr>
</tbody>
</table>

---

<table>
<thead>
<tr>
<th>Task</th>
<th>Description</th>
<th>For Instructions</th>
</tr>
</thead>
</table>
| **Add or remove a USB mass storage device.** | Select one of the following to add a USB mass storage device:  
   - Add a USB mass storage device.  
   - Add a USB camera to access digital images.  
   - Remove a USB mass storage device. | "How to Add a USB Mass Storage Device" on page 132  
   "Preparing to Use a USB Mass Storage Device" on page 135  
   "How to Remove a USB Mass Storage Device" on page 134 |
<p>| <strong>Add a non-compliant USB mass storage device.</strong> | Add a non-compliant USB mass storage device by adding an entry to /etc/scsa2usb.conf. | &quot;How to Use a Non-Complaint USB Mass Storage Device&quot; on page 131 |
| <strong>Prepare to use a USB mass storage device.</strong> | Prepare to use a USB mass storage device. | &quot;Preparing to Use a USB Mass Storage Device&quot; on page 135 |
| <strong>Display USB device information.</strong> | Display information about USB devices. | &quot;How to Display USB Device Information&quot; on page 135 |
| <strong>Create a file system on a USB mass storage device.</strong> | You must create a file system on a device before you can put data on it. | &quot;How to Create a File System on a USB Mass Storage Device&quot; on page 136 |</p>
<table>
<thead>
<tr>
<th>Task</th>
<th>Description</th>
<th>For Instructions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Modify partitions and create a file system on a USB mass storage</td>
<td>You might need to modify existing partitions before creating file systems on a USB mass storage device.</td>
<td>“How to Modify Partitions and Create a PCFS File System on a USB Mass Storage Device” on page 138</td>
</tr>
<tr>
<td>device.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Create a Solaris partition and modify the slices on a USB mass</td>
<td>You will need to create a file with slice information before using the</td>
<td>“How to Create a Solaris Partition and Modify the Slices on a USB Mass Storage Device” on page 141</td>
</tr>
<tr>
<td>storage device.</td>
<td>rmformat to modify the slice information.</td>
<td></td>
</tr>
<tr>
<td>Mount a USB mass storage device.</td>
<td>Mount a USB mass storage device.</td>
<td>“How to Mount or Unmount a USB Mass Storage Device” on page 143</td>
</tr>
<tr>
<td>(Optional) Disable a USB device driver.</td>
<td>Disable a USB device driver if you do not want USB support on your system.</td>
<td>“How to Disable Specific USB Drivers” on page 145</td>
</tr>
<tr>
<td>(Optional) Remove unused USB device links.</td>
<td>Remove USB device links with the devfsadm command.</td>
<td>“How to Remove Unused USB Device Links” on page 146</td>
</tr>
</tbody>
</table>

**Using USB Mass Storage Devices**

The following USB removable mass storage devices are supported:

- CD-RWs
- Hard disks
- DVDs
- Digital cameras
- Diskette devices
- SmartMedia and CompactFlash devices

For information about using USB mass storage devices with the Oracle Solaris ZFS file system, see “What’s New in USB Devices?” on page 112.

For a complete list of USB devices that are supported in the Oracle Solaris OS, see:


In previous Solaris releases, all USB storage devices were identified as removable media devices, which provides many of the following advantages, including automatic mounting. In the Oracle Solaris 11 release, USB mass storage devices are identified as hotpluggable devices but also enjoy the advantages of USB removable devices that are listed below. For more information about the hotpluggable behavior, see “USB and 1394 (FireWire) Support” on page 112.

- A hot-pluggable device is automatically mounted.
- USB storage devices with standard MS-DOS or Windows (FAT) file systems are supported.
You can use the user-friendly `rmformat` command to create slices. You can also use the `fdisk` command to partition a USB device, but never use the `format` utility or the `rmformat -F` command to physically format a USB drive.

Use the `rmformat` command to display all USB devices with media inserted. For example, see "How to Display USB Device Information" on page 135.

Non-root users can now access USB storage devices, since the `mount` command is no longer needed. The device is automatically mounted and is available under the `/media` directory.

These devices can be managed with or without removable media services.

Disks with FAT file systems can be mounted and accessed. For example:

```
mount -F pcfs /dev/dsk/c2t0d0s0:c /mnt
```

All USB storage devices are now power managed, except for those that support LOG SENSE pages. Devices with LOG SENSE pages are usually SCSI drives connected through a USB-to-SCSI bridge device.

Applications might work differently with USB mass storage devices. Keep the following issues in mind when using applications with USB storage devices:

- Applications might make incorrect assumptions about the size of the media since only smaller devices like diskettes were removable previously.
- Requests by applications to eject media on devices where this would be inapplicable, such as a hard drive, will succeed and do nothing.
- If you prefer the behavior in previous Solaris releases where all USB mass storage were treated as removable media devices, then you can force the old behavior by updating the `/etc/driver/drv/scsa2usb.conf` file.

For more information on using USB mass storage devices, see `scsa2usb(7D)`.

### Using USB Diskette Devices

USB diskette devices appear as removable media devices. USB diskette devices are not managed by the `fd` (floppy) driver. Applications that issue `ioctl(2)` calls intended for the `fd` (native floppy) driver will fail. Applications that issue only `read(2)` and `write(2)` calls will succeed. Other applications, such as SunPCI and `rmformat`, also succeed.

The USB diskette device is identified as a SCSI removable media device. The device is available for access under the `/media` directory.

For more information on how to use USB diskette devices, see Chapter 1, “Managing Removable Media (Overview).”
Using Non-Compliant USB Mass Storage Devices

Some devices might be supported by the USB mass storage driver even though they do not identify themselves as compliant with the USB mass storage class or identify themselves incorrectly. The scsa2usb.conf file contains an attribute-override list that lists the vendor ID, product ID, and revision for matching mass storage devices, as well as fields for overriding the default device attributes. The entries in this list are commented out by default. These entries can be copied and uncommented to enable support of particular devices.

If you connect a USB mass storage device to a system running this Solaris release and the system is unable to use it, you can check the /kernel/drv/scsa2usb.conf file to see if there is a matching, commented entry for this device. Follow the information given in the scsa2usb.conf file to see if a particular device can be supported by using the override information.

For a listing of recommended USB mass storage devices, go to:


For more information, see scsa2usb(7D).

▼ How to Use a Non-Compliant USB Mass Storage Device

1 Become an administrator.

2 Add an entry to the /etc/driver/drv/scsa2usb.conf file.
   The following entry is appropriate for a USB memory stick.
   attribute-override-list = "vid=* reduced-cmd-support=true";

3 Either reboot the system or do the following:
   a. Remove the device.
   b. Manually update the driver.
      # update_drv -f scsa2usb
   c. Add the device.

Hot-Plugging USB Mass Storage Devices

Hot-plugging a device means the device is added or removed without shutting down the operating system or powering off the system. All USB devices are hot-pluggable.

The removable media manager is now aware of hot-plugged devices. You can just plug in the device, which is mounted in a few seconds. If nothing happens, check to see if it is mounted.
Make sure that removable media services are running.

```
# svc:/system/dbus:default
# svc:/system/hal:default
# svc:/system/filesystem/rmvolmgr:default
```

The file system can be mounted from the device if it is valid and it is recognized.

If the file system on the device is not automatically mounted, try a manual mount.

Before hot-removing the device, find the name of the device in the `eject -l` command’s alias name. Then eject the device’s media. If you don’t do this, the device is released and the port is usable again, but the file system on the device might have been damaged.

When you hot-plug a USB device, the device is immediately seen in the system’s device hierarchy, as displayed in the `prtconf` command output. When you remove a USB device, the device is removed from the system’s device hierarchy, unless you are using the device.

If you are using a device when it is unplugged, the device node remains, but the driver controlling this device stops all activity on the device. Any new I/O activity issued to this device returns an error.

In this situation, the system prompts you to plug in the original device. If the device is no longer available, stop the applications. After a few seconds, the port becomes available again.

**Note** – Data integrity might be impaired if you remove an active or open device. Always close the device before removing, except the attached keyboard and mouse, which can be moved while active.

**How to Add a USB Mass Storage Device**

1. **Become an administrator.**
2. **Connect the USB mass storage device.**
3. **Verify that the USB device has been added.**

   For example:
   ```bash
   $ rmformat
   Looking for devices...
   1. Logical Node: /dev/rdsk/c3t0d0p0
      Physical Node: /pci@0,0/pci108e,534a@2,1/storage@3/disk@0,0
      Connected Device: SanDisk Cruzer Micro 0.3
      Device Type: Removable
      Bus: USB
      Size: 245.0 MB
   ```
Verify that the device is automatically mounted under the /media directory.

For example:

$ ls /media/NONAME
aa  bb

You can also use the rmount -l command to list the paths and nicknames of mountable devices. If the device has been mounted under the /media directory, you will see output similar to the following:

$ rmount -l
/dev/dsk/c3t0d0p0:1 rmdisk0,NONAME,/media/NONAME

How to Add a USB Camera

If the camera’s media uses a PCFS file system, it will be automatically mounted. If the device does not bind to the scsa2usb driver, use libusb applications for transferring the pictures. For more information, refer to /usr/sfw/share/doc/libusb/libusb.txt.

1 Become an administrator.

2 Plug in and turn on the USB camera.

The system creates a logical device for the camera. After the camera is plugged in, output is written to the /var/adm/messages file to acknowledge the device’s connection. The system treats the camera as a storage device.

3 Examine the output that is written to the /var/adm/messages file.

# more /var/adm/messages

Examining this output enables you to determine which logical device was created so that you can then use that device to access your images. The output looks similar to the following:

Jul 15 09:53:35 buffy usb: [ID 349649 kern.info] OLYMPUS, C-3040ZOOM, 00153719068
Jul 15 09:53:35 buffy generic: [ID 936769 kern.info] scsa2usb1 is /pci@0,0/pci925,123407,2/storage@2
Jul 15 09:53:36 buffy scsi: [ID 193665 kern.info] sd3 at scsa2usb1: target 0 lun 0

Match the device with a mountable /dev/dsk link entry, by doing the following:

# ls -l /dev/dsk/c*0 | grep /pci@0,0/pci925,123407,2/storage@2
lrwxrwxrwx 1 root root 58 Jun 14 2010 c3t0d0p0 -> ../../../devices/pci@0,0/pci925,123407,2/storage@2/disk@0,0:a
4 Mount the USB camera file system.
   The camera's file system is most likely a PCFS file system. If file system is PCFS, then it should be
   automatically mounted.

   To manually mount the file system on an x86 system, you would use syntax similar to the
   following:

   # mount -F pcfs /dev/dsk/c3t0d0p0:c /mnt

   To manually mount the file system on a SPARC system, you would use syntax similar to the
   following:

   # mount -F pcfs /dev/dsk/c3t0d0s0:c /mnt

   For information on mounting file systems, see "Mounting and Unmounting Oracle Solaris File
   Systems" on page 328.

   For information on mounting different PCFS file systems, see mount_pcfs(1M).

5 Verify that the image files are available.
   For example:
   
   # ls /mnt/DCIM/100OLYM*/
   P7220001.JPG* P7220003.JPG* P7220005.JPG* 
   P7220002.JPG* P7220004.JPG* P7220006.JPG*

6 View and manipulate the image files created by the USB camera.
   For example:

   # /usr/dt/bin/sdtimage P7220001.JPG &

7 Unmount the file system before disconnecting the camera.
   For example:

   # umount /mnt

8 (Optional) Turn off and disconnect the camera.

▼ How to Remove a USB Mass Storage Device

1 Become an administrator.

2 Stop any active applications that are using the device.

3 Unmount the device.
   
   $ rmount NONAME
   Or, use the umount command as an administrator. For example:

   # umount /media/NONAME
For more information about unmounting a USB device, see “How to Mount or Unmount a USB Mass Storage Device” on page 143.

4 Remove the device.

Preparing to Use a USB Mass Storage Device

You can access information on removable media with or without using removable media services. For information on accessing information on removable media with GNOME’s File Manager, see the GNOME desktop documentation.

After the USB device is formatted, it is usually mounted under the /media/label directory.

The device nodes are created under the /dev/rdsk directory for character devices and under the /dev/dsk directory for block devices. Device links are created when the devices are hot-plugged. For more information, see scsa2usb(7D).

If the device cannot be identified by removable media services, then try to manually mount the device. First, use the rmformat command to identify the device path and then use the mount command to manually mount the device as an administrator.

If a device is mounted by removable media services then you can unmount it by using the rnumount command. If the device is manually mounted, then you would unmount it using the umount command as an administrator.

For more information about mounting and unmounting USB devices, see “How to Mount or Unmount a USB Mass Storage Device” on page 143.

▼ How to Display USB Device Information

Display information about USB devices.

For example, use the prtconf command to display USB device information. The prtconf output in this example has been truncated to only display USB device information.

```bash
$ prtconf
  usb, instance #0
    hub, instance #2
      device, instance #8
        interface (driver not attached)
        printer (driver not attached)
        mouse, instance #14
        device, instance #9
          keyboard, instance #15
          mouse, instance #16
          storage, instance #7
            disk (driver not attached)
```

Chapter 8 • Using USB Devices (Tasks) 135
communications, instance #10
modem (driver not attached)
data (driver not attached)
storage, instance #0
disk (driver not attached)
storage, instance #1
disk (driver not attached)

For example, use the `rmformat` command to display USB device information.

```
$ rmformat
Looking for devices...
1. Logical Node: /dev/rdsk/c3t0d0p0
   Physical Node: /pci@0,0/pci108e,534a@2,1/storage@3/disk@0,0
   Connected Device: SanDisk Cruzer Micro 0.3
   Device Type: Removable
   Bus: USB
   Size: 245.0 MB
   Label: <None>
   Access permissions: Medium is not write protected.
```

▼ **How to Create a File System on a USB Mass Storage Device**

A USB diskette must be formatted before you can add a file system to it. All other USB mass storage devices just need a file system before they can be used. Keep the following key points in mind when formatting a USB device:

- Do not use the `rmformat -F` except on a USB diskette.
- If the default slices are not acceptable, use the `rmformat -s` command to create slices. Use the `fdisk` utility to partition a USB device, if needed. For step-by-step instructions, see:
  - “How to Modify Partitions and Create a PCFS File System on a USB Mass Storage Device” on page 138
  - “How to Create a Solaris Partition and Modify the Slices on a USB Mass Storage Device” on page 141
- If the USB device is automatically mounted, you will have to unmount it before you can create a file system on the USB device. Use the `rmount -l` command to identify the device nickname and then the `rmount` command to unmount the USB device.

**Note** – Perform steps 4-5 only if you need to format a USB diskette.

1 **Become an administrator.**

2 **Add the USB device to your system. For information on hot-plugging USB devices, see:**
   - “Hot-Plugging USB Mass Storage Devices” on page 131
3 (Optional) Identify the USB device.

For example:

```bash
# rmformat
Looking for devices...
1. Logical Node: /dev/rdsk/c2t0d0p0
   Physical Node: /pci@0,0/pci108e,534a@2,1/hub@7/floppy@1/disk@0,0
   Connected Device: MITSUMI USB FDD 1039
   Device Type: Floppy drive
   Bus: USB
   Size: 1.4 MB
   Label: <None>
   Access permissions: Medium is not write protected.
```

In this example, the diskette device is c2t0d0p0.

4 Insert a diskette into the diskette drive, if necessary.

5 Format the diskette, if necessary.

```bash
% rmformat -F long raw-device
```

For example, on a SPARC system:

```bash
% rmformat -F long /dev/rdsk/c2t0d0s2
```

For example, on an x86 system:

```bash
% rmformat -F long /dev/rdsk/c3t0d0p0
```

6 Determine the file system type and make sure the device is unmounted. Then, select one of the following:

For more information about unmounting a USB device, see “How to Mount or Unmount a USB Mass Storage Device” on page 143.

- Create a PCFS file system.

  ```bash
  # mkfs -F pcfs -o nofdisks, size=size raw-device
  ```

  Specify the -size option in 512-byte blocks.

  The following example shows how to create a PCFS file system on a 1.4-MB diskette on a SPARC system:

  ```bash
  # mkfs -F pcfs /dev/rdsk/c2t0d0p0
  Construct a new FAT file system on /dev/rdsk/c2t0d0p0: (y/n)? y
  ```

  The following example shows how to create a PCFS file system on a 1.4-MB diskette on an x86 system:

  ```bash
  # mkfs -F pcfs /dev/rdsk/c2t0d0s2
  Construct a new FAT file system on /dev/rdsk/c2t0d0s2: (y/n)? y
  ```
The following examples show how to create a PCFS filesystem on a 100-MB USB memory stick on a SPARC system:

```
# mkfs -F pcfs /dev/rdsk/c5t0d0s2:c
```

The following examples show how to create a PCFS filesystem on a 100-MB USB memory stick on an x86 system:

```
# mkfs -F pcfs /dev/rdsk/c5t0d0p0:c
```

This command can take several minutes to complete.

- Create a legacy UFS file system.
  ```
  # newfs raw-device
  ```
  For example:
  ```
  # newfs /dev/rdsk/c4t0d0s7
  ```
  Consider using the `newfs -f 4096` option or `newfs -T` option for large USB hard disks.

---

**Note** – UFS file system overhead consumes a significant portion of space on a diskette, due to a diskette’s limited storage capacity.

See the next two procedures for a detailed example of creating a PCFS file system and modifying slices on a USB mass storage device.

#### How to Modify Partitions and Create a PCFS File System on a USB Mass Storage Device

The following steps describe how to delete an existing partition, create a new partition, and then create a PCFS file system on the USB device. Make sure you backup any data before you perform this task.

1. **Become an administrator.**
2. **Start the fdisk utility.**
   For example:
   ```
   # fdisk /dev/rdsk/c3t0d0p0
   ```
3. **Delete the partition by selecting option 3.**
   For example:
   ```
   Total disk size is 29 cylinders
   Cylinder size is 2048 (512 byte) blocks
   Cylinders
   ```
### Partition Status Table

<table>
<thead>
<tr>
<th>Partition</th>
<th>Status</th>
<th>Type</th>
<th>Start</th>
<th>End</th>
<th>Length</th>
<th>%</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Solaris2</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>Active</td>
<td></td>
<td>1</td>
<td>28</td>
<td>28</td>
<td>97</td>
</tr>
</tbody>
</table>

Select one of the following:
1. Create a partition
2. Specify the active partition
3. Delete a partition
4. Change between Solaris and Solaris2 Partition IDs
5. Exit (update disk configuration and exit)
6. Cancel (exit without updating disk configuration)

Enter Selection: 3

Choose the partition number to delete.
For example:

Total disk size is 29 cylinders
Cylinder size is 2048 (512 byte) blocks

<table>
<thead>
<tr>
<th>Cylinders</th>
</tr>
</thead>
<tbody>
<tr>
<td>Partition</td>
</tr>
<tr>
<td>-----------</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td>1</td>
</tr>
</tbody>
</table>

Specify the partition number to delete (or enter 0 to exit): 1

Partition deleted.

Create a partition.
For example:

Total disk size is 29 cylinders
Cylinder size is 2048 (512 byte) blocks

<table>
<thead>
<tr>
<th>Cylinders</th>
</tr>
</thead>
<tbody>
<tr>
<td>WARNING: no partitions are defined!</td>
</tr>
</tbody>
</table>

Select one of the following:
1. Create a partition
2. Specify the active partition
3. Delete a partition
4. Change between Solaris and Solaris2 Partition IDs
5. Exit (update disk configuration and exit)
6. Cancel (exit without updating disk configuration)

Enter Selection: 1
6 Select the FAT32 partition type.

Total disk size is 29 cylinders
Cylinder size is 2048 (512 byte) blocks

<table>
<thead>
<tr>
<th>Partition</th>
<th>Status</th>
<th>Type</th>
<th>Start</th>
<th>End</th>
<th>Length</th>
<th>%</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

WARNING: no partitions are defined!

SELECT ONE OF THE FOLLOWING:
1. Create a partition
2. Specify the active partition
3. Delete a partition
4. Change between Solaris and Solaris2 Partition IDs
5. Exit (update disk configuration and exit)
6. Cancel (exit without updating disk configuration)

Select the partition type to create:
1=SOLARIS2 2=UNIX 3=PCIXOS 4=Other
5=DOS12 6=DOS16 7=DOSEXT 8=DOSBIG
9=DOS16LBA A=x86 Boot B=Diagnostic C=FAT32
D=FAT32LBA E=DOSEXTLBA F=EFI 0=Exit? c

7 Specify the percentage of disk to use for this partition.

Total disk size is 29 cylinders
Cylinder size is 2048 (512 byte) blocks

<table>
<thead>
<tr>
<th>Partition</th>
<th>Status</th>
<th>Type</th>
<th>Start</th>
<th>End</th>
<th>Length</th>
<th>%</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

WARNING: no partitions are defined!

SELECT ONE OF THE FOLLOWING:
1. Create a partition
2. Specify the active partition
3. Delete a partition
4. Change between Solaris and Solaris2 Partition IDs
5. Exit (update disk configuration and exit)
6. Cancel (exit without updating disk configuration)

Select the partition type to create:
Specify the percentage of disk to use for this partition (or type "c" to specify the size in cylinders). 100

8 Select whether the new partition should be the active partition or an inactive partition.

Total disk size is 29 cylinders
Cylinder size is 2048 (512 byte) blocks

<table>
<thead>
<tr>
<th>Partition</th>
<th>Status</th>
<th>Type</th>
<th>Start</th>
<th>End</th>
<th>Length</th>
<th>%</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

WARNING: no partitions are defined!

SELECT ONE OF THE FOLLOWING:
1. Create a partition
2. Specify the active partition
3. Delete a partition
4. Change between Solaris and Solaris2 Partition IDs
5. Exit (update disk configuration and exit)
6. Cancel (exit without updating disk configuration)

Select the partition type to create:
Should this become the active partition? If yes, it will be activated each time the computer is reset or turned on.
Please type "y" or "n".

9 Update the disk configuration and exit.

Total disk size is 29 cylinders
Cylinder size is 2048 (512 byte) blocks

<table>
<thead>
<tr>
<th>Partition</th>
<th>Status</th>
<th>Type</th>
<th>Start</th>
<th>End</th>
<th>Length</th>
<th>%</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td></td>
<td>Win95 FAT32</td>
<td>1</td>
<td>28</td>
<td>28</td>
<td>97</td>
</tr>
</tbody>
</table>

SELECT ONE OF THE FOLLOWING:
1. Create a partition
2. Specify the active partition
3. Delete a partition
4. Change between Solaris and Solaris2 Partition IDs
5. Exit (update disk configuration and exit)
6. Cancel (exit without updating disk configuration)

Enter Selection: 5

10 Create the PCFS file system on this partition.

Make sure the device is unmounted before creating the new file system. For more information about unmounting a USB device, see "How to Mount or Unmount a USB Mass Storage Device" on page 143.

For example:

```
# mkfs -F pcfs -o fat=32 /dev/rdsk/c3t0d0p0:c
Construct a new FAT file system on /dev/rdsk/c3t0d0p0:c: (y/n)? y
```

▼ How to Create a Solaris Partition and Modify the Slices on a USB Mass Storage Device

The following steps illustrate how to create a Solaris partition and modify the slices.

Make sure you back up any data before you perform this task.

1 Become superuser.

2 Start the fdisk utility.

For example:

```
# fdisk /dev/rdsk/c5t0d0s2
No fdisk table exists. The default partition for the disk is:
```
a 100% "SOLARIS System" partition

Type "y" to accept the default partition, otherwise type "n" to edit the partition table.

y

3 Display the current slices.

For example:

```
# prtvtoc /dev/rdsk/c5t0d0s2
* /dev/rdsk/c5t0d0s2 partition map
* */dev/rdsk/c5t0d0s2 partition map
* Dimensions:
* 512 bytes/sector
* 63 sectors/track
* 255 tracks/cylinder
* 16065 sectors/cylinder
* 5836 cylinders
* 5836 accessible cylinders
* Flags:
* 1: unmountable
* 10: read-only
* Partition Tag Flags Sector Count Sector Mount Directory
  0 0 00 0 93755340 93755339
  2 0 00 0 93755340 93755339
```

4 Create a text file with the slice information.

For example:

```
slices: 0 = 0, 5GB, "wm", "home":
  1 = 8225280000, 6GB :
  2 = 0, 44GB, "wm", "backup":
  6 = 16450560000, 15GB
```

Make sure each slice starts on a cylinder boundary. For example, slice 1 starts at 822280000 bytes, which is the cylinder size in bytes multiplied by 1000.

For more information, see the -s option description in rmformat(1).

5 Create the slices by including the slice file created above.

For example:

```
# rmformat -s slice_file /dev/rdsk/c5t0d0s2
```

6 View the new slice information.

For example:

```
# prtvtoc /dev/rdsk/c5t0d0s2
*/dev/rdsk/c5t0d0s2 partition map
* Dimensions:
* 512 bytes/sector
```
63 sectors/track
255 tracks/cylinder
16065 sectors/cylinder
5836 cylinders
5836 accessible cylinders

Flags:
1: unmountable
10: read-only

Unallocated space:

<table>
<thead>
<tr>
<th>First Sector</th>
<th>Last Sector</th>
<th>Count</th>
<th>Sector</th>
</tr>
</thead>
</table>
| 10485760     | 5579240     | 16064999
| 28647912     | 3482088     | 32129999
| 63587280     | 30168060    | 93755339

Partition Tag Flags Sector Count Sector Mount Directory
0  8 00 10485760 10485759
1  3 01 16065000 12582912 28647911
2  5 00 92274688 92274687
6  4 00 32130000 31457280 63587279

How to Mount or Unmount a USB Mass Storage Device

The following steps illustrate how to mount and unmount a USB mass storage device.

1 Become superuser or console user.

2 (Optional) Identify the device.
   For example:
   ```
   $ rmformat
   Looking for devices...
   1. Logical Node: /dev/rdsk/c3t0d0p0
      Physical Node: /pci@0,0/pci108e,534a@2,1/storage@3/disk@0,0
      Connected Device: SanDisk Cruzer Micro 0.3
      Device Type: Removable
      Bus: USB
      Size: 245.0 MB
      Label: <None>
      Access permissions: Medium is not write protected.
   ```
   In this example, the physical diskette device is c2t0d0p0.

3 Select one of the following to mount or unmount a USB mass storage device:
   - Mount a USB mass storage device as a console user.
     You can use the `rmount` command with device nicknames, mount points, or device paths, similar to the following:
$ rmmount rmdisk0
$ rmmount NONAME
$ rmmount /dev/dsk/c3t0d0p0:l

For example:

$ rmmount NONAME
NONAME /dev/dsk/c2t0d0p0 mounted
$ ls /media/NONAME
AA.TXT

■ Unmount a USB mass storage device as a console user.
For example:

$ rmumount NONAME
NONAME /dev/dsk/c2t0d0p0 unmounted

■ Mount a USB mass storage device as superuser.
This example shows how to mount a device with a UFS file system:

# mount /dev/dsk/c1t0d0s2 /mnt

This example shows how to mount a device with a PCFS file system on a SPARC system:

# mount -F pcfs /dev/dsk/c3t0d0s2:c /mnt

This example shows how to mount a device with a PCFS file system on an x86 system:

# mount -F pcfs /dev/dsk/c3t0d0p0:c /mnt

This example shows how to mount a CD with a read-only HSFS file system:

# mount -F hsfs -o ro /dev/dsk/c1t0d0s2 /mnt

■ Unmount a USB mass storage device as superuser.
First, be sure no one is using the file system on the device.
For example:

# fuser -c -u /mnt
# umount /mnt

4 Eject the device, which is optional for DVD, CD, or diskette devices.
For example:

# eject /dev/rdsk/c1t0d0s2

---

Troubleshooting Tips for USB Mass Storage Devices

Keep the following tips in mind if you have problems adding or removing a USB mass storage device.

---
Check the `/var/adm/messages` file for failures to enumerate the device. For enumeration failures, possibly, insert the USB hub or remove a hub and connect it directly to a root USB hub.

- If you have problems accessing a device that was connected while the system is running, try the following command:
  
  ```
  # devfsadm
  ```

- Do not move devices around if the system has been powered down by a suspend operation. For more information, see “SPARC: USB Power Management” on page 124.

- If a device has been hot removed while in use by applications and is no longer available, then stop the applications. Use the `prtconf` command to see whether the device node has been removed.

### Disabling Specific USB Drivers

You can disable specific types of USB devices by disabling their client driver. For example, USB printers can be disabled by disabling the `usbprn` driver that directs them. Disabling `usbprn` does not affect other kinds of devices, such as USB storage devices.

The following table identifies some USB device types and their corresponding drivers.

<table>
<thead>
<tr>
<th>Device Type</th>
<th>Driver to Disable</th>
</tr>
</thead>
<tbody>
<tr>
<td>Audio</td>
<td><code>usb_ac</code> and <code>usb_as</code></td>
</tr>
<tr>
<td>HID (usually keyboard and mouse)</td>
<td><code>hid</code></td>
</tr>
<tr>
<td>Storage</td>
<td><code>scsa2usb</code></td>
</tr>
<tr>
<td>Printer</td>
<td><code>usbprn</code></td>
</tr>
<tr>
<td>Serial</td>
<td><code>usbser_edge</code></td>
</tr>
</tbody>
</table>

If you disable a driver for a USB device that is still connected to the system, you see a console message similar to the following:

```
usba10: WARNING: usba: no driver found for device name
```

#### How to Disable Specific USB Drivers

1. **Become superuser.**

2. **Exclude the driver alias entry from the `/etc/system` file.**

   For example, include the following exclude statement to exclude the `usbprn` driver.

   ```
   exclude: usbprn
   ```
Reboot the system.
# init 6

How to Remove Unused USB Device Links

Use this procedure if a USB device is removed while the system is powered off. Removing the USB device while the system is powered off can leave device links for devices that do not exist.

1. Become superuser.
2. Close all applications that might be accessing the device.
3. Remove the unused links for a specific USB class.
   For example:
   # devfsadm -C -c audio
   Or, just remove the dangling links:
   # devfsadm -C

Using USB Audio Devices (Task Map)

<table>
<thead>
<tr>
<th>Task</th>
<th>Description</th>
<th>For Instructions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Add USB audio devices.</td>
<td>Add a USB microphone and speakers.</td>
<td>“How to Add USB Audio Devices” on page 148</td>
</tr>
<tr>
<td>Identify your system’s primary audio device.</td>
<td>Identify which audio device is your primary audio device.</td>
<td>“How to Identify Your System’s Primary Audio Device” on page 148</td>
</tr>
<tr>
<td>Change the primary USB audio device.</td>
<td>You might want to make one audio device the primary audio device if you remove or change your USB audio devices.</td>
<td>“How to Change the Primary USB Audio Device” on page 149</td>
</tr>
<tr>
<td>Remove unused USB device links.</td>
<td>If you remove a USB audio device while the system is powered off, the /dev/audio device might be pointing to a /dev/sound/* device that doesn't exist.</td>
<td>“How to Remove Unused USB Device Links” on page 146</td>
</tr>
<tr>
<td>Solve USB audio problems.</td>
<td>Use this section if no sound comes from the USB speakers.</td>
<td>“Troubleshooting USB Audio Device Problems” on page 149</td>
</tr>
</tbody>
</table>
Using USB Audio Devices

For information about USB audio support in specific Oracle Solaris releases, see “Oracle Solaris Support for USB Devices” on page 114.

This Oracle Solaris release provides USB audio support that is implemented by a pair of cooperating drivers, usb_ac and usb_as. The audio control driver, usb_ac, is a Solaris USB Architecture compliant client driver that provides the controlling interface to user applications. The audio streaming driver, usb_as, processes audio data messages during play and record. It sets sample frequency and precision, and encodes requests from the usb_ac driver. Both drivers comply with the USB audio class 1.0 specification.

Some audio devices can set volume under software control. A STREAMS module, usb_ah, is pushed on top of the HID driver for managing this function.

Oracle Solaris supports USB audio devices that are play-only, record-only, or record and play. Hot-plugging of USB audio devices is supported.

- USB audio devices are supported on SPARC Ultra and x86 platforms that have USB connectors.
- For fully supported audio data format information, see usb_ac(7D).

The primary audio device is /dev/audio. You can verify that /dev/audio is pointing to USB audio by using the following command:

```
% mixerctl
Device /dev/audioctl:
  Name = USB Audio
  Version = 1.0
  Config = external

Audio mixer for /dev/audioctl is enabled
```

After you connect your USB audio devices, you access them with the audioplay and audiorecord command through the /dev/sound/N device links.

Note that the /dev/audio and /dev/sound/N devices can refer to speakers, microphones, or combination devices. If you refer to the incorrect device type, the command fails. For example, the audioplay command fails if you try to use it with a microphone.

You can select a specific default audio device for most Oracle audio applications, such as audioplay and audiorecord, by setting the AUDIODEV shell variable or by specifying the -d option for these commands. However, setting AUDIODEV does not work for third-party applications that have /dev/audio hardcoded as the audio file.

When you plug in a USB audio device, it automatically becomes the primary audio device, /dev/audio, unless /dev/audio is in use. For instructions on changing /dev/audio from on-board audio to USB audio and vice versa, refer to “How to Change the Primary USB Audio Device” on page 149, and usb_ac(7D).
Hot-Plugging Multiple USB Audio Devices

If a USB audio device is plugged into a system, it becomes the primary audio device, /dev/audio. It remains the primary audio device even after the system is rebooted. If additional USB audio devices are plugged in, the last one becomes the primary audio device.

For additional information on troubleshooting USB audio device problems, see \texttt{usb_ac(7D)}.

\section*{How to Add USB Audio Devices}

1. **Plug in the USB speaker.**
   
   The primary audio device, /dev/audio, points to the USB speaker.
   
   \begin{verbatim}
   % ls -l /dev/audio
   lrwxrwxrwx 1 root root 10 Feb 13 08:46 /dev/audio -> usb/audio0
   \end{verbatim}

2. **(Optional) Remove the speaker. Then, plug it back in.**
   
   If you remove the speaker, the /dev/audio device reverts back to on-board audio.
   
   \begin{verbatim}
   % ls -l /dev/audio
   lrwxrwxrwx 1 root root 7 Feb 13 08:47 /dev/audio -> sound/0
   \end{verbatim}

3. **Add a USB microphone.**
   
   \begin{verbatim}
   % ls -l /dev/audio
   lrwxrwxrwx 1 root root 10 Feb 13 08:54 /dev/audio -> usb/audio1
   \end{verbatim}

\section*{How to Identify Your System's Primary Audio Device}

This procedure assumes that you have already connected the USB audio devices.

- **Examine your system's new audio links.**
  
  - Display your system's new audio links with the \texttt{ls} command.
    
    For example:
    
    \begin{verbatim}
    % ls -lt /dev/audio*
    lrwxrwxrwx 1 root root 7 Jul 23 15:46 /dev/audio -> usb/audio0
    lrwxrwxrwx 1 root root 10 Jul 23 15:46 /dev/audioctl -> usb/audioctl0/
    % ls -lt /dev/sound/*
    lrwxrwxrwx 1 root root 74 Jul 23 15:46 /dev/sound/1 -> ...
    lrwxrwxrwx 1 root root 77 Jul 23 15:46 /dev/sound/1ctl -> ...
    lrwxrwxrwx 1 root other 66 Jul 23 14:21 /dev/sound/0 -> ...
    lrwxrwxrwx 1 root other 69 Jul 23 14:21 /dev/sound/0ctl -> ...
    %
    \end{verbatim}
Notice that the primary audio device, /dev/audio, is pointing to the newly plugged in USB audio device, /dev/usb/audio0.

- You can also examine your system’s USB audio devices with the `prtconf` command and look for the USB device information.

```bash
% prtconf
.
.
usb, instance #0
  hub, instance #0
    mouse, instance #0
    keyboard, instance #1
    device, instance #0
      sound-control, instance #0
      sound, instance #0
      input, instance #0
    .
.
```

▼ How to Change the Primary USB Audio Device

- Select one of the following to change the primary USB audio device.
  - If you want the on-board audio device to become the primary audio device, remove the USB audio devices. The /dev/audio link then points to the /dev/sound/0 entry. If the /dev/sound/0 entry is not the primary audio device, then either shut down the system and use the `boot -r` command, or run the `devfsadm -i` command as root.
  - If you want the USB audio device to become primary audio device, just plug it in and check the device links.

Troubleshooting USB Audio Device Problems

Sometimes, USB speakers do not produce any sound, even though the driver is attached and the volume is set to high. Hot-plugging the device might not change this behavior.

The workaround is to power cycle the USB speakers.

Key Points of Audio Device Ownership

Keep the following key points of audio device ownership in mind when working with audio devices:

- When you plug in a USB audio device and you are logged in on the console, the console is the owner of the /dev/* entries. This situation means you can use the audio device as long as you are logged in to the console.
If you are not logged in to the console when you plug in a USB audio device, root becomes the owner of the device. However, if you log in to the console and attempt to access the USB audio device, device ownership changes to the console. For more information, see `logindevperm(4)`.

When you remotely log in with the `rlogin` command and attempt to access the USB audio device, the ownership does not change. This means that, for example, unauthorized users cannot listen to conversations over a microphone owned by someone else.

### Hot-Plugging USB Devices With the `cfgadm` Command (Task Map)

<table>
<thead>
<tr>
<th>Task</th>
<th>Description</th>
<th>For Instructions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Display USB bus information.</td>
<td>Display information about USB devices and buses.</td>
<td>&quot;How to Display USB Bus Information (cfgadm)&quot; on page 151</td>
</tr>
<tr>
<td>Unconfigure a USB device.</td>
<td>Logically unconfigure a USB device that is still physically connected to the system.</td>
<td>&quot;How to Unconfigure a USB Device&quot; on page 152</td>
</tr>
<tr>
<td>Configure a USB device.</td>
<td>Configure a USB device that was previously unconfigured.</td>
<td>&quot;How to Configure a USB Device&quot; on page 153</td>
</tr>
<tr>
<td>Logically disconnect a USB device.</td>
<td>You can logically disconnect a USB device if you are not physically near the system.</td>
<td>&quot;How to Logically Disconnect a USB Device&quot; on page 153</td>
</tr>
<tr>
<td>Logically connect a USB device.</td>
<td>Logically connect a USB device that was previously logically disconnected or unconfigured.</td>
<td>&quot;How to Logically Connect a USB Device&quot; on page 154</td>
</tr>
<tr>
<td>Disconnect a USB device subtree.</td>
<td>Disconnect a USB device subtree, which is the hierarchy (or tree) of devices below a hub.</td>
<td>&quot;How to Logically Disconnect a USB Device Subtree&quot; on page 154</td>
</tr>
<tr>
<td>Reset a USB device.</td>
<td>Reset a USB device to logically remove and re-create the device.</td>
<td>&quot;How to Reset a USB Device&quot; on page 155</td>
</tr>
<tr>
<td>Change the default configuration of a multi-configuration USB device.</td>
<td>Change the default configuration of a multi-configuration USB device.</td>
<td>&quot;How to Change the Default Configuration of a Multi-Configuration USB Device&quot; on page 155</td>
</tr>
</tbody>
</table>
Hot-Plugging USB Devices With the cfgadm Command

You can add and remove a USB device from a running system without using the `cfgadm` command. However, a USB device can also be *logically* hot-plugged without physically removing the device. This scenario is convenient when you are working remotely and you need to disable or reset a non-functioning USB device. The `cfgadm` command also provides a way to display the USB device tree, including manufacturer and product information.

The `cfgadm` command displays information about *attachment points*, which are locations in the system where dynamic reconfiguration operations can occur. An attachment point consists of the following:

- An occupant, which represents a hardware resource, such as a USB device, that might be configured into the system
- A receptacle, which is the location that accepts the occupant, such as a USB port

Attachment points are represented by logical and physical attachment point IDs (`Ap_Ids`). The physical `Ap_Id` is the physical path name of the attachment point. The logical `Ap_Id` is a user-friendly alternative for the physical `Ap_Id`. For more information on `Ap_Ids`, see `cfgadm_usb(1M)`.

The `cfgadm` command provides the following USB device status information.

<table>
<thead>
<tr>
<th>Receptacle State</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>empty/unconfigured</td>
<td>The device is not physically connected.</td>
</tr>
<tr>
<td>disconnected/unconfigured</td>
<td>The device is logically disconnected and unavailable, even though the device could still be physically connected.</td>
</tr>
<tr>
<td>connected/unconfigured</td>
<td>The device is logically connected, but unavailable. The device is visible in <code>prtconf</code> output.</td>
</tr>
<tr>
<td>connected/configured</td>
<td>The device is connected and available.</td>
</tr>
</tbody>
</table>

The following sections describe how to hot-plug a USB device through the software with the `cfgadm` command. All of the sample USB device information in these sections has been truncated to focus on relevant information.

How to Display USB Bus Information (cfgadm)

For examples of using the `prtconf` command to display USB configuration information, see "How to Display USB Device Information" on page 135.
1 **Display USB bus information.**

For example:

```
% cfdadm
```

<table>
<thead>
<tr>
<th>Ap Id</th>
<th>Type</th>
<th>Receptacle</th>
<th>Occupant</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>usb0/4.5</td>
<td>usb-hub</td>
<td>connected</td>
<td>configured</td>
<td>ok</td>
</tr>
<tr>
<td>usb0/4.5.1</td>
<td>usb-device</td>
<td>connected</td>
<td>configured</td>
<td>ok</td>
</tr>
<tr>
<td>usb0/4.5.2</td>
<td>usb-printer</td>
<td>connected</td>
<td>configured</td>
<td>ok</td>
</tr>
<tr>
<td>usb0/4.5.3</td>
<td>usb-mouse</td>
<td>connected</td>
<td>configured</td>
<td>ok</td>
</tr>
<tr>
<td>usb0/4.5.4</td>
<td>usb-device</td>
<td>connected</td>
<td>configured</td>
<td>ok</td>
</tr>
<tr>
<td>usb0/4.5.5</td>
<td>usb-storage</td>
<td>connected</td>
<td>configured</td>
<td>ok</td>
</tr>
<tr>
<td>usb0/4.5.6</td>
<td>usb-communi</td>
<td>connected</td>
<td>configured</td>
<td>ok</td>
</tr>
<tr>
<td>usb0/4.5.7</td>
<td>unknown</td>
<td>empty</td>
<td>unconfigured</td>
<td>ok</td>
</tr>
</tbody>
</table>

In the preceding example, `usb0/4.5.1` identifies a device connected to port 1 of the second-level external hub, which is connected to port 5 of first-level external hub, which is connected to the first USB controller’s root hub, port 4.

2 **Display specific USB device information.**

For example:

```
% cfdadm -l -s "cols=ap_id:info"
```

```
Ap Id Information
usb0/4.5.1 Mfg: Inside Out Networks Product: Edgeport/421 NConfigs: 1
Config: 0 : ...
usb0/4.5.2 Mfg: <undef> Product: <undef> NConfigs: 1 Config: 0 ...
usb0/4.5.3 Mfg: Mitsumi Product: Apple USB Mouse NConfigs: 1
Config: 0 : ...
usb0/4.5.4 Mfg: NMB Product: NMB USB KB/PS2 M NConfigs: 1 Config: 0
usb0/4.5.5 Mfg: Hagiwara Sys-Com Product: SmartMedia R/W NConfigs: 1
Config: 0 : ...
usb0/4.5.6 Mfg: 3Com Inc. Product: U.S.Robotics 56000 Voice USB Modem NConfigs: 2 ...
usb0/4.5.7
```

▼ **How to Unconfigure a USB Device**

You can unconfigure a USB device that is still physically connected to the system. However, a driver will never attach to the device. Note that a USB device remains in the `prtconf` output even after that device is unconfigured.

1 **Become superuser.**

2 **Unconfigure the USB device.**

For example:

```
# cfdadm -c unconfigure usb0/4.7
Unconfigure the device: /devices/pci@8,700000/usb@5,3/hub@4:4.7
This operation will suspend activity on the USB device
Continue (yes/no)? y
```
3 Verify that the device is unconfigured.

For example:

```
# cfgadm
```

<table>
<thead>
<tr>
<th>Ap_Id</th>
<th>Type</th>
<th>Receptacle</th>
<th>Occupant</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>usb0/4.5</td>
<td>usb-hub</td>
<td>connected</td>
<td>configured</td>
<td>ok</td>
</tr>
<tr>
<td>usb0/4.5.1</td>
<td>usb-device</td>
<td>connected</td>
<td>configured</td>
<td>ok</td>
</tr>
<tr>
<td>usb0/4.5.2</td>
<td>usb-printer</td>
<td>connected</td>
<td>configured</td>
<td>ok</td>
</tr>
<tr>
<td>usb0/4.5.3</td>
<td>usb-mouse</td>
<td>connected</td>
<td>configured</td>
<td>ok</td>
</tr>
<tr>
<td>usb0/4.5.4</td>
<td>usb-device</td>
<td>connected</td>
<td>configured</td>
<td>ok</td>
</tr>
<tr>
<td>usb0/4.5.5</td>
<td>usb-storage</td>
<td>connected</td>
<td>configured</td>
<td>ok</td>
</tr>
<tr>
<td>usb0/4.5.6</td>
<td>usb-communi</td>
<td>connected</td>
<td>configured</td>
<td>ok</td>
</tr>
<tr>
<td>usb0/4.5.7</td>
<td>unknown</td>
<td>empty</td>
<td>unconfigured</td>
<td>ok</td>
</tr>
<tr>
<td>usb0/4.6</td>
<td>usb-storage</td>
<td>connected</td>
<td>configured</td>
<td>ok</td>
</tr>
<tr>
<td>usb0/4.7</td>
<td>usb-storage</td>
<td>connected</td>
<td>unconfigured</td>
<td>ok</td>
</tr>
</tbody>
</table>

▼ How to Configure a USB Device

1 Become superuser.

2 Configure a USB device.

For example:

```
# cfgadm -c configure usb0/4.7
```

3 Verify that the USB device is configured.

For example:

```
# cfgadm usb0/4.7
```

<table>
<thead>
<tr>
<th>Ap_Id</th>
<th>Type</th>
<th>Receptacle</th>
<th>Occupant</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>usb0/4.7</td>
<td>usb-storage</td>
<td>connected</td>
<td>configured</td>
<td>ok</td>
</tr>
</tbody>
</table>

▼ How to Logically Disconnect a USB Device

If you want to remove a USB device from the system and the prtconf output, but you are not physically near the system, just logically disconnect the USB device. The device is still physically connected. However, the device is logically disconnected, unusable, and not visible to the system.

1 Become superuser.

2 Disconnect a USB device.

For example:

```
# cfgadm -c disconnect -y usb0/4.7
```
3 Verify that the device is disconnected.
   For example:
   
   # cfgadm usb0/4.7
   
<table>
<thead>
<tr>
<th>Ap_Id</th>
<th>Type</th>
<th>Receptacle</th>
<th>Occupant</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>usb0/4.7</td>
<td>unknown</td>
<td>disconnected</td>
<td>unconfigured</td>
<td>ok</td>
</tr>
</tbody>
</table>

▼ How to Logically Connect a USB Device

Use this procedure to logically connect a USB device that was previously logically disconnected or unconfigured.

1 Become superuser.

2 Connect a USB device.
   For example:
   
   # cfgadm -c configure usb0/4.7

3 Verify that the device is connected.
   For example:
   
   # cfgadm usb0/4.7
   
<table>
<thead>
<tr>
<th>Ap_Id</th>
<th>Type</th>
<th>Receptacle</th>
<th>Occupant</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>usb0/4.7</td>
<td>usb-storage</td>
<td>connected</td>
<td>configured</td>
<td>ok</td>
</tr>
</tbody>
</table>

The device is now available and visible to the system.

▼ How to Logically Disconnect a USB Device Subtree

Use this procedure to disconnect a USB device subtree, which is the hierarchy (or tree) of devices below a hub.

1 Become superuser.

2 Remove a USB device subtree.
   For example:
   
   # cfgadm -c disconnect -y usb0/4

3 Verify that the USB device subtree is disconnected.
   For example:
   
   # cfgadm usb0/4
   
<table>
<thead>
<tr>
<th>Ap_Id</th>
<th>Type</th>
<th>Receptacle</th>
<th>Occupant</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>usb0/4</td>
<td>unknown</td>
<td>disconnected</td>
<td>unconfigured</td>
<td>ok</td>
</tr>
</tbody>
</table>
How to Reset a USB Device

If a USB device behaves erratically, use the `cfgadm` command to reset the device, which logically removes and re-creates the device.

1. Become superuser.
2. Make sure that the device is not in use.
3. Reset the device.
   For example:
   ```
   # cfgadm -x usb_reset -y usb0/4.7
   ```
4. Verify that the device is connected.
   For example:
   ```
   # cfgadm usb0/4.7
   Ap_Id Type Receptacle Occupant Condition
   usb0/4.7 usb-storage connected configured ok
   ```

How to Change the Default Configuration of a Multi-Configuration USB Device

Keep the following in mind when working with multi-configuration USB devices:

- A USB device configuration defines how a device presents itself to the operating system. This method is different from system device configurations discussed in other `cfgadm` sections.
- Some USB devices support multiple configurations, but only one configuration can be active at a time.
- Multi-configuration devices can be identified by examining the `cfgadm -lv` output. `Nconfigs` will be greater than 1.
- The default USB configuration is configuration 1. The current configuration is reflected in `cfgadm -lv` output as `Config`.
- Changes to the default configuration persist across reboots, hot-removes, and the reconfiguration of the device, as long as the device is reconnected to the same port.

1. Make sure that the device is not in use.
2. Change the default USB configuration.
   For example:
   ```
   # cfgadm -x usb_config -o config=2 usb0/4
   Setting the device: /devices/pci@1f,0/usb@c,3:4
   ```
to USB configuration 2
This operation will suspend activity on the USB device
Continue (yes/no)? yes

3 Verify that the device changed.
For example:

```
# cfgadm -lv usb0/4
Ap_Id   Receptacle  Occupant      Condition  Information  When  Type
       Busy      Phys_Id
usb0/4 connected unconfigured ok      Mfg: Sun  2000
Product: USB-B0B0 aka Robotech
With 6 EPPS High Clk Mode  NConfigs: 7  Config: 2  : EVAL Board Setup
 unavailable
usb-device  n /devices/pci@1f,0/usb@c,3:4
```

Note that Config: now shows 2.
This chapter provides general overview and step-by-step instructions for using InfiniBand (IB) devices in your network.

This is a list of the overview information in this chapter.

- “Overview of InfiniBand Devices” on page 157
- “Dynamically Reconfiguring IB Devices (cfgadm)” on page 161

For information on the procedures associated with using IB devices, see the following:

- “Dynamically Reconfiguring IB Devices (Task Map)” on page 159
- “Using the uDAPL Application Interface With InfiniBand Devices” on page 170
- “Administering IPoIB Devices (dladm)” on page 173
- “Monitoring and Troubleshooting IB Devices” on page 175

For general information about dynamic reconfiguration and hot-plugging, see Chapter 6, “Dynamically Configuring Devices (Tasks).”

**Overview of InfiniBand Devices**

InfiniBand (IB) is a new I/O technology based on switched fabrics. It provides high bandwidth, low latency interconnect for attaching I/O devices to hosts and for host-to-host communication. IB devices are managed by the Solaris IB nexus driver.

Support for the following devices is provided in Oracle Solaris 11:

- IP over IB (IPoIB) devices – Enable the ability to transport the IP packets over IB connections and this feature is implemented by the ibp(7D) driver
- Socket Direct Protocol (SDP) – Provides support for sockets over IB
- Reliable Datagram Service version 3 (RDSv3) and Reliable Datagram Service (RDS)
- NFS over RDMA (NFSoRDMA) - Provides NFS services over IB using RDMA
- **iSCSI Extension for RDMA (iSER)** – Provides the RDMA data transfer capability to the iSCSI protocol
- **User Direct Access Programming Language (uDAPL)**
- **Open Fabric User Verb (OFUV)**
- **Ethernet over IB (EoIB)**

The IB nexus driver queries the Solaris IB Device Manager (IBDM) for services, referred in this guide as `communication services`, to enumerate the IB Port, HCA_SVC, and IB VPPA devices.

The IB partition link represents a new *part* class of data link and this object is managed by using the new `dladm` sub-commands. An IB partition link can be created on top of an IB physical link, one per each `P_key` on the port. The partition links are used for data transfers.

The Port devices bind a communication service to a given `port#` of a Host Channel Adapter (HCA). The VPPA devices bind a communication service to a `port#`, `P_key#` combination instead. The HCA_SVC devices bind a communication service to a given HCA. Note that the Port devices and the HCA_SVC devices always use a `P_key` (partition key) whose value is zero. The Port, HCA_SVC, and VPPA devices are children of the HCA and are enumerated through the `ib.conf` file. For more information, see `ib(7D)`.

The IOC devices are children of the IB nexus driver and are part of an I/O unit. The pseudo devices are also children of the IB nexus driver and refer to all other devices that provide their own configuration files to enumerate. For more information, see `ib(4)`.

The possible IB device tree path name(s) are listed in the following table.

<table>
<thead>
<tr>
<th>IOC device</th>
<th>/ib/ioc@1730000007f510c,1730000007f50</th>
</tr>
</thead>
<tbody>
<tr>
<td>IB pseudo device</td>
<td>/ib/&lt;driver&gt;@&lt;unit-address&gt;</td>
</tr>
<tr>
<td>IB VPPA device</td>
<td>/pci@1f,2000/pci@l/pci15b3,5a44@0/ibport@&lt;port#&gt;,&lt;P_key&gt;,&lt;service&gt;</td>
</tr>
<tr>
<td>IB HCA_SVC device</td>
<td>/pci@1f,2000/pci@l/pci15bc,5a44@0/ibport@0,&lt;service&gt;</td>
</tr>
<tr>
<td>IB Port device</td>
<td>/pci@1f,2000/pci@l/pci15b3,5a44@0/ibport@&lt;port#&gt;,0,&lt;service&gt;</td>
</tr>
<tr>
<td>HCA</td>
<td>/pci@1f,2000/pci@l/pci15b3,5a44@0</td>
</tr>
</tbody>
</table>

Note that the IB HCA_SVC devices have zero as the `port#` and the `P_key`.

The IB components in the preceding table are described as follows:

- `<services>` is a communication service. For example, `ipib` is the communication service used by the `ibd` kernel client driver.
- `<P_key>` is the partition link key value being used.
<port>  Is the port number.

<unit-address>  Refers to IB kernel client driver’s property by this name specified in its driver.conf file. For more information, see driver.conf(4).

For information about using IB diagnostic commands and utilities, see “Monitoring and Troubleshooting IB Devices” on page 175.

**InfiniBand Software Packages**

The IB related software packages are as follows:

- `driver/infiniband/connectx` – Mellanox ConnectX Family InfiniBand HCA and 10GbE NIC drivers
- `driver/infiniband/tavor` – Tavor HCA driver
- `system/io/infiniband/ethernet-over-ib` – InfiniBand device driver implementing Ethernet over InfiniBand
- `system/io/infiniband/ib-device-mgt-agent` – InfiniBand Device Manager Agent
- `system/io/infiniband/ib-sockets-direct` – InfiniBand layered Sockets Direct Protocol
- `system/io/infiniband/ip-over-ib` – Network device driver supporting the IP over InfiniBand (IPoIB) protocol
- `system/io/infiniband/open-fabrics` – Open Fabrics kernel components
- `system/io/infiniband/reliable-datagram-sockets-v3` – Reliable Datagram Sockets (RDSv3)
- `system/io/infiniband/reliable-datagram-sockets` – Reliable Datagram Sockets
- `system/io/infiniband/rpc-over-rdma` – InfiniBand RPC over RDMA Driver
- `system/io/infiniband/udapl` – UDAPL library and commands
- `system/io/infiniband` – InfiniBand Framework

---

**Dynamically Reconfiguring IB Devices (Task Map)**

<table>
<thead>
<tr>
<th>Task</th>
<th>Description</th>
<th>For Instructions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Display IB device information.</td>
<td>Display information about the IB devices on your system.</td>
<td>“How to Display IB Device Information” on page 162</td>
</tr>
<tr>
<td>Configure or unconfigure a port or VPPA device.</td>
<td>Select one of the following:</td>
<td></td>
</tr>
<tr>
<td>Task</td>
<td>Description</td>
<td>For Instructions</td>
</tr>
<tr>
<td>-----------------------------------------------</td>
<td>---------------------------------------------------------------------------------------------------</td>
<td>-----------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Unconfigure a port or a VPPA device.</td>
<td></td>
<td>&quot;How to Unconfigure an IB Port, HCA_SVC, or a VPPA Device&quot; on page 164</td>
</tr>
<tr>
<td>Configure a port or a VPPA device.</td>
<td></td>
<td>&quot;How to Configure a IB Port, HCA_SVC, or a VPPA Device&quot; on page 164</td>
</tr>
<tr>
<td>Configure or unconfigure an IB pseudo device.</td>
<td>Select one of the following:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Unconfigure an IB pseudo device.</td>
<td>&quot;How to Unconfigure an IB Pseudo Device&quot; on page 165</td>
</tr>
<tr>
<td></td>
<td>Configure an IB pseudo device.</td>
<td>&quot;How to Configure an IB Pseudo Device&quot; on page 165</td>
</tr>
<tr>
<td>Display kernel IB clients of an HCA.</td>
<td>You might need to display information about kernel IB clients of an HCA, particularly if you're going to unconfigure an HCA.</td>
<td>&quot;How to Display Kernel IB Clients of an HCA&quot; on page 166</td>
</tr>
<tr>
<td>Configure or unconfigure an IB HCA or EoIB interface.</td>
<td>Select one of the following:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Unconfigure IB devices that are connected to an HCA.</td>
<td>&quot;How to Dynamically Reconfigure an HCA With Active EoIB Devices&quot; on page 166</td>
</tr>
<tr>
<td></td>
<td>Dynamically reconfigure the HCA when EoIB devices active.</td>
<td>&quot;How to Reconfigure and Restore an EoIB Interface After Hot Removal&quot; on page 167</td>
</tr>
<tr>
<td></td>
<td>Unconfigure the not last IB HCA with EoIB interface(s) active.</td>
<td>&quot;Configuring an IB HCA&quot; on page 168</td>
</tr>
<tr>
<td></td>
<td>Configure IB devices that are connected to an HCA.</td>
<td></td>
</tr>
<tr>
<td>Update the IB P_key tables.</td>
<td>If the P_key table information of a HCA port changes, IBTF and IBDM need to be notified so that their internal P_key databases are updated.</td>
<td>&quot;How to Update the IB P_key Tables&quot; on page 168</td>
</tr>
<tr>
<td>Display IB communication services</td>
<td>Display the IB communication services that are currently in use by the IBTF.</td>
<td>&quot;How to Display IB Communication Services’ on page 169</td>
</tr>
<tr>
<td>Add or remove a VPPA communication service.</td>
<td>Select one of the following:</td>
<td></td>
</tr>
</tbody>
</table>
Dynamically Reconfiguring IB Devices (cfgadm)

One can configure or unconfigure an IB device from a running system by using the `cfgadm` CLI only. This command also provides a way to display the IB fabric, manage communication services, and update P_key table databases. For more information, see `cfgadm_ib(1M)`.

The `cfgadm` CLI manages dynamic reconfiguration, referred to in this guide as DR, of the entire IB fabric as seen by a host. The `cfgadm` operations are supported on all the IB devices, such as Port, VPPA, HCA_SVC, IOC, and pseudo devices.

The `cfgadm` command displays information about attachment points (Ap_Ids), which are locations in the system where DR operations can occur. For details on the Ap_Ids that `cfgadm` supports, see `cfgadm_ib.1M`. Note that all IB Ap_Ids are shown as connected.

The `cfgadm` command provides the following IB device status information.

<table>
<thead>
<tr>
<th>Receptacle State</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>connected/configured/ok</td>
<td>The device is connected and available. The devinfo node is present.</td>
</tr>
<tr>
<td>connected/unconfigured/unknown</td>
<td>The device is unavailable and no devinfo node or device driver exists for this device. Or, the device was never configured for use by ib_nexus driver. The device might be known to the IB Device Manager.</td>
</tr>
</tbody>
</table>

All of the procedures in this section require administrative privileges that are not generally granted to user accounts. For more information, see “How to Use Your Assigned Administrative Rights” in Oracle Solaris Administration: Security Services.

The following sections describe how to dynamically reconfigure (DR) IB devices with the `cfgadm` command. All of the sample IB device information in these sections has been truncated to focus on relevant information.
Unconfiguring IB Device Considerations

An actual dynamic reconfiguration (DR) of an HCA is beyond the scope of the IB cfgadm plugin. Although DR of an HCA can be achieved by using the plugin of the underlying bus. For example, a PCI based HCA can use the cfgadm_pci command. For more information, see cfgadm_pci(1M).

Note – RDSv3 does not support unconfiguring a HCA. If the system has a RDSv3 driver installed at the time of DR, unconfiguring the HCA fails as shown below.

```
# cfgadm -c unconfigure ib::rdsv3,0
This operation will suspend activity on the IB device
Continue (yes/no)? yes
cfgadm: Hardware specific failure: unconfigure operation failed ap_id: /devices/ib:fabric::rdsv3,0
```

```
# cfgadm -c unconfigure PCI-EM0
cfgadm: Component system is busy, try again: unconfigure failed
```

Workaround:

On a production system, you must remove the RDSv3 driver before the HCA DR operation and reboot the system.

```
# rem_drv rdsv3
Device busy
Cannot unload module: rdsv3
Will be unloaded upon reboot.
```

```
# init 6
```

How to Display IB Device Information

You can use the prtconf command to display general information about IB devices. For example:

```
$ prtconf

ib, instance #0
   rpcib, instance #0
   rdsib, instance #0
   daplt, instance #0
   rdsv3, instance #0
   sdpib, instance #0
eibnx, instance #0
   sol umad, instance #0
   sol uverbs, instance #0
   iser, instance #0
```
In the above example, pci15b3, 673c refers to an IB HCA.

Use the following steps to display specific IB device information.

1 Become an administrator.

2 Display IB fabric information.

For example:

```
# cfgadm -al
```

<table>
<thead>
<tr>
<th>Ap_Id</th>
<th>Type</th>
<th>Receptacle</th>
<th>Occupant</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>ib</td>
<td>IB-Fabric</td>
<td>connected</td>
<td>configured</td>
<td>ok</td>
</tr>
<tr>
<td>ib::21280001A0A478</td>
<td>IB-HCA</td>
<td>connected</td>
<td>configured</td>
<td>ok</td>
</tr>
<tr>
<td>ib::21280001A0A47A,0,ipib</td>
<td>IB-PORT</td>
<td>connected</td>
<td>configured</td>
<td>ok</td>
</tr>
<tr>
<td>ib::1730000008070,0,hnfs</td>
<td>IB-HCA_SVC</td>
<td>connected</td>
<td>configured</td>
<td>ok</td>
</tr>
<tr>
<td>ib::daplt,0</td>
<td>IB-PSEUDO</td>
<td>connected</td>
<td>configured</td>
<td>ok</td>
</tr>
<tr>
<td>ib::eibnx,0</td>
<td>IB-PSEUDO</td>
<td>connected</td>
<td>configured</td>
<td>ok</td>
</tr>
<tr>
<td>ib::iser,0</td>
<td>IB-PSEUDO</td>
<td>connected</td>
<td>configured</td>
<td>ok</td>
</tr>
<tr>
<td>ib::rdsib,0</td>
<td>IB-PSEUDO</td>
<td>connected</td>
<td>configured</td>
<td>ok</td>
</tr>
<tr>
<td>ib::rdsv3,0</td>
<td>IB-PSEUDO</td>
<td>connected</td>
<td>configured</td>
<td>ok</td>
</tr>
<tr>
<td>ib::rpcib,0</td>
<td>IB-PSEUDO</td>
<td>connected</td>
<td>configured</td>
<td>ok</td>
</tr>
<tr>
<td>ib::sdpib,0</td>
<td>IB-PSEUDO</td>
<td>connected</td>
<td>configured</td>
<td>ok</td>
</tr>
<tr>
<td>ib::sol_umad,0</td>
<td>IB-PSEUDO</td>
<td>connected</td>
<td>configured</td>
<td>ok</td>
</tr>
<tr>
<td>ib::sol_uverbs,0</td>
<td>IB-PSEUDO</td>
<td>connected</td>
<td>configured</td>
<td>ok</td>
</tr>
</tbody>
</table>

In the above example output, the components are described as follows:

- **Ap_Id ib::21280001A0A47A,0,ipib** Identifies an IB port device that is connected to port GUID and is bound to the ipib service.
- **Ap_ID ib::sdpib,0** Identifies a pseudo device.
- **Ap_ID hca:21280001A0A478** Identifies an HCA device.
- **Ap_ID ib::1730000008070,0,hnfs** Identifies an IB HCA_SVC device that is bound to the hnfs service.
- **Ap_ID ib::ibgen,0** Identifies a pseudo device.

3 Display specific IB device information.

For example, for an IB port device:

```
# cfgadm -al -s "cols=ap_id:info" ib::21280001A0A47A,0,ipib
```

```
<table>
<thead>
<tr>
<th>Ap_Id</th>
<th>Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>ib::21280001A0A47A,0,ipib</td>
<td>ipib</td>
</tr>
</tbody>
</table>
```
For example, for an IB HCA device:

```
# cfgadm -al -s "cols=ap_id:info" hca::1730000008070
Ap_Id     Information
hca::1730000008070 VID: 0x15b3, PID: 0x5a44, #ports: 0x2,
port1 GUID: 0x1730000008071, port2 GUID: 0x1730000008072
```

The preceding output displays the number of ports and their GUIDs.

## How to Unconfigure an IB Port, HCA_SVC, or a VPPA Device

Use the following steps if you want to remove an IB Port, HCA_SVC, or a VPPA device from the system.

The example below illustrates how to unconfigure an IB Port device, but the same procedure applies to VPPA and HCA_SVC devices as well.

1. **Become an administrator.**

2. **Unconfigure virtual IB port devices.**
   For example:
   ```
   # cfgadm -c unconfigure ib::1730000007F51,*0*,ipib
   Unconfigure the device: /devices/ib:fabric::1730000007F51,*0*,ipib
   This operation will suspend activity on the IB device
   Continue (yes/no)? Y
   ```

3. **Verify that the device is disconnected.**
   For example:
   ```
   # cfgadm -a ib::1730000007F51,*0*,ipib
   Ap_Id     Type     Receptacle Occupant    Condition
   ib::1730000007F51,*0*,ipib IB-VPPA connected unconfigured unknown
   ```

## How to Configure a IB Port, HCA_SVC, or a VPPA Device

Use the following steps if you want to configure an IB Port, HCA_SVC, or a VPPA device on the system.

The example below illustrates how to configure a VPPA device, but similar steps can be used to configure an IB Port and HCA_SVC devices as well.

1. **Become an administrator.**
2 Configure the virtual IB port devices.
For example:

```
# cfgadm -c configure ib::1730000007F51,*0*,ipib
```

3 Verify that the device is connected.
For example:

```
# cfgadm -a ib::1730000007F51,*0*,ipib
```

<table>
<thead>
<tr>
<th>Ap_Id</th>
<th>Type</th>
<th>Receptacle</th>
<th>Occupant</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>ib::1730000007F51,<em>0</em>,ipib</td>
<td>IB-VPPA</td>
<td>connected</td>
<td>configured</td>
<td>ok</td>
</tr>
</tbody>
</table>

Note – A `cfgadm` based configure or unconfigure operation of IB Port and HCA_SVC devices is similar to the preceding examples for an IB VPPA device.

▼ How to Unconfigure an IB Pseudo Device

Use the following steps if you want to remove an IB pseudo device from the system.

1 Become an administrator.

2 Unconfigure the IB pseudo device.
For example:

```
# cfgadm -c unconfigure ib::ibgen,0
```

Unconfigure the device: /devices/ib:fabric::ibgen,0
This operation will suspend activity on the IB device
Continue (yes/no)? Y

3 Verify that the device is disconnected.
```
# cfgadm -a ib::ibgen,0
```

<table>
<thead>
<tr>
<th>Ap_Id</th>
<th>Type</th>
<th>Receptacle</th>
<th>Occupant</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>ib::ibgen,0</td>
<td>IB-PSEUDO</td>
<td>connected</td>
<td>unconfigured</td>
<td>unknown</td>
</tr>
</tbody>
</table>

▼ How to Configure an IB Pseudo Device

Use the following steps to configure an IB pseudo device.

1 Become an administrator.

2 Configure the IB pseudo device.
For example:

```
# cfgadm -yc configure ib::ibgen,0
```
3 Verify that the device is connected.

For example:

```
# cfgadm -a ib::ibgen,0
```

<table>
<thead>
<tr>
<th>Ap_Id</th>
<th>Type</th>
<th>Receptacle</th>
<th>Occupant</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>ib::ibgen,0</td>
<td>IB-PSEUDO</td>
<td>connected</td>
<td>configured</td>
<td>ok</td>
</tr>
</tbody>
</table>

▼ How to Display Kernel IB Clients of an HCA

The following IB `cfgadm` plugin command can be invoked to list kernel IB clients using this HCA. Note that the last column would show a “yes” if a kernel IB client uses another HCA. IB Managers and kernel clients that do not use the HCA are shown with an Ap_Id of “-”.

- Display kernel IB clients of an HCA.

For example:

```
$ cfgadm -x list_clients hca:173000007F50
```

<table>
<thead>
<tr>
<th>Ap_Id</th>
<th>IB Client</th>
<th>Alternate</th>
<th>HCA</th>
</tr>
</thead>
<tbody>
<tr>
<td>ib::173000007F51D0</td>
<td>ibgen</td>
<td>no</td>
<td></td>
</tr>
<tr>
<td>ib::173000007F51D1</td>
<td>ibgen</td>
<td>no</td>
<td></td>
</tr>
<tr>
<td>ib::173000007F51,0,1pib</td>
<td>ibd</td>
<td>no</td>
<td></td>
</tr>
<tr>
<td>ib::ibgen,0</td>
<td>ibgen</td>
<td>no</td>
<td></td>
</tr>
<tr>
<td>-</td>
<td>ibdm</td>
<td>no</td>
<td></td>
</tr>
<tr>
<td>-</td>
<td>ibmf</td>
<td>no</td>
<td></td>
</tr>
<tr>
<td>-</td>
<td>nfs/ib</td>
<td>no</td>
<td></td>
</tr>
</tbody>
</table>

▼ How to Dynamically Reconfigure an HCA With Active EoIB Devices

When a dynamic reconfiguration (DR) operation is attempted on a system and a Connect-X family IB HCA is in use with active EoIB interfaces, (for example, EoIB datalinks are plumbed or VNICS are created over the datalinks on the HCA), the DR operation succeeds if either of the following conditions is true:

- This is the only Connect-X family HCA on the system
- Multiple Connect-X HCAs exist on the system, but this adapter is the last one to be unconfigured

Under any other condition, a dynamic reconfiguration (DR) operation on an IB HCA with active EoIB interfaces fails with a `cfgadm` error message.

If the DR operation fails, you must unplug the active EoIB interfaces and delete any VNICS on the datalink and retry the DR operation. In addition, in such a case, if a Connect-X family HCA is replaced in the same slot after the DR operation and configured again using the `cfgadm` or
hotplug commands, you must replumb the EoIB datalinks that were unplumbed earlier and recreate any VNICs that were previously deleted.

Note that a functioning EoIB interface requires that the HCA port GUID be listed in the appropriate config file on the Sun Network QDR Infiniband Gateway Switch. So, after a hot-removal and before a re-insertion, you will need to update the new HCA port GUID information in the Gateway Switch config file. For information on how to update the configuration, see the Sun Network QDR Infiniband Gateway Switch Administration Guide.

1. **Become an administrator on the system with multiple IB HCAs.**

2. **Attempt to unconfigure the attachment point associated with the EoIB datalink.**

   ```bash
   # cfgadm -c unconfigure PCI-EM0
   cfgadm: Component system is busy, try again: unconfigure failed
   ```

3. **Determine the EoIB interface that is preventing the unconfigure operation.**

   ```bash
   # dmesg | tail | grep 'failing HCA detach'
   Aug 23 12:37:20 eoib: [ID 530795 kern.warning] WARNING: eoib0 still in use, failing HCA detach
   ```

4. **Determine if eoib0 is in use because an IP interface exists on it.**

   ```bash
   # ipadm show-if eoib0
   ipadm: cannot get information for interface(s): No such interface
   ```

5. **If no IP interface over eoib0 exists, check to see if any VNIC is present that is causing eoib0 to be busy.**

   ```bash
   # dladm show-vnic
<table>
<thead>
<tr>
<th>LINK</th>
<th>OVER</th>
<th>SPEED</th>
<th>MACADDRESS</th>
<th>MACADDRTYPE</th>
<th>VID</th>
</tr>
</thead>
<tbody>
<tr>
<td>evnic0</td>
<td>eoib0</td>
<td>10000</td>
<td>2:8:20:e5:56:99</td>
<td>random</td>
<td>0</td>
</tr>
</tbody>
</table>
   ```

6. **Delete the VNIC over eoib0.**

   ```bash
   # dladm delete-vnic evnic0
   ```

7. **Retry the unconfigure operation.**

   ```bash
   # cfgadm -c unconfigure PCI-EM0
   ```

**How to Reconfigure and Restore an EoIB Interface After Hot Removal**

1. **Become an administrator on the Sun Network QDR Gateway Switch.**

2. **Edit the BXM config file on the Sun Network QDR Gateway Switch and replace all occurrences of the old-HCA port GUIDs with the port GUIDs of the new HCA replacing it.**
3 Restart BXM.

4 Become an administrator on the Oracle Solaris system.

5 Hot insert the new HCA on the Solaris system and use the \texttt{cfgadm} command to reconfigure the attachment point originally associated with the EoIB datalink.

6 Confirm the restoration of the original EoIB datalink.

\begin{verbatim}
# dladm show-link | grep eoib0
eeoi0  phys  1500  unknown  --
\end{verbatim}

7 If any VNICS had to be explicitly deleted during a previous unconfigure operation, re-create them now.

\begin{verbatim}
# dladm create-vnic -l eoib0 evnic0
\end{verbatim}

8 Confirm the presence of the VNIC over eoib0.

\begin{verbatim}
# dladm show-vnic
LINK OVER SPEED MACADDRESS MACADDRTYPE VID
evnic0 eoib0 10000 2:8:20:e5:56:99 random 0
\end{verbatim}

9 If any IP addresses had to be explicitly deleted on eoib0 during a previous unconfigure operation, recreate them by using the \texttt{ipadm} command.

### Configuring an IB HCA

Invoke the bus-specific \texttt{cfgadm} plugin to configure the HCA. The exact details are beyond the scope of this chapter.

### How to Update the IB P_key Tables

If the P_key table information of an HCA ports changes, for example, additional P_keys are enabled or disabled, InfiniBand Transport Framework (IBTF) and IBDM need to be notified so that their internal P_key databases are updated. The \texttt{cfgadm} command helps update the P_key databases of IBTF and IBDM. For more information, see \texttt{ibtl(7D)} and \texttt{ibdm(7D)}.

1 Become an administrator.

2 Update the P_key tables.

For example:

\begin{verbatim}
# cfgadm -x update_pkey_tbls -y ib
\end{verbatim}
How to Display IB Communication Services

Use the following steps to display the communication services that are currently in use by the IBTF:

1. Become an administrator.

2. Display IB communication services.
   For example:
   
   ```
   # cfgadm -x list_services ib
   Port communication services:
   srp
   VPPA communication services:
   ibd
   HCA_SVC communication services:
   hnfs
   ```

How to Add a VPPA Communication Service

Use the following steps to add a new VPPA communication service.

Similar steps can be used to add a new HCA_SVC or a port communication service.

1. Become an administrator.

2. Add a new VPPA communication service.
   For example:
   
   ```
   # cfgadm -o comm=vppa,service=new -x add_service ib
   ```

3. Verify that the new service has been added.
   For example:
   
   ```
   # cfgadm -x list_services ib
   Port communication services:
   srp
   VPPA communication services:
   ibd
   new
   HCA_SVC communication services:
   nfs_service
   ```
How to Remove an Existing IB Port, HCA_SVC, or a VPPA Communication Service

Use the following steps to delete an existing IB Port, HCA_SVC, or a VPPA communication service.

1 Become an administrator.

2 Remove a VPPA communication service.
   For example:
   
   ```
   # cfgadm -o comm=vppa,service=new -x delete_service ib
   ```

3 Verify that the communication service has been removed.
   For example:
   
   ```
   # cfgadm -x list_services ib
   Port communication services:
       srp
   VPPA communication services:
       ibd
   HCA_SVC communication services:
       hnfs
   ```

Using the uDAPL Application Interface With InfiniBand Devices

User Direct Access Programming Library (uDAPL) is a standard API that promotes data center application data messaging performance, scalability, and reliability over Remote Direct Memory Access (RDMA) capable interconnects such as InfiniBand. The uDAPL interface is defined by the DAT collaborative. For more information about the DAT collaborative, go to the following site:

http://www.datcollaborative.org

The Oracle Solaris release provides the following uDAPL features:

- A standard DAT registry library, `libdat`. For more information, see `libdat(3LIB)`.
- A standard service provider registration file, `dat.conf`. For more information, see `dat.conf(4)`.
- Support for multiple service providers so that each provider specifies their own uDAPL library path, version number, and so on, in their own `service_provider.conf` file. For more information, see, `service_provider.conf(4)`. 
An administrative tool, the datadm command, to configure dat.conf. For more information, see datadm(1M).

A new resource control property, project.max-device-locked-memory, to regulate the amount of locked down physical memory.

A naming scheme that uses either IPv4 or IPv6 addresses that leverage the IP infrastructure, such as ARP in IPv4 and neighbor discovery in IPv6, for address resolution. The Solaris uDAPL Interface Adapter directly maps to an IPoIB device instance.

Support for the standard Address Translation Scheme that is used by the DAT collaborative community.

A uDAPL service provider library to support the tavor or hermon Host Channel Adapter with automatic registration to the dat.conf registration file. For more information, see tavor(7D) or hermon(7D).

Supports both SPARC platform and x86 platforms.

How to Enable uDAPL

1 Become an administrator.

2 Confirm that the following packages are installed. Or, install them, if needed.

- driver/infiniband/connctx – InfiniBand Framework
- driver/infiniband/ – HCA Driver
- system/io/infiniband/ip-over-ib – IP over InfiniBand
- system/io/infiniband/udapl – Service Provider for package

3 Create the IPoIB interfaces.

For example:

```
# ipadm create-ip ibd1
# ipadm create-addr -T static -a 192.168.0.1/24 ibd1/ipv4
# datadm -a /usr/share/dat/ABCudaplt.conf
```

Updating the DAT Static Registry

You can use the datadm command to maintain the DAT static registry, the dat.conf file. For more information about this file, see dat.conf(4).

The datadm command can also be used to register or unregister a service provider to the dat.conf file. For more information, see datadm(1M).
When IPoIB interface adapters are added or removed, run the `datadm` command to update the `dat.conf` file to reflect the current state of the system. A new set of interface adapters for all the service providers that are currently installed will be regenerated.

▼ How to Update the DAT Static Registry

1. Become an administrator.

2. Update the DAT static registry after you add or remove IPoIB interface adapters from the system.
   ```
   # datadm -u
   ```

3. Display the updated DAT static registry.
   ```
   # datadm
   ```

▼ How to Register a Service Provider in the DAT Static Registry

1. Become an administrator.

2. Update the DAT static registry after you add the vendor’s service provider for the Mellanox Host Channel Adapter.
   ```
   # datadm -a /usr/share/dat/ABCudapl.conf
   ```

3. Display the updated DAT static registry.
   ```
   # datadm -v
   ```

▼ How to Unregister a Service Provider from the DAT Static Registry

1. Become an administrator.

2. Update the DAT static registry after you remove the vendor’s service provider for the Mellanox Host Channel Adapter from the system.
   ```
   # datadm -r /usr/share/dat/ABCudapl.conf
   ```

3. Display the updated DAT static registry.
   ```
   # datadm -v
   ```
Administering IPoIB Devices (dladm)

One physical data link is created by default per port per HCA. The physical links can be used as administrative and observability data points. You can create IB partition links over the physical data links, similar to a creating VNICs over a NIC. Keep in mind that the physical data links are not used for data transfers, so plumbing and assigning an IB address is not supported on these links. Data is transferred on the partition data links.

For information about configuring your network components, see Oracle Solaris Administration: IP Services.

▼ How to Display Physical Data Link Information

Use the `dladm show-phys` command to display physical data link information on your system. The physical link state directly corresponds to the IB HCA port state.

1 Become an administrator.

2 Display physical data link information.

For example, display information about `ibp0` — `ibp3` data links.

```
# dladm show-phys
LINK MEDIA STATE SPEED DUPLEX DEVICE
ibp0 Infiniband up 8000 unknown ibp0
ibp1 Infiniband down 8000 unknown ibp1
ibp2 Infiniband down 8000 unknown ibp2
ibp3 Infiniband up 8000 unknown ibp3
```

For example, use the `show-ib` subcommand to display only the physical links, port GUID, port# HCA GUID, and P Key present on the port at the time the command is running.

```
# dladm show-ib
LINK HCAGUID PORTGUID PORT STATE P Key
ibp0 2C9020040041C 2C9020040041D 1 up FFFF, 8001
ibp1 2C9020040041C 2C9020040041E 2 down FFFF
ibp2 3BA0001004E14 3BA0001004E15 1 down FFFF
ibp3 3BA0001004E14 3BA0001004E16 2 up FFFF, 8001
```

▼ How to Create IB Partition Links

IB partition data links can be created on top of IB physical links, one per each P Key on the port. The partition data links are used for data transfers.

1 Become an administrator.
2 Create new IB partition links.
For example, an IB partition link for P-Key 0x8001 is created on top of ibp0 physical data link.

```bash
# dladm create-part -l ibp0 -P 0x8001 p8001.ibp0
```

The above command succeeds if the port is up, the P-Key is present on the port, and IPoIB is successfully initialized.

For example, an IB partition link for the P-Key 0x9000 is created on top of ibp2.

```bash
# dladm create-part -f -l ibp2 -P 0x9000 p9000.ibp2
```

Note – P-Key 0x9000 is not configured on the port. The force option (-f) allows you to create an IB partition link even when the P-Key is not present or the port is down. The link state is marked as down. The link state transitions to up when P-Key is added to the port and port is activated.

3 Display the IB partition link information.
For example:

```bash
# dladm show-part
```

<table>
<thead>
<tr>
<th>LINK</th>
<th>P_Key</th>
<th>OVER</th>
<th>STATE</th>
<th>FLAGS</th>
</tr>
</thead>
<tbody>
<tr>
<td>p8000.ibp0</td>
<td>8001</td>
<td>ibp0</td>
<td>unknown</td>
<td>----</td>
</tr>
<tr>
<td>p9000.ibp2</td>
<td>9000</td>
<td>ibp2</td>
<td>unknown</td>
<td>f---</td>
</tr>
</tbody>
</table>

The IB partition link state is up when the link is ready to use, after it is plumbed. The partition link state is down under the following conditions:

- HCA port is down
- P_Key is absent
- Broadcast group is absent

4 Plumb and assign an IP address to an IB partition link.
For example:

```bash
# ipadm create-ip p9000.ibp2
# ipadm create-addr -T static -a 1.1.1.1 p9000.ibp2/ipv4
# ipadm show-addr
```

▼ How to Display IB Partition Link Information

1 Become an administrator.

2 Display the IB partition link information.
For example:

```bash
# dladm show-part
```

<table>
<thead>
<tr>
<th>LINK</th>
<th>P_Key</th>
<th>OVER</th>
<th>STATE</th>
<th>FLAGS</th>
</tr>
</thead>
<tbody>
<tr>
<td>p8000.ibp0</td>
<td>8001</td>
<td>ibp0</td>
<td>unknown</td>
<td>----</td>
</tr>
</tbody>
</table>
How to Remove an IB Partition Link

1. Become an administrator.

2. Remove an IB partition link.
   In this example, partition link p8001.ibp0 is removed.
   ```
   # dladm delete-part p8001.ibp0
   ```

3. Review the partition link information.
   ```
   # dladm show-part
   LINK   P.Key   OVER  STATE   FLAGS
   p9000.ibp2 9000  ibp2  down  f---
   ```

Monitoring and Troubleshooting IB Devices

In the Oracle Solaris 11 release, new commands and utilities allow you to manage the IB fabric more effectively. These commands are included in the `system/io/infiniband/open-fabrics` package and the man pages are installed automatically when the `open-fabrics` package is installed. For example:

```
% man rping
Reformatting page. Please Wait... done
```

```ruby
librdmacm RPING(1)
```

NAME
rping - RDMA CM connection and RDMA ping-pong test.

SYNOPSIS
```
   [-C message_count] [-S message_size]
rping -c [-v] [-V] [-d] [-a address] [-p port]
   [-C message_count] [-S message_size]
```

The following new commands and utilities provide the ability to list and query IB devices, diagnose and trouble shoot IB fabric issues, and measure IB performance.
<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ibv_asyncwatch</td>
<td>Monitors InfiniBand asynchronous events</td>
</tr>
<tr>
<td>ibv_devices or ibv_devinfo</td>
<td>Lists InfiniBand devices or device information</td>
</tr>
<tr>
<td>ibv_rc_pingpong, ibv_sq_pingpong, or ibv_ud_pingpong</td>
<td>Tests node to node connectivity by using RC connection, SRQs, or UD connection</td>
</tr>
<tr>
<td>mckey</td>
<td>Tests RDMA CM multicast setup and simple data transfer</td>
</tr>
<tr>
<td>rping</td>
<td>Tests RDMA CM connection and attempts RDMA ping-pong</td>
</tr>
<tr>
<td>ucmatose</td>
<td>Tests RDMA CM connection and attempts simple ping-pong</td>
</tr>
<tr>
<td>udaddy</td>
<td>Tests RDMA CM datagram setup and attempts simple ping-pong</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>rdma_bw or rdma_lat</td>
<td>Tests RDMA write transactions for streaming bandwidth or latency</td>
</tr>
<tr>
<td>ib_read_bw or ib_read_lat</td>
<td>Tests RDMA read transactions for bandwidth or latency</td>
</tr>
<tr>
<td>ib_send_bw or ib_send_lat</td>
<td>Tests RDMA send transactions for bandwidth or latency</td>
</tr>
<tr>
<td>ib_write_bw or ib_write_bw_postlist</td>
<td>Tests RDMA write transactions for bandwidth that displays one I/O request at a time or post list bandwidth that displays a list of I/O requests</td>
</tr>
<tr>
<td>ib_write_lat</td>
<td>Tests RDMA write transactions for latency</td>
</tr>
<tr>
<td>ib_clock_test</td>
<td>Tests accuracy of system clock</td>
</tr>
<tr>
<td>qperf</td>
<td>Measures socket and RDMA performance</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>rds-info</td>
<td>Displays RDS kernel module information</td>
</tr>
<tr>
<td>rds-ping</td>
<td>Determines if remote node over RDS is reachable</td>
</tr>
<tr>
<td>rds-stress</td>
<td>Sends message between processes over RDS sockets</td>
</tr>
</tbody>
</table>
### Fabric Diagnostic Tools

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ibdiagnet</td>
<td>Performs diagnostic check of the entire fabric</td>
</tr>
<tr>
<td>ibaddr</td>
<td>Queries InfiniBand address or addresses</td>
</tr>
<tr>
<td>ibnetdiscover</td>
<td>Discovers remote InfiniBand topology</td>
</tr>
<tr>
<td>ibping</td>
<td>Validates connectivity between IB nodes</td>
</tr>
<tr>
<td>ibportstate</td>
<td>Queries physical port state and link speed of an IB port</td>
</tr>
<tr>
<td>ibroute</td>
<td>Displays InfiniBand switch forwarding tables</td>
</tr>
<tr>
<td>ibstat or ibsysstat</td>
<td>Query status of InfiniBand device or devices or the status of a system on an IB address</td>
</tr>
<tr>
<td>ibtracert</td>
<td>Traces an IB path</td>
</tr>
<tr>
<td>perfquery or saquery</td>
<td>Queries IB port counters or sIB subnet administration attributes</td>
</tr>
<tr>
<td>sminfo</td>
<td>Queries IB SMInfo attribute</td>
</tr>
<tr>
<td>smpquery or smpdump</td>
<td>Queries or dumps IB subnet management attributes</td>
</tr>
<tr>
<td>ibcheckerrors or ibcheckerrs</td>
<td>Validates IB port (or node) or IB subnet and reports errors</td>
</tr>
<tr>
<td>ibchecknet, ibchecknode, or ibcheckport</td>
<td>Validates IB subnet, node, or port and reports errors</td>
</tr>
<tr>
<td>ibcheckportstate, ibcheckportwidth, ibcheckstate, or ibcheckwidth</td>
<td>Validates IB port that are link up but not active, ports for 1x (2.0 Gbps) link width, ports in IB subnet that are link up but not active, or 1x links in IB subnet</td>
</tr>
<tr>
<td>ibclearcounters or ibclearerrors</td>
<td>Clears port counters or error counters in IB subnet</td>
</tr>
<tr>
<td>ibdatacounters or ibdatacounts</td>
<td>Queries for data counters in IB subnet or IB port data counters</td>
</tr>
<tr>
<td>ibdiscover.pl</td>
<td>Annotates and compares IB topology</td>
</tr>
<tr>
<td>ibhosts</td>
<td>Displays IB host nodes in topology</td>
</tr>
<tr>
<td>iblinkinfo.pl or iblinkinfo</td>
<td>Displays link information for all links in the fabric</td>
</tr>
<tr>
<td>ibnodes</td>
<td>Displays IB nodes in topology</td>
</tr>
<tr>
<td>ibprintca.pl</td>
<td>Displays either the CA specified or the list of CAs from the ibnetdiscover output</td>
</tr>
<tr>
<td>ibprintrt.pl</td>
<td>Displays either only the router specified or a list of routers from the ibnetdiscover output</td>
</tr>
<tr>
<td>ibprintsment.switch.pl</td>
<td>Displays either the switch specified or a list of switches from the ibnetdiscover output</td>
</tr>
</tbody>
</table>
### Table 9-4 Fabric Diagnostic Tools (Continued)

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ibqueryerrors.pl</td>
<td>Queries and report non-zero IB port counters</td>
</tr>
<tr>
<td>ibrouters</td>
<td>Displays IB router nodes in topology</td>
</tr>
<tr>
<td>ibstatus</td>
<td>Queries basic status of IB devices</td>
</tr>
<tr>
<td>ibswitches</td>
<td>Displays IB switch nodes in topology</td>
</tr>
<tr>
<td>ibswportwatch.pl</td>
<td>Polls the counters on the specified switch or port and report rate of change information</td>
</tr>
<tr>
<td>set_nodedesc.sh</td>
<td>Sets or displays node description string for IB Host Controller Adapters (HCA)s</td>
</tr>
<tr>
<td>dump2psl.pl</td>
<td>Dumps PSL file based on opensm output file that is used for credit loop checking</td>
</tr>
<tr>
<td>dump2slvl.pl</td>
<td>Dumps SLVL file based on opensm output file that is used for credit loop checking</td>
</tr>
<tr>
<td>ibis</td>
<td>An extended TCL shell for IB management inband services</td>
</tr>
</tbody>
</table>
This chapter provides overview information about Oracle Solaris disk slices and introduces the format utility.

This is a list of overview information in this chapter.

- “What’s New in Disk Management?” on page 179
- “Where to Find Disk Management Tasks” on page 185
- “Overview of Disk Management” on page 186
- “Disk Terminology” on page 186
- “About Disk Labels” on page 186
- “About Disk Slices” on page 190
- “format Utility” on page 191
- “Partitioning a Disk” on page 194

For instructions on how to add a disk to your system, see Chapter 12, “SPARC: Setting Up Disks (Tasks),” or Chapter 13, “x86: Setting Up Disks (Tasks).”

**What’s New in Disk Management?**

This section describes new disk management features in the Oracle Solaris release.

- “Identifying Devices by Physical Locations” on page 180
- “Two-Terabyte Disk Support for Installing and Booting the Oracle Solaris OS” on page 183
- “iSNS Support in the Solaris iSCSI Target and Initiator” on page 184
- “Solaris COMSTAR iSCSI Support” on page 184
- “x86: Disk Management in the GRUB Boot Environment” on page 184
- “Support for SCSI Disks That are Larger Than 2 Terabytes” on page 185
Identifying Devices by Physical Locations

Oracle Solaris 11: In this release, the /dev/chassis directory provides device names that includes physical locations. You can use this information to help you identify where devices are physically located if they need to be replaced or changed. You can use the following commands to display information by chassis, receptacle, and occupant values for the devices on your system:

- **diskinfo** – Use this command to display general information about physical disk locations
- **format** – Use this command to display physical disk location information for disks when reviewing partition tables or relabeling
- **prtconf -l** – Use this command to display system configuration information that includes physical disk location information
- **zpool status -l** – Use this command to display physical disk location information for pool devices

In addition, you can use the `fmadm add-alias` command to include a disk alias name that helps you identify the physical location of disks in your environment. For example:

```
# fmadm add-alias SUN-Storage-J4200.0912QAJ0001 J4200@RACK10:U26-27
# fmadm add-alias SUN-Storage-J4200.0905QAJ000E J4200@RACK10:U24-25
```

These aliases can then be displayed with the preceding commands to display physical disk location information. For example:

```
$ diskinfo
D:devchassis-path c:occupant-compdev
---------------------------------------------------- ------------------
/dev/chassis/J4200@RACK10:U24-25/SCSI_Device__0/disk c1t13d0
/dev/chassis/J4200@RACK10:U24-25/SCSI_Device__1/disk c1t14d0
/dev/chassis/J4200@RACK10:U24-25/SCSI_Device__2/disk c1t2d0
/dev/chassis/J4200@RACK10:U24-25/SCSI_Device__3/disk c1t3d0
/dev/chassis/J4200@RACK10:U24-25/SCSI_Device__4/disk c1t15d0
/dev/chassis/J4200@RACK10:U24-25/SCSI_Device__5/disk c1t16d0
/dev/chassis/J4200@RACK10:U24-25/SCSI_Device__6/disk c1t17d0
/dev/chassis/J4200@RACK10:U24-25/SCSI_Device__7/disk c1t18d0
```

Determine where a particular disk is located:

```
% diskinfo -c c6t1ld0
D:devchassis-path c:occupant-compdev
---------------------------------------------------- ------------------
/dev/chassis/SUN-Storage-J4400/SCSI_Device__11/disk c6t1ld0
```

In this example, the /dev/chassis disk name includes an alias name that helps you locate the device in your environment.

The following `diskinfo` example shows how to display a specific disk's physical location.
If you want to identify how many disks of a certain type are included on the system, use `diskinfo` syntax similar to the following:

```
$ diskinfo -n SEAGATE ST31000N-SU0B-931.51GB -o Dcf
D:devchassis-path t:occupant-type c:occupant-compdev
---------------------------------------------------- --------------- ------------------
/dev/chassis/colab5@RACK10 24-25/SCSI_Device__0/disk disk c0t13d0
/dev/chassis/colab5@RACK10 24-25/SCSI_Device__1/disk disk c0t14d0
/dev/chassis/colab5@RACK10 24-25/SCSI_Device__2/disk disk c0t2d0
/dev/chassis/colab5@RACK10 24-25/SCSI_Device__3/disk disk c0t1d0
/dev/chassis/colab5@RACK10 24-25/SCSI_Device__4/disk disk c0t15d0
/dev/chassis/colab5@RACK10 24-25/SCSI_Device__5/disk disk c0t16d0
/dev/chassis/colab5@RACK10 24-25/SCSI_Device__6/disk disk c0t6d0
/dev/chassis/colab5@RACK10 24-25/SCSI_Device__7/disk disk c0t7d0
/dev/chassis/colab5@RACK10 24-25/SCSI_Device__8/disk disk c0t17d0
/dev/chassis/colab5@RACK10 24-25/SCSI_Device__9/disk disk c0t18d0
```

Note – The `diskinfo` command require that chassis support SES diagnostic page 0xa (Additional Element Status) and must set the Element Index Present (EIP) bit to 1. Enclosures that do not meet this criteria will not be fully enumerated, and thus, will not be properly represented.

The `format` command has been updated to provide physical device location information. For example:

```
# format
.
.
18. c0t4d0 <SEAGATE-ST345056SSUN450G-081C-419.19GB>
   /pci@0,600000/pci@0/pci9/LSILogic,sas@0/sd@0,0
   /dev/chassis/colab5@RACK10 26-27/SCSI_Device_6/disk
19. c0t27d0 <ATA-SEAGATE ST35000N-3AZQ-465.76GB>
   /pci@0,600000/pci@0/pci9/LSILogic,sas@0/sd@1b,0
   /dev/chassis/colab5@RACK10 26-27/SCSI_Device_7/disk
20. c0t23d0 <ATA-SEAGATE ST31000N-SU0B-931.51GB>
   /pci@0,600000/pci@0/pci9/LSILogic,sas@0/sd@17,0
   /dev/chassis/colab5@RACK10 26-27/SCSI_Device_8/disk
21. c0t24d0 <ATA-SEAGATE ST31000N-SU0B-931.51GB>
   /pci@0,600000/pci@0/pci9/LSILogic,sas@0/sd@18,0
   /dev/chassis/colab5@RACK10 26-27/SCSI_Device_9/disk
```

Use the `prtconf -l` to display the physical device location information. For example:

```
$ prtconf -l | more
System Configuration: Oracle Corporation sun4v
Memory size: 32640 Megabytes
```
What’s New in Disk Management?

System Peripherals (Software Nodes):

SUNW,SPARC-Enterprise-T5220 location: /dev/chassis/SUN-Storage-J4400.0918QAKA24/SCSI_Device_0/disk

... ...

pci, instance #15 location: /dev/chassis/SUN-Storage-J4400.0918QAKA24/SCSI_Device_0/disk

LSILogic,sas, instance #1 location: /dev/chassis/SUN-Storage-J4400.0918QAKA24/SCSI_Device_0/disk

smp, instance #0 (driver not attached)

sd, instance #2 location: /dev/chassis/SUN-Storage-J4400.0918QAKA24/SCSI_Device_0/disk
sd, instance #4 location: /dev/chassis/SUN-Storage-J4400.0918QAKA24/SCSI_Device_1/disk
sd, instance #6 location: /dev/chassis/SUN-Storage-J4400.0918QAKA24/SCSI_Device_2/disk
sd, instance #7 location: /dev/chassis/SUN-Storage-J4400.0918QAKA24/SCSI_Device_3/disk
sd, instance #8 location: /dev/chassis/SUN-Storage-J4400.0918QAKA24/SCSI_Device_4/disk
sd, instance #9 location: /dev/chassis/SUN-Storage-J4400.0918QAKA24/SCSI_Device_5/disk
sd, instance #10 location: /dev/chassis/SUN-Storage-J4400.0918QAKA24/SCSI_Device_6/disk
sd, instance #11 location: /dev/chassis/SUN-Storage-J4400.0918QAKA24/SCSI_Device_7/disk
sd, instance #12 location: /dev/chassis/SUN-Storage-J4400.0918QAKA24/SCSI_Device_8/disk
sd, instance #13 location: /dev/chassis/SUN-Storage-J4400.0918QAKA24/SCSI_Device_9/disk
sd, instance #14 location: /dev/chassis/SUN-Storage-J4400.0918QAKA24/SCSI_Device_10/disk

Use the zpool status -l option to display physical device location information. For example:

% zpool status -l export

pool: export
state: ONLINE
scan: resilvered 379G in 8h31m with 0 errors on Thu Jan 27 23:10:20 2011

config:

<table>
<thead>
<tr>
<th>NAME</th>
<th>STATE</th>
<th>READ</th>
<th>WRITE</th>
<th>CKSUM</th>
</tr>
</thead>
<tbody>
<tr>
<td>export</td>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>mirror-0</td>
<td>ONLINE</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>/dev/chassis/SUN-Storage-J4400.rack22/SCSI_Device_2/disk</td>
<td>ONLINE</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>mirror-1</td>
<td>ONLINE</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>/dev/chassis/SUN-Storage-J4400.rack22/SCSI_Device_3/disk</td>
<td>ONLINE</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>mirror-2</td>
<td>ONLINE</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>/dev/chassis/SUN-Storage-J4400.rack22/SCSI_Device_4/disk</td>
<td>ONLINE</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>/dev/chassis/SUN-Storage-J4400.rack22/SCSI_Device_5/disk</td>
<td>ONLINE</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>mirror-3</td>
<td>ONLINE</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>/dev/chassis/SUN-Storage-J4400.rack22/SCSI_Device_6/disk</td>
<td>ONLINE</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>mirror-4</td>
<td>ONLINE</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>/dev/chassis/SUN-Storage-J4400.rack22/SCSI_Device_7/disk</td>
<td>ONLINE</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>/dev/chassis/SUN-Storage-J4400.rack22/SCSI_Device_8/disk</td>
<td>ONLINE</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>/dev/chassis/SUN-Storage-J4400.rack22/SCSI_Device_9/disk</td>
<td>ONLINE</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>mirror-5</td>
<td>ONLINE</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>/dev/chassis/SUN-Storage-J4400.rack22/SCSI_Device_10/disk</td>
<td>ONLINE</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>/dev/chassis/SUN-Storage-J4400.rack22/SCSI_Device_11/disk</td>
<td>ONLINE</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>spares</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>/dev/chassis/SUN-Storage-J4400.rack22/SCSI_Device_0/disk</td>
<td>AVAIL</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>/dev/chassis/SUN-Storage-J4400.rack22/SCSI_Device_1/disk</td>
<td>AVAIL</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

effects: No known data errors
Multiple Disk Sector Size Support

Oracle Solaris 11: Previous Solaris releases supported a disk sector size of 512 bytes. In this release, disks with sector sizes of 512 bytes, 1024 bytes, 2048 bytes, or 4096 bytes are supported. Large sector disks provide greater disk capacity, higher reliability, and greater efficiency for data transfer and faster drive maintenance.

In this Oracle Solaris release, the only supported file system that can be used on a large sector disk is a non-root ZFS file system. For more information about using a large sector disk as a COMSTAR target, see Chapter 14, “Configuring Storage Devices With COMSTAR.”

The ability to boot and install from a large sector disk is not currently supported.

Two-Terabyte Disk Support for Installing and Booting the Oracle Solaris OS

Oracle Solaris 11: In previous Solaris releases, you could not install and boot the Solaris OS from a disk that was greater than 1 TB in size. In this Solaris release, you can install and boot the Solaris OS from a disk that is up to 2 TB in size. In previous releases, you also had to use an EFI label for a disk that is larger than 1 TB. In this release, you can use the VTOC label on any size disk, but the addressable space by the VTOC is limited to 2 TB.

The Oracle Solaris disk drivers and disk utilities have been updated to provide the following support:

- Installing and booting the Oracle Solaris OS on a two-terabyte disk must be connected to a system with a minimum of 1.5 GB of memory.
- You can use the `format -e` utility to label a disk of any size with a VTOC label, but the addressable space is limited to 2 TB.
- The default label that is used by the `format` utility and the installation software for a disk that is less than 2 TB in size is a VTOC label.
- You can use the `fdisk` utility on a disk that is greater than 1 TB on x86 systems. Support is added for up to 2-TB partitions in the MBR for non-EFI partition types. This support means that Solaris partitions can go up to 2 TB. Other non-EFI partitions may be subject to a limit depending on partition type.
  
  When the `fdisk` utility is run on a disk that is greater than 2 TB in size, a warning message is displayed to indicate that you cannot create a non-EFI partition that is greater than 2 TB.

Keep in mind that you cannot move a disk over 1 TB with a legacy MBR or a legacy VTOC to a previous Solaris release. EFI labeled disks continue to work as in previous Solaris releases.

For more information about the EFI label changes in this release, see “EFI Disk Label” on page 187.
iSNS Support in the Solaris iSCSI Target and Initiator

**Oracle Solaris 11**: This Solaris release provides support for the Internet Storage Name Service (iSNS) protocol in the Solaris iSCSI target and initiator software. The iSNS protocol allows for the automated discovery, management, and configuration of iSCSI devices on a TCP/IP network.

See the following resources for step-by-step instructions:

- For information about configuring the Oracle Solaris iSCSI target to use a third-party iSNS server, see Chapter 14, "Configuring Storage Devices With COMSTAR."
- For information about configuring the Solaris iSCSI target with a Solaris iSNS server, see Chapter 15, "Configuring and Managing the Oracle Solaris Internet Storage Name Service (iSNS)," and `isnsadm(1M)`.

Solaris COMSTAR iSCSI Support

**Oracle Solaris 11**: iSCSI is an Internet Protocol (IP)-based storage networking standard for linking data storage subsystems. By carrying SCSI commands over IP networks, the iSCSI protocol enables you to mount disk devices, from across the network, onto your local system. On your local system, you can use the devices like block devices.

Common Multiprotocol SCSI TARget, or COMSTAR, a software framework enables you to convert any Oracle Solaris 11 host into a SCSI target device that can be accessed over a storage network by initiator hosts.

For more information, see Chapter 14, "Configuring Storage Devices With COMSTAR."

x86: Disk Management in the GRUB Boot Environment

**Oracle Solaris 11**: The GRUB boot menu has replaced the previous method for booting an x86 system. In the area of disk management, you use the GRUB interface when booting from an alternative device to replace a system disk or when installing the bootblocks.

The GRUB boot environment provides the following features:

- **Network boot** – Boot from the network by pressing the F12 key during the BIOS configuration phase.
- **Single-user boot** – Boot to single-user mode by selecting this option from the Solaris failsafe boot menu:
  ```
  kernel /platform/i86pc/multiboot
  ```
  Then, use the `e` (edit) option to add the `-s` single-user option. For example:
  ```
  kernel /platform/i86pc/multiboot -s
  ```
Press return and then press the b key to boot the system. Press control-D to boot the system back to multiuser mode.

In the GRUB environment, you cannot use the fmthard command to install the boot blocks automatically when run on an x86 system. You must install the boot blocks separately.

For detailed feature information and instructions on using the new GRUB based booting on x86 systems, see *Booting and Shutting Down Oracle Solaris on x86 Platforms*.

This feature is not available on SPARC systems.

**Support for SCSI Disks That are Larger Than 2 Terabytes**

*Oracle Solaris 11*: The SCSI driver, ssd or sd, supports 2 terabytes and greater. The SCSI driver, ssd or sd, is limited to 2 TB in previous Solaris releases.

The format utility can be used to label, configure, and partition these larger disks. For information about using the EFI disk label on large disks and restrictions with the fdisk utility, see "Restrictions of the EFI Disk Label" on page 188.

**Where to Find Disk Management Tasks**

Use these references to find step-by-step instructions for managing disks.

<table>
<thead>
<tr>
<th>Disk Management Task</th>
<th>For More Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Format a disk and examine a disk label.</td>
<td>Chapter 11, &quot;Administering Disks (Tasks)&quot;</td>
</tr>
<tr>
<td>Add a new disk to a SPARC system.</td>
<td>Chapter 12, &quot;SPARC: Setting Up Disks (Tasks)&quot;</td>
</tr>
<tr>
<td>Add a new disk to an x86 system.</td>
<td>Chapter 13, &quot;x86: Setting Up Disks (Tasks)&quot;</td>
</tr>
<tr>
<td>Hot-plug a SCSI or PCI disk.</td>
<td>Chapter 6, &quot;Dynamically Configuring Devices (Tasks)&quot;</td>
</tr>
</tbody>
</table>
Overview of Disk Management

Managing disks in the Oracle Solaris OS usually involves setting up the system and running the Oracle Solaris installation program to create the appropriate disk slices and file systems and to install the Oracle Solaris OS. Occasionally, you might need to use the `format` utility to add a new disk drive or replace a defective disk drive.

Disk Terminology

Before you can effectively use the information in this section, you should be familiar with basic disk architecture. In particular, you should be familiar with the following terms:

<table>
<thead>
<tr>
<th>Disk Term</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Track</td>
<td>A concentric ring on a disk that passes under a single stationary disk head as the disk rotates.</td>
</tr>
<tr>
<td>Cylinder</td>
<td>The set of tracks with the same nominal distance from the axis about which the disk rotates.</td>
</tr>
<tr>
<td>Sector</td>
<td>Section of each disk platter. A sector holds 512 bytes.</td>
</tr>
<tr>
<td>Block</td>
<td>A data storage area on a disk. A disk block is 512 bytes.</td>
</tr>
<tr>
<td>Disk controller</td>
<td>A chip and its associated circuitry that controls the disk drive.</td>
</tr>
<tr>
<td>Disk label</td>
<td>The first sector of a disk that contains disk geometry and partition information.</td>
</tr>
<tr>
<td>Device driver</td>
<td>A kernel module that controls a hardware or virtual device.</td>
</tr>
</tbody>
</table>

For additional information, see the product information from your disk’s manufacturer.

About Disk Labels

A special area of every disk is set aside for storing information about the disk’s controller, geometry, and slices. This information is called the disk’s label. Another term that is used to described the disk label is the VTOC (Volume Table of Contents) on a disk with a VTOC label. To label a disk means to write slice information onto the disk. You usually label a disk after you change its slices.

The Oracle Solaris release supports the following two disk labels:

- **SMI** – The traditional VTOC label for disks that are less than 2 TB in size.
- **EFI** – Provides support for disks that are larger than 2 TB. The Extensible Firmware Interface GUID Partition Table (EFI GPT) disk label is also available for disks less than 2 TB.
If you fail to label a disk after you create slices, the slices will be unavailable because the OS has no way of “knowing” about the slices.

**EFI Disk Label**

The EFI label provides support for physical disks and virtual disk volumes that are greater than 2 TB in size. This release also includes updated disk utilities for managing disks greater than 2 TB in size.

The following file system products support file systems greater than 1 TB in size:

- The Oracle Solaris ZFS file system supports file systems greater than 1 TB in size.

The VTOC label is still available for disks less than 2 terabytes in size. If you are only using disks smaller than 2 TB on your systems, managing disks will be the same as in previous Solaris releases. In addition, you can use the `format -e` command to label a disk 2 TB in size or less with an EFI label. For more information, see Example 11–5.

You can use the `format -e` command to apply an EFI label to a disk if the system is running the appropriate Oracle Solaris release. However, you should review the important information in "Restrictions of the EFI Disk Label" on page 188 before attempting to apply an EFI label.

You can also use the `format -e` command to reapply a VTOC label if the EFI label is no longer needed. For example:

```
# format -e
Specify disk (enter its number): 2
selecting c0t5d0
[disk formatted]

format> label
[0] SMI Label
[1] EFI Label
Specify Label type[1]: 0
Warning: This disk has an EFI label. Changing to SMI label will erase all current partitions.
Continue? yes
Auto configuration via format.dat[no]? Auto configuration via generic SCSI-2[no]?
format> quit
```

**Caution** – Keep in mind that changing disk labels will destroy any data on the disk.
Comparison of the EFI Label and the VTOC Label

The EFI disk label differs from the VTOC disk label in the following ways:

■ Provides support for disks greater than 2 terabytes in size.
■ Provides usable slices 0-6, where slice 2 is just another slice.
■ Partitions (or slices) cannot overlap with the primary or backup label, nor with any other partitions. The size of the EFI label is usually 34 sectors, so partitions usually start at sector 34. This feature means that no partition can start at sector zero (0).
■ No cylinder information is stored in the EFI label. Sizes are reported in blocks.
■ Information that was stored in the alternate cylinders area, the last two cylinders of the disk, is now stored in slice 8.
■ If you use the format utility to change partition sizes, the unassigned partition tag is assigned to partitions with sizes equal to zero. By default, the format utility assigns the usr partition tag to any partition with a size greater than zero. You can use the partition change menu to reassign partition tags after the partitions are changed. However, you cannot change a partition with a non-zero size to the unassigned partition tag.

Restrictions of the EFI Disk Label

Keep the following restrictions in mind when determining whether using disks greater than 1 terabyte is appropriate for your environment:

■ Layered software products intended for systems with VTOC-labeled disks might be incapable of accessing a disk with an EFI disk label.
■ A disk with an EFI label is not recognized on systems running previous Solaris releases.
■ You cannot boot from a disk with an EFI disk label.
■ On x86 based systems, you can use the fdisk command on a disk with an EFI label that is greater than 2 TB in size.
■ Use the format utility to partition disks with EFI labels.
■ The EFI specification prohibits overlapping slices. The entire disk is represented by ctydz.
■ The EFI disk label provides information about disk or partition sizes in sectors and blocks, but not in cylinders and heads.
■ The following format options are either not supported or are not applicable on disks with EFI labels:
  ■ The save option is not supported because disks with EFI labels do not need an entry in the format .dat file.
  ■ The backup option is not applicable because the disk driver finds the primary label and writes it back to the disk.
Support for EFI-Labeled Disks on x86 Systems

Oracle Solaris support for the EFI disk label is available on x86 systems. Use the following command to add an EFI label on an x86 system:

```
# format -e
> [0] SMI Label
> [1] EFI Label
> Specify Label type[0]: 1
> WARNING: converting this device to EFI labels will erase all current
> fdisk partition information. Continue? yes
```

Previous label information is not converted to the EFI disk label.

You will have to recreate the label's partition information manually with the `format` command. You cannot use the `fdisk` command on a disk with an EFI label that is 2 terabytes in size. If the `fdisk` command is run on disk that is greater than 2 TB in size to create a Solaris partition, the Solaris partition is limited to 2 TB. For more information about EFI disk labels, see the preceding section.

Installing a System With an EFI-Labeled Disk

An Oracle Solaris root pool disk must have an SMI label. The Oracle Solaris 11 installation utilities automatically relabel any disk that is selected as a root pool disk with an SMI label.

Managing Disks With EFI Disks Labels

Use the following table to locate information on managing disks with EFI disk labels.

<table>
<thead>
<tr>
<th>Task</th>
<th>For More Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>If the system is already installed, connect the disk to the system.</td>
<td>“SPARC: Setting up Disks for ZFS File Systems (Task Map)” on page 217 or “x86: Setting Up Disks for ZFS File Systems (Task Map)” on page 227</td>
</tr>
<tr>
<td>Create a ZFS file system.</td>
<td>“SPARC: Setting Up Disks for ZFS File Systems” on page 218 or “x86: Setting Up Disks for ZFS File Systems” on page 228</td>
</tr>
</tbody>
</table>

Troubleshooting Problems With EFI Disk Labels

Use the following error messages and solutions to troubleshoot problems with EFI-labeled disks.

Solution

Boot a system running SPARC or x86 kernel with a disk greater than 1 terabyte.
Error Message

Dec 3 09:12:17 holoship scsi: WARNING: /sbus@a,0/SUNW,socal@d,10000/sf@1,0/ssd@w50020f23000002a4,0 (ssd1):
Dec 3 09:12:17 holoship corrupt label - wrong magic number

Cause
You attempted to add a disk to a system running an older Solaris release.

Solution
Add the disk to a system running the Solaris release that supports the EFI disk label.

About Disk Slices

Files stored on a disk are contained in file systems. Each file system on a disk is assigned to a slice, which is a group of sectors set aside for use by that file system. Each disk slice appears to the Oracle Solaris OS (and to the system administrator) as though it were a separate disk drive.

For information about file systems, see Chapter 17, “Managing File Systems (Overview).”

Note – Slices are sometimes referred to as partitions. Certain interfaces, such as the format utility, refer to slices as partitions.

When setting up slices, remember these rules:

■ Each disk slice holds only one file system.
■ No file system can span multiple slices.

Slices are set up slightly differently on SPARC and x86 platforms. The following table summarizes the differences.

<table>
<thead>
<tr>
<th>SPARC Platform</th>
<th>x86 Platform</th>
</tr>
</thead>
<tbody>
<tr>
<td>The entire disk is devoted to Oracle Solaris OS.</td>
<td>Disk is divided into fdisk partitions, one fdisk partition per operating system.</td>
</tr>
<tr>
<td>VTOC – The disk is divided into 8 slices, numbered 0-7.</td>
<td>VTOC – The Solaris fdisk partition is divided into 10 slices, numbered 0–9.</td>
</tr>
<tr>
<td>EFI – The disk is divided into 7 slices, numbered 0-6.</td>
<td>EFI – The disk is divided into 7 slices, numbered 0–6</td>
</tr>
</tbody>
</table>

Disk Slices

On a system with ZFS file systems, disk slices are only required for the ZFS root pool that contains the ZFS root file system. In general, the ZFS root pool is contained in slice 0. The ZFS
root pool must exist on a slice or mirrored slices because of a long-standing boot limitation. For non-root storage pools, you can use whole disks. ZFS file systems do not correspond to specific disk slices. Managing whole disks is easier than managing disks with slices.

On x86 systems:

- Disks are divided into `fdisk` partitions. An `fdisk` partition is a section of the disk that is reserved for a particular operating system, such as the Oracle Solaris OS.
- The Oracle Solaris OS places ten slices, numbered 0-9, on a Solaris `fdisk` partition.

**Note** – On a disk with a VTOC label, do not modify slice or use slice 2 to store a file system. The `installgrub` command does not work correctly if slice 2 is modified in any way.

**Using Raw Data Slices**

The disk label is stored in block 0 of each disk. So, third-party database applications that create raw data slices must not start at block 0. Otherwise, the disk label will be overwritten, and the data on the disk will be inaccessible.

Do not use the following areas of the disk for raw data slices, which are sometimes created by third-party database applications:

- Block 0 where the disk label is stored
- Slice 2, which represents the entire disk with a VTOC label

**format Utility**

Read the following overview of the `format` utility and its uses before proceeding to the “how-to” or reference sections.

The `format` utility is a system administration tool that is used to prepare hard disk drives for use on your Oracle Solaris system.

The following table shows the features and associated benefits that the `format` utility provides.

<table>
<thead>
<tr>
<th>Feature</th>
<th>Benefit</th>
</tr>
</thead>
</table>
| Searches your system for all attached disk drives | Reports on the following:  
  - Target location  
  - Disk geometry  
  - Whether the disk is formatted  
  - If the disk has mounted partitions |
| Retrieves disk labels | Convenient for repair operations |
TABLE 10–2 Features and Benefits of the format Utility (Continued)

<table>
<thead>
<tr>
<th>Feature</th>
<th>Benefit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Repairs defective sectors</td>
<td>Allows administrators to repair disk drives with recoverable errors instead of sending the drive back to the manufacturer</td>
</tr>
<tr>
<td>Formats and analyzes a disk</td>
<td>Creates sectors on the disk and verifies each sector</td>
</tr>
<tr>
<td>Partitions a disk</td>
<td>Divides a disk into slices. ZFS file systems do not correspond to disk slices except for the ZFS root pool.</td>
</tr>
<tr>
<td>Labels a disk</td>
<td>Writes disk name and configuration information to the disk for future retrieval (usually for repair operations)</td>
</tr>
</tbody>
</table>

The `format` utility options are described in Chapter 16, “The `format` Utility (Reference).”

**When to Use the format Utility**

Disk drives are partitioned and labeled by the Oracle Solaris installation utility when you install the Oracle Solaris release. You can use the `format` utility to do the following:

- Display slice information
- Partition a disk
- Add a disk drive to an existing system
- Format a disk drive
- Label a disk
- Repair a disk drive
- Analyze a disk for errors

The main reason a system administrator uses the `format` utility is to partition a disk. These steps are covered in Chapter 12, “SPARC: Setting Up Disks (Tasks),” and Chapter 13, “x86: Setting Up Disks (Tasks).”

See the following section for guidelines on using the `format` utility.
Guidelines for Using the format Utility

TABLE 10–3 format Utility Guidelines

<table>
<thead>
<tr>
<th>Task</th>
<th>Guidelines</th>
<th>For More Information</th>
</tr>
</thead>
</table>
| Format a disk. | ■ Any existing data is destroyed when you reformat a disk.  
■ The need for formatting a disk drive has decreased as more and more manufacturers ship their disk drives formatted and partitioned. You might not need to use the format utility when you add or replace a disk drive to an existing system.  
■ If a disk has been relocated and is displaying many disk errors, you can attempt to relabel it. | “How to Format a Disk” on page 203 or “How to Label a Disk” on page 206 |
| Set up a disk that contains a ZFS root file system. | In a non-redundant configuration, a ZFS root file system data from the damaged disk must be restored from a backup medium. Otherwise, the system will have to be reinstalled by using the installation utility. | “SPARC: How to Set Up a Disk for a ZFS Root File System” on page 219 or “x86: How to Set Up a Disk for a ZFS Root File System” on page 229, or, if the system must be reinstalled, Installing Oracle Solaris 11 Systems |
| Create a disk slice for a root pool. | ■ The best way to use a ZFS storage pool is by creating a pool with whole disks.  
■ If a disk is intended to be used for a root pool, you must create a disk slice. This is long-standing boot limitation. | “SPARC: How to Create a Disk Slice for a ZFS Root File System” on page 220 or “x86: How to Create a Disk Slice for a ZFS Root File System” on page 230 |
| Set up a disk that contains a ZFS file system. | A disk that is used for a non-root ZFS file system usually contains space for user or data files. You can attach or add another disk to a root pool or a non-root pool for more disk space. | “SPARC: How to Set Up a Disk for a ZFS File System” on page 224 or “x86: How to Set Up a Disk for a ZFS File System” on page 234 |

Formatting a Disk

In most cases, disks are formatted by the manufacturer or reseller. So, they do not need to be reformatted when you install the drive. To determine if a disk is formatted, use the format utility. For more information, see “How to Determine if a Disk Is Formatted” on page 202.

If you determine that a disk is not formatted, use the format utility to format the disk.

When you format a disk, you accomplish two steps:
The disk media is prepared for use.

A list of disk defects based on a surface analysis is compiled.

Caution – Formatting a disk is a destructive process because it overwrites data on the disk. For this reason, disks are usually formatted only by the manufacturer or reseller. If you think disk defects are the cause of recurring problems, you can use the `format` utility to do a surface analysis. However, be careful to use only the commands that do not destroy data. For details, see “How to Format a Disk” on page 203.

A small percentage of total disk space that is available for data is used to store defect and formatting information. This percentage varies according to disk geometry, and decreases as the disk ages and develops more defects.

Formatting a disk might take anywhere from a few minutes to several hours, depending on the type and size of the disk.

Partitioning a Disk

The `format` utility is most often used by system administrators to partitioning a disk. The steps are as follows:

- Determining which slices are needed
- Determining the size of each slice or partition
- Using the `format` utility to partition the disk
- Labeling the disk with new partition information
- Creating the file system for each partition

The easiest way to partition a disk is to use the `modify` command from the partition menu of the `format` utility. The `modify` command allows you to create partitions by specifying the size of each partition without having to keep track of the starting cylinder boundaries. The `modify` command also keeps tracks of any disk space that remains in the “free hog” slice.

Partition Table Terminology

An important part of the disk label is the partition table. The partition table identifies a disk’s slices, the slice boundaries (in cylinders), and the total size of the slices. You can display a disk’s partition table by using the `format` utility. The following describes partition table terminology.
TABLE 10–4  Partition Table Terminology

<table>
<thead>
<tr>
<th>Partition Term</th>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number</td>
<td>0–7</td>
<td><strong>VTOC</strong> - Partitions or slices, numbered 0–7. <strong>EFI</strong> - Partitions or slices, numbered 0–6.</td>
</tr>
<tr>
<td>Tag</td>
<td>0=UNASSIGNED 1=BOOT 2=ROOT 3=SWAP 4=USR 5=BACKUP 7=VAR 8=HOME 11=RESERVED</td>
<td>A numeric value that usually describes the file system mounted on this partition.</td>
</tr>
<tr>
<td>Flags</td>
<td>wm</td>
<td>The partition is writable and mountable.</td>
</tr>
<tr>
<td></td>
<td>wu rm</td>
<td>The partition is writable and unmountable. This state is the default for partitions that are dedicated for swap areas. (However, the mount command does not check the “not mountable” flag.)</td>
</tr>
<tr>
<td></td>
<td>rm</td>
<td>The partition is read only and mountable.</td>
</tr>
</tbody>
</table>

Partition flags and tags are assigned by convention and require no maintenance.

For more information on displaying the partition table, see the following references:

- “Displaying Partition Table Information” on page 195
- “How to Display Disk Slice Information” on page 204
- “How to Examine a Disk Label” on page 211

Displaying Partition Table Information

The following format utility output shows an example of a partition table from a 74-GB disk with a VTOC label displayed:

Total disk cylinders available: 38756 + 2 (reserved cylinders)

<table>
<thead>
<tr>
<th>Part</th>
<th>Tag</th>
<th>Flag</th>
<th>Cylinders</th>
<th>Size</th>
<th>Blocks</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>root</td>
<td>wm</td>
<td>3 - 2083</td>
<td>4.00GB</td>
<td>(2081/0/0)</td>
</tr>
<tr>
<td>1</td>
<td>swap</td>
<td>wu</td>
<td>2084 - 3124</td>
<td>2.00GB</td>
<td>(1041/0/0)</td>
</tr>
<tr>
<td>2</td>
<td>backup</td>
<td>wm</td>
<td>0 - 38755</td>
<td>74.51GB</td>
<td>(38756/0/0)</td>
</tr>
<tr>
<td>3</td>
<td>unassigned</td>
<td>wm</td>
<td>0</td>
<td>0</td>
<td>(0/0/0)</td>
</tr>
<tr>
<td>4</td>
<td>unassigned</td>
<td>wm</td>
<td>0</td>
<td>0</td>
<td>(0/0/0)</td>
</tr>
<tr>
<td>5</td>
<td>unassigned</td>
<td>wm</td>
<td>0</td>
<td>0</td>
<td>(0/0/0)</td>
</tr>
<tr>
<td>6</td>
<td>unassigned</td>
<td>wm</td>
<td>0</td>
<td>0</td>
<td>(0/0/0)</td>
</tr>
<tr>
<td>7</td>
<td>home</td>
<td>wm</td>
<td>3125 - 38755</td>
<td>68.50GB</td>
<td>(35631/0/0)</td>
</tr>
<tr>
<td>8</td>
<td>boot</td>
<td>wu</td>
<td>0 - 0</td>
<td>1.97MB</td>
<td>(1/0/0)</td>
</tr>
<tr>
<td>9</td>
<td>alternates</td>
<td>wu</td>
<td>1 - 2</td>
<td>3.94MB</td>
<td>(2/0/0)</td>
</tr>
</tbody>
</table>

The partition table displayed by the format utility contains the following information.
Partitioning a Disk

<table>
<thead>
<tr>
<th>Column Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Part</td>
<td>Partition or slice number. See Table 10–4 for a description of this column.</td>
</tr>
<tr>
<td>Tag</td>
<td>Partition tag. See Table 10–4 for a description of this column.</td>
</tr>
<tr>
<td>Flag</td>
<td>Partition flag. See Table 10–4 for a description of this column.</td>
</tr>
<tr>
<td>Cylinders</td>
<td>The starting and ending cylinder number for the slice. Not displayed on EFI-labeled disks.</td>
</tr>
<tr>
<td>Size</td>
<td>The slice size in MB.</td>
</tr>
<tr>
<td>Blocks</td>
<td>The total number of cylinders and the total number of sectors per slice. Not displayed on EFI-labeled disks.</td>
</tr>
<tr>
<td>First Sector</td>
<td>EFI – The starting block number. Not displayed on VTOC-labeled disks.</td>
</tr>
<tr>
<td>Last Sector</td>
<td>EFI – The ending block number. Not displayed on VTOC-labeled disks.</td>
</tr>
</tbody>
</table>

The following is an example of an EFI disk label displayed by using the `prtvtoc` command.

```
# prtvtoc /dev/rdsk/c4t1d0s0
* /dev/rdsk/c4t1d0s0 partition map
* Dimensions:
  * 512 bytes/sector
  * 2576941056 sectors
  * 2576940989 accessible sectors
* Flags:
  * 1: unmountable
  * 10: read-only
* Partition Tag Flags  First Sector  Last Sector  Mount Directory
  0  2  00  34  629145600  629145633
  1  4  00  629145634  629145600  1258291233
  6  4  00  1258291234 1318633404  2576924637
  8 11  00 2576924638  16384 2576941021
```

The output of the `prtvtoc` command provides information in the following three sections:

- Dimensions
- Flags
- Partition Table

<table>
<thead>
<tr>
<th>Column Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Partition</td>
<td>Partition or slice number. For a description of this column, see Table 10–4.</td>
</tr>
<tr>
<td>Tag</td>
<td>Partition tag. For a description of this column, see Table 10–4.</td>
</tr>
<tr>
<td>Flags</td>
<td>Partition flag. For a description of this column, see Table 10–4.</td>
</tr>
</tbody>
</table>
### Using the Free Hog Slice

When you use the `format` utility to change the size of one or more disk slices, you designate a temporary slice that will expand and shrink to accommodate the resizing operations.

This temporary slice donates, or “frees,” space when you expand a slice, and receives, or “hogs,” the discarded space when you shrink a slice. For this reason, the donor slice is sometimes called the free hog.

The free hog slice exists only during installation or when you run the `format` utility. There is no permanent free hog slice during day-to-day operations.

For information on using the free hog slice, see “SPARC: How to Create a Disk Slice for a ZFS Root File System” on page 220 or “x86: How to Create a Disk Slice for a ZFS Root File System” on page 230.
Administering Disks (Tasks)

This chapter contains disk administration procedures. Many procedures described in this chapter are optional if you are already familiar with how disks are managed on systems running the Oracle Solaris OS.

For information on the procedures associated with administering disks, see “Administering Disks (Task Map)” on page 200.

For overview information about disk management, see Chapter 10, “Managing Disks (Overview).”

Administering Disks (Task Map)

<table>
<thead>
<tr>
<th>Task</th>
<th>Description</th>
<th>For Instructions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Identify the disks on a system.</td>
<td>If you are not sure of the types of disks on a system, use the format utility to identify the disk types.</td>
<td>“How to Identify the Disks on a System” on page 200</td>
</tr>
<tr>
<td>Format the disk.</td>
<td>Determine whether a disk is already formatted by using the format utility. In most cases, disks are already formatted. Use the format utility if you need to format a disk.</td>
<td>“How to Determine if a Disk Is Formatted” on page 202</td>
</tr>
<tr>
<td>Display slice information.</td>
<td>Display slice information by using the format utility.</td>
<td>“How to Display Disk Slice Information” on page 204</td>
</tr>
<tr>
<td>Label the disk.</td>
<td>Create the disk label by using the format utility.</td>
<td>“How to Label a Disk” on page 206</td>
</tr>
</tbody>
</table>
Identifying Disks on a System

Use the `format` utility to discover the types of disks that are connected to a system. You can also use the `format` utility to verify that a disk is known to the system. For detailed information on using the `format` utility, see Chapter 16, “The `format` Utility (Reference).”

### How to Identify the Disks on a System

1. **Become an administrator.**
   For more information, see “How to Use Your Assigned Administrative Rights” in Oracle Solaris Administration: Security Services.

2. **Identify the disks that are recognized on the system by using the `format` utility.**
   ```bash
   # format
   ``
   The format utility displays a list of disks that it recognizes under AVAILABLE DISK SELECTIONS.

**Example 11–1** Identifying the Disks on a System

The following example shows `format` command output.

```bash
# format
AVAILABLE DISK SELECTIONS:
  0. c2t0d0 <SUN72G cyl 14087 alt 2 hd 24 sec 424>
     /pci@1c,600000/scsi@2/sd@0,0
  1. c2t1d0 <SUN72G cyl 14087 alt 2 hd 24 sec 424>
     /pci@1c,600000/scsi@2/sd@1,0
  2. c2t2d0 <SUN146G cyl 14087 alt 2 hd 24 sec 848>
     /pci@1c,600000/scsi@2/sd@2,0
  3. c2t3d0 <SUN146G cyl 14087 alt 2 hd 24 sec 848>
     /pci@1c,600000/scsi@2/sd@3,0
Specify disk (enter its number):
```

The output associates a disk's physical and logical device name to the disk's marketing name, which appears in angle brackets `<>`. See the example below. This method is an easy way to identify which logical device names represent the disks that are connected to your system. For a description of logical and physical device names, see “Device Naming Conventions” on page 72.
The following example uses a wildcard to display the four disks that are connected to a controller 0.

```bash
# format /dev/rdsk/c0t6*
AVAILABLE DISK SELECTIONS:
  0. /dev/rdsk/c0t600A0B800022024E000054AC4970A629d0p0 <...>
      /scsi_vhci/disk@g600a0b800022024e000054ac4970a629
  1. /dev/rdsk/c0t600A0B800022024E000054AE4970A711d0p0 <...>
      /scsi_vhci/disk@g600a0b800022024e000054ae4970a711
  2. /dev/rdsk/c0t600A0B800022028A000050444970A834d0p0 <...>
      /scsi_vhci/disk@g600a0b800022028a000050444970a834
  3. /dev/rdsk/c0t600A0B800022028A000050454970A8EA0d0p0 <...>
      /scsi_vhci/disk@g600a0b800022028a000050454970a8ea
```

Specify disk (enter its number):

The following example shows how to identify the disk information.

```bash
# format
0. c2t0d0 <SUN36G cyl 24620 alt 2 hd 27 sec 107>
   /pci@1c,600000/scsi@2/sd@0,0
```

The output identifies that disk 0 (target 0) is connected to the second SCSI host adapter (scsi@2), which is connected to the second PCI interface (/pci@1c,600000/...). The output also associates both the physical and logical device name to the disk’s marketing name, SUN36G.

Some disks do not have a marketing name. If the format output does not identify disks by their marketing names, then you can use the format utility’s type and label features as described in the following steps to include the disk’s marketing name.

The following steps must be done while the disk or system is inactive, which means booting from an installation DVD or the network, unless the disk is currently unused and it will not contain the Oracle Solaris release. In addition, the final step is to relabel the disk, which can remove any existing partition information or data.

```bash
# format
Searching for disks...done
AVAILABLE DISK SELECTIONS:
  0. c2t0d0 <SUN36G cyl 24620 alt 2 hd 27 sec 107>
     /pci@1c,600000/scsi@2/sd@0,0
  1. c2t1d0 <SEAGATE-ST336607LSUN36G-0307-33.92GB>
     /pci@1c,600000/scsi@2/sd@1,0
  2. c2t2d0 <SEAGATE-ST336607LSUN36G-0507-33.92GB>
     /pci@1c,600000/scsi@2/sd@2,0
  3. c2t3d0 <drive type unknown>
     /pci@1c,600000/scsi@2/sd@3,0
```

Specify disk (enter its number): 3

selecting c2t3d0

disk formatted

format> type

AVAILABLE DRIVE TYPES:
  0. Auto configure
  1. other
Formatting a Disk

Disks are typically formatted by the manufacturer or reseller. They usually do not need to be reformatted when you install the drive.

A disk must be formatted before you can do the following:

- Write data to the disk. However, most disks are already formatted.
- Use the Oracle Solaris installation utility to install the system.

Caution – Formatting a disk is a destructive process because it overwrites data on the disk. For this reason, disks are usually formatted only by the manufacturer or reseller. If you think disk defects are the cause of recurring problems, you can use the `format` utility to do a surface analysis. However, be careful to use only the commands that do not destroy data.

How to Determine if a Disk Is Formatted

1. Become an administrator.
   For more information, see "How to Use Your Assigned Administrative Rights" in *Oracle Solaris Administration: Security Services*.

2. Invoke the `format` utility.
   ```
   # format
   ```
   A numbered list of disks is displayed.

3. Type the number of the disk that you want to check.
   ```
   Specify disk (enter its number): 0
   ```
4 Verify that the disk you chose is formatted by noting the following message:
  [disk formatted]

**Example 11–2** Determining if a Disk Is Formatted

The following example shows that disk c2t3d0 is formatted.

```
# format
AVAILABLE DISK SELECTIONS:
  0. c2t0d0 <SUN72G cyl 14087 alt 2 hd 24 sec 424>
     /pci@1c,600000/scsi@2/sd@0,0
  1. c2t1d0 <SUN72G cyl 14087 alt 2 hd 24 sec 424>
     /pci@1c,600000/scsi@2/sd@1,0
  2. c2t2d0 <SUN146G cyl 14087 alt 2 hd 24 sec 848>
     /pci@1c,600000/scsi@2/sd@2,0
  3. c2t3d0 <SUN146G cyl 14087 alt 2 hd 24 sec 848>
     /pci@1c,600000/scsi@2/sd@3,0
Specify disk (enter its number): 3
selecting c2t3d0
[disk formatted]
```

▼ **How to Format a Disk**

Disks are formatted by the manufacturer. Reformating a disk should occur rarely. The process is time-consuming and removes all data from the disk.

1 **Become an administrator.**

   For more information, see “How to Use Your Assigned Administrative Rights” in *Oracle Solaris Administration: Security Services.*

2 **Invoke the format utility.**

   ```
   # format
   ```
   A numbered list of disks is displayed.

3 **Type the number of the disk that you want to format.**

   ```
   Specify disk (enter its number): 3
   ```

   **Caution** – Do not select the disk that contains the root file system. If you format a root pool disk, you delete the OS and any data on this disk.

4 **To begin formatting the disk, type format at the format> prompt. Confirm the command by typing y.**

   ```
   format> format
   ```
   The protection information is not enabled
   The disk will be formatted with protection type 0
Ready to format. Formatting cannot be interrupted and takes 169 minutes (estimated). Continue? yes

5 Verify that the disk format was successful by noting the following messages:

Beginning format. The current time is Fri Apr 1 ...

Formatting...
done

Verifying media...
pass 0 - pattern = 0xc6dec6de
14086/23/734

pass 1 - pattern = 0x6db6db6d
14086/23/734

Total of 0 defective blocks repaired.

6 Exit the format utility.

format> quit

Displaying Disk Slices

The best way to create ZFS storage pools is to use whole disks instead of disk slices because whole disks are easier to manage. The only time you need to use a disk slice is when the disk is intended for the ZFS root pool. This is a long-standing boot limitation. For non-root pools, use whole disks. When you create a pool with whole disks, an EFI label is applied. See the EFI disk label example below.

If you need to prepare a disk for use as a root pool disk, create slice 0 that contains the entire disk capacity, as shown in the VTOC disk label example below.

For information about setting up disks for use with ZFS storage pools, see Chapter 12, "SPARC: Setting Up Disks (Tasks)," or Chapter 13, "x86: Setting Up Disks (Tasks)."

Note – The format utility uses the term partition instead of slice.

▼ How to Display Disk Slice Information

You might need to display disk slice information if the disk is intended to be used for the ZFS root pool. It must also include a SMI label.

1 Become an administrator.

For more information, see “How to Use Your Assigned Administrative Rights” in Oracle Solaris Administration: Security Services.
Invokethe `format` utility.

```
# format
```

A numbered list of disks is displayed.

Type the number of the disk for which you want to display slice information.

```
Specify disk (enter its number): 1
```

Select the `partition` menu.

```
format> partition
```

Display the slice information for the selected disk.

```
partition> print
```

Exit the `format` utility.

```
partition> q
format> q
```

Verify the displayed slice information by identifying specific slice tags and slices.

If the screen output shows that no slice sizes are assigned, the disk probably does not have slices.

Example 11–3 Displaying Disk Slice Information

The following example displays slice information for a disk with a VTOC label.

```
# format
Searching for disks...done
Specify disk (enter its number): 3
Selecting c2t3d0
format> partition
partition> print
Current partition table (c2t3d0):
Total disk cylinders available: 14087 + 2 (reserved cylinders)
```

```
Part Tag Flag Cylinders Size Blocks
0 root wm 0 - 14086 136.71GB (14087/0/0) 286698624
1 swap wu 0 0 (0/0/0) 0
2 backup wu 0 - 14086 136.71GB (14087/0/0) 286698624
3 unassigned wm 0 0 (0/0/0) 0
4 unassigned wm 0 0 (0/0/0) 0
5 unassigned wm 0 0 (0/0/0) 0
6 usr wm 0 0 (0/0/0) 0
7 unassigned wm 0 0 (0/0/0) 0
partition> q
format> q
```

For a detailed description of the slice information in these examples, see Chapter 10, "Managing Disks (Overview)."

The following example shows the slice information for a disk with an EFI label.
Creating and Examining a Disk Label

The labeling of a disk is usually done during system installation or when you are using new disks. You might need to relabel a disk if the disk label becomes corrupted. For example, from a power failure.

The format utility attempts to automatically configure any unlabeled SCSI disk. If the format utility is able to automatically configure an unlabeled disk, it displays a message similar to the following:

```
c2t3d0: configured with capacity of 136.73GB
```

### How to Label a Disk

You can use the following procedure to do the following:

- Label a disk with a VTOC or an EFI label for a disk 2 terabytes in size.
- Label a disk that is greater than 2 terabytes in size with an EFI label.

If you want to put an EFI label on disk smaller than 2 terabytes, see Example 11–5.

1. **Become an administrator.**

   For more information, see “How to Use Your Assigned Administrative Rights” in *Oracle Solaris Administration: Security Services.*
2 Invoke the `format` utility.
   
   ```
   # format
   ```
   
   A numbered list of disks is displayed.

3 Type the number of the disk that you want to label.
   
   ```
   Specify disk (enter its number): 1
   ```
   
   If the `format` utility recognizes the disk type, the next step is to search for a backup label to label the disk. Labeling the disk with the backup label labels the disk with the correct partitioning information, the disk type, and disk geometry.

4 Select one of the following to label the disk:
   - If the disk is unlabeled and was successfully configured, go to Step 5 to label the disk.
     
     The `format` utility will ask if you want to label the disk.
   - If the disk is labeled but you want to change the disk type, or if the `format` utility was not able to automatically configure the disk, proceed to Step 6 to set the disk type and label the disk.

5 Label the disk by typing `y` at the `Label it now?` prompt.
   
   ```
   Disk not labeled. Label it now? y
   ```
   
   The disk is now labeled. Go to step 10 to exit the `format` utility.

6 Enter type at the `format>` prompt.
   
   ```
   format> type
   ```
   
   The Available Drive Types menu is displayed.

7 Select a disk type from the list of possible disk types.
   
   ```
   Specify disk type (enter its number)[12]: 12
   ```
   
   Or, select 0 to automatically configure a SCSI-2 disk.

8 Label the disk. If the disk is not labeled, the following message is displayed.
   
   ```
   Disk not labeled. Label it now? y
   ```
   
   Otherwise, you are prompted with this message:

   ```
   Ready to label disk, continue? y
   ```

9 Verify the disk label.
   
   ```
   format> verify
   ```

10 Exit the `format` utility.
   
   ```
   format> q
   ```
Labeling a Disk

The following example shows how to automatically configure and label a 36-GB disk.

```
# format
AVAILABLE DISK SELECTIONS:
  0. c0t0d0 <SUN36G cyl 24620 alt 2 hd 27 sec 107>
      /pci@1c,600000/scsi@2/sd00,0
  1. c0t1d0 <SEAGATE-ST336607LSUN36G-0307-33.92GB>
      /pci@1c,600000/scsi@2/sd01,0
  2. c0t2d0 <SEAGATE-ST336607LSUN36G-0507-33.92GB>
      /pci@1c,600000/scsi@2/sd02,0
  3. c0t3d0 <SEAGATE-ST336607LSUN36G-0507-33.92GB>
      /pci@1c,600000/scsi@2/sd03,0
Specify disk (enter its number): 3
  select c0t3d0: configured with capacity of 33.92GB
  Disk not labeled. Label it now? yes
  format> verify
  format> q
```

Labeling a Disk With an EFI Label

The following example shows how to use the `format -e` command to label a disk with an EFI label. Remember to verify that your layered software products will continue to work on systems with EFI-labeled disks. For general information on EFI label restrictions, see “Restrictions of the EFI Disk Label” on page 188.

```
# format
AVAILABLE DISK SELECTIONS:
  0. c2t0d0 <SUN72G cyl 14087 alt 2 hd 24 sec 424>
      /pci@1c,600000/scsi@2/sd00,0
  1. c2t1d0 <SUN72G cyl 14087 alt 2 hd 24 sec 424>
      /pci@1c,600000/scsi@2/sd01,0
  2. c2t2d0 <Fujitsu-MAP3147N SUN146G-0501-136.73GB>
      /pci@1c,600000/scsi@2/sd02,0
  3. c2t3d0 <Fujitsu-MAP3147N SUN146G-0501-136.73GB>
      /pci@1c,600000/scsi@2/sd03,0
Specify disk (enter its number): 3
  select c2t3d0:
  [disk formatted]
  format> label
  0] SMI Label
  1] EFI Label
  Specify Label type[0]: 1
  Ready to label disk, continue? yes
  format> quit
```

Change a EFI-Labeled Disk to an SMI-Labeled Disk

The following example shows how to use the `format -e` command to change an EFI labeled disk to an SMI-labeled disk that can be used for a ZFS root pool.
On an x86 system, you will first have to change the EFI fdisk partition to a Solaris partition. For example:

```
# format -e
select disk ...
format> fdisk
FORMAT MENU:
disk - select a disk
type - select (define) a disk type
Total disk size is 17833 cylinders
Cylinder size is 16065 (512 byte) blocks

Cylinders
Partition Status Type Start End Length %
============= =========== =========== =========== =========== ===========
1 EFI 0 17833 17834 100

SELECT ONE OF THE FOLLOWING:
1. Create a partition
2. Specify the active partition
3. Delete a partition
4. Change between Solaris and Solaris2 Partition IDs
5. Edit/View extended partitions
6. Exit (update disk configuration and exit)
7. Cancel (exit without updating disk configuration)
Enter Selection: 3
Specify the partition number to delete (or enter 0 to exit): 1
This will make all files and programs in this partition inaccessible (type "y" or "n"). y
Enter Selection: 1
Select the partition type to create:
1=SOLARIS2 2=UNIX 3=PCIXOS 4=Other 5=DOS12
6=DOS16 7=DOSEXT 8=DOSBIG 9=DOS16LBA A=x86 Boot
B=Diagnostic C=FAT32 D=FAT32LBA E=DOSXTLBA F=EFI (Protective)
G=EFI_SYS 0=Exit? 1
Specify the percentage of disk to use for this partition
(or type "c" to specify the size in cylinders). 100
Should this become the active partition? If yes, it will be activated each time the computer is reset or turned on.
Please type "y" or "n"). y
Enter Selection: 6
Partition 1 is now the active partition.
```

On a SPARC system, follow the steps below.

```
# format -e
Searching for disks...done
AVAILABLE DISK SELECTIONS:
0. c2t0d0 <SUN72G cyl 14087 alt 2 hd 24 sec 424>
   /pci0/1c,000000/scsi0/2/sd0/0
1. c2t1d0 <SUN72G cyl 14087 alt 2 hd 24 sec 424>
   /pci0/1c,000000/scsi0/2/sd1/0
2. c2t2d0 <FUJITSU-MAP3147N SUN146G-0501-136.73GB>
   /pci0/1c,000000/scsi0/2/sd2/0
```
You should also check the default partition table to ensure that it is optimal for a root pool slice, which means that the bulk of the disk space is in slice 0. See the steps below to increase the size of slice 0.

```
format> partition
partition> print
Current partition table (default):
Total disk cylinders available: 14085 + 2 (reserved cylinders)

<table>
<thead>
<tr>
<th>Part</th>
<th>Tag</th>
<th>Flag</th>
<th>Cylinders</th>
<th>Size</th>
<th>Blocks</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>root</td>
<td>wm</td>
<td>1 - 13</td>
<td>129.19MB</td>
<td>(13/0/0) 264576</td>
</tr>
<tr>
<td>1</td>
<td>swap</td>
<td>wu</td>
<td>14 - 26</td>
<td>129.19MB</td>
<td>(13/0/0) 264576</td>
</tr>
<tr>
<td>2</td>
<td>backup</td>
<td>wu</td>
<td>0 - 14086</td>
<td>136.71GB</td>
<td>(14087/0/0) 286698624</td>
</tr>
<tr>
<td>3</td>
<td>unassigned</td>
<td>wm</td>
<td>0</td>
<td>0 (0/0/0) 0</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>unassigned</td>
<td>wm</td>
<td>0</td>
<td>0 (0/0/0) 0</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>unassigned</td>
<td>wm</td>
<td>0</td>
<td>0 (0/0/0) 0</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>usr</td>
<td>wm</td>
<td>27 - 14084</td>
<td>136.43GB</td>
<td>(14058/0/0) 286108416</td>
</tr>
<tr>
<td>7</td>
<td>unassigned</td>
<td>wm</td>
<td>0</td>
<td>0 (0/0/0) 0</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>boot</td>
<td>wu</td>
<td>0 - 0</td>
<td>9.94MB</td>
<td>(1/0/0) 20352</td>
</tr>
<tr>
<td>9</td>
<td>alternates</td>
<td>wm</td>
<td>0</td>
<td>0 (0/0/0) 0</td>
<td></td>
</tr>
</tbody>
</table>

partition> modify
Select partitioning base:
 0. Current partition table (default)
 1. All Free Hog
Choose base (enter number) [0]? 1

<table>
<thead>
<tr>
<th>Part</th>
<th>Tag</th>
<th>Flag</th>
<th>Cylinders</th>
<th>Size</th>
<th>Blocks</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>root</td>
<td>wm</td>
<td>0</td>
<td>0 (0/0/0) 0</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>swap</td>
<td>wu</td>
<td>0</td>
<td>0 (0/0/0) 0</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>backup</td>
<td>wu</td>
<td>0 - 14084</td>
<td>136.69GB</td>
<td>(14085/0/0) 286657920</td>
</tr>
<tr>
<td>3</td>
<td>unassigned</td>
<td>wm</td>
<td>0</td>
<td>0 (0/0/0) 0</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>unassigned</td>
<td>wm</td>
<td>0</td>
<td>0 (0/0/0) 0</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>unassigned</td>
<td>wm</td>
<td>0</td>
<td>0 (0/0/0) 0</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>usr</td>
<td>wm</td>
<td>0</td>
<td>0 (0/0/0) 0</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>unassigned</td>
<td>wm</td>
<td>0</td>
<td>0 (0/0/0) 0</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>boot</td>
<td>wu</td>
<td>0 - 0</td>
<td>9.94MB</td>
<td>(1/0/0) 20352</td>
</tr>
<tr>
<td>9</td>
<td>alternates</td>
<td>wm</td>
<td>0</td>
<td>0 (0/0/0) 0</td>
<td></td>
</tr>
</tbody>
</table>

Do you wish to continue creating a new partition table based on above table? [yes] | yes
Free Hog partition[6]? 0
Enter size of partition '1' [0b, 0c, 0.00mb, 0.00gb]:
Enter size of partition '3' [0b, 0c, 0.00mb, 0.00gb]:
Enter size of partition '4' [0b, 0c, 0.00mb, 0.00gb]:
Enter size of partition '5' [0b, 0c, 0.00mb, 0.00gb]:
```
Enter size of partition '6' [0b, 0c, 0.00mb, 0.00gb]:
Enter size of partition '7' [0b, 0c, 0.00mb, 0.00gb]:

<table>
<thead>
<tr>
<th>Part</th>
<th>Tag</th>
<th>Flag</th>
<th>Cylinders</th>
<th>Size</th>
<th>Blocks</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>root</td>
<td>wm</td>
<td>1 - 14084</td>
<td>136.68GB</td>
<td>(14084/0/0) 286637568</td>
</tr>
<tr>
<td>1</td>
<td>swap</td>
<td>wu</td>
<td>0</td>
<td>0</td>
<td>(0/0/0) 0</td>
</tr>
<tr>
<td>2</td>
<td>backup</td>
<td>wu</td>
<td>0 - 14084</td>
<td>136.69GB</td>
<td>(14085/0/0) 286657920</td>
</tr>
<tr>
<td>3</td>
<td>unassigned</td>
<td>wm</td>
<td>0</td>
<td>0</td>
<td>(0/0/0) 0</td>
</tr>
<tr>
<td>4</td>
<td>unassigned</td>
<td>wm</td>
<td>0</td>
<td>0</td>
<td>(0/0/0) 0</td>
</tr>
<tr>
<td>5</td>
<td>unassigned</td>
<td>wm</td>
<td>0</td>
<td>0</td>
<td>(0/0/0) 0</td>
</tr>
<tr>
<td>6</td>
<td>usr</td>
<td>wm</td>
<td>0</td>
<td>0</td>
<td>(0/0/0) 0</td>
</tr>
<tr>
<td>7</td>
<td>unassigned</td>
<td>wm</td>
<td>0</td>
<td>0</td>
<td>(0/0/0) 0</td>
</tr>
<tr>
<td>8</td>
<td>boot</td>
<td>wu</td>
<td>0 - 0</td>
<td>9.94MB</td>
<td>(1/0/0) 20352</td>
</tr>
<tr>
<td>9</td>
<td>alternates</td>
<td>wm</td>
<td>0</td>
<td>0</td>
<td>(0/0/0) 0</td>
</tr>
</tbody>
</table>

Okay to make this the current partition table [yes]? yes
Enter table name (remember quotes): "c2t0d0"
Ready to label disk, continue? yes
partition> quit
format> quit

How to Examine a Disk Label

Examine disk label information by using the `prtvtoc` command. For a detailed description of the disk label and the information that is displayed by the `prtvtoc` command, see Chapter 10, "Managing Disks (Overview)."

1 Become an administrator.

For more information, see "How to Use Your Assigned Administrative Rights" in Oracle Solaris Administration: Security Services.

2 Display the disk label information.

```
# prtvtoc /dev/rdsk/device-name
```

where `device-name` is the raw disk device you want to examine.

Example 11–7 Examining a Disk Label

The following example shows disk label information for a disk with a VTOC label.

```
# prtvtoc /dev/rdsk/c2t3d0s0
* /dev/rdsk/c2t3d0s0 partition map
* Dimensions:
* 512 bytes/sector
* 848 sectors/track
* 24 tracks/cylinder
* 20352 sectors/cylinder
* 14089 cylinders
* 14087 accessible cylinders
```
Recovering a Corrupted Disk Label

Sometimes, a power or system failure causes a disk’s label to become unrecognizable. A corrupted disk label doesn’t always mean that the slice information or the disk’s data must be re-created or restored.

The first step to recovering a corrupted disk label is to label the disk with the correct geometry and disk type information. You can complete this step through the normal disk labeling method, by using either automatic configuration or manual disk type specification.

If the format utility recognizes the disk type, the next step is to search for a backup label to label the disk. Labeling the disk with the backup label labels the disk with the correct partitioning information, the disk type, and disk geometry.

▼ How to Recover a Corrupted Disk Label

1 Boot the system to single-user mode.

If necessary, boot the system from a local Oracle Solaris DVD or the network in single-user mode to access the disk.
For information on booting the system, see *Booting and Shutting Down Oracle Solaris on SPARC Platforms* or *Booting and Shutting Down Oracle Solaris on x86 Platforms*.

2 **Relabel the disk.**

```
# format
```

The `format` utility attempts to automatically configure any unlabeled SCSI disk. If the `format` utility is able to configure the unlabeled and corrupted disk, it will display this message:

```
cwtxdy: configured with capacity of abcMB
```

The `format` utility then displays a numbered list of disks on the system.

3 **Type the number of the disk that you need to recover.**

```
Specify disk (enter its number): 1
```

4 **Select one of the following to determine how to label the disk.**

- If the disk was configured successfully, follow Steps 5 and 6. Then go to step 12.
- If the disk was not configured successfully, follow Steps 7–11. Then go to step 12.

5 **Search for the backup label.**

```
format> verify
```

**Warning:** Could not read primary label.

```
Warning: Check the current partitioning and 'label' the disk or use the 'backup' command.
```

Backup label contents:

```
Volume name = < >
ascii name = <FUJITSU-MAP3147N SUN146G-0501>
pcyl = 14089
cyl = 14087
acyl = 2
nhead = 24
nsect = 848
```

If the `format` utility was able to find a backup label and the backup label contents appear satisfactory, use the `backup` command to label the disk with the backup label.

```
format> backup
```

Searching for backup labels...found.

```
Restoring primary label
```

The disk label has been recovered. Go to Step 12.
Adding a Third-Party Disk

If the `format` utility was not able to automatically configure the disk, specify the disk type by using the `type` command.

```
format> type
```

The Available Drives Type menu is displayed.

Select 0 to automatically configure the disk. Or, select a disk type from the list of possible disk types.

Specify disk type (enter its number)[12]: 12

If the disk was successfully configured, reply with no when the `format` utility asks if you want to label the disk.

```
Disk not labeled. Label it now? no
```

Use the `verify` command to search for backup labels.

```
format> verify
Warning: Could not read primary label.
Warning: Check the current partitioning and 'label' the disk or use the 'backup' command.
```

If the `format` utility was able to find a backup label and the backup label contents appear satisfactory, use the `backup` command to label the disk with the backup label.

```
format> backup
Disk has a primary label, still continue? y
Searching for backup labels...found.
Restoring primary label
The disk label has been recovered.
```

Exit the `format` utility.

```
format> q
```

Verify the file systems on the recovered disk.

For information on using the `zpool scrub` command for ZFS file systems, see `zpool(1M)`.

---

Adding a Third-Party Disk

The Oracle Solaris OS supports many third-party disks. However, for the disk to be recognized, you might need to supply a device driver. Other options for adding disks are as follows:

- If you are adding a SCSI disk, you might try the `format` utility’s automatic configuration feature.
- You might try hot-plugging a PCI, SCSI, or USB disk. For more information, see Chapter 5, “Managing Devices (Overview/Tasks).”
Note – Oracle cannot guarantee that its `format` utility will work properly with all third-party disk drivers. If the disk driver is not compatible with the Solaris `format` utility, the disk drive vendor should supply you with a custom disk formatting program.

Typically, you discover that software support is missing when you invoke the `format` utility and find that the disk type is not recognized.

Refer to the appropriate configuration procedure for adding system disks or secondary disks in Chapter 12, “SPARC: Setting Up Disks (Tasks),” or Chapter 13, “x86: Setting Up Disks (Tasks).”
This chapter describes how to set up disks on a SPARC system.

For information on the procedures associated with setting up disks on a SPARC system, see “SPARC: Setting up Disks for ZFS File Systems (Task Map)” on page 217.

For overview information about disk management, see Chapter 10, "Managing Disks (Overview)." For step-by-step instructions on setting up disks on an x86 based system, see Chapter 13, "x86: Setting Up Disks (Tasks)."

### SPARC: Setting up Disks for ZFS File Systems (Task Map)

The following task map identifies the procedures for setting up a ZFS root pool disk for a ZFS root file system or a non-root ZFS pool disk on a SPARC based system.

<table>
<thead>
<tr>
<th>Task</th>
<th>Description</th>
<th>For Instructions</th>
</tr>
</thead>
</table>
| 1. Set up the disk for a ZFS root file system. | *Disk for a ZFS Root File System*  
Connect the new disk or replace the existing root pool disk and boot from a local or remote Oracle Solaris DVD. | “SPARC: How to Set Up a Disk for a ZFS Root File System” on page 219 |
| 2. Create a disk slice for a ZFS root file system | Create a disk slice for a disk that is intended for a ZFS root pool. This is a long-standing boot limitation. | “SPARC: How to Create a Disk Slice for a ZFS Root File System” on page 220 |
### SPARC: Setting Up Disks for ZFS File Systems

Although the procedures that describe how to set up a disk can be used with a ZFS file system, a ZFS file system is not directly mapped to a disk or a disk slice. You must create a ZFS storage pool before creating a ZFS file system. For more information, see *Oracle Solaris Administration: ZFS File Systems*.

The root pool contains the root file system that is used to boot the Oracle Solaris OS. If a root pool disk becomes damaged and the root pool is not mirrored, the system might not boot. If a root pool disk becomes damaged, you have two ways to recover:

- You can reinstall the entire Oracle Solaris OS.
- Or, you can replace the root pool disk and restore your file systems from snapshots or from a backup medium. You can reduce system down time due to hardware failures by creating a redundant root pool. The only supported redundant root pool configuration is a mirrored root pool.

A disk that is used in a non-root pool usually contains space for user or data files. You can attach or add another disk to a root pool or a non-root pool for more disk space. Or, you can replace a damaged disk in a pool in the following ways.

- A disk can be replaced in a non-redundant pool if all the devices are currently ONLINE.
- A disk can be replaced in a redundant pool if enough redundancy exists among the other devices.
- In a mirrored root pool, you can replace a disk or attach a disk and then detach the failed disk or a smaller disk to increase a pool's size.

In general, setting up a disk on the system depends on the hardware so review your hardware documentation when adding or replacing a disk on your system. If you need to add a disk to an existing controller, then it might just be a matter of inserting the disk in an empty slot, if the system supports hot-plugging. If you need to configure a new controller, see “Dynamic Reconfiguration and Hot-Plugging” on page 77.

<table>
<thead>
<tr>
<th>Task</th>
<th>Description</th>
<th>For Instructions</th>
</tr>
</thead>
<tbody>
<tr>
<td>3. Install the boot blocks for a ZFS root file system, if necessary.</td>
<td>If you replace a disk that is intended for the root pool by using the <code>zpool replace</code> command, then you must install the boot blocks manually so that the system can boot from the replacement disk.</td>
<td>“SPARC: How to Install Boot Blocks for a ZFS Root File System” on page 223</td>
</tr>
<tr>
<td>4. Set up a disk for ZFS file system.</td>
<td>Disk for a ZFS File System&lt;br&gt;Set up a disk for a ZFS file system.</td>
<td>“SPARC: How to Set Up a Disk for a ZFS File System” on page 224</td>
</tr>
</tbody>
</table>
**SPARC: How to Set Up a Disk for a ZFS Root File System**

Refer to your hardware installation guide for information on replacing a disk.

1. **Disconnect the damaged disk from the system, if necessary.**

2. **Connect the replacement disk to the system and check the disk’s physical connections, if necessary.**

3. **Follow the instructions in the following table, depending on whether you are booting from a local Oracle Solaris DVD or a remote Oracle Solaris DVD from the network.**

<table>
<thead>
<tr>
<th>Boot Type</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>From an Oracle Solaris DVD in a local drive</td>
<td>1. Make sure the Oracle Solaris DVD is in the drive.</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>2. Boot from the media to single-user mode:</td>
</tr>
<tr>
<td></td>
<td>ok boot cdrom -s</td>
</tr>
<tr>
<td>From the network</td>
<td>Boot from the network to single-user mode:</td>
</tr>
<tr>
<td></td>
<td>ok boot net -s</td>
</tr>
</tbody>
</table>

After a few minutes, the root prompt (#) is displayed.

**More Information**

*After You Set Up a Disk for a ZFS Root File System...*

After the disk is connected or replaced, you can create a slice and update the disk label. Go to “**SPARC: How to Create a Disk Slice for a ZFS Root File System**” on page 220.

**SPARC: Creating a Disk Slice for a ZFS Root File System**

You must create a disk slice for a disk that is intended for a ZFS root pool. This is a long-standing boot limitation. Review the following root pool disk requirements:

- Must contain a disk slice and an SMI (VTOC) label.
- An EFI label is not supported for a root pool disk.
- Must be a single disk or be part of mirrored configuration. A non-redundant configuration nor a RAIDZ configuration is supported for the root pool.
- All subdirectories of the root file system that are part of the OS image, with the exception of /var, must be in the same dataset as the root file system.
- All Solaris OS components must reside in the root pool, with the exception of the swap and dump devices.
In general, you should create a disk slice with the bulk of disk space in slice 0. Attempting to use different slices on a disk and share that disk among different operating systems or with a different ZFS storage pool or storage pool components is not recommended.

▼ **SPARC: How to Create a Disk Slice for a ZFS Root File System**

In general, the root pool disk is installed automatically when the system is installed. If you need to replace a root pool disk or attach a new disk as a mirrored root pool disk, see the steps below.

1. **Become an administrator.**

2. **Offline and unconfigure the failed disk, if necessary.**
   Some hardware requires that you offline and unconfigure a disk before attempting the `zpool replace` operation to replace a failed disk. For example:
   ```
   # zpool offline rpool c2t1d0s0
   # cfgadm -c unconfigure c2::dsk/c2t1d0
   ```

3. **Physically connect the new or replacement disk to the system, if necessary.**
   a. Physically remove the failed disk.
   b. Physically insert the replacement disk.
   c. Configure the replacement disk, if necessary. For example:
      ```
      # cfgadm -c configure c2::dsk/c2t1d0
      ```
      On some hardware, you do not have to reconfigure the replacement disk after it is inserted.

4. **Confirm that the disk is accessible by reviewing the format output.**
   For example, the `format` command sees 4 disks connected to this system.
   ```
   # format -e
   AVAILABLE DISK SELECTIONS:
   0. c2t0d0 <SUN36G cyl 24620 alt 2 hd 27 sec 107>
      /pci@1c,600000/scsi@2/sd@0,0
   1. c2t1d0 <SEAGATE-ST336607LSUN36G-0307-33.92GB>
      /pci@1c,600000/scsi@2/sd@1,0
   2. c2t2d0 <SEAGATE-ST336607LSUN36G-0507-33.92GB>
      /pci@1c,600000/scsi@2/sd@2,0
   3. c2t3d0 <SEAGATE-ST336607LSUN36G-0507-33.92GB>
      /pci@1c,600000/scsi@2/sd@3,0
   ```

5. **Select the disk to be used for the ZFS root pool.**
6 Confirm that the disk has an SMI label by displaying the partition (slice) information.

For example, the partition (slice) output for c2t1d0 shows that this disk has an EFI label because it identifies first and last sectors.

Specify disk (enter its number): 1
selecting c2t1d0
[disk formatted]
format> p
PARTITION MENU:
0 - change '0' partition
1 - change '1' partition
2 - change '2' partition
3 - change '3' partition
4 - change '4' partition
5 - change '5' partition
6 - change '6' partition
expand - expand label to use whole disk
select - select a predefined table
modify - modify a predefined partition table
name - name the current table
print - display the current table
label - write partition map and label to the disk
!<cmd> - execute <cmd>, then return
quit
partition> p
Current partition table (original):
Total disk sectors available: 71116508 + 16384 (reserved sectors)

<table>
<thead>
<tr>
<th>Part</th>
<th>Tag</th>
<th>Flag</th>
<th>First Sector</th>
<th>Size</th>
<th>Last Sector</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>usr</td>
<td>wm</td>
<td>256</td>
<td>33.91GB</td>
<td>71116541</td>
</tr>
<tr>
<td>1</td>
<td>unassigned</td>
<td>wm</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>unassigned</td>
<td>wm</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>unassigned</td>
<td>wm</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>4</td>
<td>unassigned</td>
<td>wm</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>5</td>
<td>unassigned</td>
<td>wm</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>6</td>
<td>unassigned</td>
<td>wm</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>8</td>
<td>reserved</td>
<td>wm</td>
<td>71116542</td>
<td>8.00MB</td>
<td>71132925</td>
</tr>
</tbody>
</table>

partition>

7 If the disk contains an EFI label, relabel the disk with an SMI label.

For example, the c2t1d0 disk is relabeled with an SMI label, but the default partition table does not provide an optimal slice configuration.

partition> label
[0] SMI Label
[1] EFI Label
Specify Label type[1]: 0
Auto configuration via format.dat[no]? Auto configuration via generic SCSI-2[no]? partition> p
Current partition table (default):
Total disk cylinders available: 24620 + 2 (reserved cylinders)

<table>
<thead>
<tr>
<th>Part</th>
<th>Tag</th>
<th>Flag</th>
<th>Cylinders</th>
<th>Size</th>
<th>Blocks</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>root</td>
<td>wm</td>
<td>0 - 90</td>
<td>128.37MB</td>
<td>262899</td>
</tr>
<tr>
<td>1</td>
<td>swap</td>
<td>wu</td>
<td>91 - 181</td>
<td>128.37MB</td>
<td>262899</td>
</tr>
</tbody>
</table>
8 Create an optimal slice configuration for a ZFS root pool disk.

Set the free hog partition so that all the unallocated disk space is collected in slice 0. Then, press return through the slice size fields to create one large slice 0.

partition> modify
Select partitioning base:
  0. Current partition table (default)
  1. All Free Hog
Choose base (enter number) [0]? 1

Do you wish to continue creating a new partition table based on above table[yes]?
Free Hog partition[6]? 0
Enter size of partition '1' [0b, 0c, 0.00mb, 0.00gb]:
Enter size of partition '3' [0b, 0c, 0.00mb, 0.00gb]:
Enter size of partition '4' [0b, 0c, 0.00mb, 0.00gb]:
Enter size of partition '5' [0b, 0c, 0.00mb, 0.00gb]:
Enter size of partition '6' [0b, 0c, 0.00mb, 0.00gb]:
Enter size of partition '7' [0b, 0c, 0.00mb, 0.00gb]:

Okay to make this the current partition table[yes]?
Enter table name (remember quotes): "c2t1d0"
Ready to label disk, continue? yes
partition> quit
format> quit
Let ZFS know that the failed disk is replaced.

```
# zpool replace rpool c2t1d0s0
# zpool online rpool c2t1d0s0
```

On some hardware, you do not have to online the replacement disk after it is inserted.

If you are attaching a new disk to create a mirrored root pool or attaching a larger disk to replace a smaller disk, use syntax similar to the following:

```
# zpool attach rpool c0t0d0s0 c1t0d0s0
```

If a root pool disk is replaced with a new disk, apply the boot blocks after the new or replacement disk is resilvered.

For example:

```
# zpool status rpool
installboot -F zfs /usr/platform/'uname -i'/lib/fs/zfs/bootblk /dev/rdsk/c2t1d0s0
```

Verify that you can boot from the new disk.

11 If the system boots from the new disk, detach the old disk.

This step is only necessary if you attach a new disk to replace a failed disk or a smaller disk.

```
# zpool detach rpool c0t0d0s0
```

12 Set up the system to boot automatically from the new disk, either by using the eeprom command or the setenv command from the SPARC boot PROM.

More Information  After You Have Created a Disk Slice for a ZFS Root File System ...

After you have created a disk slice for the ZFS root file system and you need to restore root pool snapshots to recover your root pool, see “How to Replace a Disk in a ZFS Root Pool” in Oracle Solaris Administration: ZFS File Systems.

▼ SPARC: How to Install Boot Blocks for a ZFS Root File System

1 Become an administrator.

2 Install a boot block for a ZFS root file system.

```
# installboot -F zfs /usr/platform/'uname -i'/lib/fs/zfs/bootblk /dev/rdsk/cwtXdys0
```

For more information, see installboot(1M).
3 Verify that the boot blocks are installed by rebooting the system to run level 3.

```
# init 6
```

**Example 12–1**  SPARC: Installing Boot Blocks for a ZFS Root File System

If you physically replace the disk that is intended for the root pool and the Oracle Solaris OS is then reinstalled, or you attach a new disk for the root pool, the boot blocks are installed automatically. If you replace a disk that is intended for the root pool by using the `zpool replace` command, then you must install the boot blocks manually so that the system can boot from the replacement disk.

The following example shows how to install boot blocks for a ZFS root file system.

```
# installboot -F zfs /usr/platform/`uname -i`/lib/fs/zfs/bootblk /dev/rdsk/c0t1d0s0
```

▼ **SPARC: How to Set Up a Disk for a ZFS File System**

If you are setting up a disk to be used with a non-root ZFS file system, the disk is relabeled automatically when the pool is created or when the disk is added to the pool. If a pool is created with whole disks or when a whole disk is added to a ZFS storage pool, an EFI label is applied. For more information about EFI disk labels, see "EFI Disk Label" on page 187.

Generally, most modern bus types support hot-plugging. This means you can insert a disk in an empty slot and the system recognizes it. For more information about hot-plugging devices, see Chapter 6, "Dynamically Configuring Devices (Tasks)."

1 Become an administrator.

2 Connect the disk to the system and check the disk's physical connections.
Refer to the disk's hardware installation guide for details.

3 Offline and unconfigure the failed disk, if necessary.
Some hardware requires that you offline and unconfigure a disk before attempting the `zpool replace` operation to replace a failed disk. For example:

```
# zpool offline tank clt1d0
# cfgadm -c unconfigure cl::dsk/clt1d0
<Physically remove failed disk clt1d0>
<Physically insert replacement disk clt1d0>
# cfgadm -c configure cl::dsk/clt1d0
```

On some hardware, you do not to reconfigure the replacement disk after it is inserted.
4 **Confirm that the new disk is recognized.**
Review the output of the `format` utility to see if the disk is listed under AVAILABLE DISK SELECTIONS. Then, quit the `format` utility.
# format

5 **Let ZFS know that the failed disk is replaced, if necessary.**
# zpool replace tank c1t1d0
# zpool online tank c1t1d0
Confirm that the new disk is resilvering.

# zpool status tank

6 **Attach a new disk to an existing ZFS storage pool, if necessary.**
For example:
# zpool attach tank mirror c1t0d0 c2t0d0
Confirm that the new disk is resilvering.

# zpool status tank
For more information, see Chapter 4, “Managing Oracle Solaris ZFS Storage Pools,” in *Oracle Solaris Administration: ZFS File Systems.*
This chapter describes how to set up disks on an x86 based system.

For information on the procedures associated with setting up disks on an x86 based system, see the following sections:

- “SPARC: Setting up Disks for ZFS File Systems (Task Map)” on page 217
- “Creating and Changing Solaris fdisk Partitions” on page 235

For overview information about disk management, see Chapter 10, “Managing Disks (Overview).” For step-by-step instructions on setting up disks on a SPARC based system, see Chapter 12, “SPARC: Setting Up Disks (Tasks).”

**x86: Setting Up Disks for ZFS File Systems (Task Map)**

The following task map identifies the procedures for setting up a ZFS root pool disk for a ZFS root file system on an x86 based system.

<table>
<thead>
<tr>
<th>Task</th>
<th>Description</th>
<th>For Instructions</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Set up the disk for a ZFS root file system.</td>
<td>Disk for a ZFS Root File System Connect the new disk or replace the existing root pool disk and boot from a local or remote Oracle Solaris DVD.</td>
<td>“x86: How to Set Up a Disk for a ZFS Root File System” on page 229</td>
</tr>
<tr>
<td>2. Create or change an fdisk partition, if necessary.</td>
<td>The disk must contain a valid Solaris fdisk partition.</td>
<td>“Creating and Changing Solaris fdisk Partitions” on page 235</td>
</tr>
<tr>
<td>3. Create a disk slice for the ZFS root file system.</td>
<td>Create a disk slice for a disk that is intended for a ZFS root pool. This is a long-standing boot limitation.</td>
<td>“x86: How to Create a Solaris fdisk Partition” on page 236 and “x86: How to Create a Disk Slice for a ZFS Root File System” on page 230</td>
</tr>
</tbody>
</table>
4. Install the boot blocks for a ZFS root file system.
   If you replace a disk that is intended for the root pool by using the `zpool replace` command, then you must install the boot blocks manually so that the system can boot from the replacement disk.
   "x86: How to Install Boot Blocks for a ZFS Root File System" on page 233

5. Set up a disk for a ZFS file system.
   "Disk for a ZFS File System" Connect the disk.
   "x86: How to Set Up a Disk for a ZFS File System" on page 234

---

### x86: Setting Up Disks for ZFS File Systems

Although the procedures that describe how to set up a disk and create an fdisk partition can be used with a ZFS file systems, a ZFS file system is not directly mapped to a disk or a disk slice. You must create a ZFS storage pool before creating a ZFS file system. For more information, see Oracle Solaris Administration: ZFS File Systems.

The root pool contains the root file system that is used to boot the Oracle Solaris OS. If a root pool disk becomes damaged and the root pool is not mirrored, the system might not boot. If a root pool disk becomes damaged, you have two ways to recover:

- You can reinstall the entire Oracle Solaris OS.
- Or, you can replace the root pool disk and restore your file systems from snapshots or from a backup medium. You can reduce system down time due to hardware failures by creating a redundant root pool. The only supported redundant root pool configuration is a mirrored root pool.

A disk that is used in a non-root pool usually contains space for user or data files. You can attach or add another disk to a root pool or a non-root pool for more disk space. Or, you can replace a damaged disk in a pool in the following ways:

- A disk can be replaced in a non-redundant pool if all the devices are currently ONLINE.
- A disk can be replaced in a redundant pool if enough redundancy exists among the other devices.
- In a mirrored root pool, you can replace a disk or attach a disk and then detach the failed disk or a smaller disk to increase a pool’s size.

In general, setting up a disk to the system depends on the hardware so review your hardware documentation when adding or replacing a disk on your system. If you need to add a disk to an existing controller, then it might just be a matter of inserting the disk in an empty slot, if the system supports hot-plugging. If you need to configure a new controller, see “Dynamic Reconfiguration and Hot-Plugging” on page 77.
**x86: How to Set Up a Disk for a ZFS Root File System**

Refer to your hardware installation guide for information on replacing a disk.

1. **Disconnect the damaged disk from the system, if necessary.**
2. **Connect the replacement disk to the system, and check the disk’s physical connections.**
3. **Follow the instructions in the following table, depending on whether you are booting from a local Oracle Solaris DVD or a remote Oracle Solaris DVD from the network.**

<table>
<thead>
<tr>
<th>Boot Type</th>
<th>Action</th>
</tr>
</thead>
</table>
| From an Oracle Solaris DVD in a local drive | 1. Make sure the Oracle Solaris DVD is in the drive.  
2. Select the option to boot from the media. |
| From the network | 3. Select the option to boot from the network. |

**More Information**

After you set up a disk for a ZFS root file system...

After the disk is connected or replaced, create an fdisk partition. Go to “x86: How to Create a Solaris fdisk Partition” on page 236.

**x86: Creating a Disk Slice for a ZFS Root File System**

You must create a disk slice for a disk that is intended for a ZFS root pool. This is a long-standing boot limitation. Review the following root pool disk requirements:

- Must contain a disk slice and an SMI (VTOC) label.
- An EFI label is not supported for a root pool disk.
- A root pool disk on an x86 system must contain an fdisk partition.
- Must be a single disk or be part of mirrored configuration. A non-redundant configuration nor a RAIDZ configuration is supported for the root pool.
- All subdirectories of the root file system that are part of the OS image, with the exception of /var, must be in the same dataset as the root file system.
- All Solaris OS components must reside in the root pool, with the exception of the swap and dump devices.

On an x86 based system, you must first create an fdisk partition. Then, create a disk slice with the bulk of disk space in slice 0.

Attempting to use different slices on a disk and share that disk among different operating systems or with a different ZFS storage pool or storage pool components is not recommended.
How to Create a Disk Slice for a ZFS Root File System

In general, the root pool disk is installed automatically when the system is installed. If you need to replace a root pool disk or attach a new disk as a mirrored root pool disk, see the steps below.

For a full description of fdisk partitions, see "x86: Guidelines for Creating an fdisk Partition" on page 235.

1 Become an administrator.

2 Offline and unconfigure the failed disk, if necessary.
   Some hardware requires that you offline and unconfigure a disk before attempting the zpool replace operation to replace a failed disk. For example:
   
   ```
   # zpool offline rpool c2t1d0s0
   # cfgadm -c unconfigure c2::dsk/c2t1d0
   ```

3 Physically connect the new or replacement disk to the system, if necessary.

   a. Physically remove the failed disk.

   b. Physically insert the replacement disk.

   c. Configure the replacement disk, if necessary. For example:
      
      ```
      # cfgadm -c configure c2::dsk/c2t1d0
      ```
      
      On some hardware, you do not have to reconfigure the replacement disk after it is inserted.

4 Confirm that the disk is accessible by reviewing the format output.
   For example, the format command sees 4 disks connected to this system.
   
   ```
   # format -e
   AVAILABLE DISK SELECTIONS:
   1. c8t0d0 <Sun-STK RAID INT-V1.0 cyl 17830 alt 2 hd 255 sec 63>
      /pci@0,0/pci10de,375f/pci108e,2860/disk0,0
   2. c8t1d0 <Sun-STK RAID INT-V1.0-136.61GB>
      /pci@0,0/pci10de,375f/pci108e,2860/disk1,0
   3. c8t2d0 <Sun-STK RAID INT-V1.0-136.61GB>
      /pci@0,0/pci10de,375f/pci108e,2860/disk2,0
   4. c8t3d0 <Sun-STK RAID INT-V1.0-136.61GB>
      /pci@0,0/pci10de,375f/pci108e,2860/disk3,0
   ```

5 Select the disk to be used for the ZFS root pool.
   Specify disk (enter its number): 1
   selecting c8t0d0
   [disk formatted]
   .
   .
   .
   format>
6 **Review the status of the fdisk partition.**

- If the disk has no fdisk partition, you will see a message similar to the following:
  
  ```
  format> fdisk
  No Solaris fdisk partition found.
  ```

  If so, go to step 4 to create an fdisk partition.

- If the disk has an EFI fdisk or some other partition type, go to step 5 to create a Solaris fdisk partition.

- If the disk has a Solaris fdisk partition, go to step 6 to create a disk slice for the root pool.

7 **If necessary, create a Solaris fdisk partition by selecting the fdisk option.**

```
format> fdisk
No fdisk table exists. The default partition for the disk is:

a 100% "SOLARIS System" partition

Type 'y' to accept the default partition, otherwise type 'n' to edit the partition table. y
```

Then, go to step 6 to create a disk slice for the root pool.

8 **If the disk has an EFI fdisk partition, then you will need to create a Solaris fdisk partition.**

If you print the disk’s partition table with the format utility, and you see the partition table refers to the first sector and the size, then this is an EFI partition. You will need to create a Solaris fdisk partition as follows:

- Select fdisk from the format options.

```bash
# format -e c8t0d0
selecting c8t0d0
[disk formatted]
format> fdisk
```

- Delete the existing EFI partition by selecting option 3, Delete a partition.

```plaintext
Enter Selection: 3
Specify the partition number to delete (or enter 0 to exit): 1
Are you sure you want to delete partition 1? This will make all files and programs in this partition inaccessible (type "y" or "n"). y
```

Partition 1 has been deleted.

- Create a new Solaris partition by selecting option 1, Create a partition.

```plaintext
Enter Selection: 1
Select the partition type to create: 1
Specify the percentage of disk to use for this partition (or type "c" to specify the size in cylinders). 100
Should this become the active partition? If yes, it will be activated each time the computer is reset or turned on. Please type "y" or "n". y
```

Partition 1 is now the active partition.
- Update the disk configuration and exit.
  
  Enter Selection: 6
  
  format>

- Display the SMI partition table. If the default partition table is applied, then slice 0 might be 0 in size or it might be too small. See the next step.
  
  format> partition
  partition> print

9 Confirm that the disk has an SMI label by displaying the partition (slice) information and review the slice 0 size information.

Set the free hog partition so that all the unallocated disk space is collected in slice 0. Then, press return through the slice size fields to create one large slice 0.

partition> modify
  Select partitioning base:
  0. Current partition table (default)
  1. All Free Hog

Choose base (enter number) [0]? 1

<table>
<thead>
<tr>
<th>Part</th>
<th>Tag</th>
<th>Flag</th>
<th>Cylinders</th>
<th>Size</th>
<th>Blocks</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>root wu 0 0 (0/0/0) 0</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>swap wu 0 0 (0/0/0) 0</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>backup wu 0 - 17829 136.58GB (17830/0/0) 286438950</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>unassigned wu 0 0 (0/0/0) 0</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>unassigned wu 0 0 (0/0/0) 0</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>unassigned wu 0 0 (0/0/0) 0</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>usr wu 0 0 (0/0/0) 0</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>unassigned wu 0 0 (0/0/0) 0</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>boot wu 0 - 0 7.84MB (1/0/0) 16065</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>alternates wu 0 0 (0/0/0) 0</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Do you wish to continue creating a new partition table based on above table[yes]?

Free Hog partition[6]? 0

Enter size of partition ‘1’ [0b, 0c, 0.00mb, 0.00gb]:
Enter size of partition ‘3’ [0b, 0c, 0.00mb, 0.00gb]:
Enter size of partition ‘4’ [0b, 0c, 0.00mb, 0.00gb]:
Enter size of partition ‘5’ [0b, 0c, 0.00mb, 0.00gb]:
Enter size of partition ‘6’ [0b, 0c, 0.00mb, 0.00gb]:
Enter size of partition ‘7’ [0b, 0c, 0.00mb, 0.00gb]:

Do you wish to continue creating a new partition table based on above table[yes]? yes

Enter table name (remember quotes): "c8t0d0"

Ready to label disk, continue? yes
Let ZFS know that the failed disk is replaced.

```
# zpool replace rpool c2t1d0s0
# zpool online rpool c2t1d0s0
```

On some hardware, you do not have to online the replacement disk after it is inserted.

If you are attaching a new disk to create a mirrored root pool or attaching a larger disk to replace
a smaller disk, use syntax similar to the following:

```
# zpool attach rpool c0t0d0s0 c1t0d0s0
```

If a root pool disk is replaced with a new disk, apply the boot blocks.

For example:

```
# installgrub /boot/grub/stage1 /boot/grub/stage2 /dev/rdsk/c8t0d0s0
```

Verify that you can boot from the new disk.

If the system boots from the new disk, detach the old disk.

This step is only necessary if you attach a new disk to replace a failed disk or a smaller disk.

```
# zpool detach rpool c0t0d0s0
```

Set up the system to boot automatically from the new disk by reconfiguring the system's BIOS.

More Information

After You Have Created a Disk Slice for the ZFS Root File System ...

After you have created a disk slice for the ZFS root file system and you need to restore root pool
snapshots to recover your root pool, see "How to Replace a Disk in a ZFS Root Pool" in Oracle
Solaris Administration: ZFS File Systems.

x86: How to Install Boot Blocks for a ZFS Root File System

1 Become an administrator.

2 Install the boot blocks on the system disk.

```
# installgrub /boot/grub/stage1 /boot/grub/stage2 /dev/rdsk/cwtxdz/s
```

For more information, see `installgrub(1M)`.

3 Verify that the boot blocks are installed by rebooting the system to run level 3.

```
# init 6
```

Example 13–1 x86: Installing Boot Blocks for a ZFS Root File System

If you physically replace the disk that is intended for the root pool and the Oracle Solaris OS is
then reinstalled, or you attach a new disk for the root pool, the boot blocks are installed
automatically. If you replace a disk that is intended for the root pool by using the `zpool replace` command, then you must install the boot blocks manually so that the system can boot from the replacement disk.

The following example shows how to install the boot blocks for a ZFS root file system.

```
# installgrub /boot/grub/stage1 /boot/grub/stage2 /dev/rdsk/c1d0s0
stage2 written to partition 0, 277 sectors starting at 50 (abs 16115)
stage1 written to partition 0 sector 0 (abs 16065)
```

### x86: How to Set Up a Disk for a ZFS File System

If you are setting up a disk to be used with a non-root ZFS file system, the disk is relabeled automatically when the pool is created or when the disk is added to the pool. If a pool is created with whole disks or when a whole disk is added to a ZFS storage pool, an EFI label is applied. For more information about EFI disk labels, see “EFI Disk Label” on page 187.

Generally, most modern bus types support hot-plugging. This means you can insert a disk in an empty slot and the system recognizes it. For more information about hot-plugging devices, see Chapter 6, “Dynamically Configuring Devices (Tasks).”

1 **Become an administrator.**

   For more information, see “How to Use Your Assigned Administrative Rights” in Oracle Solaris Administration: Security Services.

2 **Connect the disk to the system and check the disk’s physical connections.**

   Refer to the disk’s hardware installation guide for details.

3 **Offline and unconfigure the failed disk, if necessary.**

   Some hardware requires that you offline and unconfigure a disk before attempting the `zpool replace` operation to replace a failed disk. For example:

   ```
   # zpool offline tank c1t1d0
   # cfgadm -c unconfigure c1::dsk/c1t1d0
   <Physically remove failed disk c1t1d0>
   <Physically insert replacement disk c1t1d0>
   # cfgadm -c configure c1::dsk/c1t1d0
   ```

   On some hardware, you do not to reconfigure the replacement disk after it is inserted.

4 **Confirm that the new disk is recognized.**

   Review the output of the `format` utility to see if the disk is listed under AVAILABLE DISK SELECTIONS. Then, quit the `format` utility.

   ```
   # format
   ```

5 **Let ZFS know that the failed disk is replaced, if necessary.**

   ```
   # zpool replace tank c1t1d0
   # zpool online tank c1t1d0
   ```
Confirm that the new disk is resilvering.

```
# zpool status tank
```

**6 Attach a new disk to an existing ZFS storage pool, if necessary.**

For example:
```
# zpool attach tank mirror c1t0d0 c2t0d0
```

Confirm that the new disk is resilvering.

```
# zpool status tank
```

For more information, see Chapter 4, "Managing Oracle Solaris ZFS Storage Pools," in Oracle Solaris Administration: ZFS File Systems.

---

## Creating and Changing Solaris fdisk Partitions

Review the following sections for guidelines and examples of creating or changing Solaris fdisk partitions.

### x86: Guidelines for Creating an fdisk Partition

Follow these guidelines when you set up one or more fdisk partitions.

- The disk can be divided into a maximum of four fdisk partitions. One of partitions must be a Solaris partition.
- The Solaris partition must be made the active partition on the disk. The active partition is partition whose operating system will be booted by default at system startup.
- Solaris fdisk partitions must begin on cylinder boundaries.
- Solaris fdisk partitions must begin at cylinder 1, not cylinder 0, on the first disk because additional boot information, including the master boot record, is written in sector 0.
- The Solaris fdisk partition can be the entire disk. Or, you might want to make it smaller to allow room for a DOS partition. You can also make a new fdisk partition on a disk without disturbing existing partitions (if sufficient space is available) to create a new partition.

---

**x86 only** – Solaris slices are also called partitions. Certain interfaces might refer to a slice as a partition.

fdisk partitions are supported only on x86 based systems. To avoid confusion, Oracle Solaris documentation tries to distinguish between fdisk partitions and the entities within the Solaris fdisk partition. These entities might be called slices or partitions.
**x86: How to Create a Solaris fdisk Partition**

**Before You Begin**
If you need information about fdisk partitions, see "x86: Guidelines for Creating an fdisk Partition" on page 235.

1. **Become an administrator.**
   For more information, see "How to Use Your Assigned Administrative Rights" in Oracle Solaris Administration: Security Services.

2. **Invoke the format utility.**
   ```bash
   # format
   ```
   A numbered list of disks is displayed.
   For more information, see `format(1M)`.  

3. **Type the number of the disk on which to create a Solaris fdisk partition.**
   Specify disk (enter its number): `disk-number`
   where `disk-number` is the number of the disk on which you want to create a Solaris fdisk partition.

4. **Select the fdisk menu.**
   ```bash
   format> fdisk
   ```
   The fdisk menu that is displayed depends upon whether the disk has existing fdisk partitions.

5. **Create and activate a Solaris fdisk partition that spans the entire disk by specifying y at the prompt. Then, go to step 13.**
   No fdisk table exists. The default partition for the disk is:
   a 100% "SOLARIS System" partition
   Type "y" to accept the default partition, otherwise type "n" to edit the partition table.
   ```bash
   y
   ```

6. **Specify n at the prompt if you do not want the Solaris fdisk partition to span the entire disk.**
   Type "y" to accept the default partition, otherwise type "n" to edit the partition table.
   ```bash
   n
   ```
   Total disk size is 17848 cylinders
   Cylinder size is 16065 (512 byte) blocks

<table>
<thead>
<tr>
<th>Partition</th>
<th>Status</th>
<th>Type</th>
<th>Start</th>
<th>End</th>
<th>Length</th>
<th>%</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

---

236 Oracle Solaris Administration: Devices and File Systems  •  February 2012
SELECT ONE OF THE FOLLOWING:
1. Create a partition
2. Specify the active partition
3. Delete a partition
4. Change between Solaris and Solaris2 Partition IDs
5. Edit/View extended partitions
6. Exit (update disk configuration and exit)
7. Cancel (exit without updating disk configuration)

Enter Selection:

7 Select option 1, Create a partition, to create an fdisk partition.

Enter Selection: 1

8 Create a Solaris fdisk partition by selecting 1 (=Solaris2).

Select the partition type to create:
1=SOLARIS2  2=UNIX  3=PCIXOS  4=Other  5=00512
6=00516  7=005EXT  8=005BIG  9=00516LBA  A=x86 Boot
B=Diagnostic  C=FAT32  D=FAT32LBA  E=005EXTLBA  F=EFI (Protective)
G=EFI_SYS  0=Exit?

9 Identify the percentage of the disk to be reserved for the Solaris fdisk partition. Keep in mind the size of any existing fdisk partitions when you calculate this percentage.

Specify the percentage of disk to use for this partition (or type "c" to specify the size in cylinders). nn

10 Activate the Solaris fdisk partition by typing y at the prompt.

Should this become the active partition? If yes, it will be activated each time the computer is reset or turned on.
Please type "y" or "n". y

The Enter Selection prompt is displayed after the fdisk partition is activated.

11 Select option 1, Create a partition, to create another fdisk partition.

See steps 8–10 for instructions on creating an fdisk partition.

12 Update the disk configuration, and exit the fdisk menu from the selection menu.

Selection: 6

13 Relabel the disk by using the label command.

format> label
Ready to label disk, continue? yes
format>

14 Quit the format utility.

format> quit
Creating and Changing Solaris fdisk Partitions

Example 13–2  x86: Creating a Solaris fdisk Partition That Spans the Entire Drive

The following example uses the format utility’s fdisk option to create a Solaris fdisk partition that spans the entire drive.

```
# format
Searching for disks...done
AVAILABLE DISK SELECTIONS:
  0. c0d0 <DEFAULT cyl 2466 alt 2 hd 16 sec 63>
      /pci@0,0/pci-ide@7,1/ide@0/cmdk@0,0
  1. c0d1 <DEFAULT cyl 522 alt 2 hd 32 sec 63>
      /pci@0,0/pci-ide@7,1/ide@0/cmdk@1,0
  2. c1d0 <DEFAULT cyl 13102 alt 2 hd 16 sec 63>
      /pci@0,0/pci-ide@7,1/ide@1/cmdk@0,0

Specify disk (enter its number): 0
selecting c0d0
Controller working list found
[disk formatted]
format> fdisk
No fdisk table exists. The default partitioning for your disk is:

    a 100% "SOLARIS System" partition.

Type "y" to accept the default partition, otherwise type "n" to edit the partition table. y
```

More Information  After You Create a Solaris fdisk Partition …

After you create a Solaris fdisk partition on the disk, you can use it for a root pool disk or non-root pool disk. For more information, see “x86: Setting Up Disks for ZFS File Systems” on page 228.

Changing the fdisk Partition Identifier

The Solaris fdisk partition identifier on x86 systems has been changed from 130 (0x82) to 191 (0xbf). All Oracle Solaris commands, utilities, and drivers have been updated to work with either fdisk identifier. There is no change in fdisk functionality.

How to Change the Solaris fdisk Identifier

A new fdisk menu option enables you to switch back and forth between the new and old identifier. The fdisk identifier can be changed even when the file system that is contained in the partition is mounted.

Two type values in the fdisk menu reflect the old and new identifiers as follows:
Solaris identifies 0x82
Solaris2 identifies 0xbf

1 **Become an administrator.**

2 **Display the current fdisk identifier.**

   For example:

   Total disk size is 39890 cylinders
   Cylinder size is 4032 (512 byte) blocks

<table>
<thead>
<tr>
<th>Partition</th>
<th>Status</th>
<th>Type</th>
<th>Start</th>
<th>End</th>
<th>Length</th>
<th>%</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Active</td>
<td>x86 Boot</td>
<td>1</td>
<td>6</td>
<td>6</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td></td>
<td>Solaris2</td>
<td>7</td>
<td>39889</td>
<td>39883</td>
<td>100</td>
</tr>
</tbody>
</table>

3 **Select option 4 from the fdisk menu to change the fdisk partition identifier back to 0x82.**

   SELECT ONE OF THE FOLLOWING:
   1. Create a partition
   2. Specify the active partition
   3. Delete a partition
   4. Change between Solaris and Solaris2 Partition IDs
   5. Exit (update disk configuration and exit)
   6. Cancel (exit without updating disk configuration)

   Enter Selection: 4

4 **Select option 5 to update your disk configuration and exit.**

5 **If necessary, select option 4 from the fdisk menu to change the fdisk partition identifier back to 0xbf.**

   For example:

   Total disk size is 39890 cylinders
   Cylinder size is 4032 (512 byte) blocks

<table>
<thead>
<tr>
<th>Partition</th>
<th>Status</th>
<th>Type</th>
<th>Start</th>
<th>End</th>
<th>Length</th>
<th>%</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Active</td>
<td>x86 Boot</td>
<td>1</td>
<td>6</td>
<td>6</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td></td>
<td>Solaris</td>
<td>7</td>
<td>39889</td>
<td>39883</td>
<td>100</td>
</tr>
</tbody>
</table>

   SELECT ONE OF THE FOLLOWING:
   1. Create a partition
   2. Specify the active partition
   3. Delete a partition
   4. Change between Solaris and Solaris2 Partition IDs
   5. Exit (update disk configuration and exit)
   6. Cancel (exit without updating disk configuration)

   Enter Selection: 4

6 **Select option 5 to update your disk configuration and exit.**
This chapter describes how to configure Common Multiprotocol SCSI TARget, or COMSTAR, a software framework that enables you to convert any Oracle Solaris 11 host into a SCSI target device that can be accessed over a storage network by initiator hosts.

This means you can make storage devices on a system available to Linux, Mac OS, or Windows client systems as if they were local storage devices. Supported storage protocols are iSCSI, FC, iSER, and SRP.

For information about the iSNS support in Oracle Solaris, see Chapter 15, “Configuring and Managing the Oracle Solaris Internet Storage Name Service (iSNS).”

For troubleshooting iSCSI configuration problems in Oracle Solaris, see “Troubleshooting iSCSI Configuration Problems” on page 283.

COMSTAR and iSCSI Technology (Overview)

iSCSI is an acronym for Internet SCSI (Small Computer System Interface), an Internet Protocol (IP)-based storage networking standard for linking data storage subsystems.

By carrying SCSI commands over IP networks, the iSCSI protocol enables you to access block devices from across the network as if they were connected to the local system. COMSTAR provides an easier way to manage these iSCSI target devices.

COMSTAR utilizes a SCSI Target Mode Framework (STMF) to manage target storage devices with the following components:

- Port providers (or plug-ins) – Implement protocols, such as Fibre Channel (FC) and iSCSI.
- Logical unit providers – Emulate various SCSI devices, such as disk and tape devices.
- The libstmf management library – Provides the COMSTAR management interface. The modules that implement the iSCSI functionality do not interact directly with the underlying transport. In a similar way, the modules that implement the transport protocol are unaware
of the SCSI-level functionality that is inherent in the packets they are transporting. Some transport examples are Fibre Channel and iSCSI. The framework separates the execution and cleanup of SCSI commands and the associated resources. This separation simplifies the task of writing SCSI or transport modules.

Use the following to administer these features:

- The `itadm` command manages Internet SCSI (iSCSI) nodes within the SCSI target mode framework.
- The `stmfadm` command configures logical units within the SCSI target mode framework.
- The `srptadm` command manages SCSI RDMA Protocol (SRP) target ports within the SCSI target mode framework.

The following solutions are available to use storage devices in your existing TCP/IP network:

- iSCSI block devices or tape – Translates SCSI commands and data from the block level into IP packets. Using iSCSI in your network is advantageous when you need to have block-level access between one system and the target device, such as a tape device or a database. Access to a block-level device is not locked so that you can have multiple users or systems accessing a block-level device such as an iSCSI target device.
- NFS – Transfers file data over IP. The advantage of using NFS in your network is that you can share file data across many systems. Access to file data is locked appropriately when many users are accessing data that is available in an NFS environment.

Here are the benefits of using iSCSI targets and initiators in Oracle Solaris:

- The iSCSI protocol runs across existing Ethernet networks.
  - You can use any supported network interface card (NIC), Ethernet hub, or Ethernet switch.
  - One IP port can handle multiple iSCSI target devices.
  - You can use existing infrastructure and management tools for IP networks.
- You might have existing Fibre-Channel devices that can be connected to clients without the cost of Fibre-Channel HBAs. In addition, systems with dedicated arrays can now export replicated storage with Oracle Solaris ZFS or UFS file systems.
- The protocol can be used to connect to Fibre Channel or iSCSI Storage Area Network (SAN) environments with the appropriate hardware.

Here are the current limitations or restrictions of using the iSCSI initiator software in Oracle Solaris:

- Support for iSCSI devices that use SLP is not currently available.
- iSCSI targets cannot be configured as dump devices.
- Transferring large amounts of data over your existing network can have an impact on performance.
COMSTAR Software and Hardware Requirements

- Oracle Solaris storage software and devices
- The `group/feature/storage-server` software package for the system that provides the storage devices
- Any supported NIC

Configuring COMSTAR (Task Map)

<table>
<thead>
<tr>
<th>Task</th>
<th>Description</th>
<th>For Instructions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Identify the COMSTAR software and hardware requirements.</td>
<td>Identify the software and hardware requirements for setting up an iSCSI storage network with COMSTAR.</td>
<td>&quot;COMSTAR Software and Hardware Requirements&quot; on page 243</td>
</tr>
<tr>
<td>Determine the iSCSI target discovery method.</td>
<td>Determine the iSCSI target discovery method best suited for your environment.</td>
<td>&quot;Configuring Dynamic or Static Target Discovery&quot; on page 246</td>
</tr>
<tr>
<td>Enable the STMF service.</td>
<td>Enable the STMF service, which provides persistent target information.</td>
<td></td>
</tr>
<tr>
<td>Create SCSI logical units and make them available.</td>
<td>Create SCSI logical units (LUNs) and make them available to all hosts or specific hosts for iSCSI or iSER configurations.</td>
<td>&quot;How to Create an iSCSI LUN&quot; on page 248</td>
</tr>
<tr>
<td>Configure the iSCSI target.</td>
<td>Configure the iSCSI target for the iSCSI storage component.</td>
<td>&quot;How to Configure the iSCSI Target&quot; on page 250</td>
</tr>
<tr>
<td>Configure the iSCSI initiator.</td>
<td>Configure the system or systems that initiate SCSI requests to the iSCSI target.</td>
<td>&quot;How to Configure an iSCSI Initiator&quot; on page 251</td>
</tr>
<tr>
<td>Access the iSCSI disks.</td>
<td>You can access your iSCSI disks with the <code>format</code> utility. You can also enable the iSCSI disks to be available automatically after the system is rebooted.</td>
<td>&quot;How to Access iSCSI Disks&quot; on page 256</td>
</tr>
<tr>
<td>Restrict LUN access to selected Systems.</td>
<td>You might want to restrict LUN access to specific systems in the network.</td>
<td>&quot;How to Restrict LUN Access to Selected Systems&quot; on page 257</td>
</tr>
</tbody>
</table>
### Task Description for Instructions

<table>
<thead>
<tr>
<th>Task</th>
<th>Description</th>
<th>For Instructions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Configure Fibre Channel devices.</td>
<td>Configure FC devices with COMSTAR if you have a FC storage array in your environment.</td>
<td>&quot;Configuring Fibre Channel Devices With COMSTAR&quot; on page 259</td>
</tr>
<tr>
<td>Configure FCoE devices.</td>
<td>Configure Fibre Channel over Ethernet (FCoE) devices with COMSTAR.</td>
<td>&quot;Configuring FCoE Devices With COMSTAR&quot; on page 263</td>
</tr>
<tr>
<td>Configure SRP devices.</td>
<td>Configure SRP devices with COMSTAR. The SRP (SCSI RDMA Protocol) accelerates the SCSI protocol by mapping the SCSI data transfer phases to Infiniband (IB) Remote Direct Memory Access (RDMA) operations.</td>
<td>&quot;Configuring SRP Devices With COMSTAR&quot; on page 266</td>
</tr>
<tr>
<td>(Optional) Set up authentication in your Oracle Solaris iSCSI configuration.</td>
<td>Decide whether you want to use authentication in your Oracle Solaris iSCSI configuration: Consider using unidirectional CHAP or bidirectional CHAP. Consider using a third-party RADIUS server to simplify CHAP management.</td>
<td>&quot;How to Configure CHAP Authentication for Your iSCSI Initiator&quot; on page 268</td>
</tr>
<tr>
<td></td>
<td></td>
<td>&quot;How to Configure CHAP Authentication for Your iSCSI Target&quot; on page 270</td>
</tr>
<tr>
<td></td>
<td></td>
<td>&quot;How to Configure a RADIUS Server for Your iSCSI Target&quot; on page 271</td>
</tr>
<tr>
<td>Monitor your iSCSI configuration.</td>
<td>Monitor your iSCSI configuration by using the <code>iscsiadm</code> command.</td>
<td>&quot;How to Display iSCSI Configuration Information&quot; on page 276</td>
</tr>
<tr>
<td>(Optional) Modify your iSCSI configuration.</td>
<td>You might want to modify your iSCSI target parameters such as the header and data digest parameters.</td>
<td>&quot;How to Modify iSCSI Initiator and Target Parameters&quot; on page 280</td>
</tr>
</tbody>
</table>
Configuring COMSTAR

Configuring your iSCSI targets and initiators with COMSTAR involves the following tasks:

- Identifying hardware and software requirements
- Configuring your IP network
- Connecting and setting up the iSCSI target device
- Configuring the initiators
- Configuring the iSCSI target discovery method
- Creating file systems on your iSCSI disks
- (Optional) Configuring iSCSI authentication between the iSCSI initiator and the iSCSI target
- Monitoring your iSCSI configuration

The iSCSI configuration information is stored in the /etc/iscsi directory, but it requires no manual administration.

COMSTAR Terminology

Review the following terminology before configuring iSCSI targets and initiators.

<table>
<thead>
<tr>
<th>Term</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>discovery</td>
<td>The process that presents the initiator with a list of available targets.</td>
</tr>
<tr>
<td>discovery method</td>
<td>The way in which the iSCSI targets can be found. Three methods are currently available:</td>
</tr>
<tr>
<td></td>
<td>- Internet Storage Name Service (iSNS) – Potential targets are discovered by interacting with one or more iSNS servers.</td>
</tr>
<tr>
<td></td>
<td>- SendTargets – Potential targets are discovered by using a discovery-address.</td>
</tr>
<tr>
<td></td>
<td>- Static – Static target addressing is configured.</td>
</tr>
<tr>
<td>initiator</td>
<td>The driver that initiates SCSI requests to the iSCSI target.</td>
</tr>
<tr>
<td>initiator group</td>
<td>A set of initiators. When an initiator group is associated with a LUN, only initiators from that group may access the LUN.</td>
</tr>
</tbody>
</table>
Term | Description
--- | ---
1qn or eui address format | An 1qn (iSCSI qualified name) address is the unique identifier for a device in an iSCSI network using the form `iqn.date.authority:uniqueid`. An iSCSI initiator or target is assigned an IQN name automatically when the iSCSI initiator or target is initialized.

An eui (extended unique identifier) address consists of 16 hexadecimal digits, and identifies a class of GUIDs that is used in both the SCSI and InfiniBand standards. SRP devices use the eui address format.

logical unit | A uniquely numbered component in a storage system. Also referred to as a LUN. When a LUN is associated with one or more SCSI targets, the target can be accessed by one or more SCSI initiators.

target device | The iSCSI storage component.

target group | A set of targets. A LUN can be made available to all targets in one target group.

target portal group | A list of IP addresses that determines which interfaces a specific iSCSI target will listen to. A TPG contains IP addresses and TCP port numbers.

**Configuring Dynamic or Static Target Discovery**

Determine whether you want to configure one of the dynamic device discovery methods or use static iSCSI initiator targets to perform device discovery.

- **Dynamic device discovery** – Two dynamic device discovery methods are available:
  - SendTargets – If an iSCSI node exposes a large number of targets, such as an iSCSI to Fibre-Channel bridge, you can supply the iSCSI node IP address/port combination and allow the iSCSI initiator to use the SendTargets features to perform the device discovery.
  - iSNS – The Internet Storage Name Service (iSNS) allows the iSCSI initiator to discover the targets to which it has access using as little configuration information as possible. It also provides state change notification to notify the iSCSI initiator when changes in the operational state of storage nodes occur. To use the iSNS discovery method, you can supply the iSNS server address/port combination and allow the iSCSI initiator to query the iSNS servers that you specified to perform the device discovery. The default port for the iSNS server is 3205. For more information about iSNS, see RFC 4171:


  The iSNS discovery service provides an administrative model to discover all targets on a network.

  For more information about setting up iSNS support in Oracle Solaris, see Chapter 15, “Configuring and Managing the Oracle Solaris Internet Storage Name Service (iSNS).”
- **Static device discovery** – If an iSCSI node has few targets or if you want to restrict the targets that the initiator attempts to access, you can statically configure the target-name by using the following static target address naming convention:

```
target,target-address[:port-number]
```

You can determine the static target address from the array’s management tool.

---

**Note** – Do not configure an iSCSI target to be discovered by both static and dynamic device discovery methods. The consequence of using redundant discovery methods might be slow performance when the initiator is communicating with the iSCSI target device.

### Configuring iSCSI Devices With COMSTAR

You can set up and configure a COMSTAR Internet SCSI (iSCSI) target and make it available over the network. The iSCSI features can work over a normal Internet connection (such as Ethernet) using the standard iSCSI protocol. The iSCSI protocol also provides naming and discovery services, authentication services using CHAP and RADIUS, and centralized management through iSNS.

If the system has at least one InfiniBand (IB) Host Channel Adapter (HCA) and connectivity exists between the initiator and the target, the iSCSI connection uses iSCSI Extensions for RDMA (iSER) for enhanced data throughput. iSER used with an IB HCA provides high bandwidth, low CPU utilization, and a single network connection that multiple protocols can share.

The iSER target and initiator both use a component called iSCSI Data Mover to provide iSCSI connection services. Use of iSER is automatic whenever both the initiator and the target are configured to use IP addresses that correspond to IB-capable devices.

### How to Enable the STMF Service

COMSTAR uses SMF to store its current, persistent configuration, such as logical unit mapping, host group definitions, and target group definitions. When the service is enabled during boot or when using the `svcadm` command, it clears any stale configuration data inside the kernel framework, and then reloads the configuration from the SMF repository into the driver. After the configuration is loaded, any changes that are made to the configuration are automatically updated inside the driver database, as well as inside the SMF repository. For example, any changes made through the `svr最大的` command are automatically updated in both areas.

The COMSTAR target mode framework runs as the `stmf` service. By default, the service is disabled. You must enable the service to use COMSTAR functionality. You can identify the service with the `svcs` command. If you have not rebooted the server since installing the `group/feature/storage-server` package, the service might not be enabled correctly.
1 **Install the COMSTAR storage server software.**

```
 target# pkg install group/feature/storage-server
 Packages to install: 75
 Create boot environment: No
 Services to restart: 7
```

<table>
<thead>
<tr>
<th>PHASE</th>
<th>ACTIONS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Install Phase</td>
<td>13347/13347</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>PHASE</th>
<th>ITEMS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Package State Update Phase</td>
<td>75/75</td>
</tr>
<tr>
<td>Image State Update Phase</td>
<td>2/2</td>
</tr>
<tr>
<td>Loading smf(5) service descriptions:</td>
<td>17/17</td>
</tr>
<tr>
<td>Loading smf(5) service descriptions:</td>
<td>3/3</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>PHASE</th>
<th>ITEMS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Reading Existing Index</td>
<td>8/8</td>
</tr>
<tr>
<td>Indexing Packages</td>
<td>75/75</td>
</tr>
<tr>
<td>Indexing Packages</td>
<td>75/75</td>
</tr>
<tr>
<td>Optimizing Index...</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>PHASE</th>
<th>ITEMS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Indexing Packages</td>
<td>573/573</td>
</tr>
</tbody>
</table>

2 **Either reboot the system or enable the stmf service.**

```
 target# svcadm enable stmf
 # svcsm stmf
```

```
 STATE STIME FMRI
 online 09:42:32 svc:/system/stmf:default
```

### How to Back Up and Restore a COMSTAR Configuration

After you complete your COMSTAR configuration, make a copy that can be restored, if needed.

1 **Become an administrator.**

2 **Export the current COMSTAR configuration.**

```
 # svcsm export -a stmf > COMSTAR.backup
```

3 **If necessary, restore the exported configuration.**

```
 # svcsm import COMSTAR.backup
```

#### How to Create an iSCSI LUN

The logical unit provider for creating disk-type LUNs is called std. However, you must initialize the storage for the logical unit before you can share a disk-type LUN.
The disk volume provided by the server is referred to as the target. When the LUN is associated with an iSCSI target, it can be accessed by an iSCSI initiator.

The process for creating SCSI LUNs is as follows:

- Initialize the storage for the LUN, also known as the backing store.
- Create a SCSI LUN by using the backing store.

When a LUN is created, it is assigned a global unique identifier (GUID), for example, 600144F0B518B0000004DDAC7C10001. The GUID is used to refer to the LUN in subsequent tasks, such as mapping a LUN to select hosts.

The following steps are completed on the system that is providing the storage device.

1. **Create a ZFS storage pool.**
   
   ```
   target# zpool create sanpool mirror c2t3d0 c2t4d0
   ```

2. **Create a ZFS volume to be used as a SCSI LUN.**
   
   ```
   target# zfs create -V 2g sanpool/vol1
   ```

3. **Create a LUN for the ZFS volume.**
   
   ```
   target# stmfdm create-lu /dev/zvol/rdsk/sanpool/vol1
   Logical unit created: 600144F0B518B0000004DDAC7C10001
   ```
   You can find the device path for the ZFS volume in the /dev/zvol/rdsk/pool-name/ directory.

4. **Confirm that the LUN has been created.**
   
   ```
   target# stmfdm list-lu
   LU Name: 600144F0B518B0000004DDAC7C10001
   ```

5. **Add the LUN view.**
   
   This command makes the LUN accessible to all systems.
   
   ```
   target# stmfdm add-view 600144F0B518B0000004DDAC7C10001
   ```
   If you want to restrict the LUN view to specific systems, see “How to Restrict LUN Access to Selected Systems” on page 257.

6. **Verify the LUN configuration.**
   
   ```
   target# stmfdm list-view -l 600144F0B518B0000004DDAC7C10001
   View Entry: 0
   Host group : All
   Target group : All
   LUN : 0
   ```
How to Create the iSCSI Target

This procedure assumes that you are logged in to the local system will contain the iSCSI target.

1. **Enable the iSCSI target service.**

   ```
   target# svcadm enable -r svc:/network/iscsi/target:default
   ```

   Confirm that the service is enabled.

   ```
   target# svcs -l iscsi/target
   fmri svc:/network/iscsi/target:default
   name iscsi target
   enabled true
   state online
   next_state none
   state_time Mon May 23 14:48:59 2011
   logfile /var/svc/log/network-iscsi-target:default.log
   restarter svc:/system/svc/restarter:default
   dependency require_any/error svc:/milestone/network (online)
   dependency require_all/none svc:/system/stmf:default (online)
   ```

2. **Create the iSCSI target.**

   ```
   target# itadm create-target
   Target iqn.1986-03.com.sun:02:73d12edc-9bb9-cb44-efc4-c3b36c039405 successfully created
   ```

3. **Display the iSCSI target information.**

   ```
   target# itadm list-target -v
   TARGET NAME                      STATE   SESSIONS
   iqn.1986-03.com.sun:02:73d12edc-9bb9-cb44-efc4-c3b36c039405 online 0
   alias: -
   auth: none (defaults)
   targetchapuser: -
   targetchapsecret: unset
   tpg-tags: default
   ```

How to Configure an IB HCA for iSER

An InfiniBand (IB) Host Channel Adapter (HCA) is required to take full advantage of the iSCSI Extensions for RDMA (iSER) capabilities. To use iSER, you must configure the HCA on both the target and the initiator.

1. **Become an administrator on the host (initiator) system.**

2. **Connect the HCA to an IB switch.**

   See the vendor documentation for details.
3 Configure the target and the initiator for the HCA.
The target and the initiator must be on the same subnet. This example uses ibd0 as the driver.
```
# ipadm create-addr ibd0
```

4 Configure the IP address and port combination for the HCA.
```
# ipadm create-addr -T static -a local=10.1.190.141/24 ibd0/v4addr
```

5 Verify the interface configuration.
```
# ipadm show-addr
```
```
ADDROBJ TYPE STATE ADDR
lo0/v4 static ok 127.0.0.1/8
e1000g0/_b dhcp ok 10.1.190.141/24
lo0/v6 static ok ::1/128
e1000g0/_a addrconf ok fe80::214:4fff:fe27:360c/10
```

6 Become an administrator on the target system and repeat steps 3-5 for all other HCA hosts on the network.

7 Verify connectivity on both the target and the initiator.
```
target# ping initiator-ip
initiator# ping target-ip
```

▼ How to Configure an iSCSI Initiator

Part of the initiator configuration process is to identify the iSCSI target discovery method, which presents an initiator with a list of available targets. You can configure iSCSI targets for static, SendTargets, or iSNS dynamic discovery. Dynamic discovery using the SendTargets option is the optimum configuration for an iSCSI initiator that accesses a large number of targets, such over an iSCSI to Fibre Channel bridge. SendTargets dynamic discovery requires the IP address and port combination of the iSCSI target for the iSCSI initiator to perform the target discovery. The most common discovery method is SendTargets.

When configuring the target discovery method, you must provide the following information, depending on which method you choose:

- SendTargets – Target IP address
- iSNS – iSNS server address
- Static – Target IP address and target name

For more information about configuring target discovery methods, see “Configuring Dynamic or Static Target Discovery” on page 246.

1 Enable the iSCSI initiator service.
```
initiator# svcadm enable network/iscsi/initiator
```
2 Verify the target’s name and IP address while logged in to the server that is providing the target.

```
target# ipadm show-addr
ADDROBJ   TYPE   STATE   ADDR
lo0/v4    static ok  127.0.0.1/8
ether0/g0/1 dhcp ok  10.80.227.189/24
lo0/v6    static ok  ::1/128
ether0/g0/2 addrconf ok fe80::1a:b9ff:fe4b:18a/10
```

3 Configure the target to be statically discovered.

```
initiator# iscsiadm add static-config iqn.1986-03.com.sun:02:73d12edc-9bb9-cb44-efc4-c3b36c039405, 10.80.227.189
```

4 Review the static configuration information.

```
initiator# iscsiadm list static-config
 alias: -
 auth: none (defaults)
 targetchapuser: -
 targetchapsecret: unset
 tpg-tags: default
```

The iSCSI connection is not initiated until the discovery method is enabled. See the next step.

5 Configure one of the following target discovery methods:

- If you have configured a dynamically discovered (SendTargets) target, configure the SendTargets discovery method.
  ```
  initiator# iscsiadm add discovery-address 10.80.227.189
  ```

- If you have configured a dynamically discovered (iSNS) target, configure the iSNS discovery method.
  ```
  initiator# iscsiadm add isns-server 10.80.227.189
  ```

6 Enable one of the following the target discovery methods:

- If you have configured a dynamically discovered (SendTargets) target, enable the SendTargets discovery method.
  ```
  initiator# iscsiadm modify discovery --sendtargets enable
  ```

- If you have configured a dynamically discovered (iSNS) target, enable the iSNS discovery method.
  ```
  initiator# iscsiadm modify discovery --iSNS enable
  ```

- If you have configured static targets, enable the static target discovery method.
  ```
  initiator# iscsiadm modify discovery --static enable
  ```
7 Reconfigure the /dev namespace to recognize the iSCSI disk, if necessary.
   
   initiator# devfsadm -i iscsi

▼ How to Remove Discovered iSCSI Targets

The associated targets are logged out after you perform any of the following actions:

- Remove a discovery address
- Remove an iSNS server
- Remove a static configuration
- Disable a discovery method

If these associated targets are still in use, for example, they have mounted file systems, the logout of these devices will fail, and they will remain on the active target list.

This optional procedure assumes that you are logged in to the local system where access to an iSCSI target device has already been configured.

1 Become an administrator.

2 (Optional) Disable an iSCSI target discovery method by using one of the following:

- If you need to disable the SendTargets discovery method, use the following command:
  
  initiator# iscsiadm modify discovery --sendtargets disable

- If you need to disable the iSNS discovery method, use the following command:

  initiator# iscsiadm modify discovery --iSNS disable

- If you need to disable the static target discovery method, use the following command:

  initiator# iscsiadm modify discovery --static disable

3 Remove an iSCSI device discovery entry by using one of the following:

- Remove an iSCSI SendTargets discovery entry.
  
  For example:

  initiator# iscsiadm remove discovery-address 10.0.0.1:3260

- Remove an iSCSI iSNS discovery entry.
  
  For example:

  # iscsiadm remove isns-server 10.0.0.1:3205

- Remove a static iSCSI discovery entry.
  
  For example:

  initiator# iscsiadm remove static-config eui.5000ABCD78945E2B,10.0.0.1
Note – If you attempt to disable or remove a discovery entry that has an associated logical unit (LUN) in use, the disable or remove operation fails with the following message:

**logical unit in use**

If this error occurs, stop all associated I/O on the LUN, unmount the file systems, and so on. Then, repeat the disable or remove operation.

4 Remove the iSCSI target device.

For example:

```
target# itadm delete-target target-IQN
```

This command might error if the target is still online or busy:

```
The target is online or busy. Use the -f (force) option, or 'stmfadm offline-targetiqn.1986-03.com.sun:02:99619b8a-a4dc-4cfb-93f0-ee3debe7b0c8'itadm delete-target failed with error 16
```

**Creating iSCSI Target Portal Groups**

You can create a target portal group (TPG) to manage the discovery of multiple iSCSI and iSER targets. A TPG is a list of IP addresses to determine upon which interfaces a specific iSCSI target will listen.

A TPG contains IP addresses and TCP port numbers. To use this capability, you need to do the following:

- Create a TPG as a list of `ip-address:port` specifiers by using the `itadm create-tpg` command.
- Bind a specific iSCSI target to a TPG by using the `itadm modify-target -t` command.
- When an iSCSI target is made active, an iSCSI listener is created for each IP address and port belonging to a TPG associated with that target.

A TPG is an efficient way to control which targets are discovered through specific ports. For example, you could restrict your iSCSI target so that it is available only through one specific IP address or only through a set of iSER-capable IP addresses.

Note – Do not confuse target portal groups with target groups. A target group is a list of SCSI target ports that are all treated the same when creating views. Creating a view can help you facilitate LUN mapping. Each view entry specifies a target group, as host group, and a LUN. For more information on Target Groups and LUN mapping, see “Making SCSI Logical Units Available” on page 256 and `stmfadm(1M)`.
To learn about static and iSNs target discovery, see “Configuring Dynamic or Static Target Discovery” on page 246. The iSCSI initiator uses the `iscsiadm` command to discover TPGs. For more information, see `iscsiadm(1M)` and `itadm(1M)`.

**Using TPGs with iSER**

When you use the SendTargets discovery and iSER at the same time, a common convention is to use a TPG to associate a specific iSCSI target port with only iSER-capable IP addresses. For example, if a target system has four IP addresses, A, B, C, and D, and only addresses B and C are iSER-capable, then addresses B and C could be added to a TPG, and assigned to a target T.

An iSCSI initiator with both Ethernet and InfiniBand (IB) interfaces could use the SendTargets discovery method to discover the possible storage targets. Without the use of TPGs, the initiator might always prefer the use of the Ethernet interfaces over the IB interfaces. By associating target T only with the IB interfaces, the initiator correctly prefers using its IB-capable interface when connecting to target T.

**How to Create a Target Portal Group for iSCSI Targets**

You can create a target portal group (TPG) by providing a unique name, and a TPG Tag (ranging from 2–65535) is automatically generated. TPG Tag 1 is reserved for the default TPG that is used when you do not explicitly set a TPG on the target. The portal for the default TPG matches requests from all network interfaces on port 3260.

The following steps shows how to create two TPGs, TPGA and TPGB, that use port 8000 for the IP addresses in TPGB.

1. **Become an administrator.**

2. **Create two TPGs.**

   ```bash
   target# itadm create-tpg TPGA 192.168.0.1 192.168.0.2
   target# itadm create-tpg TPGB 192.168.0.2:8000 192.168.0.2:8000
   ```

   **Note** – IPv4 portals are specified in dotted address notation (for example, 192.168.0.1). IPv6 portal addresses must be enclosed in square brackets.

3. **Configure an existing iSCSI target to use the TPGs, TPGA and TPGB.**

   ```bash
   # itadm modify-target -t TPGA,TPGB eui.20387ab8943ef7548
   ```

4. **Verify the TPGs that you created.**

   ```bash
   # itadm list-tpg -v
   ```

   You can remove a TPG with the `itadm delete-tpg` command.
How to Access iSCSI Disks

After the devices have been discovered by the Oracle Solaris iSCSI initiator, the login negotiation occurs automatically. The Oracle Solaris iSCSI driver determines the number of available LUNs and creates the device nodes. Then, the iSCSI devices can be treated as any other SCSI device.

You can create a ZFS storage pool on the LUN and then create a ZFS file system.

You can view the iSCSI disks on the local system by using the format utility.

1. Review the iSCSI LUN information in the format output.

   initiator# format
   0. c0t600144f0b5418b00000040dac7c10001d0 <SUN-COMSTAR-1.0 cyl 1022 alt 2 hd 128 sec 32>
   /scsi_vhci/disk@g600144f0b5418b00000040d0ac7c10001
   1. c8t0d0 <Sun-STK RAID INT-V1.0 cyl 17830 alt 2 hd 255 sec 63>
   /pci@0,0/pci10de,375/f/pci108e,280@/disk@0,0
   2. c8t1d0 <Sun-STK RAID INT-V1.0 cyl 17830 alt 2 hd 255 sec 63>
   /pci@0,0/pci10de,375/f/pci108e,280@/disk@0,0
   3. c8t2d0 <Sun-STK RAID INT-V1.0 cyl 17830 alt 2 hd 255 sec 63>
   /pci@0,0/pci10de,375/f/pci108e,280@/disk@0,0
   4. c8t3d0 <Sun-STK RAID INT-V1.0 cyl 17830 alt 2 hd 255 sec 63>
   /pci@0,0/pci10de,375/f/pci108e,280@/disk@0,0

   Specify disk (enter its number): 0
   selecting c0t600144f0b5418b00000040dac7c10001d0
   [disk formatted]

   In the above output, disk 0 is an iSCSI LUN under MPxIO control. Disks 1-4 are local disks.

2. You can create a ZFS storage pool and ZFS file systems on the iSCSI LUN.

   initiator# zpool create pool-name c0t600144f0b5418b00000040dac7c10001d0
   initiator# zfs create pool-name/fs-name

   The ZFS file system is automatically mounted when created and is remounted at boot time.

Making SCSI Logical Units Available

- “How to Make a Logical Unit Available to All Systems” on page 257
- “How to Restrict LUN Access to Selected Systems” on page 257

Simply registering a logical unit (LUN) with the STMF framework does not make it available to hosts (initiators) on the network. This section describes how to make LUNs visible to initiator hosts for the following configurations.

For iSCSI, Fibre Channel, and FCoE configurations, a LUN must be mapped before it can be accessed. You can choose one of the following methods, both of which use the stmfdm command:

- **Simple mapping** – Exposes the LUN to all initiators through all the ports, using one command. Making LUNs available to all hosts uses this method.
Selective mapping—Enables you to specify the hosts that can access the LUN. Making LUNs available to selected hosts uses this method. This process includes the following steps:

1. Defining host groups—A host group is a name given to a set of hosts (initiators) that are allowed to access the same LUNs. This step is not needed if the same set of LUNs is visible to all the hosts, as in simple mapping.

2. Defining target groups—A target group is a name given to a set of target ports that export the same set of LUNs to the same set of host groups. This step is not needed if the same set of LUNs is visible to all the hosts, as in simple mapping.

3. Adding one or more views for each logical unit—Adding a view creates a bridge between the LUN and the host initiator. When an initiator from the host group logs in to a target port contained in the target group, the LUN is visible.

Note—Don’t confuse a target group with a target portal group (TPG). A TPG is a list of IP addresses that an iSCSI target listens to. A TPG can help you restrict an iSCSI target so that it is available only through one specific IP address. For more information on target groups, see stmfadm(1M).

A view entry consists of four components: host group, target group, logical unit (LUN), and LUN identifier. Of these four components, only the LUN identifier is required. If the other components are omitted, the following default values are assumed:

- If the host group is omitted, the all initiators value is assumed.
- If the target group is omitted, the all targets value is assumed.
- If the LUN is omitted, the system chooses a suitable LUN for the entry.

How to Make a Logical Unit Available to All Systems

This procedure makes a LUN available to all initiator hosts on a storage network.

1. Obtain the Global Unique Identification (GUID) number for the LUN.
   
   # stmfadm list-lu -v

2. Add a view for the logical unit.

   3 # stmfadm add-view GUID-number

How to Restrict LUN Access to Selected Systems

Use this procedure to restrict LUNs to selected hosts on a storage network. If you are using Fibre Channel ports, first identify the World Wide Names (WWN). Then, selectively map a logical unit number (LUN) to the ports on host-a, for example. A target group (targets -0) is also defined for a given set of target ports that export the same set of LUNs to the same host group.
For information about configuring a target group, see \texttt{stmfadm(1M)}.

1. \textbf{Become an administrator.}

2. \textbf{Identify the Fibre Channel (FC) port on the initiator.}

   \begin{verbatim}
   initiator# fcinfo hba-port
   HBA Port WWN: 210000e08b195dae
   Port Mode: Initiator
   Port ID: 0
   OS Device Name: /dev/cfg/c8
   Manufacturer: QLogic Corp.
   Model: 375-3108-xx
   Firmware Version: 03.03.28
   FCode/BIOS Version: fcode: 1.13;
   Serial Number: not available
   Driver Name: qlc
   Driver Version: 20100408-3.01
   Type: unknown
   State: offline
   Supported Speeds: 1Gb 2Gb
   Current Speed: not established
   Node WWN: 200000e08b195dae
   NPIV Not Supported
   \end{verbatim}

3. \textbf{Create a host group.}

   \begin{verbatim}
   target# stmfadm create-hg host-a 210000e08b195dae 210100e08b395dae
   \end{verbatim}

4. \textbf{Add the WWNs identified in the preceding output as members of the host group.}

   \begin{verbatim}
   target# stmfadm add-hg-member -g host-a
   \end{verbatim}

5. \textbf{Create a target group.}

   \begin{verbatim}
   target# stmfadm create-tg targets-0
   \end{verbatim}

6. \textbf{Specify the target group members by adding the target names.}

   Each SCSI target can be a member of only one target group.

   \begin{verbatim}
   target# stmfadm add-tg-member -g targets-0 wwn.guide-number
   \end{verbatim}

7. \textbf{Identify the GUID number for the LUN.}

   \begin{verbatim}
   target# stmfadm list-lu -v
   \end{verbatim}

8. \textbf{Make the LUN available by adding a view entry, specifying the host group name and the LUN GUID number.}

   \begin{verbatim}
   target# stmfadm add-view -h host-a -t targets-0 -n 1 guide-number
   \end{verbatim}
Configuring Fibre Channel Devices With COMSTAR

You can set up and configure a COMSTAR target on a SPARC system or x86 system in a FC network environment and make it accessible to the storage network. Ensure the following prerequisite step has been completed:

- “How to Create an iSCSI LUN” on page 248

Configuring Fibre Channel Ports For COMSTAR

The Fibre Channel (FC) port provider can use the same HBAs that are used for the FC initiators. A given FC port can be used as an initiator or as a target, but not as both. You can also configure a dual port or quad port FC HBA so that a subset of the HBA ports is in target mode and the rest of the ports are in initiator mode.

The procedures in this section are specific to QLogic HBAs. The driver to use with HBAs in initiator mode is qlc, which is a Qlogic driver. The qlc driver works only in initiator mode, and cannot be used for target mode. The COMSTAR driver for a target QLogic 4G HBA is qlt.

Because initiator mode and target mode use different drivers, the driver you attach to an HBA port defines its function as a target or initiator. You can specify a driver for all the ports by specifying the PCI device ID of the HBA. Or, you can configure the driver on a port-by-port basis. Both methods use the `update_drv` command and are described in this section. For more information, see `update_drv(1M)`.

▼ How to Display Existing FC Port Bindings

Before making changes to the HBA ports, first check the existing port bindings.

1 Become an administrator.

2 Display what is currently bound to the port drivers.

In this example, the current binding is `pciex1077,2432`.

```
# mdb -k
Loading modules: [ unix krtld genunix specs ... 
> :/dev/bindings -q qlc
30001617a08 pciex1077,2432, instance #0 (driver name: qlc)
30001617e0 pciex1077,2432, instance #1 (driver name: qlc)
> 5q
```

▼ How to Set All FC Ports to a Specific Mode

This procedure changes all ports on all HBAs with a specific PCI device ID to target mode. The PCI device ID number binds a driver to a port, thereby setting all the HBA ports with that PCI device ID, for example, all QLogic 4G PCI express HBAs, to target mode.

1 Become an administrator.
2  Remove the current binding.
   In this example, the qlc driver is actively bound to pciex1077,2432. You must remove the
   existing binding for qlc before you can add that binding to a new driver. Single quotation
   marks are required in this syntax.
   
   ![](image)

   This message does not indicate an error. The configuration files have been updated but the qlc
   driver remains bound to the port until the system is rebooted.

3  Establish the new binding.
   In this example, qlt is updated. Single quotes are required in this syntax.
   
   ![](image)

   This message does not indicate an error. The qlc driver remains bound to the port, until the
   system is rebooted. The qlt driver attaches when the system is rebooted.

4  Reboot the system to attach the new driver. Then, recheck the bindings.
   
   ![](image)

5  Verify that the target mode framework has access to the HBA ports.
   
   ![](image)

   ▼ How to Set Selected FC Ports to Initiator or Target Mode
   This procedure uses path-based bindings. It shows you how to use a specific device path to bind
   a port to a driver that is different from the driver to which it is currently bound.
1 Become an administrator.

2 Display a list of the HBA ports and their respective device paths.
   This example shows the device paths for a single HBA with two ports.

   # luxadm -e port
   /devices/pci@780/QLGC,qlc@0,1/fp@0,0:devctl CONNECTED
   /devices/pci@780/QLGC,qlc@0/fp@0,0:devctl CONNECTED

3 Set the top port to target mode, and leave the bottom port in initiator mode.
   Remove the initial /devices portion of the path, and include everything up to /fp@0.... The
   path with the /devices portion removed is the path to which the system binds the qlt driver.

   Single quotation marks are required in this syntax.

   # update_drv -a -i '/pci@780/QLGC,qlc@0,1' qlt
   devfsadm: driver failed to attach: qlt
   Warning: Driver (qlt) successfully added to system but failed to attach.
   This message does not indicate an error. The qlc driver remains bound to the port until reboot.
   The qlt driver attaches during reboot.

4 Reboot the system to attach the new driver. Then, recheck the bindings.
   You should see that the port changed from initiator mode (qlc) to target mode (qlt).

   # init 6
   ...
   ...
   # mdb -k
   ...
   ...
   > $q

5 Verify that the target mode framework has access to the HBA ports.

   # stmfadm list-target -v

Making Logical Units Available for FC and FCoE

Simply registering a logical unit (LUN) with the STMF framework does not make it available to
hosts (initiators) on the network. You must make logical units visible to initiator hosts for Fibre
Channel and FCoE configurations by mapping the logical unit. To determine which method to
use and how to map the logical unit, see "How to Make a Logical Unit Available to All Systems"
on page 257. Both methods use the stmfadm command. The additional steps below are for FC
and FCoE configurations.
How to Make Logical Units Available for FC and FCoE

This procedure makes the LUN available to all hosts or selected hosts for FC or FCoE configurations on a storage network. The steps are run on the host.

1 Become an administrator.

2 Make a LUN available to hosts.

Obtain the global unique identification (GUID) number for the LUN.

```
# sbdadm list-lu
# stmfdm list-lu -v
```

Identify the WWNs for the FC or FCoE ports of the host.

```
# fcinfo hba-port
HBA Port WN: *210000e08b83378d*
OS Device Name: /dev/cfg/c4
Manufacturer: QLogic Corp.
Model: QLA2462
Firmware Version: 4.0.27
Fcode/BIOS Version: N/A
Type: N-port
State: online
Supported Speeds: 1Gb 2Gb 4Gb
Current Speed: 4Gb
Node WN: 210000e08b83378d
HBA Port WN: *210100e08ba3378d*
OS Device Name: /dev/cfg/c5
Manufacturer: QLogic Corp.
Model: QLA2462
Firmware Version: 4.0.27
Fcode/BIOS Version: N/A
Type: N-port
State: online
Supported Speeds: 1Gb 2Gb 4Gb
Current Speed: 4Gb
Node WN: 210100e08ba3378d
```

3 Add a view and perform mapping.

Follow the instructions in “How to Make a Logical Unit Available to All Systems” on page 257.

4 Verify that the LUN is visible on an Oracle Solaris initiator host by running the following script.

```
#!/bin/ksh
fcinfo hba-port | grep "HBA" | awk '{print $4}' | while read ln
do      fcinfo remote-port -p $ln -s >/dev/null 2>&1
done
```

All subsequent LUNs appear in the format output, because the script forces the initiator to touch all the LUNs through all the ports. If you do not see the LUNs, run the format command again. If you still do not see the LUNs, ensure that the service is enabled on the target by using the svcs stmfd command. Also ensure that you added view entries for the LUN, as described in “How to Make a Logical Unit Available to All Systems” on page 257.
5 Verify that the LUN is visible on other systems.

- For a Linux initiator host, verify that the LUN is visible by running the utility provided by the HBA vendor. The utility scans for configuration changes.
- For a Windows initiator host, verify that the logical unit is visible by selecting Control Panel → Administrative Tools → Computer Management → Disk Management. Then, from the Action menu, choose Rescan Disks.

**Configuring FCoE Devices With COMSTAR**

You can set up and configure a COMSTAR target in a Fibre Channel over Ethernet (FCoE) network environment, then make it accessible to the storage network. Ensure that your system meets the necessary prerequisites before you begin:

- “How to Create an iSCSI LUN” on page 248
- “How to Create the iSCSI Target” on page 250
- “How to Configure an iSCSI Initiator” on page 251

**Configuring FCoE Ports**

- “Enabling 802.3x PAUSE and Jumbo Frames on the Ethernet Interface” on page 264
- “How to Create FCoE Target Ports” on page 264
- “How to Verify That an FCoE Target Port Is Working” on page 264
- “How to Delete FCoE Target Ports” on page 265

FCoE functionality is provided through Ethernet interfaces. Fibre Channel over Ethernet (FCoE) ports are logical entities associated with Ethernet interfaces. Within an Oracle Solaris system, a one-to-one mapping of FCoE ports and Ethernet interfaces exist. You can associate only one FCoE port with a given Ethernet interface. FCoE and IP cannot share the same Ethernet interface. So, before you create an FCoE port on an Ethernet interface, ensure that the interface has been unplumbed.

The FCoE port configuration is persistent across reboots. All configured FCoE ports are created and placed online automatically after the system reboots.

For FCoE target ports, you must enable the following service before creating FCoE target ports to get persistent behavior.

```
# svcadm enable svc:/system/fcoe_target:default
```
Enabling 802.3x PAUSE and Jumbo Frames on the Ethernet Interface

These settings can vary for different Ethernet hardware and drivers. In most cases, you must modify the `driver.conf` file of the Ethernet interface and then reboot. See the `driver.conf` file for your Ethernet interface for details on how to enable these features.

Perform one of the following prerequisite steps before you create an FCoE port on an Ethernet interface:

- Enable 802.3x (also called PAUSE) on the Ethernet interface.
  Doing so ensures a consistent Ethernet transport.
- Enable jumbo frames (> 2.5 Kbyte) on the Ethernet interface.
  A Fibre Channel data frame can be as large as 2136 bytes.

How to Create FCoE Target Ports

1. Create an FCoE target port on a specified network interface.
   ```
   # fcadm create-fcoe-port -t nxge0
   ```
   If the Ethernet interface you selected does not support multiple unicast address (for example, the VMWare network interface), you are prompted to explicitly enable promiscuous mode on that interface.

2. Enable promiscuous mode, if prompted.
   ```
   # fcadm create-fcoe-port -t -f e1000g0
   ```
   If no error messages appear, the FCoE target port is created, and the FCoE target is online.

How to Verify That an FCoE Target Port Is Working

1. Display the FCoE ports that you created.
   ```
   # fcadm list-fcoe-ports
   HBA Port WWN: 200000144fda7f66
   Port Type: Target
   MAC Name: nxge0
   MTU Size: 9194
   MAC Factory Address: 00144fda7f66
   MAC Current Address: 0efc009a002a
   Promiscuous Mode: On
   ```
2 Display all target mode Fibre Channel HBA ports on the host.

   # fcinfo hba-port -t
   HBA Port WWN: 200000144fda7f66
      Port Mode: Target
      Port ID: 9a002a
      OS Device Name: Not Applicable
      Manufacturer: Sun Microsystems, Inc.
      Model: FCoE Virtual FC HBA
      Firmware Version: N/A
      FCode/BIOS Version: N/A
      Serial Number: N/A
      Driver Name: COMSTAR FCOET
      Driver Version: 1.0
      Type: F-port
      State: online
      Supported Speeds: 1Gb 10 Gb
      Current Speed: 10Gb
      Node WWN: 100000144fda7f66

3 View a list of the FCoE target ports.

   The -v option displays additional information about the target, along with SCSI session
   information for logged-in initiators.

   # stmfadm list-target -v
   Target wwn.200000144FDA7F66
      Operational Status: Online
      Provider Name : fcoet
      Alias : fcoet1
      Sessions : 1
      Initiator: wwn.210000E08B818343
         Alias: QLA2342 FW:v3.03.25 DVR:v8.02.14.01
         Logged in since: ...

▼ How to Delete FCoE Target Ports

You can disable FCoE functionality when needed.

1 Take the FCoE target port offline.

   # stmfadm offline-target wwn.200000144fda7f66

2 Remove the FCoE target port.

   # fcadm delete-fcoe-port nxge0
Configuring SRP Devices With COMSTAR

The SCSI RDMA Protocol (SCSIRDMA) accelerates the SCSI protocol by mapping the SCSI data transfer phases to Infiniband (IB) Remote Direct Memory Access (RDMA) operations. As a result, an SRP initiator can read and write data from a COMSTAR SRP target at high data rates with relatively low CPU utilization.

You can set up and configure a COMSTAR SRP target and make it available over an Infiniband (IB) fabric. The SRP target is available wherever a supported IB Host Channel Adapter (HCA) is installed on the target system.

- **One SCSI target per IB HCA** – The COMSTAR SRP target uses a simple model in which each supported IB HCA is registered as a SCSI target. The SCSI target is a virtual object that contains a task router and acts as the connection between the SCSI transport (in this case, SRP) and the SCSI back end (STMF and SBD).

  An HCA can contain multiple physical ports. The same SCSI target is shared between all the ports of an HCA. The SCSI target representing the HCA is automatically available for incoming connections through all the active ports of that HCA.

- **SRP target eui identifier** – In the IB architecture, each HCA and each port is assigned a 64-bit GUID by the manufacturer. The COMSTAR SCSI target created for each HCA is given a name corresponding to the GUID of that HCA, which has the format: eui.HCA-GUID. For example, if the target system includes a supported IB HCA with an HCA GUID of 0003BA0001002E48, then a SCSI target will be created using the name eui.0003BA0001002E48. The string eui stands for extended unique identifier and names a class of GUIDs that is used in both the SCSI and IB standards.

- **SRP initiator eui identifier** – Similarly, SRP uses a 64-bit initiator GUID to identify the initiator system. The choice of which GUID to use is determined by the SRP initiator implementation. Many initiators use the GUID of the HCA that is being used for the outgoing connection. For example, an initiator using the GUID of 0003BA0001002EA5 is known to COMSTAR as eui.0003BA0001002EA5.

Using COMSTAR Views With SRP

The COMSTAR view facility can be used to create target groups and host groups that restrict and configure which storage logical units (LUNs) can be accessed through each SCSI target or initiator as described in “Making SCSI Logical Units Available” on page 256. The eui identifier of the SRP initiator is added to a host group. The eui identifier of the SRP SCSI Target is added to a target group. The view entries for each LUN then determine the particular set of LUNs that each initiator can access.
How to Enable the SRP Target Service

The COMSTAR port provider for the COMSTAR SRP target is managed by the Service Management Facility (SMF). The primary SRP target service is

svc:/system/ibsrp/target:default, which can be abbreviated to ibsrp/target.

The SRP package is storage/scsi-rdma/scsi-rdma-target.

1 Become an administrator.

2 Recursively enable the SRP target service.

   # svcadm enable -r ibsrp/target

3 Display the SRP target service information.

   # svcs -l ibsrp/target

How to Verify SRP Target Status

1 Become an administrator.

2 Verify the presence of the expected SRP SCSI target on the system.

   # srptadm list-target

   Target HCA 21280001A0D0F0:
     Enabled : true
     SRP Target Name : eui.0021280001A0D0F0
     Operational Status : online

Configuring Authentication in Your iSCSI-Based Storage Network

Setting up authentication for your iSCSI devices is optional.

In a secure environment, authentication is not required because only trusted initiators can access the targets.

In a less secure environment, the target cannot determine if a connection request is truly from a given host. In that case, the target can authenticate an initiator by using the Challenge-Handshake Authentication Protocol (CHAP).

CHAP authentication uses the notion of a challenge and response, which means that the target challenges the initiator to prove its identity. For the challenge/response method to work, the target must know the initiator’s secret key, and the initiator must be set up to respond to a challenge. Refer to the array vendor’s documentation for instructions on setting up the secret key on the array.

Chapter 14 • Configuring Storage Devices With COMSTAR
iSCSI supports unidirectional and bidirectional authentication as follows:

- **Unidirectional authentication** enables the target to authenticate the identity of the initiator. Unidirectional authentication is done on behalf of the target to authenticate the initiator.
- **Bidirectional authentication** adds a second level of security by enabling the initiator to authenticate the identity of the target. Bidirectional authentication is driven from the initiator, which controls whether bidirectional authentication is performed. The only setup required for the target is that the chap user and chap secret must be correctly defined.

#### ▼ How to Configure CHAP Authentication for Your iSCSI Initiator

This procedure assumes that you are logged in to the local system where you want to securely access the configured iSCSI target device.

- The length of the CHAP secret key for the COMSTAR iSCSI target must be a minimum of 12 characters and a maximum of 255 characters. Some initiators support only a shorter maximum length for the secret key.
- Each node identifying itself using CHAP must have both a username and a password. In the Oracle Solaris 11 environment, the CHAP user name is set to the initiator or target node name (that is, the \texttt{iqn} name) by default. The CHAP user name can be set to any length of text that is less than 512 bytes. The 512-byte length limit is an Oracle Solaris 11 limitation. However, if you do not set the CHAP user name, it is set to the node name upon initialization.

You can simplify CHAP secret key management by using a third-party RADIUS server, which acts as a centralized authentication service. When you use RADIUS, the RADIUS server stores the set of node names and matching CHAP secret keys. The system performing the authentication forwards the node name of the requester and the supplied secret of the requester to the RADIUS server. The RADIUS server confirms whether the secret key is the appropriate key to authenticate the given node name. Both iSCSI and iSER support the use of a RADIUS server.

For more information about using a third-party RADIUS server, see “Using a Third-Party RADIUS Server to Simplify CHAP Management in Your iSCSI Configuration” on page 270.

1. **Become an administrator.**
   
   For more information, see “How to Use Your Assigned Administrative Rights” in *Oracle Solaris Administration: Security Services*.

2. **Determine whether you want to configure unidirectional or bidirectional CHAP.**

   - Unidirectional authentication, the default method, enables the target to validate the initiator. Complete steps 3–5 only.
Bidirectional authentication adds a second level of security by enabling the initiator to authenticate the target. Complete steps 3–9.

3 **Unidirectional CHAP: Set the secret key on the initiator.**
The following command initiates a dialogue to define the CHAP secret key:

```
initiator# iscsiadm modify initiator-node --CHAP-secret
Enter CHAP secret: ************
Re-enter secret: ************
```

4 *(Optional)* **Unidirectional CHAP: Set the CHAP user name on the initiator.**
By default, the initiator’s CHAP user name is set to the initiator node name.

Use the following command to use your own initiator CHAP user name:

```
initiator# iscsiadm modify initiator-node --CHAP-name new-CHAP-name
```

5 **Unidirectional CHAP – Enable CHAP authentication on the initiator.**

```
initiator# iscsiadm modify initiator-node --authentication CHAP
```

CHAP requires that the initiator node have both a username and a password. The user name is typically used by the target to look up the secret key for the given user name.

6 **Select one of the following to enable or disable bidirectional CHAP.**

- Enable bidirectional CHAP for connections with the target.

```
initiator# iscsiadm modify target-param -B enable target-ign
```

- Disable bidirectional CHAP.

```
initiator# iscsiadm modify target-param -B disable target-ign
```

7 **Bidirectional CHAP: Set the authentication method to CHAP for the target.**

```
initiator# iscsiadm modify target-param --authentication CHAP target-ign
```

8 **Bidirectional CHAP: Set the target device secret key that identifies the target.**
The following command initiates a dialogue to define the CHAP secret key:

```
initiator# iscsiadm modify target-param --CHAP-secret target-ign
```

9 **Bidirectional CHAP: If the target uses an alternate CHAP user name, set the CHAP name that identifies the target.**

By default, the target’s CHAP name is set to the target name.

You can use the following command to change the target’s CHAP name:

```
initiator# iscsiadm modify target-param --CHAP-name target-CHAP-name
```
How to Configure CHAP Authentication for Your iSCSI Target

This procedure assumes that you are logged into the local system that contains the iSCSI targets.

1 Become an administrator.

2 Determine whether you want to configure unidirectional or bidirectional CHAP.
   ■ Unidirectional authentication is the default method. Complete steps 3–5 only.

3 Unidirectional/Bidirectional CHAP: Configure the target to require that initiators identify themselves using CHAP.
   ```bash
   target# itadm modify-target -a chap target-qn
   ```

4 Unidirectional/Bidirectional CHAP: Create an initiator context that describes the initiator.
   Create the initiator context with the initiator’s full node name and with the initiator’s CHAP secret key.
   ```bash
   target# itadm create-initiator -s initiator-qn
   Enter CHAP secret: ************
   Re-enter secret: ************
   ```

5 Unidirectional/Bidirectional CHAP: If the initiator uses an alternate CHAP name, then configure the initiator-context with the alternate name.
   ```bash
   target# itadm modify-initiator -u initiator-CHAP-name initiator-qn
   ```

6 Bidirectional CHAP: Set the target device secret key that identifies this target.
   ```bash
   target# itadm modify-target -s target-qn
   Enter CHAP secret: ************
   Re-enter secret: ************
   ```

7 (Optional) Bidirectional CHAP: If the target uses an alternate CHAP user name other than the target node name (iqn), modify the target.
   ```bash
   target# itadm modify-target -u target-CHAP-name target-qn
   ```

Using a Third-Party RADIUS Server to Simplify CHAP Management in Your iSCSI Configuration

You can use a third-party RADIUS server that acts as a centralized authentication service to simplify CHAP key secret management. With this method, the recommended practice is to use the default CHAP name for each initiator node. In the common case when all initiators are using the default CHAP name, you do not have to create initiator contexts on the target.
How to Configure a RADIUS Server for Your iSCSI Target

This procedure assumes that you are logged in to the local system where you want to securely access the configured iSCSI target device.

1 Become an administrator.

2 Configure the initiator node with the IP address and the port of the RADIUS server.
   The default port is 1812. This configuration is completed once for all iSCSI targets on the target system.
   `initiator# itadm modify-defaults -r RADIUS-server-IP-address`
   Enter RADIUS secret: ************
   Re-enter secret: ************

3 Configure the shared secret key that is used for communication between the target system and the RADIUS server.
   `initiator# itadm modify-defaults -d`
   Enter RADIUS secret: ************
   Re-enter secret: ************

4 Configure the target system to require RADIUS authentication.
   This configuration can be performed for an individual target or as a default for all targets.
   `initiator# itadm modify-target -a radius target-qn`

5 Configure the RADIUS server with the following components:
   - The identity of the target node (for example, its IP address)
   - The shared secret key that the target node uses to communicate with the RADIUS server
   - The initiator’s CHAP name (for example, it’s iqn name) and the secret key for each initiator that needs to be authenticated

How to Configure a RADIUS Server for Your iSCSI Initiator

You can use a third-party RADIUS server that acts as a centralized authentication service to simplify CHAP secret key management. This setup is only useful when the initiator is requesting bidirectional CHAP authentication. You must still specify the initiator’s CHAP secret key, but you are not required to specify the CHAP secret key for each target on an initiator when using bidirectional authentication with a RADIUS server. RADIUS can be independently configured on either the initiator or the target. The initiator and the target do not have to use RADIUS.

1 Become an administrator.
2 Configure the initiator node with the IP address and the port of the RADIUS server.
   The default port is 1812.
   
   # iscsiadm modify initiator-node --radius-server ip-address:1812

3 Configure the initiator node with the shared secret key of the RADIUS server.
   The RADIUS server must be configured with a shared secret for iSCSI to interact with the server.
   
   # iscsiadm modify initiator-node --radius-shared-secret
   Enter secret:
   Re-enter secret

4 Enable the use of the RADIUS server.
   
   # iscsiadm modify initiator-node --radius-access enable

5 Set up the other aspects of CHAP bidirectional authentication.
   
   # iscsiadm modify initiator-node --authentication CHAP
   # iscsiadm modify target-param --bi-directional-authentication enable target-iqn
   # iscsiadm modify target-param --authentication CHAP target-iqn

6 Configure the RADIUS server with the following components:
   
   - The identity of this node (for example, its IP address)
   - The shared secret key that this node uses to communicate with the RADIUS server
   - The target’s CHAP name (for example, its iqn name) and the secret key for each target that needs to be authenticated

Oracle Solaris iSCSI and RADIUS Server Error Messages

This section describes the error messages that are related to an Oracle Solaris iSCSI and RADIUS server configuration. Potential solutions for recovery are also provided.

empty RADIUS shared secret
   Cause: The RADIUS server is enabled on the initiator, but the RADIUS shared secret key is not set.
   
   Solution: Configure the initiator with the RADIUS shared secret key. For more information, see “How to Configure a RADIUS Server for Your iSCSI Target” on page 271.

WARNING: RADIUS packet authentication failed
   Cause: The initiator failed to authenticate the RADIUS data packet. This error can occur if the shared secret key that is configured on the initiator node is different from the shared secret key on the RADIUS server.
   
   Solution: Reconfigure the initiator with the correct RADIUS shared secret. For more information, see “How to Configure a RADIUS Server for Your iSCSI Target” on page 271.
Setting Up iSCSI Multipathed Devices in Oracle Solaris

Multipathed I/O (MPxIO) enables I/O devices to be accessed through multiple host controller interfaces from a single instance of the I/O device.

Consider the following guidelines when using iSCSI multipathed (MPxIO) devices in Oracle Solaris:

- **Oracle Solaris iSCSI and MPxIO** – MPxIO supports target port aggregation and availability in Oracle Solaris iSCSI configurations that configure multiple sessions per target (MS/T) on the iSCSI initiator.
  - Use IP network multipathing (IPMP) for aggregation and failover of two or more NICs.
  - A basic configuration for an iSCSI host is a server with two NICs that are dedicated to iSCSI traffic. The NICs are configured by using IPMP. Additional NICs are provided for non-iSCSI traffic to optimize performance.
  - Active multipathing can only be achieved by using the iSCSI MS/T feature in Oracle Solaris, and the failover and redundancy of an IPMP configuration.
    - If one NIC fails in an IPMP configuration, IPMP handles the failover. The MPxIO driver does not detect the failure. In a non-IPMP configuration, the MPxIO driver fails and offline the path.
    - If one target port fails in an IPMP configuration, the MPxIO driver detects the failure and provides the failover. In a non-IPMP configuration, the MPxIO driver detects the failure and provides the failover.

For information about configuring multiple sessions per target, see "How to Enable Multiple iSCSI Sessions for a Target" on page 274. For information about configuring IPMP, see Chapter 15, "Administering IPMP," in Oracle Solaris Administration: Network Interfaces and Network Virtualization.

- **Oracle Solaris iSCSI, Fibre Channel (FC), and MPxIO** – The MPxIO driver provides the following behavior in more complex iSCSI/FC configurations:
  - If you have dual iSCSI to FC bridges in an FC SAN, iSCSI presents target paths to MPxIO. MPxIO matches the unique SCSI per LUN identifier, and if they are identical, presents one path to the iSCSI driver.
  - If you have a configuration that connects a target by using both iSCSI and FC, the MPxIO driver can provide different transports to the same device. In this configuration, MPxIO utilizes both paths.
  - If you are using iSCSI and FC with MPxIO, make sure that the MPxIO parameters in the /etc/driver/drv/fp.conf and the /driver/drv/iscsi.conf files match the MPxIO configuration that you want supported. For example, in fp.conf, you can determine whether MPxIO is enabled globally on the HBA or on a per-port basis.

- **Third-party hardware considerations** – Find out if your third-party HBA is qualified to work with Oracle Solaris iSCSI and MPxIO.
If you are using a third-party HBA, you might need to ask the vendor for the symmetric-option information for the `/driver/drv/scsi_vhci.conf` file.

### How to Enable Multiple iSCSI Sessions for a Target

You can use this procedure to create multiple iSCSI sessions that connect to a single target. This scenario is useful with iSCSI target devices that support login redirection or have multiple target portals in the same target portal group. Use iSCSI multiple sessions per target with the SCSI Multipathing (MPxIO) feature of Oracle Solaris. You can also achieve higher bandwidth if you utilize multiple NICs on the host side to connect to multiple portals on the same target.

The MS/T feature creates two or more sessions on the target by varying the initiator's session ID (ISID). Enabling this feature creates two SCSI layer paths on the network so that multiple targets are exposed through the iSCSI layer to the Oracle Solaris I/O layer. The MPxIO driver handles the reservations across these paths.

For more information about how iSCSI interacts with MPxIO paths, see "Setting Up iSCSI Multipathed Devices in Oracle Solaris" on page 273.

Review the following items before configuring multiple sessions for an iSCSI target:

- A typical MS/T configuration has two or more configured-sessions.
  However, if your storage supports multiple TPGTs and if you are using the SendTarget discovery method on your host system, then the number of configured sessions can be set to 1. SendTarget discovery automatically detects the existence of multiple paths, and multiple target sessions are created.
- Confirm that the `mpxio` configuration parameter is enabled in the `/etc/driver/drv/iscsi.conf` file.

```
# cd /etc/driver/drv
# grep mpxio iscsi.conf
iscsi.conf:mpxio-disable="no";
```
- Confirm that the multiple network connections are configured by using IPMP.
- Confirm that the multiple network connections are available.

```
# ipadm show-addr
```

1. **Become an administrator.**

2. **List the current parameter values for the iSCSI initiator and the target.**

   a. **List the current parameter values for the iSCSI initiator.**

```
initiator# iscsiadm list initiator-node
Initiator node name: ign.1986-03.com.sun:01:0003ba4d233b.425c293c
Initiator node alias: zkr1200
```

Oracle Solaris Administration: Devices and File Systems • February 2012
b. **List the current parameter values for the iSCSI target device.**

```
initiator# iscsiadm list target-param -v iqn.1992-08.com.abcstorage:sn.84186266
  Alias: -
  ...
  Configured Sessions: 1
```

The configured sessions value is the number of configured iSCSI sessions that will be created for each target name in a target portal group.

3 **Select one of the following to modify the number of configured sessions either at the initiator node to apply to all targets or at a target level to apply to a specific target:**

The number of sessions for a target must be between 1 and 4.

- Apply the desired parameter value to the iSCSI initiator node.
  
  ```
  initiator# iscsiadm modify initiator-node -c 2
  ```

- Apply the desired parameter value to the iSCSI target.
  
  ```
  ```

- Bind configured sessions to one or more local IP addresses.

  Configured sessions can also be bound to a local IP address. Using this method, one or more local IP addresses are supplied in a comma-separated list. Each IP address represents an iSCSI session. This method can also be used at the initiator-node or target-param level. For example:

  ```
  initiator# iscsiadm modify initiator-node -c 10.0.0.1,10.0.0.2
  ```

  **Note** – If the specified IP address is not routable, the address is ignored and the default Oracle Solaris route and IP address are used for this session.

4 **Verify that the parameter was modified.**

a. **Display the updated information for the initiator node.**

  ```
  initiator# iscsiadm list initiator-node
  Initiator node name: iqn.1996-03.com.sun:01:0003ba4d233b.425c293c
  Initiator node alias: zzr1200
  ...
  ...
  Configured Sessions: 2
  ```
b. Display the updated information for the target node.

```
initiator# iscsiadm list target-param -v iqn.1992-08.com.abcstorage:sn.84186266
  Alias: -
  -
  -
  Configured Sessions: 2
```

5 List the multiple paths to confirm that the OS device name matches the `iscsiadm list` output, and that the path count is 2 or more.

```
initiator# mpathadm list lu
```

---

## Monitoring Your iSCSI Configuration

You can display information about the iSCSI initiator and target devices by using the `iscsiadm list` command.

### How to Display iSCSI Configuration Information

1 Become an administrator.

2 Display information about the iSCSI initiator.
   
   For example:

   ```
   # iscsiadm list initiator-node
   Initiator node name: iqn.1986-03.com.sun:01:0003ba4d233b.425c293c
   Initiator node alias: zzr1200
   Login Parameters (Default/Configured):
     Header Digest: NONE/
     Data Digest: NONE/
     Authentication Type: NONE
     RADIUS Server: NONE
     RADIUS access: unknown
   Configured Sessions: 1
   ```

3 Display information about which discovery methods are in use.
   
   For example:

   ```
   # iscsiadm list discovery
   Discovery:
     Static: enabled
     Send Targets: enabled
     iSNS: enabled
   ```

### Example 14–1 Displaying iSCSI Target Information

The following example shows how to display the parameter values for a specific iSCSI target:
The `iscsiadm list target-param -v` command displays the following information:

- The authentication values for the target
- The default values for the target login parameters
- The configured value for each login parameter

The `iscsiadm list target-param -v` command displays the *default* parameter value before the `/` designator and the *configured* parameter value after the `/` designator. If you have not configured a parameter, its value displays as a hyphen (`-`). For more information, see the following examples.

```
# iscsiadm list target-param -v iqn.1992-08.com.abcstorage:sn.33592219

The authentication values for the target

- The default values for the target login parameters
- The configured value for each login parameter

The following example output displays the parameters that were negotiated between the target and the initiator:

```
# iscsiadm list target -v eui.50060e8004275511
  Target: eui.50060e8004275511

  TPGT: 1
  ISID: 4000002a0000
  Connections: 1
  CID: 0
  IP address (Local): 172.90.101.71:32813
  IP address (Peer): 172.90.101.40:3260
  Discovery Method: Static
  Login Parameters (Negotiated):
    Data Sequence In Order: yes
    Data PDU In Order: yes
    Default Time To Retain: 0
    Default Time To Wait: 3
    Error Recovery Level: 0
    First Burst Length: 65536
    Immediate Data: yes
```
Modifying iSCSI Initiator and Target Parameters

You can modify parameters on both the iSCSI initiator and the iSCSI target device. However, the only parameters that can be modified on the iSCSI initiator are the following:

- iSCSI initiator node name – You can change the initiator node name to a different name. If you change the initiator node name, the targets that were discovered by iSNS might be removed from the initiator’s target list, depending on the discovery domain configuration on the iSNS server at the time when the name was changed. For more information, see “How to Modify iSCSI Initiator and Target Parameters” on page 280.
- Header digest – NONE, the default value or CRC32.
- Data digest – NONE, the default value or CRC32.
- Authentication and CHAP secret key – For more information about setting up authentication, see “How to Configure CHAP Authentication for Your iSCSI Initiator” on page 268.

The iSCSI driver provides default values for the iSCSI initiator and iSCSI target device parameters. If you modify the parameters of the iSCSI initiator, the modified parameters are inherited by the iSCSI target device, unless the iSCSI target device already has different values.

Caution – Ensure that the target software supports the parameter to be modified. Otherwise, you might be unable to log in to the iSCSI target device. See your array documentation for a list of supported parameters.

You can modify the iSCSI parameters only after I/O between the initiator and the target is complete. The iSCSI driver reconnects the session after the changes are made by using the `iscsiadm modify` command.

Tuning iSCSI Parameters

iSCSI parameters can be tuned to adjust various response or connection time values of the iSCSI initiator. You can tune the iSCSI parameters depending on whether you want to adjust a parameter on the initiator for all targets for which the initiator system is connected or if you want to adjust parameters for a specific target.
Use the following command to change a parameter value of a specific iSCSI target:

```
iscsiadm modify target-param -T tunable-prop=value target-name
```

Use the following command to adjust a parameter value for all targets:

```
iscsiadm modify initiator-node -T tunable-prop=value
```

The tunable parameters below apply to the active connection, and change the behavior of the iSCSI initiator and the targets that connect to the initiator. The capability to dynamically tune parameters provides flexibility when configuring your iSCSI initiators.

**TABLE 14-1 iSCSI Tunable Parameters**

<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Description</th>
<th>Valid Values (seconds)</th>
<th>Default Value (seconds)</th>
</tr>
</thead>
<tbody>
<tr>
<td>recv-login-rsp-timeout</td>
<td>Session login response time – Specifies how long an iSCSI initiator waits for the response of an iSCSI session login request from a given iSCSI target.</td>
<td>0 – 3600</td>
<td>60</td>
</tr>
<tr>
<td>conn-login-max</td>
<td>Maximum connection retry time – Determines the maximum number of times the iSCSI initiator tries to connect to the target, after the iSCSI initiator to target I/O times out, or the connection fails.</td>
<td>0 – 3600</td>
<td>180</td>
</tr>
<tr>
<td>polling-login-delay</td>
<td>Login retry time interval – Determines the time interval between each iSCSI session login retry, after the iSCSI initiator to target I/O times out or the connection fails.</td>
<td>0 – 3600</td>
<td>60</td>
</tr>
</tbody>
</table>

▼ **How to Tune iSCSI Parameters**

1 **Display all tunable iSCSI parameters.**

Display the iSCSI parameter information for all targets.

```
# iscsiadm list initiator-node
Initiator node name: iqn.1986-03.com.sun:01:e00000000000.4e36d278
Initiator node alias: unknown
  Login Parameters (Default/Configured):
    Header Digest: NONE/-
    Data Digest: NONE/-
    Max Connections: 65535/-
    Authentication Type: NONE
    RADIUS Server: NONE
    RADIUS Access: disabled
    Tunable Parameters (Default/Configured):
      Session Login Response Time: 60/-
```

Chapter 14 • Configuring Storage Devices With COMSTAR 279
Maximum Connection Retry Time: 180/-
Login Retry Time Interval: 60/-
Configured Sessions: 1

Display the iSCSI parameter information for a specific target.

```bash
# iscsiadm list target-param [target-name]
```

For example:

```bash
# iscsiadm list target-param -v iqn.1986-03.com.sun:02:47ac0506-cd48-67f5-fc0d-ab7544d37538
Target: iqn.1986-03.com.sun:02:47ac0506-cd48-67f5-fc0d-ab7544d37538
  Alias: -
  Bi-directional Authentication: disabled
  Authentication Type: NONE
  Login Parameters (Default/Configured):
    Data Sequence In Order: yes/-
    Data PDU In Order: yes/-
    Default Time To Retain: 20/-
    Default Time To Wait: 2/-
    Error Recovery Level: 0/-
    First Burst Length: 65536/-
    Immediate Data: yes/-
    Initial Ready To Transfer (R2T): yes/-
    Max Burst Length: 262144/-
    Max Outstanding R2T: 1/-
    Max Receive Data Segment Length: 8192/-
    Max Connections: 65535/-
    Header Digest: NONE/-
    Data Digest: NONE/-
  Tunable Parameters (Default/Configured):
    Session Login Response Time: 60/-
    Maximum Connection Retry Time: 180/-
    Login Retry Time Interval: 60/-
  Configured Sessions: 1
```

2 Tune an iSCSI parameter.

For example, to set the maximum connection retry time to 90 seconds for one target:

```bash
# iscsiadm modify target-param -T conn-login-max=90 iqn.1986-03.com.sun:02:47ac0506-cd48-67f5-fc0d-ab7544d37538
```

For example, to set the maximum connection retry time to 90 seconds for all targets:

```bash
# iscsiadm modify initiator-node -T conn-login-max=90
```

How to Modify iSCSI Initiator and Target Parameters

The first part of this procedure illustrates how modified parameters of the iSCSI initiator are inherited by the iSCSI target device. The second part of this procedure shows how to actually modify parameters on the iSCSI target device.
This optional procedure assumes that you are logged in to the local system where access to an iSCSI target device has already been configured.

1  **Become an administrator.**

2  **List the current parameter values for the iSCSI initiator and the target device.**

   a.  **List the current parameter values for the iSCSI initiator.**

      ```
      initiator# iscsiadm list initiator-node
      Initiator node name: iqn.1986-03.com.sun:01:0003ba4d233b.425c293c
      Initiator node alias: zzr1200
      Login Parameters (Default/Configured):
      Header Digest: NONE/-
      Data Digest: NONE/-
      Authentication Type: NONE
      RADIUS Server: NONE
      RADIUS access: unknown
      Configured Sessions: 1
      ```

   b.  **List the current parameter values for the iSCSI target device.**

      ```
      initiator# iscsiadm list target-param -v iqn.1992-08.com.abcstorage:sn.84186266
      Alias: -
      Bi-directional Authentication: disabled
      Authentication Type: NONE
      Login Parameters (Default/Configured):
      Data Sequence In Order: yes/-
      Data PDU In Order: yes/-
      Default Time To Retain: 20/-
      Default Time To Wait: 2/-
      Error Recovery Level: 0/-
      First Burst Length: 65536/-
      Immediate Data: yes/-
      Initial Ready To Transfer (R2T): yes/-
      Max Burst Length: 262144/-
      Max Outstanding R2T: 1/-
      Max Receive Data Segment Length: 65536/-
      Max Connections: 1/-
      Header Digest: NONE/-
      Data Digest: NONE/-
      Configured Sessions: 1
      ```

      Note that both header digest and data digest parameters are currently set to NONE for both the iSCSI initiator and the iSCSI target device.

      To review the default parameter values for the iSCSI target device, see the `iscsiadm list target-param` output in Example 14–1.

3  **Modify the parameter of the iSCSI initiator.**

   For example, set the header digest to CRC32.

   ```
   initiator# iscsiadm modify initiator-node -h CRC32
   ```
If you change the initiator node name, the targets that were discovered by iSNS might be logged out and removed from the initiator's target list, if the new name does not belong to the same discovery domain as that of the targets. However, if the targets are in use, they are not removed. For example, if a file is open or a file system is mounted on these targets, the targets will not be removed.

You might also see new targets after the name change if these targets and the new initiator node name belong to the same discovery domain.

4 Verify that the parameter was modified.

a. Display the updated parameter information for the iSCSI initiator.

```
initiator# iscsiadm list initiator-node
Initiator node name: iqn.1986-03.com.sun:01:0003ba4d233b.425c293c
Initiator node alias: zzr1200
Login Parameters (Default/Configured):
  Header Digest: NONE/CRC32
  Data Digest: NONE/-
  Authentication Type: NONE
  RADIUS Server: NONE
  RADIUS access: unknown
Configured Sessions: 1
```

Note that the header digest is now set to CRC32.

b. Display the updated parameter information for the iSCSI target device. For example:

```
initiator# iscsiadm list target-param -v iqn.1992-08.com.abcstorage:sn.84186266
  Alias: -
  Bi-directional Authentication: disabled
  Authentication Type: NONE
Login Parameters (Default/Configured):
  Data Sequence In Order: yes/-
  Data PDU In Order: yes/-
  Default Time To Retain: 20/-
  Default Time To Wait: 2/-
  Error Recovery Level: 0/-
  First Burst Length: 65536/-
  Immediate Data: yes/-
  Initial Ready To Transfer (R2T): yes/-
  Max Burst Length: 262144/-
  Max Outstanding R2T: 1/-
  Max Receive Data Segment Length: 65536/-
  Max Connections: 1/-
  Header Digest: CRC32/-
  Data Digest: NONE/-
Configured Sessions: 1
```

Note that the header digest is now set to CRC32.

5 Verify that the iSCSI initiator has reconnected to the iSCSI target.

```
initiator# iscsiadm list target -v iqn.1992-08.com.abcstorage:sn.84186266
  TPGT: 2
```
6 (Optional) Reset an iSCSI initiator parameter or an iSCSI target device parameter.

You can reset a parameter to its default value by using the `iscsiadm modify` command. Or, you can use the `iscsiadm remove` command to reset all parameters to their default values.

The `iscsiadm modify target-param` command changes only the parameters that are specified on the command line.

The following example shows how to reset the header digest to NONE:

```
initiator# iscsiadm modify target-param -h none iqn.1992-08.com.abcstorage:sn...
```

For information about the `iscsiadm remove target-param` command, see `iscsiadm(1M)`.

## Troubleshooting iSCSI Configuration Problems

The following tools are available to troubleshoot general iSCSI configuration problems:

- `snoop` – This tool has been updated to support iSCSI packets.
- `wireshark` – This product is available from [http://www.wireshark.org/](http://www.wireshark.org/).

Both tools can filter iSCSI packets on port 3260.

The following sections describe how to troubleshoot various iSCSI issues and error messages.
No Connections to the iSCSI Target From the Local System

▼ How to Troubleshoot iSCSI Connection Problems

1. Become an administrator.

2. List your iSCSI target information.
   For example:
   ```
   initiator# iscsiadm list target
   TPGT: default
   ISID: 4000002a0000
   Connections: 0
   ```

3. If no connections are listed in the `iscsiadm list target` output, check the `/var/adm/messages` file for possible reasons why the connection failed.
   You can also verify whether the connection is accessible by using the `ping` command or by connecting to the storage device's iSCSI port by using the `telnet` command to ensure that the iSCSI service is available. The default port is 3260.

   In addition, check the storage device's log file for errors.

4. If your target is not listed in the `iscsiadm list target` output, check the `/var/adm/messages` file for possible causes.
   If you are using SendTargets as the discovery method, try listing the `discovery-address` by using the `-v` option to ensure that the expected targets are visible to the host. For example:
   ```
   initiator# iscsiadm list discovery-address -v 10.0.0.1
   Discovery Address: 10.0.0.1:3260
   Target name: eui.210000203787dfc0
   Target address: 10.0.0.1:11824
   Target name: eui.210000203787e07b
   Target address: 10.0.0.1:11824
   ```

   If you are using iSNS as the discovery method, try enabling the iSNS discovery method and listing the `isns-server` using the `-v` option to ensure that the expected targets are visible to the host. For example:
   ```
   initiator# iscsiadm list isns-server -v
   iSNS Server IP Address: 10.20.56.3205
   Target name: iqn.1992-08.com.xyz:sn.1234566
   Target address: 10.20.57.161:3260, 1
   Target address: 10.20.56.206:3260, 1
   Target address: 10.20.56.206:3260, 1
   ```
iSCSI Device or Disk Is Not Available on the Local System

❖ How to Troubleshoot iSCSI Device or Disk Unavailability

1 Become an administrator.

2 Identify the LUNs that were discovered on this target during enumeration.
   The -S option shows which LUNs were discovered on this target during enumeration.
   For example:
   ```
   # iscsiadm list target -S
   Target: iqn.2001-05.com.abcstorage:6-8a0900-37ad70401-bcfff02df8a421df-zzr1200-01
   TPGT: default
   ISID: 4000002a0000
   Connections: 1
   LUN: 0
   Vendor: ABCSTOR
   Product: 0010
   OS Device Name: /dev/rdsk/c3t34d0s2
   
   ```

3 Review the /var/adm/messages file to see if an error was reported.
   If you think a LUN should be listed but it is not, then check this log file.

4 Check the storage device's log files for errors.

5 Ensure that any storage device LUN masking is properly configured.

Use LUN Masking When Using the iSNS Discovery Method

Avoid using the iSNS discovery domain as the means to control storage authorization to specific initiators. Use LUN masking instead to ensure that only authorized initiators can access a LUN.

If you remove a target from a discovery domain while the target is in use, the iSCSI initiator does not log out from this target. If you do not want this initiator to access this target (and the associated LUNs), you must use LUN masking. Removing the target from the discovery domain is not sufficient.
General iSCSI Error Messages

This section describes the iSCSI messages that might be found in the /var/adm/messages file and potential solutions for recovery.

The message format is as follows:

```
iscsi TYPE (OID) STRING (STATUS-CLASS#/STATUS-DETAIL#)
```

- **TYPE**
  Is either connection or session.

- **OID**
  Is the object ID of the connection or session. This ID is unique for an OS instance.

- **STRING**
  Is a description of the condition.

- **STATUS-CLASS#/STATUS-DETAIL#**
  These values are returned in an iSCSI login response as defined by RFC 3720.

### iscsi connection(OID) login failed - Miscellaneous iSCSI initiator errors.

- **Cause:** The device login failed due to some form of initiator error.

### iscsi connection(OID) login failed - Initiator could not be successfully authenticated.

- **Cause:** The device could not successfully authenticate the initiator.

- **Solution:** If applicable, verify that the settings for CHAP names, CHAP passwords, or the RADIUS server are correct.

### iscsi connection(OID) login failed - Initiator is not allowed access to the given target.

- **Cause:** The device cannot allow the initiator access to the iSCSI target device.

- **Solution:** Verify your initiator name and confirm that it is properly masked or provisioned by the storage device.

### iscsi connection(OID) login failed - Requested ITN does not exist at this address.

- **Cause:** The device does not provide access to the iSCSI target name (ITN) that you are requesting.

- **Solution:** Verify that the initiator discovery information is specified properly and that the storage device is configured properly.
iscsi connection (OID) login failed - Requested ITN has been removed and no forwarding address is provided.

**Cause:** The device can no longer provide access to the iSCSI target name (ITN) that you are requesting.

**Solution:** Verify that the initiator discovery information has been specified properly and that the storage device has been configured properly.

iscsi connection (OID) login failed - Requested iSCSI version range is not supported by the target.

**Cause:** The initiator's iSCSI version is not supported by the storage device.

iscsi connection (OID) login failed - No more connections can be accepted on this Session ID (SSID).

**Cause:** The storage device cannot accept another connection for this initiator node to the iSCSI target device.

iscsi connection (OID) login failed - Missing parameters (e.g., iSCSI initiator and/or target name).

**Cause:** The storage device is reporting that the initiator or target name has not been properly specified.

**Solution:** Properly specify the iSCSI initiator or target name.

iscsi connection (OID) login failed - Target hardware or software error.

**Cause:** The storage device encountered a hardware or software error.

**Solution:** Consult the storage documentation, or contact the storage vendor for further assistance.

iscsi connection (OID) login failed - iSCSI service or target is not currently operational.

**Cause:** The storage device is currently not operational.

**Solution:** Consult the storage documentation, or contact the storage vendor for further assistance.

iscsi connection (OID) login failed - Target has insufficient session, connection or other resources.

**Cause:** The storage device has insufficient resources.

**Solution:** Consult the storage documentation, or contact the storage vendor for further assistance.
iscsi connection\((OID)\) login failed - unable to initialize authentication  
Cause: The initiator was unable to initialize or set authentication properly.  
Solution: Verify that your initiator settings for authentication are properly configured.

iscsi connection\((OID)\) login failed - unable to make login pdu  
Cause: The initiator was unable to make a login payload data unit (PDU) based on the initiator or storage device settings.  
Solution: Try resetting any target login parameters or other nondefault settings.

iscsi connection\((OID)\) login failed - failed to transfer login  
iscsi connection\((OID)\) login failed - failed to receive login response  
Cause: The initiator failed to transfer or receive a login payload data unit (PDU) across the network connection.  
Solution: Verify that the network connection is reachable.

iscsi connection\((OID)\) login failed - received invalid login response (OP CODE)  
Cause: The storage device has responded to a login with an unexpected response.

iscsi connection\((OID)\) login failed - login failed to authenticate with target  
Cause: The initiator was unable to authenticate the storage device.  
Solution: Verify that your initiator settings for authentication are properly configured.

iscsi connection\((OID)\) login failed - initiator name is required  
Cause: An initiator name must be configured to perform all actions.  
Solution: Verify that the initiator name is configured.

iscsi connection\((OID)\) login failed - authentication receive failed  
iscsi connection\((OID)\) login failed - authentication transmit failed  
Cause: The initiator was unable to transmit or receive authentication information.  
Solution: Verify network connectivity with the storage device or the RADIUS server, as applicable.
iscsi connection(OID) login failed - login redirection invalid
   **Cause:** The storage device attempted to redirect the initiator to an invalid destination.

   **Solution:** Consult the storage documentation, or contact the storage vendor for further assistance.

iscsi connection(OID) login failed - target protocol group tag mismatch,
   expected <TPGT>, received <TPGT>
   **Cause:** The initiator and target had a TPGT (target portal group tag) mismatch.

   **Solution:** Verify your TPGT discovery settings on the initiator or the storage device.

iscsi connection(OID) login failed - can’t accept PARAMETER in security stage
   **Cause:** The device responded with an unsupported login parameter during the security phase
   of login.

   **Solution:** The parameter name is noted for reference. Consult the storage documentation, or
   contact the storage vendor for further assistance.

iscsi connection(OID) login failed - HeaderDigest=CRC32 is required, can’t accept VALUE
iscsi connection(OID) login failed - DataDigest=CRC32 is required, can’t accept VALUE
   **Cause:** The initiator is only configured to accept a HeaderDigest or DataDigest that is set to
   CRC32 for this target. The device returned the value of VALUE.

   **Solution:** Verify that the initiator and device digest settings are compatible.

iscsi connection(OID) login failed - HeaderDigest=None is required, can’t accept VALUE
iscsi connection(OID) login failed - DataDigest=None is required, can’t accept VALUE
   **Cause:** The initiator is only configured to accept a HeaderDigest or DataDigest that is set to
   NONE for this target. The device returned the value of VALUE.

   **Solution:** Verify that the initiator and device digest settings are compatible.

iscsi connection(OID) login failed - can’t accept PARAMETER
   **Cause:** The initiator does not support this parameter.

iscsi connection(OID) login failed - can’t accept MaxOutstandingR2T VALUE
   **Cause:** The initiator does not accept MaxOutstandingR2T of the noted VALUE.
iscsi connection(OID) login failed - can't accept MaxConnections VALUE
Cause: The initiator does not accept the maximum connections of the noted VALUE.

iscsi connection(OID) login failed - can't accept ErrorRecoveryLevel VALUE
Cause: The initiator does not accept an error recovery level of the noted VALUE.

iscsi session(OID) NAME offline
Cause: All connections for this target NAME have been removed or have failed.

iscsi connection(OID) failure - unable to schedule enumeration
Cause: The initiator was unable to enumerate the LUNs on this target.

Solution: You can force LUN enumeration by running the devfsadm -i iscsi command. For more information, see devfsadm(1M).

iscsi connection(OID) unable to connect to target NAME (errno:ERRNO)
Cause: The initiator failed to establish a network connection.

Solution: For information about the specific ERRNO on the connection failure, see the /usr/include/sys/errno.h file.
Configuring and Managing the Oracle Solaris Internet Storage Name Service (iSNS)

This chapter provides an overview of the Internet Storage Name Service (iSNS), and describes how to configure the Oracle Solaris iSNS server, manage the iSNS server, and manage iSNS clients. This chapter discusses the following topics:

- “The iSNS Technology (Overview)” on page 291
- “Configuring the iSNS Server” on page 293
- “Managing the iSNS Server and Clients” on page 299

The iSNS Technology (Overview)

The Internet Storage Name Service (iSNS) is a protocol that allows dynamic discovery of iSCSI initiators and targets within an IP storage area network SAN. The iSNS protocol enables identification, connection to, and management of iSCSI devices by providing the following services:

- **Name registration and discovery**: The source of data that is to be stored (known as the *initiator*) and the storage object (known as the *target*) register their attributes and address, and then can obtain information about accessible storage devices dynamically.

- **Discovery domains and logon control**: Resources in a typical storage network are divided into groups called *discovery domains*, which can be administered through network management applications. Discovery domains enhance security by providing access control to targets that are not enabled with their own access controls, while limiting the logon process of each initiator to a relevant subset of the available targets in the network.

- **State-change notification**: The iSNS server notifies relevant iSNS clients of network events, for example, a newly created disk Logical Unit Number (LUN), storage resources going offline, discovery domain membership changes and link failures in a network. These notifications let a network quickly adapt to changes in topology, which is key to scalability and availability. This is an optional service.

- **Entity status inquiry**: The iSNS server verifies that an iSNS client is available. As a result, a status change notification might be issued. This is an optional service.
In a simple configuration, the source of data that is to be stored (the initiator) exchanges data with a storage object (the target). The initiator can locate the target and the target always recognizes the initiator. For example, the Oracle StorageTek 5320 Network Attached Storage (NAS) appliance is an iSCSI target because it stores data. The data comes from various iSCSI clients such as a data management applications or network interface cards which act as initiators. However, in large and complex configurations, it is difficult and time-consuming to configure every initiator for every target and for every target to recognize every initiator. The iSNS server resolves this by using discovery and security mechanisms to dynamically and automatically identify initiators and targets, and manage their connections to authorized resources.

After a Oracle Solaris system has been configured as an iSNS server, all targets and initiators can register with the server. The targets and initiators become iSCSI clients or nodes of the iSNS server. These clients are members of the default discovery domain, the only domain in the default discovery domain set. When you enable the default discovery domain set, the iSNS server can provide the iSCSI Name Service (iSNS) for the clients in a simple manner.

To take advantage of the iSCSI Name Service’s abilities, create several discovery domain sets and discovery domains. Then assign the clients to different domains, overlapping their memberships. The iSNS server keeps track of the clients’ status as a member of one or more discovery domains. For example, when a new storage device is added to the storage network and is registered with the iSNS server, it is in the default discovery domain in the default discovery domain set. You then assign this target to the discovery domains whose initiators will use it as a resource. The iSNS server then removes this target as a member of the default discovery domain in the default discovery domain set.

All initiators and targets are assigned to at least one discovery domain. Assigning an initiator to one discovery domain restricts its access to those targets in the same discovery domain set. Assigning an initiator to several discovery domains allows it to find and use targets in all of the discovery domain sets that include the initiator’s discovery domain. You can manage access to clients by disabling and enabling their discovery domain sets without affecting the clients in other discovery domain sets.

For example, a site has two discovery domain sets in addition to the default one: Production and Research. Within the two discovery domain sets are three domains in addition to the default one: Development, Operations, and Finance. The Development discovery domain is in the Research discovery domain set, Operations is in the Production domain set, and Finance is a member of both discovery domain sets. Each client has been assigned to the discovery domain set that uses it the most. A data application in the Operations discovery domain can locate and get access to storage devices in the Production discovery domain set because it is a member of that discovery domain set but it cannot get access to a storage device in the Research discovery domain set. A data application in the Finance discovery domain can locate storage devices in both the Production and Research discovery domain sets because it is a member of both sets. If the Research discovery domain set were disabled, initiators in the Finance discovery domain would not have access to the Research storage devices but would continue to have access to those in the Production discovery domain set.
Configuring the iSNS Server

You can configure the iSNS server using as described in the following task maps and sections.

<table>
<thead>
<tr>
<th>Task</th>
<th>For Instructions</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Accept the default properties of the iSNS server or change them.</td>
<td></td>
</tr>
<tr>
<td>A. Notification of state changes of the server</td>
<td>“How to Set Notifications for Changes in Server State” on page 294</td>
</tr>
<tr>
<td>B. Number of attempts to determine a client’s availability</td>
<td>“How to Set the Number of Retries for Client Inquiries” on page 295</td>
</tr>
<tr>
<td>C. Location of file that stores client data.</td>
<td>“How to Specify the Data Store Location” on page 295</td>
</tr>
<tr>
<td>2. Enable the iSNS server and display the settings.</td>
<td>“How to Install the iSNS Server Package” on page 294</td>
</tr>
<tr>
<td></td>
<td>“How to Display the Current Server Configuration” on page 296</td>
</tr>
<tr>
<td>3. Register all clients with the iSNS server.</td>
<td>Use the client’s management interface’s iSCSI configuration function to specify the IP address of the iSNS server and to allow discovery.</td>
</tr>
<tr>
<td>4. Enable the default discovery domain set.</td>
<td>“How to Enable the Default Discovery Domain Set” on page 296</td>
</tr>
</tbody>
</table>

After these tasks, the iSNS server is operating in a minimal manner. All of the clients are in the default discovery domain and are unassigned. Each one can identify and get access to all of the other ones.

<table>
<thead>
<tr>
<th>Task</th>
<th>For Instructions</th>
</tr>
</thead>
<tbody>
<tr>
<td>5. Create the discovery domain sets for your site.</td>
<td>“How to Create the Discovery Domain Sets” on page 296</td>
</tr>
<tr>
<td>6. Create the discovery domains for your site.</td>
<td>“How to Create the Discovery Domains” on page 297</td>
</tr>
<tr>
<td>7. Add each discovery domain to one or more discovery domain set.</td>
<td>“How to Add a Discovery Domain to a Discovery Domain Set” on page 297</td>
</tr>
<tr>
<td>8. Assign clients to one or more discovery domains.</td>
<td>“How to Assign Clients to a Discovery Domain” on page 298</td>
</tr>
<tr>
<td>9. Verify the membership of clients in discovery domains and the membership of discovery domains in discovery domain sets.</td>
<td>“How to Display the Status of a Discovery Domain Set” on page 299</td>
</tr>
<tr>
<td></td>
<td>“How to Display the Status of a Discovery Domain” on page 300</td>
</tr>
<tr>
<td></td>
<td>“How to Display the Status of Clients” on page 300</td>
</tr>
</tbody>
</table>
The next section provides instructions for setting up the iSNS environment. The following topics are discussed:

- “Setting Up the iSNS Administrative Settings” on page 294
- “Using the Command Line Interface to Configure iSNS” on page 296

### Setting Up the iSNS Administrative Settings

This section provides the procedures for changing the default administrative settings of the iSNS service and for starting the iSNS daemon. If you change a setting after the iSNS server has been started, you need to refresh the iSNS server. If you change the data store location, you need to restart the iSNS server.

- “How to Install the iSNS Server Package” on page 294
- “How to Set Notifications for Changes in Server State” on page 294
- “How to Set the Number of Retries for Client Inquiries” on page 295
- “How to Specify the Data Store Location” on page 295

See the man page for the `isns(1M)` command details about these operations.

#### How to Install the iSNS Server Package

Install the iSNS server package and start the iSNS service.

1. Become an administrator.
2. Install the iSNS server package.
   ```
   # pkg install service/storage/isns
   ```
3. Enable the iSNS service.
   ```
   # svcadm enable isns_server
   ```
4. Verify that the service is running.
   ```
   # svcsc svc:/network/isns_server:default
   STATE  STIME   FMRI
   online 16:10:49 svc:/network/isns_server:default
   ```

#### How to Set Notifications for Changes in Server State

By default, all clients are notified when the iSNS server is not available. To disable these notifications, change the `Management_SCNs_Enabled` property.

1. Use the “iSNS Server Management” RBAC profile to obtain the authorizations needed for managing the iSNS service.

Roles contain authorizations and privileged commands. For more information about roles, see “Initially Configuring RBAC (Task Map)” in *Oracle Solaris Administration: Security Services*. 
2 Use the svccfg command to disable the property:
   # svccfg -s svc:/network/isns_server setprop config/Management_SCN_Enabled=false

3 Reload the server configuration:
   # svcadm refresh svc:/network/isns_server

▼ How to Set the Number of Retries for Client Inquiries
   The default number of retries is 3. If the server does not get a response to three inquiries, it
   registers that client as unavailable. To change the number of retries, change the value of the
   ESI Retry Threshold property.

   1 Use the “iSNSServerManagement” RBAC profile to obtain the authorizations needed for
      managing the iSNS service.
      Roles contain authorizations and privileged commands. For more information about roles, see

   2 Use the svccfg command to change the property to, for example, 6 retries:
      # svccfg -s svc:/network/isns_server setprop config/ESI_retry_threshold_count=6

   3 Reload the server configuration:
      # svcadm refresh svc:/network/isns_server

▼ How to Specify the Data Store Location
   The default location and name for the file that contains the client data is
   /etc/isns/isnsdata.xml If you have a complex network environment that includes one or
   more backup iSNS servers, the data store must reside in a common location so that all servers
   can use it. Use the data_store_location property to specify the new location. You can also
   change the name of the file.

   1 Use the “iSNSServerManagement” RBAC profile to obtain the authorizations needed for
      managing the iSNS service.
      Roles contain authorizations and privileged commands. For more information about roles, see

   2 Use the svccfg command to change the location to, for example, /etc/isns2/isns_data.xml:
      # svccfg -s svc:/network/isns_server setprop config/data_store_location="/etc/isns2/isns_data.xml"

   3 If you change the data store location after the server has been enabled, you must restart the
      server:
      # svcadm restart svc:/network/isns_server
Using the Command Line Interface to Configure iSNS

This section provides the procedures for configuring the iSNS server using the command line interface. The following topics are discussed:

■ “How to Display the Current Server Configuration” on page 296
■ “How to Enable the Default Discovery Domain Set” on page 296
■ “How to Create the Discovery Domain Sets” on page 296
■ “How to Create the Discovery Domains” on page 297
■ “How to Add a Discovery Domain to a Discovery Domain Set” on page 297
■ “How to Assign Clients to a Discovery Domain” on page 298

These procedures use the isnsadm(1M) command. See the man page for a complete description of all of the command options.

▼ How to Display the Current Server Configuration

The following command shows the properties of the iSNS server:

```bash
# isnsadm show-config
```

Data Store Location: /etc/isns/isnsdata.xml
Entity Status Inquiry Non-Response Threshold: 3
Management SCN Enabled: yes
Authorized Control Node Names: -

▼ How to Enable the Default Discovery Domain Set

1 Use the “iSNS Server Management” RBAC profile to obtain the authorizations needed for managing the iSNS service.

Roles contain authorizations and privileged commands. For more information about roles, see “Initially Configuring RBAC (Task Map)” in Oracle Solaris Administration: Security Services.

2 Activate the default discovery domain set:

```bash
# isnsadm enable-dd-set Default
```

▼ How to Create the Discovery Domain Sets

1 Use the “iSNS Server Management” RBAC profile to obtain the authorizations needed for managing the iSNS service.

Roles contain authorizations and privileged commands. For more information about roles, see “Initially Configuring RBAC (Task Map)” in Oracle Solaris Administration: Security Services.

2 Create a discovery domain set:

```bash
# isnsadm create-dd-set set_name
```
3 Enable the discovery domain set:

   # isnsadm enable-dd-set set_name

4 View all the discovery domain sets, including the new one:

   # isnsadm list-dd-set -v
   
   DD Set name: Default
   State: Enabled
   DD Set name: set_name
   State: Enabled

   The list of discovery domain sets includes the default discovery domain set as well as the new one.

▼ How to Create the Discovery Domains

New discovery domains are members of the default discovery domain set. After you create them, you add them to the new discovery domain set.

1 Use the “ISNS Server Management” RBAC profile to obtain the authorizations needed for managing the iSNS service.

   Roles contain authorizations and privileged commands. For more information about roles, see “Initially Configuring RBAC (Task Map)” in Oracle Solaris Administration: Security Services.

2 Create the discovery domain:

   # isnsadm create-dd domain_name

3 View the new discovery domain in the Default discovery domain set:

   # isnsadm list-dd-set
   
   DD name: name
   DD set(s): Default

4 Create other discovery domains.

▼ How to Add a Discovery Domain to a Discovery Domain Set

This task removes the discovery domain from the default discovery domain set and adds it the discovery domain set that you specify. Because the new discovery domain set has been enabled, all the clients in its discovery domains can be discovered by the iSNS server.

   You do not need to have privileges to list the members of the discovery domains and discovery domain sets.

1 Use the “ISNS Server Management” RBAC profile to obtain the authorizations needed for managing the iSNS service.

   Roles contain authorizations and privileged commands. For more information about roles, see “Initially Configuring RBAC (Task Map)” in Oracle Solaris Administration: Security Services.
2 List the discovery domains to identify the one you want to add.
   # isnsadm list-dd -v

3 List the discovery domain sets to identify the one you want as the container for the new
discovery domain.
   # isnsadm list-dd-set

4 Move the discovery domain to the discovery domain set that you want:
   # isnsadm add-dd domain_name -s set_name

5 View the new addition to the discovery domain set:
   # isnsadm list-dd-set -v domain_name

How to Assign Clients to a Discovery Domain

Before You Begin
Use the client’s management interface to register the client. Using the iSCSI configuration
function, specify the IP address of the iSNS server and allow discovery of the client by the iSNS
server.

1 Use the “iSNS Server Management” RBAC profile to obtain the authorizations needed for
managing the iSNS service.

Roles contain authorizations and privileged commands. For more information about roles, see

2 Verify that the client has been registered with the iSNS server:
   # isnsadm list-node
   iSCSI Name: iqn.1986-03.com.sun:01:000e0c9f10da.45173FEA.engr
   Alias: STK5320_NAS
   Type: Target
   
   iSCSI Name: iqn.1986-03.com.sun:01:000e0c9f10da.454F00A2.acct
   Alias: 
   Type: Initiator

The output shows the clients’ iSCSI names.

3 Verify the name of the discovery domain:
   # isnsadm list-dd

4 Add the client to the discovery domain:
   # isnsadm add-node -d domain_name iSCSI_Name

For example, to add the target called “STK5320_NAS” to the Eng-dd discovery domain:

   # isnsadm add-node -d Eng-dd iqn.1986-03.com.sun:01:000e0c9f10da.454F00A2.engr
List all the clients in the discovery domain to verify the client has been added:

```
isnsadm list-dd -v domain_name
```

For example, to check the Eng-dd discovery domain:

```
isnsadm list-dd -v Eng-dd
```

### Managing the iSNS Server and Clients

This section describes how to maintain the iSNS discovery domain sets and their members, the initiators and targets. As the site grows, continue to add clients, discovery domains, and discovery domain sets as described in the following sections:

- “How to Create the Discovery Domain Sets” on page 296
- “How to Create the Discovery Domains” on page 297
- “How to Add a Discovery Domain to a Discovery Domain Set” on page 297
- “How to Assign Clients to a Discovery Domain” on page 298

This section provides the other procedures for managing the iSNS server, using the command line interface. The following topics are discussed:

- “How to Display the Status of a Discovery Domain Set” on page 299
- “How to Display the Status of a Discovery Domain” on page 300
- “How to Display the Status of Clients” on page 300
- “How to Remove a Client from a Discovery Domain” on page 300
- “How to Remove a Discovery Domain from a Discovery Domain Set” on page 301
- “How to Disable a Discovery Domain Set” on page 301
- “How to Remove a Discovery Domain Set” on page 301

### How to Display the Status of a Discovery Domain Set

Show the status of the discovery domain set and list the discovery domains that are its members:

```
isnsadm list-dd-set -v set_name
```
How to Display the Status of a Discovery Domain

- Show the status of the discovery domain and lists the clients that are its members:
  
  ```
  # isnsadm list-dd -v domain_name
  ```

How to Display the Status of Clients

- Select one of the following to display client status:
  - Show the status of all clients:
    ```
    # isnsadm list-node -v
    ```
  - Show the status of only the clients that are targets, that is, storage objects:
    ```
    # isnsadm list-node -t
    ```

How to Remove a Client from a Discovery Domain

1. Use the "ISNS Server Management" RBAC profile to obtain the authorizations needed for managing the ISNS service.

   Roles contain authorizations and privileged commands. For more information about roles, see "Initially Configuring RBAC (Task Map)" in Oracle Solaris Administration: Security Services.

2. List the clients to identify the one you want to remove.

   ```
   # isnsadm list-node -v
   iSCSI Name: iqn.1986-03.com.sun:01:000e0c9f10da.45173FEA.engr
   Alias: STK5320_NAS
   Type: Target
   Network Entity: SE5310
   Portal: 172.20.57.95:3260
   Portal Group: 1
   Portal: 172.20.56.95:3260
   Portal Group: 1
   DD Name: Research,Finance
   ```

   The output shows the client’s iSCSI name and the name of the discovery domains of which it is a member.

3. Remove the client from the discovery domain.

   ```
   # isnsadm remove-node -d domain_name iSCSI_name
   ```
How to Remove a Discovery Domain from a Discovery Domain Set

1. Use the “ISNS Server Management” RBAC profile to obtain the authorizations needed for managing the ISNS service.
   Roles contain authorizations and privileged commands. For more information about roles, see “Initially Configuring RBAC (Task Map)” in Oracle Solaris Administration: Security Services.

2. List the discovery domains to identify the one you want to remove.
   
   # isnsadm list-dd -v

3. Remove the discovery domain from the discovery domain set.
   
   # isnsadm remove-dd set_name domain_name

How to Disable a Discovery Domain Set

1. Use the “ISNS Server Management” RBAC profile to obtain the authorizations needed for managing the ISNS service.
   Roles contain authorizations and privileged commands. For more information about roles, see “Initially Configuring RBAC (Task Map)” in Oracle Solaris Administration: Security Services.

2. Deactivate a discovery domain set:
   
   # isnsadm disable-dd-set set_name

3. Verify that the state of the discovery domain set has changed to Disabled:
   
   # isnsadm list-dd-set set_name

How to Remove a Discovery Domain Set

After you remove a discovery domain set, its discovery domains remain. A discovery domain must be a member of at least one discovery domain set.

1. Use the “ISNS Server Management” RBAC profile to obtain the authorizations needed for managing the ISNS service.
   Roles contain authorizations and privileged commands. For more information about roles, see “Initially Configuring RBAC (Task Map)” in Oracle Solaris Administration: Security Services.

2. List the discovery domain sets to identify the one you want to remove.
   
   # isnsadm list-dd-set -v
3  Remove the discovery domain set.

   # isnsadm remove-dd-set set_name
This chapter describes the `format` utility’s menus and commands.

This is a list of the reference information in this chapter.

- “Recommendations and Requirements for Using the `format` Utility” on page 303
- “`format` Menu and Command Descriptions” on page 304
- “Rules for Input to `format` Commands” on page 310
- “Getting Help on the `format` Utility” on page 311

For a overview of when to use the `format` utility, see “`format` Utility” on page 191.

**Recommendations and Requirements for Using the `format` Utility**

You must be an administrator or have assumed an equivalent role to use the `format` utility. Otherwise, the following error message is displayed when you try to use the `format` utility:

```
$ format
Searching for disks...done
No permission (or no disks found)!
```

Keep the following guidelines in mind when you use the `format` utility and want to preserve the existing data:

- Back up all files on the disk drive.
- Save all your defect lists in files by using the `format` utility’s `dump` command. The file name should include the drive type, model number, and serial number.
- Save the paper copies of the manufacturer’s defect list that was shipped with your drive.
format Menu and Command Descriptions

The format main menu appears similar to the following:

FORMAT MENU:
  disk      - select a disk
  type      - select (define) a disk type
  partition - select (define) a partition table
  current   - describe the current disk
  format    - format and analyze the disk
  fdisk     - run the fdisk program (x86 only)
  repair    - repair a defective sector
  label     - write label to the disk
  analyze   - surface analysis
  defect    - defect list management
  backup    - search for backup labels
  verify    - read and display labels
  save      - save new disk/partition definitions
  inquiry   - show vendor, product and revision
  volname   - set 8-character volume name
  !<cmd>    - execute <cmd>, then return
  quit

The following table describes the main menu items for the format utility.

TABLE 16–1 The Main Menu Item Descriptions for the format Utility

<table>
<thead>
<tr>
<th>Menu Item</th>
<th>Command or Menu?</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>disk</td>
<td>Command</td>
<td>Lists all of the system’s drives. Also lets you choose the disk you want to use in subsequent operations. This disk is referred to as the current disk.</td>
</tr>
<tr>
<td>type</td>
<td>Command</td>
<td>Identifies the manufacturer and model of the current disk. Also displays a list of known drive types. Choose the Auto configure option for all SCSI-2 disk drives.</td>
</tr>
<tr>
<td>partition</td>
<td>Menu</td>
<td>Creates and modifies slices. For more information, see “partition Menu” on page 306.</td>
</tr>
</tbody>
</table>
| current   | Command          | Displays the following information about the current disk:  
  ■ Device name and device type  
  ■ Number of cylinders, alternate cylinders, heads and sectors  
  ■ Physical device name |
### TABLE 16–1 The Main Menu Item Descriptions for the format Utility (Continued)

<table>
<thead>
<tr>
<th>Menu Item</th>
<th>Command or Menu?</th>
<th>Description</th>
</tr>
</thead>
</table>
| format    | Command          | Formats the current disk by using one of these sources of information in this order:  
1. Information that is found in the `format.dat` file  
2. Information from the automatic configuration process  
3. Information that you type at the prompt if no `format.dat` entry exists  
This command does not apply to IDE disks. IDE disks are preformatted by the manufacturer. |
| fdisk     | Menu             | x86 platform only: Runs the `fdisk` program to create a Solaris `fdisk` partition.  
The `fdisk` command cannot be used on disks with an EFI label that are greater than 1 terabyte in size. |
| repair    | Command          | Repairs a specific block on the current disk. |
| label     | Command          | Writes a new label to the current disk. |
| analyze   | Menu             | Runs read, write, and compare tests. For more information, see "analyze Menu" on page 307. |
| defect    | Menu             | Retrieves and displays defect lists. For more information, see "defect Menu" on page 309. This feature does not apply to IDE disks. IDE disks manage defects automatically. |
| backup    | Command          | VTOC – Searches for backup labels.  
EFI – Not supported. |
| verify    | Command          | Displays the following information about the current disk:  
- Device name and device type  
- Number of cylinders, alternate cylinders, heads and sectors  
- Partition table |
| save      | Command          | VTOC – Saves new disk and partition information.  
EFI – Not applicable. |
| inquiry   | Command          | **SCSI disks only** – Displays the vendor, product name, and revision level of the current drive. |
| volname   | Command          | Labels the disk with a new eight-character volume name that you specify. |
| quit      | Command          | Exits the `format` menu. |
partition Menu

The partition menu appears similar to the following:

```
format> partition
PARTITION MENU:
  0 - change '0' partition
  1 - change '1' partition
  2 - change '2' partition
  3 - change '3' partition
  4 - change '4' partition
  5 - change '5' partition
  6 - change '6' partition
  7 - change '7' partition
  select - select a predefined table
  modify - modify a predefined partition table
  name - name the current table
  print - display the current table
  label - write partition map and label to the disk
  quit
partition>
```

The following table describes the partition menu items.

**TABLE 16-2 Descriptions for partition Menu Items**

<table>
<thead>
<tr>
<th>Subcommand</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>change 'n' partition</td>
<td>Enables you to specify the following information for the new partition:</td>
</tr>
<tr>
<td></td>
<td>■ Identification tag</td>
</tr>
<tr>
<td></td>
<td>■ Permission flags</td>
</tr>
<tr>
<td></td>
<td>■ Starting cylinder</td>
</tr>
<tr>
<td></td>
<td>■ Size</td>
</tr>
<tr>
<td>select</td>
<td>Enables you to choose a predefined partition table.</td>
</tr>
<tr>
<td>modify</td>
<td>Enables you to change all the slices in the partition table. This command</td>
</tr>
<tr>
<td></td>
<td>is preferred over the individual change 'x' partition commands.</td>
</tr>
<tr>
<td>name</td>
<td>Enables you to specify a name for the current partition table.</td>
</tr>
<tr>
<td>print</td>
<td>Displays the current partition table.</td>
</tr>
<tr>
<td>label</td>
<td>Writes the partition map and the label to the current disk.</td>
</tr>
<tr>
<td>quit</td>
<td>Exits the partition menu.</td>
</tr>
</tbody>
</table>

x86: fdisk Menu

The fdisk menu appears on x86 based systems only and appears similar to the following.
format> fdisk
    Total disk size is 14169 cylinders
    Cylinder size is 2510 (512 byte) blocks

<table>
<thead>
<tr>
<th>Partition</th>
<th>Status</th>
<th>Type</th>
<th>Start</th>
<th>End</th>
<th>Length</th>
<th>%</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Active</td>
<td>x86 Boot</td>
<td>1</td>
<td>9</td>
<td>9</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td></td>
<td>Solaris2</td>
<td>10</td>
<td>14159</td>
<td>1459</td>
<td>100</td>
</tr>
</tbody>
</table>

SELECT ONE OF THE FOLLOWING:

1. Create a partition
2. Specify the active partition
3. Delete a partition
4. Change between Solaris and Solaris2 Partition IDs
5. Exit (update disk configuration and exit)
6. Cancel (exit without updating disk configuration)

Enter Selection:

The following table describes the fdisk menu items.

TABLE 16-3  x86: Descriptions for fdisk Menu Items

<table>
<thead>
<tr>
<th>Menu Item</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Create a partition</td>
<td>Creates an fdisk partition. You must create a separate partition for each OS such as Oracle Solaris or DOS. There is a maximum of four partitions per disk. You are prompted for the size of the fdisk partition as a percentage of the disk.</td>
</tr>
<tr>
<td>Specify the active partition</td>
<td>Enables you to specify the partition to be used for booting. This menu item identifies where the first stage boot program looks for the second stage boot program.</td>
</tr>
<tr>
<td>Delete a partition</td>
<td>Deletes a previously created partition. This command destroys all the data in the partition.</td>
</tr>
<tr>
<td>Change between Solaris and Solaris2 Partition IDs</td>
<td>Changes partition IDs from 130 (0x82) to 191 (0xbf) and back again.</td>
</tr>
<tr>
<td>Exit (update disk configuration and exit)</td>
<td>Writes a new version of the partition table and exits the fdisk menu.</td>
</tr>
<tr>
<td>Cancel (exit without updating disk configuration)</td>
<td>Exits the fdisk menu without modifying the partition table.</td>
</tr>
</tbody>
</table>

analyze Menu

The analyze menu appears similar to the following.

format> analyze

ANALYZE MENU:
The following table describes the analyze menu items.

<table>
<thead>
<tr>
<th>Subcommand</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>read</td>
<td>Reads each sector on the current disk. Repairs defective blocks as a default.</td>
</tr>
<tr>
<td>refresh</td>
<td>Reads then writes data on the current disk without harming the data. Repairs defective blocks as a default.</td>
</tr>
<tr>
<td>test</td>
<td>Writes a set of patterns to the disk without harming the data. Repairs defective blocks as a default.</td>
</tr>
<tr>
<td>write</td>
<td>Writes a set of patterns to the disk then reads back the data on the disk. Destroys existing data on the disk. Repairs defective blocks as a default.</td>
</tr>
<tr>
<td>compare</td>
<td>Writes a set of patterns to the disk, reads back the data, and then compares it to the data in the write buffer. Destroys existing data on the disk. Repairs defective blocks as a default.</td>
</tr>
<tr>
<td>purge</td>
<td>Removes all data from the disk so that the data cannot be retrieved by any means. Data is removed by writing three distinct patterns over the entire disk (or a section of the disk). If the verification passes, a hex-bit pattern is written over the entire disk (or a section of the disk). Repairs defective blocks as a default.</td>
</tr>
<tr>
<td>verify</td>
<td>In the first pass, writes unique data to each block on the entire disk. In the next pass, reads and verifies the data. Destroys existing data on the disk. Repairs defective blocks as a default.</td>
</tr>
<tr>
<td>print</td>
<td>Displays the data in the read/write buffer.</td>
</tr>
</tbody>
</table>
TABLE 16–4  Descriptions for analyze Menu Items  (Continued)

<table>
<thead>
<tr>
<th>Subcommand</th>
<th>Description</th>
</tr>
</thead>
</table>
| setup      | Enables you to specify the following analysis parameters:  
  Analyze entire disk? yes  
  Starting block number: depends on drive  
  Ending block number: depends on drive  
  Loop continuously? no  
  Number of passes: 2  
  Repair defective blocks? yes  
  Stop after first error? no  
  Use random bit patterns? no  
  Number of blocks per transfer: 126 (0/n/m)  
  Verify media after formatting? yes  
  Enable extended messages? no  
  Restore defect list? yes  
  Restore disk label? yes |
| config     | Displays the current analysis parameters. |
| quit       | Exits the analyze menu. |

**defect Menu**

The defect menu appears similar to the following:

```
format> defect
DEFECT MENU:  
  primary - extract manufacturer's defect list  
  grown - extract manufacturer's and repaired defects list  
  both - extract both primary and grown defects list  
  print - display working list  
  dump - dump working list to file  
  quit  
```

defect>

The following table describes the defect menu items.

TABLE 16–5  The defect Menu Item Descriptions

<table>
<thead>
<tr>
<th>Subcommand</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>primary</td>
<td>Reads the manufacturer's defect list from the disk drive and updates the in-memory defect list.</td>
</tr>
<tr>
<td>grown</td>
<td>Reads the grown defect list and then updates the in-memory defect list. Grown defects are defects that have been detected during analysis.</td>
</tr>
<tr>
<td>both</td>
<td>Reads both the manufacturer's defect list and the grown defect list. Then, updates the in-memory defect list.</td>
</tr>
</tbody>
</table>
### Rules for Input to `format` Commands

When you use the `format` utility, you need to provide various kinds of information. This section describes the rules for this information. For information on using `format`'s help facility when you specify data, see “Getting Help on the `format` Utility” on page 311.

### Specifying Numbers to `format` Commands

Several places in the `format` utility require number as input. You must either specify the appropriate data or select a number from a list of choices. In either case, the help facility causes `format` to display the upper and lower limits of the number expected. Simply enter the appropriate number. The number is assumed to be in decimal format unless a base is explicitly specified as part of the number (for example, 0x for hexadecimal).

The following are examples of integer input:

```
Enter number of passes [2]: 34
Enter number of passes [34] 0xf
```

### Specifying `format` Command Names

Command names are needed as input whenever the `format` utility displays a menu prompt. You can abbreviate the command names, as long as what you type is sufficient to uniquely identify the command desired.

For example, use `p` to access the `partition` menu from the `format` menu. Then, type `p` to display the current slice table.

```
format> p
PARTITION MENU:
  0   - change ‘0’ partition
  1   - change ‘1’ partition
  2   - change ‘2’ partition
  3   - change ‘3’ partition
  4   - change ‘4’ partition
  5   - change ‘5’ partition
```
6 - change '6' partition
7 - change '7' partition
select - select a predefined table
modify - modify a predefined partition table
name - name the current table
print - display the current table
label - write partition map and label to the disk
quit

Specifying Disk Names to format Commands

At certain points in the format utility, you must name something. In these cases, you are free to specify any string you want for the name. If the name has white space in it, the entire name must be enclosed in double quotation marks ("). Otherwise, only the first word of the name is used.

For example, if you want to identify a specific partition table for a disk, you can use the name subcommand that is available from the partition menu:

    partition> name
    Enter table name (remember quotes): "new disk3"

Getting Help on the format Utility

The format utility provides a help facility that you can use whenever the format utility is expecting input. You can request help about what input is expected by typing a question mark (?). The format utility displays a brief description of what type of input is needed.

If you type a ? at a menu prompt, a list of available commands is displayed.

The man pages associated with the format utility include the following:

- **format(1M)** – Describes the basic format utility capabilities and provides descriptions of all command-line variables.
- **format.dat(4)** – Describes disk drive configuration information for the format utility.
Managing file systems is one of your most important system administration tasks.

This is a list of the overview information in this chapter.

- “What’s New in Oracle Solaris File Systems?” on page 313
- “Where to Find File System Management Tasks” on page 314
- “Overview of File Systems” on page 315
- “Types of Oracle Solaris File Systems” on page 315
- “Default Oracle Solaris File Systems” on page 320
- “Swap Space” on page 320
- “Overview of Mounting and Unmounting File Systems” on page 321
- “Determining a File System’s Type” on page 325

**What's New in Oracle Solaris File Systems?**

This section describes new file system features in the Oracle Solaris release.

- “File System Monitoring Tool (fsstat)” on page 313
- “Oracle Solaris ZFS File System” on page 314

**File System Monitoring Tool (fsstat)**

**Oracle Solaris 11:** A new file system monitoring tool, fsstat, is available to report file system operations. You can use several options to report activity, such as by mount point or by file system type.

For example, the following fsstat command displays all ZFS file system operations since the ZFS module was loaded:

```
$ fsstat zfs
new  name  name  attr  attr lookup  rddir  read  read  write  write
```

For example, the following `fsstat` command displays all file system operations since the `/export/ws` file system mounted.

```bash
$ fsstat /export/ws
```

The default form is to report statistical information in easy to understand values, such as GB, KB, and MB.

For more information, see `fsstat(1M)`.

### Oracle Solaris ZFS File System

**Oracle Solaris 11**: Oracle Solaris ZFS, a revolutionary new file system, provides simple administration, transactional semantics, end-to-end data integrity, and immense scalability. In addition, ZFS provides the following administration features:

- Device management support
- Persistent snapshots and cloning features
- Quotas that can be set for file systems
- ACL-based access control
- Storage pool space reservations for file systems
- Support for Oracle Solaris systems that have zones installed

For more information about using ZFS, see *Oracle Solaris Administration: ZFS File Systems*.

## Where to Find File System Management Tasks

Use these references to find step-by-step instructions for managing file systems.

<table>
<thead>
<tr>
<th>File System Management Task</th>
<th>For More Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Connect and configure new disk devices.</td>
<td>Chapter 10, &quot;Managing Disks (Overview)&quot;</td>
</tr>
<tr>
<td>Create and mount new file systems.</td>
<td>Chapter 18, &quot;Creating and Mounting File Systems (Tasks)&quot;</td>
</tr>
<tr>
<td>Make remote files available to users.</td>
<td>Chapter 5, &quot;Network File System Administration (Tasks),&quot; in <em>Oracle Solaris Administration: Network Services</em></td>
</tr>
</tbody>
</table>
Overview of File Systems

A file system is a structure of directories that is used to organize and store files. The term file system is used to describe the following:

- A particular type of file system: disk-based, network-based, or virtual
- The entire file tree, beginning with the root (/) directory
- The data structure of a disk slice or other media storage device
- A portion of a file tree structure that is attached to a mount point on the main file tree so that the files are accessible

Usually, you know from the context which meaning is intended.

The Oracle Solaris OS uses the virtual file system (VFS) architecture, which provides a standard interface for different file system types. The VFS architecture enables the kernel to handle basic operations, such as reading, writing, and listing files. The VFS architecture also makes it easier to add new file systems.

Types of Oracle Solaris File Systems

The Oracle Solaris OS supports three types of file systems:

- Disk-based
- Network-based
- Virtual

To identify the file system type, see "Determining a File System's Type" on page 325.

Oracle Solaris Disk-Based File Systems

Disk-based file systems are stored on physical media such as hard disks and DVDs. Disk-based file systems can be written in different formats. The available formats are described in the following table.

<table>
<thead>
<tr>
<th>Disk-Based File System</th>
<th>Format Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ZFS</td>
<td>ZFS is the default disk-based and root file system in the Oracle Solaris 11 release. For more information, see the Oracle Solaris Administration: ZFS File Systems.</td>
</tr>
<tr>
<td>UFS</td>
<td>Legacy UNIX file system (based on the BSD Fat Fast File system that was provided in the 4.3 Tahoe release).</td>
</tr>
</tbody>
</table>
Overview of File Systems

Each type of disk-based file system is customarily associated with a particular media device, as follows:

- ZFS or UFS with hard disk
- HSFS with CD-ROM
- PCFS with USB diskette
- UDF with DVD

However, these associations are not restrictive. For example, DVDs can have ZFS or UFS file systems created on them.

The Universal Disk Format (UDFS) File System

For information about creating a UDFS file system on removable media, see “How to Create a File System on Removable Media” on page 31.

The UDF file system is the industry-standard format for storing information on DVD (Digital Versatile Disc or Digital Video Disc) optical media.

The UDF file system is provided as dynamically loadable 32-bit and 64-bit modules, with system administration utilities for creating, mounting, and checking the file system on both SPARC and x86 platforms. The Oracle Solaris UDF file system works with supported ATAPI and SCSI DVD drives, CD-ROM devices, and disk drives. In addition, the Oracle Solaris UDF file system is fully compliant with the UDF 1.50 specification.

The UDF file system provides the following features:

- Ability to access the industry-standard CD-ROM and DVD-ROM media when they contain a UDF file system
- Flexibility in exchanging information across platforms and operating systems

<table>
<thead>
<tr>
<th>Disk-Based File System</th>
<th>Format Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>HSFS</td>
<td>High Sierra, Rock Ridge, and ISO 9660 file system. High Sierra is the first CD-ROM file system. ISO 9660 is the official standard version of the High Sierra file system. The HSFS file system is used on CD-ROMs, and is a read-only file system. Oracle Solaris HSFS supports Rock Ridge extensions to ISO 9660. When present on a CD-ROM, these extensions provide all file system features and file types, except for writability and hard links.</td>
</tr>
<tr>
<td>PCFS</td>
<td>PC file system, which allows read- and write- access to data and programs on DOS-formatted disks that are written for DOS-based personal computers.</td>
</tr>
<tr>
<td>UDFS</td>
<td>The Universal Disk Format (UDFS) file system, the industry-standard format for storing information on the optical media technology called DVD (Digital Versatile Disc or Digital Video Disc).</td>
</tr>
</tbody>
</table>
A mechanism for implementing new applications rich in broadcast-quality video, high-quality sound, and interactivity using the DVD video specification based on UDF format

The following features are not included in the UDF file system:

- Support for write-once media, (CD-RW), with either the sequential disk-at-once recording and incremental recording

The UDF file system requires the following:

- Supported SPARC or x86 platform
- Supported CD-ROM or DVD-ROM device

The Oracle Solaris UDF file system implementation provides the following:

- Support for industry-standard read/write UDF version 1.50
- Fully internationalized file system utilities

Network-Based File Systems

Network-based file systems can be accessed from the network. Typically, network-based file systems reside on one system, typically a server, and are accessed by other systems across the network.

With the NFS service, you can provide distributed resources (files or directories) by sharing them from a server and mounting them on individual clients. For more information, see “The NFS Environment” on page 323.

With the Oracle SMB service, you can provide distributed resources (files or directories) to Windows and Mac OS systems by sharing them from a server and mounting them on individual clients. For more information, see “The Oracle Solaris SMB Service” on page 325.

Virtual File Systems

Virtual file systems are memory-based file systems that provide access to special kernel information and facilities. Most virtual file systems do not use file system disk space. Also, some virtual file systems, such as the temporary file system (TMPFS), use the swap space on a disk.

Temporary File System

The temporary file system (TMPFS) uses local memory for file system reads and writes. Using TMPFS can improve system performance by saving the cost of reading and writing temporary files to a local disk or across the network. For example, temporary files are created when you compile a program. The OS generates a much disk activity or network activity while manipulating these files. Using TMPFS to hold these temporary files can significantly speed up their creation, manipulation, and deletion.
Files in TMPFS file systems are not permanent. These files are deleted when the file system is unmounted and when the system is shut down or rebooted.

TMPFS is the default file system type for the /tmp directory in the Oracle Solaris OS. You can copy or move files into or out of the /tmp directory, just as you would in a ZFS or UFS file system.

The TMPFS file system uses swap space as a temporary backing store. If a system with a TMPFS file system does not have adequate swap space, two problems can occur:

- The TMPFS file system can run out of space, just as regular file systems do.
- Because TMPFS allocates swap space to save file data (if necessary), some programs might not execute because of insufficient swap space.

For information about creating TMPFS file systems, see Chapter 18, "Creating and Mounting File Systems (Tasks)." For information about increasing swap space, see Chapter 19, “Configuring Additional Swap Space (Tasks).”

The Loopback File System

The loopback file system (LOFS) lets you create a new virtual file system so that you can access files by using an alternative path name. For example, you can create a loopback mount of the root (/) directory on /tmp/newroot. This loopback mounts make the entire file system hierarchy appear as if it is duplicated under /tmp/newroot, including any file systems mounted from NFS servers. All files will be accessible either with a path name starting from root (/), or with a path name that starts from /tmp/newroot.

For information on how to create LOFS file systems, see Chapter 18, “Creating and Mounting File Systems (Tasks).”

Process File System

The process file system (PROCFS) resides in memory and contains a list of active processes, by process number, in the /proc directory. Information in the /proc directory is used by commands such as ps. Debuggers and other development tools can also access the address space of the processes by using file system calls.

**Caution** – Do not delete files in the /proc directory. The deletion of processes from the /proc directory does not kill them. /proc files do not use disk space, so there is no reason to delete files from this directory.

The /proc directory does not require administration.
Additional Virtual File Systems

These additional types of virtual file systems are listed for your information. They do not require administration.

<table>
<thead>
<tr>
<th>Virtual File System</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CTFS (the contract file system)</td>
<td>CTFS (the contract file system) is the interface for creating, controlling, and observing contracts. A contract enhances the relationship between a process and the system resources it depends on by providing richer error reporting and (optionally) a means of delaying the removal of a resource. The service management facility (SMF) uses process contracts (a type of contract) to track the processes which compose a service, so that a failure in a part of a multi-process service can be identified as a failure of that service.</td>
</tr>
<tr>
<td>FIFOFS (first-in first-out)</td>
<td>Named pipe files that give processes common access to data</td>
</tr>
<tr>
<td>FDFS (file descriptors)</td>
<td>Provides explicit names for opening files by using file descriptors</td>
</tr>
<tr>
<td>MNTFS</td>
<td>Provides read-only access to the table of mounted file systems for the local system</td>
</tr>
<tr>
<td>NAMEFS</td>
<td>Used mostly by STREAMS for dynamic mounts of file descriptors on top of files</td>
</tr>
<tr>
<td>OBJFS</td>
<td>The OBJFS (object) file system describes the state of all modules currently loaded by the kernel. This file system is used by debuggers to access information about kernel symbols without having to access the kernel directly.</td>
</tr>
<tr>
<td>SHAREFS</td>
<td>Provides read-only access to the table of shared file systems for the local system</td>
</tr>
<tr>
<td>SPECFS (special)</td>
<td>Provides access to character special devices and block devices</td>
</tr>
<tr>
<td>SWAPFS</td>
<td>Used by the kernel for swapping</td>
</tr>
</tbody>
</table>

Extended File Attributes

The ZFS, UFS, NFS, and TMPFS file systems have been enhanced to include extended file attributes. Extended file attributes enable application developers to associate specific attributes to a file. For example, a developer of an application used to manage a windowing system might choose to associate a display icon with a file. Extended file attributes are logically represented as files within a hidden directory that is associated with the target file.

You can use the runat command to add attributes and execute shell commands in the extended attribute namespace. This namespace is a hidden attribute directory that is associated with the specified file.
To use the `runat` command to add attributes to a file, you first have to create the attributes file.

```
$ runat filea cp /tmp/attrdata attr.1
```

Then, use the `runat` command to list the attributes of the file.

```
$ runat filea ls -l
```

For more information, see the `runat(1)` man page.

Many Oracle Solaris file system commands have been modified to support file system attributes by providing an attribute-aware option. Use this option to query, copy, or find file attributes. For more information, see the specific man page for each file system command.

**Swap Space**

The Oracle Solaris OS uses some disk slices for temporary storage rather than for file systems. These slices are called `swap` slices, or `swap space`. Swap space is used for virtual memory storage areas when the system does not have enough physical memory to handle current processes.

Since many applications rely on swap space, you should know how to plan for, monitor, and add more swap space, when needed. For an overview about swap space and instructions for adding swap space, see Chapter 19, "Configuring Additional Swap Space (Tasks)."

### Default Oracle Solaris File Systems

The ZFS file system is hierarchical, starting with the root directory (`/`) and continuing downwards through a number of directories. The Oracle Solaris installation process enables you to install a default set of directories and uses a set of conventions to group similar types of files together.

For a brief overview of Oracle Solaris file systems and directories, see `filesystem(5)`.

The following table provides a summary of the default Oracle Solaris file systems.

<table>
<thead>
<tr>
<th>File System or Directory</th>
<th>File System Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>root (<code>/</code>)</td>
<td>ZFS</td>
<td>The top of the hierarchical file tree. The root (<code>/</code>) directory contains the directories and files that are critical for system operation, such as the kernel, the device drivers, and the programs used to boot the system. The root (<code>/</code>) directory also contains the mount point directories where local and remote file systems can be attached to the file tree.</td>
</tr>
</tbody>
</table>
### TABLE 17–1 The Default Oracle Solaris File Systems

<table>
<thead>
<tr>
<th>File System or Directory</th>
<th>File System Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>/usr</td>
<td>ZFS</td>
<td>System files and directories that can be shared with other users. Files that run only on certain types of systems are in the /usr file system or directory (for example, SPARC executables). Files that can be used on all types of systems, such as the man pages, might be placed in the /usr/share directory.</td>
</tr>
<tr>
<td>/export/home or /home</td>
<td>NFS or ZFS</td>
<td>The mount point for user home directories, which store user work files. By default, the /home directory is an automounted file system.</td>
</tr>
<tr>
<td>/var</td>
<td>ZFS</td>
<td>System files and directories that are likely to change or grow over the life of the local system. These include system logs, such as vi and ex backup files.</td>
</tr>
<tr>
<td>/opt</td>
<td>NFS or ZFS</td>
<td>Optional mount point for third-party software. On some systems, the /opt directory might be a UFS file system or ZFS file system.</td>
</tr>
<tr>
<td>/tmp</td>
<td>TMPFS</td>
<td>Temporary files, which are removed each time the system is booted or the /tmp file system is unmounted.</td>
</tr>
<tr>
<td>/proc</td>
<td>PROCFS</td>
<td>A list of active processes, by process number.</td>
</tr>
<tr>
<td>/etc/mnttab</td>
<td>MNTFS</td>
<td>A virtual file system that provides read-only access to the table of mounted file systems for the local system.</td>
</tr>
<tr>
<td>/system/volatile</td>
<td>TMPFS</td>
<td>A memory-based file system for storing temporary files that are not needed after the system is booted.</td>
</tr>
<tr>
<td>/system/contract</td>
<td>CTFS</td>
<td>A virtual file system that maintains contract information.</td>
</tr>
<tr>
<td>/system/object</td>
<td>OBJFS</td>
<td>A virtual file system that is used by debuggers to access information about kernel symbols without having to access the kernel directly.</td>
</tr>
</tbody>
</table>

### Overview of Mounting and Unmounting File Systems

Before you can access the files on a file system, you need to mount the file system. When you mount a file system, you attach that file system to a directory (mount point) and make it available to the system. The root (/) file system is always mounted. Any other file system can be connected or disconnected from the root (/) file system.

Most file systems are automatically mounted by SMF services at system boot time. Generally, you do not need to mount or unmount file systems manually. For more information about mounting different file system types, see "Mounting and Unmounting Oracle Solaris File Systems" on page 328.
When you mount a file system, any files or directories in the underlying mount point directory are unavailable as long as the file system is mounted. These files are not permanently affected by the mounting process. They become available again when the file system is unmounted. However, mount directories are typically empty because you usually do not want to obscure existing files.

For step-by-step instructions on how to mount file systems, see "Mounting and Unmounting Oracle Solaris File Systems" on page 328.

The Mounted File System Table

Whenever you mount or unmount a file system, the /etc/mnttab (mount table) file is modified with the list of currently mounted file systems. You can display the contents of this file by using the cat or more commands. However, you cannot edit this file. Here is an example of an /etc/mnttab file:

```
$ more /etc/mnttab
rpool/ROOT/zfsBE / zfs dev=40d0002 0
/devices /devices devfs dev=9500000 1307562777
/dev /dev dev=9580000 1307562777
cf5 /system/contract cf5 dev=95c0001 1307562777
swap /system/volatile tmpfs xattr,dev=9640001 1307562777
objfs /system/object objfs dev=9600001 1307562777
sharefs /etc/dfs/sharetab sharefs dev=96c0001 1307562777
fd /dev/fd fd rw,dev=97c0001 1307562811
rpool/export /export zfs rw,devices,setuid,nonbmand,exec,rstchown,
xattr,atime,dev=40d0003 1307562818
rpool/export/home /export/home zfs rw,devices,setuid,nonbmand,
exect,chown,xattr,atime,dev=40d0004 1307562818
```

The Virtual File System Table

Most file systems are mounted automatically by an SMF service at system boot time.

You might need to edit the /etc/vfstab file to mount legacy file systems or to make changes to the ZFS swap volume. For information about changing a ZFS swap volume, see Chapter 19, “Configuring Additional Swap Space (Tasks).”

To add an entry for mounting a legacy file system, the information you need to specify is as follows:

- The device where the file system resides
- The file system mount point
- File system type
Whether you want the file system to mount automatically when the system boots (by using the mountall command)

Any mount options

The following `vfstab` example if from a system that runs a ZFS root file system.

```bash
# cat /etc/vfstab
#device device mount FS fsck mount mount
#to mount to fsck point type pass at boot options
#
fd - /dev/fd fd - no -
/proc - /proc proc - no -
/dev/zvol/dsk/rpool/swap - /devices swap - no -
sharefs - /etc/dfs/sharetabsharefs - no -
ctfs - /system/contract ctfs - no -
objfs - /system/object objfs - no -
swap - /tmp tmpfs - yes -
```

ZFS file systems are mounted automatically by the SMF service at boot time. You can mount ZFS file systems from the `vfstab` by using the legacy mount feature. For more information, see Oracle Solaris Administration: ZFS File Systems.

For descriptions of each `/etc/vfstab` field and information on how to edit and use the file, see `vfstab(4).

**The NFS Environment**

NFS is a distributed file system service that can be used to share resources (files or directories) from one system, typically a server, with other systems on the network. For example, you might want to share third-party applications or source files with users on other systems.

NFS makes the actual physical location of the resource irrelevant to the user. Instead of placing copies of commonly used files on every system, NFS allows you to place one copy on one system's disk and let all other systems access it from the network. Under NFS, remote files are virtually indistinguishable from local files.

For more information, see Chapter 4, “Managing Network File Systems (Overview),” in Oracle Solaris Administration: Network Services.

A system becomes an NFS server if it has resources to share on the network. A server keeps a list of currently shared resources and their access restrictions (such as read/write or read-only access).

When you share a resource, you make it available for mounting by remote systems.

You can share a resource in these ways:

- Create a ZFS share by using the ZFS share property.
Create a legacy share by using the `share` command.

For a complete description of NFS, see Chapter 4, "Managing Network File Systems (Overview)," in Oracle Solaris Administration: Network Services.

**NFS Version 4**

Oracle's implementation of the NFS version 4 distributed file access protocol is included in the Oracle Solaris release.

NFS version 4 integrates file access, file locking, and mount protocols into a single, unified protocol to ease traversal through a firewall and improve security. The Oracle Solaris implementation of NFS version 4 is fully integrated with Kerberos V5, also known as SEAM, thus providing authentication, integrity, and privacy. NFS version 4 also enables the negotiation of security flavors to be used between the client and the server. With NFS version 4, a server can offer different security flavors for different file systems.

For more information about NFS Version 4 features, see "What's New With the NFS Service" in Oracle Solaris Administration: Network Services.

**Automounting (autofs)**

You can mount NFS file system resources by using a client-side service called automounting (or autofs). The autofs service enables a system to automatically mount and unmount NFS resources whenever you access them. The resource remains mounted as long as you remain in the directory and are using a file within that directory. If the resource is not accessed for a certain period of time, it is automatically unmounted.

The autofs service provides the following features:

- NFS resources don't need to be mounted when the system boots, which saves booting time.
- Users don't need to know the root password to mount and unmount NFS resources.
- Network traffic might be reduced because NFS resources are mounted only when they are in use.

The autofs service is initialized by the `autofs` utility, which runs automatically when a system is booted. The `autofs` daemon runs continuously and is responsible for the mounting and unmounting of NFS file systems on an as-needed basis. By default, the `/home` file system is mounted by the `autofs` daemon.

With autofs, you can specify multiple servers to provide the same file system. This way, if one of these servers is down, autofs can try to mount the file system from another machine.

For complete information on how to set up and administer autofs, see Chapter 5, "Network File System Administration (Tasks)," in Oracle Solaris Administration: Network Services.
The Oracle Solaris SMB Service

The Oracle Solaris OS provides a Server Message Block (SMB) protocol server and client implementation that includes support for numerous SMB dialects including NTLM 0.12 and Common Internet File System (CIFS). The terms CIFS and SMB can be considered interchangeable.

The Solaris SMB server allows a native Oracle Solaris system to serve files as SMB shares to SMB enabled clients that mount the file system shares. A Windows, Mac OS, or Solaris client can interoperate with the Solaris SMB server as it would with a Windows server. A Solaris SMB server can operate in either workgroup mode or in domain mode. In workgroup mode, the Solaris SMB server is responsible for authenticating users locally when access is requested to shared resources. This authentication process is referred to as local login. In domain mode, the Solaris SMB server uses pass-through authentication, in which user authentication is delegated to a domain controller.

For more information, see Oracle Solaris Administration: SMB and Windows Interoperability.

Determining a File System's Type

You can determine a file system's type by using one of the following:

- The `FS type` field in the virtual file system table (the `/etc/vfstab` file)
- The `/etc/default/fs` file for local file systems
- The `/etc/dfs/fstypes` file for NFS file systems

How to Determine a File System's Type

These commands work whether or not the file system is mounted.

If you have the raw device name of a disk slice, you can use the `fstyp` command to determine a file system's type (if the disk slice contains a file system). For more information, see `fstyp(1M)`.

**EXAMPLE 17-1 Determining a File System's Type**

The following example uses the `fstyp` command to determine the file system type.

```
# fstyp /dev/rdsk/c0t0d0s0
zfs
```

The following example uses the `/etc/vfstab` file to determine the file system type for the `/legacy` file system.

```
$ grep /legacy /etc/vfstab
/dev/dsk/c0t3d0s6 /dev/rdsk/c0t3d0s6 /legacy ufs 2 yes -
```
This chapter describes how to create and mount ZFS, temporary (TMPFS), and loopback (LOFS) file systems. Because TMPFS and LOFS are virtual file systems, you actually “access” them by mounting them. In addition, creating and mounting a legacy UFS file system is also covered.

This is a list of the step-by-step instructions in this chapter.

- “How to Create an ZFS File System” on page 332
- “How to Create and Mount a Legacy UFS File System” on page 333
- “How to Create and Mount a TMPFS File System” on page 334
- “How to Create and Mount an LOFS File System” on page 335

Creating Oracle Solaris File Systems

This section provides an overview of creating Oracle Solaris file systems.

Creating ZFS File Systems

A ZFS file system is not directly tied to a specific disk partition. A ZFS file system is contained with a ZFS storage pool that can contain many devices. Either whole disks or disk slices can be added to a ZFS storage pool. Within a pool, you will probably want to create additional file systems. File systems provide points of administration that allow you to manage different sets of data within the same pool.

A ZFS file system is created by using the `zfs create` command. A ZFS file system is automatically mounted when it is created. For more information, see “How to Create an ZFS File System” on page 332.

For more information about creating ZFS storage pools and file systems, see Oracle Solaris Administration: ZFS File Systems.
Creating a Temporary File System

A temporary file system (TMPFS) uses local memory for file system reads and writes. TMPFS file systems can improve system performance by saving the cost of reading and writing temporary files to a local disk or across the network. Files in TMPFS file systems do not survive across reboots or unmounts.

If you create multiple TMPFS file systems, be aware that they all use the same system resources. Files created under one TMPFS file system use up space available for any other TMPFS file system, unless you limit TMPFS sizes by using the -o size option of the mount command.

For more information about creating and mounting a TMPFS file system, see “How to Create and Mount a TMPFS File System” on page 334.

For general information, see the tmpfs(7FS).

Creating a LOFS File System

An LOFS file system is a virtual file system that provides an alternate path to an existing file system. When other file systems are mounted onto an LOFS file system, the original file system does not change.

For more information about creating and mounting a LOFS file system, see “How to Create and Mount an LOFS File System” on page 335.

For general information, see the lofs(7FS).

Note – Be careful when creating LOFS file systems. Because LOFS file systems are virtual file systems, the potential for confusing both users and applications is enormous.

Mounting and Unmounting Oracle Solaris File Systems

ZFS file systems are mounted and unmounted automatically. You can make a legacy UFS file system available by mounting it, which attaches the file system to the system directory tree at the specified mount point. The root (/) file system is always mounted.

The following table provides guidelines on mounting file systems based on how you use them.

<table>
<thead>
<tr>
<th>Mount Type Needed</th>
<th>Suggested Mount Method</th>
</tr>
</thead>
<tbody>
<tr>
<td>Local or remote file systems that need to be mounted infrequently.</td>
<td>The mount command that you type manually from the command line.</td>
</tr>
</tbody>
</table>
Mounting and Unmounting Oracle Solaris File Systems

<table>
<thead>
<tr>
<th>Mount Type Needed</th>
<th>Suggested Mount Method</th>
</tr>
</thead>
<tbody>
<tr>
<td>Local legacy UFS file systems that need to be mounted frequently. Local ZFS file</td>
<td>The <code>/etc/vfstab</code> file, which mounts the file system automatically when the system is</td>
</tr>
<tr>
<td>systems are automatically mounted by an SMF service.</td>
<td>booted in multiuser state.</td>
</tr>
<tr>
<td>Remote legacy UFS file systems, such as home directories, that need to be</td>
<td>■ The <code>/etc/vfstab</code> file, which automatically mounts the file system when the system is</td>
</tr>
<tr>
<td>mounted frequently.</td>
<td>booted in multiuser state.</td>
</tr>
<tr>
<td></td>
<td>■ <code>autofs</code>, which automatically mounts the file system when you access it or unmounts</td>
</tr>
<tr>
<td></td>
<td>the file system when you change to another directory.</td>
</tr>
</tbody>
</table>

For more information on mounting removable media, see Chapter 1, "Managing Removable Media (Overview)."

You can determine which file systems are already mounted by using the `mount` command.

```
$ mount [-v ]
```

The `-v` displays the list of mounted file systems in verbose mode.

**EXAMPLE 18-1 Determining Which File Systems Are Mounted**

This example shows how to use the `mount` command to display information about the file systems that are currently mounted.

```
$ mount
/r on rpool/ROOT/zfsBE read/write/setuid/devices/rstchown/dev=40d0002 on Wed ...
/devices on /devices read/write/setuid/devices/rstchown/dev=9500000 on Wed ...
/dev on /dev read/write/setuid/devices/rstchown/dev=9500000 on Wed Jun ...
/system/contract on ctrfs read/write/setuid/devices/rstchown/dev=95c80001 ...
/proc on proc read/write/setuid/devices/rstchown/dev=9540000 on Wed Jun ...
/etc/mnttab on mnttab read/write/setuid/devices/rstchown/dev=9600001 on Wed ...
/system/volatile on swap read/write/setuid/devices/rstchown/xattr/dev=96400001 ...
/system/object on objfs read/write/setuid/devices/rstchown/dev=9680000 on Wed ...
/etc/dfs/sharetab on sharefs read/write/setuid/devices/rstchown/dev=96c0001 on ...
/dev/fd on fd read/write/setuid/devices/rstchown/dev=97c0001 on Wed Jun 8 ...
/tmp on swap read/write/setuid/devices/rstchown/xattr/dev=9640002 on Wed Jun ...
/export on rpool/export read/write/setuid/devices/rstchown/nonbmand/exec/xattr/
/export/home on rpool/export/home read/write/setuid/devices/rstchown
/rpool on rpool read/write/setuid/devices/rstchown/nonbmand/exec/
/home/rimmer on pluto:/export/home/rimmer remote/read/write/setuid/xattr/...
```

This example shows how to use the `zfs mount` command to display information about ZFS file systems that are currently mounted.

```
$ zfs mount
rpool/ROOT/solaris
rpool/ROOT/solaris/var
rpool               /rpool
```

Chapter 18 • Creating and Mounting File Systems (Tasks) 329
Determining Which File Systems Are Mounted (Continued)

<table>
<thead>
<tr>
<th>Fields</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>rpool/export</td>
<td>/rpool/export</td>
</tr>
<tr>
<td>rpool/export/home</td>
<td>/rpool/export/home</td>
</tr>
<tr>
<td>rpool/export/home/admin</td>
<td>/rpool/export/home/admin</td>
</tr>
</tbody>
</table>

Field Descriptions for the /etc/vfstab File

An entry in the /etc/vfstab file has seven fields, which are described in the following table.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>device to mount</td>
<td>This field identifies one of the following:</td>
</tr>
<tr>
<td></td>
<td>• The block device name for a local legacy UFS file system (for example, /dev/dsk/c8t1d0s7).</td>
</tr>
<tr>
<td></td>
<td>• The resource name for a remote file system (for example, myserver:/export/home). For more information about NFS, see Oracle Solaris Administration: IP Services.</td>
</tr>
<tr>
<td></td>
<td>• The block device name of the slice on which to swap (for example, /dev/dsk/c8t1d0s1).</td>
</tr>
<tr>
<td></td>
<td>• A directory for a virtual file system.</td>
</tr>
<tr>
<td>device to fsck</td>
<td>The raw (character) device name that corresponds to the legacy UFS file system identified by the device to mount field (for example, /dev/dsk/c8t1d0s7). This field determines the raw interface that is used by the fsck command. Use a dash (-) when there is no applicable device, such as for a read-only file system or a remote file system.</td>
</tr>
<tr>
<td>mount point</td>
<td>Identifies where to mount the file system (for example, /usr).</td>
</tr>
<tr>
<td>FS type</td>
<td>Identifies the type of file system.</td>
</tr>
</tbody>
</table>
### TABLE 18–1
Field Descriptions for the /etc/vfstab File

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>fsck pass</strong></td>
<td>The pass number used by the fsck command to decide whether to check a legacy UFS file system. When the field contains a dash (-), the file system is not checked. Currently, the fsck pass value in the /etc/vfstab file is ignored during the boot process. When the field contains a zero, legacy UFS file systems are not checked. When the field contains a value greater than zero, the UFS file system is always checked. All legacy UFS file systems with a value of 1 in this field are checked one at a time in the order they appear in the vfstab file. When the fsck command is run on multiple UFS file systems that have fsck pass values greater than 1 and the preen option (-o p) is used, the fsck command automatically checks the file systems on different disks in parallel to maximize efficiency. Otherwise, the value of the pass number does not have any effect.</td>
</tr>
<tr>
<td><strong>mount at boot</strong></td>
<td>Set to yes or no for whether the file system should be automatically mounted by the mountall command when the system is booted. Note that this field has nothing to do with autoofs. The root (/), /usr and /var file systems are not mounted from the vfstab file initially. This field should always be set to no for these file systems and for virtual file systems such as /proc and /dev/fd.</td>
</tr>
<tr>
<td><strong>mount options</strong></td>
<td>A list of comma-separated options (with no spaces) that are used for mounting the file system. Use a dash (-) to indicate no options. For more information, see vfstab(4).</td>
</tr>
</tbody>
</table>

**Note** – You must have an entry in each field in the /etc/vfstab file. If there is no value for a field, be sure to specify a dash (-). Otherwise, the system might not boot successfully. Similarly, white space should not be used as a field value.

### Prerequisites for Unmounting Oracle Solaris File Systems

The prerequisites for unmounting file systems include the following:

- You must be an administrator or assume an equivalent role.
- You cannot unmount a file system that is busy. A file system is considered busy if a user is accessing a directory in the file system, if a program has a file open in that file system, or if the file system is being shared. You can make a file system available for unmounting by doing the following:
  - Changing to a directory in a different file system.
  - Logging out of the system.
Using the `fuser` command to list all processes that are accessing the file system and to stop them, if necessary. For more details, see “How to Stop All Processes Accessing a File System” on page 340.

Notify users if you need to unmount a file system that they are using.

- Unsharing the file system. For information about unsharing a file system, see `unshare(1M)`.

To verify that you unmounted a file system or a number of file systems, examine the output from the `mount` command.

```
$ mount | grep unmounted-file-system
```

Creating and Mounting Oracle Solaris File Systems

This section provides examples of creating and mounting Oracle Solaris file systems.

### How to Create an ZFS File System

1. **Become an administrator.**
   
   For more information, see "How to Use Your Assigned Administrative Rights" in Oracle Solaris Administration: Security Services.

2. **Create a ZFS storage pool.**
   
   The following example illustrates how to create a simple mirrored storage pool named `tank` and a ZFS file system named `tank` in one command. Assume that the whole disks `/dev/dsk/c1t0d0` and `/dev/dsk/c2t0d0` are available for use.
   
   ```
   # zpool create tank mirror c1t0d0 c2t0d0
   ```

3. **Create a ZFS file system.**
   
   ```
   # zfs create tank/fs
   ```

   The new ZFS file system, `tank/fs`, can use as much of the disk space as needed, and is automatically mounted at `/tank/fs`.

4. **Confirm that the file system is created.**

   ```
   # zfs list -r tank
   NAME     USED   AVAIL  REFER  MOUNTPOINT
   tank     117K   268G   21K    /tank
   tank/fs  21K    268G   21K    /tank/fs
   ```
How to Create and Mount a Legacy UFS File System

Before You Begin
Ensure that you have met the following prerequisites:

- The disk must be formatted and divided into slices.
- If you are re-creating an existing legacy UFS file system, unmount it.
- You need to know the device name of the slice that will contain the file system.

For information on finding disks and disk slice numbers, see Chapter 11, “Administering Disks (Tasks).”

For information on formatting disks and dividing disks into slices, see Chapter 10, “Managing Disks (Overview).”

1 Become an administrator.
For more information, see “How to Use Your Assigned Administrative Rights” in Oracle Solaris Administration: Security Services.

2 Create a legacy UFS file system.

```bash
# newfs [-N] [-b size] [-i bytes] /dev/rdsk/device-name
```

The system asks for confirmation.

Caution – Be sure you have specified the correct device name for the slice before performing this step. If you specify the wrong slice, you will erase its contents when the new file system is created. This error might cause the system to panic.

3 To verify the creation of the legacy UFS file system, check the new file system.

```bash
# fsck /dev/rdsk/device-name
```

where device-name argument specifies the name of the disk device that contains the new file system.

The fsck command checks the consistency of the new file system, reports any problems, and prompts you before it repairs the problems. For more information on the fsck command, see fsck(1M).

4 Mount the legacy UFS file system.

```bash
# mkdir /directory-name
# mount /dev/dsk/device-name /directory-name
```

Example 18–2
Creating and Mounting a Legacy UFS File System

The following example shows how to create and mount a UFS file system /dev/rdsk/c0t1d0s0 on /legacy.

Creating and Mounting Oracle Solaris File Systems
Creating and Mounting Oracle Solaris File Systems

```
# newfs /dev/rdsk/c0t1d0s0
newfs: construct a new file system /dev/rdsk/c0t1d0s0: (y/n)? y
/dev/rdsk/c0t1d0s0: 286722656 sectors in 46668 cylinders of 48 tracks, 128 sectors
140001.3MB in 2917 cyl groups (16 c/g, 48.00MB/g, 5624 i/g)
super-block backups (for fsck -F ufs -o b=#) at:
   32, 98464, 196906, 295328, 393760, 492192, 590624, 689056, 787488, 885920,
Initializing cylinder groups:
..........................................................
super-block backups for last 10 cylinder groups at:
285773216, 285871648, 285970080, 286068512, 286166944, 286265376, 286363808,
286462240, 286560672, 286659104
# fsck /dev/rdsk/c0t1d0s0
# mkdir /legacy
# mount /dev/dsk/c0t1d0s0 /legacy
```

More Information: After You Create a Legacy UFS File System ...

To mount the legacy UFS file system automatically at boot time, go to “How to Add an Entry to the /etc/vfstab File” on page 336.

▶ How to Create and Mount a TMPFS File System

1 Become an administrator.
For more information, see “How to Use Your Assigned Administrative Rights” in Oracle Solaris Administration: Security Services.

2 Create the directory that you want to mount as the TMPFS file system, if necessary.
   
   # mkdir /mount-point
   
   where `mount-point` is the directory on which the TMPFS file system is mounted.

3 Mount the TMPFS file system.
   
   # mount -F tmpfs [-o size=number] swap mount-point
   -o size=number Specifies the size limit of the TMPFS file system in MB.
   mount-point Specifies the directory on which the TMPFS file system is mounted.

   To set up the system to automatically mount a TMPFS file system at boot time, see Example 18–4.

4 Verify that the TMPFS file system has been created.
   
   # mount -v

Example 18–3 Creating and Mounting a TMPFS File System

The following example shows how to create, mount, and limit the size of the TMPFS file system, /export/reports, to 50 MB.

```
Example 18–3
```
Creating and Mounting Oracle Solaris File Systems

Example 18–4  Mounting a TMPFS File System at Boot Time

You can set up the system to automatically mount a TMPFS file system at boot time by adding an /etc/vfstab entry. The following example shows an entry in the /etc/vfstab file that mounts /export/test as a TMPFS file system at boot time. Because the size=number option is not specified, the size of the TMPFS file system on /export/test is limited only by the available system resources.

Example 18–4  Mounting a TMPFS File System at Boot Time

```
swap - /export/test tmpfs - yes -
```

▼ How to Create and Mount an LOFS File System

1 Become an administrator.
For more information, see “How to Use Your Assigned Administrative Rights” in Oracle Solaris Administration: Security Services.

2 Create the directory you want to mount as an LOFS file system, if necessary.
```
# mkdir loopback-directory
```

3 Grant the appropriate permissions and ownership on the newly created directory.

4 Create the mount point where you want to mount the LOFS file system, if necessary.
```
# mkdir /mount-point
```

5 Mount the LOFS file system.
```
# mount -F lofs loopback-directory /mount-point
```
`loopback-directory`  Specifies the file system to be mounted on the loopback mount point.
`/mount-point`  Specifies the directory on which to mount the LOFS file system.

6 Verify that the LOFS file system has been mounted.
```
# mount -v
```

Example 18–5  Creating and Mounting an LOFS File System

The following example shows how to create, mount, and test new software in the /new/dist directory as a loopback file system without actually having to install it.
Mounting an LOFS File System at Boot Time

You can set up the system to automatically mount an LOFS file system at boot time by adding an entry to the end of the /etc/vfstab file. The following example shows an entry in the /etc/vfstab file that mounts an LOFS file system for the root (/) file system on /tmp/newroot.

```
/ - /tmp/newroot lofs - yes -
```

Ensure that the loopback entries are the last entries in the /etc/vfstab file. Otherwise, if the /etc/vfstab entry for a loopback file system precedes the file systems to be included in it, the loopback file system cannot be mounted.

▼ How to Add an Entry to the /etc/vfstab File

Use this procedure to mount non-ZFS file systems at boot time unless legacy mount behavior is needed for some ZFS file systems. For more information about mounting ZFS file systems, see Oracle Solaris Administration: ZFS File Systems.

1 Become an administrator.

For more information, see "How to Use Your Assigned Administrative Rights" in Oracle Solaris Administration: Security Services.

2 Create a mount point for the file system to be mounted, if necessary.

```
# mkdir /mount-point
```

There must be a mount point on the local system to mount a file system. A mount point is a directory to which the mounted file system is attached.

3 Edit the /etc/vfstab file and add an entry. Ensure that you do the following:

a. Separate each field with white space (a space or a tab).

b. Specify a dash (-) if a field has no contents.

c. Save the changes.

---

Note – Because the root (/) file system is mounted read-only by the kernel during the boot process, only the remount option (and options that can be used in conjunction with remount) affect the root (/) entry in the /etc/vfstab file.
Adding an Entry to the /etc/vfstab File

The following example shows how to mount the disk slice /dev/dsk/c0t3d0s7 as a legacy UFS file system to the mount point /files1. The raw character device /dev/rdsk/c0t3d0s7 is specified as the device to fsck. The fsck pass value of 2 means that the file system will be checked, but not sequentially.

Example 18–7

<table>
<thead>
<tr>
<th>#device</th>
<th>device</th>
<th>mount</th>
<th>FS</th>
<th>fsck</th>
<th>mount</th>
<th>mount</th>
<th>mount</th>
</tr>
</thead>
<tbody>
<tr>
<td>/dev/dsk/c0t3d0s7</td>
<td>/dev/rdsk/c0t3d0s7</td>
<td>/files1</td>
<td>ufs</td>
<td>2</td>
<td>yes</td>
<td>-</td>
<td></td>
</tr>
</tbody>
</table>

The following example shows how to mount the /export/man directory from the system pluto as an NFS file system on mount point /usr/man. Neither a device to fsck nor a fsck pass is specified because it’s an NFS file system. In this example, mount options are ro (read-only) and soft.

Example 18–8

<table>
<thead>
<tr>
<th>#device</th>
<th>device</th>
<th>mount</th>
<th>FS</th>
<th>fsck</th>
<th>mount</th>
<th>mount</th>
<th>mount</th>
</tr>
</thead>
<tbody>
<tr>
<td>pluto:/export/man</td>
<td>-</td>
<td>/usr/man</td>
<td>nfs</td>
<td>-</td>
<td>yes</td>
<td>ro,soft</td>
<td></td>
</tr>
</tbody>
</table>

How to Mount a File System (/etc/vfstab File)

1. Become an administrator.

For more information, see “How to Use Your Assigned Administrative Rights” in Oracle Solaris Administration: Security Services.

2. Mount a file system listed in the /etc/vfstab file.

   ```bash
   # mount /mount-point
   ```

   where /mount-point specifies an entry in the mount point or device to mount field in the /etc/vfstab file. It is usually easier to specify the mount point.

Example 18–8 Mounting a File System (/etc/vfstab File)

The following example shows how to mount the /legacy file system that is listed in the /etc/vfstab file.

   ```bash
   # mount /legacy
   ```

Example 18–9 Mounting All File Systems (/etc/vfstab File)

The following example shows the messages that are displayed when you use the mountall command and the file systems are already mounted.
The following example shows how to mount all the local systems that are listed in the /etc/vfstab file.

```bash
# mountall
mount: /tmp is already mounted or swap is busy
```

The following examples show how to mount all available ZFS file systems.

```bash
# zfs mount -a
```

The following example shows how to mount all the remote file systems that are listed in the /etc/vfstab file.

```bash
# mountall -r
```

### How to Mount an NFS File System (mount Command)

1. **Become an administrator.**
   For more information, see “How to Use Your Assigned Administrative Rights” in Oracle Solaris Administration: Security Services.

2. **Create a mount point for the file system to be mounted, if necessary.**
   ```bash
   # mkdir /mount-point
   ```
   There must be a mount point on the local system to mount a file system. A mount point is a directory to which the mounted file system is attached.

3. **Ensure that the resource (file or directory) is available from a server.**
   To mount an NFS file system, the resource must be made available on the server by using the share command. For information on how to share resources, see “About the NFS Service” in Oracle Solaris Administration: Network Services.

4. **Mount the NFS file system.**
   ```bash
   # mount -F nfs [-o mount-options] server:/directory /mount-point
   ```

**Example 18–10** Mounting an NFS File System (mount Command)

The following example shows how to mount the /export/packages directory on /mnt from the server pluto.

```bash
# mount -F nfs pluto:/export/packages /mnt
```
x86: How to Mount a PCFS (DOS) File System From a Hard Disk (mount Command)

Use the following procedure to mount a PCFS (DOS) file system from a hard disk.

1. Become an administrator.
   For more information, see “How to Use Your Assigned Administrative Rights” in Oracle Solaris Administration: Security Services.

2. Create a mount point for the file system to be mounted, if necessary.
   
   ```
   # mkdir /mount-point
   ```
   
   There must be a mount point on the local system to mount a file system. A mount point is a directory to which the mounted file system is attached.

3. Mount the PCFS file system.
   
   ```
   # mount -F pcfs [-o rw | ro] /dev/dsk/device-name:logical-drive /mount-point
   ```
   
   Specifies that you can mount a PCFS file system read/write (rw) or read-only (ro). If you do not specify this option, the default is rw.

   `/dev/dsk/device-name` Specifies the device name of the whole disk (for example, `/dev/dsk/c0t0d0p0`).

   `logical-drive` Specifies either the DOS logical drive letter (c through z) or a drive number (1 through 24). Drive c is equivalent to drive 1 and represents the primary DOS slice on the drive. All other letters or numbers represent DOS logical drives within the extended DOS slice.

   `/mount-point` Specifies the directory on which to mount the file system.

   Note that the `device-name` and `logical-drive` must be separated by a colon.

Example 18–11 x86: Mounting a PCFS (DOS) File System From a Hard Disk (mount Command)

The following example shows how to mount the logical drive in the primary DOS slice on the `/pcfs/c` directory.

```
# mount -F pcfs /dev/dsk/c0t0d0p0:c /pcfs/c
```

The following example shows how to mount read-only the first logical drive in the extended DOS slice on the `/mnt` directory.

```
# mount -F pcfs -o ro /dev/dsk/c0t0d0p0:2 /mnt
```
How to Stop All Processes Accessing a File System

1 Become an administrator.
   For more information, see “How to Use Your Assigned Administrative Rights” in Oracle Solaris Administration: Security Services.

2 List all the processes that are accessing the file system so that you know which processes you are going to stop.
   
   ```
   # fuser -c [ -u ] /mount-point
   -c Reports on files that are mount points for file systems and any files within those mounted file systems.
   -u Displays the user login name for each process ID.
   /mount-point Specifies the name of the file system for which you want to stop processes.
   ```

3 Stop all processes that are accessing the file system.
   
   ```
   # fuser -c -k /mount-point
   A SIGKILL is sent to each process that is using the file system.
   ```

   Note – You should not stop a user’s processes without first warning the user.

4 Verify that no processes are accessing the file system.
   
   ```
   # fuser -c /mount-point
   ```

Example 18–12 Stopping All Processes Accessing a File System

The following example shows how to stop process 4006c that is using the /export/home file system.

   ```
   # fuser -c /export/home
   /export/home: 4006c
   # fuser -c -k /export/home
   /export/home: 4006c
   # fuser -c /export/home
   /export/home:
   ```

How to Unmount a File System

Use the following procedure to unmount a file system.
1. Ensure that you have met the prerequisites listed in “Prerequisites for Unmounting Oracle Solaris File Systems” on page 331.

2. Unmount the file system.

   ```bash
   # umount /mount-point
   ```

   where `/mount-point` is the name of the file system that you want to unmount. This can be one of the following:
   - The directory name where the file system is mounted
   - The device name path of the file system
   - The resource for an NFS file system
   - The loopback directory for an LOFS file system

**Example 18–13 Unmounting a File System**

The following example shows how to unmount a legacy UFS file system.

   ```bash
   # umount /Legacy
   ```

The following example shows how to forcibly unmount the UFS `/Legacy` file system.

   ```bash
   # umount -f /Legacy
   #
   ```

The following example shows to unmount all ZFS file systems.

   ```bash
   # zfs umount -a
   ```

All file systems are unmounted, except for those file systems that are busy.
This chapter provides guidelines and step-by-step instructions for configuring additional swap space for a ZFS root file system after the Oracle Solaris OS is installed.

This is a list of the topics in this chapter.

- “About Swap Space” on page 343
- “How Do I Know If I Need More Swap Space?” on page 346
- “How Swap Space Is Allocated” on page 347
- “Planning for Swap Space” on page 347
- “Monitoring Swap Resources” on page 349
- “Adding or Changing Swap Space in an Oracle Solaris ZFS Root Environment” on page 350

### About Swap Space

You should understand the features of the SunOS swap mechanism to determine the following:

- Swap space requirements
- The relationship between swap space and the TMPFS file system
- How to recover from error messages related to swap space

### Swap Space and Virtual Memory

Oracle Solaris OS software and application software can use some disk space for temporary storage rather than for file systems. The reserved area of the disk is called swap space. Swap space is used as virtual memory storage areas when the system does not have enough physical memory to handle current processes. In a ZFS root file system, the disk space reserved for swap is a ZFS volume.

The virtual memory system maps physical copies of files on disk to virtual addresses in memory. Physical memory pages that contain the data for these mappings can be backed by
regular files in the file system, or by swap space. If the memory is backed by swap space it is referred to as anonymous memory because no identity is assigned to the disk space that is backing the memory.

The Oracle Solaris OS uses the concept of virtual swap space, a layer between anonymous memory pages and the physical storage (or disk-backed swap space) that actually back these pages. A system’s virtual swap space is equal to the sum of all its physical (disk-backed) swap space plus a portion of the currently available physical memory.

Virtual swap space has these advantages:

- The need for large amounts of physical swap space is reduced because virtual swap space does not necessarily correspond to physical (disk) storage.
- A pseudo file system called SWAPFS provides addresses for anonymous memory pages. Because SWAPFS controls the allocation of memory pages, it has greater flexibility in deciding what happens to a page. For example, SWAPFS might change the page’s requirements for disk-backed swap storage.

### Swap Space and the TMPFS File System

The TMPFS file system is activated automatically in the Oracle Solaris environment by an entry in the /etc/vfstab file. The TMPFS file system stores files and their associated information in memory (in the /tmp directory) rather than on disk, which speeds access to those files. This feature results in a major performance enhancement for applications such as compilers and DBMS products that use /tmp heavily.

The TMPFS file system allocates space in the /tmp directory from the system’s swap resources. This feature means that as you use up space in the /tmp directory, you are also using up swap space. So, if your applications use the /tmp directory heavily and you do not monitor swap space usage, your system could run out of swap space.

Do use the following if you want to use TMPFS, but your swap resources are limited:

- Mount the TMPFS file system with the size option (-o size) to control how much swap resources TMPFS can use.
- Use your compiler’s TMPDIR environment variable to point to another larger directory. Using your compiler’s TMPDIR variable only controls whether the compiler is using the /tmp directory. This variable has no effect on other programs’ use of the /tmp directory.

### Swap Space and Dump Device Configuration

A dump device is usually disk space that is reserved to store system crash dump information. When a system is installed, a ZFS swap volume and dump volume are created automatically.
You can change a system’s dump volume by using the `dumpadm` command. For more information, see Chapter 17, “Managing System Crash Information (Tasks),” in Oracle Solaris Administration: Common Tasks.

In a ZFS root environment, swap and dump are configured as separate ZFS volumes. The advantages to this model are as follows:

- You don’t have to partition a disk to include swap and dump areas.
- Swap and dump devices benefit from the underlying ZFS I/O pipeline architecture.
- You can set characteristics, such as compression, on swap and dump devices.
- You can reset swap and dump device sizes. For example:

  ```
  # zfs set volsize=2G rpool/dump
  # zfs get volsize rpool/dump
  NAME    PROPERTY       VALUE SOURCE
  rpool/dump volsize   2G -
  ```

  Keep in mind that reallocating a large dump device is a time-consuming process.
- On an active system, you might consider adding a second swap volume to increase overall swap size.

For more information about using ZFS swap and dump devices, see “Managing Your ZFS Swap and Dump Devices” in Oracle Solaris Administration: ZFS File Systems.

### Swap Space and Dynamic Reconfiguration

A good practice is to allocate enough swap space to support a failing CPU or system board during dynamic reconfiguration. Otherwise, a CPU or system board failure might result in your host or domain rebooting with less memory.

Without having this additional swap space available, one or more of your applications might fail to start due to insufficient memory. This problem would require manual intervention either to add additional swap space or to reconfigure the memory usage of these applications.

If you have allocated additional swap space to handle a potential loss of memory on reboot, all of your intensive applications might start as usual. This means the system will be available to the users, perhaps possibly slower due to some additional swapping.

For more information, see your hardware dynamic reconfiguration guide.

### Configuring Swap Space in a SAN Environment

Review the following points to determine whether you might configure swap space on a network-connected disk, such as in a SAN environment:
Diagnosing swap space issues on a locally-attached disk is easier than diagnosing swap space issues on a network-connected disk.

The performance of swap space over a SAN should be comparable to swap space configured on a locally-attached disk.

Adding more memory to a system with performance issues, after analyzing performance data, might resolve a swap over SAN performance problem better than moving the swap to a locally-attached disk.

How Do I Know If I Need More Swap Space?

Use the `swap -l` command to determine if your system needs more swap space.

For example, the following `swap -l` output shows that this system’s swap space is almost entirely consumed or at 100% allocation.

```
% swap -l
swap -l
swapfile  dev  swaplo  blocks  free
/dev/zvol/dsk/rpool/swap  182,2   16 67108848 67102992
```

When a system’s swap space is at 100% allocation, an application’s memory pages become temporarily locked. Application errors might not occur, but system performance will likely suffer.

Swap-Related Error Messages

These messages indicate that an application was trying to get more anonymous memory. However, no swap space was left to back it.

```
application is out of memory
malloc error 0
messages.1:Sep 21 20:52:11 mars genunix: [ID 470503 kern.warning]
WARNING: Sorry, no swap space to grow stack for pid 100295 (myprog)
```

TMPFS-Related Error Messages

The following message is displayed if a page could not be allocated when a file was being written. This problem can occur when TMPFS tries to write more than it is allowed or if currently executed programs are using a lot of memory.

```
directory: File system full, swap space limit exceeded
```
The following message means that TMPFS ran out of physical memory while attempting to create a new file or directory:

```
directory: File system full, memory allocation failed
```

For information on recovering from the TMPFS-related error messages, see `tmpfs(7FS)`.

## How Swap Space Is Allocated

Initially, swap space is allocated as part of the Oracle Solaris installation process. In a ZFS root environment, the default swap size is generally in the 512 MB to 2 GB range.

For general guidelines on allocating swap space, see “Planning for Swap Space” on page 347.

## Swap Areas and the `/etc/vfstab` File

After the system is installed, swap areas and swap files are listed in the `/etc/vfstab` file. They are activated by the `/sbin/swapadd` script when the system is booted.

An entry for a swap device in the `/etc/vfstab` file contains the following:

- The full path name of the swap volume path name on a system with a ZFS root file system
- File system type of the swap slice or swap file

The file system that contains a swap file must be mounted before the swap file is activated. So, in the `/etc/vfstab` file, ensure that the entry that mounts the file system comes before the entry that activates the swap file.

## Planning for Swap Space

The most important factors in determining swap space size are the requirements of the system’s software applications. For example, large applications such as computer-aided design simulators, database management products, transaction monitors, and geologic analysis systems can consume as much as 200–1000 MB of swap space.

Consult your application vendors for swap space requirements for their applications.

If you are unable to determine swap space requirements from your application vendors, use the following general guidelines based on your system type to allocate swap space.
Note – Crash dump content is compressed so the dump device does not have to be the same size as physical memory. By default, the dump content value is set to kernel pages. However, if the dump content value is set to dump all memory pages, then consider increasing the dump size to half the size of physical memory or more.

### TABLE 19–1 Swap and Dump Volume Sizes for ZFS File Systems

<table>
<thead>
<tr>
<th>System Type</th>
<th>Swap Volume Size</th>
<th>Dump Volume Size</th>
</tr>
</thead>
<tbody>
<tr>
<td>System with about 4 GB of physical memory</td>
<td>1 GB</td>
<td>2 GB</td>
</tr>
<tr>
<td>Mid-range server with about 8 GB of physical memory</td>
<td>2 GB</td>
<td>4 GB</td>
</tr>
<tr>
<td>High-end server with about 16 to 128 GB of physical memory</td>
<td>4 GB</td>
<td>8-64 GB</td>
</tr>
<tr>
<td>High-end server with more than 128 GB of physical memory</td>
<td>1/4 of physical memory size</td>
<td>1/2 of physical memory size</td>
</tr>
</tbody>
</table>

Note – A busy system with many active ZFS file systems might use 1/2 to 3/4 the size of physical memory for the size of the dump device.

On a system with a ZFS root file system, if you attempt to designate a dump device that is too small to hold a system crash system with the `dumpadm -d` command, you will see a message similar to the following:

```
dumpadm: dump device /dev/zvol/dsk/rpool/dump is too small to hold a system dump dump size 43467329536 bytes, device size 42949672960 bytes
```

### Allocating Swap Space for ZFS-Based Systems

During an initial installation of a ZFS root file system, a swap area is automatically created on a ZFS volume in the ZFS root pool.

In a ZFS root pool, swap devices are not pre-allocated to fixed-size slices, so it is fairly easy to modify the swap size later.

After you assess the swap requirements of your applications, you can use the default swap size or adjust the swap volume size during an initial installation or after the installation, if necessary.

During an initial installation, the default dump volume size is calculated by the kernel based on `dumpadm` information and the size of physical memory.
In a ZFS environment, file systems consume space from the pool so the /var/crash directory consumes what it needs depending on how many crash dumps are saved.

## Monitoring Swap Resources

The /usr/sbin/swap command is used to manage swap areas. Two options, -l and -s, display information about swap resources.

Use the `swap -l` command to identify a system’s swap areas. Activated swap devices or files are listed under the `swapfile` column. For example:

```
$ swap -l
swapfile   dev   swaplo  blocks  free
/dev/dsk/c0t0d0s1 136,1  16   1638608 1600528
```

On a system with a ZFS root file system, the `swap -l` command identifies similar output except that it identifies the ZFS volume path name. For example:

```
$ swap -l
swapfile   dev   swaplo  blocks  free
/dev/zvol/dsk/rpool/swap 256,1 16   1058800 1058800
```

Use the `swap -s` command to monitor swap resources.

```
# swap -s
total: 57416k bytes allocated + 10480k reserved = 67896k used,
833128k available
```

The `used` value plus the `available` value equals the total swap space on the system, which includes a portion of physical memory and swap devices (or files).

You can use the amount of available and used swap space (in the `swap -s` output) as a way to monitor swap space usage over time. If a system’s performance is good, use `swap -s` to determine how much swap space is available. When the performance of a system slows down, check the amount of available swap space to determine if it has decreased. Then you can identify what changes to the system might have caused swap space usage to increase.

When using this command, keep in mind that the amount of physical memory available for swap usage changes dynamically as the kernel and user processes lock down and release physical memory.

### Note

The `swap -l` command displays swap space in 512-byte blocks. The `swap -s` command displays swap space in 1024-byte blocks. If you add up the blocks from `swap -l` and convert them to KB, the result is less than `used + available` (in the `swap -s` output). The reason is that `swap -l` does not include physical memory in its calculation of swap space.

The output from the `swap -s` command is summarized in the following table.
### Output of the `swap -s` Command

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>bytes allocated</td>
<td>The total amount of swap space in 1024-byte blocks that is currently allocated as backing store (disk-backed swap space).</td>
</tr>
<tr>
<td>reserved</td>
<td>The total amount of swap space in 1024-byte blocks that is not currently allocated, but claimed by memory for possible future use.</td>
</tr>
<tr>
<td>used</td>
<td>The total amount of swap space in 1024-byte blocks that is either allocated or reserved.</td>
</tr>
<tr>
<td>available</td>
<td>The total amount of swap space in 1024-byte blocks that is currently available for future reservation and allocation.</td>
</tr>
</tbody>
</table>

---

**Adding or Changing Swap Space in an Oracle Solaris ZFS Root Environment**

The following section describes how to add or change swap space in a ZFS root environment. See the previous sections to determine if your system or applications need additional swap space.

#### How to Add Swap Space in an Oracle Solaris ZFS Root Environment

1. **Become an administrator.**
   
   For more information, see “How to Use Your Assigned Administrative Rights” in *Oracle Solaris Administration: Security Services.*

2. **Identify the current swap volume.**
   
   A swap volume cannot be removed if it is in use. You can tell if the current swap volume is in use by comparing the blocks identified in the blocks column and blocks identified in the free column. If the blocks in the two columns are equal, the swap area is not busy. For example:

   ```
   # swap -l
   swapfile    dev    swaplo    blocks    free
   /dev/zvol/dsk/rpool/swap 256,1 16 1058800 1058800
   ```
Select one of the following to resize the swap volume.

a. If the current swap area is not in use, you can resize the size of the current swap volume.

For example:

```
# zfs get volsize rpool/swap
NAME PROPERTY VALUE SOURCE
rpool/swap volsize 517M -
# zfs set volsize=2g rpool/swap
# zfs get volsize rpool/swap
NAME PROPERTY VALUE SOURCE
rpool/swap volsize 2G -
```

b. If the current swap area is in use, you can add another swap volume.

For example:

```
# zfs create -V 2G rpool/swap2
```

c. Activate the second swap volume.

For example:

```
# swap -a /dev/zvol/dsk/rpool/swap2
# swap -l
swapfile dev swaplo blocks free
/dev/zvol/dsk/rpool/swap 256,1 16 1058800 1058800
/dev/zvol/dsk/rpool/swap2 256,3 16 4194288 4194288
```
This chapter describes how to copy files and file systems to disk, tape, and diskettes by using various backup commands.

This is a list of the step-by-step instructions in this chapter.

- “How to Copy Directories Between File Systems (cpio)” on page 356
- “How to Copy Files to a Tape (tar)” on page 357
- “How to List the Files on a Tape (tar)” on page 358
- “How to Retrieve Files From a Tape (tar)” on page 359
- “Copying Files to a Tape With the pax Command” on page 360
- “How to Copy All Files in a Directory to a Tape (cpio)” on page 361
- “How to List the Files on a Tape (cpio)” on page 362
- “How to Retrieve All Files From a Tape (cpio)” on page 362
- “How to Retrieve Specific Files From a Tape (cpio)” on page 363
- “How to Copy Files to a Remote Tape Device (tar and dd)” on page 364
- “How to Extract Files From a Remote Tape Device” on page 366

Commands for Copying File Systems

When you want to copy or move individual files, portions of file systems, or complete file systems, you can use the procedures described in this chapter.

The following table describes various backup and restore commands that are available in the Oracle Solaris release. For enterprise environments, consider using a enterprise-level backup product. Information about enterprise-level backup products is available on the Oracle Technical Network.
### TABLE 20–1  Summary of Various Backup Commands

<table>
<thead>
<tr>
<th>Command Name</th>
<th>Aware of File System Boundaries?</th>
<th>Supports Multiple Volume Backups?</th>
<th>Physical or Logical Copy?</th>
</tr>
</thead>
<tbody>
<tr>
<td>volcopy</td>
<td>Yes</td>
<td>Yes</td>
<td>Physical</td>
</tr>
<tr>
<td>tar</td>
<td>No</td>
<td>No</td>
<td>Logical</td>
</tr>
<tr>
<td>cpio</td>
<td>No</td>
<td>Yes</td>
<td>Logical</td>
</tr>
<tr>
<td>pax</td>
<td>Yes</td>
<td>Yes</td>
<td>Logical</td>
</tr>
<tr>
<td>dd</td>
<td>Yes</td>
<td>No</td>
<td>Physical</td>
</tr>
<tr>
<td>zfs send and zfs receive</td>
<td>Yes</td>
<td>N/A</td>
<td>Logical</td>
</tr>
<tr>
<td>zfs snapshot</td>
<td>Yes</td>
<td>N/A</td>
<td>Logical</td>
</tr>
</tbody>
</table>

The following table describes the advantages and disadvantages of some of these commands.

### TABLE 20–2  Advantages and Disadvantages of tar, pax, and cpio Commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Function</th>
<th>Advantages</th>
<th>Disadvantages</th>
</tr>
</thead>
<tbody>
<tr>
<td>tar</td>
<td>Use to copy files and directory subtrees to a single tape.</td>
<td>Available on most UNIX operating systems</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Public domain versions are readily available</td>
<td>Is not aware of file system boundaries</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Length of full path name cannot exceed 255 characters</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Cannot be used to create multiple tape volumes</td>
</tr>
<tr>
<td>pax</td>
<td>Use to copy files, special files, or file systems that require multiple tape volumes. Or, use when you want to copy files to and from POSIX-compliant systems.</td>
<td>Better portability than the tar or cpio commands for POSIX-compliant systems</td>
<td>Same disadvantages as the tar command, except that the pax command can create multiple tape volumes.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Multiple vendor support</td>
</tr>
</tbody>
</table>
TABLE 20–2  Advantages and Disadvantages of tar, pax, and cpio Commands (Continued)

<table>
<thead>
<tr>
<th>Command</th>
<th>Function</th>
<th>Advantages</th>
<th>Disadvantages</th>
</tr>
</thead>
</table>
| cpio    | Use to copy files, special files, or file systems that require multiple tape volumes. Or, use when you want to copy files from systems running current Oracle Solaris releases systems to systems running older Solaris releases. | ■ Packs data onto tape more efficiently than the tar command  
■ Skips over any bad spots in a tape when restoring  
■ Provides options for writing files with different header formats, such as (tar, ustar, crc, odc, bar), for portability between different system types  
■ Creates multiple tape volumes | The command syntax is more difficult than the tar or pax commands. |

The following sections describes step-by-step instructions and examples of how to use these commands.

**Copying Directories Between File Systems (cpio Command)**

You can use the cpio (copy in and out) command to copy individual files, groups of files, or complete file systems. This section describes how to use the cpio command to copy complete file systems.

The cpio command is an archiving program that copies a list of files into a single, large output file. This command inserts headers between the individual files to facilitate recovery. You can use the cpio command to copy complete file systems to another slice, another system, or to a media device, such as a tape or USB diskette.

Because the cpio command recognizes end-of-media and prompts you to insert another volume, it is the most effective command to use to create archives that require multiple tapes or USB diskettes.

With the cpio command, you frequently use the ls and find commands to list and select the files you want to copy, and then to pipe the output to the cpio command.
How to Copy Directories Between File Systems (cpio)

1 Become an administrator.

2 Change to the appropriate directory.
   # cd filesystem1

3 Copy the directory tree from filesystem1 to filesystem2 by using a combination of the find and cpio commands.
   # find . -print -depth | cpio -pdm filesystem2
   - Starts in the current working directory.
   -print Prints the file names.
   -depth Descends the directory hierarchy and prints file names from the bottom up.
   -p Creates a list of files.
   -d Creates directories as needed.
   -m Sets the correct modification times on directories.

For more information, see cpio(1).

The files from the directory name you specify are copied. The symbolic links are preserved.

You might also specify the -u option. This option forces an unconditional copy. Otherwise, older files do not replace newer files. This option might be useful if you want an exact copy of a directory, and some of the files being copied might already exist in the target directory.

4 Verify that the copy was successful by displaying the contents of the destination directory.
   # cd filesystem2
   # ls

5 If appropriate, remove the source directory.
   # rm -rf filesystem1

Example 20–1  Copying Directories Between File Systems (cpio)

# cd /data1
# find . -print -depth | cpio -pdm /data2
19013 blocks
# cd /data2
# ls
# rm -rf /data1
Copying Files and File Systems to Tape

You can use the `tar`, `pax`, and `cpio` commands to copy files and file systems to tape. The command that you choose depends on how much flexibility and precision you require for the copy. Because all three commands use the raw device, you do not need to format or make a file system on tapes before you use them.

The tape drive and device name that you use depend on the hardware configuration for each system. For more information about tape device names, see “Choosing Which Media to Use” on page 367.

Copying Files to Tape (tar Command)

Here is information that you should know before you copy files to tape with the `tar` command:

- Copying files to a tape with the `-c` option to the `tar` command destroys any files already on the tape at or beyond the current tape position.
- You can use file name substitution wildcards (`?` and `*`) as part of the file names that you specify when copying files. For example, to copy all documents with a `.doc` suffix, type `*.doc` as the file name argument.
- You cannot use file name substitution wildcards when you extract files from a `tar` archive.

⚠️ How to Copy Files to a Tape (tar)

1. Change to the directory that contains the files you want to copy.

2. Insert a write-enabled tape into the tape drive.

3. Copy the files to tape.

```bash
$ tar cvf /dev/rmt/n filenames
```

- `-c` Indicates that you want to create an archive.
- `-v` Displays the name of each file as it is archived.
- `-f /dev/rmt/n` Indicates that the archive should be written to the specified device or file.
- `filenames` Indicates the files and directories that you want to copy. Separate multiple files with spaces.

   The file names that you specify are copied to the tape, overwriting any existing files on the tape.

4. Remove the tape from the drive. Write the names of the files on the tape label.
Verify that the files you copied are on the tape.

```bash
$ tar tvf /dev/rmt/
```

For more information on listing files on a `tar` tape, see “How to List the Files on a Tape (tar)” on page 358.

**Example 20–2**  Copying Files to a Tape (tar)

The following example shows how to copy three files to the tape in tape drive 0.

```bash
$ cd /export/home/kryten
$ ls reports
reports A reportB reportC
$ tar cvf /dev/rmt/0 reports
a reports/ 0 tape blocks
a reports/reportA 59 tape blocks
a reports/reportB 61 tape blocks
a reports/reportC 63 tape blocks
$ tar tvf /dev/rmt/0
```

**How to List the Files on a Tape (tar)**

1. Insert a tape into the tape drive.

2. Display the tape contents.

```bash
$ tar tvf /dev/rmt/
```

- **t**: Lists the table of contents for the files on the tape.
- **v**: Used with the `t` option, and provides detailed information about the files on the tape.
- **f /dev/rmt/**: Indicates the tape device.

**Example 20–3**  Listing the Files on a Tape (tar)

The following example shows a listing of files on the tape in drive 0.

```bash
$ tar tvf /dev/rmt/0
```

```
drwxr-xr-x 0/0 0 Jul 14 13:50 2010 reports/
-r--r--r-- 0/0 206663 Jul 14 13:50 2010 reports/reportC
-r--r--r-- 0/0 206663 Jul 14 13:50 2010 reports/reportB
-r--r--r-- 0/0 206663 Jul 14 13:50 2010 reports/reportA
```
How to Retrieve Files From a Tape (tar)

1. Change to the directory where you want to put the files.
2. Insert the tape into the tape drive.
3. Retrieve the files from the tape.
   \$ tar xvf /dev/rmt/0 [filenames]
   - x Indicates that the files should be extracted from the specified archive file. All files on the tape in the specified drive are copied to the current directory.
   - v Displays the name of each file as it is retrieved.
   - f /dev/rmt/0 Indicates the tape device that contains the archive.
   - filenames Specifies a file to retrieve. Separate multiple files with spaces.

   For more information, see the tar(1) man page.

4. Verify that the files have been copied.
   \$ ls -l

Example 20–4 Retrieving Files on a Tape (tar)

The following example shows how to retrieve all the files from the tape in drive 0.

\$ cd /var/tmp
\$ tar xvf /dev/rmt/0
   x reports/, 0 bytes, 0 tape blocks
   x reports/reportA, 0 bytes, 0 tape blocks
   x reports/reportB, 0 bytes, 0 tape blocks
   x reports/reportC, 0 bytes, 0 tape blocks
   x reports/reportD, 0 bytes, 0 tape blocks
\$ ls -l

Troubleshooting The names of the files extracted from the tape must exactly match the names of the files that are stored on the archive. If you have any doubts about the names or paths of the files, first list the files on the tape. For instructions on listing the files on the tape, see “How to List the Files on a Tape (tar)” on page 358.
Copying Files to a Tape With the pax Command

How to Copy Files to a Tape (pax)

1. Change to the directory that contains the files you want to copy.

2. Insert a write-enabled tape into the tape drive.

3. Copy the files to tape.
   $ pax -w -f /dev/rmt/\n filenames
   -w Enables the write mode.
   -f /dev/rmt/\n Identifies the tape drive.
   filenames Indicates the files and directories that you want to copy. Separate multiple files with spaces.

   For more information, see the pax(1) man page.

4. Verify that the files have been copied to tape.
   $ pax -f /dev/rmt/\n
5. Remove the tape from the drive. Write the names of the files on the tape label.

Example 20–5  Copying Files to a Tape (pax)

The following example shows how to use the pax command to copy all the files in the current directory.

$ pax -w -f /dev/rmt/0 .
$ pax -f /dev/rmt/0
filea fileb filec
Copying Files to Tape With the cpio Command

How to Copy All Files in a Directory to a Tape (cpio)

1. Change to the directory that contains the files you want to copy.
2. Insert a write-enabled tape into the tape drive.
3. Copy the files to tape.
   
   `$ ls | cpio -oc > /dev/rmt/n`
   
   `ls` Provides the cpio command with a list of file names.
   
   `cpio -oc` Specifies that the cpio command should operate in copy-out mode (-o) and write header information in ASCII character format (-c). These options ensure portability to other vendors’ systems.
   
   `> /dev/rmt/n` Specifies the output file.
   
   All files in the directory are copied to the tape in the drive you specify, overwriting any existing files on the tape. The total number of blocks that are copied is shown.
4. Verify that the files have been copied to tape.
   
   `$ cpio -civt < /dev/rmt/n`
   
   `-c` Specifies that the cpio command should read files in ASCII character format.
   
   `-i` Specifies that the cpio command should operate in copy-in mode, even though the command is only listing files at this point.
   
   `-v` Displays the output in a format that is similar to the output from the `ls -l` command.
   
   `-t` Lists the table of contents for the files on the tape in the tape drive that you specify.
   
   `< /dev/rmt/n` Specifies the input file of an existing cpio archive.
5. Remove the tape from the drive. Write the names of the files on the tape label.

Example 20–6
Copying All Files in a Directory to a Tape (cpio)

The following example shows how to copy all of the files in the `/export/home/kryten` directory to the tape in tape drive 0.
Copying Files to Tape With the `cpio` Command

$ cd /export/home/kryten
$ ls | cpio -oc > /dev/rmt/0
1280 blocks
$ cpio -civt < /dev/rmt/0
  -r--r--r-- 1 kryten staff 206663 Jul 14 13:52 2010, filea
  -r--r--r-- 1 kryten staff 206663 Jul 14 13:52 2010, fileb
  -r--r--r-- 1 kryten staff 206663 Jul 14 13:52 2010, filec
  drwxr-xr-x 2 kryten staff 0 Jul 14 13:52 2010, letters
  drwxr-xr-x 2 kryten staff 0 Jul 14 13:52 2010, reports
1280 blocks

▼ How to List the Files on a Tape (cpio)

**Note** – Listing the table of contents on a tape takes a long time because the `cpio` command must process the entire archive.

1. Insert an archive tape into the tape drive.
2. List the files on the tape.

   $ cpio -civt < /dev/rmt/n

**Example 20–7** Listing the Files on a Tape (cpio)

The following example shows how to list the files on the tape in drive 0.

$ cpio -civt < /dev/rmt/0
  -r--r--r-- 1 kryten staff 206663 Jul 14 13:52 2010, filea
  -r--r--r-- 1 kryten staff 206663 Jul 14 13:52 2010, fileb
  -r--r--r-- 1 kryten staff 206663 Jul 14 13:52 2010, filec
  drwxr-xr-x 2 kryten staff 0 Jul 14 13:52 2010, letters
  drwxr-xr-x 2 kryten staff 0 Jul 14 13:52 2010, reports
1280 blocks

▼ How to Retrieve All Files From a Tape (cpio)

If the archive was created using relative path names, the input files are built as a directory within the current directory when you retrieve the files. If, however, the archive was created with absolute path names, the same absolute paths are used to re-create the file on your system.

**Caution** – The use of absolute path names can be dangerous because you might overwrite existing files on your system.

1. Change to the directory where you want to put the files.
2 Insert the tape into the tape drive.

3 Extract all files from the tape.
   
   $ cpio -icvd < /dev/rmt
   
   -i Extracts files from standard input.
   -c Specifies that the cpio command should read files in ASCII character format.
   -v Displays the files as they are retrieved in a format that is similar to the output from the ls command.
   -d Creates directories as needed.
   < /dev/rmt/ Specifies the output file.

4 Verify that the files were copied.
   
   $ ls -l

Example 20–8 Retrieving All Files From a Tape (cpio)

The following example shows how to retrieve all files from the tape in drive 0.

   $ cd /var/tmp
   cpio -icvd < /dev/rmt/0
   answers
   sc.directives
   tests
   8 blocks
   $ ls -l

▼ How to Retrieve Specific Files From a Tape (cpio)

1 Change to the directory where you want to put the files.

2 Insert the tape into the tape drive.

3 Retrieve a subset of files from the tape.
   
   $ cpio -icv "file" < /dev/rmt
   
   -i Extracts files from standard input.
   -c Specifies that the cpio command should read headers in ASCII character format.
Display the files as they are retrieved in a format that is similar to the output from the `ls` command.

"*file" Specifies that all files that match the pattern are copied to the current directory. You can specify multiple patterns, but each pattern must be enclosed in double quotation marks.

< /dev/rmt/n Specifies the input file.

For more information, see the `cpio(1)` man page.

4 Verify that the files were copied.

```
$ ls -l
```

Example 20–9 Retrieving Specific Files From a Tape (cpio)

The following example shows how to retrieve all files with the `chapter` suffix from the tape in drive 0.

```
$ cd /home/smith/Book
$ cpio -icv "*chapter" < /dev/rmt/0
Boot.chapter
Directory.chapter
Install.chapter
Intro.chapter
31 blocks
$ ls -l
```

Copying Files to a Remote Tape Device

How to Copy Files to a Remote Tape Device (tar and dd)

1 The following prerequisites must be met to use a remote tape drive:

a. The local host name and optionally, the user name of the user doing the copy, must appear in the remote system's `/etc/hosts.equiv` file. Or, the user doing the copy must have his or her home directory accessible on the remote machine, and have the local machine name in `$HOME/.rhosts`.

   For more information, see the `hosts.equiv(4)` man page.

b. An entry for the remote system must be in the local system's `/etc/inet/hosts` file or in the nameservice `hosts` file.
To test whether you have the appropriate permission to execute a remote command, try the following:

```
$ rsh remotehost echo test
```

If `test` is echoed back to you, you have permission to execute remote commands. If `Permission denied` is echoed back to you, check your setup as described in Step 1.

3 Change to the directory where you want to put the files.

4 Insert the tape into the tape drive.

5 Copy the files to a remote tape drive.

```
$ tar cvf filenames | rsh remote-host dd of=/dev/rmt/n obs=block-size
```

- `tar cf` Creates a tape archive, lists the files as they are archived, and specifies the tape device.
- `v` Provides additional information about the tar file entries.
- `-` (Hyphen) Represents a placeholder for the tape device.
- `filenames` Identifies the files to be copied. Separate multiple files with spaces.
- `rsh | remote-host` Pipes the `tar` command's output to a remote shell.
- `dd of=/dev/rmt/n` Represents the output device.
- `obs=block-size` Represents the blocking factor.

6 Remove the tape from the drive. Write the names of the files on the tape label.

---

Example 20–10 Copying Files to a Remote Tape Drive (tar and dd)

```
# tar cvf - * | rsh mercury dd of=/dev/rmt/0 obs=126b
a answers/ 0 tape blocks
a answers/test129 1 tape blocks
a sc.directives/ 0 tape blocks
a sc.directives/sc.190089 1 tape blocks
a tests/ 0 tape blocks
a tests/test131 1 tape blocks
6+9 records in
0+1 records out
```
How to Extract Files From a Remote Tape Device

1. Insert the tape into the tape drive.

2. Change to a temporary directory.
   
   ```bash
   $ cd /var/tmp
   ```

3. Extract the files from a remote tape device.
   
   ```bash
   $ rsh remote-host dd if=/dev/rmt/ | tar xvBpf -
   ```
   
   `rsh remote-host` Indicates a remote shell that is started to extract the files from the tape device by using the `dd` command.

   `dd if=/dev/rmt/n` Indicates the input device.

   `| tar xvBpf -` Pipes the output of the `dd` command to the `tar` command, which is used to restore the files.

4. Verify that the files have been extracted.
   
   ```bash
   $ ls -l
   ```

Example 20–11  Extracting Files From a Remote Tape Drive

```bash
$ cd /var/tmp
$ rsh mercury dd if=/dev/rmt/0 | tar xvBpf -
x answers/  0 bytes, 0 tape blocks  
x answers/test129, 48 bytes, 1 tape blocks  
20+0 records in  
20+0 records out  
x sc.directives/, 0 bytes, 0 tape blocks  
x sc.directives/sc.190089, 77 bytes, 1 tape blocks  
x tests/, 0 bytes, 0 tape blocks  
x tests/test131, 84 bytes, 1 tape blocks  
$ ls -l
```
Managing Tape Drives (Tasks)

This chapter describes how to manage tape drives in the Oracle Solaris OS.

This is a list of the step-by-step instructions in this chapter.

- "How to Display Tape Drive Status" on page 370
- "Retensioning a Magnetic Tape Cartridge" on page 371
- "Rewinding a Magnetic Tape Cartridge" on page 371

This is a list of overview information in this chapter.

- "Choosing Which Media to Use" on page 367
- "Backup Device Names" on page 368
- "Displaying Tape Drive Status" on page 370

Choosing Which Media to Use

You typically back up Oracle Solaris systems by using the following tape media:

- 1/2-inch reel tape
- 1/4-inch streaming cartridge tape
- 8-mm cartridge tape
- 4-mm cartridge tape (DAT)

The media that you choose depends on the availability of the equipment that supports it and of the media (usually tape) that you use to store the files. Although you must do the backup from a local system, you can write the files to a remote device.

The following table shows typical tape devices that are used for backing up file systems. The storage capacity for each device depends on the type of drive and the data being written to the tape.
### Backup Device Names

You specify a tape or device to use for backup by supplying a logical device name. This name points to the subdirectory that contains the “raw” device file and includes the logical unit number of the drive. Tape drive naming conventions use a logical, not a physical, device name. The following table shows this naming convention.

#### TABLE 21–2 Basic Device Names for Backup Devices

<table>
<thead>
<tr>
<th>Device Type</th>
<th>Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tape</td>
<td>/dev/rmt/n</td>
</tr>
</tbody>
</table>

In general, you specify a tape device as shown in the following figure.

![Figure 21-1 Tape Drive Device Names](image-url)
If you don’t specify the density, a tape drive typically writes at its “preferred” density. The preferred density usually means the highest density the tape drive supports. Most SCSI drives can automatically detect the density or format on the tape and read it accordingly. To determine the different densities that are supported for a drive, look at the /dev/rmt subdirectory. This subdirectory includes the set of tape device files that support different output densities for each tape.

Also, a SCSI controller can have a maximum of seven SCSI tape drives.

**Specifying the Rewind Option for a Tape Drive**

Normally, you specify a tape drive by its logical unit number, which can run from 0 to n. The following table describes how to specify tape device names with a rewind or a no-rewind option.

<table>
<thead>
<tr>
<th>Drive and Rewind Value</th>
<th>Use This Option</th>
</tr>
</thead>
<tbody>
<tr>
<td>First drive, rewind</td>
<td>/dev/rmt/0</td>
</tr>
<tr>
<td>First drive, no rewind</td>
<td>/dev/rmt/0n</td>
</tr>
<tr>
<td>Second drive, rewind</td>
<td>/dev/rmt/1</td>
</tr>
<tr>
<td>Second drive, no rewind</td>
<td>/dev/rmt/1n</td>
</tr>
</tbody>
</table>

**Specifying Different Densities for a Tape Drive**

By default, the drive writes at its “preferred” density, which is usually the highest density the tape drive supports. If you do not specify a tape device, the command writes to drive number 0 at the default density the device supports.

To transport a tape to a system whose tape drive supports only a certain density, specify a device name that writes at the desired density. The following table describes how to specify different densities for a tape drive.

<table>
<thead>
<tr>
<th>Drive, Density, and Rewind Value</th>
<th>Use This Option</th>
</tr>
</thead>
<tbody>
<tr>
<td>First drive, low density, rewind</td>
<td>/dev/rmt/0l</td>
</tr>
<tr>
<td>First drive, low density, no rewind</td>
<td>/dev/rmt/0ln</td>
</tr>
<tr>
<td>Second drive, medium density, rewind</td>
<td>/dev/rmt/1m</td>
</tr>
<tr>
<td>Second drive, medium density, no rewind</td>
<td>/dev/rmt/1mn</td>
</tr>
</tbody>
</table>
The additional density values are shown in "Backup Device Names" on page 368.

Displaying Tape Drive Status

You can use the status option with the mt command to get status information about tape drives. The mt command reports information about any tape drives that are described in the /kernel/drv/st.conf file.

▼ How to Display Tape Drive Status

1. Load a tape into the drive you want information about.
2. Display the tape drive status.
   
   # mt -f /dev/rmt/n status

3. Repeat steps 1–2, substituting tape drive numbers 0, 1, 2, 3, and so on to display information about all available tape drives.

Example 21–1

Displaying Tape Drive Status

The following example shows the status for a QIC-150 tape drive (/dev/rmt/0):

```
$ mt -f /dev/rmt/0 status
Archive QIC-150 tape drive:
sense key(0x0)= No Additional Sense residual= 0 retries= 0
   file no= 0   block no= 0
```

The following example shows the status for an Exabyte tape drive (/dev/rmt/1):

```
$ mt -f /dev/rmt/1 status
Exabyte EXB-8200 8mm tape drive:
sense key(0x0)= NO Additional Sense residual= 0 retries= 0
   file no= 0   block no= 0
```

The following example shows a quick way to poll a system and locate all of its tape drives:

```
$ for drive in 0 1 2 3 4 5 6 7
   > do
   >   mt -f /dev/rmt/$drive status
   > done
Archive QIC-150 tape drive:
sense key(0x0)= No Additional Sense residual= 0 retries= 0
   file no= 0   block no= 0
/dev/rmt/1: No such file or directory
/dev/rmt/2: No such file or directory
/dev/rmt/3: No such file or directory
```
Handling Magnetic Tape Cartridges

If errors occur when a tape is being read, you can retension the tape, clean the tape drive, and then try again.

Retensioning a Magnetic Tape Cartridge

Retension a magnetic tape cartridge with the `mt` command.

For example:

```
$ mt -f /dev/rmt/1 retension
```

**Note** – Do not retension non-QIC tape drives.

Rewinding a Magnetic Tape Cartridge

To rewind a magnetic tape cartridge, use the `mt` command.

For example:

```
$ mt -f /dev/rmt/1 rewind
```

Guidelines for Drive Maintenance and Media Handling

A backup tape that cannot be read is useless. So, periodically clean and check your tape drives to ensure correct operation. See your hardware manuals for instructions on procedures for cleaning a tape drive. You can check your tape hardware by copying some files to the tape, reading the files back, and then comparing the original files with the copied files.

Be aware that hardware can fail in ways that the system does not report.

Always label your tapes after a backup. This label should never change. Every time you do a backup, make another tape label that contains the following information:
- The backup date
- The name of the machine and file system that is backed up
- The backup level
- The tape number (1 of n, if the backup spans multiple volumes)
- Any information specific to your site

Store your tapes in a dust-free safe location, away from magnetic equipment. Some sites store archived tapes in fireproof cabinets at remote locations.

You should create and maintain a log that tracks which media (tape volume) stores each job (backup) and the location of each backed-up file.
Index

Numbers and Symbols
/export/home directory, 321
1394 (FireWire) support, description, 113
4.3 Tahoe file system, 315
9660 DVD format, 36

A
accessing
disk devices, 73
iSCSI disks (how to), 256
removable media (how to), 38
tape devices, 75
adding
a SCSI device to a SCSI bus (how to), 88
a USB camera (how to), 133
a USB mass storage device (how to), 132
adding a new removable media drive (how to), 37
entry to /etc/vfstab file (how to), 336
PCI adapter card (how to), 97
swap to vfstab, 347
USB audio devices (how to), 148
VPPA communication service (how to), 169
autoconfiguration process, 62
autofs, 324
automounting, and /home, 324

B
backup, device names, 368–370
block disk device interface
definition, 73
when to use, 74
BSD Fat Fast File system, 315
bus-oriented disk controllers, 75

C
cdrw command
description, 43
restricting access to (how to), 46
writing data CDs and DVDs and audio CDs
(overview), 45
cfgadm
PCI hot-plugging (overview), 78
SCSI hot-plugging (overview), 78
changing, primary USB audio device (how to), 149
checking, a file system on removable media (how to), 32
clients
iSNS, 292
iSNS, display status, CLI, 300
iSNS, managing, CLI, 299–302
configuring
a SCSI controller (how to), 85
a SCSI device (how to), 86
a USB device (how to), 153
IB Port, HCA_SVC, or a VPPA device (how to), 165
IB pseudo device (how to), 165
iSNS server, 293, 296–299
SATA device (how to), 101
configuring (Continued) unidirectional or bidirectional CHAP authentication for iSCSI (how to), 268

connecting
a SCSI controller (how to), 87
a USB device, logically (how to), 154
copying
directories between file systems with cpio command (overview), 355
groups of files with cpio command (overview), 355
individual files with cpio command (overview), 355
cpio command (overview), 355
copying directories between file systems (how to), 356
extract all files from tape (how to), 363
listing files on tape (how to), 362
creating
a data CD or DVD file system (how to), 48
a disk slice for a ZFS root file system
SPARC, 220
a file system on a DVD-RAM (how to), 32
a file system on a USB mass storage device (how to), 136
a file system on removable media (how to), 31
a Solaris partition and modifying the slices on a USB mass storage device (how to), 141
disk slice for a ZFS root file system (how to) x86, 230
loopback file system (overview), 328
CTFS file system, 319
/disk directory, 73
/dev file system, description, 58
/dev directory, 73
devfsadm command, 72
device driver
adding, 71
definition, 60
device names, backup, 368–370
devices, accessing, 71
directories
copying between file systems with cpio command (overview), 355
/proc, 318
/tmp, 318

disabling
removable media services, 24
removable media services (how to), 37
write protection on removable media (how to), 33
disconnect, a USB device, logically (how to), 153
disconnecting
a SCSI controller (how to), 86
a USB device subtree, logically (how to), 154
discovery domain
client
removing, CLI, 300
iSNS
adding, CLI, 297–298
assigning clients, CLI, 298–299
creating, CLI, 297
displaying status, CLI, 300
removing, 301
discovery domain set
disabling
displaying status, CLI, 301
iSNS
creating, CLI, 296–297
displaying status, CLI, 299
removing, 301–302
disk
formatting a (overview), 193
setting up for ZFS file systems (overview)
x86, 228–235
when to format (overview), 202
disk-based file systems, 315

D
datadm command, 172
default
file system for /tmp (TMPFS), 318
SunOS file system, 320
default discovery domain, iSNS, 292
default discovery domain set, enabling, 296
detecting end of media, cpio command, 355
determining
file system types, 325
mounted file systems, 329
disk controllers, 74

disk for a ZFS file system
setting up (how to)
  SPARC, 224

disk label
  creating (overview), 206
  description, 186

disk setup for a ZFS root file system
connecting (how to)
  SPARC, 219
  x86, 229
disk slice
  creating for a ZFS root file system
    SPARC, 220
  for a ZFS root file system (how to)
    x86, 230
disk slices
  definition, 190
  displaying information about (overview), 204
diskettes
  creating a file system on (how to), 31
  formatting with rmformat (how to), 30
  loading with volume management (how to), 29
disks
  determining if formatted (how to), 203
  recovering a corrupted disk label (how to), 212
  recovering a corrupted disk label (overview), 212
  setting up for ZFS file systems (overview)
    SPARC, 218
displaying
  disk slice information (overview), 204
  IB communication services (how to), 169
  InfiniBand device information (how to), 162
  information about SCSI devices, 84
  kernel IB clients of an HCA (how to), 166
  PCI slot configuration information (how to), 94
  removable media user (how to), 39
  SATA devices (how to), 100
  swap space, 349–350
  system configuration information, 63, 66
  USB bus information (how to), 152
  USB device information (how to), 135
displaying settings, iSNS server, 296
DOS, file system, 316
driver not attached message, 63
dumpadm command, 74
DVD-ROM, creating a file system on (how to), 317
DVDs, ISO 9660 format, 36
dynamic reconfiguration, InfiniBand devices, 161
dynamic reconfiguration (overview), 78

E
EFI label
  (overview), 187
  comparison with VTOC label, 188
  installing a system with, 189
  restrictions of, 188
  troubleshooting problems, 189
  eject command, removable media (how to), 39
  ejecting, removable media (how to), 39
  enabling
    removable media services (how to), 37
    uDAPL, 171
  write protection on removable media (how to), 33
  end-of-media detection, cpio command, 355
  error messages, iSCSI, 286

F
FDFS file system, 319
FIFOFS file system, 319
file systems
  4.3 Tahoe, 315
  BSD Fat Fast, 315
  creating (overview)
    loopback (LOFS), 328
    CTFS, 319
  default SunOS, 320
disk-based, 315
DOS, 316
/export/home, 321
FDFS, 319
FIFOFS, 319
finding types, 325
High Sierra, 316

375
file systems (Continued)
ISO 9660, 316
making available (overview), 328
MNTFS, 321
mount table, 322
NAMEFS, 319
network-based, 317
OBJFS, 319
/opt, 321
PCFS, 316
/etc/dfs/fstypes, 325
in the /proc directory, 318
retrieving from tape with tar command (how to), 359
-sharing, 323–324
finding, type of file system, 325
format utility
(format utility (Continued)
identifying disks on a system (examples of), 201
identifying disks on a system (how to), 200
input to, 310, 311
labeling a disk
example of, 208
main menu, 304
partition menu, 306
recovering corrupted disk label (how to), 213
using help facility, 311
when to use, 192
formatting, diskettes with rmformat (how to), 30
formatting a disk, (overview), 193
free hog slice, See donor slice
fsstat command, description, 313
fsstat command (examples of), 313
fstypes file, 325
fuser command
finding if removable media is in use (how to), 39
killing processes accessing removable media (how to), 39

G
GRUB
managing disks with GRUB
x86, 184

H
halt, description, 22
High Sierra filesystem, 316
/home (automounted), 324
hot-plugging
(overview), 78
adding a SCSI device to a SCSI bus (how to), 88
adding PCI adapter card (how to), 97
configuring a SCSI controller (how to), 85
configuring a SCSI device (how to), 153
configuring a USB device (how to), 153
connecting a SCSI controller (how to), 87
disconnecting a SCSI controller with cfgadm command (how to), 86
hot-plugging (Continued)
logically connecting a USB device (how to), 154
logically disconnecting a USB device (how to), 153
logically disconnecting a USB device subtree (how to), 154
PCI devices (overview), 94
removing a SCSI device (how to), 90
removing PCI adapter card (how to), 96
replacing an identical SCSI device on a SCSI controller (how to), 89
unconfiguring a SCSI device (how to), 85
unconfiguring a USB device (how to), 152
HSFS, See High Sierra file system

Index
identifying
devices, 64
disks on a system (how to), 200
primary USB audio device (how to), 148
InfiniBand devices
adding a VPPA communication service (how to), 169
configuring an IB Port, HCA_SVC, or a VPPA device (how to), 165
configuring an IB pseudo device (how to), 165
displaying (how to), 162
displaying IB communication services (how to), 169
displaying kernel IB clients of an HCA (how to), 166
dynamic reconfiguration (overview), 161
overview, 157
removing an existing IB port, HCA_SVC, or a VPPA communication service (how to), 170
unconfiguring an IB Port, HCA_SVC, or a VPPA device (how to), 164
unconfiguring an IB pseudo device (how to), 165
updating the IB P_key tables (how to), 168
initiators
iSNS, display status, CLI, 300
iSNS, managing, CLI, 299–302
installboot command for a ZFS root file system, 223
installgrub command for a ZFS root file system, 233
installing boot blocks for a ZFS root file system (how to), SPARC, 223
Internet Storage Name Service, See iSNS
iSCSI
(overview), 241
accessing iSCSI disks (how to), 256
configuring unidirectional or bidirectional CHAP authentication for (how to), 268
general iSCSI error messages, 286
modifying iSCSI initiator and target parameters (how to), 280
monitoring your iSCSI configuration (how to), 276
removing discovered iSCSI targets (how to), 253
static and dynamic target discovery, 246
troubleshooting iSCSI configuration problems (how to), 283
iscsiadm list, displaying iSCSI configuration information (example of), 276
iscsiadm modify command enabling CHAP (example of), 269
enabling or disabling static or dynamic targets (example of), 253
iscsiadm remove command, removing static or dynamic targets (example of), 253
iSNS
clients, 292
default discovery domain, 292
display client status, 300
display initiator status, 300
display target status, 300
managing clients, 299–302
managing initiators, 299–302
managing targets, 299–302
nodes, 292
technology overview, 291
iSNS server
configuring, 293
configuring, CLI, 296–299
discovery domain
adding, CLI, 297–298
assigning clients, CLI, 298–299
creating, CLI, 297
displaying status, CLI, 300
removing, CLI, 301
iSNS server, discovery domain (Continued)
   removing a client, 300
   discovery domain set
      creating, CLI, 296–297
      disabling, CLI, 301
      displaying status, CLI, 299
      removing, 301–302
      display settings, 296
   enable, disable server state notifications, 294–295
   enabling default discovery domain set, 296
   set retry threshold, 295
   specify data store location, 291, 295
ISO 9660 file system, 316
ISO standards, 9660 DVD format, 36

K
/kernel/drv directory, 62
killing
   all processes accessing a file system (how to), 340
   processes accessing removable media (how to), 39

L
loading, diskettes (how to), 29
logical device name
   definition, 72
   disk, 73
   tape, 75
logical device names, removable media, 76
loopback file system (LOFS), creating (overview), 328

M
maintaining tape drives, 371
managing
   disks with GRUB
      x86, 184
   iSNS, 300
   clients, CLI, 299–302
media was found message, 29
memory storage (virtual), definition, 343
mkisofs command, create a data CD or DVD file
   system (how to), 48
MNTFS file system, 321
mnttab file, 322
modifying
   iSCSI initiator and target parameters (how to), 280
   partitions and creating a PCFS file system on a USB
   mass storage device (how to), 138
   monitoring, your iSCSI configuration (how to), 276
   mount point, definition, 322
   mount table, 322
   mounting
      a file system with /etc/vfstab, 337
      a USB mass storage device (how to), 143
      file systems automatically, 324
      NFS file systems, 337
      remote removable media manually (example of), 42
      removable media
         automatic mounting compared to, 25, 26
      UFS file systems, 337
   mt command, 371

N
NAMEFS file system, 319
network-based file systems, 317
NFS
   description, 323
   server description, 323
   vfstab entry for, 337
   nfs daemon
      starting, 41
      verifying if running, 40
no media was found message, 30
nodes
   iSNS, 292
   iSNS, display status, CLI, 300
   iSNS, managing, CLI, 299–302

O
OBJFS file system, 319
/opt directory, 321
uDAPL
  (overview), 170
  enabling (how to), 171
  registering a service provider in the DAT static registry (how to), 172
  unregistering a service provider in the DAT static registry (how to), 172
  updating the DAT static registry (how to), 172
UDF filesystem, 316
UFS filesystem, 315
  mounting, 337
  mounting with /etc/vfstab, 337
unconfiguring
  a SCSI controller (how to), 85
  a USB device (how to), 152
  IB Port, HCA_SVC, or a VPPA Device (how to), 164
  IB pseudo device (how to), 165
  SATA device (how to), 100
UNIX file system, 315
unmounting
  a USB mass storage device (how to), 143
unregistering, service provider in the DAT static registry (how to), 172
unsupported devices, 62
updating
  DAT static registry (how to), 172
  IB P_key tables (how to), 168
USB devices
  (overview), 115
  acronyms, 115
  adding a USB camera (how to), 133
  adding a USB mass storage device (how to), 132
  adding audio devices (how to), 148
  audio (overview), 147
  changing the primary device (how to), 149
  device ownership, 149
  bus description, 115
  bus-powered devices, 122
  cables for, 124
  composite device, 116
  compound device, 116
  USB devices (Continued)
    configuring a USB device (how to), 153
    connect a USB device (how to), 154
    creating a file system on a mass storage device (how to), 136
    creating a Solaris partition and modifying the slices on a mass storage device (how to), 141
    device classes, 117
    device nodes, 119
    diskette devices (overview), 130
    displaying bus information (how to), 152
    displaying USB device information (how to), 135
    drivers, 117
    EHCI, OHCI, and UHCI support, 119
    host controller and root hub, 123
    hot-plugging (overview), 132
    hotpluggable attribute, 112
    identifying primary audio device (how to), 148
    Interface Association Descriptor (IAD) support, 112
    keyboards and mouse devices, 122
    logically disconnecting a USB device (how to), 153
    logically disconnecting a USB device subtree (how to), 154
    modifying partitions and creating a PCFS file system on a mass storage device (how to), 138
    mounting mass storage (how to), 143
    mounting or unmounting a mass storage device (how to), 143
    names of, 116
    Oracle Solaris USB Architecture (USBA), 119
    overview of 2.0 devices, 120
    physical device hierarchy, 116
    power management, 124
    Prolific and Keyspan serial adapter support, 118
    removable mass storage (overview), 129
    removing a USB mass storage device (how to), 134
    resetting a USB device (how to), 155
    troubleshooting audio device problems, 149
    troubleshooting tips for mass storage devices, 144
    unconfiguring a device (how to), 152
    unmounting mass storage (how to), 143
    USB 2.0 support, 120
USB devices (Continued)
  using non-compliant mass storage devices (overview), 131
  virtual keyboard and mouse support, 113
  wheel mouse support, 123

V
/var directory, 321
verifying, nfsd daemon is running, 40
vfstab file, 325, 347
   adding entries to (how to), 336
   adding swap to, 347
   entry for LOFS, 336
virtual memory storage, definition, 343
vold, removal of, 22
volfs, removal of, 22

W
writing, data CDs and DVDs and audio CDs (overview), 45

Z
ZFS file system, 315
   setting up a disk (how to)
      x86, 234