



USB Administration Guide

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

Part No: 816-1387-10
October 2001

Copyright 2001 Sun Microsystems, Inc. 901 San Antonio Road, Palo Alto, CA 94303-4900 U.S.A. All rights reserved.

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

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

Sun, Sun Microsystems, the Sun logo, docs.sun.com, AnswerBook, AnswerBook2, and Solaris are trademarks, registered trademarks, or service marks of Sun Microsystems, Inc. in the U.S. and other countries. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. in the U.S. and other countries. Products bearing SPARC trademarks are based upon an architecture developed by Sun Microsystems, Inc.

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

Federal Acquisitions: Commercial Software—Government Users Subject to Standard License Terms and Conditions.

DOCUMENTATION IS PROVIDED "AS IS" AND ALL EXPRESS OR IMPLIED CONDITIONS, REPRESENTATIONS AND WARRANTIES, INCLUDING ANY IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT, ARE DISCLAIMED, EXCEPT TO THE EXTENT THAT SUCH DISCLAIMERS ARE HELD TO BE LEGALLY INVALID.

Copyright 2001 Sun Microsystems, Inc. 901 San Antonio Road, Palo Alto, CA 94303-4900 U.S.A. Tous droits réservés.

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

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

Sun, Sun Microsystems, le logo Sun, docs.sun.com, AnswerBook, AnswerBook2, et Solaris sont des marques de fabrique ou des marques déposées, ou marques de service, de Sun Microsystems, Inc. aux Etats-Unis et dans d'autres pays. Toutes les marques SPARC sont utilisées sous licence et sont des marques de fabrique ou des marques déposées de SPARC International, Inc. aux Etats-Unis et dans d'autres pays. Les produits portant les marques SPARC sont basés sur une architecture développée par Sun Microsystems, Inc.

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

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



040210@7940



Contents

Preface	5
1 USB Devices (Overview)	9
Overview of USB Devices	9
Commonly Used USB Acronyms	10
USB Bus Description	11
About USB in the Solaris Environment	13
USB Keyboards and Mouse Devices	13
USB Host Controller and Root Hub	14
USB Storage Devices	14
SPARC Only: USB Power Management	15
Hot-Plugging USB Devices	15
USB Cables	16
USB Printer Support	16
2 USB Devices (Tasks)	17
Managing USB Devices	18
Managing USB Mass Storage Devices With <code>vold</code> Running	18
▼ How to Mount or Unmount a USB Mass Storage Device With <code>vold</code> Running	19
▼ How to Remove a Hot-Pluggable USB Mass Storage Device With <code>vold</code> Running	20
▼ How to Add a Hot-Pluggable USB Mass Storage Device With <code>vold</code> Running	20
Managing USB Mass Storage Devices Without <code>vold</code> Running	20

How to Mount or Unmount a USB Mass Storage Device Without vold Running	21
▼ How to Remove a Hot-Pluggable USB Mass Storage Device Without vold Running	22
How to Add a Hot-Pluggable USB Mass Storage Device Without vold Running	22
Creating Data on or Extracting Data From a USB CD	22
▼ How to Prepare for Creating Data on or Extracting Data From a USB CD	23
▼ How to Add a Hot-Pluggable USB Camera	23
USB Audio Overview	24
Hot-Plugging Multiple USB Audio Devices	25
▼ How to Add Hot-Pluggable USB Audio Devices	25
Troubleshooting USB Audio Device Problems	26
Audio Device Ownership Key Points	26
▼ How to Identify Your System's Primary Audio Device	26
▼ How to Change the Primary USB Audio Device	28
▼ How to Remove Unused USB Audio Device Links	29
Index	31

Preface

The *USB Administration Guide* provides an overview of managing USB devices on systems that run the Solaris release.

Note – The Solaris operating environment runs on two types of hardware, or platforms: SPARC™ and IA (Intel Architecture). The Solaris operating environment also runs on both 64-bit and 32-bit address spaces. The information in this document pertains to both platforms and address spaces unless called out in a special chapter, section, note, bullet, figure, table, example, or code example.

Who Should Use This Book

This book is intended for Solaris system administrators or developers who are responsible for connecting USB devices to systems that run the Solaris release.

How This Book Is Organized

This book consists of the following chapters:

Chapter 1 provides an overview of managing USB devices in the Solaris environment.

Chapter 2 provides step-by-step instructions for managing USB devices in the Solaris environment.

Related Books

See the following books for additional information on managing devices and printers in the Solaris environment.

- “Managing Devices Topics” in *System Administration Guide, Volume 1*
- “Managing Printing Services Topics” in *System Administration Guide, Volume 2*
- *Solaris 8 Reference Manual Collection*

Ordering Sun Documents

Fatbrain.com, an Internet professional bookstore, stocks select product documentation from Sun Microsystems, Inc.

For a list of documents and how to order them, visit the Sun Documentation Center on Fatbrain.com at <http://www1.fatbrain.com/documentation/sun>.

Accessing Sun Documentation Online

The docs.sun.comSM Web site enables you to access Sun technical documentation online. You can browse the docs.sun.com archive or search for a specific book title or subject. The URL is <http://docs.sun.com>.

Typographic Conventions

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

TABLE P-1 Typographic Conventions

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

Shell Prompts in Command Examples

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

TABLE P-2 Shell Prompts

Shell	Prompt
C shell prompt	machine_name%
C shell superuser prompt	machine_name#
Bourne shell and Korn shell prompt	\$
Bourne shell and Korn shell superuser prompt	#

USB Devices (Overview)

This chapter provides an overview of managing USB devices in the Solaris environment.

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

- “Overview of USB Devices” on page 9
- “About USB in the Solaris Environment” on page 13
- “USB Printer Support” on page 16

For general information about device management in Solaris, see “Device Management (Overview)” in *System Administration Guide, Volume 1*.

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, one way. Devices can connect to hub devices, which connect several devices, including other hub devices. The primary design motivation for USB is to alleviate the need for multiple connector types for different devices, thereby reducing the clutter on the back panel of a system. Additional advantages of using USB devices are:

- USB devices are hot-pluggable. See “Hot-Plugging USB Devices” on page 15 for more information.
- Supports a maximum of 126 devices in the Solaris environment.
- Supports a maximum of 12 Mbit/sec data transfer.
- Supports low speed (1.5 Mbit/sec) and full speed (12 Mbit/sec) devices.

- The bus can be easily extended by adding low-cost external hubs. Hubs can be connected to hubs to form a tree topology.

Sun Microsystems support for USB devices includes the following:

- Sun Blade™ 100 and Sun Blade 1000 systems that run the Solaris 8 10/00, Solaris 8 1/01, Solaris 8 4/01, Solaris 8 7/01, or the Solaris 8 10/01 release provide USB device support.
- Sun Ray™ systems also support USB devices.
- IA systems that run the Solaris 8 Intel Platform Edition provide USB support for keyboard and mouse devices, and for certain mass-storage devices, such as Zip drives. See `scsa2usb(7D)` for more information.

This table provides a listing of specific USB devices that are supported in the Solaris environment.

These USB Devices	Are Supported on These Systems
Keyboards and mouse devices	SPARC systems with Sun USB support based on the <code>ohci(7D)</code> controller. IA systems with a USB bus based on the <code>uhci(7D)</code> controller. Only onboard USB controllers are supported. Plug-in host controller PCI cards are not supported.
Mass storage	SPARC and IA.
Printers	SPARC and IA.
Hub	SPARC and IA.

Commonly Used USB Acronyms

The following table describes the USB acronyms that are used in the Solaris environment. See <http://www.usb.org> for a complete description of USB components and acronyms.

Acronym	Definition
USB	Universal Serial Bus
USBA	Universal Serial Bus Architecture (Solaris)
USBAI	USBA Client Driver Interface (Solaris)

Acronym	Definition
HCD	USB host controller driver

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.

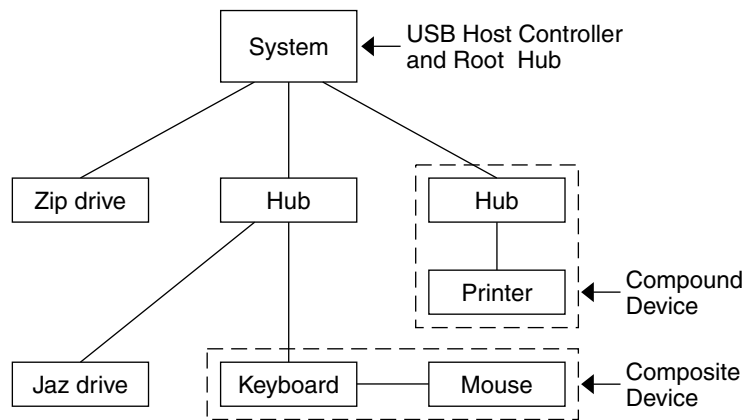


FIGURE 1-1 USB Physical Device Hierarchy

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. See “USB Hub Devices” on page 14 for more information about hubs.

The previous example shows a system with three active USB ports. The first USB port has a Zip drive that does not have an embedded hub, so you cannot attach additional devices. The second USB port has a hub with a Jaz drive and a composite keyboard/mouse device connected. One of the ports from the secondary hub has a keyboard with an embedded hub where the mouse is attached.

The device tree path name for some of the devices that are displayed in the previous example are listed in this table.

Zip drive	/pci@1f,4000/usb@5/storage@1
Keyboard	/pci@1f,4000/usb@5/hub@2/keyboard@1
Mouse	/pci@1f,4000/usb@5/hub@2/mouse@2
Jaz drive	/pci@1f,4000/usb@5/hub@2/storage@3

Printer	/pci@1f,4000/usb@5/hub@3/printer@1
---------	------------------------------------

USB Devices and Drivers

The USB devices are divided into device classes. Each device class has a corresponding driver. Devices within a class are managed by the same device driver. However, the USB specification also allows for vendor-specific devices that are not part of a specific class. Devices with similar attributes and services are grouped.

The Human Interface Device (HID) class contains devices that are user controlled such as keyboards, mouse devices, and joysticks. The Communication Device class contains devices that connect to a telephone, such as modems or an ISDN interface. Other device classes include the Audio, Monitor, Printer, and Storage Device classes. 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 the <http://www.usb.org> site.

Solaris USB Architecture (USBA)

USB devices are represented as two levels of device tree nodes. A device node represents the entire USB *device*, and one or more child *interface* nodes represent the individual USB interfaces on the device. For special cases, the device and interface nodes are *combined* into a single combined node.

Driver binding is achieved by using the compatible name properties. Refer to 3.2.2.1 of the IEEE 1275 USB binding and *Writing Device Drivers* for more information. A driver can either bind to the entire device and control all the interfaces, or a driver can bind to just one interface, for example, a keyboard or mouse. 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 1275 binding.

Figure 1–1 shows an example of a hub and printer as a *compound device*. Both the hub and the printer are enclosed in the same plastic case, but the hub and the printer have separate USB bus addresses. The same diagram shows an example of a *composite device*. The composite keyboard and controller are also enclosed in the same plastic case, but they have the same USB bus address. A cable connects the USB mouse to the composite keyboard/controller in this example.

The Solaris USB Architecture (USBA) adheres to the USB 1.0 and 1.1 specification plus Solaris driver requirements. The USBA model is similar to Sun Common SCSI Architecture (SCSA). The USBA is a thin layer that provides a generic USB transport-layer abstraction to the client driver.

The differences between SCSA and USBA are that the SCSA relies on `.conf` files to probe the bus, while USB hub drivers are self-probing nexus drivers.

About USB in the Solaris Environment

The following sections describe specific information you should know about USB in the Solaris environment.

USB Keyboards and Mouse Devices

Keep only one USB keyboard and mouse on the system at all times because multiple USB keyboards and mouse devices are not supported in the Solaris environment. See the following items for specific details.

- A keyboard and mouse that are connected anywhere on the bus are configured as console keyboard and mouse. Booting the system is slower if the keyboard and mouse are not on the root hub.
- You can move a console keyboard and mouse to another hub at any time *after* a system reboot. You cannot move the console keyboard and mouse *during* a reboot or at the `ok` prompt. After you plug in the keyboard and mouse, they are fully functional again.
- **SPARC only** – The power key on a USB keyboard behaves differently than the one on the Sun Type-5 keyboard. On a USB keyboard, you can suspend or shut down the system by using the `SUSPEND/SHUTDOWN` key, but you cannot power-on the system.
- The left side of the keypad functionality is unavailable on non-Sun USB keyboards.
- Multiple keyboards are not supported:
 - The keyboards enumerate and are usable, but they are not plumbed as console keyboards.
 - The first keyboard that is probed at boot time becomes the console keyboard. The result of this probing might cause confusion if multiple keyboards are plugged in at boot time.
 - If you unplug the console keyboard, the next available USB keyboard doesn't become the console keyboard. The next hot-plugged keyboard becomes the console keyboard.
- Multiple mouse devices are not supported:
 - The mouse devices enumerate and are usable, but they are not plumbed as console mouse devices.
 - The first mouse that is probed at boot time becomes the console mouse. The result of this probing might cause confusion if you have multiple mouse devices plugged in at boot time.
 - If you unplug the console mouse, the next available USB mouse doesn't become the console mouse. The next hot-plugged mouse becomes the console mouse.

- If you have a non-Sun (third-party) composite keyboard with a PS/2 mouse, and it is the first one to be probed, it becomes the console keyboard/mouse even if the PS/2 mouse is not plugged in. This means another USB mouse plugged into the system cannot work because it is not configured as the console mouse.
- Only two-button and three-button mouse devices are supported. A wheel-on-wheel mouse acts like a plain-button mouse. A mouse with more than three buttons functions like a three-button mouse.

USB Host Controller and Root Hub

A USB hub is responsible for:

- 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 back panel are the ports of the root hub. The USB host controller is responsible for:

- Directing the USB bus. Individual devices cannot arbitrate for the bus.
- Polling the devices by using a polling interval 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 or IA systems. On SPARC systems, the Open Boot PROM (OBP) cannot reliably probe beyond four levels of devices.
- Do not cascade bus-powered hubs. This means you cannot plug a bus-powered hub into another bus-powered hub. A bus-powered hub does not have its own power supply. A USB diskette device derives all its power from the bus and might not work on a bus-powered hub.

USB Storage Devices

Removable mass storage devices such as USB Zip, Jaz, Klik!, SmartMedia, CompactFlash, and ORB are supported, starting with the Solaris 8 10/00 release. See `scsa2usb(7D)` for a complete list of devices that are supported in the Solaris environment.

These devices can be managed with or without volume management. See `vold(1M)` for information on managing devices with volume management.

SPARC Only: USB Power Management

If the system has enabled power management, the USB framework makes a best effort to power-manage all devices. Power-managing a USB device means the hub driver suspends the port to which the device is connected. The device might or might not support remote wakeup. If the device supports remote wakeup, it wakes up the hub it is connected to, depending on the event, such as moving the mouse. The host system could also wake the device if an application sends an I/O to it.

All HID (keyboard, mouse, and so forth), hub, and storage devices are power-managed by default if they support the remote wakeup capability. A USB printer is power-managed only between two print jobs.

When you power-manage to reduce power consumption, USB leaf devices are powered down first, and after some delay, the parent hub is powered down. When all devices that are connected to this 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.

Hot-Plugging USB Devices

When you plug in a USB device, the device is immediately seen in the system's device hierarchy, as displayed in the `prtconf(1M)` command output. When you remove a USB device, the device is removed from the system's device hierarchy, unless the device is in use.

If the USB device is in use when it is removed, the hot-plug behavior is a little different. If a device is in use 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 is returned with an error.

In this situation, the system prompts you to plug in the original device. To recover from accidentally removing a busy USB device, do the following:

1. Plug the original device into the same port.
2. Stop the application that is using the device.
3. Remove the device.

The USB port remains unusable until the original device has been plugged in again. If the device is no longer available, the port remains unusable until the next reboot.

Note – Data integrity might be impaired if you remove an active or open device. Always close the device before removing, except the console keyboard and mouse, which can be moved while active.

USB Cables

Never use USB cable extenders that are available in the market. Always use a hub with longer cables to connect devices. Always use fully rated (12 Mbit/sec) 20/28 AWG cables for connecting USB devices.

USB Printer Support

You can use Solaris Print Manager to set up a USB printer that is attached to a SPARC system with USB ports, starting with the Solaris 8 10/00 release. You can also set up USB printers on IA systems, starting with the Solaris 8 04/01 release.

The new logical device names for USB printers are:

```
/dev/printers/N
```

Therefore, when you add a USB printer to a printer server, select one of these devices for a USB printer under Printer Port on the Add New Attached Printer screen. See the *System Administration Guide, Volume 2* for more information on using Solaris Print Manager to set up printers.

Although the new Solaris USB printer driver supports all USB printer-class compliant printers, a list of recommended PostScript™ printers is in the `usbprn(7D)` man page.

The `usbprn` driver is compliant with non-PostScript printers that utilize third-party PostScript conversion packages like GhostScript. You can obtain conversion packages from the Solaris 8 Software Companion CD, available at <http://www.sun.com/software/solaris/binaries/package.html>.

Refer to the Notes and Diagnostics sections of the `usbprn(7D)` man page for information and cautions about hot-plugging USB printers.

USB Devices (Tasks)

This chapter provides step-by-step instructions for managing USB devices in the Solaris environment.

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

- “How to Mount or Unmount a USB Mass Storage Device With `vold` Running” on page 19
- “How to Remove a Hot-Pluggable USB Mass Storage Device With `vold` Running” on page 20
- “How to Add a Hot-Pluggable USB Mass Storage Device With `vold` Running” on page 20
- “How to Mount or Unmount a USB Mass Storage Device Without `vold` Running” on page 21
- “How to Remove a Hot-Pluggable USB Mass Storage Device Without `vold` Running” on page 22
- “How to Add a Hot-Pluggable USB Mass Storage Device Without `vold` Running” on page 22
- “How to Prepare for Creating Data on or Extracting Data From a USB CD” on page 23
- “How to Add a Hot-Pluggable USB Camera” on page 23
- “How to Add Hot-Pluggable USB Audio Devices” on page 25
- “How to Identify Your System’s Primary Audio Device” on page 26
- “How to Change the Primary USB Audio Device” on page 28
- “How to Remove Unused USB Audio Device Links” on page 29

Managing USB Devices

Removable mass storage devices such as USB Zip, Jaz, Klik!, SmartMedia, CompactFlash, and ORB are supported, starting with the Solaris 8 10/00 release. See `scca2usb(7D)` for a complete list of devices that are supported in the Solaris environment.

These devices can be managed with or without volume management. See `vold(1M)` for information on managing devices with volume management.

The following sections describes the tasks for managing USB devices in the Solaris environment.

Managing USB Mass Storage Devices With `vold` Running

If you are running Solaris Common Desktop Environment (CDE), the USB removable mass storage devices are managed by the Removable Media Manager component of the CDE File Manager. See `dtfile(1)` for more information on the CDE File Manager.

Note – You must include the `/usr/dt/man` in your `MANPATH` variable to display the man pages listed in this section. You must also have `/usr/dt/bin` in your path and have CDE running to use these commands, or have a `DISPLAY` variable set to use these commands remotely.

The following table identifies the commands Removable Media Manager uses to manage storage devices from the CDE environment.

Command	Task
<code>sdtmedia_format(1)</code>	Format and label USB devices
<code>sdtmedia_prop(1)</code>	Display properties of the device
<code>sdtmedia_prot(1)</code>	Change device protection
<code>sdtmedia_slice(1)</code>	Create or modify slices on the device

After the USB device is formatted, it is usually mounted under the `/rmdisk/label` directory. See `rmmount.conf(4)` for details on how to configure removable storage devices.

The following procedures describe how to manage USB mass storage devices with volume management. The device nodes are created under the `/vol/dev` directory. See `scsa2usb(7D)` for more information. The following procedures also describe how to add or remove hot-pluggable 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.

▼ How to Mount or Unmount a USB Mass Storage Device With `vold` Running

1. Display device aliases for all removable mass storage devices, including USB mass storage devices.

```
$ eject -n
.
.
.
rmdisk0 -> /vol/dev/rdisk/c4t0d0/clik40      (Generic USB storage)
cdrom0 -> /vol/dev/rdisk/c0t6d0/audio_cd    (Generic CD device)
zip1 -> /vol/dev/rdisk/c2t0d0/fat32        (USB Zip device)
zip0 -> /vol/dev/rdisk/c1t0d0/zip100       (USB Zip device)
jaz0 -> /vol/dev/rdisk/c3t0d0/jaz1gb       (USB Jaz device)
```

2. Mount a USB mass storage device by using the device aliases listed previously.

```
$ volrmount -i device-alias
```

This example mounts a USB Jaz drive under `/rmdisk/jaz0`.

```
$ volrmount -i jaz0
```

3. Unmount a USB mass storage device.

```
$ volrmount -e device-alias
```

This example unmounts a USB Zip drive from `/rmdisk/zip0`.

```
$ volrmount -e zip0
```

4. Eject a USB device from a generic USB drive.

```
$ eject device-alias
```

For example:

```
$ eject rmdisk0
```

Note – The `eject` command also unmounts the device if it is not unmounted already. The command also terminates any active applications that access the device.

▼ How to Remove a Hot-Pluggable USB Mass Storage Device With `vold` Running

The following procedure uses a Zip drive as an example of removing a hot-pluggable USB device with `vold` running.

1. **Unmount the device.**

```
$ volrmount -e zip0
```

2. **(Optional) Stop any active applications that are using the device.**

3. **Eject the device.**

```
$ eject zip0
```

4. **Become superuser and stop `vold`.**

```
# /etc/init.d/volmgt stop
```

5. **Remove the USB mass storage device.**

6. **Start `vold`.**

```
# /etc/init.d/volmgt start
```

▼ How to Add a Hot-Pluggable USB Mass Storage Device With `vold` Running

This procedure describes how to add a hot-pluggable USB device with `vold` running.

1. **Insert the USB mass storage device.**

2. **Restart `vold`.**

```
# pkill -HUP vold
```

3. **Verify the device has been added.**

```
$ ls device-alias
```

Managing USB Mass Storage Devices Without `vold` Running

You can use USB mass storage devices without the volume manager (`vold`) running. Here are two ways to avoid using the volume manager.

- Stop `vold` by issuing this command.

```
# /etc/init.d/volmgt stop
```

- Keep `vold` running, but do not register the USB mass storage devices with it. Remove volume manager registration of USB mass storage devices by commenting the following line in the `/etc/vold.conf` file, like this:

```
# use rmdisk drive /dev/rdisk/c*s2 dev_rmdisk.so rmdisk%d
```

After this line is commented, restart `vold`.

```
# /etc/init.d/volmgt start
```



Caution – If you comment out this line and other SCSI or ATAPI Zip or Jaz removable devices are in the system, `vold` registration for these devices would be disabled as well.

See `vold.conf(4)` for details.

The following procedures describe how to manage USB mass storage devices without `vold(1M)` running. The device nodes are created under the `/dev/rdsk` directory for character devices and under the `/dev/dsk` directory for block devices. See `scsa2usb(7D)` for details.

How to Mount or Unmount a USB Mass Storage Device Without `vold` Running

1. Become superuser.

2. Mount a USB mass storage device.

```
# mount -F fs-type /dev/dsk/cntndnsn /mount-point
```

This command might fail if the device is read only. Use the following command for CD-ROM devices.

```
# mount -F fs-type -o ro /dev/dsk/cntndnsn /mount-point
```

For example:

```
# mount -F hfs -o ro /dev/dsk/c0t6d0s2 /mnt
```

3. Unmount a USB mass storage device.

```
# umount /mount-point
```

4. Eject the device.

```
# eject /dev/[r]dsk/cntndnsn
```

▼ How to Remove a Hot-Pluggable USB Mass Storage Device Without `vold` Running

This procedure describes how to remove a hot-pluggable USB device without `vold` running.

1. **Become superuser.**
2. **Remove the hot-pluggable USB device.**
 - a. **Unmount the device.**

```
# umount /mount-point
```
 - b. **(Optional) Stop any active applications that are using the device.**
 - c. **Remove the device.**

How to Add a Hot-Pluggable USB Mass Storage Device Without `vold` Running

This procedure describes how to add a hot-pluggable USB device without `vold` running.

1. **Add a hot-pluggable USB device into the USB port.**
2. **Verify the USB device has been added.**

```
$ ls /dev/rdisk/cntndnsn
```

Creating Data on or Extracting Data From a USB CD

You can use the `cdrw` command to create and extract data from audio CDs. The `cdrw` command is available on the Software Supplement for the Solaris 8 Operating Environment 1/01 CD.

- SCSI, ATAPI, and USB CD devices are supported. Currently, the only CD-RW device supported by Sun is the Sony Spress USB CD-RW.
- The CD-R or CD-RW drive must be MMC compliant.

See the `cdrw` man page in the *Solaris on Sun Hardware Reference Manual Supplement* for information on using this command.

▼ How to Prepare for Creating Data on or Extracting Data From a USB CD

The `cdrw` command works with or without `vold` running. See the `cdrw(1)` and `mkisofs(1M)` man pages for more information.

1. Insert a CD into the CD-RW device.

The CD can be any CD that the device can read.

2. Check that the CD-RW drive is connected properly by listing the device.

```
# cdrw -l
Node | Connected Device | Device type
-----+-----+-----
/dev/rdisk/c0t0d0s2 | SONY CD-RW CRX120E 1.0k | CD Reader/Writer
```

3. (Optional) If you do not see the drive in the list, you might have to do a reconfiguration boot so that the system recognizes the device.

```
# touch /reconfigure
# init 6
```

▼ How to Add a Hot-Pluggable USB Camera

Use this procedure to add a hot-pluggable USB camera.

1. Plug in and turn on the USB camera.

The USB subsystem and the SCSA subsystem create a logical device for the camera. After the camera is plugged in, output is written to the console window and the `/var/adm/messages` file.

2. Examine the output written to the console window.

Examining this output enables you to determine what logical device was created so that you can then use that device to access your images. The output will look similar to the following:

```
Jul 15 09:53:35 buffy usba: [ID 349649 kern.info] OLYMPUS, C-3040ZOOM,
000153719068
Jul 15 09:53:35 buffy genunix: [ID 936769 kern.info] scsa2usb1 is
/pci@0,0/pci925,1234@7,2/storage@2
Jul 15 09:53:36 buffy scsi: [ID 193665 kern.info] sd3 at scsa2usb1:
target 0 lun 0
```

3. Mount the USB camera file system.

The camera's file system is most likely a PCFS file system. In order to mount the file system on the device created, the slice that represents the disk must be specified. The slice is normally `s0` for a SPARC system, and `p0` for an Intel system. For example, to mount the file system on an Intel system, execute the following command:

```
% mount -F pcfs /dev/dsk/c3t0d0p0:c /mnt
```

To mount the file system on a SPARC system, execute the following command:

```
% mount -F pcfs /dev/dsk/c3t0d0s0:c /mnt
```

For information on mounting file systems, see “Mounting and Unmounting File Systems (Tasks)” in *System Administration Guide, Volume 1*.

4. Verify that the image files are available.

For example:

```
% ls /mnt/DCIM/100OLYMP/  
P7220001.JPG* P7220003.JPG* P7220005.JPG*  
P7220002.JPG* P7220004.JPG* P7220006.JPG*
```

5. View and manipulate the image files created by the USB camera.

```
% /usr/dt/bin/sdtimage &
```

6. Unmount the file system before disconnecting the camera.

Unmount the file system by executing the following command:

```
% umount /mnt
```

7. Turn off and disconnect the camera.

USB Audio Overview

This Solaris release provides USB audio support which is implemented by a pair of cooperating drivers, `usb_ac` and `usb_as`. The audio control driver, `usb_ac`, a USBA (Solaris USB Architecture) compliant client driver provides the controlling interface to user applications. The audio streaming driver, `usb_as`, is provided to process audio data messages during play and record and set sample frequency, precision, and encoding requests from the `usb_ac` drive.

Both drivers comply to the USB audio class 1.0 specification.

Solaris supports external USB audio devices that are play-only or record-only. Onboard USB audio devices are not supported. See the `usb_ac` man page for supported audio data formats.

- Only USB audio devices with one volume, bass, or treble control are supported. See the USB audio class specification for more information at <http://www.usb.org>.
- USB audio devices are supported on SPARC Ultra and Intel platforms that provide USB connectors.
- Hot-plugging USB audio devices is supported.

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/audioc1:
  Name      = USB Audio
  Version   = 1.0
  Config    = external
```

Audio mixer for `/dev/audioc1` is enabled

After you connect your USB audio devices, you access them with the `audioplay` and `audiorecord` command through the following files:

```
/dev/sound/N
```

You can select a specific audio device by setting the `AUDIODEV` environment variable or by specifying the `-d` option to the `audioplay` and `audiorecord` commands. However, setting `AUDIODEV` does not work for 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. Refer to “How to Change the Primary USB Audio Device” on page 28 and `usb_ac(7D)` for instructions on changing `/dev/audio` from onboard audio to USB audio and vice versa.

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.

See `usb_ac(7D)` for additional information on troubleshooting USB audio device problems.

▼ How to Add Hot-Pluggable USB Audio Devices

Use this procedure to add hot-pluggable USB audio devices.

1. Plug in the USB speakers and microphone.

The primary audio device, `/dev/audio`, usually points to the onboard audio. After you connect USB audio devices, `/dev/audio` points to the USB audio devices that are identified in the `/dev/sound` directory.

2. Verify that the audio device files have been created.

```
% ls /dev/sound
0      0ct1  1      1ct1  2      2ct1
```

3. Test the left and right USB speakers.

```
% cd /usr/demo/SOUND/sounds
% audioplay -d /dev/sound/1 -b 100 spacemusic.au
% audioplay -d /dev/sound/1 -b -100 spacemusic.au
```

4. Test the USB microphone.

```
% cd $HOME/au
% audiorecord -d /dev/sound/2 -p mic -t 30 test.au
```

Troubleshooting USB Audio Device Problems

This section describes how to troubleshoot USB audio device problems.

Audio Device Ownership Key Points

Keep the following audio device ownership key points 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 means you can use the audio device as long as you are logged into the console.
- If you are not logged into the console when you plug in a USB audio device, root becomes the owner of the device. However, if you log into the console and attempt to access the USB audio device, device ownership changes to the console. See `logindevperm(4)` for more information.
- When you remotely login 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.

▼ How to Identify Your System's Primary Audio Device

This procedure assumes that you have already connected USB audio devices.

1. Identify the state of your current audio device links.

For example:

```

% ls -lt /dev/audio*
lrwxrwxrwx  1 root    root      7 Jul 23 15:41 /dev/audio -> sound/0
lrwxrwxrwx  1 root    root     10 Jul 23 15:41 /dev/audiocctl ->
sound/0cctl
% ls -lt /dev/sound/*
lrwxrwxrwx  1 root    other    66 Jul 23 14:21 /dev/sound/0 ->
../../devices/pci@1f,4000/ebus@1/SUNW,CS4231@14,200000:sound,audio
lrwxrwxrwx  1 root    other    69 Jul 23 14:21 /dev/sound/0cctl ->
../../devices/pci@1f,4000/ebus@1/SUNW,CS4231@14,200000:sound,audiocctl
%

```

The primary audio device, `/dev/audio`, is currently pointing to the onboard audio, which is `/dev/sound/0`.

2. (Optional) Add a new USB audio device.

3. Examine your system's new audio links.

For example:

```

% ls -lt /dev/audio*
lrwxrwxrwx  1 root    root      7 Jul 23 15:46 /dev/audio -> sound/1
lrwxrwxrwx  1 root    root     10 Jul 23 15:46 /dev/audiocctl ->
sound/1cctl
% ls -lt /dev/sound/*
lrwxrwxrwx  1 root    root     74 Jul 23 15:46 /dev/sound/1 ->
../../devices/pci@1f,4000/usb@5/hub@1/device@3/sound-control@0:sound,audio
lrwxrwxrwx  1 root    root     77 Jul 23 15:46 /dev/sound/1cctl ->
../../devices/pci@1f,4000/usb@5/hub@1/device@3/sound-control@0:sound,aud...
lrwxrwxrwx  1 root    other    66 Jul 23 14:21 /dev/sound/0 ->
../../devices/pci@1f,4000/ebus@1/SUNW,CS4231@14,200000:sound,audio
lrwxrwxrwx  1 root    other    69 Jul 23 14:21 /dev/sound/0cctl ->
../../devices/pci@1f,4000/ebus@1/SUNW,CS4231@14,200000:sound,audiocctl
%

```

Notice that the primary audio device, `/dev/audio`, is pointing to the newly plugged in USB audio device, `/dev/sound/1`.

If you remove the USB audio device now, the primary audio device, `/dev/audio`, does not revert back to the onboard audio. See the procedure below for instructions on changing the primary audio device back to the system's onboard audio.

You can also examine your system's USB audio devices with the `prtconf` command and look for the USB device information.

```

% prtconf
.
.
.
usb, instance #0
  hub, instance #0
  mouse, instance #0
  keyboard, instance #1
  device, instance #0
    sound-control, instance #0
    sound, instance #0
    input (driver not attached).

```

.
.
.

▼ How to Change the Primary USB Audio Device

Follow the steps below if you remove or change your USB audio devices and you want to make one particular audio device the primary audio device. The procedure changes the primary audio device to the onboard audio device as an example.

1. **Become superuser.**
2. **Close all audio applications.**
3. **Determine whether the audio and USB drivers are loaded.**

```
# modinfo | grep -i audio
124 780e6a69 bb6e - 1 audiosup (Audio Device Support 1.12)
# modinfo | grep -i usb
48 13dba67 18636 199 1 ohci (USB OpenHCI Driver 1.31)
49 78020000 1dece - 1 usba (USBA: USB Architecture 1.37)
50 12e5f1f 35f 195 1 hubd (USB Hub Driver 1.4)
51 13ef53d 5e26 194 1 hid (USB HID Client Driver 1.16)
54 13f67f2 1b42 10 1 usbms (USB mouse streams 1.6)
56 127bbf0 2c74 11 1 usbkm (USB keyboard streams 1.17)
#
```

4. **Load and attach the onboard audio driver.**

```
# devfsadm -i audiocs
```

5. **Verify the primary audio device link is pointing to the onboard audio.**

```
# ls -lt /dev/audio*
lrwxrwxrwx 1 root other 7 Jul 23 15:49 /dev/audio -> sound/0
lrwxrwxrwx 1 root other 10 Jul 23 15:49 /dev/audioc1 ->
sound/0c1
# ls -lt /dev/sound/*
lrwxrwxrwx 1 root other 66 Jul 23 14:21 /dev/sound/0 ->
../../devices/pci@1f,4000/ebus@1/SUNW,CS4231@14,200000:sound,audio
lrwxrwxrwx 1 root other 69 Jul 23 14:21 /dev/sound/0c1 ->
../../devices/pci@1f,4000/ebus@1/SUNW,CS4231@14,200000:sound,audioc1
#
```

6. **Confirm the onboard audio is the primary audio device.**

```
% audioplay /usr/demo/SOUND/sounds/bark.au
```

The audioplay command defaults to the onboard audio device.

7. **(Optional) Unload all the audio drivers that can be unloaded before plugging in another USB audio device.**

- a. **Close all the audio applications.**

b. Display the audio driver information to verify no audio drivers are currently loaded.

```
# modinfo | grep -i audio
60 78048000 bb6e - 1 audiosup (Audio Device Support 1.12)
61 78152000 39a97 - 1 mixer (Audio Mixer 1.49)
62 78118000 bf9f - 1 amsrc1 (Audio Sample Rate Conv. #1 1.3)
128 7805e000 14968 54 1 audiocs (CS4231 mixer audio driver 1.21)
#
```

c. Unload the audio drivers.

```
# modunload -i 0
# modinfo | grep -i audio
60 78048000 bb6e - 1 audiosup (Audio Device Support 1.12)
61 78152000 39a97 - 1 mixer (Audio Mixer 1.49)
#
```

At this point, `audiocs`, the onboard audio driver, has been unloaded and guaranteed not to be open. However, the primary audio device, `/dev/audio`, does not change if it is held open by an application.

8. (Optional) Plug in a USB audio device.

9. (Optional) Examine the new audio links.

```
% ls -lt /dev/audio*
lrwxrwxrwx 1 root root 7 Jul 23 16:12 /dev/audio -> sound/1
lrwxrwxrwx 1 root root 10 Jul 23 16:12 /dev/audioc1 ->
sound/1c1
% ls -lt /dev/sound/*
lrwxrwxrwx 1 root root 77 Jul 23 16:12 /dev/sound/1c1 ->
../../devices/pci@1f,4000/usb@5/hub@1/device@3/sound-control@0:sound,aud...
lrwxrwxrwx 1 root root 74 Jul 23 16:12 /dev/sound/1 ->
../../devices/pci@1f,4000/usb@5/hub@1/device@3/sound-control@0:sound,aud...
lrwxrwxrwx 1 root root 66 Jul 23 15:59 /dev/sound/0 ->
../../devices/pci@1f,4000/ebus@1/SUNW,CS4231@14,200000:sound,audio
lrwxrwxrwx 1 root root 69 Jul 23 15:59 /dev/sound/oct1 ->
../../devices/pci@1f,4000/ebus@1/SUNW,CS4231@14,200000:sound,aud...
%
```

▼ How to Remove Unused USB Audio Device Links

Use the procedure below if a USB audio device is removed while the system is powered off. It is possible that removing the USB audio device while the system is powered off will leave the `/dev/audio` device still pointing to a `/dev/sound/*` device that doesn't exist.

1. Become superuser.
2. Close all audio applications.
3. Remove the unused audio links.

```
# devfsadm -C -c audio
```

Index

A

- adding
 - hot-pluggable USB camera, 23
 - USB audio device, 25
 - USB mass storage device with vold running, 20
 - USB mass storage device without vold running, 22
- audio, USB, 24

C

- cables for USB devices, 16
- changing, primary USB audio device, 28
- composite USB device, 12
- compound USB device, 12

D

- description of USB bus, 11
- device classes of USB devices, 12
- device names of USB devices, 11
- device nodes of USB devices, 12

H

- hot-plugging USB devices, 15
- hub devices, 9

M

- maximum USB device support, 9
- maximum USB transfer rate, 9
- mounting
 - USB mass storage devices with vold running, 19
 - USB mass storage devices without vold running, 21

O

- overview of USB devices, 9

P

- power management of USB devices, 15
- printers with USB support, 16

R

- removing
 - unused USB audio device links, 29
 - USB mass storage device with vold running, 20
 - USB mass storage device without vold running, 22

S

- Solaris USB architecture, 12
- systems with USB device support, 10

U

- unmounting
 - USB mass storage devices with vold running, 19
 - USB mass storage devices without vold running, 21
- USB acronyms, 10
- USB audio device
 - adding a, 25
 - changing the primary, 28
- USB audio device ownership, 26
- USB audio devices, 24
 - removing unused links, 29
- USB bus description, 11
- USB camera, adding, 23
- USB device names, 11
- USB devices
 - adding a hot-plug a USB camera, 23
 - adding a USB mass storage device
 - with vold running, 20
 - without vold running, 22
 - audio
 - overview of, 24
 - audio device
 - adding a, 25
 - changing the primary, 28
 - identifying primary, 26
 - audio device ownership, 26
 - audio devices
 - removing unused links, 29
 - cables for, 16
 - device classes, 12
 - device names, 11
 - device nodes, 12
 - driver information, 12
 - drivers, 12
 - host controller and root hub, 14
 - hot-plugging, 15
 - hub devices, 14
 - keyboards and mouse devices, 13
 - mass storage device
 - mounting with vold running, 19

- USB devices, mass storage device (Continued)
 - mounting without vold running, 21
 - unmounting with vold running, 19
 - unmounting without vold running, 21
 - maximum devices supported, 9
 - maximum transfer rate, 9
 - overview, 9
 - power management, 15
 - printers, 16
 - removing a USB mass storage device
 - with vold running, 20
 - without vold running, 22
 - storage devices, 15
 - supported devices, 10
 - systems with USB device support, 10
- USB host controller, 14
- USB hub devices, 14
- USB keyboards, 13
- USB mouse devices, 13
- USB physical device hierarchy, 11
- USB power management, 15
- USB printers, 16
- USB root hub, 14
- USB storage devices, 15
- USBA, Solaris USB Architecture, 12