

Read This First

Software READ THIS FIRST Release 3.5

Getting Help

If you have any problems installing or using this release, call Sun Microsystems at: 1-800-USA-4SUN (1-800-872-4786). Have your system's model number, release number (for software), and Sun operating system (SunOS™) release number ready to give to the dispatcher.

You can also send questions by electronic mail to `sun!hotline`. Be sure to include your name, company, phone number, release number, and SunOS release number in your mail message.

If you have questions about Sun's support services or your shipment, call your sales representative.

To see the SunOS release number, type: `cat /etc/motd`

Upgrading to 3.5

When you are upgrading client partitions to 3.5, ignore the following error message:

```
x ./etc/yppbind, 46376 bytes, 91 tape blocks
./lib: File exists
tar: can't create ./lib/: No such file or directory
x ./private/usr/adm/aculog, 0 bytes, 0 tape blocks
```

Installing an ALM-2

Although SunOS system software supports the ALM-2 board, there is still a small amount of configuration you must perform before using the ALM-2 in your system. This involves creating the special device entries in the file system needed to use your ALM-2 board(s).

After installing your system software and configuring your kernel as described in *Installing UNIX on the Sun Workstation*, change to the `/dev` directory. Then use the command `MAKEDEV` to create the necessary special device entries for Sun standard and optional hardware.

To create the necessary device entries:

1. Determine how many ALM-2 boards are present in your system. (There will be between 1 and 4.)
2. Run `MAKEDEV` with arguments corresponding to the boards in your as shown below:

If there are four boards in your system enter:

```
hostname# MAKEDEV mcp0 mcp1 mcp2 mcp3
```

If there are two boards in your system enter:

```
hostname# MAKEDEV mcp0 mcp1
```

(The ALM-2 is known internally to the system as `mcp`.)

This step creates a set of device entries in the form of `/dev/ttyxy`, where `x` will be a letter `h`, `i`, `j`, or `k`. These letters refer to each ALM-2 board: `h` is the first board (`mcp0`), `i` is the second, and so on. `y` is a hexadecimal digit in the range 0-9, a-f which specifies which of the sixteen serial lines on an ALM-2 board is being accessed. Thus the third line on the third board is `/dev/ttyj2`, and the twelfth line on the first board is `/dev/ttyhb`.

For more information on installing modems, configuring terminal lines, and so on, refer to *System Administration for the Sun Workstation—Adding Hardware to Your System*. For more information on the ALM-2 device entries, refer to the ALM-2 manual page `mcp(4s)`.

Installing Servers

When installing your 3.5 tape set, you must load the optional *networking tools and programs* if you are installing a server. Otherwise, the server will be unable to run the Yellow Pages, and diskless clients will be unable to boot.

Installing a File Server as a Remote Host

If you plan to use the file server you are installing as a remote host for remote installation of another workstation, you will need an additional 5 Mbytes of free space in one of your file systems, for example, `/pub`.

Major Bug Fixed in Release 3.5

Incorrect `last` Report

`last` no longer reports that you are still logged in after the `login` process is complete.

Software Modifications for MCP

If you are using SunLink MCP on your system, you must modify your kernel configuration if you are running:

- Sun 3/110, 3/120 -P/N 501-1134-01 through 05
- Sun 3/140, 3/160, 3/180 -P/N 501-1164-01 through 08

The following procedure should be considered a workaround in the event that the appropriate ECO's as specified by Field Alert #111, were not applied to your equipment. (Consult your local Field Engineer to verify the ECO level of your equipment if you have any doubt.)

When installing an MCP under SunOS 3.5 or later, incorporate the following instructions at the BEGINNING of Step 2 of Section 2.3 of the *MCP Software Installation and Configuration Guide*.

1. Change to the directory that contains the kernel configuration file you are using. Usually this is under `/usr/sys/conf` and the configuration file is named `GENERIC`, or is specific to your machine and named the same as your machine.

2. Create a copy of the GENERIC configuration file. Use this copy as a working file, keeping the original as a back-up.
3. In your GENERIC configuration file, locate the four mcp configuration lines.

```
device      mcp0 at vme32d32 ? csr 0x01000000 flags 0x1ffff priority 4
vector mcpintr 0x8b
device      mcp1 at vme32d32 ? csr 0x01010000 flags 0x1ffff priority 4
vector mcpintr 0x8a
device      mcp2 at vme32d32 ? csr 0x01020000 flags 0x1ffff priority 4
vector mcpintr 0x89
device      mcp3 at vme32d32 ? csr 0x01030000 flags 0x1ffff priority 4
vector mcpintr 0x88
```

4. For each MCP board to be installed, determine the board number (0, 1, 2, or 3).
 5. Edit the line corresponding to each board, and change the text, vme32d32 to read vme32d16.
- For example, the line corresponding to MCP board 0, (the first MCP board installed), would now read:

```
device      mcp0 at vme32d16 ? csr 0x01000000 flags 0x1ffff priority 4
vector mcpintr 0x8b
```

6. Now continue with the remaining instructions of Step 2 from the *MCP Software Installation and Configuration Guide*.

Reinstalling SunLink

Whenever you upgrade to a new version of the SunOS, you must *reinstall all SunLink* products that modify the standard kernel or provide custom versions of standard SunOS files.

The only exceptions are SunLink TE100 and SunLink MCP. SunLink TE100 contains no kernel modules so you do not need to reinstall it.

With SunOS 3.5, the 3.5 mcp_async.o should be used instead of the SunLink MCP mcp_async.o module. So, if you have a SunLink MCP, after upgrading to SunOS 3.5, copy the kernel module /usr/sunlink/mcp/sys/mcp_async.o to a directory where it will not be overwritten when you reinstall the MCP. Then, reinstall the MCP and move the 3.5 mcp_async.o back to /usr/sunlink/mcp/sys.

Documentation Errata and Additions

In Chapter 2 of the *Release 3.5 Manual for the Sun Workstation*, on page 45, the name entry should read:

name=machine name to perform the operation on

The release entry should read:

release=release the machine is currently running, (i.e., 3.2, 3.4, 3.5Beta)

Obtaining Pre-formatted Manual Pages

If you want to maintain formatted manual pages on your system, you should do the following:

1. Completely install Release 3.5 and reconfigure the kernel, as described in Chapters 2 and 3 of *Release 3.5 Manual for the Sun Workstation*.
2. Ensure that you have 8 Mbytes of available disk space available in the file system that contains the `/usr/man` directory, which is usually located in the `/usr` file system.
3. Run `catman(8)`. `catman` formats the man pages and builds the `whatis` database.

For more information about `catman(8)` and `whatis(1)`, refer to their respective man pages.

Sun Abort Sequence Bug

If you abort the boot sequence on some Sun machines, attempting to boot from tape may result in the output of a long series of capital Y's until the boot is again aborted. The workaround is to enter `k1` at the boot PROM prompt (`>`).

This bug seems to result from the keyboard UART being left in an improper state and therefore will not be seen on machines using terminals rather than Sun monitors/keyboards.

Booting from SMD disks

When booting mini-UNIX on a VME-based workstation from a Xylogics disk controller jumpered for 20-bit addressing, the kernel may panic with the following error:

```
unable to mount root
```

This happens when booting for the first time. When booted a second time (without reloading mini-UNIX), the workstation should boot without any problems.

Normally pins JM on the controller are jumpered for 24-bit addressing.

Using `echo -n` in Bourne Shell Scripts

If you specify System V behavior (for example, `PATH=/usr/5bin:$PATH`), shell scripts that contain lines such as

```
echo -n args
```

will echo

```
-n args
```

with a trailing newline. Specify a path without `/usr/5bin` to get the 4.2BSD behavior.

Undocumented Bugs Fixed Since Release 3.2

The following updates were made to the release software after the *Release 3.4 Manual for the Sun Workstation* was printed.

Changes to *Bug Fixes Since Release 3.3*

Language-Related Bug Fixes

Correct Evaluation of Bit Fields

Release 3.2, `cc` incorrectly evaluated a bit field compared against an integer constant with the value of 0. This was fixed in Release 3.4.

Compiler Lost Track of Register Variables

The C compiler sometimes lost track of register variables after function calls. For example, if you declared a program register variable `a` and assigned it to a register, and then, in another function, declared another register variable `b`, the compiler would incorrectly assign `b` to the same register. The value of `a` would then be lost. This problem was fixed in Release 3.4.

Window Bug Fixes

`TEXTSW_INSERTION_POINT` Takes Effect Immediately

In previous releases, if you set the `TEXTSW_INSERTION_POINT` attribute, the caret did not display in the new window until the user moved the mouse cursor out of, then back into the text subwindow. The caret now jumps to the new window position immediately (though, as before, if the new location is not in the window the caret will not be visible unless you call `textsw_possibly_normalize()`).

`cmdtool` Handles Child's Exit Correctly

Release 3.2, if you 'Quit' a `cmdtool` while still running a program from its shell (for example, listing a long file using `cat`), the program would still be running. If you then started a new `cmdtool`, if the new `cmdtool` happened to use the same `pty` as the previous one you 'Quit,' then the new `cmdtool` would get a `SIGHUP` signal from the previous shell and it would exit. Now when you 'Quit' from `cmdtool`, the `cmdtool` sends a `SIGHUP` signal to make the child exit, so a new invocation of `cmdtool` should not encounter the previous `cmdtool`'s children.

