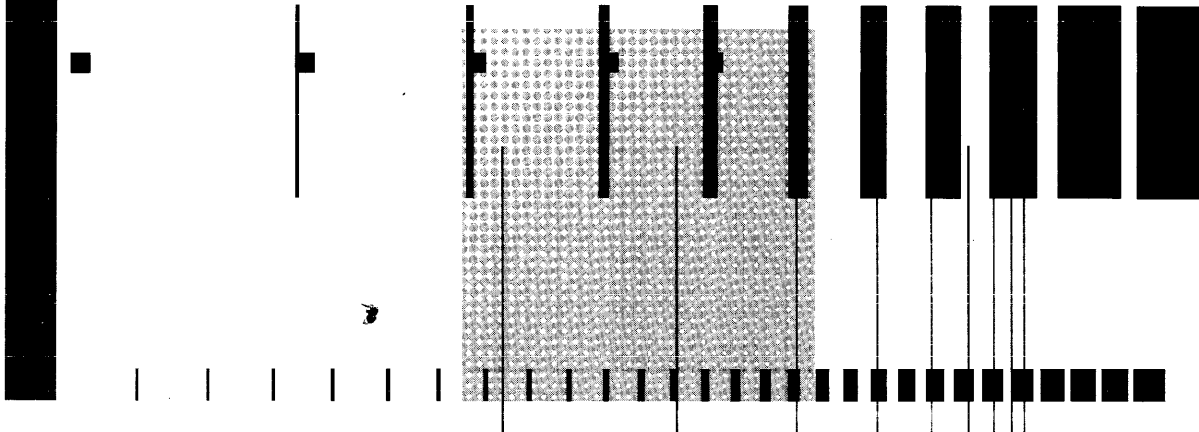# MIPS
Software Binary Release Notes
RISCos 4.50 Release

75-00118 (A)

mips

The power of RISC is in the system.

MIPS
Software Binary Release Notes
RISCos 4.50 Release

75-00118 (A)

UNIX is a registered trademark of AT&T
Ethernet is a trademark of XEROX
NFS is a trademark of SUN Microsystems
RISC/os is a trademark of MIPS Computer Systems, Inc.
RISCompiler is a trademark of MIPS Computer Systems, Inc.
RISComputer is a trademark of MIPS Computer Systems, Inc.
RISCstation is a trademark of MIPS Computer Systems, Inc.
RISCwindows is a trademark of MIPS Computer Systems, Inc.

1. RISC/os 4.50 Errata

1.1 tftpd

There is a bug in the RISC/os 4.50 version of **tftpd** that causes it to always be invoked with the **-s** (secure) option. Because of this, **tftpd** will not work if there are any symbolic links in the directory **tftpd** will use (that is, the default **tftpboot** directory or any directory specified by the **-d** option to **tftpd**), because **tftpd** does not follow symbolic links.

1.2 Installing on an Rx3230

Because the Rx3230 kernel is one of the last kernels on the tape, **sash** sometimes times out when trying to boot the kernel. If this happens, the work around is to ctrl-c to the prom monitor prompt, and then boot the kernel from there, as follows:

```
sash: ^c (ctrl-c)
>> boot -f tqsd(0,6,11)unix.r3030_std root=sdc0d0s1
```

1.3 Login Quota Warnings

If disk quota warning messages are not displayed upon login, this feature may need to be enabled by creating an empty */etc/login.quotawarn.ok* file. This file acts as a global switch to the quota warning messages. On systems with many NFS mounted filesystems which have disk quotas enabled, logging in may be very slow.

1. Introduction

This document contains the information required to install and run RISC/os 4.50 on a MIPS system. RISC/os 4.50 is based on AT&T System V Release 3.2 UNIX™ release with substantial enhancements, including NFS 4.0. For any other version of RISC/os, see the release notes specific to that release.

Since RISC/os 4.00 was released, there have been three minor releases based on RISC/os 4.00 to support new hardware platforms:

    RISC/os 4.10 for the Rx2030
    RISC/os 4.20 for the RC3260
    RISC/os 4.30 for the RC3240

In addition to other new features, RISC/os 4.50 merges all of these operating systems into one. RISC/os 4.50 is compatible with older UMIPS/V and RISC/os binaries, but any binaries built on a UMIPS-BSD system must be recompiled. Because of header file changes between pre-4.50 and 4.50, it is not generally possible to cross link objects made with 2.0 and 2.11 compilers. In general, RISC/os binaries will not run on any earlier versions of RISC/os or UMIPS. In particular, binaries made on RISC/os 4.50 will not run on any previous RISC/os releases.

RISC/os 4.50 conforms to System V Interface Definition (SVID), Issue 2, Base Systems and Kernel Extensions, with the exception that MIPS supports IEEE P754 Floating Point Standard. MIPS does not yet claim strict conformance to additional SVID extensions, the Basic Utilities, Advanced Utilities, Software Development, Administered Systems or Terminal Interface. In many cases, this release does conform at this point. Please see **Appendix A** for a more detailed summary of the functionality and conformance provided in this release.

1.1 Using RISC/os Release Notes

The main purpose of MIPS release notes is to describe new features of the release, problems resolved, known problems, and installation instructions. The release notes are divided into the following sections:

Introduction
An overview of the release, release requirements, and the testing matrix.

General MIPS Release Information
Describes a MIPS software installation, binary kernel reconfiguration, lists some troubleshooting tips, and explains how to report problems to MIPS. This section should be read by customers new to MIPS, and referred to as necessary by users familiar with MIPS.

RISC/os 4.50
Describes new features, problems resolved, and known problems in the current release. Everyone should read this section.

Installation Instructions
Contains transcriptions of update, remote tape, and scratch installations; and should be followed while performing the RISC/os installation.

Post-Installation Modifications
Describes changes that should be made to the system before it is used with the new operating system.

1.2 Release Identification Information

Software Version
    RISC/os 4.50
Release Date
    June 14, 1990
MIPS Part Numbers:
    01-00116 RISC/os 4.50
    75-00118 RISC/os 4.50 Release Notes

1.3 Release Requirements

The RISC/os System Requirements are:

Package	Approximate Disk Space	Inodes
root	7.99Mb	404
r2030	5.19Mb	757
r2300	5.33Mb	436
r2400	5.30Mb	776
r3200	5.40Mb	748
r3260	5.40Mb	748
r3030	5.42Mb	773
r6000	5.43Mb	10
r6280	0Mb	738
usr	40.84Mb	2717
bsd43	13.84Mb	600
reconfig	23.14Mb	582
compat	14.93Mb	1069
cmplrs	6.29Mb	368
cmplrs-bsd43	1.36Mb	88
posix	2.38Mb	72
man	6.50Mb	2269
mh	13.92Mb	63
emacs	8.25Mb	446
uucp	2.15Mb	56
sccs	.14Mb	3
news_readers	.61Mb	21
games	2.89Mb	39
Total for */usr* packages	137.24Mb	8393

An update install requires approximately .5 megabyte of additional space on the root filesystem for an update install, and if all subpackages are installed, approximately 35 additional megabytes on the */usr* filesystem.

1.4 Compilers and Development Tools

RISC/os 4.50 includes the 2.11 base compiler (the C compiler front-end and the back-end components that are common to all MIPS compilers), which provides fixes for bugs reported throughout the compiler and tools. The 2.11 C compiler was used to compile the RISC/os 4.50 kernel, the compilers, and commands, all optimized.

The 2.11 compilers are simply the 2.10 sources recompiled with RISC/os 4.50 header files, but note that this does make a significant difference in some cases. Do not try to link objects

2

built with 4.50 include files with objects built with pre-4.50 includes. To use the 2.0 or 2.10 compilers and the RISC/os 4.00 build environment on RISC/os 4.50, you must link with the corresponding header files and libraries. This is accomplished by setting the environment variable COMP_TARGET_ROOT to /RISCos4.0 as follows:

```
# csh
# setenv COMP_TARGET_ROOT /RISCos4.0
```

For more information on building this way, see the section titled **RISC/os 4.00 Compatibility Package**. For more information on the base compiler, see the *2.11 MIPS-C RISCompiler Release Notes*, (part number 75-00123), included with this release.

1.5 Testing

The following is a list of system configurations tested for the RISC/os 4.00 release:

- M/2000-6 RISComputer with 20MHz CPU, floating point unit, 128KB of I&D caches, 32MB main memory, console port, one 4400 SMD disk controllers, disk tray with two 665MB formatted 8" disks, 6250/1600bpi 1/2" magnetic tape drive, QIC-120 cartridge tape with SCSI controller, two block mode Ethernet controllers and 8 RS-232C user ports.

- M/2000-6 RISComputer with 20MHz CPU, floating point unit, 128KB of I&D caches, 32MB main memory, console port, two dual SMD 4200 block mode disk controllers, disk tray with four 665MB formatted 8" disks, three SMD 4400 block mode disk controllers, 8 665MB formatted 8" disks, two 6250/1600bpi 1/2" magnetic tape drive, QIC-120 cartridge tape with SCSI controller, and one 2GByte 8mm cartridge drive, two block mode Ethernet controllers, and 64 RS-232C user ports.

- M/1000 RISComputer with 15MHz CPU, floating point unit, 128KB of I&D caches, 16MB main memory with ECC, console port, dual SMD disk controller, one 337MB formatted 8" disk, 60MB 1/4" QIC-24 cartridge tape with controller, Ethernet controller, and 16 RS-232C user ports.

- M/1000 RISComputer with 15MHz CPU, floating point unit, 128KB of I&D caches, 68MB main memory with ECC, console port, two dual SMD disk controllers, three 8" disks, 60MB 1/4" QIC-24 cartridge tape with controller, Ethernet controller, and 32 RS-232C user ports.

- M/800 RISComputer with 12.5MHz CPU, floating point unit, 128KB of I&D caches, 16MB main memory with ECC, console port, dual SMD disk controller, one 337MB formatted 8" disk, 60MB 1/4" QIC-24 cartridge tape with controller, two Ethernet controllers, and 16 RS-232C user ports.

- M/800 RISComputer with 12.5MHz CPU, floating point unit, 128KB of I&D caches, 68MB main memory with ECC, console port, two dual SMD disk controllers, three 8" disks, 60MB 1/4" QIC-24 cartridge tape with controller, Ethernet controller, and 32 RS-232C user ports.

- M/500 RISComputer with 8MHz CPU, floating point unit, 16Kbytes of Instruction caches and 8 Kbytes of Data caches, 8MB main memory with ECC, console port, dual SMD disk controller, one 337MB formatted 8" disk, 60MB 1/4" QIC-24 cartridge tape with controller, two Ethernet controllers, and 16 RS-232C user ports.

- M/500 RISComputer with 8MHz CPU, floating point unit, 16KB of Instruction caches and 8 KB of Data caches, 20MB main memory with ECC, console port, two dual SMD disk controllers, three 8" disks, 60MB 1/4" QIC-24 cartridge tape with controller, Ethernet controller, and 32 RS-232C user ports.

- RC3260 RISComputer with 25MHz CPU, floating point unit, 128KB of I&D caches, 128MB main memory, console port, one SCSI device controller, four 663MB SCSI disks, one 150MB cartridge tape drive, one Ethernet controller, 16 serial I/O ports.

- RC3260 RISComputer with 25MHz CPU, floating point unit, 128KB of I&D caches, 128MB main memory, console port, two SCSI device controllers, eight 663MB SCSI disks, one 120MB cartridge tape drive, four Ethernet controllers, 16 serial I/O ports.

- RC3260 RISComputer with 25MHz CPU, floating point unit, 128KB of I&D caches, 128MB main memory, console port, one SCSI device controller, four 1GB SCSI disks, one 150MB cartridge tape drive, one Ethernet controller, 16 serial I/O ports.

- RC3240 RISComputer with 25 MHz R3000/R3010, 128 KB Instruction and Data Cache 8 MB main memory, DMA SCSI controller, 5.25" 328 MB disk, 120 Mbyte 1/4" cartridge tape, DMA Ethernet controller, 1/2" Table-top tape and 8 RS-232 usr ports.

- RC3240 RISComputer with 25 MHz R3000/R3010, 128 KB Instruction and Data Cache 8 MB main memory, DMA SCSI controller, 1 5.25" 328 MB disk and 1 120 MByte 1/4" cartridge tape in the base system, 4 5.25" disks and 1 2 GByte 8 mm Exabyte cartridge tape in the expansion : disk #1 328 MB, disk #2 156 MB, disk #3 330 MB , disk #4 156 MB, DMA Ethernet controller and 64 RS-232 usr ports.

- RC3240 RISComputer with 25 MHz R3000/R3010, 128 KB Instruction and Data Cache 48 MB main memory, DMA SCSI controller, 1 5.25" 328 MB disk and 1 150 MByte 1/4" cartridge tape in the base system, 4 5.25" disks and 1 2 GByte 8 mm Exabyte cartridge tape in the expansion : disk #1 328 MB, disk #2 controller, 1/2" Table-top tape, 4 Async Ports (2 Modem) and 32 RS-232 usr ports.

- M/120 RISComputer with 16.7 MHz R2000/R2010, floating point unit, 128 KB Instruction and Data Cache, 8 MB main memory, DMA SCSI controller, 5.25" 328 MB disk, 120 MByte 1/4" cartridge tape, DMA Ethernet controller.

- M/120 RISComputer with 16.7 MHz R2000/R2010, floating point unit, 128 KB Instruction and Data Cache, 16 MB main memory, DMA SCSI controller, 5.25" 328 MB disk, 120 MByte 1/4" cartridge tape, DMA Ethernet controller.

- Magnum 3000 monochrome workstation with 128 MB main memory, 200 MB hard disk, floppy drive, 663 MB hard disk, 1 GB hard disk, 1 Exabyte tape drive.

- Magnum 3000 dual headed workstation, 24 MB main memory, 2-200 MB hard disks, QIC 150 tape drive, 2-663 hard disks, and Exabyte tape drive.

- Magnum 3000 color workstation, 16 MB main memory, 2-200 MB hard disks, floppy drive, 663 MB hard disk.

- Magnum 3000 dual headed workstation with 16 MB main memory, 2-200 MB hard disks, 663 hard disk.

- RC3230 server with 32 MB main memory, 5-663 MB hard disk, Exabyte tape drive.

- RC3230 server with 32 MB main memory, 2-200 MB hard disk, 2-1 GB hard disk, Exabyte tape drive.

- RC3230 server with 24 MB main memory, 2-200 MB hard disk, 1-663 MB hard disk, Exabyte tape drive.

RISC/os 4.50 passes SVVS release 2.0 except for problems that are beyond the control of MIPS. These include problems with the tests themselves and deviations from other industry standards. There are a few deviations from SVID that are not tested by SVVS. For review purposes **Appendix A** is included, which specifies the state of all commands mentioned in the SVID.

1.6 End of Life for Old Products

MIPS will begin to phase out software QA testing of the older M/500, M/800. and M/1000 products in 1990. Supporting such a large base of hardware platforms complicates the testing matrix, and therefore makes it difficult to uphold high quality standards. RISC/os 4.50 provides fully tested support for the M/500, M/800 and M/1000. but for the next major release, these products will be dropped from the Software QA testing matrix. MIPS Customer Service, of course. will continue to support customers with these products.

2. General MIPS Release Information

This section describes information about RISC/os releases that does not change with each release. It should be read once and used as reference material.

2.1 Documentation and On-Line Manuals

RISC/os 4.50 includes updated on-line manual pages (optionally installable) which are always current and may be more up to date than the printed versions, so in case of discrepancy, use the on-line version. Printed manuals include:

RISC/os User's Reference Manual
RISC/os Programmer's Reference Manual
RISC/os System Administration Reference Manual
RISC/os User's Guide
RISC/os Programmer's Guide
RISC/os System Administrator's Guide
RISC/os Streams Primer and Programmer's Guide
Guide to Using BSD on RISC/os

2.1.1 man Command

The RISC/os **man** command is built twice, once to live in */bsd43/bin* as (4.3 BSD) **man**, **apropos**, and **whatis**, and once to live in */usr/bin* as (System V) **man**. The user's path determines which **man** command is executed.

The System V **man** defaults to **-a** (all matching manual pages) and prompts between files. The 4.3 BSD **man** defaults to printing only the first matching page, with **-a** being an option. An option has been added to both versions of **man**, **-S** *systype*, which the user can specify to tell **man** the "universe" (System V, 4.3 BSD, or POSIX) to search first when looking for a particular man page. See **man**(1) for more information.

2.1.2 On-Line Release Notes

MIPS supplies all release notes on-line with the software, in addition to the printed copy. A formatted version of these release notes can be found in */usr/pkg/rn/RISCos* for on-line viewing. The directory */usr/pkg/rn* will contain other release notes files as well, depending on what software was ordered. Any last minute changes to release notes may not be included on the distribution tape, so always note that the printed version is more accurate than the on-line file.

2.2 RISC/os Distribution Media

RISC/os is distributed on four QIC-24 format tapes. MIPS systems software is organized as a set of packages which comprise several subpackages. The RISC/os operating system software package includes the following subpackages:

Tape 1:

File #	Name	Contents
0	id:	tape identification
1	instd:	packaging information and tools
2	tapevol.std:	a bootable tape header and the stand-alone utilities for non-2030 systems, including the stand-alone shell (sash.std) and the disk

		formatter (format.std).
3	**tapevol.2030**:	a bootable tape header and the stand-alone utilities for Rx2030 systems, including the stand-alone shell (**sash.2030**) and the disk formatter (**format.2030**).
3	**tapevol.2030**:	same as above for 2030 systems.
4	**miniroot**:	a minimal file system used to complete installation of the new os.
5	**unix.r2300_std.boot**	M/500, 800, 1000 tape bootable kernel
6	**unix.r2400_std.boot**	M/120, RC3240 tape bootable kernel
7	**unix.r3200_std.boot**	M/2000 SMD tape bootable kernel
8	**unix.i2000_std.boot**	Rx2030 tape bootable kernel
9	**unix.r3200_ijc.boot**	M/2000 SCSI, RC3260 tape bootable kernel
10	**unix.r6000_std.boot**	RC6280 tape bootable kernel
11	**unix.r3030_std.boot**	Rx3230 tape bootable kernel
12-19	**9 space holders**	
20	**archive1**:	**tar** archive of root filesystem
21	**archive2**:	**tar** archive of M/500, 800, 1000 device and kernel files
22	**archive3**:	**tar** archive of M/2000 device and kernel files
23	**archive4**:	**tar** archive of RC3260 device and kernel files
24	**archive5**:	**tar** archive of M/120 and RC3240 device and kernel files
25	**archive6**:	**tar** archive of Rx2030 device and kernel files
26	**archive7**:	**tar** archive of RC6280 kernel files
27	**archive8**:	**tar** archive of RC6280 device files
28	**space holder**	
29	**archive9**:	**tar** archive of Rx3230 device and kernel files

Tape 2:

File #	Name	Contents
0	**id**:	tape identification number.
1	**archive1**:	standard *usr* filesystem **tar** archive
2	**archive2**:	base compiler system **tar** archive
3	**archive3**:	4.3 BSD compilation environment (libraries and compiler include files) **tar** archive
4	**archive4**:	on-line manuals **tar** archive

Tape 3:

File #	Name	Contents
0	**id**:	tape identification number.
1	**archive1**:	RISC/os 4.00 compatibility **tar** archive
2	**archive2**:	4.3 BSD commands **tar** archive
3	**archive3**:	**tar** archive of files used for binary kernel reconfiguration
4	**archive4**:	**emacs tar** archive

Tape 4:

File #	Name	Contents
0	**id**:	tape identification number.
1	**archive1**:	POSIX libraries **tar** archive.
2	**archive2**:	**uucp tar** archive
3	**archive3**:	**sccs tar** archive

4	archive4:	tar archive of netnews reading programs
5	archive5:	games tar archive
6	archive6:	mh tar archive

A Note Regarding QIC Compatibility

The following matrix demonstrates QIC tape device compatibility with various cartridge/medium types made by 3M. The number in parentheses indicates the amount of data that can be recorded given format and tape type.

| | Cartridge Type | | |
	DC300XL	DC600A	DC600XTD/DC6150
Drive Type			
QIC-24	read/write (45MB)	read/write (60MB)	read/write (60MB)
QIC-120	read (45MB)	read/write (120MB)	read/write (120MB)
QIC-150	read (45MB)	read/write (120MB)	read/write (150MB)

A QIC-24 drive can read all DC300XL cartridges written in QIC-24 format. A QIC-24 drive **cannot** read tapes written in QIC-120 or QIC-150 format.

A QIC-120 drive can record only on DC600A or 600XTD cartridges. It can read any QIC-24 format on any medium, but can only record in QIC-120 format. **Tapes made on a QIC-120 drive cannot be read on a QIC-24 drive.**

QIC-150 drives (supported **only** on the Rx3230) are identical to the QIC-120 drive except in the case where DC600XTD media is used. If XTD media is recorded in a QIC-150 drive, the drive will record 150MB of data on the tape and it can only be recovered by a QIC-150 drive. The QIC-150 drive will read both QIC-120 and QIC-24 formats regardless of media type, but writes in QIC-120 format on DC600A cartridges.

2.3 Domestic and International Distributions

MIPS has obtained permission from the United States Department of Commerce to distribute the encryption software found in RISC/os 4.50 outside the United States. Previously, distribution of this software outside the United States was prohibited, necessitating separate domestic and international binary releases. This permission means that MIPS now has one binary release for both United States domestic and international use.

2.4 MIPS Software Installations

All new systems shipped from MIPS are supplied with the operating system (RISC/os) already installed. This means that new systems can be used without performing any of the installation procedures described in these release notes. If a system is running an earlier release of RISC/os and requires a software update, or for some reason requires that the software be installed from scratch, follow the appropriate procedure described in the **Installation Instructions** section.

An **update installation** is used when the previous release of RISC/os or UMIPS-BSD 2.1 operating system software is already resident on disk, and you want to install a new version of

8

RISC/os without disturbing user applications, data files, or system administrative and configuration files. One of the goals of the installation tools is to minimize the downtime. In the nominal case, a new release of RISC/os should take about two hours to install as an update, and a system should not require any extensive reconfiguration before being placed back into normal service.

It is important to realize that any binaries created on a UMIPS-BSD system will not run on a RISC/os system without recompilation. Because of this, some users with UMIPS-BSD machines may prefer to perform a scratch install of RISC/os, and then reload any non-executable files from tape back up. If this is not an option, it is possible to do a UMIPS-BSD to RISC/os update installation, but it requires more work than a regular update. When an update installation is performed on a UMIPS-BSD system, the /etc/ttys file is converted to System V-style /etc/inittab and /etc/ttytype files as part of the installation procedure, but after the installation is complete, the system administrator will have to remove any binaries compiled under UMIPS-BSD to avoid confusion. Please see the section **Post-Installation Work for UMIPS-BSD Systems** for more information.

A **scratch installation** is used when RISC/os operating system software is loaded for the first time, or in the event of an operating system data loss, and presumes only that the disks to be installed upon are correctly formatted; new filesystems are made, and default versions of all administration and configuration files are installed along with the rest of the RISC/os software. Because new filesystems are made, all files on the root and usr partitions will be lost during the installation.

A **remote tape installation** is performed at a site which has a MIPS system that has no tape drive but does have another machine on the network with a QIC tape drive. The software is installed from the remote tape drive, across the network to the local system. Either an upgrade or a scratch installation can be done this way but the two systems **must** be on the same network.

2.4.1 A Few Words About Backups

Update installs do not disturb existing user files. Although a goal of the MIPS installation tools is that an operating system update can be performed without **requiring** that user and administrative files be backed up prior to the installation, making backups is still a prudent step. Nominally, after the update has been installed, the system can be rebooted and placed back into normal multiuser operation without any additional actions to restore user or administrative configuration files. However, to account for the possibility of hardware failures during the installation procedure which could result in losing user files, MIPS recommends that good backups be on hand prior to performing an update install.

Exact, step-by-step instructions for performing the backup operation is beyond the scope of these release notes. We recommend the use of the **dump** utility to perform the backup. We recommend that this procedure be performed by an experienced UNIX system administrator (see the *System Administrator's Reference Manual*, if necessary). Care should be taken to ensure that files from the new release are not overwritten when the backups are restored.

2.4.2 A Few Words About Disk Formatting

All disks are formatted before being loaded with software and shipped to customers. The only time a customer should format a disk is when the system has reported disk errors. If for some reason the system software is lost without any disk corruption, all that is necessary is a reload of the operating system. If it is necessary to reformat a system disk, see the *Technical Reference Manual* for your system for instructions.

2.4.3 Comply and BOM Files

A program called comply is used in conjunction with 'bom' (Bill of Materials) files to drive the installation. The bom files are a list of all files in the release, specifying file modes, ownership, and links. comply uses the bom files to ensure all files are installed correctly and to create links (both hard and symbolic) that are included in the package.

The boms describing the package installed are present on the installed system as /usr/pkg/lib/<package><version>/boms/*, and can be used at any time to verify whether all of the files originally installed are still present and correct with respect to mode, ownership, etc.

As part of the installation, any discrepancies comply finds between what the boms specify should be installed, and what was actually installed, are reported on the console. Only in rare instances will there be comply messages. This output is also saved in a file: /usr/pkg/lib/<package><version>/lib/comply.out, where <package> is the package name ("umips", for a RISC/os release), and <version> is the version number of the package. For example, after installing RISC/os 4.50, the comply output for the installation can be found in /usr/pkg/lib/umips4.50/lib/comply.out.

See the comply(1M) manual page for more information on running comply.

2.4.4 Disk Space Accounting

The MIPS installation tools provide an optional check of disk space to ensure that the files to be installed will fit on the disk(s) present on the system before any files are actually installed. This is especially important for update installations on systems with many user files. It can also be important for scratch installations on small disks, where some combinations of optional subpackages may not fit on the disk concurrently.

The space check properly accounts for the local disk, partitioning, and filesystem configuration based on the system's existing /etc/fstab file.

If the space check determines that there is not enough space to proceed with the requested configuration, it will report the shortfall, and the installation procedure will be terminated. At this point, you should make the required space, select a different set of optional subpackages, or modify the disk, partitioning, or filesystem configuration to provide enough space, and then retry the installation. Note that if the optional space check was declined and there is not enough disk space, the installation will corrupt the root and usr filesystems.

2.4.5 Loading RISC/os

Installing an operating system is more complex than installing an application software package. The operating system must be loaded from a "miniroot filesystem", or "miniroot", a minimal operating system. After the system is either shutdown to the prom monitor prompt, or powered on, a standalone shell, sash, is booted from the distribution tape, the miniroot is copied onto a swap partition of the disk, and the kernel is then booted from tape, allowing the machine to come up on the miniroot filesystem. This scheme allows the root and usr filesystems to be loaded with the new operating system, accomplished by the inst program resident on the miniroot.

In order not to disturb user or administrative files during an update installation, the preserve program is run before the files are extracted from the archives containing the new release files. preserve ensures that any user customized files are not overwritten by files from the new release by renaming them, from "name" to "name:<version>:", where "<version>" is the version number of the software being installed. After the files in the new release are loaded onto the system, preserve is run once again to restore the original files. This is done by renaming the file from the new release from "name" to "name:<version>+", and naming

10

the original file from "name:<version>:" back to "name". As an example:

	Before Installation	During	After
Your File	passwd	passwd:4.50:	passwd
Release File	N/A	passwd	passwd:4.50+

It is important to understand the **preserve** naming conventions, in case the installation procedure is interrupted, so that you can tell which files were from your system before the installation and which have been installed from the new release. If for some reason the install procedure was interrupted prior to running **preserve** to restore files, you could look for your original password file as /etc/passwd:4.50:.

It is important to note that the <version> in "name:version:" is the version number of the release being installed; thus "name:<version>:" should be read as "the file 'name' present on the system when <version> was installed", rather than "the file 'name' from release <version>". Likewise, files from the new release preserved as "name:<version>+" should be read as "the file 'name' distributed with the release <version>".

The set of files subject to the **preserve** program is the union of two sets, called the "default preserve list", and the "findmods list". The default preserve list is always used, and contains entries for the files which are usually customized locally. The findmods list, which is used at the discretion of the system administrator, is generated at install time by the program **findmods**. **findmods** identifies files present in the new release which appear to have been modified locally since the previous release was installed.

In theory, **findmods** alone should be sufficient to identify all of the files on a system which need to be preserved. However, **findmods** depends on two assumptions which may not always hold: 1) that the installation log, /etc/installlog, is present on the system and valid, and 2) that the modification timestamps on files released from MIPS which have not been modified locally have not been disturbed since the previous release was installed. It was felt that without careful system administration, these assumptions could not be made.

Therefore, the default preserve list is provided and, by default, in an update install it is used as the sole input to **preserve**; **findmods** is not executed. However, if a) your site **does** have many locally modified files, b) the /etc/installlog file has not been removed, and c) the modify timestamps on unmodified files from the previous release have not been changed, then you can enable the use of **findmods** prior to starting the install procedure by entering the following command just before executing the **inst** command:

```
# Findmods=y
```

In some instances, it is important that a file from the new release replace a file on the system; for example, if the format of the file /etc/inittab were to change from one release to the next. In many such cases, the install scripts will provide an automatic conversion of the existing local file to the new format, thus maintaining the local configuration as well as compatibility with the new syntax of the file. However, there may be cases where an automatic conversion cannot be provided, and it is important to ensure that the file from the new release gets installed along with the new software. To provide for this case, the preserve lists can include a flag (-u) specifying that instead of restoring a preserved local file from "name:<version>:" to "name", the file should be left on the local system as "name:<version>:", and the file from the new release be installed as "name".

2.4.6 Disk Drive Types

During a scratch installation, you will be prompted to enter the disk type for controller 0 drive 0. The following lists the drive types and what disk type number to enter:

Disk Drive	Enter:
138MB SCSI CDC	94351-138
140MB SCSI FUJI	2246sa
156MB CDC	94161
170MB SMD FUJI	2322
170MB ESDI FUJI	2246
160MB SCSI CDC	94161
172MB SCSI CDC	94351-172
172MB CDC	94351
200MB Maxtor	LXT-200S
325MB SCSI FUJI	2249sa
328MB SCSI Seagate/CDC	94171
330MB CDC	94181
330MB SMD FUJI (63 sectors)	2333-63
330MB SMD FUJI (64 sectors)	2333-64
663MB Seagate/CDC	94191
689MB SMD (63 sectors)	2344-63
689MB SMD (64 sectors)	2344-64
824MB FUJI (63 sectors)	2372-63
824MB FUJI (64 sectors)	2372-64
1GB SCSI Seagate/CDC	94601

2.5 Booting the System

If the system is off, turn the power on (press RESET or use the keyswitch on the front panel if the system is already powered on) to get the monitor prompt (>>).

To boot RISC/os, enter auto at the monitor prompt. The system may display some messages, fsck the root partition, and reboot. The default run level is now "2" (multi-user). To override the default and boot in single user mode, instead of entering auto, use:

>> **boot dksd()unix initarg=s**   *(for Rx2030, Rx3230)*
>> **boot dkis()unix initarg=s**   *(for M/120 or RC3240)*
>> **boot dkip()unix initarg=s**   *(for M/500, 800, 1000, M/2000 SMD, or RC6280)*
>> **boot dkij()unix initarg=s**   *(for RC3260 or M/2000 SCSI)*

When the system is booted to multi-user mode (run level 2), the following message will be displayed on the console:

```
***** Normally all file systems are fscked.
```

```
***** To fsck only dirty ones, type 'yes' within 5 seconds:
```

Note that in RISC/os 4.50, there is a long pause while the filesystems are being fsck'ed.

Other useful options to boot are:

root	specify root disk "root=*ipc0d0s1*" for example
askme	Interactive equivalent of "root". Note: with no disk specified, the administrator is prompted, hence the name. *ipc0d0s1* is used only as an example.
showconfig	More verbose on boot and for non-fatal errors that are encountered.
initfile	Initial file to exec in place of */etc/init*.

initarg   Arguments to **initfile** program. Note: multiple arguments can be specified; all will get passed down. "initarg=a initarg=b".

2.6 Binary Kernel Reconfiguration

RISC/os supports binary reconfiguration of the kernel. This allows users without complete source to change tunable kernel parameters and add device drivers. This section describes the procedure for building kernels with modified parameters. Subsequent sections describe what parameters may be modified, how to modify them, and how to rebuild the kernel with modified parameters. The man pages for *lboot*(1M), *system*(4), and *master*(4) also provide information on binary kernel reconfiguration.

2.6.1 Modifying Tunable Parameters

Tunable parameters are values in the kernel which control kernel configuration or its run time behavior. The optimal values for many of these parameters depend on system specific hardware configuration and system specific workloads.

The tunable parameters are set in files in the directory */usr/src/uts/mips/master.d*. The files:

kernel.i2000_std (Rx2030)	*sysgen.i2000_std* (Rx2030)
kernel.r3030_std (Rx3230)	*sysgen.r3030_std* (Rx3230)
kernel.r2300_std (M/500, M/800, and M/1000)	*sysgen.r2300_std* (M/500, M/800, and M/1000)
kernel.r2400_std (M/120, RC3240)	*sysgen.r2400_std* (M/120, RC3240)
kernel.r3200_std (M/2000)	*sysgen.r3200_std* (M/2000)
kernel.r3200_ijc (M/2000 SCSI, RC3260)	*sysgen.r3200_ijc* (M/2000 SCSI, RC3260)
kernel.r6000_std (RC6280)	*sysgen.r6000_std* (RC6280)

describe the standard configuration as it is shipped. To make local changes, copy the appropriate *kernel.XXXX0_std* to *kernel.XXXX0_local* and make the changes there. Also copy the appropriate *sysgen.XXXX0_std* to *sysgen.XXXX0_local*. *sysgen.XXXX0_local* won't be modified in this procedure, but it must exist.

The file *kernel.XXXX0_local* is an **lboot** *master.d* file which contains a short **lboot** header followed by C code. During the reconfiguration process, the C code is compiled and then linked with the binaries for other kernel modules to produce a new kernel. The tunable parameters are simply #defines in this C code.

To change a parameter, edit the file *kernel.XXXX0_local*. Find the appropriate #define and change the value. For instance, the line in *kernel.r3200_std* which sets the number of process table entries looks like this:

  #define NPROC 500

Note that defaults for most such values are defined in a file, *kernel_common*, included by the various *kernel.** files. If you define a symbol in *kernel.local*, it replaces the default defined in *kernel_common*.

Reconfiguring the kernel for proper buffer cache size is not necessary, because RISC/os determines the amount of physical memory at boot time, and adjusts the number of buffers allocated accordingly. In addition, the kernel adjusts the size of the buffer cache as it runs. Depending on the memory requirements from user processes, and from other parts of the kernel, the buffer cache may use anywhere from 10% to 50% of the system's physical memory.

13

Some of the parameters you may want to modify are:

For System V Facilities

The kernel may be configured at startup time to disallow certain System V.3 facilities not permitted by 4.3 BSD or the POSIX FIPS, such as the non-privileged use of *chown()*. For example, "_posix_chown_restricted" determines whether a *chown()* which attempts to change the ownership of a file requires root privilege. (See *kopt(8)* for details.) The boot-time defaults for all such new options may be set via binary reconfiguration.

NFILE

NFILE describes the number of open file entries allowed in the system. If the kernel prints the message "NOTICE: file table overflow", then the value of NFILE should be increased.

NBLK4, ... NBLK4096

NBLK4 describes the number of stream buffers of size 4. There are also constants for buffers of size 16, 64, 128, 256, 512, 1024, 2048 and 4096, with the obvious names. On systems with many incoming tty lines and heavy tty activity, these values may need to be increased.

NQUEUE

The system must have one stream per active tty or remote login. NQUEUE should be roughly three times the number of active streams expected, where each real terminal line uses one stream and each pseudo-terminal (pty) uses two streams.

2.6.2 How to Build a Reconfigured Kernel

To build a kernel with the changes described in the modified *kernel.local* file:

- login as root

- make sure you are **not** using the Berkeley C compiler (i.e., make sure that */bin* is in your path before */bsd43/bin*)

- make sure the SHELL variable is set to */bin/sh* and proceed as follows:

```
# cd /usr/src/uts/mips
# set -a
# BUILD_TYPE=reconfig
# SHELL=/bin/sh
# make -e unix.local
```

This build will take 2 to 4 minutes, and when it is finished the new kernel will be in the current directory with the name *unix.local*.

The system should now be rebooted using the new kernel:

```
# mv /usr/src/uts/mips/unix.local /
# ln /unix.local /unix
# telinit 0
>> boot dksd()unix      (for Rx2030, Rx3230)
>> boot dkis()unix      (for M/120 or RC3240)
>> boot dkip()unix      (for M/500, 800, 1000, 2000 SMD, RC6280)
>> boot dkij()unix      (for SCSI M/2000 or RC3260)
```

Because the system is shipped with the *unix.XXXX0_std* files, even after the mv and ln above, a copy of the standard *unix* is still available. If for some reason the local unix does

14

not work, the standard *unix* can be rebooted by typing:

```
>> boot dksd()unix.i2000_std  (for Rx2030)
>> boot dksd()unix.r3030_std  (for Rx3230)
>> boot dkis()unix.r2400_std  (for M/120 or RC3240)
>> boot dkip()unix.r2300_std  (for M/500, 800, 1000)
>> boot dkip()unix.r3200_std  (for M/2000 SMD)
>> boot dkij()unix.r3200_ijc  (for RC3260 or M/2000 SCSI)
>> boot dkip()unix.r6000_std  (for RC6280)
```

2.7 Troubleshooting Tips

This section describes important information that may help in configuring your system or solving common problems.

2.7.1 NFS Authentication Errors

RISC/os 4.50 allows up to 16 groups to be sent over NFS as an authentication object, where previous releases allowed a maximum of only 8. That is, if a user id appears in more than 8 groups in the */etc/group* file, there is a problem when using NFS for a RISC/os 4.50 client talking to a server running an earlier RISC/os release. The symptom is the error message:

```
NFS getattr failed for server <x>: RPC: Authentication error
```

To work around this problem, modify the client's */etc/group* file so that no user is a member of more than 8 groups if a RISC/os 4.50 client is talking to a server running a release earlier than RISC/os 4.50.

2.7.2 Automatic Reboot

On currently released proms for the M/2000 (both SCSI and SMD) and RC3260 products, the system does not always automatically reboot. To boot the system, type **init** at the prom monitor prompt before giving the boot command.

2.7.3 Rx2030 Performance Tuning

Since the Rx2030 has a limited amount of memory (16MB) and a relatively slow disk, and because RISC/os 4.50 has larger binaries, a performance penalty may be obvious to some Rx2030 users. Here are some tricks that we use to tune Rx2030's to help them run more efficiently:

1. Kill off all unnecessary daemons, e.g., **rwhod** and **nfsd** (if you don't export any file systems from your Rx2030, you don't need **nfsd**)

2. **renice** X and **mwm** to -20 (better interactive performance)

3. Be sure that rotdelay=0 on root and usr

2.7.4 mount

When a mount request comes in, the MIPS **mount** daemon allows only requests which come in on a reserved port. This is known as "port monitoring" and if other vendor's systems adhere to this protocol, there are no problems. However, some companies allow and even use the non-reserved ports. This can cause problems on a MIPS system. If this is a problem

at your site, you can start up the **mount** daemon with the -n option, which says not to check that the clients are root users. Note that this is a security hazard. See the *mount*(1M) man page.

2.7.5 Slow Logins

If a system has many NFS mounted filesystems and the use of **quota** is enabled with the */etc/login.quotawarn.ok* file, a user can speed up the login time by creating an empty file *~/.hushlogin*.

2.7.6 Serial I/O Cabling

There are two general classes of serial devices, DTE (data terminal equipment, usually terminals and computers) and DCE (data communications equipment, usually modems). Each talks to the opposite class of device.

In order to allow one kind of device to talk directly to the same kind of device (i.e. DTE to DTE, as in terminal to computer communication) each side must think the other device is the opposite kind of device. This is what a null modem accomplishes. Therefore, a null modem is required for any DTE to DTE or DCE to DCE communication but not between DTE and DCE.

All MIPS systems have built-in serial ports. In addition, many have optional extra serial ports. Most machines have two ports (tty0 and tty1), except for the M/120 and RC3240, which have four (tty0, tty1, tty2, and tty3). All of these ports are DTE, except for port tty0 on the M/500, M/800, M/1000, M/2000, RC3260, and RC6280.

The tty1 port on the RC2030, and all tty0 and tty2 ports, except for the tty0 port on the Rx3230, have no modem control signals (DTR, DCD, DSR, RTS, and CTS) at all. Therefore, these are not suitable for use with dial-in or dial-out modems.

The tty1 port on the M/500, M/800, M/1000, M/2000, RC3260, and RC6280 and the tty1 and tty3 ports on the M/120 and RC3240 have partial modem control. They can recognize DCD and they can assert DTR. However, the pin-out of these ports is non-standard. In particular, all ports expect DCD to be supplied on pin 5, instead of the standard pin 8. Also, the M/2000, RC3260, and RC6280 assert DTR on pins 6 and 8, instead of on the standard pin 20.

In order to convert tty1 on an M/2000, RC3260, or RC6280 to a standard DTE, you must construct a cable that connects pin 8 on the modem side to pin 5 on the computer side, pin 20 on the modem side to pin 8 on the computer, and disconnect pin 6 on the computer side. Pins 1, 2, 3, and 7 go straight through.

Here are the signals and pin # on various serial boards:

	DCD	DTR	RTS	CTS	DSR
M/120, RC3240 & Rx3230 "Com-8" (ttyd*) serial I/O card	8	20	4	5	6
M/120, RC3240 & Rx3230 "Com-16" (ttyd*) serial I/O card	10	9	3	8	2
Rx3230 tty0 (DB9)	1	4	7	8	
Rx3230 tty1	8	20	4	5	6

Note: The "Com-16" serial I/O card has RJ-45 jacks, not DB25.

To hook up a terminal (DTE) device:

16

On RISComputer	To Port	Use
M/120, RC3240	ttyd*	null modem
M/120, RC3240	tty0, tty2	null modem
M/120, RC3240	tty1, tty3	null modem, w/ magic cable A
M/500, M/800, M/1000	tty0	straight through
M/500, M/800, M/1000	tty1	null modem, w/ magic cable A
M/500, M/800, M/1000	tty[hi]*, w/ non-MIPS adapter	straight through
M/500, M/800, M/1000	tty[hi]*, w/ MIPS adapter	null modem
M/2000, RC3260, RC62?0,	tty0	straight through
M/2000, RC3260, RC62?0	tty1	null modem, w/ magic cable B
M/2000, RC3260, RC62?0	tty[hijklmno]* w/ non-MIPS adapter	straight through
M/2000, RC3260, RC62?0	tty[hijklmno]* w/ MIPS adapter	null modem
Rx2030	tty0, tty1	null modem
Rx3230	tty0, tty1	null modem

To hook up a modem (DCE) device:

On RISComputer	To Port	Use
M/120, RC3240	ttyd*	straight through
M/120, RC3240	tty0, tty2	straight through
M/120, RC3240	tty1, tty3	straight through, w/ magic cable A
M/500, M/800, M/1000	tty0	null modem
M/500, M/800, M/1000	tty1	straight through, w/ magic cable A
M/500, M/800, M/1000	tty[hi]* w/ non-MIPS adapter	null modem
M/500, M/800, M/1000	tty[hi]* w/ MIPS adapter	straight through
M/2000, RC3260, RC62?0	tty0	null modem
M/2000, RC3260, RC62?0	tty1	null modem
M/2000, RC3260, RC62?0	tty[hijklmno]* w/ non-MIPS adapter	null modem
M/2000, RC3260, RC62?0	tty[hijklmno]* w/ MIPS adapter	straight through
Rx2030	tty0, tty1	straight through
Rx3230	tty0, tty1	straight through

Whenever a magic cable is used, make sure that it is connected directly to the host port side. If a null modem is also used, the null modem must be placed between the magic cable and device (terminal, modem, etc).

To make a magic cable for M/120, RC3240, M/500, M/800 and M/1000 (cable A):

1. Connect Host 5 to Modem 8

2. Connect all other pins straight through.

To make a magic cable for an M/2000, RC3260, and RC6280 (cable B):

1. Connect Host 1, 2, 3, 4, 7 to Modem 1, 2, 3, 4, 7

2. Connect Host 5 to Modem 8

3. Connect Host 8 to Modem 20

4. Disconnect Host 6

A null modem cable has the following pinout:

Host Pin	Device Pin
2	3
3	2
4	5
5	4
6, 8	20
7	7
20	6, 8

For cable length requirements refer to the RS-232C standard.

2.7.7 Determining if your VME machine is DTE or DCE

Most M/500, M/800, M/1000 tty[hi]* ports are DCE, whereas most other tty[hijklmno]* ports are DTE; however, this is not always true. On the M/500, M/800, M/1000 and M/2000, RC3260 ribbon cables run from the front of the "VME-SIO" (tty[hijklmno]*) serial I/O controller to the rear of the machine where each connects to an adapter card with eight DB25 connectors.

There are three flavors of these cards: Some were purchased from a vendor, some were purchased from the same vendor but had rework done, and some were made by MIPS. The cards with rework can be easily identified by a black cable running across the board connecting pin 1 of all the ports. The MIPS board will have MIPS imprinted in the board. M/500, M/800, M/1000s usually have the cards with the black rework cable, while M/2000, RC3260s usually have the card manufactured by MIPS. You should check your board to be certain which one you have. The MIPS made board will turn the computer into a DTE, while either of the other two boards will make it a DCE.

2.7.8 Modem Support and Pinouts

RISC/os 4.50 supports the following modem signals:

Signal	Software Support
DCD	Used in conjunction with modem ports (i.e., those ports with -clocal set) for standard UNIX behavior. RISC/os 4.50 now supports DCD/DTR hardware flow control, which is enabled via stty mdmbuf.
RTS/CTS	RTS is raised when data is ready to be sent. RTS/CTS hardware flow control is now supported in RISC/os 4.50. This is enabled by stty rtscts.
DTR	Dropped when a port is closed or when host can no longer accept input and stty mdmbuf is set.

RISC/os 4.50 is capable of both software flow control (XON/XOFF) and hardware flow control (DCD/DTR and RTS/CTS); however, since not all MIPS serial ports support all modem signals, not all ports support all types of flow control. Most ports support both DCD and DTR and therefore support DCD/DTR flow control. The ports that do not support these signals cannot support this protocol. These ports are: all even number console ports (e.g. tty0 and tty2) and both tty ports on the Rx2030. All other ports support DCD/DTR flow control. DCD/DTR flow control is enabled by stty mdmbuf. The clocal option should be set (otherwise the port will be closed when DCD drops) and no other flow control should be set (i.e., ixon, ixoff, ixany).

18

Some ports support both RTS and CTS and therefore support RTS/CTS flow control. The ports that do not support these signals cannot support this protocol. The ports that do support RTS/CTS protocol are all DigiBoard ports and tty0 and tty1 on the Rx3230. No other ports support this protocol. RTS/CTS flow control is enabled by the command **stty rtscts**. No other flow control should be set.

No modem signals are required to implement software flow control; therefore, all ports support this protocol (XON/XOFF).

Many of MIPS' serial ports implement only a subset of the standard modem control signals. Sometimes even this subset is supported on non-standard pins. Below is a chart describing exactly which port supports which signals and on which pin this signal is supported.

	DCD	DTR	RTS	CTS	DSR
M/120 and RC3240 "Com-8" (ttyd*) serial I/O card	8	20	4	5	6
M/120 and RC3240 "Com-16" (ttyd*) serial I/O card	10	9	3	8	2
M/500, M/800, M/1000, M/2000, RC3260, and RC6280 "VME-SIO" (tty[hijklmno]*)	8	20			
M/120, RC3240 tty0 and tty2	5	20	4		
M/120, RC3240 tty1 and tty3					
M/500, M/800, M/1000 tty0	5				
M/500, M/800, M/1000 tty1					
M/2000, RC3260, RC6280 tty0	5	8	4		
M/2000, RC3260 tty1	8	20	4		
RC6280 tty1	20				
Rx2030 tty0 and tty1	1	4	7	8	
Rx3230 tty0 (DB9)	8	20	4	5	6
Rx3230 tty1					

Notes: The "Com-16" serial I/O card has RJ-45 jacks, not DB25. tty0 on the Rx3230 is a DB9 connection. All odd numbered console ports (tty1, tty3) have a limited subset of modem signals and they are implemented on non-standard pins. The RC6280 supports the same signals but we have fixed the pinouts so these signals are supported on the standard pins.

The VME-SIO card only supports DCD and DTR.

2.7.9 Hardware Settings

M/120, RC3240 and Rx3230 "Com-8" Serial I/O Board

Jumpers
    There is a set of ten jumpers located near the left side of the board. These jumpers are used for board identification and they set interrupt request levels for each port. These must be configured as follows:

Board #	Jumper #	Connects These Pins
Board 0	1 thru 8	1-2
	9 thru 10	2-3
Board 1	1 thru 8, 10	1-2
	9	2-3
Board 2	1 thru 9	1-2
	10	2-3
Board 3	1 thru 10	1-2
All Boards	87	1-2

Switch Settings

The DIP switches on the DigiBoard are located at the bottom of the board. There are 9 banks each with 8 switches. All switches listed below must be set "on".

Switch #	Board 0	Board 1	Board 2	Board 3
DS1	1,3,4,10	1,3,4,10	1,3,4,10	1,3,4,10
DS2	1,3,5,6,7,8	1,4,5,6,7,8	1,5,6,7,8	2,3,4,5,6,7,8
DS3	1,3,5,6,8	1,4,5,6,8	1,5,6,8 2,3,4,5,6,8	
DS4	1,3,5,7,8	1,4,5,7,8	1,5,7,8 2,3,4,5,7,8	
DS5	1,3,5,8	1,4,5,8	1,5,8	2,3,4,5,8
DS6	1,3,6,7,8	1,4,6,7,8	1,6,7,8 2,3,4,6,7,8	
DS7	1,3,6,8	1,4,6,8	1,6,8	2,3,4,6,8
DS8	1,3,7,8	1,4,7,8	1,7,8	2,3,4,7,8
DS9	1,3,8	1,4,8	1,8	2,3,4,8

Using more than one "Com-8" in a system requires daisy chaining the status register of each card together. This is done by chaining P2 of each board.

P2 daisy chaining for multiple boards

---

1 Board	Connect Pins 2-3
2, 3, or 4 Boards	Connect Pin 2 on each board to pin 3 on the board above.
	Pin 2 on the top board gets jumpered to pin 3 on the bottom board.

M/120, RC3240 and Rx3230 "Com-16" Serial I/O Board

The 16 port DigiBoards do not have any address switches; the addresses are burned into PALs. The only way to tell what board you have is by the part number or the numbers on the PALs:

Board Number	PAL Part Numbers
COM-16 #1	40000336A 40000337A
COM-16 #2	40000338A 40000339A
COM-16 #3	40000384A 40000385A
COM-16 #4	40000386B 40000387B

Using CP-16 boards 3 or 4 requires a jumper on the motherboard of the M/120 or RC3240. This is done as follows:

1. With the power off, remove the CPU board.

2. Remove the jumper from JPF9 that joins pins 5 and 12 and the jumper on JPD8 that joins pins 6 and 11.

3. Use a jumper cable to connect pin 5 on JPF9 and pin 11 on JPD8.

4. Replace the cpu board.

For other jumpers, refer to the *Technical Reference Manual* for your system.

DigiBoard Jumpers

For the first and second board (PALs 4000336A, 4000337A, 4000338A, 4000339A), set jumper J6. This selects IRQ7. For the third and fourth boards (PALs 4000384A, 4000385A, 4000386B, 4000387B), set Jumper J5. This selects IRQ 6. Do not set both on one board.

For the first and third boards (PALs 4000336A, 4000337A, 4000384A, 4000385A), you must connect pins 2 and 3 of jumper J7. These are the two pins toward the bottom of the board. For the second and fourth boards (PALs 4000338A, 4000339A, 4000386B, 4000387B), you must connect pins 1 and 2 of jumper J7. These are the two pins toward the top of the board. (The lower number PAL must always be towards the bottom of the board.)

Daisy Chaining

Using more than one "Com-16" in a system requires daisy chaining the interrupt request line of different cards together. The 4 boards can be viewed as 2 sets of two cards each. Cards 1 and 2 get chained together as a pair that run on interrupt 7 and boards 3 and 4 get chained together as another pair on interrupt 6. If one card of a pair is being used alone that board must be chained to itself. Daisy chaining is done by connecting the pins of P3 as follows:

P3 Jumper settings for multiple 16 port boards boards

1 Board of either set	Connect Pins 1-2, do not connect the cards together.
2 Boards of either set	Connect Pin 1 on each board to pin 2 on the partner board and pin 2 to pin 1 on the partner board. Boards 1 and 2 are partners, and boards 3 and 4 are partners.

Mixing 8- and 16-Port DigiBoards

All 16 port boards are simulating two 8-port boards. For example, 16-port board 1 simulates 8-port boards 1 and 2. Various 8- and 16-port boards may be combined as long as they do not overlap with each other. The boards need not be consecutive. One 16-port board may be daisy chained to a maximum of two 8-port boards. 8-port board 2 cannot be used with 16-port board 1 because the ports would overlap.

A system can have non-consecutive boards: 8-port board 1 and 16-port board 4 will function properly. In this case the 8-port board is set as usual, daisy chained to itself. The 16-port board will have jumper J5 set and pins 1 and 2 on J7. This board will also be daisy chained to itself.

Pinout for COM-16

Each COM-16 board has 16 RJ-45 connectors. The following table lists the pinouts for these ports and also describes how to adapt these to a DB25 connector.

Pin #		Signal	Connect to this pin on a DB25 connector
1	RI	Ring Indicator	22
2	DSR	Data Send Ready	6
3	RTS	Request to Send	4
4	GND	Chasis Ground	1
5	SOUT	Serial Out	2
6	SIN	Serial In	3
7	GND	Signal Ground	7
8	CTS	Clear to Send	5
9	DTR	Data Terminal Ready	20
10	DCD	Data Carrier Detect	8

"VME SIO" Board

Jumpers

These are the jumpers you need:

Jumper E1-E10, E16-E39	Meaning
E2-E3	Centronics Printer
No jumper on E10	Centronics interface
E5, E6, E17	Interrupt Levels
E7	Jumper present for >= 16 Mbytes of memory. Remove it for < 16 Mbytes.
E8	Present for 8 ports, remove for 16 ports.
E9	No jumper on E9 Allows 19200 baud
E23,E28-E32, E29-E33 E30-E34,E27-E31,E35-E39	Bus Request Levels

Address Jumper Settings E11-E15

Board #	Jumpers Address
Board 0 NO address jumpers	FFF520
Board 1 E12	FFF580
Board 2 E11	FFF5A0
Board 3 E12,E13	FFF5C0
Board 4 E14	Dff520
Board 5 E13,E14	DFF560
Board 6 E12,E14	DFF80
Board 7 E12,E13,E14	DFFC0

2.7.10 New Serial Enhancements

With RISC/os 4.50, all machines will see improved modem control, and on machines with DigiBoards (M/120 and RC3240), RTS/CTS and DTR handshaking support is included. In addition, RISC/os 4.50 now supports 19200 and 38400 baud serial lines on the M/120, RC3240, and Rx3230 on the DigiBoard ports.

Two new handshaking protocols have been added to RISC/os 4.50: rtc/cts and carrier/nocarrier (mdmbuf). The stty commands associated with these are:

stty [-]rtscts
stty [-]mdmbuf

The stty setting **mdmbuf** sets DTR handshaking. This method of handshaking relies on the fact that DCD is tied to DTR. The serial driver only reacts to changes in DCD and not DTR; therefore, a null modem that ties these lines together must be used. Notice that this implies the use of a direct connect (i.e., no modem) and that **clocal** is set. If **clocal** is not set (i.e., **-clocal**), when DCD drops, the line will be disconnected, which is not the desired effect.

A new **stty** setting, **rtscts**, sets RTS/CTS handshaking. This causes the port to raise RTS when input can be accepted and drop it when input may not be accepted. The port will suspend output when it detects CTS is false and resume output when it detects CTS is true.

Due to hardware idiosyncrasies not all hardware supports all forms of handshaking. The following table describes which hardware supports which features:

Machine/Port	RTS/CTS	MDMBUF
M/120 and RC3240		
ttyd*	yes	yes
tty0 and tty2	no	no
tty1 and tty3	no	yes
M/1000, M/2000, and RC3260		
tty[hijkl]*	no	yes
tty0	no	no
tty1	no	yes
Rx2030		
tty0 and tty1	no	no
Rx3230		
tty0 and tty1	yes	yes

New ioctl()s:

TIOCMSET	set modem status.
TIOCMGET	get modem status.
TIOCMBIC	clear status modem according to mask.
TIOCMBIS	set modem status according to mask.

Refer to the *tty*(4-BSD) man page for full documentation.

2.8 Reporting Problems

Report any problems concerning release installation, software or hardware operation, or documentation errors to the MIPS Customer Service Center. The center will be able to solve your problems more quickly and efficiently if you provide as much information as possible listed on the form included at the end of this section. An on-line version of this bug form can be found in */usr/pkg/rn/bug.form* to assist in electronically mailing in bug reports. Bug reports can be reported in one of the following ways:

MIPS Customer Service Telephone Numbers:
US and Canada: (800) 443-MIPS
International: +1 415 330 7966

Mail comments to:
     Customer Service
     MIPS Computer Systems, Inc.
     950 DeGuigne Drive
     Sunnyvale, CA 94086

If you are submitting a question or a bug report that is medium to low priority, send it to the following electronic mail address, and we will enter it or act on it within five working days:

     csmail@mips.com or
     {ames,pyramid,pacbell,uunet}!mips!csmail

If you are submitting a high priority bug report, mail your message to csmail, then call the Customer Response Center and tell the support engineer that you have sent urgent mail.

2.8.1 Bug Tracking System

MIPS has developed a custom system for tracking bugs. At the end of this section, we provide a bug report form reflecting the system that you can reproduce, fill out, and send to us as needed. The explanations for the relevant fields are as follows:

Synopsis

A one line description of the problem.

User Severity

Choose from the following:

- **severe**
  A fix is of the utmost importance. For example, a production system crashes and is completely crippled; a development project is stopped and cannot progress until the problem is fixed; the problem causes data to be lost or corrupted. Problems of this severity may result in special scheduling of a plan for a fix.

- **serious**
  A fix is very important; work can progress but is seriously disrupted or hindered. Examples: important functionality of a production system is lost; important development tools are unavailable due to the problem; an OEM customer can proceed with development work, but cannot ship product until a fix is supplied by MIPS.

- **nuisance**
  The problem can be worked around readily; some advertised functionality is missing or incorrect. A fix should be supplied, but can probably wait until the next regular software release from MIPS.

- **cosmetic**
  The problem is primarily a "rough-edge". Example: an illegal usage of a command causes it to dump core. These should be fixed, but the fix can definitely wait until the next regular release from MIPS.

- **enhancement**
  The report requests a modification or addition to the functionality of the software, which would be nice, but is not urgently required. Note that it is conceivable that some problems which are actually requests for enhancements may be given a higher severity level, if the lack of the enhancement seriously hinders the use of the MIPS system for the user's purposes.

Description

The information is intended to give as much detail as possible about the problem, and should be sufficient to allow it to be reproduced by MIPS. Bug reports which do not allow the bug to be reproduced may be rejected. The description should contain source files and/or session scripts wherever practical.

Some things to try to include to help us trace the bug:

1. What version the OS and utility is used. Use **uname -A** to determine the OS version, and use **which** to determine the command path, then use **what** with the command path as an argument to determine the version number. For instance, if **cat** is causing problems, report the results from **which cat** and **what /bin/cat**.

2. What is the exact command line, e.g. **cc -c -g3 -O myprog.c**, and environment which reproduces the problem?

3. If a diagnostic or error message was printed, what was the exact wording of that message?

4. If the compiler reports a *fatal error*, it usually specifies which portion of the compiler is at fault (e.g., **fcom, ugen, asl**, etc). Most portions of the compiler reside in the directory */usr/lib*; a few, like the link editor, reside in */usr/bin*. After determining which component is at fault, and in which directory it can be found, execute the following commands:

Command   **Explanation**

```
# script              Make a typescript file.
script started, file is typescript
# printenv            Display environment variables.
HOME=/user/joe
SHELL=/bin/csh
TERM=wyse60
LOGNAME=joe
PATH=/bin:/usr/bin:/user/joe:/user/local/bin
# which cc
/bin/cc
# what /bin/cc
 /bin/cc:
    RISCompiler C 2.11 (from RISC/os 4.50)
# cc -c -g3 -O myprog.c   Command that caused compiler failure.
Fatal error in fcom ...
# dbx /usr/lib/fcom core
(dbx) where       dbx will print a trace of the stack
(dbx) printregs   dbx will print the registers
(dbx) $pc/i       dbx will print the offending instruction
(dbx) $pc-16/10i  dbx will print the vicinity of the offending instruction
(dbx) quit!
# exit
```

This produces a file called *typescript* in the current directory which can be printed and sent to MIPS.

These suggestions will not cover every problem (if, for example, a program compiles correctly but executes incorrectly, often information specific to that program is needed), but they will go a long way toward eliminating situations where MIPS cannot reproduce a bug and must later go back to the customer and ask for more information.

Package

This specifies the package in which the bug occurs. To be complete, a package name and version number should be given. Examples:

    riscos 4.50
    cmplrs 2.11
    dwb 1.0B

OS

This identifies under which operating system version(s) the bug exists.

Configuration

This information is used to record configuration information allowing MIPS to identify what version(s) and combinations of hardware and software are being used on the system(s) experiencing the bug. This information can be very important in allowing MIPS to reproduce hardware-dependent bugs. At a minimum, the following information should be provided:

    machine type (Rx2030, Rx3230, RC3240, M/120, M/500, 800, 1000, 2000, RC3260, RC6280)
    memory configuration (how much, composed of what model boards)

The **uname -A** command and **/etc/hwconf** can be used to provide machine attributes. Other hardware configuration information should be provided if it is applicable to the problem at hand. For example, a problem wherein a screen oriented program outputs garbage should include the make and model of the terminal(s) which manifest the problem; a problem with the disk formatting utility should include the make and model numbers of the disk controller(s) and disk(s) involved.

The more detail provided, the better. If it is possible to do so, include the revision level of involved hardware components. This additional information can save the delay of an extra round of communication between the bug submitter and MIPS.

Suggested Workaround

This information allows the bug submitter to suggest a workaround. This field is optional. MIPS may pass on this solution to other users experiencing the problem, so providing a workaround can be a service to the entire MIPS software user community.

MIPS Bug Report Form

Synopsis: _____

_____

User severity: _____

Description:

_____

_____

_____

_____

_____

_____

_____

_____

_____

Package/Version: _____   OS: _____

Configuration: _____

_____

_____

Suggested workaround: _____

_____

Report date: _____   Submitter name: _____

Organization: _____

Submitter address:

_____

_____

_____

Electronic address: _____   Phone: _____

3. RISC/os 4.50

This section describes new features, problems resolved, and known problems with RISC/os 4.50.

3.1 Hardware Support

RISC/os 4.50 provides support for the new Magnum 3000 and RC3230 systems. For the remainder of this document, both systems will be referred to as the Rx3230 system, except for items specifically related to one or the other. For more information, see the subsection titled **Rx3230** in this section.

3.1.1 DigiBoard Serial I/O

RISC/os 4.50 has been enhanced to allow non-contiguous DigiBoards. This is especially useful when supporting 16 port cards, since the addresses on these cards are not easily changed. See the Troubleshooting Tips section for more information.

3.1.2 Exabyte 8mm Tape Drive

The switch settings for the Exabyte drive must be as follows: switches 1 thru 4 should all be on, switches 5 thru 8 should all be set off.

The minimum prom revision for the Exabyte drive that is **required** for the Rx3230 is 4S25.

3.1.3 Boot Proms

In conjunction with the RISC/os 4.50 release, there is a 5.xx release of boot proms. The 5.xx release of the proms have a new naming convention:

   X.yz

where X is the major prom release number, y is the platform (machine) type, and z is the minor prom release number. The platforms are as follows:

0   M/500, 800, 1000
1   M/120, RC3240
2   M/2000, RC3260
3   RC6280
4   Rx3230

The major prom release number at the time of the RISC/os 4.50 is 5. Note that it was not necessary to release proms for all systems at this time. The only proms released in conjunction with RISC/os 4.50 are 5.10, for the M/120 and RC3240 systems, and 5.40, for the Rx3230 systems. Note that systems already in the field do not require these new proms to run RISC/os 4.50, but that new machines shipped from MIPS will be built with the new proms.

Note that the Rx2030 prom releases are handled separately from these proms.

3.1.4 Controller Firmware Revision Levels

RISC/os 4.50 requires some upgrades for controller firmware. The following is a list of what we recommend for firmware revision level:

4200 Disk Controller

    040 Minimally acceptable level. Does not support MACSI

    050 Currently shipping. Supports MACSI.
          RISC/os 4.50 works best with this.

4400 Quad Disk Controller

    04C Does not work with MACSI

    05A Currently shipping. Supports MACSI.
          RISC/os 4.50 works best with this.

4210 SCSI Controller

    XOW Unacceptable.

    XOY Minimally acceptable for non-disk devices. Has many bugs.

    XAD Minimally acceptable for non-disk devices. Has many bugs.

    XAF Minimally acceptable level for non-disk **and** disk
          devices. Has bugs with variable-length tape records.

    XAL Currently shipping.

3.1.5 Rx3230

- When you first bring up the Rx3230, it will not have a *hostname, internet address, netmask, broadcast address,* or *domainname.* It will ask you for each of these items on the console. If you take the defaults you will get a login prompt (RISCwindows if you have a graphics display). Answer the questions to configure your machine for the network.

- The Rx3230 prom has some environment variables that it uses to help configure the system. This is a list of those variables and their meaningful values:

bootmode	m	default (boot to monitor prompt)
	c	cold boot to Unix
	e	Error running diagnostics
bootfile		dksd(0,0,8)sash (default)
		bfs()server:/myunix_pathname option1 ... (for diskless)
console		console output goes to:
	a	all (tty0, tty1, mono, and color)
	l	(the letter 'el') color or mono or tty1 (default)
	r	(color or mono) and tty1
	1	tty1
	0	tty0
use_bootparams	0	disk boot
	1	diskless boot
keyswitch	0	

- The **askme** option on the boot line is not functional with a keyboard on the Rx3230. The **root=** option can be used in its place to specify the root file system.

- If you are using a headed Rx3230 and a UNIX keyboard, the **keyswitch** prom variable needs to be set to 0.

- Floppy jumper settings are jumper D0 should be on and jumper OP should be on.

-

QIC Jumper settings are:
There are 9 jumpers on the QIC tape.
Looking at the unit from the rear with the power connector at the top left
corner, the jumpers are in the lower left corner. The jumpers look as
follows:

```
              --   --   --

              --   --   --

              --   --   --
```

The correct jumper settings are:

```
              --   XX   XX   4
              --   --   XX   2
              XX   XX   --   1


              ^    ^    ^
              |    |    |____  scsi address (SCSI id 6)
              |    |_____  disconnect size: 16k
              |_____  parity
```
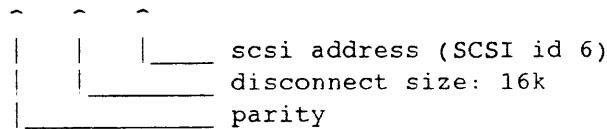
The default SCSI id for the QIC tape is 6.

- Exabyte Jumper settings:
  The SCSI id is set by looking at the unit from the rear with the dipswitches at the upper
  left corner. Switch 3 is bit 2, Switch 2 is bit 1 and Switch 1 is bit 0. SCSI id 4 is Switch 3
  ON and Switch 1 & 2 OFF. The Exabyte has 8 internal switches.

  Looking at the unit from the right side, you can see through the small fins on the second
  board from the rear there are 8 switches. The first four, on the left, must be on and the
  last 4, on the right, must be off. These switches are labelled U30 by Exabyte on the MX
  card and the switches are J1 through J8. J1-J4 are on and J5-J8 are off. Unfortunately,
  you cannot read any of these labels.

  The default SCSI id for the Exabyte tape is 4.

- 200 MB Maxtor drive jumper settings:
  There is a set of 5 jumpers on the underside of the drive. They are right next to the
  power connector. The power connector is labelled J3 and the selectable jumpers are
  labelled J6. From left to right holding the unit with the jumpers/power connector toward
  the top the two leftmost jumpers should always be on. The first jumper enables parity
  checking, the second jumper is the motor start jumper which spins up the disk if installed.
  The next 3 jumpers are the SCSI id jumpers. They are bits 2, 1, and 0. SCSI id 6 would
  have jumpers on the first four connectors from left to right and none on the last
  connector. The default SCSI ids for the 200 MB drive are 0, 1, & 2.

- 663 MB & 1 GB CDC/Seagate drive jumper settings:
  Looking at the drive on end with the power connector on the right, the jumpers are on the
  left. If the jumpers are labelled 1 to N then you might have a drive with either 8 or 9
  jumpers. Jumpers 6 and 7 must be installed. Jumper 6 is the parity jumper which enables
  parity checking and jumper 7 is the motor start jumper which stops the drive from spining
  up on power up. Jumpers 3-5 are the SCSI id jumpers. 3 is bit 2, 4 is bit 1 and 5 is bit 0.
  SCSI id 6 would have jumpers installed in 3, 4, 6 and 7. Jumpers 8 and/or 9 are unused.
  The default SCSI ids for the 663 MB drive are 0, 1, 2, 3, 5. SCSI id 4 or 6 can be used
  depending upon which tape is in the system.

- If the RC3230 has a keyboard, the default console is the color screen, if it exists. If there
  is no color screen, the mono screen is the console. If there is no keyboard, the console is
  tty1. The mouse is on tty0.

3.2 New Features

The following sections describe new features and general enhancements added to RISC/os 4.50.

3.2.1 NFS

RISC/os 4.50 includes release 4.0 of the Sun NFS code, including external commands supporting NFS. This replaces the RISC/os 4.00 file system dispatch structure (via the File System Switch, FSS) with an implementation based on the *vnode* model, including conversion of the current Fast File System (FFS), common file operations and */proc*. The conversion is binary compatible with disks under RISC/os 4.00.

The following external commands are supported:

automount	rpc.mountd
exportfs	rpc.rstatd
chkey	rpc.ruserd
keylogin	rpc.rwalld
keyserv	rpc.sprayd
mount	rpc.yppasswdd
ndbootd	rpc.ypupdated
netdisk	rpcinfo
nfsstat	rwall
rexd	showmount
rpc.bootparamd	spray

RISC/os 4.50 also includes the NFS quota mechanisms and supporting utilities. This includes the *quota*, *quotactl*, and *setquota* system calls and the following commands:

edquota(1M)	edit user quotas
quota(1)	display disk usage and limits
quotacheck(1M)	check file system quota consistency
quotaon, quotaoff(1M)	turn file system quotas on and off
repquota(1M)	summarize quotas for a file system

3.2.2 NFS Writes

NFS typically performs synchronous writes on a file server. This enables an NFS server to crash and reboot without causing a client to suffer, which is especially useful during a compilation. However, in doing this, NFS clients suffer tremendous performance penalties because every write must be written to disk before a write request can return. Simple testing suggests a 30-70% performance slow down.

In MIPS' implementation, we allow asynchronous writes on a file server. This is and always has been the default mode. In a typical situation, a client's data integrity will decrease slightly with much faster throughput. For non-transactional tasks, the added performance is well worth the risk. If, however, you require absolute data integrity, an option exists which enables synchronous writes on a per file system basis.

See the **nfs_sync** and **nfs_async** options described in *fstab*(4). These are server side options which affect locally mounted but exported file systems. All NFS clients mounting these file systems will see the same write semantics. Therefore, we envision system administrators mounting critical file systems with the **nfs_sync** option. Global default setting of this option can be accomplished through the *kopt*(8-BSD) command.

3.2.3 VIS

The MIPS Virtual Information Service (VIS) provides the ability to use multiple information provider services on a single host. New in RISC/os 4.50, VIS supports Sun's Network Information Service (NIS, formerly known as Yellow Pages or YP) and the DARPA Domain Name Service, as well as the standard UNIX information files. VIS is implemented as modifications to the library routines which provide interfaces to these information databases, such as *getpwent*, *gethostbyname*, etc. This significantly eases the task of application development because the application itself does not need to know about the different information sources.

If the */etc/vis.conf* file is empty or does not exist, the default behavior is exhibited. A sample */etc/vis.conf* is as follows:

```
host: files dns nis
net: files nis
netgroup: files nis
services:  dns files nis
proto: files nis
rpc: files nis
host: files nis
group: files nis
passwd: files nis
```

The line of the form:

```
passwd: files nis
```

configures VIS so that a local file is used for the special user id's. This causes the local *passwd* file to be used first, but if the user is not listed in this file, the NIS facility is used. See the *vis*(4) man page for more information.

Important Note For RISC/os 4.50 BETA Customers:

If you received a BETA version of RISC/os 4.50 and you used the *vis.conf* file, you must replace all occurrences of 'yp' with 'nis' before performing the update to RISC/os 4.50.

3.2.4 NIS

RISC/os 4.50 provides integrated support for NIS including peaceful coexistence with *named*. The following commands are supported:

ypcat	**ypmatch**
yppasswd	**ypwhich**
ypbind	**ypinit**
ypmake	**yppasswdd**
yppoll	**yppush**
ypserv	**ypset**
ypupdated	**ypwhich**
ypxfr	

Please note the discrepancy between the user id's representing such "special" users as "bin" and "adm" in System V and Berkeley UNIX based systems will create problems if NIS is used for the entire *passwd* file on RISC/os systems in a mixed environment. MIPS suggests using a local *passwd* file for these kinds of "special" user entries, and sharing information only on "real" users with NIS.

32

Please see the appropriate man pages and the chapter on NIS in the *System Administrator's Guide* for more information.

3.2.5 SLIP

SLIP, or Serial Line Interface Protocol, provides TCP/IP protocol family over point-to-point serial connections. It is merely a packet framing protocol: SLIP defines a sequence of characters that frame IP packets on a serial lines, and nothing more. It provides no addressing, packet type identification, error detection/correction, nor compression mechanism.

SLIP is implemented as a stream module that interprets data coming from below it in the stream according to the SLIP protocol description. It unwraps asynchronous frames and forwards the contents to the IP network code. It also creates network interfaces and receives packets from these which it wraps into SLIP frames, and sends downstream to the terminal ports. For more information, please refer to *slconfig*(1M), *slattach*(1M), and *slip*(7N).

3.2.6 Terminal Support

RISC/os 4.50 supports more than 66 local terminals (RISC/os 4.00 supported a maximum of 66) and at least 1024 pseudo-terminals (via **telnet** or **rlogin** only; 256 via most other users of pseudo-terminals).

3.2.7 Secure RPC

Secure RPC is an authentication system that greatly improves the security of network environments. The system is general enough to be used by other UNIX and non-UNIX systems. RPC is at the core of the new network security system. Its authentication is open-ended: a variety of authentication systems may be plugged into it and may coexist on the network. Currently, MIPS has two: UNIX and DES. UNIX authentication is the older, weaker system; DES authentication is the new system included in RISC/os 4.50. Details of how secure RPC works and how to configure necessary files are described in the *System Administrator's Guide*.

Part of RPC is **rpcgen**, a compiler that accepts a remote program interface definition written in a language, called RPC Language, which is similar to C. It produces a C language output which includes stub versions of the client routines, a server skeleton, XDR filter routines for both parameters and results, and a header file that contains common definitions. The client stubs interface with the RPC library and effectively hide the network from their callers. The server stub similarly hides the network from the server procedures that are to be invoked by remote clients. **rpcgen**'s output files can be compiled and linked in the usual way. The developer writes server procedures (in any language that observes MIPS calling conventions) and links them with the server skeleton produced by **rpcgen** to get an executable server program. To use a remote program, a programmer writes an ordinary main program that makes local procedure calls to the client stubs produced by **rpcgen**. Linking this program with **rpcgen**'s stubs creates an executable program. (At present the main program must be written in C). **rpcgen** options can be used to suppress stub generation and to specify the transport to be used by the server stub.

Like all compilers, **rpcgen** reduces development time that would otherwise be spent coding and debugging low-level routines. All compilers, including **rpcgen**, do this at a small cost in efficiency and flexibility. However, many compilers allow escape hatches for programmers to mix low-level code with high-level code. **rpcgen** is no exception. In speed-critical applications, handwritten routines can be linked with the **rpcgen** output without any difficulty. Also, one may proceed by using **rpcgen** output as a starting point, and then rewriting it as necessary.

3.2.8 Improved Virtual Memory Support

The virtual memory system in RISC/os 4.50 is substantially improved over that in previous RISC/os releases. It now efficiently handles processes which have virtual spaces much larger than physical memory.

The operation of the new virtual memory system is now largely self-regulating. As a consequence, the following tunable or configurable kernel parameters are no longer meaningful:

        tune.t_gpgslo
        tune.t_gpgshi
        tune.t_gpgslmsk
        tune.t_gpgshmsk
        tune.t_vhandr
        tune.t_vhandl
        tune.t_maxsc
        tune.t_maxfc
        var.v_vhndfrac

Also note that because the internal structure of the RISC/os 4.50 kernel is so much different than previously released kernels in the VM area, any programs that open /dev/kmem must be re-written. Please see mem(7) for more information.

3.2.9 I/O Performance Improvement

In previous RISC/os releases, every I/O transfer through a raw (character) disk device would flush the entire cache. In RISC/os 4.50, only the appropriate portions are flushed, resulting in much better I/O performance.

3.2.10 /proc

In RISC/os 4.50, the /proc filesystem, like all other filesystems, has been converted from operating with the SVR3 filesystem switch to operate with the vnode-based filesystem model used in RISC/os 4.50. There are no functional changes to /proc nor is the interface changed in any way. The only user-visible change is one header file, /usr/include/fs/prinode.h, used for kernel internal structures and not likely to have been included in compilation of application programs. This file was removed from the system and the file /usr/include/sys/fs/proc_fs.h was added to perform similar functions.

3.2.11 SNMP

RISC/os 4.50 includes support for the SNMP (Simple Network Management Protocol) and supporting utilities from the SNMP distribution. This includes adding SNMP support in our protocol stack and supporting utilities.

The Simple Network Management Protocol (SNMP) allows network entities to be managed remotely. It has been chosen by the Internet Activities Board (IAB) as the standard by which TCP/IP Internets are to be managed. It communicates with remote network elements over UDP. The information which SNMP manages is contained in the Management Information Base (MIB), the collection of variables which together present a virtual representation of the state of all constituents of a managed network. SNMP enables a network station to get and set variables in the MIB of a remote element.

The MIPS implementation of SNMP is based on the work done by James R. Davin at the Massachusetts Institute of Technology's Laboratory for Computer Science. This provides the basic protocol agent (an agent or server is the local entity resident on network elements which

34

supports the network management function). The agents we support are /usr/etc/snmpd (an SNMP agent) and /usr/etc/snmptrapd (an SNMP trap agent for SNMP trap messages). Our agent has support for reading all SNMP variables but does not include support for the SNMP set function. In addition to the agents, RISC/os 4.50 supports the following routines for accessing agents:

snmpget	retrieve an SNMP variable.
snmpnext	retrieve the next SNMP variable in lexographic order.
snmpset	set an SNMP variable.
snmptrap	send an SNMP trap message.
snmpwalk	walk through a branch of the MIB.

For more information concerning SNMP, please consult RFC1065, RFC1066, RFC1067 and the man pages for the various commands.

3.2.12 Accounting and Printer Daemon Installation

Prototypes of the various types of accounting and printer daemon invocation and shutdown have been provided to ease installation. The system startup scripts /etc/rc2.d/S22account and /etc/rc2.d/S76printer invoke /etc/init.d/account and /etc/init.d/printer respectively. These may be linked to the appropriate prototype in /etc/init.d.

Accounting:

init.d/acct	starts/stops System V Process Accounting data collection
init.d/bsd43_accton	starts/stops BSD 4.3 Process Accounting data collection
init.d/noacct	dummy file for no accounting data collection

Printer daemons:

init.d/bsd43_lpd	starts/stops **lpd**
init.d/lpsched	starts/stops **/usr/lib/lpsched**

Similarly, the scripts /etc/rc0.d/K21account and /etc/rc0.d/K20printer invoke /etc/init.d/account and /etc/init.d/printer respectively for cleanly shutting down the accounting and printer services. The creation of the link mentioned above will also cause the service to be shutdown, as the prototype scripts contain both startup and shutdown functions.

Should you wish to run System V Accounting on your system, it is also necessary to invoke data reduction scripts from **cron**. The command (script) **/usr/lib/acct/install_acct** adds the necessary entries to /usr/spool/cron/crontabs/adm and links the prototype /etc/init.d/acct to /etc/init.d/account. You may wish to adjust the time at which accounting data is processed (by default 4:00 AM) if your system has other periodic activities scheduled for this time.

In RISC/os 4.50, the process accounting data is collected in /usr/adm/pacct for both System V and BSD 4.3 accounting. The 4.3 BSD programs which process this data have been modified to default to this file. 4.3 BSD systems usually default to /usr/adm/acct for this purpose, but System V accounting requires that this be a directory.

3.2.13 Rationalization of Login Accounting Data Files

Previous versions of RISC/os have been inconsistent in what file was used for collection of login accounting data. Programs were approximately evenly split between /usr/adm/wtmp and /etc/wtmp. For RISC/os 4.50, all programs now use /etc/wtmp. If you create programs which are surrogates for **login**, they should use /etc/wtmp. This name is contained in the define of WTMP_FILE found in /usr/include/utmp.h. Such programs should not create the file, and should add entries to it only if it already exists. The following code fragment from **login**

illustrates the correct use of the file:

```
/*      Now attempt to write out this entry to the wtmp file if we        */
/*      were successful in getting it from the utmp file and the          */
/*      wtmp file exists.                                                 */

        if (((fp = fopen(WTMP_FILE,"r+")) != NULL)
                )) {
                fseek(fp,0L,2);             /* Seek to end of file. */
                fwrite(&utmp,sizeof(utmp),1,fp);
                fclose(fp);
        }
```

3.2.14 System Administrative Scripts Revised

The System Administrative Scripts available through the sysadm command have been revised to take advantage of the features found in RISC/os. In particular, the scripts in the file management section now invoke dump.ffs and restore.ffs for backup and restore respectively. This change improves the speed of these operations. These changes mean that any tapes that were created with the older scripts must be processed manually for restoring files. In previous versions of RISC/os the backup script was only marginally functional, so it is unlikely that customers will have created tapes with it. These tapes have a volume label, followed by a cpio format tape.

3.2.15 Log File Roll Scripts

RISC/os 4.50 contains a number of scripts used to "roll" various log files. That is, the scripts save the last n log files and restart a new one. The n+1 log file is discarded by the scripts. These scripts are all located in the directories under /usr/adm/periodic. The following is a list of log files that are rolled, the frequency of this rolling, the number of files saved, and the script that does the work:

File	Frequency	Number Saved	Script
/usr/adm/gated.log	hourly	3	hourly/XX.gated_log.SAMPLE (sample only)
/usr/lib/cron	daily	4	daily/10.cronlog.system
syslog output	daily	5	daily/10.syslogd.system (rolls all files mentioned in /etc/syslog.conf)
/usr/lib/spell/spellhist	monthly	3	monthly/10.spellhist.system
/etc/wtmp	monthly	4	monthly/10.wtmp.system

The gated and routed daemons can optionally be directed to log their actions. Such logging can be an invaluable source of insight into complex network routing problems; however, when logging is enabled, potentially enormous log files can be created in active networks with many systems. The file /usr/adm/periodic/hourly/XX.gated_log.SAMPLE is a sample file for rolling the gated log files once per hour if they have gotten beyond a specified size. If you enable gated or routed logging, you should copy this file to a file named something like 10.xxxx.system (the xxxx is arbitrary), in the same directory (/usr/adm/periodic/hourly) and edit the copy to roll the log files. Because the administrator can select the name of the log file, this copy file must be edited to add the proper names of the files to be rolled.

The 10.syslogd.system script rolls all the output files used by the System Logging Daemon (syslogd). It parses the daemon configuration file, /etc/syslog.conf, and rolls the files mentioned there.

The spell command will optionally keep track of words it feels are misspelled in the file /usr/lib/spell/spellhist if this file exists (the local administrator must create it if desired; it is not automatically created). The intent is to add words to the dictionary that are commonly used by your local users, by periodically examining the *spellhist* file and using the **hashmake**, **spellin**, and **hashcheck** programs. This *spellhist* file will now be rotated monthly, saving the last three months by the script /usr/adm/periodic/monthly/10.spellhist.system. Note that the *spellhist* file must be generally writable in order to capture the output from all users.

The process accounting file, /etc/wtmp will be rolled by the script **10.wtmp.system** when you have not enabled System V accounting. If System V accounting is enabled, this file is rolled as part of the nightly accounting runs as part of the script **/usr/lib/acct/runacct**, which is normally invoked by **cron**. Note that process accounting is optional, so if the file /etc/wtmp does not exist, none of the programs which write to it will create it, and the roll script for it will do nothing.

Other New Periodic Scripts

Several other new periodic scripts are provided in this release:

Hourly:

10.ttys.system	rebuilds a copy of /etc/ttys from *inittab*

Daily:

50.calendar.system	runs the command **calendar** to process user's calendars
50.checkdev.system	looks for files in /dev and deletes them
50.df.system	does a **df** to check on the space available

Weekly:

10.update_bsd_find_db.system	rebuilds the database used by /bsd43/bin/find

3.2.16 Administrative Script to Configure A More 4.3 BSD-Like System

A new administrative script has been provided to set various options so that the system will behave more like a 4.3 BSD environment. This script, **bsd_like**, is part of the System Setup Menu, **syssetup**. The script is invoked by first invoking the **sysadm** command, then selecting the **syssetup** menu, and from it selecting the **bsd_like** menu. As with other **sysadm** scripts, on line help is available. The **bsd_like** script touches the following sentinel files (i.e. files which if present cause programs to alter their behavior from the default):

- /etc/login.initgroups.ok - causes **login** to use the *initgroups()* call to put the user in all of the groups to which s/he belongs

- /etc/login.nohome.ok - allows logins when the user's home directory does not exist. The directory / is used instead.

- /etc/login.quotawarn.ok - **login** will warn the user if s/he is over quota.

- /usr/lib/sendmail.ok - (separately optional) causes mail to use **sendmail** if it is running.

The **bsd_like** script also attempts to alter the system startup script **/etc/init.d/set_kopts** (also new in this RISC/os 4.50) to enable the setting of several kernel configuration options (**kopts**). This script may be locally modified, so if the **bsd_like** script is unable to automatically modify **set_kopts**, it describes the necessary changes. The following **kopts** are set:

- **_posix_chown_restricted** - restricts the use of **chown** so unprivileged users cannot give away files. Also restricts the use of **chgrp** so that the group owner of a file may not be changed by an unprivileged user to a group to which the user does not belong.

- _riscos_group_parent - newly created files inherit the group of the directory in which they reside rather than the primary group of the creator.

- _riscos_kill_stopped_orphans - when the parent of a traced process exits, the child will be continued and sent a SIGKILL signal.

- _riscos_utime_owner - only the owner or the super-user may issue the *utime()* call to change the times on a file.

There are other options and sentinel files which a given site may wish to set. Please refer to the **login** and **kopt** manual pages for more information.

4.3 BSD systems restrict access to logged in terminals by changing them to be owned by the group tty, with permissions of 620 (-rw-r----). Programs which should be able to access another terminal are made to be owned by the group tty and have the setgid bit set. The bsd_like script allows the system administrator to have RISC/os also have this behavior. This is a separate option in the script, since imposing this restriction may cause other programs (than those supplied in RISC/os) to fail if they access another terminal. This is because System V does not place these restrictions on the programs which access another terminal, and other programs which access another terminal may not be setgid tty. The group tty must be added before this feature is enabled. Note that the numeric value for this group in 4.3 BSD systems, 4, has already been used for the group adm in System V systems like RISC/os. This feature creates the file */etc/login.mygroup.ok* and makes the following programs setgid:

/bin/write
/bsd43/bin/talk
/bsd43/bin/wall
/bsd43/bin/write
/etc/dump
/etc/rdump
/etc/wall

3.2.17 RISC/os 4.00 Compatibility Package

An optionally installed subpackage provides a compilation and linking environment essentially identical to that found in RISC/os 4.00, 4.10, 4.20, and 4.30. This package should be installed if your system makes use of unlinked object files (*.o) or libraries produced under RISC/os 4.00. For consistent results, source files must be compiled and linked in the same environment in which the unlinked binaries or libraries were produced. Also included in this compatibility package is Release 2.0 of the C Compiler which was distributed as part of RISC/os 4.00.

Generally, the compatibility package is only necessary if either of two situations are present for your system:

1. You have installed software which includes compilation and linking against binaries or libraries which were produced under RISC/os 4.00. For example, most of the RISCware Data Base Management systems include such software.

2. You need to produce binaries that will run on both RISC/os 4.00 and RISC/os 4.50. RISC/os 4.50 binaries will generally not operate properly on RISC/os 4.00 due to functional improvements in the libraries. With very limited exceptions noted elsewhere, RISC/os 4.00 binaries or binaries produced with the compatibility package will run on both RISC/os 4.00 and RISC/os 4.50 environments.

The compatibility package consists of two subpackages: one providing basic compatibility support, and the second which adds support for -systype bsd43 compilation and linking. Together, the packages occupy about 15 megabytes of disk space.

38

The compatibility packages are used by setting the environment variables COMP_TARGET_ROOT and COMP_HOST_ROOT to /RISCos4.0, as follows:

% setenv COMP_TARGET_ROOT /RISCos4.0

This forces the use of the compatibility package include files during compilation, and will link against the libraries there instead of the standard system libraries.

% setenv COMP_HOST_ROOT /RISCos4.0

This forces the use of the compiler found in the compatibility package in preference to the standard system default compiler. It is less common to require that the compatibility package's compiler be used; it is provided for completeness.

3.2.18 POSIX

RISC/os 4.50 includes support for the IEEE Standard Portable Operating System Interface for computer enironments, IEEE Std 1003.1-1988. Programs can be compiled using the POSIX interface by specifying -systype posix on the compile line. Note that only the library routines specified by IEEE standard 1003.1 have been implemented, so only a restricted set of programs will currently compile in this "universe".

3.2.19 4.3 BSD-Tahoe

The following 4.3 BSD-Tahoe commands have been added in RISC/os 4.50:

efl(1-BSD)	**lex**(1-BSD)
Mail(1-BSD)	**plot**(1G-BSD)
struct(1-BSD)	**style**(1-BSD)
users(1-BSD)	**vgrind**(1-BSD)
yacc(1-BSD)	

See the appropriate man pages for more information.

The following BSD commands are included in RISC/os 4.50, but are not from the 4.3 BSD Tahoe baseline:

chmod	**df**
du	**error**
from	**nroff**
ptx	**rdist**
refer	**rev**
sysline	**tbl**
tcopy	**tip**
touch	**tr**
troff	**tset**
tsort	**tty**
ul	**window**
yes	

RISC/os 4.50 does not include the following 4.3 BSD-Tahoe commands:

intro	**learn**
mkdep	**mset**
sendbug	**symorder**
tn3270	**tp**

| uusend | uuq |
| wait |

3.2.20 International Character Set Support

RISC/os 4.50 provides "8 bit clean" support for most programs. That is, most programs now process 8 bit characters, allowing for better support of non-US environments. This support permits file names to use all 8 bits. One major area of change has been in *libcurses*, the curses screen management library. Previous versions of *libcurses* and the programs built with it utilized the 8th bit for internal purposes. This has been changed, as well as a number of new features added to *libcurses* support. The major system editors support 8 bit characters. The only major exception to this 8 bit support is in the 4.3 BSD *libcurses*, and programs built with it. The **sysline** and **talk** programs from the 4.3 BSD environment, and most of the 4.3 BSD games, are not fully 8 bit clean due to their use of *libcurses*. Making commands 8 bit clean is in preparation for further internationalization support in future releases.

3.2.21 Pseudo Ttys

RISC/os 4.50 has been enhanced to support up to four major numbers for the *pty* pseudo-device, each with 256 *pty*'s (the minor number for the device). That makes the default 1024 *pty*'s on all systems except the Rx2030, where the default is 512. Each possible *pty* costs four bytes of storage, allocated at system startup time. The symbol MAXPTY may be defined in a *kernel.** file to override the default maximum. See the section titled **Binary Kernel Reconfiguration** for more information.

The new device */dev/ptcm* (10, 43) is the clone device for allocating *pty*'s from the larger set. Since it allocates first from *pty*'s above the first 256, and then only from the first 256 when the former are all in use, there is maximum compatibility with older applications which support only the first 256 *pty*'s. The new device, and the associated *ttyq** files, are created via the optional MKDEV target *optpty*. If the new clone device does not exist, clients which know about it fall back on using */dev/ptc*. This makes the clients compatible with systems which do not need the extra *pty*'s.

3.2.22 Mailers

The RISC/os 4.50 versions of **mailx** and **/bin/mail** now support the **metoo** option so that the sender is included in group expansion of aliases. In addition to this, the following changes have been made to the various mailers on the system:

Mail
/usr/ucb/Mail has been ported from 4.3 BSD to replace the symbolic link to */usr/bin/mailx* in RISC/os 4.00. */usr/ucb/mail* is a symbolic link to */usr/ucb/Mail*.

The */usr/lib/Mail.rc* file now exists to list the system-wide default **Mail** options. A template for this file might be:

```
set append dot save
ignore Received Message-Id Resent-Message-Id Status Mail-From Return-Path Via
```

The system administrator should set up this file if it does not already exist.

The default mailer for **Mail** is **/usr/lib/sendmail** and can be changed by setting the **sendmail** variable to other mailers, i.e., **set sendmail=/bin/mail**. The **sendmail** variable can bet set in the */usr/lib/Mail.rc* for a system wide setting, or for a more individual setting, in a user's *$HOME/.mailrc* file.

40

mailx

The */usr/lib/mailx/mailx.rc* file now exists to list the system-wide default **mailx** options. A template for this file might be: **set sendmail="/bin/mail"** to make sure mail is delivered by **/bin/mail**. This setting preserves the default System V behavior for **mailx**. The system administrator should set up this file if it does not already exist.

If desired, messages sent by **mailx** can be changed to be delivered by **/usr/lib/sendmail** by doing one of the following: **set sendmail="/usr/lib/sendmail"** or create a zero length */usr/lib/sendmail.ok* file by executing the command: **touch /usr/lib/sendmail.ok**

/bin/mail

The RISC/os 4.50 version of **/bin/mail** is the System V version of **/bin/mail** enhanced with **sendmail** capability. If */usr/lib/sendmail.ok* exists, it will attempt to deliver mail first by calling **/usr/lib/sendmail**, then by other means. The *sendmail.ok* file is used so that the default behavior is compatible with System V; */usr/lib/sendmail.ok* is not created by the default installation. See the *mail*(1-SysV) man page for information on this feature.

sendmail

In RISC/os 4.50, **sendmail** has been updated to revision 5.61 with MX (mail forwarding) record support for **named**. See the **named**(8-BSD) man page and the *Sendmail Installation and Operation Guide* (in the *System Administrator's Guide*) for more information on this feature.

The */usr/lib/sendmail.smtp* file and the associated periodic daily script (*/usr/adm/periodic/daily/15.sendmail.smtp.system*) that updates it has been removed from the system. The *sendmail.smtp* was used in pre-RISC/os 4.50 systems by the *sendmail.cf* file to resolve local host names. Because **named** is now used, this is no longer necessary.

A new sample configuration file, *sendmail.cf*, and the directory *cf* containing off-the-shelf configurations are provided in */usr/lib*. Be sure to fill in the "blanks" as shown in the instructions in the sample configuration file. This sample file is for Internet sites running the nameserver (or using */etc/hosts* host table wherein the fully qualified domain name of each host is listed first). Please refer to the *Sendmail Installation and Operation Guide* for information on how to build your configuration table, and the *Name Server Operations for BIND* in the *System Administrator's Guide*, on how to set up your **named** files.

Helpful Hints for Mail

- If using a remote nameserver running on other machines, be sure to create the file */etc/resolv.conf* to contain the following lines:

```
domain YOUR_DOMAIN_NAME                    e.g. domain mips.com
   nameserver ADDRESS_OF_NAMESERVER_1        nameserver 130.62.14.10
   nameserver ADDRESS_OF_NAMESERVER_2        nameserver 130.62.2.9
   :
   :
```

- add the following line to */etc/vis.conf*:

```
host: files dns
```

- If using */etc/hosts* to resolve host names, be sure to list a fully qualified domain name for each host in */etc/hosts* and create the file */etc/reolv.conf* containing the following line:

```
domain YOUR_DOMAIN_NAME          e.g domain mips.com
```

3.2.23 New 4.3 BSD Support

MIPS continues to expand 4.3 BSD support in RISC/os:

- RISC/os 4.50 provides better support for the user expecting 4.3 BSD compatibility by making the behavior of shell (**csh** and **sh**) built-in functions such as **kill** and **echo** depend upon the user's path settings. There are minor, but annoying, differences in these functions between the 4.3 BSD and System V environments. If */bsd43/bin* is ahead of */usr/bin* in the path then the shell emulates the 4.3 BSD behaviors for the built-ins, otherwise, it emulates the System V behaviors.

- The following new library routines have been added to */bsd43/usr/lib/libc.a* in RISC/os 4.50:

  usleep(3-BSD)
  vprintf(3S-BSD)
  vfprintf(3S-BSD)
  vsprintf(3S-BSD)
  strcasecmp(3-BSD)
  strncasecmp(3-BSD)

- The *bcopy* in */bsd43/usr/lib/libc.a* and in */usr/lib/libbsd.a* have been enhanced to support overlapped copying.

- The 4.3 BSD commands **systat**(1-BSD), **iostat**(1-BSD), and **vmstat**(1-BSD) are now included with RISC/os 4.50. See the appropriate man page for more information.

3.2.24 Quota Disk Usage System

The **quota** system is fully documented in the *Administrator's Reference Manual*, but general information and some caveats are listed here.

To use **quota**, NFS 4.0 semantics must be used. 4.3 BSD-Tahoe semantics are **not** allowed. NFS 4.0 semantics require adding the option *quota* to the options list of a filesystem. 4.3 BSD-Tahoe used the option *rq*. These changes must be seen at mount time, meaning remounting or re-booting is necessary.

To disable quotas on a file system, remove the *quota* option from the respective *fstab*(4) entry and delete the *quotas* file from the respective top directory. If the file is not deleted, **quota**(1) will report quotas on file systems where it finds the *quotas* file. For instance, the following files would have to be removed (or renamed) from these directories:

Directory	File
/	*/quotas*
/usr	*/usr/quotas*

When the hard and soft limits of a disk resource (blocks/inodes) are zero, the disk usage for that resource is disabled and hence is not kept current. The **quotacheck**(1M) program updates the quantities temporarily.

3.2.25 Automounter

RISC/os 4.50 supports the automounter. The *automount*(8) man page describes the automounter in detail, but it is important not to automount directories at levels where other non-automounted directories must still be accessed, i.e. the **root** directory, */*. If directories are automounted in such a place, the non-automounted directories will disappear. To avoid this we suggest you use */n/<machine-name>* as a mount point.

If the automounter is to be used, **named** should also be started to avoid system performance degradation. In starting **named**, the */etc/resolv.conf* file should reflect a local **named**. Details about **named** can be found in the *System Administrator's Guide*.

Note that the automounter wants control of the parent directory where something is mounted, so it is not a good idea to mix automounts with manual mounts. See Chapter 18 of the *System Administrator's Guide* for more information.

3.2.26 File System Changes

Support for the S51K filesystem is dropped in RISC/os 4.50. MIPS has been planning this change for quite some time and has given warning in past release notes. The standard filesystem remains the "ffs" filesystem. With RISC/os 4.50, the strings "4.3" and "ufs" are aliases for "ffs". See the *mount*(1M) man pages for more details.

Because the S51K filesystem is no longer supported, the following binary files have been removed from this release:

etc/clri.S51K	*etc/fsdb1b.S51K*
etc/ff.S51K	*etc/fsstat.S51K*
etc/finc.S51K	*etc/fstyp.d/20_fstyp.S51K*
etc/frec.S51K	*etc/labelit.S51K*
etc/fsck.S51K	*etc/mkfs.S51K*
etc/fsck1b.S51K	*etc/ncheck.S51K*
etc/fsdb.S51K	*etc/omkfs.S51K*

3.2.27 Shadow Password and Dialup Password Support

RISC/os 4.50 provides support for improvements in system security through the optional use of a shadow password file and dialup passwords. A common form of attack on UNIX systems is to attempt to determine system passwords by "brute force" methods in which a program tries to guess the password. The power of RISComputers make these brute force attacks a reasonably effective way of penetrating the system. These attacks are possible because the encrypted password is stored in a publicly readable file, */etc/passwd*. The program can encrypt possible passwords and compare them against the one in */etc/passwd* until a match is found. The program needs no special privileges to do this. To thwart this type of attack, the system now supports a shadow password file. The System Administrator invokes the program **pwconv** which processes the regular password file to delete the passwords and place them in */etc/shadow*, which is not readable by unprivileged programs. Should the administrator desire to revert to the "standard" password file scheme, **pwunconv** reverses the process. Users can manipulate their passwords as before using the **passwd** command (both System V and BSD 4.3 variants) and the BSD 4.3 commands **chfn** and **chsh**. The new administrator command **passmgmt** facilitates administrative management of the system. All of these programs are described in their manual pages in more detail. Also refer to the manual page for **login**.

Dialup passwords allow the system to place additional security via an additional password being required for externally connected tty lines. The file */etc/dialups* lists the tty lines which require an additional password. The file */etc/d_passwd* contains the passwords. A dialup password will be requested if the tty line is listed in */etc/dialups*, and the login shell for the user is found in */etc/d_passwd*. This is done so that service programs such as **uucp** can be configured to not require a dialup password. The format of */etc/dialups* is a list of devices, one per line, e.g. */dev/ttyh1*. The format of */etc/d_passwd* is lines of the form `shell:encrypted_passwd`, e.g. `bin/sh:qWerTyuioP`, where the password is encrypted in

the same way as other passwords. These features are described on the manual page for login.

3.2.28 Libcurses

RISC/os 4.50 includes the full System V Release 3.2 *libcurses.a*, for -systype sysv compilations, and the 4.3 BSD-Tahoe *libcurses.a* and *libtermcap.a* (also known as *libtermlib.a*), for -systype bsd43 compilations.

3.2.29 format For Rx3230 Systems

format has changed. It automatically calculates the partitions for the volume header.

3.2.30 sash For non-Rx2030 systems

sash has one new command: **sprobe**. This will probe the SCSI bus and identify all the devices attached. It is particularly useful when changing your hardware configuration. It verifies that devices are plugged in correctly with the correct SCSI id's.

3.2.31 Installation Tools Changes

The organization of files on the media have changed in RISC/os 4.50. Due to size restrictions, the kernels have been moved off the miniroot. The kernel is now booted directly from the tape. This process adds a few minutes to the length of the installation, but it allows the same tape to be installed on each MIPS machine. The installation tools allow you to copy the kernel from tape to miniroot once it is on disk. This step is not necessary nor entirely useful, but does allow one to run **ps** from the miniroot, and in case the miniroot is shut down but swap space is not cleared, rebooting the miniroot is faster.

The RISC/os 4.50 tapes include two versions of **sash** and **format**:

> **sash.std** and **format.std** for non-Rx2030 machines
> **sash.2030** and **format.2030** for Rx2030 machines

Only the versions for the appropriate system type are installed.

S51K Filesystem:
In RISC/os 4.50, support of the S51K filesystem has been dropped, and therefore, the installation will abort if a S51K file system is detected in */etc/fstab*.

Subpackage Listing:
As part of the installation, all subpackages are listed and it is possible to look at a listing of what files are included with each subpackage, which could be helpful in determining whether or not to install the subpackage.

Miniroot Additions:
The miniroot now has **pg** and **restore**.

Space Check:
The installation script asks the installer if he/she wishes to skip the space check. We recommend that the space check always be performed. If the space check is skipped and there is not enough disk space to complete the installation, the entire installation procedure will have to be repeated.

The space check now warns the installer if more that 90% of the disk will be filled by the installation. The installer may abort or continue the installation at that point.

44

Selecting Subpackages:
A list of all available subpackages and their descriptions are now presented on the screen before the selection process. The installer is given the option of selecting all of the subpackages by answering one question (which saves the time and effort of selecting all of the subpackages one by one.)

fstab
Because of new **/etc/mountall** and **/etc/fsck.ffs** commands in RISC/os 4.50, all file systems except root will **not** be fsck'ed at boot time unless the *letc/fstab* file is changed.

Rx2030 Installations
Please note that the default partition for */usr* on the Rx2030 systems is now partition 3, in order to allow RISCwindows to fit on a 172 megabyte disk drive. If you are doing an update install and your *usr* partition is not 3, be sure to do one of the following:

- select only a minimum set of subpackages for the update installation, but note that if you do this and some of the RISC/os 4.10 binaries are left on the system, they may not produce the desired results on RISC/os 4.50.

- backup all important *usr* filesystem files and do a scratch install

- backup all important *usr* filesystem files, make partition 3 the */usr* partition, and proceed with the update installation.

3.2.32 Miscellaneous New Commands and Changes

find
The **-follow** option was added to traverse symbolic links to directories. Normally **find** does not traverse symbolic links.

format
Previously, the **format** command did **not** write out the volume header by default; the user had to specifically write out the volume header. Now, by default, **format** writes the volume header, and if you do not want it written, you must specifically tell **format** not to write it.

ftp
If a user attempts to log in (via **ftp**) to a machine where he or she does not have a home directory, if the file *letc/login.nohome.ok* exists, the **ftp** is successful and the login directory is /.

kermit
RISC/os 4.50 provides the **kermit** command.

lock
The **-p** option was added to allow use of the user's standard password instead of requesting another one.

login
The "password required" and "root login on console only" features have been added to the RISC/os 4.50 **login**. Both features are optionally enabled, if suitable sentinel files exist. See the *login*(1-SysV) man page for details.

Also, **login** executes **/usr/ucb/quota** only if the file *letc/login.quotawarn.ok* exists and if `~/.hushlogin` does not exist.

mh
RISC/os 4.50 includes the **mh** command. For more information, see the *mh*(1) man page.

mount

There is a new **mount(1M)** program that knows how to deal with the **quota** subsystem. If quotas are not used, the old **mount** will work.

mount also has two new options: **nfs_sync** and **nfs_async**. See the **nfs_sync** and **nfs_async** options described in *fstab*(4).

printf(3S-BSD)

Added **%i, %n, %p, %U, %D,** and **%O** conversion specifications. All formatted output routines (**printf, fprintf, sprintf, vprintf, vfprintf,** and **vsprintf**) return EOF (-1) if an error occurred. *sprintf* returns the first argument; *vprintf, vfprintf,* and *vsprintf* return the number of characters printed.

rpc.mountd and /etc/exportfs

The **rpc.mountd** is completely new. Prior to RISC/os 4.50, **rpc.mountd** would automatically pick up changes to the */etc/exports* file. With RISC/os 4.50, when the */etc/exports* file is modified, **/etc/exportfs** must be re-run, or the changes in */etc/exports* will not be seen until the system is rebooted (and **/etc/exportfs** is automatically run).

Note that we do not ship a system with the hide/nohide options in */etc/exports*.

sendmail

The **sendmail** program (and its *sendmail.cf* file) was updated to a newer version. Be sure to run **newaliases** (or **sendmail -bi**) to rebuild the alias database after updating your system to RISC/os 4.50 (these instructions are included in the **Post-Installation Instructions** section of the release notes).

tar

The format of a **tar** archive has been modified to comply with the Extended Tar Format defined by the IEEE 1003.1 standard (also known as POSIX 1003.1). The **tar** command has also been modified to archive/extract block and character special files and FIFOs. In order to remain upward-compatible with earlier versions of **tar**, these files are only archived/extracted if the **-S** option is specified.

vi/ex

In RISC/os 4.50, **vi** is now 8-bit clean. Also, */usr/lib/ex.rc* was added for a system-side setting of options, with a **-n** option to override reading this system file.

vn, rn

RISC/os 4.50 provides two new commands, **/usr/new/vn** and **/usr/new/rn**, for reading network news. This is part of the MIPS' unsupported software that we include as a service to our customer base. If you already have these packages installed on your system, it may not be necessary to install this package. If you do choose to install it, the installation could possibly break the configuration (i.e., any symbolic links) you may have previously set up. For more information on this command, see the *vn*(1) and *rn*(1) man pages.

vsar

The precision of some of the information displayed by **vsar** has been altered in RISC/os 4.50 to increase the clarity of information and to reduce displaying non-significant numbers. In addition, the units used to display disk and tape I/O performance were made uniform and rational for all the fields where it was displayed. Please see the *vsar*(1) man page for full details.

3.3 Problems Resolved in RISC/os 4.50

The following sections describe various problems resolved in RISC/os 4.50 that may not have been entered as formal bugs. For a list of numbered bug fixes, see Appendix C.

System Calls

The system calls *truncate* and *ftruncate* are generally used to truncate a file to the requested length. In RISC/os 4.50, if the requested length is larger than the file size, the file will be grown to the requested length.

NFS

Because RISC/os 4.50 contains NFS 4.0, all of the bug fixes that went into NFS 4.0 are now contained in RISC/os.

crash

The following changes were made to the **crash** command for RISC/os 4.50:

- Added *vfs* as an alias for the **mount** command. Modify the **mount/vfs** command to list RISC/os 4.50 vfs structures.

- Modified the **fs** command to list the *vfssw* array.

- Modified the **inode** command to list **ufs** inodes, instead of system inodes.

- Added **snode**, **vnode**, and **rnode** commands, to list **specfs** snodes, system vnodes, and NFS rnodes.

- Deleted useless **adv**, **sndd**, and **rcvd** commands.

- Deleted **lck** command, because it is currently not implementable. (Locks are not implemented in a system daemon.)

- Modified the **stream** and **queue** commands to search the specfs vnodes for references to stream and queue blocks. (These commands were previously non-functional.)

- Modified all commands to reflect current structure contents.

A number of commands accept only object addresses, instead of either slot numbers or object addresses, since many of the "tables" are in fact not arrays. Notable among these commands are **mount** (vfs), **snode**, **vnode**, **rnode**, **stream**, and **queue**. Also, listings which refer to the relevant objects now just print object addresses (in hexadecimal), instead of slot numbers.

One known deficiency of long standing, is that **crash** does not do a true **dbx**-style stack trace, using symbol table information.

fsck.ffs

The -p option is now allowed. New options are **-a**, **-r** and **-C**. See the *fsck.ffs*(1) man page for more details.

mountall

Now works in parallel. See *mountall*(1M).

/etc/init.d/nfs

nfs "stop" now works.

learn

A bug that caused the escape character to be echoed too many times on output when in line discipline 0 has been fixed. This bug confused some shell scripts and broke the **learn** program.

login

tty groups now work properly in BSD and System V environments, and disk quotas are

recognized.

/etc/init.d/netdaemons

routed is now started by default; **timed** starts up only when a configuration file is present.

named

As shipped, **named**, the Internet domain name server, does not start up at system boot-time. Within the directory /etc/named.d, sample files can be found and used along with the named(8) man page to configure your system. In general, /etc/named.boot is the configuration file and must be created before named will operate properly.

Sticky Bits

When the sticky bit on a directory is set, in order to delete a file in that directory, the parent directory must be writable and at least one of the following must be true:

- the user owns the file

- the user owns the parent directory

- the file itself is writable by the user

- the user has superuser privileges

uugetty

The default setting for the **uugetty** SANE option has been changed from 7 bits and even parity to 8 bits with no parity. The default setting now matches that for **getty**. This setting can be changed by editing the /etc/gettydefs file.

3.3.1 Fixed in RISC/os 4.01 and 4.02

Included in RISC/os 4.50 are the following items which were fixed in the RISC/os 4.01 and 4.02 patches, which some people may not have received.

Virtual Memory and Swapping

The RISC/os 4.02 VM system corrected hangs and panics that existed in RISC/os 4.00 and 4.01. Changes were made to allow stack pages to be swapped which frees up more physical memory.

SCSI Driver

End of media handling on tape devices was improved. ENOSPACE will be returned on the early warning point. Ten additional commands will be allowed with no error returned (assuming physical EOT is reached first) to allow the program to "clean up" and then ENOSPACE will be returned for all commands except rewind.

Symbolic Links

A bad error return in the resolution of symbolic links was eliminated.

X-off/X-on Flow Control Causes Excessive System Load

In certain situations, if X-off/X-on (Ctrl-S/Ctrl-Q) flow control was used on terminal lines, a very high system load would occur due to leaving transmitter interrupts enabled. This was fixed in the kernels and the components for binary kernel reconfigurations.

Ctrl-C During Console Output Causes Kernel Panic

If CTRL-C (Interrupt) was sent during console output, it could cause a kernel panic. This was fixed in the kernels and the kernel components.

newfs.ffs Sets Rotational Delay Too High

The default case in newfs.ffs would set the rotational delay to 7 ms., which is much higher than necessary. This leads to degraded performance in disk-bound applications. This was

48

fixed by setting the delay to 0 by default.

etc/magic Has Incorrect Value for Compressed Files

The value of the magic number was incorrect, leading to these files being identified as "data" by file.

Library Routines fcvt() Give Too Long Results for 0.0

The library routines in the fcvt() family handled 0.0 incorrectly and put out too many 0's. This was fixed in /usr/lib/libc.a.

O_SYNC Mode Did Not Complete File Flush

If the O_SYNC flag was specified, which specifies that the writes to the file are to be flushed to the disk, the write operations could return before the write was completely flushed. This was fixed in the kernels and their components.

Connecting a Socket to Itself Causes Infinite Loop

If a socket connected to itself the kernel would loop.

BSD 4.3 dup2() System Call Causes System Crash If ofd == nfd

The dup2() system call would cause the system to fail if the two file descriptors were the same.

BSD 4.3 ioctl() Call Returns Incorrect Values

The TIOCGETP call always returned XTABS and CR3. This caused certain programs to reset the terminal to XTABS and CR3 mode, leading to sluggish behavior.

Programs Using Floating Point in Newly Forked Children May Encounter Errors

If a program which had done floating point calculation, forked a child process, and did another floating point computation in the child without an exec(), there could be interference between the two processes FP operations. This was found in portations benchmark suite.

Receipt of a Signal with an Invalid SP Causes System Crash

If a user program had an invalid Stack Pointer, and received a signal, the system would panic.

PC-NFS Reports 0 Free Space

Due to an arguably incorrect implementation of the NFS protocols in PC-NFS, the PC-NFS software incorrectly reported the free space to be 0 on NFS volumes.

syslogd Exits

The system logging daemon, syslogd, would exit during system startup, or if it received a message from an "old style" (named pipe) program, e.g. a program created under RISC/os 3.10. This was fixed. This problem resulted in system logs not being updated. With this fix, these logs, defined by /etc/syslog.conf will now be updated. You should ensure that your nightly cleanup scripts correctly rotate the log files to avoid filling up the disk.

Programs Using syslog Could Crash If syslogd Not Running

The library call syslog() had an error which could generate a core dump if a syslog() call was issued and syslogd was not running. This library change resulted in many of the system service daemons being re-linked and included in this release.

bsd43/bin/diff Contained Incorrect Pathnames

As noted in the RISC/os 4.00 Release Notes, the internal values of the pathnames for programs invoked by /bsd43/bin/diff were incorrect. This fix allows the "/n/barney/..." symbolic links described in the Release Notes to be deleted.

49

bsd43/bin/tip Had Incorrect Permissions and Symbolic Link
This program may also be invoked now as **/bsd43/bin/cu**.

csh Core Dumps
Several instances of core dumps by **csh** were fixed. These occurred when:

- 8 Ctrl-U's were followed by another character while in linedit mode

- an invalid command name was given inside backquotes. e.g. echo 'bogus'

In addition, the **csh** could hang if you entered Ctrl-Z during its processing of your *.login*.

addbib Required Two CR
Two CRs were required for every command input.

libbsd opendir Did Not Check for Directory Correctly
The opendir() library call contained an incomplete check on whether the file was a directory.

closedir Library Routine May Close Wrong File
closedir() could wind up closing the wrong file, typically stdin.

mkdirp() Leaves errno Set to ENOENT
The mkdirp() library routine was leaving errno set to ENOENT when it was returning successfully.

Televideo 970 Terminfo File Missing
The terminal information files for the Televideo 970 terminal were not present in the release.

rexecd Could Hang if Interrupted by a Signal
The **rexecd** daemon could hang if it was interrupted by a signal while waiting for input.

System Administrator Commands Permissions
Several system maintenance commands had permissions which allowed any user to invoke them. These have generally been changed to 550 mode. The affected commands include: /etc/link, /etc/unlink, helpadm and killall.

Obsolete Version of telnet
An obsolete version of **telnet** was included in RISC/os 4.0. This led to incorrect mapping of the LF character on output. A typical manifestation would be when using vi on an empty file, the ⁻ characters appeared on the right hand side of the screen.

sa2 Script Attempted to Delete Directory
The **sa2** accounting script attempted to delete the directory */usr/adm/sa*.

3.4 Known Problems and Limitations

The following is a list of known problems in RISC/os 4.50. A numbered summary of all open bugs can be obtained by contacting a MIPS Customer Service representative.

3.4.1 Booting RS2030 and Magnum 3000 Workstations

If you are using an RS2030 or Rx3230 system with a graphics head, the **console** variable in NVRAM must be set to 1 (the letter 'el') in order to boot the kernel.

50

3.4.2 Two Disk Installation on Rx3230

The two disk installation that can be done on an Rx3230 (and **only** on an Rx3230) only works as a scratch install in RISC/os R. An update install to two disks does not work. This problem will be fixed in the next release of RISC/os.

3.4.3 sash Hangs

Sometimes **sash** hangs on Rx3230 systems. A reset or a power cycle is required to clear this problem.

3.4.4 Floppy Formatting

Cross-density floppy formatting for the Rx2030 and Rx3230 systems is not supported.

3.4.5 SCSI Devices With Variable Length Records

If a user program reads from a variable length record device, such as the 1/2" or Exabyte tape drives, into a buffer that is not aligned on a 512 byte boundary **and** crosses a page boundary, data corruption can occur for other units which are on the same SCSI controller as the tape (i.e., corruption occurs in the controller, not system memory). To avoid this problem, use *malloc* to get the buffer. *malloc* tends to give nicely aligned buffers, especially if the buffer is allocated early in the program, so there is a smaller chance of corruption occurring.

3.4.6 NFS Authentication Errors

RISC/os 4.50 allows up to 16 groups to be sent over NFS as an authentication object, whereas previous releases allowed a maximum of only 8. That is, if a user id appears in more than 8 groups in the */etc/group* file, there is a problem when using NFS for a RISC/os 4.50 client talking to a server running an earlier RISC/os release. The symptom is the error message:

```
NFS getattr failed for server <x>: RPC: Authentication error
```

To work around this problem, modify the client's */etc/group* file so that no user is a member of more than 8 groups if a RISC/os 4.50 client is talking to a server running a release earlier than RISC/os 4.50.

3.4.7 Network Initialization When BIND (DNS) Used for Host Names

RISC/os 4.50 provides a flexible scheme for various system databases, the Virtual Information Service (VIS). This scheme is described in the *vis*(4) manual page. It is possible for you to specify that the host database is found via the Domain Name Service exclusively. This can be done on a running system without problems; however, it will not be possible to reboot the system because **ifconfig** cannot recognize the local host name. The work around is to always have a local */etc/hosts* file containing at least your system's name, and to specify in */etc/vis.conf*:

host: dns files

That is, try DNS first, and if it is not there, look to the local files. (The **ifconfig** failure occurs if this line says: "host: dns").

3.4.8 NFS

Certain operations will fail when a RISC/os 4.50 client is used with a server machine running something older than RISC/os 4.50. Any **touch** and **chown** commands done on the older system may fail with a message something like `cannot change times on foobar`.

3.4.9 NFS Locking

NFS file locking does not work in conjunction with local file locking. For example, reading mail locally works fine, but reading mail through NFS is not a good idea because the NFS file locking will not work properly with the local file locking.

3.4.10 M/2000-SCSI and RC3260 SCSI Disk Delays on Booting

When a SCSI M/2000 or RC3260 system with more than two disks is used with bootmode "d" (which means that the power-on diagnostics are skipped and memory is not cleared) and is booted without waiting 60 seconds after powering on the system, not all filesystems get mounted. Because the power-on diagnostics are skipped with bootmode set to "d", the disk drives do not have time to come up to full rotational speed before the system starts booting and the filesystems do not get mounted.

To work around this problem, either wait 60 seconds after powering on the system before giving the **auto** command to boot, or set the bootmode to "c" for an autoboot. Bootmode "c" runs the power-on diagnostics, giving the disks time spin up before the system begins booting.

If the system boots without mounting all the drives, you can type **mount -a** to mount all file systems in listed in */etc/fstab*.

3.4.11 Networks With Gateway Machines

On networks with gateway machines, it is a good idea to add a delay after the routing daemon (either **gated** or **routed**) is started in */etc/init.d/netdaemons* so that the daemons that are started after that have a better chance of working when another system on the network is down. To add the delay, add the lines in bold text to the */etc/init.d/netdaemons* file:

```
                                           fi
                                           echo " routed\c"
        fi
        echo " waiting...\c"
        sleep 30
        echo " done\c"

        # snmpd
```

3.4.12 SCSI Command Timeout

On systems with certain intermittent hardware problems, you may see some SCSI command timeout messages which then cause a proess to hang and be impossible to kill. This situation was very rare, and a fix was discovered too late to be included in RISC/os 4.50. If you encounter this problem, you can request a fixed version of the *common_scsi.c* file from MIPS Customer Service and then binary reconfigure your kernel.

3.4.13 csh

If you are logged in on the console port, using csh. have lineedit set, and a background job which has output completes fairly quickly, you may see part of the [1] Done ... message printed out twice.

3.4.14 rpc.rstatd Error Messages

In RISC/os 4.50, there is a line in /etc/inetd.conf to run rpc.rstatd, which is not included in the release. These messages can be safely ignored, or the line from /etc/inetd.conf can be removed. This will be resolved in a future release.

3.4.15 rexd

The Sun remote execution daemon, /usr/etc/rpc.rexd, cannot be invoked by inetd. Sites which wish to use rexd should add its invocation to their local system startup scripts.

3.4.16 Timezone

Please note that SystemV programs may display the wrong time/date when choosing timezones outside of the U.S. because of the algorithm used for calculating how far from GMT a particular timezone lies. 4.3 BSD programs use the information in /etc/zoneinfo, which contains data files for the different timezones around the world. The value that the environment variable TZ takes on will determine which file is looked at in this directory.

The SystemV library uses only the name in determining how far off of GMT a current timezone lies. Therefore, a 4.3 BSD program can correctly operate with TZ=Japan because /etc/timezone/Japan performs the appropriate mapping; but a System V program needs to have TZ=JST-9JDT. This cryptic abbreviation means that Japan Standard Time is 9 hours ahead of Greenwich Mean Time. If the file /etc/zoneinfo/JST-9JDT contains the appropriate mapping information (setup with zic(8)), both 4.3 BSD and System V can operate properly when JST-9JDT is placed in /etc/TZ.

Since the name "Japan" is more descriptive than JST-9JDT, we have left it in /etc/zoneinfo in addition to JST-9JDT. Other countries should set up similar timezone files by following the format used in /etc/zoneinfo/SOURCES/asia for Japan.

3.4.17 format For Non-Rx2030 Systems

A problem exists in format that sometimes makes it impossible to read the manufacturer's defect list from SMD drives. This will occur only on non-formatted drives that are purchased from somewhere other than MIPS. If you are experiencing problems, please contact MIPS Customer Service.

3.4.18 sadp

The sadp program never worked properly in previous releases, so it is omitted in RISC/os 4.50.

3.4.19 fsck on Miniroot

If a dirty filesystem is encountered when trying to mount it while running on the miniroot, an error message advises to use fsck or mount -c to clean the filesystem. These will not work from the miniroot, and /etc/fsck.ffs must explicitly be used instead.

3.4.20 Nroff and Troff Macros

If you have customized any of the **n/troff** macros, you may want to first make a copy of your macro files so that your copy is not destroyed by the installation. These files are not included on the preserve list.

4. Installation Notes

This section illustrates the installation of the RISC/os 4.50 binary release onto a MIPS RISComputer system. **Please note that the installation instructions for RISC/os 4.50 are slightly different from what you may be used to. Please read and follow all of the instructions carefully.** The instructions for updating to RISC/os 4.50 are given first, and a later section describes the procedure for installing RISC/os 4.50 from scratch, in the event that a complete reinstallation is required.

In the procedure examples given in this section, different fonts are used to show output from the system and user responses. System output is shown in the `typewriter` font; user responses are shown in the **bold** font; comments about the procedure are shown in *Italics*. All user responses should be typed as shown and entered with a return.

Prompts from the installation scripts are always in the form:

```
prompt (choice1 choice2 ...) [default]?
```

or, simply:

```
prompt [default]?
```

where "(choice1 choice2 ...)", if present gives the range of legal responses, and "[default]" gives the default choice; pressing return by itself will always select the default choice.

4.1 Installing RISC/os 4.50 as an Update

This section describes the procedure for performing an update install. If you have received this release of RISC/os as a software update, you will need to perform this procedure to install it on your system running an earlier release. Please read the **Known Problems and Limitations** before attempting the installation.

A sample update install on an M/120 system is shown below. The procedure is the same for all other systems, except that the subpackage "m500" would be installed on an M/500, and so forth. Some of particulars of the messages shown below will also vary slightly from machine to machine, but the procedure will be essentially the same as shown for all system types. **It is important to note that the RISC/os 4.50 installation instructions are slightly different from what what you are used to. Please follow the procedure shown below carefully:**

Installation of the RISC/os 4.50 *usr* filesystem requires approximately 35 megabytes of free space. To save yourself some time, it is best to ensure this space is available before beginning the update procedure.

Before beginning the update procedure, there are a few cautionary items to note:

- The update procedure requires that all of the system's filesystems be "clean" (i.e., unmounted when the system is brought down before loading the miniroot). If for some reason they are not, the install will note that they cannot be mounted, and halt. If this happens, use */etc/fsck.ffs* from the miniroot to clean the filesystem(s) before redoing the install.

- Any directories in the *root* or *usr* filesystems that may have been linked to a separate filesystem in order not to fill up the *root* or *usr* filesystems should be added to the preserve list before giving the **inst** command. If any of these special directories are not added to the preserve lists, the links (and thus the files in the directories) will be lost during installation. Take a good look at the *$Pkg/lib/root.preserves* and *usr.preserves* files, and add any appropriate entries (mount */usr* on */mnt* if you prefer an editor other than **ed**).

55

- If any directories (MIPS' filesystems, such as /usr, /usr/man, or any other directories where MIPS software is installed) are links to filesystems across the net through NFS, the installation procedure will fail with a message similar to this:

```
space: error: fstabind(): couldn't find device file for dev = 0x1436
space: fatal: pathname: ./usr/man/catman/u_man/man1/pc.1
```

To work around this problem, delete the link to the remote filesystem, create the directory on the target system and proceed with the installation. Once the installation is complete, the filesystem (or directory) can be removed and the link recreated.

For example, if /usr/man is a link, type:

```
# ls -l /usr/man
l---------  1 bin      bin              15 May 13 15:33 /usr/man@ -> /n/system2/usr/man
# mv /usr/man /usr/man.temp
# mkdir /usr/man
```

Install the package following the installation instructions. Then type:

```
# rm -rf /usr/man
# mv /usr/man.temp /usr/man
```

and the manual pages will not be installed.

- The /etc/fstab file must be in correct format. In particular, no blank lines are allowed. Use "#" for a blank comment line.

- **For Rx2030 Installations:**
  Please note that the default partition for /usr on the Rx2030 systems is now partition 3, in order to allow RISCwindows to fit on a 172 megabyte disk drive. If you are doing an update install and your usr partition is not 3, be sure to do one of the following:

  - select only a minimum set of subpackages for the update installation, but note that if you do this and some of the RISC/os 4.10 binaries are left on the system, they may not produce the desired results on RISC/os 4.50.

  - backup all important usr filesystem files and do a scratch install

  - backup all important usr filesystem files, make partition 3 the /usr partition, and proceed with the update installation.

- And finally, if it looks like anything at all went wrong during the installation procedure, the upgrade should be redone to ensure no bizarre behavior results.

To begin the update installation, you must first shutdown the system. To shut down a MIPS RISComputer running RISC/os or UMIPS, perform the following steps from the console device:

```
login: root
Passwd:

# cd /
# /etc/telinit 0
```

Before pressing reset or turning off the power, the system administrator should wait for the console message:

```
The system is down.
```

56

Under normal circumstances the system will then drop into the monitor and the console will display the monitor prompt:

```
MIPS Monitor Version 4.00 MIPS OPT Sat May 7 13:12:36 PDT 1988 opsys
Memory size: 8388608 (0x800000) bytes
Icache size: 16384 (0x4000) bytes
Dcache size: 8192 (0x2000) bytes
>>
```

The terminal session below shows the installation and booting of the miniroot filesystem. Each series of a machine requires slightly different commands. In the script below, we include all versions of the command with a comment in parenthesis identifying which command to use for which system type. Put tape 1 into the tape drive and proceed as follows:

```
>>  boot -f tqsd(,6,3)sash.2030    (for Rx2030)
>>  boot -f tqsd(,6,2)sash.std  (for Rx3230)
>>  boot -f tqis(,,2)sash.std    (for M/120 or RC3240)
>>  boot -f tpqic(,,2)sash.std  (for M/500, 800, 1000)
>>  boot -f tqij(0,6,2)sash.std (for M/2000, RC3260, RC6280)
Interphase 4210 Jaguar controller @  (paddr=bd009000)
Jaguar Version (077-30-XAL) Date 03221990 with 126 Kbytes ram.
Work Queue 7 for device 'ARCHIVE -VIPER 125  21006--010'
206000+57168+226912 entry: 0xa0300000
Standalone Shell: Version 5.0 OPT Thu May 24 01:40:20 PDT 1990 root
sash:  cp -b 16k tqsd(,6,4) dksd(,,1)     (for Rx2030, Rx3230)
sash:  cp -b 16k tqis(,,4) dkis(,,1)      (for M/120 or RC3240)
sash:  cp -b 16k tpqic(,,4) dkip(,,1)     (for M/500, 800, 1000)
sash:  cp -b 16k tqij(0,6,4) dkij(,,1)    (for RC3260 or M/2000 SCSI)
sash:  cp -b 16k tqij(0,6,4) dkip(,,1)    (for M/2000 SMD, RC6280)
.............................................................................
.............................................................................
.............................................................................
.............................................................................
.............................................................................
.............................................................................
.............................................................................
13824000 (0xd2f000) bytes copied
```

enter one of the following commands, depending on your system type:

```
sash:  boot -f tqsd(0,6,8)unix.i2000_std root=sdc0d0s1 console=g  (for RS2030)
sash:  boot -f tqsd(0,6,8)unix.i2000_std root=sdc0d0s1 console=0  (for RC2030 tty 0)
sash:  boot -f tqsd(0,6,8)unix.i2000_std root=sdc0d0s1 console=1   (for RC2030 tty1)
sash:  boot -f tqsd(0,6,11)unix.r3030_std root=sdc0d0s1 (for Rx3230)
sash·  boot -f tqis(,,6)unix.r2400_std root=isc0d0s1        (for M/120, RC3240)
sash:  boot -f tpqic(,,5)unix.r2300_std root=ipc0d0s1       (for M/500, 800, 1000)
sash:  boot -f tqij(0,6,9)unix.r3200_ijc root=ijc0d0s1      (for RC3260 or M/2000 SCSI)
sash:  boot -f tqij(0,6,7)unix.r3200_std root=ipc0d0s1      (for M/2000 SMD)
sash:  boot -f tqij(0,6,10)unix.r6000_std root=ipc0d0s1     (for RC6280)
```

```
834416+108720+990960 entry: 0x80021000
CPU: MIPS R3000 Processor Chip Revision: 2.0
FPU: MIPS R3010 VLSI Floating Point Chip Revision: 2.0
```

```
RISC/os Release 4_50 mips Version UMIPS
Total real memory  = 33554432
Available memory   = 30924800
Root on dev 0x401, Swap on dev 0x401
Root fstype ffs
New swplo: 26992  swap size: 6376K bytes
Available memory   = 29315072


Miniroot run level 1

Making miniroot device files for m2000-6 system...
erase=^H, kill=^U, interrupt=^C
#
```

At this point, the miniroot file system has been copied to the swap partition and is running the Bourne shell, **sh**. To begin the update, proceed as follows from the miniroot prompt:

set -a
Install=update
inst
Software package installation

Installation Information:

This is an update install.
Packages will be read in from the local Q24 tape drive.
Machine type: m120-5

Is the information above correct? (y n) [y]? y
Note: It is not necessary to install a kernel on the miniroot
      The process will take a few minutes.
Would you like to install the kernel to the miniroot (y n) [n]? **n**

========== checking subpackages ==========

The following subpackages may be installed:

```
root       -- RISC/os Standard Root Filesystem
m120-5     -- RISC/os m/120 Kernel and Devices
usr        -- RISC/os Standard /usr Filesystem
cmplrs     -- MIPS-C Compiler
cmplrs-bsd43      -- MIPS-C 4.3 BSD Include Files and Libraries
man        -- RISC/os Manual Pages
compat     -- RISC/os 4.00 Compatibility Libraries
bsd43      -- RISC/os ^.3 BSD Utilities, Include Files and Libraries
reconfig           -- Kernel Binary Reconfiguration Components
emacs      -- emacs
posix      -- RISC/os POSIX P1003.1 Include Files, Commands and Libraries
uucp       -- UUCP
sccs       -- SCCS
news_readers      -- News Readers
games      -- Games
```

58

```
mh         -- mh

========== selecting subpackages ==========

You may select all of the above subpackages by answering "y" to the
following question.  If you answer "n" then you will be asked to
select the optional subpackages you would like to have installed.

Install ALL subpackages (y n) [n]? n

When asked if you want to install a subpackage, please answer with
one of the following:

   y - Yes, you want to install the subpackage
   n - No, you do NOT want to install the subpackage
   l - List the contents of the subpackage and ask me again

Subpackage root will be installed.
Subpackage m120-5 will be installed.
Subpackage usr will be installed.
Subpackage cmplrs will be installed.
Install subpackage cmplrs-bsd43 (l y n) [n]? y
Install subpackage man (l y n) [n]? y
Install subpackage compat (l y n) [n]? y
Install subpackage bsd43 (l y n) [n]? y
Subpackage reconfig will be installed.
Install subpackage emacs (l y n) [n]? y
Install subpackage posix (l y n) [n]? y
Install subpackage uucp (l y n) [n]? y
Install subpackage sccs (l y n) [n]? y
Install subpackage news_readers (l y n) [n]? y
Install subpackage games (l y n) [n]? y
Install subpackage mh (l y n) [n]? y

Selected subpackages:
    root m120-5 usr cmplrs cmplrs-bsd43 man compat bsd43 reconfig emacs posix uucp sccs
Is this what you want (y n) [y]? y

========== setting system clock/calendar ==========

The current value of the clock is: Wed Jun 6 18:08:32 PST 1990
Is the clock correct (y n) [y]? y
========== verifying single-user mode ==========

The system is in a single-user run level.
Do you want to install the volume header (please do so unless you really
understand the consequences) (y n) [y]? y
========== installing sash to volume header ==========

========== mounting filesystems ==========

/dev/root mounted on /mnt
/dev/usr mounted on /mnt/usr
```

```
Partition                  Megs     Mounted File System or Partition Usage
--------------------------------------------------------------------------
0   /dev/dsk/isc0d0s0      22       /dev/root
1   /dev/dsk/isc0d0s1      20       /
6   /dev/dsk/isc0d0s6      250      /dev/usr
7   /dev/dsk/isc0d0s7      22       -**** Available Partition ****-
--------------------------------------------------------------------------
Disk Device     /dev/dsk/isc0d0s2        312      Megabytes Total Size
--------------------------------------------------------------------------
```

Do you wish to change swap partition configuration (y n) [n]? **y**

*In most cases, you will probably not want to change the swap partition
configuration here, but this example is included for completeness.*

========== configuring swap space ==========

System has 32 megabytes of main memory

The following is a listing of available disk partitions and sizes.
NOTE: the miniroot installation system is mounted as / on /dev/swap

```
--------------------------------------------------------------------------
Partition                  Megs     Mounted File System or Partition Usage
--------------------------------------------------------------------------
1   /dev/dsk/isc0d0s1      20       /
7   /dev/dsk/isc0d0s7      22       -**** Available Partition ****-
--------------------------------------------------------------------------
Disk Device     /dev/dsk/isc0d0s2        312      Megabytes Total Size
--------------------------------------------------------------------------
```
(Press return to continue...)
It IS recommended that you select an additional swap partition
Here are partitions you may select from for additional swap space
```
--------------------------------------------------------------------------
Partition                  Megs     Mounted File System or Partition Usage
--------------------------------------------------------------------------
7   /dev/dsk/isc0d0s7      22       -**** Available Partition ****-
--------------------------------------------------------------------------
Disk Device     /dev/dsk/isc0d0s2        312      Megabytes Total Size
--------------------------------------------------------------------------
```
Select from the above displayed available partitions as in
```
--------------------------------------------------------------------------
                                     -**** Available Partition ****-
--------------------------------------------------------------------------
```
If no -**** Available Partition ****- partitions are displayed
then you will need to add an ADDITIONAL DISK if you wish to provide
more swap space. Note that the size of the partition is in megabytes.
It is recommended that the system be configured with 2 or 3 times the
swap space as there is system memory. For example a 16 megabyte memory
system should have 32 megabytes of swap disk space available.
Do you wish to add any swap partitions (y n) [n]? **y**
(Press return to continue...)
Which partition should be added for swap [# or q to Quit Adding] [q]? **7**
Current alternate swap
/dev/dsk/isc0d0s7 none swap rw,noauto 0 0

60

Which partition should be added for swap [# or q to Quit Adding] [q]? **q**
Add additional swap partitions to system /etc/fstab (y n) [n]? **y**
========== preserving local files ==========

Running preserve -s for subpackage root... 54 files preserved.
No preserve list or findmods list for m120-5- preserve not executed.
Running preserve -s for subpackage usr... 25 files preserved.
No preserve list or findmods list for cmplrs- preserve not executed.
No preserve list or findmods list for cmplrs-bsd43- preserve not executed.
No preserve list or findmods list for man- preserve not executed.
No preserve list or findmods list for compat- preserve not executed.
No preserve list or findmods list for bsd43- preserve not executed.
No preserve list or findmods list for reconfig- preserve not executed.
No preserve list or findmods list for emacs- preserve not executed.
No preserve list or findmods list for posix- preserve not executed.
No preserve list or findmods list for uucp- preserve not executed.
No preserve list or findmods list for sccs- preserve not executed.
No preserve list or findmods list for news_readers- preserve not executed.
No preserve list or findmods list for games- preserve not executed.
No preserve list or findmods list for mh- preserve not executed.

========== verifying disk space ==========

Do you want to check for space (please do so unless you really
understand the consequences) (y n) [y]? **y**
The system will now be checked to verify that there is enough disk
space with the current configuration to successfully install the package
(and any selected optional subpackages). For large packages (especially
operating system packages), this can be time consuming...

You will see one of the following responses from the system:

device	bfree	ifree	breq	ireq	bcred	icred	
/dev/root	5032	8827	13081	396	11344		348
/dev/usr	19607	102942	87942	4744	72948		4574

WARNING! This package will fit on the disk,
but it will cause more than 90% of the disk to be used.
This may cause problems for non-root users.
It is recommended that you abort the installation now.

Abort the installation? (y n) [y]? **n**

or:

There is enough space.

========== stripping old links ==========

Stripping links for subpackage root...
Stripping links for subpackage m120-5...
Stripping links for subpackage usr...
Stripping links for subpackage cmplrs...
Stripping links for subpackage cmplrs-bsd43...

61

```
Stripping links for subpackage man...
Stripping links for subpackage compat...
Stripping links for subpackage bsd43...
Stripping links for subpackage reconfig...
Stripping links for subpackage emacs...
Stripping links for subpackage posix...
Stripping links for subpackage uucp...
Stripping links for subpackage sccs...
Stripping links for subpackage news_readers...
Stripping links for subpackage games...
Stripping links for subpackage mh...
========== extracting files from subpackage archives ==========

Verifying tape id... ok
Forward spacing the tape...

Loading subpackage: root...
Forward spacing the tape...
Forward spacing the tape...
Loading subpackage: m120-5...
Forward spacing the tape...
rewinding the tape...

Please mount umips tape number 2 and press return:
rewinding the tape...
Verifying tape id... ok
Forward spacing the tape...

Loading subpackage: usr...
Forward spacing the tape...
Loading subpackage: cmplrs...
Forward spacing the tape...
Loading subpackage: cmplrs-bsd43...
Forward spacing the tape...
Loading subpackage: man...
Forward spacing the tape...
rewinding the tape...

Please mount umips tape number 3 and press return:
rewinding the tape...
Verifying tape id... ok
Forward spacing the tape...


Loading subpackage: compat...
Forward spacing the tape...
Loading subpackage: bsd43...
Forward spacing the tape...
Loading subpackage: reconfig...
Forward spacing the tape...
Loading subpackage: emacs...
Forward spacing the tape...
rewinding the tape...

Please mount umips tape number 4 and press return:
```

```
rewinding the tape...
Verifying tape id... ok
Forward spacing the tape...

Loading subpackage: posix...
Forward spacing the tape...
Loading subpackage: uucp...
Forward spacing the tape...
Loading subpackage: sccs...
Forward spacing the tape...
Loading subpackage: news_readers...
Forward spacing the tape...
Loading subpackage: games...
Forward spacing the tape...
Loading subpackage: mh...
Forward spacing the tape...
rewinding the tape...
========== making device special files ==========

========== running comply ==========

running first comply pass...
running second comply pass...
There are comply messages from the second pass in:
   /umips4.50/lib/comply.out
(A copy will be placed on the installed system as:)
   /usr/pkg/lib/umips4.50/lib/comply.out
```

Normally, there should be no second-pass comply messages. The presence of these messages may indicate problems with the installation.

View the messages now (y n) [y]? **y**
You may see the following harmless message on M/2000, RC3260, and RC6280 installations:

```
comply: error: link count dev/scsi/ijc0d0s8 3 should be: 2

(Press return to continue...)
```

```
========== cleaning up old versions ==========
```

An attempt will now be made to clean up any files left over from previous versions of the software which has just been installed.

It's probably safe to remove old versions of standard OS packages.
If you want the 2.0 version of the compiler left on the system in addition
to or instead of to using the RISCos4.0 compatibility package, you won't
want to remove the compiler package.

```
Searching for old versions to remove...

========== restoring preserved user files ==========

Running preserve -r for subpackage root...
No preserve list or findmods list for m120-5- no files restored.
```

```
Running preserve -r for subpackage usr...
No preserve list or findmods list for cmplrs- no files restored.
No preserve list or findmods list for cmplrs-bsd43- no files restored.
No preserve list or findmods list for man- no files restored.
No preserve list or findmods list for compat- no files restored.
No preserve list or findmods list for bsd43- no files restored.
No preserve list or findmods list for reconfig- no files restored.
No preserve list or findmods list for emacs- no files restored.
No preserve list or findmods list for posix- no files restored.
No preserve list or findmods list for uucp- no files restored.
No preserve list or findmods list for sccs- no files restored.
No preserve list or findmods list for news_readers- no files restored.
No preserve list or findmods list for games- no files restored.
No preserve list or findmods list for mh- no files restored.


========== running conversion scripts ==========

========== root.fstab    Thu May 24 15:29:55 PST 1990 ==========


No ips devices found in /etc/fstab.


To take advantage of the improved parallel fsck, /etc/fstab will now
be modified to allow the root partition to be fsck'd on the first pass.
All other local filesystems will be fsck'd on the second pass.

A copy of /etc/fstab will be saved as /etc/fstab.save.1.


No changes made.

Press return to continue:


========== cleaning up ==========

Copying packaging information directory to /mnt/usr/pkg/lib/umips4.50...
```

Various messages about Removing Duplicate File are seen here

```
Unmounting filesystems...
/mnt/usr: Unmounted
/mnt: Unmounted
========== installation complete ==========

#
```

Please proceed to the "Post-Installation System Modifications" section to make sure everything is correct before shutting down the miniroot and booting multi-user.

4.2 Installing RISC/os 4.50 From Scratch

This section describes how to install RISC/os onto disk for the first time, or what to do in the event of an unrecoverable system failure, when the disks must be reformatted and RISC/os re-installed from scratch. If anything about the installation appears to be wrong, it is best to redo the install, to ensure no strange behavior results.

A sample scratch install on an M/2000 system is illustrated below. The procedure is the same for all MIPS RISComputers, except that the appropriate subpackage is installed for each system instead of the subpackage "m2000". Some of particulars of the messages shown below also vary slightly from machine to machine, but the procedure is essentially the same as shown for all system types.

To install RISC/os 4.50, put tape 1 into the tape drive and proceed as follows:

```
>> boot -f tqsd(,6,3)sash.2030      (for Rx2030)
>> boot -f tqsd(,6,2)sash.std  (for Rx3230)
>> boot -f tqis(,,2)sash.std   (for M/120, RC3240)
>> boot -f tpqic(,,2)sash.std  (for M/500, 800, 1000)
>> boot -f tqij(0,6,2)sash.std  (for M/2000, RC3260, RC6280)


 Interphase 4210 Jaguar controller @ (paddr=bd009000)
Jaguar Version (077-30-XAF) Date 08211989 with 126 Kbytes ram.
Work Queue 7 for device 'ARCHIVE -VIPER 125  21006--010'
207632+57504+228704 entry: 0xa0300000
Standalone Shell: Version 5.0 OPT Wed Jun 6 14:48:44 PDT 1990 root
```

```
sash: cp -b 16k tqsd(,6,4) dksd(,,1)     (for Rx2030, Rx3230)
sash: cp -b 16k tqis(,,4) dkis(,,1)      (for M/120, RC3240)
sash: cp -b 16k tpqic(,,4) dkip(,,1)     (for M/500, 800, 1000)
sash: cp -b 16k tqij(0,6,4) dkij(,,1)    (for RC3260 or M/2000 SCSI)
sash: cp -b 16k tqij(0,6,4) dkip(,,1)    (for M/2000 SMD, RC6280)
Interphase 4210 Jaguar controller @  (paddr=bd009000)
Jaguar Version (077-30-XAL) Date 03221990 with 126 Kbytes ram.
 .............................................................................
 .............................................................................
 .............................................................................
 .............................................................................
 .............................................................................
 .............................................................................
 .............................................................................
 ..............................................
13824000 (0xd2f000) bytes copied
```

enter one of the following commands, depending on your system type:

```
sash: boot -f tqsd(0,6,8)unix.i2000_std root=sdc0d0s1 console=g  (for RS2030)
sash: boot -f tqsd(0,6,8)unix.i2000_std root=sdc0d0s1 console=0  (for RC2030 tty 0)
sash: boot -f tqsd(0,6,8)unix.i2000_std root=sdc0d0s1 console=1  (for RC2030 tty1)
sash: boot -f tqsd(0,6,11)unix.r3030_std root=sdc0d0s1 (for Rx3230)
sash: boot -f tqis(,,6)unix.r2400_std root=isc0d0s1     (for M/120, RC3240)
sash: boot -f tpqic(,,5)unix.r2300_std root=ipc0d0s1     (for M/500, 800, 1000)
sash: boot -f tqij(0,6,9)unix.r3200_ijc root=ijc0d0s1     (for RC3260 orM/2000 SCSI)
sash: boot -f tqij(0,6,7)unix.r3200_std root=ipc0d0s1     (for M/2000 SMD)
sash: boot -f tqij(0,6,10)unix.r6000_std root=ipc0d0s1 (for RC6280)
```

815424+102192+1286560 entry: 0x80021000

```
CPU  MIPS R3000 Processor Chip Revision: 2.0
FPU: MIPS R3010 VLSI Floating Point Chip Revision: 2.0

RISC/os Release 4_50 mips Version UMIPS
Total real memory  = 33554432
Available memory   = 30924800
Root on dev 0x401, Swap on dev 0x401
Root fstype ffs
New swplo: 26992  swap size: 6376K bytes
Available memory   = 29310976


Miniroot run level 1

Making miniroot device files for m2000-6 system...
erase=^H, kill=^U, interrupt=^C
```

At this point, the miniroot file system has been copied to the swap partition and is running the Bourne shell, **sh**. To begin the installation, proceed as follows from the miniroot prompt:

inst

```
Software package installation



Installation Information:

This is a SCRATCH install.  Data on the root  and /usr disks will be lost.
Packages will be read in from the local Q24 tape drive.
Machine type: m2000-6
Root disk type: SMD

Is the information above correct? (y n) [y]?
Note: It is not necessary to install a kernel on the miniroot
      The process will take a few minutes.
Would you like to install the kernel to the miniroot (y n) [n]? y
rewinding the tape... DKVJ 0:6 unit attention; media change or drive reset

Forward spacing the tape...
Reading the unix.r3200_std kernel from tape...
4285+0 records in
4285+0 records out
Done.



========== checking subpackages ==========

The following subpackages may be installed:

root     -- RISC/os Standard Root Filesystem
m2000-6 -- RISC/os m/2000 Kernel and Devices
usr      -- RISC/os Standard /usr Filesystem
cmplrs  -- MIPS-C Compiler
cmplrs-bsd43    -- MIPS-C 4.3 BSD Include Files and Libraries
```

```
man       -- RISC/os Manual Pages
compat    -- RISC/os 4.00 Compatibility Libraries
bsd43     -- RISC/os 4.3 BSD Utilities, Include Files and Libraries
reconfig          -- Kernel Binary Reconfiguration Components
emacs     -- emacs
posix     -- RISC/os POSIX P1003.1 Include Files, Commands and Libraries
uucp      -- UUCP
sccs      -- SCCS
news_readers     -- News Readers
games     -- Games
mh        -- mh


========== selecting subpackages ==========

You may select all of the above subpackages by answering "y" to the
following question.  If you answer "n" then you will be asked to
select the optional subpackages you would like to have installed.

Install ALL subpackages (y n) [n]?

When asked if you want to install a subpackage, please answer with
one of the following:

   y - Yes, you want to install the subpackage
   n - No, you do NOT want to install the subpackage
   l - List the contents of the subpackage and ask me again

Subpackage root will be installed.
Subpackage m2000-6 will be installed.
Subpackage usr will be installed.
Subpackage cmplrs will be installed.
Install subpackage cmplrs-bsd43 (l y n) [n]? y
Install subpackage man (l y n) [n]? y
Install subpackage compat (l y n) [n]? y
Install subpackage bsd43 (l y n) [n]? y
Subpackage reconfig will be installed.
Install subpackage emacs (l y n) [n]? y
Install subpackage posix (l y n) [n]? y
Install subpackage uucp (l y n) [n]? y
Install subpackage sccs (l y n) [n]? y
Install subpackage news_readers (l y n) [n]? y
Install subpackage games (l y n) [n]? y
Install subpackage mh (l y n) [n]? y

Selected subpackages:
   root m2000-6 usr cmplrs cmplrs-bsd43 man compat bsd43 reconfig emacs posix uucp sc
Is this what you want (y n) [y]? y

========== setting system clock/calendar ==========

The current value of the clock is: Wed Jun 6 18:08:32 PST 1990
Is the clock correct (y n) [y]?

========== verifying single-user mode ==========
```

The system is in a single-user run level.

Please answer "y" to the following question unless you really understand the consequences.

Do you want to install sash to the volume header (y n) [y]?

========== installing sash to volume header ==========

========== determining /usr partition ==========

```
---------------------------------------------------------------------
Partition              Megs     Mounted File System or Partition Usage
---------------------------------------------------------------------
3    /dev/dsk/ipc0d0s3   587     -**** Available Partition ****-
4    /dev/dsk/ipc0d0s4   375     -**** Available Partition ****-
5    /dev/dsk/ipc0d0s5   188     -**** Available Partition ****-
6    /dev/dsk/ipc0d0s6   562     -**** Available Partition ****-
---------------------------------------------------------------------
Disk Device    /dev/dsk/ipc0d0s2          626      Megabytes Total Size
---------------------------------------------------------------------
```

Possible partitions to use are marked by -**** Available Partition ****-
         select either partition 3, 4, 5 or 6
Which partition should /usr be installed on [6]?
/usr partition will be installed on partition 6

========== initializing filesystems ==========

A scratch install of an operating system package is being performed
from the miniroot. Normally in this case the filesystems are
initialized. When a filesystem is initialized, any existing data will
be lost. You will be given a chance to override initialization of
each individual filesystem below.

Initialize filesystems (y n) [y]?
Disk type for controller 0 drive 0 [2333-64]? **2372-64**

See the section titled **Disk Drive Types** *for a list of possible choices, if necessary*

Initialize filesystem on /dev/root (y n) [y]?
Initialize filesystem on /dev/usr (y n) [y]?

Initializing the filesystem on /dev/root...
/dev/root: 39744 sectors in 23 cylinders of 27 tracks, 64 sectors
        20.3Mb in 2 cyl groups (16 c/g, 14.16Mb/g, 2048 i/g)
super-block backups (for fsck -) at:
 32, 27744,
rotational delay between contiguous blocks changes from 7ms to 0ms

Checking the filesystem on /dev/root...
** /dev/root
** Last Mounted on

```
** Phase 1 - Check Blocks and Sizes
** Phase 2 - Check Pathnames
** Phase 3 - Check Connectivity
** Phase 4 - Check Reference Counts
** Phase 5 - Check Cyl groups
2 files, 9 used, 19294 free (14 frags, 2410 blocks, 0.1% fragmentation)


***** FILE SYSTEM WAS MODIFIED *****

Initializing the filesystem on /dev/usr...
mkfs.ffs: warning: resetting filesystem size to 1150832
Warning: 16 sector(s) in last cylinder unallocated
/dev/usr:       1150832 sectors in 666 cylinders of 27 tracks, 64 sectors
        589.2Mb in 42 cyl groups (16 c/g, 14.16Mb/g, 2048 i/g)
super-block backups (for fsck -) at:
 32, 27744, 55456, 83168, 110880, 138592, 166304, 194016, 221728, 249440,
 277152, 304864, 332576, 360288, 388000, 415712, 443424, 471136, 498848,
526560,
 554272, 5856, 775968,
803680,
 831392, 859104, 884768, 912480, 940192, 967904, 995616, 1023328, 1051040,
1078752,
 1106464, 1134176,
rotational delay between contiguous blocks changes from 7ms to 0ms

Checking the filesystem on /dev/usr...
** /dev/usr
** Last Mounted on
** Phase 1 - Check Blocks and Sizes
** Phase 2 - Check Pathnames
** Phase 3 - Check Connectivity
** Phase 4 - Check Reference Counts
** Phase 5 - Check Cyl groups
2 files, 9 used, 563966 free (14 frags, 70494 blocks, 0.0% fragmentation)


***** FILE SYSTEM WAS MODIFIED *****

========== mounting filesystems ==========

/dev/root mounted on /mnt
/dev/usr mounted on /mnt/usr
```

```
-----------------------------------------------------------------------------
Partition                Megs    Mounted File System or Partition Usage
-----------------------------------------------------------------------------

0   /dev/dsk/ipc0d0s0    20      /dev/root
1   /dev/dsk/ipc0d0s1    20      /
6   /dev/dsk/ipc0d0s6    562     /dev/usr
7   /dev/dsk/ipc0d0s7    25       -**** Available Partition ****-
-----------------------------------------------------------------------------
Disk Device    /dev/dsk/ipc 626     Megabytes Total Size
-----------------------------------------------------------------------------


Do you wish to change swap partition configuration (y n) [n]?
```

```
========== verifying disk space ==========
```

Do you want to check for space (please do so unless you really
understand the consequences) (y n) [y]?
The system will now be checked to verify that there is enough disk
space with the current configuration to successfully install the package
(and any selected optional subpackages). For large packages (especially
operating system packages), this can be time consuming...

You will see either this:

```
        device    bfree     ifree      breq      ireq     bcred     icred
     /dev/root     7873      8858     15146       440      9221       320
      /dev/usr    61418    109649    108483      7264     50649      5788
```


```
    WARNING!  This package will fit on the disk,
    but it will cause more than 90% of the disk to be used.
    This may cause problems for non-root users.
    It is recommended that you abort the installation now.
```


Or this:

There is enough space.

```
========== extracting files from subpackage archives ==========
```

rewinding the tape... DKVJ 0:6 unit attention; media change or drive reset

Verifying tape id... ok
Forward spacing the tape...

Loading subpackage: root...
Forward spacing the tape...
Forward spacing the tape...
Loading subpackage: m2000-6...
Forward spacing the tape...
rewinding the tape...

Please mount umips tape number 2 and press return:
rewinding the tape...·
Verifying tape id... ok
Forward spacing the tape...

Loading subpackage: usr...
Forward spacing the tape...
Loading subpackage: cmplrs...
Forward spacing the tape...
Loading subpackage: cmplrs-bsd43...
Forward spacing the tape...
Loading subpackage: man...
Forward spacing the tape...
rewinding the tape...

```
Please mount umips tape number 3 and press return:
rewinding the tape...
Verifying tape id... ok
Forward spacing the tape...

Loading subpackage: compat...
Forward spacing the tape...
Loading subpackage: bsd43...
Forward spacing the tape...
Loading subpackage: reconfig...
Forward spacing the tape...
Loading subpackage: emacs...
Forward spacing the tape...
rewinding the tape...

Please mount umips tape number 4 and press return:
rewinding the tape...
Verifying tape id... ok
Forward spacing the tape...

Loading subpackage: posix...
Forward spacing the tape...
Loading subpackage: uucp...
Forward spacing the tape...
Loading subpackage: sccs...
Forward spacing the tape...
Loading subpackage: news_readers...
Forward spacing the tape...
Loading subpackage: games...
Forward spacing the tape...
Loading subpackage: mh...
Forward spacing the tape...
rewinding the tape...

========== making device special files ==========


========== running comply ==========

running first comply pass...
running second comply pass...
There are comply messages from the second pass in:
    /umips4.50/lib/comply.out
(A copy will be placed on the installed system as:)
    /usr/pkg/lib/umips4.50/lib/comply.out

Normally, there should be no second-pass comply messages. The presence
of these messages may indicate problems with the installation.

View the messages now (y n) [y]? y
```
You may see this harmless error message when installing on an M/2000, RC3260, or RC6280.

```
comply: error: link count dev/scsi/ijc0d0s8 3 should be: 2
```

```
(Press return to continue...)


Do you wish to configure the network (y n) [n]? y

========== making special network files ==========

Set hostname [no_hostname]? target
Set netmask [0xffff0000]?
Set broadcast address [255.255.255.0]? 97.10.255.255
Set net address [127.1.0.0]? 97.10.0.55
Should we create the /etc/local_hostname file (y n) [y]?


target 97.10.255.255
Should we add the above entry to the /etc/hosts file (y n) [y]?


Set domain name [mips.com]?
Should we create the /etc/local_domainname file (y n) [y]?




========== cleaning up ==========

Copying packaging information directory to /mnt/usr/pkg/lib/umips4.50...


Copying miniroot fstab to installed system...
Unmounting filesystems...
/mnt/usr: Unmounted
/mnt: Unmounted

========== installation complete ==========
```

Please proceed to the "Post-Installation System Modifications" section to make sure everything is correct before shutting down the miniroot and booting multi-user.

4.3 Installing RISC/os on a Two Disk Rx3230

On a Magnum 3000 or RC3230 with two 200 Megabyte drives, the install variable **TWODISK** must be set before running the **inst** script in order to install the software across both disks. Note that this type of installation in supported **only** on the Rx3230. Unfortunately in RISC/os 4.50, the two disk update installation is broken, and if you are installing on two disks, you **must** do a scratch installation. This will be fixed in the next version of RISC/os. Also note that adding swap space during the installation procedure does not work on a two disk install. You can change the swap partition configuration manually after the installation.

To set the variable, type the following:

TWODISK=y
export TWODISK

When running **inst** the following message will be displayed during the **Installation Information** section:

The usr partition will be on drive 1.

If that message is not displayed, abort the installation and be sure you are installing on an Rx3230 and that TWODISK is set to y.

During the Rx3230 two disk install, you are not asked to select the /usr partition (as you are during a normal install) and partition *sdc0d0s6* will be mounted as /usr1; partition *sdc0d0s7* will be mounted as /usr2. Proceed now with the appropriate installation instructions (update, scratch, or remote tape).

Proceed as follows from the prom monitor prompt:

```
>> boot -f tqsd(,6,2)sash.std
Rewinding the tape.....Done

Forward spacing the tape 2 files.....Done
205648+57312+226400
Rewinding the tape.....Done
 entry: 0xa0300000
kbd rst status 0xffffffff
Standalone Shell: Version 5.0 OPT Tue Apr 24 07:03:07 PDT 1990 rootcsh
sash: cp -b 16k tqsd(,6,4) dksd(,,1)

Rewinding the tape.....Done

Forward spacing the tape 4 files.....Done
13824000 (0xd2f000) bytes copied

Rewinding the tape.....Done
sash: boot -f tqsd(,6,11)unix.r3030_std root=sdc0d0s1

Rewinding the tape.....Done

Forward spacing the tape 11 files.....Done
835440+107152+779056
Rewinding the tape.....Done
 entry: 0x80021000
Parity Enabled
CPU: MIPS R3000 Processor Chip Revision: 2.0
FPU: MIPS R3010 VLSI Floating Point Chip Revision: 2.0

RISC/os Release 4_50 mips Version UMIPS
Total real memory  = 16777216
Available memory   = 14589952
Root on dev 0x2101, Swap on dev 0x2101
Root fstype ffs
New swplo: 26976  swap size: 6168K bytes
Available memory   = 12947456


Miniroot run level 1

Making miniroot device files for RC3230 system...
erase=^H, kill=^U, interrupt=^C
# set -a
# TWODISK=y
```

```
# inst

Software package installation


Installation Information:

This is a SCRATCH install.  Data on the root  and /usr disks will be lost.
Packages will be read in from the local Q24 tape drive.
Machine type: RC3230
The usr partition will be on drive 1.

Is the information above correct? (y n) [y]?

Note: It is not necessary to install a kernel on the miniroot
      The process will take a few minutes.
Would you like to install the kernel to the miniroot (y n) [n]? y
rewinding the tape...
Forward spacing the tape...
Reading the unix.r3030_std kernel from tape...
134+1 records in
134+1 records out
4290+0 records in
4290+0 records out
Done.



========== checking subpackages ==========

The following subpackages may be installed:

root      -- RISC/os Standard Root Filesystem
RC3230    -- RISC/os RC3230/RS3230 Kernel and Devices
usr       -- RISC/os Standard /usr Filesystem
cmplrs    -- MIPS-C Compiler
cmplrs-bsd43     -- MIPS-C 4.3 BSD Include Files and Libraries
man       -- RISC/os Manual Pages
compat    -- RISC/os 4.00 Compatibility Libraries
bsd43     -- RISC/os 4.3 BSD Utilities, Include Files and Libraries
reconfig         -- Kernel Binary Reconfiguration Components
emacs     -- emacs
posix     -- RISC/os POSIX P1003.1 Include Files, Commands and Libraries
uucp      -- UUCP
sccs      -- SCCS
news_readers     -- News Readers
games     -- Games
mh        -- mh


========== selecting subpackages ==========

You may select all of the above subpackages by answering "y" to the
following question.  If you answer "n" then you will be asked to
select the optional subpackages you would like to have installed.

Install ALL subpackages (y n) [n]? y
```

```
========== setting system clock/calendar ==========

The current value of the clock is: Wed Jun 6 18:08:32 PST 1990
Is the clock correct (y n) [y]?

========== verifying single-user mode ==========

The system is in a single-user run level.

Please answer "y" to the following question unless you really understand
the consequences.

Do you want to install sash to the volume header (y n) [y]?

========== installing sash to volume header ==========


========== determining /usr partition ==========

Changing /dev/usr link from /dev/dsk/sdc0d0s6 to /dev/dsk/sdc0d1s2

========== initializing filesystems ==========

A scratch install of an operating system package is being performed
from the miniroot. Normally in this case the filesystems are
initialized.  When a filesystem is initialized, any existing data will
be lost.  You will be given a chance to override initialization of
each individual filesystem below.

Initialize filesystems (y n) [y]? y
Disk type for controller 0 drive 0 [LXT-200S]?
```

See the section titled **Disk Drive Types** *for a list of possible choices, if necessary*

```
Initialize filesystem on /dev/root (y n) [y]?
Disk type for controller 0 drive 1 [LXT-200S]?
```

See the section titled **Disk Drive Types** *for a list of possible choices, if necessary*

```
Initialize filesystem on /dev/usr (y n) [y]?

Initializing the filesystem on /dev/root...
mkfs.ffs: warning: resetting filesystem size to 44336
Warning: 16 sector(s) in last cylinder unallocated
/dev/root:      44336 sectors in 132 cylinders of 7 tracks, 48 sectors
        22.7Mb in 9 cyl groups (16 c/g, 2.75Mb/g, 1088 i/g)
super-block backups (for fsck -b#) at:
 32, 5456, 10880, 16304, 21728, 27152, 32576, 38000, 43040,
rotational delay between contiguous blocks changes from 7ms to 0ms

Checking the filesystem on /dev/root...
** /dev/root
** Last Mounted on
```

```
** Phase 1 - Check Blocks and Sizes
** Phase 2 - Check Pathnames
** Phase 3 - Check Connectivity
** Phase 4 - Check Reference Counts
** Phase 5 - Check Cyl groups
2 files, 9 used, 20774 free (14 frags, 2595 blocks, 0.1% fragmentation)

***** FILE SYSTEM WAS MODIFIED *****

Initializing the filesystem on /dev/usr...
mkfs.ffs: warning: resetting filesystem size to 389744
Warning: 16 sector(s) in last cylinder unallocated
/dev/usr:        389744 sectors in 1160 cylinders of 7 tracks, 48 sectors
        199.5Mb in 37 cyl groups (32 c/g, 5.51Mb/g, 512 i/g)
super-block backups (for fsck -b#) at:
 32, 10832, 21632, 32432, 43232, 54032, 64832, 75632, 86048, 96848,
 107648, 118448, 129248, 140048, 150848, 161648, 172064, 182864, 193664, 204464,
 215264, 226064, 236864, 247664, 258080, 268880, 279680, 290480, 301280, 312080,
 322880, 333680, 344096, 354896, 365696, 376496, 387296,
rotational delay between contiguous blocks changes from 7ms to 0ms

Checking the filesystem on /dev/usr...
** /dev/usr
** Last Mounted on
** Phase 1 - Check Blocks and Sizes
** Phase 2 - Check Pathnames
** Phase 3 - Check Connectivity
** Phase 4 - Check Reference Counts
** Phase 5 - Check Cyl groups
2 files, 9 used, 191886 free (14 frags, 23984 blocks, 0.0% fragmentation)

***** FILE SYSTEM WAS MODIFIED *****

========== mounting filesystems ==========

/dev/root mounted on /mnt
/dev/usr mounted on /mnt/usr
```

Partition		Megs	Mounted File System or Partition Usage
0	/dev/dsk/sdc0d0s0	22	/dev/root
1	/dev/dsk/sdc0d0s1	20	/
3	/dev/dsk/sdc0d0s3	150	-**** Available Partition ****-
4	/dev/dsk/sdc0d0s4	86	-**** Available Partition ****-
5	/dev/dsk/sdc0d0s5	43	-**** Available Partition ****-
6	/dev/dsk/sdc0d0s6	128	-**** Available Partition ****-
7	/dev/dsk/sdc0d0s7	22	-**** Available Partition ****-

Disk Device	/dev/dsk/sdc0d0s2	191	Megabytes Total Size

```
Do you wish to change swap partition configuration (y n) [n]? y
```

```
========== configuring swap space ==========

System has 16 megabytes of main memory

The following is a listing of available disk partitions and sizes.
NOTE: the miniroot installation system is mounted as / on /dev/swap


-------------------------------------------------------------------
Partition                Megs    Mounted File System or Partition Usage
-------------------------------------------------------------------
1   /dev/dsk/sdc0d0s1     20      /
7   /dev/dsk/sdc0d0s7     22      -**** Available Partition ****-
-------------------------------------------------------------------
Disk Device     /dev/dsk/sdc0d0s2        191     Megabytes Total Size
-------------------------------------------------------------------
(Press return to continue...)


It IS NOT recommended that you add an additional swap partition
But you may add more swap space if desired.


-------------------------------------------------------------------
Partition                Megs    Mounted File System or Partition Usage
-------------------------------------------------------------------
3   /dev/dsk/sdc0d0s3     150     -**** Available Partition ****-
4   /dev/dsk/sdc0d0s4     86      -**** Available Partition ****-
5   /dev/dsk/sdc0d0s5     43      -**** Available Partition ****-
6   /dev/dsk/sdc0d0s6     128     -**** Available Partition ****-
7   /dev/dsk/sdc0d0s7     22      -**** Available Partition ****-
15  /dev/dsk/sdc0d0s15    54      -**** Available Partition ****-
-------------------------------------------------------------------
Disk Device     /dev/dsk/sdc0d0s2        191     Megabytes Total Size
-------------------------------------------------------------------
Select from the above displayed available partitions as in
-------------------------------------------------------------------
                          -**** Available Partition ****-
-------------------------------------------------------------------
If no -**** Available Partition ****-  partitions are displayed
then you will need to add an ADDITIONAL DISK if you wish to provide
more swap space. Note that the size of the partition is in megabytes.
It is recommended that the system be configured with 2 or 3 times the
swap space as there is system memory. For example a 16 megabyte memory
system should have 32 megabytes of swap disk space available.
Do you wish to add any swap partitions (y n) [n]? y

(Press return to continue...)

Which partition should be added for swap [# or q to Quit Adding] [q]? 7

Current alternate swap
7 none swap rw,noauto 0 0
Which partition should be added for swap [# or q to Quit Adding] [q]? q
Add additional swap partitions to system /etc/fstab (y n) [n]? y

========== verifying disk space ==========
```

Do you want to check for space (please do so unless you really
understand the consequences) (y n) [y]? **n**

========== extracting files from subpackage archives ==========

rewinding the tape...
Verifying tape id... ok
Forward spacing the tape...

Loading subpackage: root...
Forward spacing the tape...
Forward spacing the tape...
Loading subpackage: RC3230...
Forward spacing the tape...
Loading subpackage: usr...
Forward spacing the tape...
Loading subpackage: cmplrs...
Forward spacing the tape...
Loading subpackage: cmplrs-bsd43...
Forward spacing the tape...
Loading subpackage: man...
Forward spacing the tape...
rewinding the tape...

Please mount umips tape number 2 and press return:
rewinding the tape...
Verifying tape id... ok
Forward spacing the tape...

Loading subpackage: compat...
Forward spacing the tape...
Loading subpackage: bsd43...
Forward spacing the tape...
Loading subpackage: reconfig...
Forward spacing the tape...
Loading subpackage: emacs...
Forward spacing the tape...
Loading subpackage: posix...
Forward spacing the tape...
Loading subpackage: uucp...
Forward spacing the tape...
Loading subpackage: sccs...
Forward spacing the tape...
Loading subpackage: news_readers...
Forward spacing the tape...
Loading subpackage: games...
Forward spacing the tape...
Loading subpackage: mh...
Forward spacing the tape...
rewinding the tape...

========== making device special files ==========

modifying bom file r3030 for new location of /dev/usr...

78

```
========== running comply ==========

running first comply pass...
running second comply pass...
There are comply messages from the second pass in:
    /umips4.50/lib/comply.out
(A copy will be placed on the installed system as:)
    /usr/pkg/lib/umips4.50/lib/comply.out

Normally, there should be no second-pass comply messages. The presence
of these messages may indicate problems with the installation.

View the messages now (y n) [y]? y
```
You may see the following harmless message on M/2000, RC3260, and RC6280 installations:

```
comply: error: link count dev/scsi/ijc0d0s8 3 should be: 2



Do you wish to configure the network (y n) [n]? y

========== making special network files ==========

Set hostname [no_hostname]? mips
Set netmask [0xffff0000]? 0xffffff00
Set broadcast address [255.255.255.0]? 130.62.10.255
Set net address [127.1.0.0]? 130.62.10.105
Should we create the /etc/local_hostname file (y n) [y]?

mips 130.62.10.105
Should we add the above entry to the /etc/hosts file (y n) [y]?


Set domain name [mips.com]?
Should we create the /etc/local_domainname file (y n) [y]?



========== cleaning up ==========

Copying packaging information directory to /mnt/usr/pkg/lib/umips4.50...
Copying miniroot fstab to installed system...
Unmounting filesystems...
/mnt/usr: Unmounted
/mnt: Unmounted

========== installation complete ==========

# init 0
#
INIT: New run level: 0
/mnt: Unmounted

Miniroot shutdown
```

```
RC3230 MIPS Monitor: Version 5.40 MIPS OPT Wed Apr 4 19:34:16 PDT 1990 root
Memory size: 16777216 (0x1000000) bytes, 16 MB
Icache size: 32768 (0x8000) bytes
Dcache size: 32768 (0x8000) bytes
>>
```

4.4 Installing RISC/os 4.50 from Remote Tape

To install RISC/os 4.50 from a remote tape drive on a RISComputer over the network, follow the instructions provided in this section. Note that the network installation cannot be done through a gateway machine; both machines must be on the same network. Also note that the remote tape installation is time consuming. If the network is loaded and/or dirty, the installation time will be increased.

Before beginning, there are a couple of definitions to clarify:

Server Machine:
A MIPS system with cartridge tape drive.
Target Machine:
The MIPS machine where the software is to be installed.

1. Determine which machine to use as the server. The server must be on the same network as the target machine. The server must also have a cartridge tape drive and about 25 or 30 megabytes of free disk space to be used during the installation process.

2. Log on to the server as root.

3. A few items must be in place on the installing machine (the server) so that it can communicate with the network that is attached to the target machine. Specifically, the IP address and name must exist in the */etc/hosts* file, and the server must have the network of the target in the **netstat -r** information. This information should be supplied and checked by your network administrator.

4. Check to see that **bfsd** (boot file server daemon) is running on the server. To check, run the following command:

   ps -e | grep bfsd

   You may see two processes, one for **bfsd** itself and one for the **grep** command. If **bfsd** is not running, start it with the following command:

   /etc/bfsd

5. The server's */.rhosts* file must include a line:

   ```
   target_name root
   ```

   where *target_name* is the name of the target machine being installed. If the name is not already in the file, the easiest way to do this is with the following command:

   echo target_name root >> /.rhosts

6. The target's */etc/hosts* file must be updated to include its IP address and name and the server's IP address and name. If necessary, the easiest way to do this is with the following two commands:

   echo xx.xx.xx.xx target_name >> /etc/hosts
   echo yy.yy.yy.yy server_name >> /etc/hosts

where *xx.xx.xx.xx* is the IP address assigned to the target, and *yy.yy.yy.yy* is the IP address of the server machine. All this information is decided by the network administrator at your site.

7. Put the tape into the server's tape drive.

8. Load the miniroot, both versions of **sash**, and the kernels onto the server's disk in a directory on a filesystem with at least 35 megabytes (which can be freed up after the installation) of free space (in this example, */usr/netinstall* is used), as follows:

Notes:

- Be careful to use the correct device, as listed here, for the next few commands (either Q24-0 or Q24n-0).

- The **dd** of the miniroot takes several minutes.

```
#  cd /usr/netinstall  (or some other directory where there is 35 meg free)
#  mt -f /dev/rmt/Q24-0 rewind
#  mt -f /dev/rmt/Q24n-0 fsf 2
#  dd if=/dev/rmt/Q24n-0 of=sash.std skip=1
1080+0 records in
1080+0 records out
#  dd if=/dev/rmt/Q24n-0 of=sash.2030 skip=1
768+0 records in
768+0 records out
#  dd if=/dev/rmt/Q24n-0 of=miniroot bs=5k
2700+0 records in
2700+0 records out
#  dd if=/dev/rmt/Q24n-0 of=unix.r2300_std skip=1
4150+0 records in
4150+0 records out
#  dd if=/dev/rmt/Q24n-0 of=unix.r2400_std skip=1
4094+0 records in
4094+0 records out
#  dd if=/dev/rmt/Q24n-0 of=unix.r3200_std skip=1
4285+0 records in
4285+0 records out
#  dd if=/dev/rmt/Q24n-0 of=unix.i2000_std skip=1
4163+0 records in
4163+0 records out
#  dd if=/dev/rmt/Q24n-0 of=unix.r3200_ijc skip=1
4285+0 records in
4285+0 records out
#  dd if=/dev/rmt/Q24n-0 of=unix.r6000_std skip=1
4360+0 records in
4360+0 records out
#  dd if=/dev/rmt/Q24n-0 of=unix.r3030_std skip=1
4330+0 records in
4330+0 records out

#  mt -f /dev/rmt/Q24-0 rewind
```

You are now done issuing commands on the server. The remaining commands are to be issued on the target machine.

9. Log on to the target machine as root and bring down the target machine:

   # telinit 0

10. Boot **sash** from the server machine (substitute your server's name in place of *servername* in the following commands):

```
M/1000 MIPS Monitor Version 4.22 MIPS OPT Tue Aug 29 09:59:10 PDT 1989 root
Memory size: 50331648 (0x3000000) bytes
Icache size: 65536 (0x10000) bytes
Dcache size: 65536 (0x10000) bytes
```
>> **boot -f bfs()servername:/usr/netinstall/sash.std** *for non-Rx2030*
>> **boot -f bfs()servername:/usr/netinstall/sash.2030** *for Rx2030*
```
Obtaining servername:/usr/netinstall/sash.XXX from server servername
207632+57504+228704 entry: 0xa0300000
Standalone Shell: Version 5.0 OPT Wed Jun 6 14:48:44 PDT 1990 root
```

11. Copy the miniroot from the server to the target machine.

   sash: **cp -b 16k bfs()servername:/usr/netinstall/miniroot dksd(,,1)** *for Rx2030, Rx3230*
   sash: **cp -b 16k bfs()servername:/usr/netinstall/miniroot dkis(,,1)** *for M/120 or RC3240)*
   sash: **cp -b 16k bfs()servername:/usr/netinstall/miniroot dkip(,,1)** *for M/500, 800, 1000, M/2000 SMD, or RC6280*
   sash: **cp -b 16k bfs()servername:/usr/netinstall/miniroot dkij(,,1)** *for M/2000 SCSI or RC3260*
```
Obtaining servername:/usr/netinstall/miniroot from server servername
13824000 (0xd2f000) bytes copied
```

12. Boot the kernel:

   sash: **boot -f bfs()servername:/usr/netinstall/unix.i2000_std root=sdc0d0s1 console=g** *(for RS2030)*
   sash: **boot -f bfs()servername:/usr/netinstall/unix.i2000_std root=sdc0d0s1 console=0** *(RC2030 tty 0)*
   sash: **boot -f bfs()servername:/usr/netinstall/unix.i2000_std root=sdc0d0s1 console=1** *(RC2030 tty1)*
   sash: **boot -f bfs()servername:/usr/netinstall/unix.r3030_std root=sdc0d0s1** *(Rx3230)*
   sash: **boot -f bfs()servername:/usr/netinstall/unix.r2400_std root=isc0d0s1** *(M/120, RC3240)*
   sash: **boot -f bfs()servername:/usr/netinstall/unix.r2300_std root=ipc0d0s1** *(M/500, 800, 1000)*
   sash: **boot -f bfs()servername:/usr/netinstall/unix.r3200_ijc root=ijc0d0s1** *(RC3260 orM/2000 SCSI)*
   sash: **boot -f bfs()servername:/usr/netinstall/unix.r3200_std root=ipc0d0s1** *(M/2000 SMD)*
   sash: **boot -f bfs()servername:/usr/netinstall/unix.r6000_std root=ipc0d0s1** *(RC6280)*
```
Obtaining servername:/usr/netinstall/unix.r2300_std from servername server
812560+99312+776624 entry: 0x80021000
CPU: MIPS R2000 Processor Chip Revision: 5.0

RISC/os Release 4_50 mips Version UMIPS
Total real memory  = 50331648
Available memory   = 47816704
Root on dev 0x401, Swap on dev 0x401

WARNING: lost battery backup clock

WARNING: CHECK AND RESET THE DATE!
Root fstype ffs
New swplo: 26992  swap size: 6344K bytes
Available memory   = 46522368


UMIPS Miniroot run level 1
```

```
Making miniroot device files for m1000 system...
erase=^H, kill=^U, interrupt=^C
#
```

13. At this point, the miniroot file system has been copied to the swap partition and is running the Bourne shell, **sh**. Set the *EnTapehost* variable and perform the installation:

> # **set -a**
> # **EnTapehost=servername**
> # **Install=update** *for update installs only*
> # **inst**

```
Software package installation

Remote Tape Installation selected.
The tape must be mounted on the machine: servername


=====  initializing the network  =====

Enter the hostname of this machine []? target
Enter the netmask [0xffff0000]? 0xffffff00
Enter the broadcast address [97.0.0.0]? 130.62.9.255

hostname:   target
server:     servername
netmask:    0xffffff00
broadcast:  130.62.9.255

Ok (y n) [y]?


Installation Information:

This is a remote tape install.
Packages will be read in from the tape drive on target.
Machine type: m120-5

Is the information above correct? (y n) [y]? y
Note: It is not necessary to install a kernel on the miniroot
      The process will take a few minutes.
Would you like to install the kernel to the miniroot (y n) [n]? n


========== checking subpackages ==========

The following subpackages may be installed:

root    -- RISC/os Standard Root Filesystem
m1000   -- RISC/os m/500, m/800, m/1000 Kernel and Devices
usr     -- RISC/os Standard /usr Filesystem
cmplrs  -- MIPS-C Compiler
cmplrs-bsd43    -- MIPS-C 4.3 BSD Include Files and Libraries
man     -- RISC/os Manual Pages
compat  -- RISC/os 4.00 Compatibility Libraries
```

```
bsd43    -- RISC/os 4.3 BSD Utilities, Include Files and Libraries
reconfig        -- Kernel Binary Reconfiguration Components
emacs    -- emacs
posix    -- RISC/os POSIX P1003.1 Include Files, Commands and Libraries
uucp     -- UUCP
sccs     -- SCCS
news_readers    -- News Readers
games    -- Games
mh       -- mh


========== selecting subpackages ==========


You may select all of the above subpackages by answering "y" to the
following question.  If you answer "n" then you will be asked to
select the optional subpackages you would like to have installed.

Install ALL subpackages (y n) [n]? n

When asked if you want to install a subpackage, please answer with
one of the following:

   y - Yes, you want to install the subpackage
   n - No, you do NOT want to install the subpackage
   l - List the contents of the subpackage and ask me again

Subpackage root will be installed.
Subpackage m120-5 will be installed.
Subpackage usr will be installed.
Subpackage cmplrs will be installed.
Install subpackage cmplrs-bsd43 (l y n) [n]? y
Install subpackage man (l y n) [n]? y
Install subpackage compat (l y n) [n]? y
Install subpackage bsd43 (l y n) [n]? y
Subpackage reconfig will be installed.
Install subpackage emacs (l y n) [n]? y
Install subpackage posix (l y n) [n]? y
Install subpackage uucp (l y n) [n]? y
Install subpackage sccs (l y n) [n]? y
Install subpackage news_readers (l y n) [n]? y
Install subpackage games (l y n) [n]? y
Install subpackage mh (l y n) [n]? y


Selected subpackages:
    root m120-5 usr cmplrs cmplrs-bsd43 man compat bsd43 reconfig emacs posix uucp sc
Is this what you want (y n) [y]? y

========== setting system clock/calendar ==========


The current value of the clock is: Wed Jun 6 18:08:32 PST 1990
Is the clock correct (y n) [y]? y
========== verifying single-user mode ==========


The system is in a single-user run level.
Do you want to install the volume header (please do so unless you really
understand the consequences) (y n) [y]? y
```

84

```
========== installing sash to volume header ==========

========== mounting filesystems ==========

/dev/root mounted on /mnt
/dev/usr mounted on /mnt/usr
```

```
---------------------------------------------------------------------
Partition                 Megs      Mounted File System or Partition Usage
---------------------------------------------------------------------
0   /dev/dsk/isc0d0s0     22        /dev/root
1   /dev/dsk/isc0d0s1     20        /
6   /dev/dsk/isc0d0s6     250       /dev/usr
7   /dev/dsk/isc0d0s7     22        -**** Available Partition ****-
---------------------------------------------------------------------
Disk Device     /dev/dsk/isc0d0s2       312       Megabytes Total Size
---------------------------------------------------------------------
```

```
Do you wish to change swap partition configuration (y n) [n]? y
```

*In most cases, you will probably not want to change the swap partition
configuration here, but this example is included for completeness.*

```
========== configuring swap space ==========

System has 32 megabytes of main memory

The following is a listing of available disk partitions and sizes.
NOTE: the miniroot installation system is mounted as / on /dev/swap
```

```
---------------------------------------------------------------------
Partition                 Megs      Mounted File System or Partition Usage
---------------------------------------------------------------------
1   /dev/dsk/isc0d0s1     20        /
7   /dev/dsk/isc0d0s7     22        -**** Available Partition ****-
---------------------------------------------------------------------
Disk Device     /dev/dsk/isc0d0s2       312       Megabytes Total Size
---------------------------------------------------------------------
(Press return to continue...)
It IS recommended that you select an additional swap partition
Here are partitions you may select from for additional swap space
---------------------------------------------------------------------
Partition                 Megs      Mounted File System or Partition Usage
---------------------------------------------------------------------
7   /dev/dsk/isc0d0s7     22        -**** Available Partition ****-
---------------------------------------------------------------------
Disk Device     /dev/dsk/isc0d0s2       312       Megabytes Total Size
---------------------------------------------------------------------
Select from the above displayed available partitions as in
---------------------------------------------------------------------
                                     -**** Available Partition ****-
---------------------------------------------------------------------
If no -**** Available Partition ****-  partitions are displayed
then you will need to add an ADDITIONAL DISK if you wish to provide
more swap space. Note that the size of the partition is in megabytes.
```

It is recommended that the system be configured with 2 or 3 times the
swap space as there is system memory. For example a 16 megabyte memory
system should have 32 megabytes of swap disk space available.
Do you wish to add any swap partitions (y n) [n]? **y**
(Press return to continue...)
Which partition should be added for swap [# or q to Quit Adding] [q]? **7**
Current alternate swap
/dev/dsk/isc0d0s7 none swap rw,noauto 0 0
Which partition should be added for swap [# or q to Quit Adding] [q]? **q**
Add additional swap partitions to system /etc/fstab (y n) [n]? **y**
========== preserving local files ==========

Running preserve -s for subpackage root... 54 files preserved.
No preserve list or findmods list for m120-5- preserve not executed.
Running preserve -s for subpackage usr... 25 files preserved.
No preserve list or findmods list for cmplrs- preserve not executed.
No preserve list or findmods list for cmplrs-bsd43- preserve not executed.
No preserve list or findmods list for man- preserve not executed.
No preserve list or findmods list for compat- preserve not executed.
No preserve list or findmods list for bsd43- preserve not executed.
No preserve list or findmods list for reconfig- preserve not executed.
No preserve list or findmods list for emacs- preserve not executed.
No preserve list or findmods list for posix- preserve not executed.
No preserve list or findmods list for uucp- preserve not executed.
No preserve list or findmods list for sccs- preserve not executed.
No preserve list or findmods list for news_readers- preserve not executed.
No preserve list or findmods list for games- preserve not executed.
No preserve list or findmods list for mh- preserve not executed.

========== verifying disk space ==========

Do you want to check for space (please do so unless you really
understand the consequences) (y n) [y]? **y**
The system will now be checked to verify that there is enough disk
space with the current configuration to successfully install the package
(and any selected optional subpackages). For large packages (especially
operating system packages), this can be time consuming...

You will see one of the following responses from the system:

device	bfree	ifree	breq	ireq	bcred	icred	
/dev/root	5032	8827	13081	396	11344	348	
/dev/usr	19607	102942	87942	4744	72948		4574

WARNING! This package will fit on the disk,
but it will cause more than 90% of the disk to be used.
This may cause problems for non-root users.
It is recommended that you abort the installation now.

Abort the installation? (y n) [y]? **n**

or:

There is enough space.

86

```
========== stripping old links ==========

Stripping links for subpackage root...
Stripping links for subpackage m120-5...
Stripping links for subpackage usr...
Stripping links for subpackage cmplrs...
Stripping links for subpackage cmplrs-bsd43...
Stripping links for subpackage man...
Stripping links for subpackage compat...
Stripping links for subpackage bsd43...
Stripping links for subpackage reconfig...
Stripping links for subpackage emacs...
Stripping links for subpackage posix...
Stripping links for subpackage uucp...
Stripping links for subpackage sccs...
Stripping links for subpackage news_readers...
Stripping links for subpackage games...
Stripping links for subpackage mh...
========== extracting files from subpackage archives ==========

Verifying tape id... ok
Forward spacing the tape...

Loading subpackage: root...
Forward spacing the tape...
Forward spacing the tape...
Loading subpackage: m120-5...
Forward spacing the tape...
rewinding the tape...

Please mount umips tape number 2 and press return:
rewinding the tape...
Verifying tape id... ok
Forward spacing the tape...

Loading subpackage: usr...
Forward spacing the tape...
Loading subpackage: cmplrs...
Forward spacing the tape...
Loading subpackage: cmplrs-bsd43...
Forward spacing the tape...
Loading subpackage: man...
Forward spacing the tape...
rewinding the tape...

Please mount umips tape number 3 and press return:
rewinding the tape...
Verifying tape id... ok
Forward spacing the tape...


Loading subpackage: compat...
Forward spacing the tape...
Loading subpackage: bsd43...
Forward spacing the tape...
```

```
Loading subpackage: reconfig...
Forward spacing the tape...
Loading subpackage: emacs...
Forward spacing the tape...
rewinding the tape...

Please mount umips tape number 4 and press return:
rewinding the tape...
Verifying tape id... ok
Forward spacing the tape...

Loading subpackage: posix...
Forward spacing the tape...
Loading subpackage: uucp...
Forward spacing the tape...
Loading subpackage: sccs...
Forward spacing the tape...
Loading subpackage: news_readers...
Forward spacing the tape...
Loading subpackage: games...
Forward spacing the tape...
Loading subpackage: mh...
Forward spacing the tape...
rewinding the tape...
========== making device special files ==========

========== running comply ==========

running first comply pass...
running second comply pass...
There are comply messages from the second pass in:
    /umips4.50/lib/comply.out
(A copy will be placed on the installed system as:)
    /usr/pkg/lib/umips4.50/lib/comply.out
```

Normally, there should be no second-pass comply messages. The presence
of these messages may indicate problems with the installation.

```
View the messages now (y n) [y]? y
```
You may see the following harmless message on M/2000, RC3260, and RC6280 installations:

```
comply: error: link count dev/scsi/ijc0d0s8 3 should be: 2

(Press return to continue...)
```

```
========== cleaning up old versions ==========
```

An attempt will now be made to clean up any files left over from previous
versions of the software which has just been installed.

It's probably safe to remove old versions of standard OS packages.
If you want the 2.0 version of the compiler left on the system in addition
to or instead of to using the RISCos4.0 compatibility package, you won't
want to remove the compiler package.

Searching for old versions to remove...

========== restoring preserved user files ==========

Running preserve -r for subpackage root...
No preserve list or findmods list for m120-5- no files restored.
Running preserve -r for subpackage usr...
No preserve list or findmods list for cmplrs- no files restored.
No preserve list or findmods list for cmplrs-bsd43- no files restored.
No preserve list or findmods list for man- no files restored.
No preserve list or findmods list for compat- no files restored.
No preserve list or findmods list for bsd43- no files restored.
No preserve list or findmods list for reconfig- no files restored.
No preserve list or findmods list for emacs- no files restored.
No preserve list or findmods list for posix- no files restored.
No preserve list or findmods list for uucp- no files restored.
No preserve list or findmods list for sccs- no files restored.
No preserve list or findmods list for news_readers- no files restored.
No preserve list or findmods list for games- no files restored.
No preserve list or findmods list for mh- no files restored.

========== running conversion scripts ==========

========== root.fstab   Thu May 24 15:29:55 PST 1990 ==========

No ips devices found in /etc/fstab.

To take advantage of the improved parallel fsck, /etc/fstab will now
be modified to allow the root partition to be fsck'd on the first pass.
All other local filesystems will be fsck'd on the second pass.

A copy of /etc/fstab will be saved as /etc/fstab.save.1.

No changes made.

Press return to continue:

========== cleaning up ==========

Copying packaging information directory to /mnt/usr/pkg/lib/umips4.50...

Various messages about Removing Duplicate File are seen here

Unmounting filesystems...
/mnt/usr: Unmounted
/mnt: Unmounted
========== installation complete ==========

#

Please proceed to the "Post-Installation System Modifications" section to make sure everything is correct before shutting down the miniroot and booting multi-user.

If a remote tape install is required on more than one target machine, just repeat the above steps on each target system. When you have completed all of the installations, remove the **sash** and miniroot files in /usr/netinstall (or the filesystem used) on the server.

4.5 Post-Installation System Modifications

After the software is installed on a new system, some administration is required. For instance, as shipped, the user "root" is not protected with a password. The root user should be provided with a password for security. Login as root and edit the appropriate administrative files to set up other user names and login directories. See also the *System Administrator's Guide* and *telinit*(1m), *inittab*(4), and *passwd*(4) in the *System Administrator's Reference Manual*.

4.5.1 NFS Performance Tuning

Depending on your network configuration, some tuning can be done in the /etc/init.d/nfs file to improve NFS performance. The number of block I/O daemons (**biod**) that are started should always be 4. Increasing this number will probably only begin to flood NFS servers since there are usually many clients per server. The number of NFS daemons (**nfsd**) can be changed. The default is 4, but the formula to be used for optimum performance is 4 times the min(# of disk controllers, # of ethernet interfaces). Basically, the ethernet or disk controllers represent bottlenecks, so the minimum of these is used. Four per unit should be sufficient to retire typical loads. Diskless machines, of course, need 0 **nfsds**.

4.5.2 For Update Installs

As with prior releases, during an upate installation RISC/os 4.50 preserves files which may have been locally modified, such as various start up scripts and configuration files. It is **critically important** that the RISC/os 4.50 versions of these files replace the current versions, possibly after merging with any local changes. Successful booting of the system after unscheduled system shutdowns requires that these files be at least the RISC/os 4.50 version. One way to find all the candidates for merging and replacement is to use **find**(1) to look for all files on the root and /usr partitions whose names include ":".

```
<<< boot to single user mode >>>
# /etc/mount /dev/usr
# find / -name "*:*" -print > /tmp/colon_files
```

Typically there are a great many of these files "left over" from previous installations. These older versions may be simply deleted. In replacing or merging these files, do not forget /.login, /.profile and /.cshrc, as the changes there are also important for correct system operations.

The following is a list of critical files that **must** be the RISC/os 4.50 version plus any local modifications:

> /etc/syslog.conf
> /etc/fstab
> /etc/init.d/MOUNTFSYS
> /etc/init.d/RMTMPFILES
> /etc/init.d/netdaemons
> /etc/init.d/nfs

/etc/init.d/sysetup
/etc/init.d/tcp
/etc/timed.conf
/usr/spool/cron/crontabs/root

If you have not modified any of these files, they will be replaced with the new RISC/os 4.50 version. If a file has been modified, you will need to alter the RISC/os 4.50 version in a corresponding manner before replacing your customized version of the file.

All of the *:*<version>*+ files should be reviewed, but pay particular attention to the files in */etc* and */etc/init.d*. **Note: It is very important to use the new** *fstab*(4) **prioritization scheme before any file systems are mounted. If not done in this order, file systems may be mounted without fsck checking.**

Note that when you modify */etc/fstab*, the last field **must** be set to some non-zero number before that filesystem will be checked by **fsck** during normal mode of operation. Therefore, the last field on a swap device should remain zero. In general, you want to set the root filesystem to "1" and everything else to "2". A "0" means that partition will not be checked.

Without the 4.50 versions of the */etc/init.d/{MOUNTFSYS, tcp, nfs, netdaemons}* files, some of the system daemons will not start up properly and disk quotas will not be enabled.

4.5.3 Sendmail Configuration File

Because the **sendmail** configuration file contains data structures built into it that rely on the particular version of sendmail that uses it, a previously frozen configuration file will not work. The *.cf* version of the **sendmail** file is preserved, but the *.fc* version must be remade with the command **sendmail -bz**. Note that the empty file */usr/lib/sendmail.ok* must be created if */bin/mail* is to hand off mail to **sendmail** for routing. See *mail*(1) and *sendmail*(1) for more information.

4.5.4 Sendmail and VIS

Note that if there is no */etc/vis.conf* file, mail may bounce with a host unknown message. If you see these errors, be sure to check for a *vis.conf* file. The default value for */etc/vis.conf* when it does not exist is:

host: files

Unfortunately, if you are on a machine whose */usr/lib/aliases* file uses domain-style names but the */etc/hosts* file does not, this will be a problem. When VIS is configured as the default, **sendmail** ignores **named** and looks only at */etc/hosts*. If */etc/hosts* and */usr/lib/aliases* do not use consistent host names, **sendmail** gets "not found" from the sending machine.

4.5.5 Adding ttys

In addition to setting up user accounts, **getty** entries in the */etc/gettydefs* file must be activated. The word "off" in field 3 of the file must be changed to "respawn" in order to activate these entries. See the *System Administrator's Guide* and the *gettydefs*(4) man page for more information.

4.5.6 Device Files

If your system is configured with additional peripherals, you may need to create new device files in */dev* with the */dev/MKDEV* shell script. For instance, to create node names for a disk on a second controller of the M/2000, one would give the commands:

```
# cd /dev
# ./MKDEV -p ipc1d0
```

For more information. see the *MKDEV*(1M) man page and the *README* and *m*.system* file (whichever is appropriate) in */dev/DEV_DB*.

4.5.7 Permissions on Device Files

If you have modified any of the permissions on device files in */dev*, they will need to be reset after the installation procedure, since these files are re-made with standard permissions as part of the installation.

4.5.8 Swap Space

If your system requires additional swap partitions and additional partitions were not configured during installation, they must be added to */etc/fstab* so they are added when the system is booted multi-user. The format for a valid swap partition entry is:

```
block-device ignored swap rw,noauto 0 0
```

For example, the entry

```
/dev/dsk/ipc0d1s7 none swap rw,noauto 0 0
```

will result in controller 0, disk 1, partition 7 being added as a swap device.

4.5.9 Network Configuration

And finally, if a network is to be used on the new system, there are a couple of files that require modification: */etc/hosts* and */etc/local_hostname*, if not already done during the installation. For the *hosts* file, simply add the system names and internet addresses for the machines on your network. For *local_hostname*, change the 'no_hostname_set' to the hostname for the system, and change '255.255.255.255' to the broadcast address for your system. Note that the internet address and the broadcast address are **not** the same numbers. For more information on how to determine these addresses, see the *System Administrator's Guide*.

4.5.10 Post-Installation Work for BSD Installs

Not everything involved in a UMIPS-BSD to RISC/os conversion can be automated, and the following sections describe things the system administrator must take care of manually.

UMIPS-BSD Binaries

All the UMIPS-BSD executables will dump core with bad system calls so they must be deleted. First set the path to avoid them:

 # set path=(/bin /etc /usr/bin)

This **find** looks for executables from the UMIPS-BSD distribution. In this example, the number 24 comes from the difference between the date when this system was installed and the date on the release executables. The RISC/os 4.50 executables are timestamped with the date June 06 1990 18:47 PST (or 644723220 from a **/usr/pkg/bin/dates -l** command). To find the correct number to use, subtract 644723220 from the number given from a **dates -l** *installation_date* command (where *installation_date* is the date and time of the installation) and substitute that number for 24.

92

```
# find / -mtime +24 -perm -0111 -print > /tmp/bsd_exe
# more /tmp/bsd_exe
/bin/domainname
/bin/ed.int
/bin/ed.usa
/bin/hostid
/bin/hostname
/bin/pagesize
/bin/rcp
/bin/tp
/bin/wall
/dev/MAKEDEV
/etc/XNSrouted
/etc/ac
/etc/analyze
/etc/badsect
/etc/catman
/etc/chown
/etc/comsat
/etc/config
/etc/dcheck
/etc/rc:2.1:
/etc/dmesg
/etc/dumpfs
/etc/dxpr
/etc/edquota
/etc/fastboot
/etc/fasthalt
/etc/flcopy
/etc/fsirand
/etc/fstat
/etc/gettable
/etc/halt
/etc/rc
/etc/htable
/etc/icheck
/etc/implog
/etc/implogd
/etc/kgmon
/etc/mkhosts
/etc/mklost+found
/etc/mkpasswd
/etc/mkpdata
/etc/pac
/etc/mkproto
/etc/newfs
/etc/pstat
/etc/rc.newsystem
/etc/rc.yp
/etc/reboot
/etc/renice
/etc/repquota
/etc/sa
/etc/slattach
/etc/swapon
```

```
/etc/timed
/etc/timedc
/etc/trpt
/etc/trsp
/etc/quot
/etc/tunefs
/etc/vipw
/etc/diskpart
/etc/format
/etc/pcnfsd
/etc/comply.2.0
/etc/rc.local.orig
/etc/in.tftpd
/etc/rexd
/etc/mkprntr
/etc/imadb_prt
/etc/rpc.bootparamd
/etc/quotacheck
/etc/quotaoff
/etc/quotaon
/etc/rc.local
/stand/dbgmon
```

Any of these files that are actual binaries should be removed. If they are shellscripts, they can remain on the system. The following script helps you to sort through these files by displaying information about each file and removing it if you enter y. This allows you to keep selected files.

```
#  foreach i ('cat /tmp/bsd_exe')
?  ls -l $i
?  if ( $< == "y" ) rm $i
?  end
```

At this point there will be lots of ls -l output displayed, and prompts for removal of files. Once this has finished, mount the *usr* filesystem and repeat the process:

```
#  mount /dev/dsk/ipc0d0s6 /usr
#  find /usr -mtime +24 -perm -0111 -print > /tmp/bsd_exe
#  foreach i ( 'cat /tmp/bsd_exe' )
?  ls -l $i
?  if ( $< == "y" ) rm $i
?  end
```

The system can now be rebooted.

/etc/exports

On UMIPS-BSD systems, there is a file */etc/exports*. On RISC/os 4.00, */etc/exports* is a link to */usr/etc/exports* because of space constraints on the root filesystem. When converting from UMIPS-BSD to RISC/os 4.00, be sure to move */etc/exports* to */usr/etc/exports*, leaving a link in */etc* pointing to the actual file.

4.6 Shutting Down the Miniroot and Booting Multi-User

Once all necessary configuration is done on the miniroot, proceed as follows to shutdown the miniroot:

```
#  cd /
#  sync; sync
#  /etc/telinit 0
```

To boot RISC/os, enter **auto** at the monitor prompt. The system may display some messages, **fsck** the root partition, and reboot. The default run level is now "2" (multi-user). To override the default and boot in single user mode, instead of entering **auto**, use:

>> **boot dkip()unix initarg=s**   *(for M/500, 800, 1000, 2000, RC6280)*
>> **boot dksd()unix initarg=s**   *(for Rx2030, Rx3230)*
>> **boot dkis()unix initarg=s**   *(for M/120 or RC3240)*
>> **boot dkij()unix initarg=s**   *(for SCSI M/2000, RC3260)*

5. Appendix A. SVID Conformance

The following is an overview of current SVID components (sections) and subsections. Components marked (SVVS) are tested by SVVS; each component is also classified as one of "RISC/os high conformance", "RISC/os general conformance", or "RISC/os unsupported". RISC/os 4.00 supports NFS and TCP/IP and does not support RFS and its underlying protocols as defined by the optional Network Services Extensions.

Note: This release conforms to SVID, Issue 2, Base Systems and Kernel Extensions sections, with the exception that MIPS supports IEEE P754 Floating Point Standard. MIPS has chosen to follow this standard, with the understanding that AT&T has stated conformance to IEEE P754 as a SVID Future Direction.

"High conformance" indicates that we consider it important to conform as highly as possible.

"General conformance" indicates that, in general, we try to support all of the functionality, and conform to the maximum extent deemed practical. All divergences will be documented. However, to-the-letter conformance is not guaranteed.

Currently, the SVID is broken down into:

Base System (SVVS) (RISC/os high conformance)

    BA_ENV   BA_OS   BA_LIB

Kernel Extension (SVVS) (RISC/os high conformance)

    KE_ENV   KE_OS

Basic Utilities Extension (RISC/os general conformance)

    BU_CMD

Advanced Utilities Extension (RISC/os general conformance)

    AU_CMD

Software Development Extension (RISC/os general conformance)

    SD_LIB   SD_CMD

Administered Systems Extension (RISC/os general conformance)

    AS_CMD

Terminal Interface Extension (SVVS) (RISC/os high conformance)

    TI_ENV   TI_LIB

Network Services Extension (SVVS) (RISC/os unsupported)

    NS_LIB   NS_DEV   NS_OS   NS_CMD

5.1 Base System (SVVS)

(RISC/os high conformance)

5.1.1 BA_ENV

5.1.2 BA_OS

5.1.3 BA_LIB

Floating-point and error handling:

All MIPS machines conform to the IEEE 754 floating-point standard. The MIPS math library (*libm.a*) is primarily based on algorithms from Cody and Waite. Some of the functions are taken from the 4.3 BSD math library which was produced under the direction of W. Kahan. a major contributor to the IEEE 754 standard.

MIPS handling of floating point errors is consistent with the IEEE 754 standard. In the future, SVID will match this standard but currently there are differences. Most notably 'errno' is not set on floating-point errors. Where a correct result overflows, a properly signed infinity is returned instead of returning +-HUGE and setting 'errno' to ERANGE. Where the operation's arguments are invalid, the default NaN is returned, unlike SVID which sets 'errno' to EDOM and returns an operation specific value. These are the results in the default rounding mode. Correct results in other rounding modes are not supported.

MIPS intends to support IEEE user trap handlers in a future release. We are currently investigating methods of doing this with reasonable costs to non-exceptional operations. At that time, function calls will look atomic to the user and the information presented to the trap handler will indicate the function which caused the exception, not an instruction in that function's code.

Currently, only the raw interface, intended to be used by the code to implement IEEE user trap handlers, is available. IEEE floating-point exceptions are enabled by setting the enable bit for a particular exception in the floating-point control and status register. When an exception occurs. the UNIX signal SIGFPE is sent to the process. It is up to the signal handler to determine the instruction that caused the exception and to take the desired action. The identity of the instruction that caused the exception can be found in one of two places. If the older MIPS floating-point board (R2360) is used, then the instruction that caused the exception is in the floating-point exception instruction register. In all other implementations the instruction that caused the exception is at the address in the program counter, as modified by the branch delay bit in the cause register. Both the program counter and cause register are in the sigcontext structure passed to the signal handler. This is the way it is done on 4.3 BSD systems.

The signal handler function is declared as follows:

```
sigfpe_handler(sig, code, scp)
int sig, code;

struct sigcontext *scp;
```

If program execution is to be continued past the instruction that caused the exception, the program counter in the signal context must be advanced. If the instruction is in a branch delay slot, then the branch must be emulated to determine whether the branch is taken; the resulting program counter can then be calculated.

5.1.4 BESSEL(BA_LIB)

- Non-positive arguments to y0(), y1() or yn() return a NaN indicating the invalid operation.

- There are no error returns for j0(), j1() or jn().

- Arguments too large in magnitude cause j0(), j1(), y0() and y1() to return zero with no indication of the total loss of precision. In the SVVS 3.0 release, tests of jn(4, HUGE) and yn(9, HUGE) are expected to return zero and print a message about TLOSS. This expected behavior is not documented in any of the SVID volumes 1, 2, or 3 (Issue 2). In the MIPS implementation, these calls return a quiet NaN, in keeping with the behavior of the standard 4.3 BSD version of libm.a, which is based on the IEEE standard.

5.1.5 EXP(BA_LIB)

- The function exp() returns an Infinity when the correct value would overflow and returns the smallest non-zero number when the correct value would underflow.

- The functions log(x) and log10(x) for values of x less than 0 return the default quiet NaN to indicate the invalid operation.

- The functions log(x) and log10(x) for x == 0 return -Infinity.

- The function pow() for values that overflow returns +Infinity. If x is negative and y is not an integer, pow(x,y) returns a NaN.

- pow(x,0) returns 1 for any value of x.

- pow(0,y) returns +Infinity for any value of y.

- The function sqrt() for negative values of x returns the default NaN to indicate the invalid operation.

5.1.6 FLOOR(BA_LIB)

- fabs() is in libc.a not in libm.a.

5.1.7 FREXP(BA_LIB)

- If ldexp() would cause an overflow, a properly signed Infinity is returned. There currently are bugs in ldexp():

If the double argument to ldexp() is a denormalized value, the result is not correct, or the result should have been a denormalized number, the result is 0. If the sum of the integer argument and the double argument's exponent overflows or underflows an integer, then the wrong value gets returned.

5.1.8 GAMMA(BA_LIB)

- The gamma() function has been renamed lgamma() per 4.3 BSD, since what it really returns is ln(gamma(x)) so as to reserve the function name gamma() for the real gamma function. For negative integer arguments +Infinity is returned. If the correct value would overflow, an Infinity is returned.

5.1.9 HYPOT(BA_LIB)

- The function hypot() returns +Infinity if the correct value would overflow or if either argument is an Infinity. It returns a NaN either argument is a NaN.

5.1.10 MATHERR(BA_LIB)

- Not supported, see notes on floating point error handling above.

5.1.11 PRINTF(BA_LIB)

- Floating point infinities and NaNs produce the strings "Infinity" and "NaN", with infinities having a sign if appropriate.

5.1.12 SINH(BA_LIB)

- The functions *sinh*() and *cosh*() return infinities when the correct result would overflow.

5.1.13 STRTOD(BA_LIB)

- If the correct value would overflow, a properly signed Infinity is returned.

5.1.14 TRIG(BA_LIB)

- If the magnitude of the argument of *asin*() or *acos*() is greater than one, a NaN is returned to indicate the invalid operation.

- The result of atan2(0,0) is 0 without any errors (see the RISC/os man page *sin*(3M)).

5.2 Kernel Extension (SVVS)

RISC/os high conformance

5.2.1 KE_ENV

5.2.2 KE_OS

5.3 Basic Utilities Extension

RISC/os general conformance

5.3.1 BU_CMD

5.3.2 AR(BU_CMD)

- The **ar** command does not rebuild the symbol table when the archive is created or updated if it contains an object file. The symbol table is only rebuilt when the 's' option is specified.

- The **ar** command does not allow the option switches to be preceded by a '-'.

5.4 Advanced Utilities Extension

RISC/os general conformance

5.4.1 AU_CMD

5.5 Software Development Extension

RISC/os general conformance

5.5.1 SD_LIB

5.5.2 MARK(SD_LIB) - optional

- Not supported.

5.5.3 MONITOR(SD_LIB) - optional

- To profile the entire program, you must start at address "eprol" rather than "2":

  extern int eprol(), etext();

  monitor(eprol /*not (int(*)())2*/, etext, buf, bufsize,nfunc);

- If the program was compiled with -p2 or -p3, and "monitor" is invoked explicitly, bufsize must be set >= (eprol - etext)/2, and "nfunc" must be set to any nonzero value. All routines between "eprol" and "etext" will be profiled, using the first (eprol - etext)/2 bytes of the buffer. (The SVID allows the buffer to be any size, and will profile only the first "nfunc" routines using the first nfunc*4 bytes of the buffer). Both the MIPS version and the SVID devote the remainder of the buffer to pc samples, if pc-sampling is enabled.

- When environment variable PROFDIR is set to "dirname", the monitoring output file is always *dirname/pid.progname* in the MIPS compiler system; SVID requires "pid.progname" when you link with -p, but "pid.mon.out" when you call monitor explicitly.

5.5.4 NLIST(SD_LIB)

- Only n_value is set.

- n_type is returned either 1 or 0 depending on whether the symbol is found or not, respectively.

- There is a bug in the 1.1 version of the compiler release in which n_value is not set to 0 if the value is not found (n_type gets set to zero, but there is no other indication that the value was not found).

- This routine is found in libmld.a rather than libc.a

5.5.5 SD_CMD

5.5.6 AS(SD_CMD) - optional

- The -V option, in addition to printing "the version of the assembler" (the as command itself), also prints other version information about the compiler passes it uses via the what command.

- as runs the C-preprocessor (cpp) on the source file before assembling by default; this can be turned off with the -nocpp option.

- The future direction for the -Y flag semantics for finding passes in other places is implemented in the current as using the -t, -h and -B flags (see the MIPS man page). The value of the environment variable TMPDIR, if set, is used for temporary file directory.

5.5.7 CC(SD_CMD)

- The -g option generates information for use with dbx; sdb is not supported in the MIPS compiler system.

- The -F option is reserved, but the MIPS specific compiler optimization directives are specified with the -O[0123] options (see the MIPS man page).

- The -Wc,arg1[,arg2 ...] is supported, but a 'c' of '0' or '2' is not used to indicate the compiler (front end) and optimizer; instead 'f' and 'o' are used, respectively.

100

- The future direction for the -Y flag semantics for finding passes in other places is implemented in the current cc using the -t, -h and -B flags (see the MIPS man page).

- The value of the environment variable TMPDIR, if set, is used for temporary file directory.

5.5.8 CPP(SD_CMD)

- The LIBDIR is /usr/lib in the MIPS compiler system. This is not a violation of SVID; however SVID states, "LIBDIR is usually /lib".

5.5.9 DIS(SD_CMD) - optional

- The **dis** command exists with the same basic functionality, but the options specified in the SVID are not supported except for "-F function", which is specified as "-p procedure" (see the MIPS man page).

5.5.10 LD(SD_CMD)

- The MIPS archive table of contents is a hash table, not a linear list. If multiple library members define the same external symbol, the symbol encountered in the first object file that defines it gets placed in the hash table; others are ignored when the archive is created. This has the effect of the link editor never using anything but the first definition of a symbol in an archive to resolve undefined symbols.

5.5.11 PROF(SD_CMD) - optional

- The **prof** command exists, with the same intent: to interpret profile data. However, the options and the form of the information produced is completely different than in the SVID (see the MIPS man page).

5.5.12 SDB(SD_CMD)

- The **sdb** command is not supported; a greatly improved version of the **dbx** command is the symbolic debugger for the MIPS compiler system (see the MIPS man page).

5.6 Administered Systems Extension

RISC/os general conformance

5.6.1 AS_CMD

5.7 Terminal Interface Extension (SVVS)

RISC/os high conformance

5.7.1 TI_ENV

5.7.2 TI_LIB

5.8 Network Services Extension (SVVS)

There is no support in RISC/os for this optional SVID component.

5.9 SVID Commands Summary

This section lists the commands in each section of the SVID along with each each command's state. The possible states are as follows.

ATT - Vanilla port; from AT&T System V.3.

MIPS - Command required modification by MIPS.

PLANNED - Command isn't ported yet, but is on our list.

NOPLAN - Currently, no plan to provide this command.

BU_CMD - Basic Utilities Extension.

ar(BU_CMD)	- MIPS
awk(BU_CMD)	- ATT
banner(BU_CMD)	- ATT
basename(BU_CMD)	- ATT
cal(BU_CMD)	- ATT
calendar(BU_CMD)	- ATT
cat(BU_CMD)	- ATT
cd(BU_CMD)	- Shell Builtin (SVID notwithstanding)
chmod(BU_CMD)	- ATT
cmp(BU_CMD)	- ATT
col(BU_CMD)	- ATT
comm(BU_CMD)	- ATT
cp(BU_CMD)	- ATT
cpio(BU_CMD)	- ATT
cut(BU_CMD)	- ATT
date(BU_CMD)	- ATT
df(BU_CMD)	- ATT
diff(BU_CMD)	- ATT
dirname(BU_CMD)	- ATT
du(BU_CMD)	- ATT
echo(BU_CMD)	- ATT
ed(BU_CMD)	- ATT
expr(BU_CMD)	- ATT
false(BU_CMD)	- ATT
file(BU_CMD)	- ATT
find(BU_CMD)	- ATT
grep(BU_CMD)	- ATT
kill(BU_CMD)	- ATT
line(BU_CMD)	- ATT
ln(BU_CMD)	- ATT
ls(BU_CMD)	- ATT
mail(BU_CMD)	- ATT
mkdir(BU_CMD)	- ATT
mv(BU_CMD)	- ATT
nl(BU_CMD)	- ATT
nohup(BU_CMD)	- ATT
pack(BU_CMD)	- ATT
paste(BU_CMD)	- ATT
pcat(BU_CMD)	- ATT
pg(BU_CMD)	- ATT
pr(BU_CMD)	- ATT
ps(BU_CMD)	- ATT
pwd(BU_CMD)	- ATT
red(BU_CMD)	- ATT

rm(BU_CMD)	- ATT
rmail(BU_CMD)	- ATT
rmdir(BU_CMD)	- ATT
rsh(BU_CMD)	- ATT
sed(BU_CMD)	- ATT
sh(BU_CMD)	- MIPS
sleep(BU_CMD)	- ATT
sort(BU_CMD)	- ATT
spell(BU_CMD)	- ATT
split(BU_CMD)	- ATT
sum(BU_CMD)	- ATT
tail(BU_CMD)	- ATT
tee(BU_CMD)	- ATT
test(BU_CMD)	- Shell Builtin (SVID notwithstanding)
touch(BU_CMD)	- ATT
tr(BU_CMD)	- ATT
true(BU_CMD)	- ATT
umask(BU_CMD)	- Shell Builtin (SVID notwithstanding)
uname(BU_CMD)	- ATT
uniq(BU_CMD)	- ATT
unpack(BU_CMD)	- ATT
wait(BU_CMD)	- Shell Builtin (SVID notwithstanding)
wc(BU_CMD)	- ATT

AU_CMD - Advanced Utilities Extension.

at(AU_CMD)	- ATT
batch(AU_CMD)	- ATT
cancel(AU_CMD)	- ATT
chgrp(AU_CMD)	- ATT
chown(AU_CMD)	- ATT
cron(AU_CMD)	- ATT
crontab(AU_CMD)	- ATT
csplit(AU_CMD)	- ATT
cu(AU_CMD)	- ATT
dd(AU_CMD)	- ATT
dircmp(AU_CMD)	- ATT
egrep(AU_CMD)	- ATT
ex(AU_CMD)	- ATT
fgrep(AU_CMD)	- ATT
id(AU_CMD)	- ATT
join(AU_CMD)	- ATT
logname(AU_CMD)	- ATT
lp(AU_CMD)	- ATT
lpstat(AU_CMD)	- ATT
mailx(AU_CMD)	- ATT
mesg(AU_CMD)	- ATT
newgrp(AU_CMD)	- ATT
news(AU_CMD)	- ATT
od(AU_CMD)	- ATT
passwd(AU_CMD)	- ATT
shl(AU_CMD)	- NOPLAN (doesn't run under streams)

stty(AU_CMD)	- ATT
su(AU_CMD)	- ATT
tabs(AU_CMD)	- PLANNED
tar(AU_CMD)	- MIPS (from BSD)
tty(AU_CMD)	- ATT
uucp(AU_CMD)	- ATT
uulog(AU_CMD)	- ATT
uuname(AU_CMD)	- ATT
uupick(AU_CMD)	- ATT
uustat(AU_CMD)	- ATT
uuto(AU_CMD)	- ATT
uux(AU_CMD)	- ATT
vi(AU_CMD)	- ATT
wall(AU_CMD)	- ATT
who(AU_CMD)	- ATT
write(AU_CMD)	- ATT

AS_CMD - Administered Systems Extension.

acctcms(AS_CMD)	- ATT
acctcom(AS_CMD)	- ATT
acctcon1(AS_CMD)	- ATT
acctcon2(AS_CMD)	- ATT
acctdisk(AS_CMD)	- ATT
acctmerg(AS_CMD)	- ATT
accton(AS_CMD)	- ATT
acctprc1(AS_CMD)	- ATT
acctprc2(AS_CMD)	- ATT
acctwtmp(AS_CMD)	- ATT
chargefee(AS_CMD)	- ATT
ckpacct(AS_CMD)	- ATT
clri(AS_CMD)	- ATT
devnm(AS_CMD)	- ATT
diskusg(AS_CMD)	- ATT
dodisk(AS_CMD)	- ATT
fsck(AS_CMD)	- ATT
fsdb(AS_CMD)	- ATT
fuser(AS_CMD)	- ATT
fwtmp(AS_CMD)	- ATT
grpck(AS_CMD)	- ATT
init(AS_CMD)	- ATT
ipcrm(AS_CMD)	- ATT
ipcs(AS_CMD)	- ATT
killall(AS_CMD)	- ATT
labelit(AS_CMD)	- ATT
lastlogin(AS_CMD)	- ATT
link(AS_CMD)	- ATT
mkfs(AS_CMD)	- ATT (See note 1 above)
mknod(AS_CMD)	- ATT
monacct(AS_CMD)	- ATT
mount(AS_CMD)	- MIPS (from SUN NFS)
mvdir(AS_CMD)	- ATT

ncheck(AS_CMD)   - ATT (See note 1 above)
nice(AS_CMD)   - ATT
prctmp(AS_CMD)   - ATT
prdaily(AS_CMD)   - ATT
prtacct(AS_CMD)   - ATT
pwck(AS_CMD)   - ATT
runacct(AS_CMD)   - ATT
sa1(AS_CMD)   - ATT
sa2(AS_CMD)   - ATT
sadc(AS_CMD)   - ATT
sadp(AS_CMD)   - PLANNED
sar(AS_CMD)   - ATT
setmnt(AS_CMD)   - NOPLAN (Not required with new NFS mtab format)
shutacct(AS_CMD)   - ATT
startup(AS_CMD)   - ATT
sync(AS_CMD)   - ATT
sysdef(AS_CMD)   - NOPLAN
timex(AS_CMD)   - ATT
turnacct(AS_CMD)   - ATT
umount(AS_CMD)   - MIPS (from SUN NFS)
unlink(AS_CMD)   - ATT
volcopy(AS_CMD)   - PLANNED
whodo(AS_CMD)   - ATT
wtmpfix(AS_CMD)   - ATT

SD_CMD - Software Development Extension.

admin(SD_CMD)   - ATT
as(SD_CMD)   - MIPS
cc(SD_CMD)   - MIPS
cflow(SD_CMD)   - ATT
chroot(SD_CMD)   - ATT
cpp(SD_CMD)   - MIPS
cxref(SD_CMD)   - ATT
delta(SD_CMD)   - ATT
dis(SD_CMD)   - MIPS
env(SD_CMD)   - ATT
get(SD_CMD)   - ATT
ld(SD_CMD)   - MIPS
lex(SD_CMD)   - ATT
lint(SD_CMD)   - MIPS
lorder(SD_CMD)   - ATT
m4(SD_CMD)   - ATT
make(SD_CMD)   - ATT
nm(SD_CMD)   - MIPS
prof(SD_CMD)   - MIPS
prs(SD_CMD)   - ATT
rmdel(SD_CMD)   - ATT
sact(SD_CMD)   - ATT
sdb(SD_CMD)   - NOPLAN (use dbx)
size(SD_CMD)   - MIPS
strip(SD_CMD)   - MIPS

```
time(SD_CMD)     - ATT
tsort(SD_CMD)    - ATT
unget(SD_CMD)    - ATT
val(SD_CMD)      - ATT
what(SD_CMD)     - ATT
xargs(SD_CMD)    - ATT
yacc(SD_CMD)     - ATT
```

TI_CMD - Terminal Interface Extension.

```
tic(TI_CMD)      - ATT
tput(TI_CMD)     - ATT
```

Network File System (NFS)- Networking Utilities and Daemons

```
biod(EXTRA)
domainname(EXTRA)     - MIPS (from SUN NFS)
havenfs(EXTRA)        - MIPS (from SUN NFS)
nfsd(EXTRA)
nfsstat(EXTRA)
portmap(EXTRA)        - MIPS (from SUN NFS)
rpc.mountd(EXTRA)     - MIPS (from SUN NFS)
rpc.passwd(EXTRA)     - MIPS (from SUN NFS)
rpc.ruserd(EXTRA)     - MIPS (from SUN NFS)
rpc.rwalld(EXTRA)     - MIPS (from SUN NFS)
rpc.sprayd(EXTRA)     - MIPS (from SUN NFS)
rpcinfo(EXTRA)
showmount(EXTRA)      - MIPS (from SUN NFS)
spray(EXTRA)
```

5.10 SVVS Results

5.10.1 BA_ENV

All tests in this section passed.

5.11 BA_OS

All but one test (>95%) pass; the test that fails does so because of a problem with System V struct nfiles versus stdio nfiles, and therefore not ready for this release.

RISC/os is currently in transition between two different methods of determining the number of open files per process. The new method is described in the kernel lboot configuration file, *master.d/kernel*:

```
* Maximum number of open files per process.  Now a configurable
* parameter but only applies to direct open and close calls (i.e
* stdio, sh, etc will not make use of more than 20 in this release).

NOFILES = 100
```

A new option has been added to the *ulimit* call to allow users to determine the number of open files allowable on the current system. When called with a command argument of 4, the ulimit call returns the current value of v.v_nofile.

nofiles = ulimit(4, 0);

The file */usr/include/sys/param.h*, #defines NOFILE as the minimum value which the kernel can be set, so that a process can always create at least NOFILE files. See */usr/include/sys/param.h* for more information.

5.12 BA_LIB

100% pass non-Floating Point tests. The Floating Point tests have numerous failures, most of which we believe to be because MIPS returns IEEE numbers and the test expects SVID values.

5.13 KE_ENV

All tests in this section passed.

5.14 KE_OS

All tests in this section passed.

5.15 TI_ENV

5.16 TI_LIB

MIPS has not yet received a bug-free version of the TI_* section tests from AT&T. Therefore, MIPS will not provide test results for these sections at this time.

6. Appendix B: Preserve Files

The default preserve list for RISC/os 4.50 is as follows:

```
#
#   root.preserves - RISC/os 4.50
#
#   This preserve list is fed to preserve following the list generated
#   by findmods.
#


#
#   local files which should be saved "as-is":
#
etc/TIMEZONE    -k
etc/TZ          -k
etc/bupsched    -k
etc/cshrc    -k
etc/coredirs    -k
etc/disktab -k
etc/dumpdates   -k
etc/exports -k
etc/fstab       -k
etc/gettydefs   -k
etc/group       -k
etc/hosts       -k
etc/hosts.equiv     -k
etc/inetd.conf -k
etc/init.d/ANNOUNCE    -k
etc/init.d/MOUNTFSYS -k
etc/init.d/README -k
etc/init.d/RMTMPFILES     -k
etc/init.d/addswap     -k
etc/init.d/autoconfig    -k
etc/init.d/bfsd     -k
etc/init.d/cron     -k
etc/init.d/disks    -k
etc/init.d/firstcheck    -k
etc/init.d/lockfix     -k
etc/init.d/netdaemons    -k
etc/init.d/nfs -k
etc/init.d/perf    -k
etc/init.d/savecore    -k
etc/init.d/set_kopts -k
etc/init.d/smtp    -k
etc/init.d/sysetup     -k
etc/init.d/syslog -k
etc/init.d/tcp -k
etc/init.d/tzsetup     -k
etc/init.d/uucp    -k
etc/inittab -k
etc/local_hostname     -k
etc/motd -k
etc/networks    -k
etc/passwd    -k
```

```
etc/profile -k
etc/protocols  -k
etc/preserves  -k
etc/remote  -k
etc/services    -k
etc/sysdirlist -k
etc/syslist -k
etc/syslog.conf    -k
etc/termcap -k
etc/timed.conf -k
etc/ttytype -k
tftpboot -k
tmp        -k

#
#  force use of the following update files:
#
etc/magic    -u

#
#  remove these files:
#
etc/checklist   -d

#
#  end root.preserves
#




#
#  usr.preserves - RISC/os 4.50
#
#  This preserve list is fed to preserve following the list generated
#  by findmods.
#

#
#  local files which should be saved "as-is":
#
usr/adm/crash   -k
usr/etc/exports    -k
usr/etc/bootptab -k
usr/etc/timed.conf    -k
usr/lib/aliases    -k
usr/lib/sendmail.cf  -k
usr/lib/me/local.me  -k
usr/tmp         -k

#
#  uucp config
#
usr/lib/uucp/Devconfig  -k
```

```
usr/lib/uucp/Devices -k
usr/lib/uucp/Dialcodes  -k
usr/lib/uucp/Dialers -k
usr/lib/uucp/Maxuuscheds   -k
usr/lib/uucp/Maxuuxqts  -k
usr/lib/uucp/Permissions   -k
usr/lib/uucp/Poll -k
usr/lib/uucp/Sysfiles   -k
usr/lib/uucp/Systems -k
usr/lib/uucp/Uutry    -k

usr/spool/cron/crontabs/adm    -k
usr/spool/cron/crontabs/periodic -k
usr/spool/cron/crontabs/root   -k
usr/spool/cron/crontabs/sys    -k
usr/spool/cron/crontabs/sysadm   -k
usr/spool/cron/crontabs/uucp   -k

usr/spool/lp/pstatus -k
usr/spool/lp/qstatus -k

#
#   force use of the following update files:
#
usr/lib/crontab    -u

#
#   end of usr.preserves
#
```

7. Appendix C: Numbered Bug Fixes

- #512
  csh now knows how to glob symbolic links.

- #829
  cxref now sets -Dmips for its cpp pass.

- #1147
  awk is fixed so that split and substr now work together.

- #1373
  cxref now supports the C extensions *signed* and *volatile*.

- #1774
  dvhtool -p has been removed, since it was never supported.

- #2281
  usleep(3) routine no longer missing from *libc.a*.

- #2429
  tail -0f now works correctly.

- #2473
  sysadm backup command now uses dump.ffs and restore.ffs.

- #2699
  fsck can now determine the filesystem type for */dev/usr*.

- #2859
  fsck now properly determines filesystem type.

- #2997
  cxref now prints a usage message when there is no file listed as an argument.

- #2998
  cxref now prints a usage message when the required arguments are not supplied.

- #2999
  cxref no longer dumps core when run in background with an invalid option.

- #3024
  cxref now checks file types and processes only C files.

- #3036
  netstat -f now prints a usage message instead of dumping core.

- #3037
  netstat -I now prints usage message when no argument is given.

- #3115
  The number of jobs cron can run is now changeable at system startup time. The default is still 25, but you can use the option maxrun to increase this limit.

- #3171
  rcp -p option now works correctly.

- #3190
  There now exists a lint library for *libbsd.a*.

- #3505, #4888
  /usr/adm/periodic/monthly/10.wtmp.system now rolls the correct *wtmp* file.

- #3508
  There now exists a */usr/include/bsd/sys/timeb.h*.

- #3569
  There now exists a default no-rewind tape device in /usr/include/sys/mtio.h.

- #3585
  timed code has been added to /etc/init.d/netdaemons.

- #3613
  csh newgrp command now functions properly.

- #3662
  Freeing the original pointer after a **malloc** with a negative value no longer core dumps.

- #3663
  Freeing the original pointer after a **realloc**(max_int) no longer core dumps.

- #3664
  calloc now works with a negative element size.

- #3665
  Freeing the original pointer after a **realloc** with a negative size no longer core dumps.

- #3666
  malloc(max_int) no longer produces a segmentation violation core dump.

- #3723
  crypt -k no longer dumps core.

- #3752 **mallopt (libmalloc)** works correctly.

- #3753
  calloc with an element size of 1 no longer core dumps when the original pointer is freed.

- #3779 RISC/os is no longer missing fpi.h.

- #3798
  realloc(max_int) no longer core dumps when the original pointer is freed.

- #4081
  Invalid **sockaddr** sockaddr length values are no longer allowed.

- #4215
  The callout table no longer gets incorrectly ordered.

- #4216, #5757
  Dumpsize is not calculated so that it is less than or equal to the size of the dump device and a multiple of 4k and a multiple of pagesize, so that if **savecore** doesn't read a partial block at the end of the dump device and get an error.

- #4220
  date no longer prints time in GMT when the system is in single-user mode.

- #4224
  The default partition for **dvhtool** is now 'ipc' (it was 'ips').

- #4225 **dvhtool** now writes out the volume header only if it was read in and modified.

- #4226
  Error checking was added to **dvhtool**.

- #4266, #4898
  ecvt() and fcvt() no longer return wrong string length.

- #4272
  diff3 no longer dumps core.

112

- #4276
  Since RISC/os 4.50 contains NFS 4.0 and its **fsck.ffs**, we now support parallel file system checking.

- #4282
  libc.a is no longer missing *vprintf* and *vsprint*.

- #4287, #6239, #6300
  whereis now searches in more directories, specifically */bsd43/bin*.

- #4291
  vprintf no longer missing from *libc.a*.

- #4332
  ptrace() works better with regard to SIGCHLD.

- #4335, #6013
  rsh now correctly handles file descriptors greater than 16.

- #4354
  rcp -r now works when source and destination are local.

- #4369
  plot(1G) is no longer missing.

- #4392
  Prototypes of startup and shutdown scripts for System V accounting, 4.3 BSD accounting, System V **lpsched**, and 4.3 BSD **lpd** have been added to the system.

- #4496, #5311
  Fixed an NFS problem where files on DECstation 3100 and SPARCstation1 systems served by MIPS servers would get 7777 permissions and bad owner/group.

- #4566
  /usr/include/bsd43/sys/gprof.h and its associated man page have been removed from the system.

- #4620
  csh now echoes the first character when doing filename expansion.

- #4679
  SIGPIPE error fixed.

- #4696
  getty no longer respawns at incorrectly.

- #4708
  csh now understands line disciplines in file completion mode.

- #4726
  A filename with more than NAME_MAX characters is no longer unseen by **ls**, **file**, or **cat**.

- #4738
  gated.conf has been renamed to *gated.conf.sample* so that **gated** is no longer started instead of **routed** at system boot by default.

- #4740
  The */etc/init.d/netdaemons* file now reads the file */etc/routed.conf* and inserts the given options so that **routed** no longer starts in noisy mode by default.

- #4745
  The **lp** spooler no longer truncates files.

- #4756
  The ISI ICP parallel port driver no longer drops characters.

- #4760. #4764
  lseek done on a socket no longer generates EISPIPE error.

- #4777, #4809, #4810
  The *flock*(2-BSD) system call was fixed to return EWOULDBLOCK rather than EACCES when a process would have to block in order to create the requested lock and the LOCK_NB flags was specified. The *flock*(2-BSD) system call was fixed so that if two calls are made to create an exclusive lock on the same file, the second call blocks or returns EWOULDBLOCK, depending on whether LOCK_NB was specified. The second call was incorrectly succeeding in RISC/os 4.00.

- #4786
  timed now reads */etc/timed.conf*.

- #4791
  telnet now quits properly when the login fails.

- #4797
  Signal handler now correctly ignores a specified signal.

- #4816, #4817, #5717
  date no longer incorrectly sets the time because of time zone problems.

- #4841
  RISC/OS operates a little differently than BSD in the **bcheckrc** area. Any filesystem which is shutdown cleanly flushes a clean bit to disk such that on bring-up, it need not be **fsck**'ed. This includes */dev/root*. Therefore, it is normal operation to have **fsstat** return a code indicating that */dev/root* needs no **fsck**'ing.

- #4843
  getty has been fixed so that hard wired terminals no longer run into the problem of not being able to read the *passwd* file.

- #4848
  /etc/init.d/nfs no longer prints an error message when changing from single-user state to init 0.

- #4856
  /etc/initd.nfs no longer prints an error message regarding **kill** and an incorrect number of arguments.

- #4858
  Fixed SCSI tape handling problems, especially non-standard record sizes and end of media.

- #4859
  The built-in **echo** in the **csh** now understands C-like escape conventions.

- #4865
  su now sets HOME and SHELL environment variables when executing the target's shell so that the *.cshrc* file is sourced.

- #4874
  getty now supports LDISC2.

- #4890
  /etc/init.d/nfs no longer incorrectly skips comments in the */etc/fstab* file.

- #4891
  The *ftp*(1C) no longer gives erroneous information about file ownership.

114

- #4894
  There is no longer a problem with the kernel being unable to allocate more memory because *availsmem* goes negative.

- #4897
  4.3 BSD *gettimeofday*(2) and *ftime*(3) now understand about timezone.

- #4902
  lpd now uses */etc/hosts.lpd*.

- #4920, #6261
  Anonymous file transfer, **tftp**, has been made more restrictive for security purposes.

- #4925
  malloc no longer leaks memory.

- #4940
  exit() no longer hangs uninterruptibly while closing a serial port.

- #4942
  tail now has a 1 megabyte buffer.

- #4972
  rlogind and **telnetd** no longer leave file descriptors open.

- #4977, #5039
  /bsd43/bin/at now has correct permissions (555 owned by user root, group bin).

- #4982, #6216, #6270, #6354, #6470
  When large amounts of data are sent to tty ports, the system no longer deadlocks.

- #4983
  The *10.clean.system* scheme has been changed and now deletes directories in */usr/preserve* that are older than 7 days. Because the date on a directory is updated when the files in that directory are removed, it may take two weeks for directories to be removed.

- #4987, #4988, #4996, #5018, #5028, #5260, #5322
  Many miscellaneous **telnet** problems were fixed among them: **telnet** now tolerates control-d; **telnet** now handles the **quit** command properly; **telnet** no longer gives incorrect "connection refused" message; and **telnet** now allows one to quit after using the **open** command.

- #4997
  vi macro now works with map command input.

- #5010
  /usr/ucb/mail now saves message to *dead.letter* if control-c is entered at the Cc: prompt.

- #5022
  man -k now prints correct page headings.

- #5023
  csh no longer core dumps with backquote substitution within a backquote substitution.

- #5040, #5049
  login correctly sets user's tty permissions.

- #5065
  The *tty* field in the **finger** display has been expanded to six characters.

- #5080
  /etc/stdcshrc no longer causes **rcp** to produce the message `stty:: Can't assign requested address`.

115

- #5100
  /etc/init.d no longer fails to start **timed**.

- #5182, #5201. #5338
  Various **awk** bugs fixed: **awk** now correctly parses "&&"; now recognizes right curly brace as an end of statement; and can now handle multiple "||" expressions.

- #5193
  rcp and **rsh** now support fully qualified domain hostnames.

- #5195, #5296
  Various *libcurses* problems were fixed.

- #5198
  /etc/stdcshrc no longer causes problems for Rx2030 systems.

- #5204
  The *inst*(1M) man page has been updated.

- #5298
  It is no longer possible for a user to reduce a processes priority, unless the user has super-user privileges.

- #5299
  select support was added to the /dev/tty driver so that it no longer times out.

- #5335
  sar now works on M/120 systems.

- #5353, #5388
  telnet no longer drops characters (like line feeds and the escape character) when connected to a VMS machine.

- #5380 **mailx** has been modified to create a mail file and set its permissions in the same manner as the BSD mailer.

- #5384, #5823
  There is now a man page for **sag**(1G); and **graph** and **tplot** live in /usr/bin with symlinks to /bsd43/bin/graph.

- #5385
  The spelling history file /usr/lib/spell/spellhist is not automatically created on a newly installed system. The **spell** command has been fixed so that it does not exit with an error if the file does not exist. Instead, it sends the spelling history information to /dev/null. System Administrators may want to create this file and monitor it for locally used acronyms and spelling, and add these to the spell database. This file will grow without bounds and should periodically be pruned.

- #5400
  syslogd now works as advertised.

- #5410
  apropos now works correctly.

- #5418
  cxref no longer fails when its define table overflows.

- #5430
  sar now works on non-SMD based systems.

- #5431
  sar -s now works correctly.

- #5499
  yacc now has larger table size capacity.

- #5504
  Specifying an alternative root device on RC3260 systems now works correctly.

- #5519
  spray -l now works correctly.

- #5520
  dump no longer states that the no rewind device is being rewound (it never actually rewound the tape).

- #5521
  rcp now works correctly with files that do not have write permission.

- #5523
  The system no longer crashes intermittently when a non-existent filesystem is mounted.

- #5527
  The **pac** command was ported.

- #5532
  syslog no longer core dumps when given a long string.

- #5552
  Various problems with the communications driver have been fixed.

- #5563
  mboot now correctly generates the number of controllers per major number.

- #5647
  ps -l now works correctly.

- #5702
  Erroneous error messages are no longer displayed when booting a SCSI based system.

- #5718
  /bsd43/bin/date no longer complains about */usr/adm/wtmp*.

- #5733 */usr/spool/cron/crontabs/root* no longer contains empty lines that caused a syntax error.

- #5737
  prtvtoc now displays */usr* partition correctly.

- #5751
  Typing ctrl-z in **vi** while running under **sh** in an **rlogin** session no longer logs out the user.

- #5759
  csh now functions correctly when LOGNAME is not set.

- #5780
  getpw() routine in *libc* now uses VIS and works with NIS.

- #5797
  mount now understands how to mount */proc*.

- #5808
  The dkip driver can now handle dumps on a partition other than just partition 1.

- #5817
  MBUFs are no longer allocated outside the 16 megabytes physical memory.

- #5819
rcp to /dev/null no longer crashes the system.

- #5842
There is no longer a problem with restarting the read(), write(), or ioctl() system calls.

- #5843
syslog.conf "*.foo" entries must come first in the file or following entries will be deleted. The default syslog.conf file was fixed.

- #5881
The sduart driver no longer sets the default baud rate to whatever speed the port was at when last set.

- #5888
The master side of a pty now notices when the slave side does a close.

- #5902
Ported some missing programs for **refer** that are installed in /bsd43/usr/lib/refer and **indxbib** now uses these new programs.

- #5912, #5921
setregid and **setreuid** now correctly set group and user id.

- #5916
The **more**(1) man page now explains that the -b option is not supported for reading stdin in a pipeline.

- #5943
cpio -pdl now correctly copies files when links fail across file systems.

- #5957
Added improved checking for attempts to edit an executable.

- #5961
portmapper no longer dies and leaves core dump.

- #5991
SIGHUP to **inetd** no longer causes the system to disconnect from the network.

- #6075
cron now calls **initgroups** if the file /etc/login.initgroups.ok exists so that scripts run by **cron** know about multiple groups.

- #6093
Ethernet controllers are now initialized even if TCP/IP is not started (as in the case of running RISComm/DN on a controller).

- #6121
access() now checks for valid argument mode.

- #6126
cpio no longer calls utime for symbolic links since it changes the time on the target file, not the link.

- #6128
The dkvj driver now enables parity.

- #6172
telnetd now displays the file /etc/banner, if it exists, instead of using an internal banner.

- #6177
The dates in syslog are no longer reported in GMT.

118

- #6188
  gcore no longer missing.

- #6191, #6240
  man now prints the man page for a command before the library routine, in cases where there are man pages in multiple sections with the same name. **man** also no longer gets confused by the **man 1 uadmin** syntax.

- #6206
  /usr/sbin was removed from PATH in root's *.profile* and *.cshrc* files. #6230
  XON/XOFF handling is now done properly with respect to IXANY, so there is no longer data loss on peripherals who send consecutive XOFF's while IXANY is set.

- #6237
  VMSbackup is now able to recover large files.

- #6242
  The **askcc** option to **/usr/ucb/Mail** now works with the dot option (no longer requires a control-d).

- #6246
  restore now understands FIFO's.

- #6252 **/bsd43/bin/ps -t** now selects terminals properly.

- #6266
  mailx no longer loses mail when a new message arrives at the time two instances of mailx are running in two different windows.

- #6275
  lp spooler configuration files were added to the *usr.preserves* list so that they no longer get trashed during an update installation.

- #6278
  The rewind timeout in the SCSI tape driver has been increased.

- #6288
  /bsd43/bin/kill no longer sends the wrong signals.

- #6306
  /bsd43/bin/at now correctly runs queued jobs.

- #6356
  acctmerg is no longer limited to hnadling 10 pacct files on the command line.

- #6358
  A_SSIZE has been increased from 1000 to 3001 for **sar**.

- #6359
  ucmp now makes proper comparisons.

- #6360
  acctcom -n no longer dumps core.

- #6361
  The **sar** accounting is no longer incorrect if the user does not logout at the end of a day.

- #6379
  addswap now works on RC3260 systems.

- #6428
  cp /dev/rmt/ctape0 /dev/null no longer causes a SEGV panic.

- #6513
  Various programs that write to terminals are not setgid to tty.

- #6519
  A typographical error was fixed on the *passwd*(1) man page.

- #6567
  ftp help no longer references an alias (ftp-bugs) that does not exist.

- #6645
  The 4.3 BSD **roff** man pages have been added.