NOTICE
The information in this document is subject to change without notice. AT&T assumes no responsibility for any errors that may appear in this document.

Hayes is a registered trademark of Hayes Microcomputer Products Inc.
IBM is a registered trademark of International Business Machines Corporation.
UNIX is a registered trademark of AT&T.
AT&T Products and Services

To order documents from the Customer Information Center:

• Within the continental United States, call 1-800-432-6600
• Outside the continental United States, call 1-317-352-8556
• Send mail orders to:
  AT&T Customer Information Center
  Customer Service Representative
  P.O. Box 19901
  Indianapolis, Indiana  46219

To sign up for UNIX system or AT&T computer courses:

• Within the continental United States, call 1-800-221-1647
• Outside the continental United States, call 1-609-639-4458

To contact marketing representatives about AT&T computer hardware products and UNIX software products:

• Within the continental United States, call 1-800-372-2447
• Outside the continental United States, call collect 1-215-266-2973 or 1-215-266-2975
To find out about UNIX system source licenses:

- Within the continental United States, except North Carolina, call 1-800-828-UNIX

- In North Carolina and outside the continental United States, call 1-919-279-3666

- Or write to:
  Software Licensing
  Guilford Center
  P.O. Box 25000
  Greensboro, NC 27420
# Table of Contents

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Preface</td>
<td>1</td>
</tr>
<tr>
<td>Features Of 386 UNIX System V</td>
<td>3</td>
</tr>
<tr>
<td>286 Binary Compatibility</td>
<td>13</td>
</tr>
<tr>
<td>Floating Point Co-processor Support</td>
<td>15</td>
</tr>
<tr>
<td>Boot Procedures</td>
<td>16</td>
</tr>
<tr>
<td>386 UNIX System Base System Devices</td>
<td>17</td>
</tr>
<tr>
<td>Shutdown Procedures</td>
<td>19</td>
</tr>
<tr>
<td>Instructions for Updating Selected Files from the Release</td>
<td>20</td>
</tr>
<tr>
<td>Uucp Notes</td>
<td>21</td>
</tr>
<tr>
<td>Software Notes</td>
<td>22</td>
</tr>
<tr>
<td>Security Notes</td>
<td>62</td>
</tr>
<tr>
<td>Internationalization Software Notes</td>
<td>69</td>
</tr>
<tr>
<td>Future Directions</td>
<td>76</td>
</tr>
<tr>
<td>Documentation</td>
<td>78</td>
</tr>
<tr>
<td>Remote File Sharing Notes</td>
<td>79</td>
</tr>
<tr>
<td>Network Support Utilities Notes</td>
<td>91</td>
</tr>
<tr>
<td>Appendix A: Installation Diskette Files</td>
<td>A-1</td>
</tr>
</tbody>
</table>
Preface

These Release Notes contain important information about 386 UNIX System V Release 3.1 on your computers. First, they briefly describe the new features of this release. Software notes, additional information about installation, and compatibility notes are also provided. Second, the Remote File Sharing and Network Support Utilities software packages are discussed. Finally, presented in appendix B, are last minute change pages for the Operations/System Administration Guide and the User's/System Administrator's Reference Manual.

For a comprehensive description of the software and documentation available for 386 UNIX System V Release 3.1 for your computers, see the Product Overview. For a complete description of the procedures used in the administration of your computer running 386 UNIX System V Release 3.1, see the Operations/System Administration Guide.

Conventions Used in These Release Notes

In this document certain typesetting conventions are followed when command names, command line formats, files, and directory names are described. File names, directory names, path names, and references to other documents are shown in italic font. Bold font is used to indicate command names and variable names.

There are also conventions for displays of terminal input and output. The following apply to displays.

- You must type words that are in bold font as they appear.
- Italic words are variables; you substitute the appropriate values. These values may be file names or they may be data values, as applicable.
- CRT or terminal output and examples of source code are presented in constant-width font.
- Characters or words in square brackets, [ ], are optional. (Do not type the brackets.)
A command name followed by a number, for example, ed(1), refers to that
command’s manual page, where the number refers to the section of the
manual. Manual pages from sections (2) and (3) appear in the Programmer’s
Reference Manual. Manual pages from sections (1), (1M), (4), (5), and (7)

Examples in these Release Notes show the default system prompt for UNIX
System V, the dollar sign ($). They also show the default prompt when you
log in as the superuser, the pound sign (#).
Features Of 386 UNIX System V

UNIX System V Release 3.1

UNIX System V Release 3.1 provides the following new features. For a more detailed description of them, see the Product Overview.

- Internationalization
  - Support for 8-bit Code Sets
  - Support for Alternate Date and Time Formats [see cftime(4) and environ (5) in the User's/System Administrator's Reference Manual]
  - Support for Alternate Character Classification and Conversion Rules [see chrtbl(1M) in the User's/System Administrator's Reference Manual]

- New awk (nawk)

- Performance Improvements
  - Paging the User-Area
  - Remote File Sharing Client Caching
  - Faster and Smaller curses

- Additional Features
  - Remote File Sharing Loop-Back
  - Improved Recovery of Files from cpio Archives.

Support for 8-bit Code Sets

The cat, ed, egrep, expr, find, grep, ls, pg, sed, sort, and vi commands and the curses library were changed so they no longer use the eighth bit of each byte. This change enables these commands to handle code sets where all 8 bits are used. (nawk provides this capability for awk users.) Because ASCII only uses 7 of the available 8 bits in a byte, some commands made special use of this eighth bit; other commands assumed that if the bit was set, the byte was invalid. Other commonly-used commands, such as sh (the shell), already support 8-bit code sets.
In addition, 8-bit characters can be sent between a terminal and the system [see the init(1M) and gettydefs(4) manual pages in the User's/System Administrator's Reference Manual].

**Support for Alternate Date and Time Formats**

The cpio, date, ls, mount, pr, and sort commands were changed to provide the date and time in the language and format determined by the value of the LANGUAGE environment variable. While the United States conventions remain the default, other languages can be supported by creating and installing a file for the language desired in the /lib/cftime directory. The content of that file includes: month and weekday names (full and abbreviated), default local time, date, pre-noon, and post-noon formats, and the default output of the date command if the CFTIME environment variable is not set. In addition, time zones and alternate time zones (such as daylight time) now can be defined in terms of hours and minutes using the TZ environment variable.

For more information, see the date(1), timezone(4) and environ(5) manual pages in the User's/System Administrator's Reference Manual.

**Support for Alternate Character Classification and Conversion Rules**

cat, ed, egrep, grep, ls, pg, sed, sort, and vi (commands that convert characters from upper to lower case or classify characters as alphabetic, printable, uppercase or lowercase, etc.), were changed to support code sets or classification rules according to the value of the CHRCLASS environment variable. While ASCII remains the default for these operations, other conversion and classification rules are supported by creating and installing a file describing these rules in the /lib/chrclass directory. A new administrative command, chrtbl, is used to create this file.

For more information, see the chrtbl(1M) and environ(5) manual pages in the User's/System Administrator's Reference Manual.

**New awk (nawk)**

awk is a programming language for information retrieval and data manipulation that is often used by people with no programming background. Customers using awk in an international environment must use the new version (nawk), because the old version does not support 8-bit code sets. Some of the other enhancements to this version of awk include:
Features Of 386 UNIX System V

- The ability to define functions
- New keywords: delete, do, func, function, return
- New built-in functions: atan2, cos, sin, rand, srand, gsub, sub, match, close, and system
- New pre-defined variables: FNR, ARGV, RSTART, and RLENGTH
- The input field-separator variable, FS, and the third argument to split are treated as regular expressions
- The precedence of operations now matches C Language precedence.

Because some of these enhancements may not be compatible with some existing awk programs, you will get the old version when you type awk. To take advantage of these new features, you must use the nawk command. For more information, see the "awk Tutorial" packaged within the Programmer's Guide, Vol. 1, and the "Software Notes," and "Future Directions" sections of these Release Notes.

Paging the User Area

The user area for each process (approximately 8192 bytes) can now be paged, like the rest of each process. This enhancement significantly improves the performance of memory-limited systems, such as 2-Mbyte configurations, by making more memory available.

Remote File Sharing Client Caching

This feature enables a client system (one that is accessing data from another system) in an RFS arrangement to maintain a local copy of the data it needs. This means that when a block of data is read from, or written to, a remote system, it is placed in a local buffer where it can be accessed by subsequent requests for data by local processes. This can significantly reduce the amount of data that needs to be sent across the network, resulting in significant performance improvements for many patterns of remote file use. For example, application programs that access data from a small set of files, such as data base systems, will benefit from this feature.

Mechanisms are built into the client caching feature to ensure file consistency across the network when multiple machines have opened a file and when a process re-opens a previously closed cached file.
By default, client caching is enabled for all systems running Release 1.1 of the Remote File Sharing Utilities; however, caching can be turned off for a particular resource [see the `mount(1M)` manual page in the User's/System Administrator's Reference Manual, or for an entire system see the discussion of tunable parameters in the new version of the "Remote File Sharing" chapter of the Operations/System Administration Guide.

Administrative tools have been added to monitor caching activities. Using this information, administrators can modify several new tunable parameters to maximize caching efficiency.

This feature is provided by Release 1.1 of the Remote File Sharing Utilities. For more information, see the "Remote File Sharing" procedures and chapter in the Operations/System Administration Guide.

Remote File Sharing Loop-Back

The loop-back feature enables you to simulate the higher levels of Remote File Sharing (RFS) functionality within one computer. Application programs designed to use RFS can be partially tested without actually communicating with a remote computer. This feature can also be used to demonstrate RFS when only one computer is available.

This feature is provided by Release 1.1 of the Remote File Sharing Utilities. For information describing how to use this feature, see the `-o` option on the `rfadmin(1M)` manual page in the User's/System Administrator's Reference Manual.

Improved Recovery of Files From cpio Archives

If errors are encountered while restoring a file from floppy disk using the "Restore From Removable Media" function (described in the Operations/System Administration Guide) or `cpio`, you can now skip over the bad blocks and continue the restore with the next file. This procedure will automatically be invoked when you use the "Restore From Removable Media" function or use the `-k` option described on the `cpio(1)` manual page in the User's/System Administrator's Reference Manual.
UNIX System V Release 3.0

Following is a list of new features provided in UNIX System V Release 3.0 that are also included with Release 3.1. For more information see the Product Overview.

- Remote File Sharing
- Network Support Utilities
- Enhanced Basic Networking Utilities
- Shared Libraries
- Command Syntax Standard
- Signal Mechanism Enhancements
- Improved Terminal Support Facilities
  - Remote Terminal Package Enhancements
  - AT&T Windowing Utilities Package (included in Base System)
  - 80286 Binary Compatibility
  - Floating Point Co-processor Support
- Additional Features
  - `crash(1M)` Command Changes
  - New System Header Files
  - Encryption Mechanisms Repackaged
Remote File Sharing

To take advantage of Remote File Sharing, you must have Remote File Sharing Utilities, the Network Support Utilities, and a transport provider. (See "Network Support Utilities" for a description of a transport provider.)

Remote File Sharing lets you share files, directories, devices, and named pipes transparently among computers that are linked by a network. Files are shared transparently by mounting a remote directory as one would mount a file system. Each computer on the network controls which local resources are available to other computers and which remote resources local users can access. For example, with Remote File Sharing you might share a directory among several departments of your business, or a letter-quality printer or typesetter that no one department could support by itself. For more information, see the Operations/System Administration Guide.

Network Support Utilities

The Network Support Utilities provide STREAMS tools, the AT&T Transport Interface, and the Listener. STREAMS is a set of tools for development of communication and networking services within the UNIX System; the Transport Interface is based on the Transport Service Definition (Level 4) of the International Standards Organization (ISO) Open Systems Interconnection (OSI) reference model, and defines how a user accesses the services of a transport protocol. The Listener receives requests for network services from another system, interprets which network service is needed, and starts the appropriate process that has been named to provide the requested network service. The Listener then drops out of the communications path and continues to listen for new service requests.

For more information about the Network Support Utilities and the Listener, read these Release Notes and the Operations/System Administration Guide.
Enhanced Basic Networking Commands

NOTE Media-independent basic networking commands are provided in the Basic Networking Utilities package, but they require the installation of the Network Support Utilities package.

Basic networking commands [for example, uucp(1C) and uux(1C)] have been enhanced to conform to the AT&T Transport Interface. These utilities can communicate using any Transport Provider that conforms to the AT&T Transport Interface. The operation of the Basic Networking Utilities commands is the same regardless of whether or not you install the Network Support Utilities product.

The Network Support Utilities can show which, if any, Transport Providers are available with nlsadmin -x. If you install additional Transport Providers, no changes are needed to the software to accommodate the underlying media or protocols; you need only register Basic Networking Utilities services with the network Listener for those Transport Providers, and register the Transport Providers in the administrative files for the Basic Networking Utilities [see nlsadmin(1M) in the User's/System Administrator's Reference Manual].

Even with these enhancements, the syntax of the basic networking commands is no different from before. See the Operations/System Administration Guide for details about how to manage this facility.

Shared Libraries

The C Software Development Set package, within the 386 UNIX System V Software Development Set product, provides the Advanced Programming Utilities and the C Programming Language Utilities which allow the programmer to build a shared library of routines accessed at run-time, rather than having those routines combined with an application program at load time.

The most commonly-used routines from the C Library and the Networking Services Library are available. Most of the UNIX System commands use these two libraries, as do any applications that were built with them. For more information about generating Shared Libraries, see the chapter devoted to this topic in the Programmer's Guide, Vol. 1.
Features Of 386 UNIX System V

Command Syntax Standard

getopt(1) allows shell procedures to parse command lines to check for legal options and to process optional arguments. A new command, getopt(1), is an enhanced version of the getopt(1) command. getopt is consistent with and supports rules 3 through 10 of the UNIX System command syntax standard. [The standard is described on the intro(1) manual page.]

NOTE

You may use getopt in shell scripts with UNIX System V Release 3. However, you should use getopt instead of getopt, because beginning with the next major UNIX System V release, getopt will no longer be supported.

To assist in the conversion of shell scripts that are affected by a change from getopt to getopt, a conversion command, /usr/lib/getoptcvt, is provided. (See the getopt(1) manual page for details.)

Signal Mechanism Enhancements

A new set of system calls [see the sigset(2) manual page in the Programmer's Reference Manual] provides a mechanism to catch and hold signals without losing them during later processing and to guarantee that a process reaches the signal handler before it is interrupted by another signal. Some additional signal-handling features, provided by other popular operating systems, are also available.

Improved Facilities for Supporting Terminals

Support for terminals is improved with new features of the Remote Terminal Package, and also as part of the Base System.

Remote Terminal Package

The "Remote Terminal Package" (often called curses/terminfo) has the following new features:

- Expanded support for terminal filters, soft labels, and new AT&T terminals.
New commands: *captinfo*(1M) converts *termcap* entries to *terminfo* entries; *infocmp*(1M) compares two *terminfo* entries or prints entries in several formats. [Section (1M) is in the *User's/System Administrator's Reference Manual*.]

New options to the *tput*(1) command to initialize, reset, and learn the "long name" of a terminal.

An improved version of the *terminfo* compiler, *tic*.

New documentation on the manual pages and in the "curses and *terminfo*" chapter of the *Programmer's Guide*.

**AT&T Windowing Utilities (included in the Base System)**

This new package contains software that is required by AT&T windowing terminals, such as AT&T's line of DMD terminals (for example, the 630). Routines included are for creating, deleting, and manipulating terminal windows, querying terminal window status, and providing statistics about using windowing routines. For more information about this package, see the *layers*(1) manual page, and the *libwindows*(3X) manual page in the *Programmer's Reference Manual*.

**Additional Features (crash, encryption)**

386 UNIX System V Release 3.1 provides improvements to the *crash*(1M) command, and repackaging of encryption mechanisms.

**crash Command Changes**

In addition to providing debugging support for the new operating system features included in Release 3.0, *crash* has been changed extensively to make it easier to use. The syntax of all the functions has been standardized so that similar functions share similar syntax. There is a *help* function within *crash*, a number base converter, a memory search function, and a disassembler capability. The *crash* (1M) manual page in the *User's/System Administrator's Reference Manual* describes the details of this new *crash* command.
Encryption

The Security Administration Package is optionally installable and consists of an encryption mechanism for encoding sensitive files. The Security Administration Package is distributed only within the United States.

New System Header Files

New header files were added to /usr/include: unistd.h [definitions for symbolic constants introduced and used throughout the /usr/group Standards document; see unistd(4) in the Programmer's Reference Manual] and limits.h [definitions for commonly used values that vary from implementation to implementation, see limits(4) in the Programmer's Reference Manual]. Several new definitions were added to the header file /usr/include/sys/stat.h to make it easier for programmers to write portable code.
286 Binary Compatibility

The 286 binary compatibility permits use of the 286 binary executable files compiled under the AT&T 6300 PLUS, UNIX System V Release 2.0, Version 2.0 and Version 2.5 using the SGSs provided in those systems. These 286 binary files can be loaded onto the system and executed just like 386 binary files. The 286 binary compatibility is subject to the following restrictions:

- When using the 386 console, the TERM type will be set to 'at386'. Some 6300 PLUS applications use their own termcap file that does not include 'at386' terminal type descriptions and the application will fail because the terminal type 'at386' cannot be found. If this happens, the terminal type must be changed to PC6300PLUS as follows:

  \$ TERM=PC6300PLUS
  \$ export TERM

  After the term type is changed, the application may be invoked. After you exit the application, you should reset your terminal type to 'at386' as follows:

  \$ TERM=at386
  \$ export TERM

- The 386 sdb(1) debugger cannot debug a 286 program because of format differences between 286 and 386 executable files and processes.

- The ptrace() system call is not supported. It is used only by debuggers and is not necessary for running applications.

- prof(1) does not report execution profiles.

- The effective maximum number of open files when a 286 program is exec'ed is one less than NOFILES. NOFILES should be configured to at least 21.

- The number of segment descriptors available in the LDT for 286 programs is limited to 256 unless LDTSZ is increased. This is sufficient to map 16 Mbytes of text and data. LDTSZ is defined in sys/seg.h, and is not a configurable parameter, because the end of the LDT could run over the end of the U block if LDTSZ is set too high.

- A 286 program may require more stack space running on the 386 than when running on a 286 because of increased system call and signal-handling overhead.
The following `ioctl(2)` commands defined in `sys/termio.h` are supported. They are: `TCGETA`, `TCSETA`, `TCSETAW`, `TCSETAF`, `TCSBRK`, `TCXONC`, and `TCFLSH`.

The 286 executable file must be readable (not simply executable) by the user that `execs` the program.
Floating Point Co-processor Support

This release provides support for the Intel 287/387 floating point co-processor. It also provides 387 emulation when the co-processor is absent.

At boot time the kernel checks for the presence of the chip and automatically configures the systems accordingly.
Boot Procedures

The following list of instructions will boot a 386 UNIX System V Release 3.1 system after the Base System has been installed. Apply power to the system and wait 1 or 2 minutes; the second-stage boot should be loaded automatically by the PROMs.

The boot program will load and give you the following message:

Booting UNIX System

If you immediately hit ANY key, the boot process will be interrupted and the following message will be displayed:

Enter name of a UNIX System kernel to boot (eg. /unix):

If you enter the name of a kernel to boot, the boot process will continue. If you don’t enter anything, after 1 minute /unix will boot automatically; if you want to boot /unix immediately, press the [Enter] key.
# 386 UNIX System Base System Devices

The following list defines the contents of the master device file (`/etc/conf/cf.d/mdev`) before any add-on packages are installed.

<table>
<thead>
<tr>
<th>Device Name</th>
<th>Device or software module controlled by &quot;device driver&quot; (See Note 4).</th>
</tr>
</thead>
<tbody>
<tr>
<td>Device Name</td>
<td>Nodes in <code>/dev</code> directory</td>
</tr>
<tr>
<td>character (block) device number</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Device Name</th>
<th>Device or software module controlled by &quot;device driver&quot; (See Note 4).</th>
</tr>
</thead>
<tbody>
<tr>
<td>asy</td>
<td>Serial Port (com1, com2)</td>
</tr>
<tr>
<td>fd</td>
<td>Floppy Disk</td>
</tr>
<tr>
<td>hd</td>
<td>Hard Disk</td>
</tr>
<tr>
<td>kd</td>
<td>Keyboard</td>
</tr>
<tr>
<td>lp</td>
<td>Lineprinter (parallel interface)</td>
</tr>
<tr>
<td>mem</td>
<td>Kernel Memory Driver</td>
</tr>
<tr>
<td>rtc</td>
<td>Real time Clock</td>
</tr>
<tr>
<td>du</td>
<td>Distributed UNIX system (RFS) stubs</td>
</tr>
<tr>
<td>fp</td>
<td>Floating Point Support</td>
</tr>
<tr>
<td>cram</td>
<td>CMOS RAM (memory)</td>
</tr>
<tr>
<td>gentty</td>
<td>Generic tty (STREAMS support)</td>
</tr>
<tr>
<td>ld0</td>
<td>Line Discipline 0 (STREAMS support)</td>
</tr>
<tr>
<td>ntty</td>
<td>Network tty (STREAMS support)</td>
</tr>
<tr>
<td>s52k</td>
<td>2K File System Support stubs</td>
</tr>
<tr>
<td>ipc</td>
<td>Interprocess Communications</td>
</tr>
<tr>
<td>msg</td>
<td>IPC Messages</td>
</tr>
<tr>
<td>sem</td>
<td>IPC Semaphores</td>
</tr>
<tr>
<td>shm</td>
<td>IPC Shared Memory</td>
</tr>
<tr>
<td>sxt</td>
<td>Shell Layers</td>
</tr>
<tr>
<td>xt</td>
<td>Layers (Bit Mapped Terminal)</td>
</tr>
<tr>
<td>prf</td>
<td>Kernel Profiler</td>
</tr>
<tr>
<td>cpyrt</td>
<td>System Initialization Messages</td>
</tr>
<tr>
<td>weitek</td>
<td>Numeric Chip Support stubs</td>
</tr>
<tr>
<td>vx</td>
<td>SimulTask 386 stubs</td>
</tr>
<tr>
<td>osm</td>
<td>Kernel printout Monitor</td>
</tr>
</tbody>
</table>

**Notes:**

1. Nodes are not required for this device driver. See Note 4.

2. These devices are required to support add-on packages, or are stubs (place holders) for add-on device driver packages. The add-on packages may install nodes in the `/dev` directory.
3. The base system does not have nodes for this device. Nodes can be added later via the `/etc/mknod` command.

4. Several *device drivers* are software only drivers. That is, they provide a kernel software function packaged as a *device* that can be added to or removed from the system.
Shutdown Procedures

The following steps are necessary to shut down a 386 UNIX System V Release 3.1 computer properly:

1. Log in as root and change your directory to /.
2. Run the shutdown program with the following options:

   shutdown -y -gTIME

   where TIME is the number of seconds to be allotted before the system is actually halted. A time factor of at least 120 seconds (2 minutes) is recommended for your systems that are being used as multiuser sites. The time factor provides a means to allow users a chance to exit editors and save programs before the system goes down.

The system will proceed to shut itself down after the allotted time, and when the message

   Boot the System Now

appears, the system can be turned off. The [Ctrl] [Alt] [Delete] key combination can be pressed, or the system can be turned off and then turned on again in order to reboot at this point.
Instructions for Updating Selected Files from the Release

If necessary, selected files may be extracted from the installation diskettes. Appendix A contains a list of all the files on the Foundation Set diskettes. All the installation diskettes, with the exception of diskette 1 of 1 of the Base System and diskette 1 of 1 of the Remote Terminal Package, are cpio diskettes. Diskette 1 of 1 of the Base System and diskette 1 of 1 of the Remote Terminal Package are mountable file systems. If the file that you want to extract is on a cpio diskette, you should insert the diskette into the floppy diskette drive and use the following command to extract the selected file:

```
cpio -icBvd < /dev/dsk/f0q15d 'filename'
```

If the file you want to extract is on a mountable file system diskette, you should insert the diskette into the floppy diskette drive and use the following commands to extract the selected file:

```
/etc/mount -r /dev/dsk/f0q15d /mnt
cp /mnt/ <filename> <newfile>
/etc/umount /dev/dsk/f0q15d
```
**Uucp Notes**

Uucp can be set up to connect to other UNIX System machines from your UNIX System machine. Although more detailed uucp information exists in the *Operations/System Administration Guide*, keep in mind the following things:

1. When connecting the serial lines to your system for a shared dial-in/dial-out line, make sure that the serial lines are capable of dropping DTR. If DTR is not dropped properly, your system may have a tendency to keep switching to a lower baud rate, thus making it harder to log in a second time.

2. *Uugetty* with the `-r` flag does not output anything until it first reads in input. This makes it possible for two systems to share one line and both call each other. It also requires that DTR not be tied high, and that when a user logs in to the line, carriage returns must be sent before a login prompt will show up.

3. These are basic instructions only, and the user should read the *Operations/System Administration Guide* for further information.
Software Notes

This section offers some additional information about 386 UNIX System V Release 3.1. Notes about commands, system calls, and files are listed alphabetically and are organized by the Reference Manual where those commands, system calls, or files appear. For example, the section "User Commands" contains notes about commands that are listed in the User's/System Administrator's Reference Manual.

User Commands

awk 3.1

These are the known incompatibilities between the Release 3.0 awk and Release 3.1 awk (nawk).

1. New keywords: delete, do, func (and function), and return. nawk programs may now contain user-defined functions.
3. New pre-defined variables: FNR, ARGC, ARGV, RSTART, RLENGTH.
4. Parameters that appear before the first input file name are available to BEGIN. Until Release 3.1, however, they were not available until after the first input file was opened, which happened after BEGIN.
5. The for (i in arrayname) construct may produce array elements in an order different than before.
6. The input field separator variable, FS, now specifies a regular expression, rather than just a single character. (The third argument to split also behaves this way.) Previously, awk only used the first character in the string assigned to the input field separator if the string had more than one character.
7. Strings may contain escape sequences like their C counterparts: \b, \f, \n, \r, \t, \ddd. Previously, the \ddd notation was not interpreted. Also, if a backslash was not part of an escape sequence, it was retained in the string. To be consistent with C, backslashes are dropped in nawk if they are not part of escape sequences. For
example, awk prints \c for the command print "\c", and nawk prints c.

8. The precedence of operators has changed. nawk firmly establishes their precedence, which matches C precedence, but which differs from their previous awk precedence in some cases. Two examples of code that breaks are:

```c
while ( n /= 10 > 1 ) ...

if ( !"wk" /bwk/ ) ...
```

The old version of awk is provided to avoid compatibility problems. See the section on "Future Directions" in these Release Notes for further information.

**bc**

When you enter the following command line and bc cannot open file2, the error message displayed says that bc cannot open file1:

```c
bc file1 file2
```

Also, when you enter the following command line and bc cannot open file, the error message displayed contains garbled characters instead of the name of file:

```c
bc file
```

**bc**

bc does not handle the following two constructs in the same way:

1. `if ( expr ) {
   ...
   }

2. `if ( expr )
   {
   ...
   }`
The first case produces what one would expect. The second case is equivalent to an if followed by an empty statement, and the compound statement always is executed. If nothing else, the second case should produce a syntax error, but it does not. It dumps core silently.

**bc**

The `bc` command prints an incorrect remainder when dividing with a negative number.

```plaintext
$ bc
-1.2 % 1
9.8
<Ctrl-d>
```

Clearly, when dividing `-1.2` by `1`, the remainder that returns should not be `9.8`.

**cdc**

The `cdc(1)` command ends abnormally when you invoke it without the `-m` option on an SCCS file that does not have the `v` flag set.

**cpio**

`cpio` can overwrite files when invoked with the `-u` option. An overwrite can also occur if the files in the archive are newer than the existing files.

If an I/O error occurs while reading the archive or if the file system runs out of space, the file being created is either corrupted or truncated, and the old existing file is lost because it has been overwritten.

**crypt**

The `crypt` command now takes options that begin with a dash (`-`), such as `-k`. These can no longer be given as keys. This change will only affect those who have the Security Administration Utilities package installed on their system.
**cu**

When you use the `cu` command over a direct line, you should press the `Enter` key to get a `login:` prompt. After the `login:` prompt appears, wait about 5 seconds before entering your login. If you do not wait, the first character that you type will not be displayed on your terminal screen even though it is accepted by the computer.

**cu**

If you do a `cu` from machine_A to machine_B, and then do a `cu` from machine_B to machine_C, the command `~%take file` will not transfer `file` to machine_A.

Transfer files over one link at a time by one of two methods.

- Login to machine_A; `cu` to machine_B and then to machine_C as described above. The command `~%take file` transfers `file` to machine_B. Then type `~..` From machine_B, use the command `~%take file` to transfer `file` to machine_A.

- If possible, `cu` from machine_A to machine_C. You can successfully transfer files over a single link.

**cu**

Occasionally, `cu` fails with the following message even though all devices are available:

```
NO DEVICES AVAILABLE
```

**cu**

If there is a command in `/etc/inittab` that is respawning too rapidly and it generates an error message while the console user does a `cu` to another system, the console user will find that every command typed in is echoed twice on the console, although it is only executed once (as it should be).

**date**

The default timezone for `date` is now GMT.

Another change affecting `date` involves the `TZ` environment variable. Previously, if `TZ` was unset or null, the timezone used would be `EST` or `EDT`. Now the timezone used is `GMT`. For example,
**Software Notes**

TZ='' date

now prints

Thu May 1 21:29:08 GMT 1986

instead of

Thu May 1 17:29:08 EDT 1986.

date also recognizes the extended syntax in the TZ environment variable for alternate time zones. This change should not cause problems since the old syntax continues to be recognized. The changes involving the TZ environment variable occur because the date command uses the new cftime(3C) function to compute the date and time. The cpio, ls, mount, pr, and vi commands also use cftime, and their functionality in displaying times is affected by the TZ environment variable in the same manner.

**ex, vi**

The ex and vi commands exit with a return code equal to the number of errors encountered during the editing session.

Because all return codes must be between 0 and 255, if more than 255 errors are encountered, the return code will not be accurate. If an integral multiple of 256 errors is found, then ex and vi exit with a zero return code.

**ex, vi**

The ex and vi commands no longer set the eighth bit in the characters of the % expansion of the current filename.

When the percent sign (%) is used in a shell escape from ex or vi via the exclamation mark (!), the % is replaced with the name of the file being edited. Previously, each character in this replacement had the eighth bit set to 1 to quote it; now it is left alone.

Generally, you can use older versions of the ex or vi commands on Release 3, but you cannot use the percent sign in a shell escape via the exclamation mark, even if the file being edited has no special characters in it.
ipcs

\texttt{ipcs(1)} always reports the number of processes attached to shared memory segments, \texttt{NATTCH}, as zero, even when running processes are currently attached to shared memory segments.

The information in the \texttt{NATTCH} column is incorrect. To obtain the correct information, follow these steps:

Step 1  Invoke \texttt{crash} as \texttt{root}.

Step 2  Type \texttt{od shminfo 3}. The last word on the line will be the number (in hexadecimal) of shared memory identifiers.

Step 3  Now type \texttt{od shm}em 6.

Step 4  Examine the first \texttt{shm}em identifier. If the leftmost digit of the third word is in the range of 8 to f (hexadecimal) the identifier is in use, and you can get the address for the next step in the last word on the second line. If the leftmost digit is not 8-f, go to Step 6.

Step 5  Type \texttt{od address 3}. The left 4 digits of the last word is a hexadecimal number for the number of processes attached to this identifier.

Step 6  To go to the next identifier, add (hexadecimal) 30 to the address reported in response to \texttt{od shm}em 6 and enter \texttt{od address 6}.

Step 7  Repeat Steps 4-6 until all identifiers in use have been displayed. (The data reported by Step 6 will be all zeros.)

login

After executing \texttt{exec login}, the following warning message will be produced intermittently:

\begin{verbatim}
no utmp entry ... execute from the lowest level shell "sh"
\end{verbatim}

If the \texttt{who} command shows that someone is still logged in on that line, you must kill the \texttt{getty} and allow it to respawn. Otherwise, no action is required.

You can avoid the problem by executing \texttt{exec su} - xxxx (where xxxx stands for a login), or by logging out (that is, hanging up or executing an \texttt{exit} command) and calling back in.
**Ip**

If you issue the `lp` command on `file` within a directory that has 700 permissions, the following error messages will be displayed:

```
lp: can't access file file
lp: request not accepted
```

**cat** or **pr file** and pipe the output to **lp**.

**mailx**

`mailx` changes the modification time of your mailbox (`/usr/mail/login`) so that the timestamp on the mailbox selects the last time new mail arrived, not the last time the mailbox was accessed and changed. In the past, the modification time was the last time the mailbox was accessed and changed.

**mailx**

Using the command `mailx -f a/b`, where `a` is a nonexistent directory, causes `mailx` to pause for a long time, as it continuously tries to access the nonexistent directory. Eventually, `mailx` will time out.

The argument supplied to the `mailx -f` option must be a valid directory.

**mailx**

The `mailx` command now utilizes the tunable parameter `NOFILES` to represent the maximum number of open files per process. This allows `mailx` to open as many files as are configured on the system.

**mv**

When using `mv(1)` to move files within a file system, the file's original owner and group attributes are retained. When a file is moved, using `mv(1)`, across file systems, the file is assigned new owner and group attributes that match the user that invokes the `mv(1)`.

**od**

If a file has an odd number of bytes, `od -c` will report a trailing null byte.
pr

When pr is used to print a file containing lines greater than the column width, 80 (for example, pr -n -i -w80 -l60 0f file 1 lp), the printer prints a new line and carriage return <CR> and then prints the rest of the line. This causes the printer to print more than the specified number of lines on a page. If more than four or five lines wrap around in this manner, the form feed issued after printing the page will cause a blank page to appear in the listing.

ps

The ps command now correctly interprets a non-numeric argument to the -g or -p options as an error, where previously it was treated as zero.

You should change any shell scripts that rely on the previous incorrect behavior of the ps command to reflect the correct operation.

NOTE

The -g option is used to examine the processes belonging to a particular process group leader, not to a particular user group identifier.

sar

sar produces a table in which CPU time is represented as user, system, waiting for I/O, or idle. This idle figure may not be accurate. If memory is being used extensively, the idle time reflected by the output may not be totally idle. Part of the time could be attributed to waiting for memory.

sdiff

When doing an sdiff, if one of the files contains more than 196 characters in one line, the sdiff output will lose the separator symbol (1 ) or loop infinitely, or both.

Do not sdiff a file that contains more than 196 characters in a single line.
sh

When the command `sh -c string` is executed, the flag value returned is blank. The correct value is `flag=s`. If the `-c` option and the `string` are piped into the shell command, the correct value will be returned.

Do not try to execute the `sh` command using the `-t` and `-c` options together. It does not work.

sh

A trailing colon in the shell `PATH` variable causes the current directory to be included in command searches.

sh

The `test` and `[… ]` commands now use the effective user and group IDs to determine permissible file access, instead of the real IDs as previously used.

The only way to invoke the `test` command with different effective and real IDs is to invoke the command, or a shell script containing it, from a compiled program that has the set user (group) ID on execution permission. Otherwise, the effective IDs are the same as the real IDs, and this change will have no effect. If your program relies on the test operators to behave as they did previously, which is to have `test` use the real user and group IDs, you should change it to use the `setuid(2)` or `setgid(2)` system calls to set the effective ID to the real ID before invoking the `test` command or shell script.

sh

The shell no longer treats the eighth bit in the characters of a command line argument specially; it also no longer strips the eighth bit from the characters of an error message.

If you have any programs that set the eighth bit of characters, they should be changed. You should use one of the standard shell-quoting mechanisms, such as the backslash, instead of setting the eighth bit.

sh

The shell `type` command displays backslashes before each character that was quoted initially.
This change is a result of the change described previously, where the eighth bit is no longer used by the shell to quote characters.

sh

The result of a parameter substitution in a command like

```bash
ls "${a:=xyz abc} lmnop"
```

is now correct.

In general, the parameter substitution

```bash
${parameter:=word }
```

when used inside double quotes and when the `word` contains spaces, now works correctly. If you have programs that rely on the previous incorrect behavior, you should change them to reflect the correct behavior.

shl

If you hang up while in shell layers, `/etc/utmp` may not be cleaned up (that is, the `who` command still shows that you are logged in, and the `ps` command shows a `getty` running on that line). If someone else then calls into that line, the `login` fails because the `utmp` entry cannot be found. To kill the `getty` corresponding to the affected line, follow this procedure:

**Step 1** Execute `ps -ef` to find the process ID of the `getty` running on the line erroneously reported as occupied.

**Step 2** Execute `kill -9 PID` to kill the `getty`. When the `getty` automatically respawns, the problem will be cleared.

shl

When using the `shl` command, trapping a "hang-up" signal inside a layered shell suspends the layer. This renders the device unusable until the shell is killed. You receive a warning message that a layer is still running, and then you are returned to the UNIX System shell.

If your terminal does not accept input commands because of the suspended layer, you have to log in at another terminal. To kill the suspended shell, follow this procedure:
Step 1  Find the process number of the shell by executing the following command:

    ps -ef

Step 2  Look at the output of the command for the appropriate tty (for example, sxt001) and for the process number of the shell.

Step 3  Enter the following command:

    kill -9 PID

    where PID is the process number of the shell that you want to kill.

**shl**

If you are at the console in shl and you execute the following sequence of commands, the shell layer does not respond properly:

1. An stty command with a carriage return select style argument (for example, stty cr3)
2. An echo command
3. An stty command with a carriage return select style argument (for example, stty cr0)

In particular, the layer's prompt does not return until the **Break** or **Delete** key is hit. Once the prompt appears, the output of any command executed does not print fully or at all until you press the **Enter** key one or more times.

Return to the shl control layer and use the shl delete command to delete the layer.
shl

If you execute a background process not in a shell layer that sends output to your terminal, and then you enter shell layers [shl(1)], some screen output from the original process may be lost. When you enter shl, the output from the original process temporarily stops printing on the screen. The screen output resumes when you exit shl(1). Loss of data, if it occurs, is noticeable when the screen output resumes.

This is not normal use of the system, and should be avoided.

shl

When resuming a layer of shl, the resuming xyz prompt is often garbled.

vi

Using a named buffer twice in a vi map sequence results in the following error message:

    Can't put partial line inside macro

Use named buffers only once in map sequences.

who

The who -q command now lists the login names in space padded fields of equal size and no longer sorts the entries.

If you have any programs that process the output of the who -q command, you should inspect them to see if they will still work with the new form of the output.

uucp

In the /usr/lib/uucp/Permissions file, read/write/execute permissions are defined for each uucp login. These permissions authorize remote hosts to read/write/execute local files using uucp and uux commands. Currently, a file is not readable by giving it read permission in the Permissions file.

To make a file readable, both read and write permissions should be given.
**Software Notes**

**uucp**

uucp and uux are supposed to allow the use of a tilde (˜) with a login name to designate your home directory (for example, phyllis for phyllis' home directory). This works in most cases, but uux does not handle tilde expansion in the /usr/lib/uucp/Permissions file correctly.

You should use the full pathname for the home directory in the Permissions file.

**uucp**

uucp logins, such as uucp and nuucp, should have distinct user ID numbers. In the /usr/lib/uucp/Permissions file, different read/write/execute permissions are defined for each uucp login. Login names in this file are used to associate permissions with the login. These permissions do not work correctly if more than one uucp login has the same user ID. For example, if uucpa and uucpb both have user ID 10, and uucpa appears before uucpb in the /usr/lib/uucp/Permissions file, then whether uucp logs in as uucpa or uucpb, the permissions will be those of the first user ID (uucpa) that matches the user ID requested. Logging in as uucpb, which has user ID 10, means that the permissions will be those of uucpa.

To avoid this problem, use distinct user ids for uucp logins.

**uucp 3.1**

The uucp command now utilizes the tunable parameter NOFILES to represent the maximum number of open files per process. This allows uucp to open as many files as are configured on the system.

**uulog**

If you specify the f or number option with uulog (see uucp(1C)), you will not receive a prompt after you execute uulog. To regain access to your terminal, press the [Break] key.
System Administrator Commands

accton

Linking message displays once an hour when system accounting is activated. When system accounting is activated, `/usr/lib/ckpacct` attempts to link `cklock1` to `cklock` even though `cklock1` already exists. This results in an annoying message, which displays every hour, informing you about the link.

adm

Console Monitor Type (monochrome or color)

When the system is installed, the default terminal type for the console is set up for monochrome monitor types. This does not present any problems provided that the console monitor connected to the system is a monochrome monitor. If the console monitor is a color monitor, the console monitor will work like a monochrome monitor. To make the color monitor work correctly, the default terminal type must be changed. This is done by changing `AT386-M` to `AT386` in `/etc/profile` and `/etc/stdprofile` before users are added to the system. The files may be edited using `ed(1)` [`vi(1)` can also be used if the Editing Package has been installed]. If users have already been added to the system, then this change must be made in their `.profile` file in their home directory.

adm

Arrow keys do not place the cursor as expected

When using the arrow keys to place the cursor at different fields on the screen, the following applies:

UP moves the cursor to the next field either to the right or to the first field on the next line (whichever applies)

DOWN moves the cursor to the previous field either to the left or to the last field on the prior line (whichever applies)

LEFT places cursor at the beginning of the field

RIGHT places cursor at the end of the field
adm

'working' message should appear during input validation
The *working* message displays in the top right-hand corner to indicate that the system is processing the user’s request. This message does not appear when the system is validating input entered by the user. Processing is taking place and should be indicated as such.

adm

Upper Case
Most fields require entries to be in uppercase. If the entries are not in uppercase, the entry will be rejected.

adm

SHIFT/TAB
Pressing the SHIFT/TAB keys on the console keyboard elicits a beep instead of changing to the previous form.

adm

Shutdown is Automatic
User is instructed to shut down the system when it happens automatically. When setting up a second serial port, the system displays a message instructing the user to shut down the system. The system will be shut down automatically to configure the port.

adm

Adding/Removing Second Serial Port
When adding or deleting a second serial port, the user is permitted to do this via a remote terminal. The problem is that a system shutdown must be performed during this operation, and that command can only be executed on the system console. So, the addition or deletion of the port will fail when being done remotely. The workaround is to add or delete the port on the system console only.
Unmounting a file system that is not mounted

Unmounting a file system already unmounted does not display error message. When trying to unmount a file system that is already unmounted, the system displays, "Please wait while file system is being unmounted." No processing is done at this point and the system just waits. The user should press the CANCEL SLK to exit.

Mount

Mount File System loops when mounting a hard disk already mounted. When trying to mount a hard disk in the "Mount File System" menu, the User Interface gets into a loop when the hard disk partition is already mounted. An error message displays informing the user that the file system is already mounted, but the user is instructed to press CONT to continue. This forces the system to try to mount the file system again resulting with the same error message and instructions. The user should press the CANCEL SLK to exit the menu.

Mount Help Messages

No 'help' message when CHOICES SLK is pressed in Mount/Unmount File System menus. When in the "Mount File System" or "Unmount File System" menus in the User Interface, no help message appears when the CHOICES SLK is pressed.

Unmount

Unmount File System displays file systems that are not currently mounted. When in the "Unmount File System" menu, the system displays file systems that are currently not mounted. It should only display those that are mounted.
Creating File Systems

Creating a file system may dump core with a damaged 360 Kb floppy disk. When in the "Create File System" menu, to create a file system on a 360 Kb floppy disk, if the disk is damaged, the command can fail with a core dump.

Display User Logins

No pop-up menu when there are fewer than four logins in Display User Logins. When in the "Display User Logins" form and there fewer less than four user logins in /etc/passwd, a pop-up menu cannot be displayed. The only way to select the desired login is to toggle between them, using the CHOICES SLK.

Delete User Login

Users' files are moved to /lost+found. When deleting a user's login in the "Delete User Logins" form, the files are moved to /lost+found. This can result in running out of space in the / directory. Check the size of /lost+found periodically and clean it up when necessary.

Formatting a Floppy Disk

Cannot format another floppy of the same density after pressing CONT SLK. After a successful format of a floppy in the "Disk Operations" menu, you are prompted to remove the floppy and press CONT to continue. The problem is that you are returned to the "Disk Operations" menu instead of being prompted if you would like to format another floppy of the same density.

Printer Queue

Scrolling through the printer queue misplaces cursor. After scrolling through the list of jobs in the printer queue, the cursor is misplaced on the screen. To fix this, the user should use the up/down arrow keys to place the cursor as desired.
Software Notes

adm

Printer Setup
If more than one printer is configured on the system, the system will assume that the default printer is the last one to be configured via the "Printer Setup" menu. If the user wants another printer to be the default in this case, the workaround is to enter the following command at the shell prompt:

    lpadmin -d <printer-name>

adm

Printer Setup
Filter field in Printer Setup does not display true value. When in the "Printer Setup" menu, the "filter" field value will always be displayed as YES, regardless to whether the value was changed to NO at some prior point. The system does use the appropriate model file (i.e., raw or filtered) even though the value being displayed may be incorrect.

adm

Printer Setup
The AT&T Administration interface (adm) does not allow two different printers of the same type to be set up. For example, if you have two AT&T 475 serial printers that you want to connect to your system, you may add the first printer using 'adm'. Then lpadmin(1M) must be used to add the second printer of the same type:

    /usr/lib/lpadmin -p<name> -i<model> -v<device>

For example, if you already have an AT&T 475 printer connected to /dev/tty00, and you want to add a second AT&T 475 printer and connect it to /dev/tty01, you should log in as root and type the following commands:

    /usr/lib/lpadmin -patt-475.2 -i/usr/spool/lp/model/att455 -v/dev/tty01
    /usr/bin/enable att-475.2
    /usr/lib/accept att-475.2
adm

Printer Setup
Changing the device in Printer Setup does not retain other changes. When updating, for example, the "description" field in either "Serial Setup" or "Printer Setup", if you change the device (e.g., tty port number) and then press SAVE, the "description" field update will not be saved. This is because the device is considered the "key" and changing its value forces the system to display the data for that device, overwriting all the other fields on the form. The user should press SAVE when updating other fields on the form before changing the device.

adm

System Mail
System mail names are not added to the Permissions file. When adding a new system mail name, the system is not being added to the list of "friendly" systems in /usr/lib/uucp/Permissions. The workaround is to edit the Permissions file manually to add the system name.

at
When the system is installed, the file at.allow is delivered with just root permitted to use the at command. If other logins should be able to use it, the user should be instructed to update this file.

backup
If a multiple floppy diskette backup is cancelled by entering a q before the backup completes, the following error message is displayed:

    An error was encountered while writing to the floppy disk.
This error message should be ignored.

backup
When the user invokes backup and selects the medium (i.e., floppy or cartridge tape), the software approximates how many floppies or cartridge tapes are needed for the backup. This calculation becomes inaccurate when large amounts of data are being backed up and the amount of removable media is underestimated. Therefore, extra media should be ready in case it is needed.
**cpio**

Cartridge Tape Driver hangs in rare instances when using **cpio**. This problem occurs rarely when using **cpio -i**. If the files you are reading are about 4K in size, the tape drive light may not go off even though the read has finished. **ps** verifies that the processes have indeed been killed. Upon trying to access the tape again, the user will see the message

```
/dev/rmt/c0s0: cannot create
```

**crash**

The **t** command in **crash** does not work correctly. Sometimes it will cause **crash** to core dump if the kernel stack is corrupted.

**crash**

**crash** prints out **swapped** while the **trace** or **proc** command examines a swapped-out process. Because u-blocks are not swapped in System V Release 3.0, **trace** and **proc** have no problem with swapped-out processes. In Release 3.1, **trace** works for swapped-out processes only when **crash** is examining a live system. While reading a system dump, the Release 3.1 **crash** cannot access swapped-out u-blocks because those u-blocks are not included in the system dump. An error message will be printed out when the **trace** command is invoked on a swapped-out process. However, in Release 3.1 **proc** can get command names from both the core image and the system dump.

**crash**

The **crash(1M)** command now utilizes the tunable parameter **NOFILES** to represent the maximum number of open files per process. This allows **crash** to open as many files as are configured on the system.

**crash**

The **help** command in **crash** displays commands that are not supported. When using **help** in the **crash** command, it displays usage messages for commands that are not supported (i.e., **major** and **mmu**).

**cron**

The following message means that during shutdown, **cron** processes are killed:

```
cron aborted: SIGTERM
```

This message is normal.
Software Notes

**dcopy**

The usage message issued by **dcopy** does not list all options appropriately. When summarizing, **dcopy** prints the new gap and cylinder sizes when it says that it is printing the old information. The new sizes are, however, printed correctly.

**dcopy**

**dcopy** -f recreates the file system based on the block:inode count given; however, the blocks specified are assumed to be logical blocks (physical blocks = 2 x the number specified), and the inode count is assumed to be 16 x the number of inodes requested. Thus, **dcopy** does not accept the inode or block count specified to be the actual number of inodes or physical blocks required, as other commands (for example, **mkfs**) do.

**dd**

Some floppy disk operations allow a 360 Kb diskette to be accessed as a 1.2 Mb diskette, since errors will not occur until at least 2 blocks have been read.

For example, the following will succeed:

```
    dd if=/dev/rdsk/f0q15d of=/dev/null count=1
```

while this will fail because the error is detected after 2 blocks have been read:

```
    dd if=/dev/rdsk/f0q15d of=/dev/null bs=15b count=3
```

**dfspace**

When the system has no more space available on the hard disk, the `/etc/dfspace` command reports negative space available instead of no space.

**dirname**

The **dirname** command now properly parses the names `//` and `// anything /`.

**errdump**

When no errors have been logged, the **errdump** command does not cleanly handle it.
If messages show that **NVRAM** is invalid, or that panics occurred in January, 1970, it means that nothing has been logged.

**file**

The **file** command incorrectly reports that a core file is file type *English Text* instead of *Data*.

**fdisk**

**fdisk** is not completely compatible with DOS 3.3. First, it does not recognize DOS extended partitions (i.e., a second logical disk on the first physical disk drive). Second, partitions of type "other" created under the UNIX system hang the system when running DOS 3.3 **fdisk**. Finally, **fdisk** allows you to create multiple DOS partitions of which only the first one can be used under DOS 3.3.

**fdisk**

A problem exists when invoking **fdisk** on a hard disk which contains an odd number of cylinders. If the user enters percentages (i.e., changes the defaults) to specify the sizes of various partitions, the resulting sum will not equal 100%. The reason for this is that one cylinder is lost due to the rounding of the first cylinder selected. In the case of an odd number of cylinders, the last cylinder in the partition will always be an integer + .5, rounding up the next highest integer. As a result, the second partition will appear in the **fdisk** table as being a cylinder less than what it actually is. The entire disk is being utilized, but the table information is incorrect for this case.

**format**

**format(1)** will not complain when an attempt is made to format a 360 Kb floppy when a 1.2 Mb floppy is inserted into the drive.

For example, with a 1.2 Mb floppy diskette inserted into the drive, type:

```
format /dev/dsk/f0d9dt
```

This will not complain that the diskette that was inserted into the drive is not the correct density, and the format will appear to have been completed successfully when indeed it has not.
format

When attempting to perform some floppy disk operation without first inserting a floppy disk into the drive, the system will continue to attempt to access the floppy until a floppy disk is inserted. If this should occur on the system, a floppy disk should be inserted.

fsba

512-byte file systems are not supported. 1 Kb is the default filesystem size.

fsck

If you use lseek to skip over entire blocks of a file being written, the kernel assigns a block number of 0 to the skipped blocks. Reads of any bytes in those blocks correctly return a value of 0. However, because the number of "non-zeroed" blocks allocated to the file is consistent with the length of the file, an fsck of the file system produces the following message:

POSIBLE FILE SIZE ERROR

Core dumps sometimes occur in these circumstances.

There is no damage to the file system. If the file size error messages are bothersome, you can eliminate them by determining the name of the file corresponding to the inode that has the "possible error" with ncheck(1M) and, after mounting the file system, copying that file to a temporary file and then back to its original name. Blocks are allocated on the copy to hold all the bytes with values of 0.

fsck

If /tmp is filled up with enough file names to run the root file system out of inodes, an fsck of the file system can produce a POSSIBLE DIR SIZE ERROR message. The file system itself is not damaged.

If the error messages are bothersome, you can eliminate them by running init 1, moving the contents of the directory that provokes the message to a temporary place, removing and recreating the directory, and then moving the files back. For example:
Do not change directory modes and permissions or the modes and permissions of the files in that directory.

**fuser**

fuser \(-k\) file1 file2 does not kill all processes using file2. It will kill the processes using file1 (as specified on the manual page), but the \(-k\) flag is turned off before the processes using file2 are found.

fuser \(-k\) file1 \(-k\) file2 works correctly.

**fuser**

The fuser command now utilizes the tunable parameter **NOFILES** to represent the maximum number of open files per process. This allows fuser to open as many files as are configured on the system.

**getty**

If a getty is spawned from /etc/inittab on a non-existent terminal, it will fail after printing an appropriate error message. However, if this getty fails before the console getty opens the console terminal, it will be the first process to open it, and the console becomes the controlling tty for the failed getty. This action prevents the console getty from getting the console as the controlling tty, and you will be prevented from logging in to the console with the following error message:

Login incorrect

To correct this problem, log in to the contty or one of the ports, edit /etc/inittab to turn off the getty producing the error message, and re-boot the machine.
init

init S drops the remote line and changes speed to 9600 baud when entered from a remote terminal.

Only use init from the console.

init

Executing init s from within shl layers causes inconsistent results. Sometimes, the machine may hang after printing the following message:

init: single user mode

Other times, the system may not really change run states. At no time within shl layers does init s do what it is supposed to do.

init

The init command now reads an environment file when it first begins to run. It uses the contents of that file to retrieve default environment variable settings, and it passes these settings into the environment of all processes it spawns.

The init command now reads the file /etc/TIMEZONE to retrieve the TZ environment variable setting for the system it is running on. The retrieved TZ value, as well as a default PATH value, is then passed into the environment of every process spawned by init. Formerly, the only environment information passed to processes spawned by init was a value for PATH; all other environment variables were left uninitialized. Now, up to five environment variables can be specified in the /etc/TIMEZONE file. This provides the ability to specify systemwide default values for environment variables, including values that may be useful for international systems, such as default character sets.

init

The init command now utilizes the tunable parameter NOFILES to represent the maximum number of open files per process. This allows init to open as many files as are configured on the system.
layers

Cannot run layers when $DMD$ shell variable is set. Running layers when the $DMD$ shell variable is set does not work because layers and layersys are located in /usr/bin and /usr/lib respectively. They are located in $DMD/bin$ and $DMD/lib$ on other versions of the UNIX system. In order for layers to work, you must not set $DMD$. This is a limitation of the 386 system if it is to be compatible with other versions of UNIX System V.

mkpart

mkpart -A 44001 disk0 produces a warning message. When the user enters

```
mkpart -A 44001 disk0
```

to add sector number 44001 as a bad sector, the system responds with the warning message

```
WARNING: V_REMOUNT io control failed.
```

This may result from one or more of the partitions on the hard disk being open. The disk must be remounted manually or the system rebooted for any changes in the partition or alternate sectors to be usable. The bad sector is added and replaced by a good sector, but the warning message is somewhat confusing.

mkpart

When formatting a hard disk using the -F option to mkpart, the factory marked bad tracks are preserved. Also, if mkpart detects factory marked bad sectors, mkpart will mark the entire track bad. Mkpart will display messages when bad tracks are detected and when they are marked bad. A summary message will be displayed when mkpart completes formatting.

mkpart

Since automatic bad block handling is supported by this release, the -A option to manually map bad sectors is no longer appropriate. The -A option to mkpart should be avoided.

mkfs

512-byte file systems are not supported. 1 Kb is the default filesystem size.
Software Notes

**mkfs**

It is possible to build a file system with mkfs (using default inode number) larger than the size of the floppy. It is also possible to create a file larger than the actual number of blocks on this file system (2400 blocks for a 1.2 Mbyte floppy, and 720 blocks for a 368 Kbyte floppy). A later fsck does not complain about either anomaly. It is only when you have to read the file that the read fails.

Create file systems on floppies through the user interface instead of with mkfs.

**mount**

When a mount is done, the permissions used for directory access to the mounted file system are those of the mount point directory. However, the visible permissions when executing ls -al at the mount point are the permissions of the ".." directory in the mounted file system and are misleading. For example:

- The mount point directory before mounting the file system:
  
  ```
  drwxr-x---   mnt  (permissions 0750)
  ```

- The entry in the file system to be mounted:
  
  ```
  drwxrwxrwx   mnt  (permissions 0777)
  ```

- After the file system is mounted on the mount point directory, the visible permissions are:
  
  ```
  drwxrwxrwx   mnt  (permissions 0777)
  ```

  while the real permissions are:

  ```
  drwxr-x---   mnt  (permissions 0750)
  ```

Thus, any program that attempts to read /mnt/.. after the file system is mounted fails, since read access is denied because it is based on the real permissions of the mount point directory.
**NOFILES**

When the NOFILES parameter is set to 70 and above, a file table overflow occurs if you execute two processes that open the maximum number of open files. When incrementing the NOFILES parameter to 70 and above, you should also increase the NFILE parameter by 2.5 to 3 times the NOFILES parameter. This will alleviate any file table overflows since NFILE’s default value is only 150.

**passwd**

When attempting to change a user’s password using the passwd(1) command, the new password can be set to a carriage return.

**prfld**

If the number of text symbols in /unix exceeds 2048, the prfld command will fail with the following error message:

```
error: too many text symbols.
```

A new configuration parameter, PRFMAX, was added to the /etc/conf/cf.d/mtune master file. If the system contains more than 2048 symbols, the PRFMAX value should be updated, and the system administrator should:

```
# cd /etc/conf
# /etc/conf/bin/idbuild
# mv /unix /ounix
# mv /etc/conf/cf.d/unix /unix
# shutdown -y -i6 -g0
```

to create a system that is capable of handling more symbols.

**pwd**

`pwd` fails for path lengths greater than 512 bytes.
**Software Notes**

**restore**

restore incorrectly prompts the user to insert the floppy. When doing a selective restore, the screen clears and the restore command instructs the user to insert the floppy and not to remove it. Control returns to the User Interface so that the user can mark the files to be restored. The screen clears and restore prompts you again to insert the floppy.

**shutdown**

You cannot use `shutdown(1M)` to return to init state 2 from state 3. If you try, you will get the following error message:

```
shutdown: Initstate is not for system shutdown
```

**spell**

Running spell on an executable file causes a core dump. The spell command dumps core when it is run on an executable file.

**swap**

If there is a heavy process load on a system with a small memory capacity, the swap command may silently fail (that is, complete without any error indication) when attempting to add a swap area.

Retry the command when the load on the system decreases. If this is a recurring problem, try to reduce the load on the system or increase the amount of memory on the machine.

**sysdef**

The usage message for sysdef is incorrect. Currently, the usage message for `/etc/sysdef` is:

```
/etc/sysdef [system_name] [master.d]
```

Since `/etc/master.d` has been replaced by `/etc/conf`, the usage message should be changed to:

```
/etc/sysdef [system_name] [conf]
```
**sysdef**

The `sysdef` command only lists tunable parameters. The output of the `/etc/sysdef` command lists only the tunable parameters. It does not list all hardware devices, their local bus addresses, unit count, pseudo devices, loadable modules, and system devices as might be expected.

**sysdef**

`sysdef` does not display the following tunable parameters: `SHLBMAX`, `GPGSMSK`, and `PUTBUFSZ`.

**uadmin**

The `uadmin` reboot option is not supported. The reboot option of the `uadmin` command is not supported on the 386. This precludes an administrator or installation script from rebooting the machine when a new UNIX system kernel has been configured.

**uadmin**

Machine can be rebooted from a remote terminal via `uadmin`. When a user is logged on via a remote terminal and uses `su` to become `root`, the machine can be rebooted from the remote terminal by executing the command, `uadmin 2 0`. This conflicts with the `shutdown` command, which cannot be executed remotely.

**umount**

If you attempt to `umount -d` a disk file system, the error message returned will be as follows:

```
Warning: /dev/dsk/1sx was not in mount table
```

The `-d` option is valid only for remote `umounts`.

**uucheck**

`uucheck` without any options prints nothing; you should always use the `-v` option with `uucheck`.

You cannot ask for different levels of debugging information with `uucheck -x`.
**Software Notes**

**uucheck, uucleanup, Uutry**

Most of the Basic Networking Utilities commands can be executed by users. The exceptions are *uucheck* and *uucleanup*, which require either an administrative (*uucp*) login or a *root* login.

*uucheck*, *uucleanup*, and *Uutry* are located in the `/usr/lib/uucp` directory, which is not in the search path for most logins, including those for *uucp* or *root*. Therefore, you must give the full path name, or you must be in the `/usr/lib/uucp` directory to execute these three commands.

Another alternative is to link the command where it may be easily accessed, for example, `/usr/bin`.

**uucico**

It is possible to get *uucico* into a runaway state when you use it through the STARLAN network under extremely heavy network load. For example, the process may accumulate too much time (2500 minutes of CPU time).

Suppose on machine_A you have a *uucico* to machine_B with suspiciously high CPU time. Log in to machine_B and execute `ps -ef`. Look at the output of the `ps` command for a *uucico* process talking to machine_A (that is, with a command line argument like `-s machine_A`). If there is such a process, the connection is still active. If there is not, the *uucico* on machine_A is in a runaway state.

If *uucico* is in a runaway state, do the following:

**Step 1** Kill the *uucico* process with `kill -9 PID`, where `PID` is the process ID of the looping *uucico*.

**Step 2** Remove the associated lock file with

```
rm /usr/spool/locks/LCK..machine
```

where `machine` is the system where the *uucico* originated, for example, machine_A.

**Uutry**

When you are using the Basic Networking Utilities over a transport provider, and a remote system listens on an address different from that in the local *Systems* file, trying to *Uutry* to the remote system results in the following error message:

**52 RELEASE NOTES**
Connect failed: NO DEVICES AVAILABLE
This message does not necessarily imply that there are no available devices on
the local system. However, it does mean that **Uutry** has failed after opening a
device and before achieving a connection.

This failure could be caused by a variety of problems, including no devices
available on the local system or the address in the **Systems** file being
incorrect. To see the local devices that are in use, type **uustat -p**.

**volcopy**
While executing **volcopy(1M)**, you have the option of hitting [Delete] to
obtain a shell. On exiting the shell, **volcopy** fails, dumps core and prints the
error message:

```
bus error- core dumped.
```

**volcopy**
Even though **volcopy -y** is supposed to answer all the questions **volcopy**
asks affirmatively, you still have to reply to the first question. **-y** is an undoc­
dumented option.

**volcopy**
**volcopy** prompts you for several questions; however, if **volcopy** is run in
the background, the prompts do not appear on your screen even though **vol­
copy** is still waiting for the response from your terminal. Do not run **volcopy**
in the background.

**Miscellany**

**Cartridge Tape Utilities**
The Cartridge Tape Add-on installation changes the permissions of the
'tape.h' header file. After installing the Cartridge Tape Add-on, the installa­
tion script will have changed the permissions of **tape.h** in **/usr/include/sys**
from modes 444 to 755.
Floppy Operations
Some error messages from floppy operations appear only at the console, regardless of which terminal invoked the floppy command from which the error originated. This is true for all errors detected and displayed by the device drivers.

Floppy Operations
Sometimes reading from a 360 Kb disk drive may fail and it will appear as though the floppy disk drive door is not closed. If this should occur, reinsert the floppy disk and close the drive door.

Floppy Operations
On some 80386 computers, the system will not recognize that the floppy disk drive door is open if the floppy disk is completely inserted into the disk drive and the door is left open. To correct this, make sure that the floppy drive door is always closed when a floppy disk is completely inserted.

Keyboards
Some keyboard keys are not mapped properly for the console. There are three keys on the console keyboard which are not being mapped correctly due to problems with the terminfo entry for at386. They are as follows:

- Prt Sc is mapped to Insert
- Scroll is mapped to Delete
- Lock is mapped to Num Lock

Longest Allowed Pathnames
The longest pathname is now restricted to 1024 bytes. System calls that require pathnames as arguments will now fail, setting errno to ENOENT, if a longer pathname is given.

Previously, the pathname was not restricted by the UNIX Operating System; however, most programs gave an ad hoc limit to the length. Generally, these limits were well below 1024 bytes, so most programs should not be affected by this change.

The limits.h file defines a macro PATH_MAX to be the longest length of a pathname. In Release 3 this file incorrectly sets the macro to 256, but it will probably be changed in a future release to 1024. Local system administrators can safely change the value for PATH_MAX to 1024 without harm, since the Release 3 system internally uses the longer limit.
We encourage you to include the `limits.h` file with a statement like

```c
#include <limits.h>
```

and refer to the **PATH_MAX** macro for the longest pathname allowed.

**console**

If you are running `shl` on the console, and you then run `shutdown -is`, the console will hang.

Exit `shl` before executing `shutdown`.

**Kernel**

The process accounting computation of a process’s memory usage is incorrectly maintained. The value calculated for process memory usage is slightly less than the real result.

**Kernel**

When a program executes integer division by zero, the following error message is displayed:

```
floating exception - core dumped
```

This message does not accurately describe the error.

**Kernel**

Processes spawned by the kernel at boot time (`sched`, `/etc/init`, `vhand`, `bdflush`) have start times (`STIME`), that is the time the system was last brought down, not the time they were spawned.

**Kernel**

Some core dumps may have possible file size errors reported by `fsck`, but these are only warnings and can be ignored. To determine whether the possible file size errors reported are resulting from core dumps, execute `ncheck -i i-number`, where `i-number` is given in the `fsck` message:

```
POSSIBLE FILE SIZE ERROR I=i-number
```

`ncheck` will generate the pathname of a file from its inode number, `i-number`. 
Kernel

If the operating system runs out of free clists, all input/output activity from/to terminal ports and the console will cease. No warning message is printed by the system to show that it is out of clists.

Kernel

The value of the SHMALL tunable parameter specifies the maximum number of in-use shared memory segments allowable systemwide. This parameter is not checked by the system [that is, shmget(2) does not check this limit].

Saving Device Files When Backing Up root File System

When you back up the root (/) file system, the device files (/dev directory) are not saved as part of the backup. To save the device files, become the superuser, mount a blank formatted floppy that has a file system on it, and enter the following commands:

```
# mount /dev/* /mnt
# find /dev -print ! cpio -pd1 /mnt
```

where /mnt is the directory on which the floppy disk file system is mounted. The cpio options are lowercase letters p, d, and l.

To restore the files, insert the floppy on which the files were saved, and enter the following commands:
Read/Write Permissions for Basic Networking Do Not Work

The read/write permissions for the Basic Networking Utilities do not work correctly. For a system to be able to read a directory, the target machine must also have granted the system write permissions (/usr/lib/uucp/Permissions). Suppose that your system has read permissions in root and write permissions in /usr/tmp. The following command fails:

```
uux "A!pr A!/etc/inittab > A!/usr/tmp/B.out"
```

However, if your system has read permissions in root and write permissions in /usr/tmp and /etc, the above command line succeeds.

Converting to getopts By Hand

getcode [see getopts(1)] adds about 30 lines of code to a shell script, so you may want to convert scripts by hand instead. Converting by hand probably will make the code cleaner and easier to understand. Also, you do not have to worry about parsing option-arguments that are also options.

Follow these guidelines to convert most scripts that currently use the getopt(1) command.

Step 1 Delete the old invocation line and the if statement that checks the exit code.

Step 2 Change the for loop to a while loop that invokes getopt(1).
Step 3  Change the patterns in the case statement from -option to single option letters.

Step 4  Delete the case for --.

Step 5  Add a case for '?'. This case may be used to print the usage message and to exit with a non-zero exit code. Note that the ? is quoted since it is interpreted for filename expansion.

Step 6  Remove all shift commands within the case statement.

Step 7  Change $2 to $OPTARG for cases that require an option argument.

Step 8  Add the statement shift 'expr $OPTIND - 1' after the while loop so the remaining arguments can be referenced as before. Following is an example of a script before and after conversion:

```bash
# before conversion
set -- 'getopt abo: $*'
if [ $? != 0 ]
then
    echo $USAGE
    exit 2
fi
for i in $*
do
case $i in
    -a | -b) FLAG=$i; shift;;
    -o) OARG=$2; shift 2;;
    --) shift; break;;
esac
done
```
# after conversion
while getopts abo: i
do
  case $i in
    a | b)  FLAG=$i;;
    o)     OARG=$OPTARG;;
    ?)     echo $USAGE
       exit 2;;
  esac
done
shift 'expr $OPTIND - 1'
If you want your script to work on releases before UNIX System V Release 3 (that is, use either `getopts` or `getopt`), convert it as the following example shows:

```bash
if [ "$OPTIND" = 1 ]
then
    while getopt abo: $*
do
        case $i in
            a | b) FLAG=$i;;
            o) OARG=$OPTARG;;
            ?) echo $USAGE
                exit 2;;
        esac
done
    shift 'expr $OPTIND - 1'
    echo $*
else
    set -- 'getopt abo: $*
    if [ $? != 0 ]
    then
        echo $USAGE
        exit 2
    fi
    for i in $*
do
        case $i in
            -a | -b) FLAG=$i; shift;;
            -o) OARG=$2; shift 2;;
            --) shift; break;;
        esac
done
    fi
    echo $*
```
Basic Networking Utilities:
Intelligent Modems

Features have been added to the /usr/lib/uucp/Dialers and
/usr/lib/uucp/Devices files to prevent problems that occur when using System
75s, System 85s, Hayes-compatible modems, and other intelligent modems
that do not keep Carrier Detect (CD) high all the time.

Devices Adding a ,M to the second field of an entry in the Devices file will
cause the O_NDELAY flag to be set when the device is opened.
This prevents BNU software from blocking on the device while
waiting for CD. The example below shows how to add the ,M to a
Devices file entry for a device connected to an automatic call unit
for a Hayes modem.

ACU tty11,M - 1200 hayes \T

Dialers Adding \M before the chat script in a Dialers file entry will set
CLOCAL, preventing any change in the CD lead from resetting the
state of the device. Once the conversation is established, \m will
clear CLOCAL. This will allow BNU to again monitor changes in
CD (for example, to notice if the line drops). The following exam­
ple shows how to add \M and \m to an entry for a Hayes modem
in the Dialers file.

hayes ",-," " \M\DAT\r\c OK\r \EATDT\T\r\c CONNECT \m\c

NOTE For some devices, it may be necessary to add a \p after the \M.
Security Notes

Enhancements to Security

Some improvements in security have been made in 386 UNIX System V Release 3.1. This section provides information about some of these improvements.

Sticky Bit

Public directories like /tmp, /usr/tmp, and any other directories writable by the world or by a group, are vulnerable to the removal of their files by any process. This poses a serious problem to the integrity of files contained in those directories, as well as to the overall security of the system. To avoid potential security problems, a solution is to use the "sticky bit" on a directory to show the restriction on the removal of objects within the directory.

In this release the sticky bit is set (by default) on the public directories /tmp and /usr/tmp. Before a user can remove files or directories from these publicly readable and writable directories, some special requirements (given later in this discussion) must be satisfied. Formerly, anyone could remove any file or directory in /tmp and /usr/tmp.

New in this release is the added functionality of the sticky bit on a directory being settable by a user. In previous releases of UNIX System V, the sticky bit could only be set by the superuser. If a directory is writable and the sticky bit is on, a user can only remove a file in that directory if one or more of the following is true:

1. The user owns the file.
2. The user owns the directory.
3. The file is writable by the user.
4. The user is the superuser.

For information on how to set the sticky bit, see chown(1) in the User's/System Administrator's Reference Manual.

The sticky bit on a directory is set by a regular user via the chmod(1) command and the chmod(2) system call. The new file deletion semantics will be controlled by the unlink(2) and rmdir(2) system calls, because all commands that remove files [i.e, mv(1), rm(1), and rmdir(1)] use those calls to do
the removal.

**Shadow Password**

The shadow password file has been implemented in 386 UNIX System V Release 3.1. Encrypted passwords and their aging information are being moved to an access-restricted file called the "shadow password" file (`/etc/shadow`). The shadow password file will only be readable by `root`.

The shadow password file contains one entry per login. Each entry consists of the following information:

- **username**: The user's login name (ID).
- **password**: A 13 character encrypted password for the user or a lock string to indicate that the login is not accessible.
- **lastchanged**: The number of days since January 1, 1970 that the password has been modified.
- **min**: The minimum number of days required between password changes.
- **max**: The maximum number of days the password is valid.

**User Commands**

**cron**

When `cron(1)` completes a job, it sends a mail message to the user. This mail message had been sent by `root` but will now be sent with the ID of the user who initiated the cron job.

**login**

In the past, 10 login attempts were permitted before the line would be dropped. That limit is now changed to 5 attempts. After the fifth unsuccessful login attempt `login(1)` will sleep for 20 seconds before dropping the line.

**mail**

The following special characters are not valid in the mail forwarding line:

```
; & ! < > ' " ? * [ ] { } ( ) $ # ~ \n```

If a special character is encountered in the "forward to" line, the mail will be
returned to the sender with the message

invalid address

**uucp**

System names must not contain unprintable characters or any of these special characters:

```
; & ! < > ' " * [ ] { } ( ) $ # ~ \n```

The uuxqt daemon will not perform remote execution requests for systems whose names contain any of these characters.

**System Administrator Commands**

**cu**

A new entry called **uudirect** is defined in `/usr/lib/uucp/Devices`. When doing a `cu(1)` from machine A to machine B on a direct line where `uugetty(1M)` is running on that line on the remote machine, an extra carriage return must be sent so that `uugetty` knows whether it is an incoming or outgoing line. In the `/usr/lib/uucp/Systems` file, **uudirect** should be used for the `devices` field. For more information on this file, refer to the User's/System Administrator's Guide.

**loginlog**

To enhance **login(1)** security the time of last login will be displayed after logging in. To turn on the mechanism that logs unsuccessful attempts to access the system, the administrator must create the file `/usr/adm/loginlog`. If the file `loginlog(4)` exists and five consecutive unsuccessful login attempts occur, all will be logged in `loginlog`, then **login** will sleep for 20 seconds before dropping the line. In other words, if a person tries five times, unsuccessfully, to log in at a terminal, all five attempts will be logged in `/usr/adm/loginlog` if the file exists. The **login** component will then **sleep** for 20 seconds and drop the line. If a person has one or two unsuccessful attempts, none of them will be logged.

To enable logging, the log file should be created with read and write permission for owner only. It should be owned by **root** with group **sys**.
The /usr/adm/loginlog file is a text file. It will contain one entry for each unsuccessful attempt. Entries in /usr/adm/loginlog will have the following format:

```
login name:tty specification:time
```

The login name field contains the login name used in the failed login attempt. The tty specification field contains the terminal location of the login attempt, and time contains the approximate time of the login attempt. The default is for this file not to exist and for logging to be turned off.

**lp Commands**

The disable command can be made non-executable for regular users by logging in as root or lp and executing

```
chmod 4550 /usr/bin/disable
```

The cancel command can be made non-executable for regular users by logging in as root or lp and executing

```
chmod 4550 /usr/bin/cancel
```

**pwconv**

If you attempt to run pwconv(1m) on /etc/passwd that has an incorrectly formatted line (for example, one with too many colons), pwconv will stop scanning at the bad line. It will complete the conversion, but the new password file and the /etc/shadow files will both have too few lines. You will still be able to recover the original /etc/passwd by restoring the old file stored as /etc/opasswd. To restore the old password file, follow these directions:

Step 1: Replace the new password file with the original one.

```
mv /etc/opasswd /etc/passwd
```

Step 2: Remove the shadow password file.

```
rm /etc/shadow
```

Step 3: Find and correct the bad entries in /etc/password using vi(1) or ed(1).
The recommended way of changing the password file is with the tools provided, not with the editors. However, because the only way to corrupt a password file is by using an editor, that becomes the only way to correct the error as well. The use of any of the editors is still NOT recommended.

Step 4: Run `/usr/bin/pwconv` again.

`/usr/lib/uucp/remote.unknown`

The `remote.unknown` file has been changed from a shell script to a C language executable program. `remote.unknown` is part of the Basic Networking Utilities software. When an unknown machine starts a conversation with the local machine, `remote.unknown` logs the conversation attempt.

`/usr/spool/cron`

The `/usr/spool/cron` directory that contains directories for `at(1)`, `cron(1)`, and `crontab(1)` jobs will no longer be accessible by users. The directory mode will now be set to 700.

`/usr/spool/uucp`

Except for `root`, users will no longer be able to write in the `/usr/spool/uucp` directory or any directories under it. The directory mode will now be set to 755.

**Compatibility Notes**

**Shell Scripts**

It is strongly recommended that all applications convert any shell scripts into binary programs if specific user (group) permissions are required in the shell script command lines. To pass permissions, the binary program must have the setuid (gid) mode bit on and the owner (group) of the binary program set to the ID required. Then the `exec(2)` system call can be invoked with the binary program as the argument and the correct permissions will be passed.
If it is not possible to convert the shell scripts into binary programs, then a binary interface program must be written that would have the setuid mode bit on and the owner of the file set to \texttt{root}. Next, the process would have to do a \texttt{setuid (gid)} system call internally with the uid (gid) that must be passed to a sub-shell. This is only possible because the \texttt{setuid (setgid)} system call sets both the real and effective uid (gid) when called by a process with the effective uid of \texttt{root}. Finally, the binary interface would then call the shell script. This is a potentially dangerous procedure unless the programmer is aware of all the implications.

\textbf{PATH}

The default \texttt{PATH} environment variable searches the current directory first. A superuser unknowingly may run a program in the current directory. The superuser should change the \texttt{ROOTPATH} so that the current directory is searched last instead of first.

\textbf{login}

To discourage intruders, the encrypted password and password aging information formerly found in \texttt{/etc/passwd} has been moved to \texttt{/etc/shadow}. This file can only be read by the superuser. Users will still be able to change their passwords using the \texttt{passwd(1)} command. Password and aging information is added to \texttt{/etc/shadow} by running a new program, \texttt{pwconv(1M)}. This program can only be executed by the superuser.

If you have an application or program that writes password and/or aging information into \texttt{/etc/passwd}, the program will have to be modified so that \texttt{pwconv(1M)} is executed after the information is appended to \texttt{/etc/passwd}. Until the modification can be made, the administrator with superuser privilege will have to run the program before the user who has been added or whose password information has been modified can log in.

\textbf{ps}

In 386 UNIX System V Release 3.1 the usage of \texttt{ps} has changed. Now \texttt{ps} checks and sets the user's effective UID to the real UID and the effective GID to the real GID. Therefore, only users with a real user id of \texttt{root} or a real group id of \texttt{sys} will be able to use these options to \texttt{ps}. 

RELEASE NOTES 67
Security Notes

edit, ex, vedit, vi, view

The edit, ex, vedit, vi, and view commands allow separate .exrc files in any directory. In addition, if you change directory to another user's directory and use any of these editors to edit a file in that other user's directory, the editor will execute the .exrc file if it exists in the second user's directory. This functionality has security implications depending on the contents of the .exrc file, because the commands are executed as the user invoking the editor and not as the person who owns the .exrc file.

In this release a new option has been added to the vi/ex commands to allow you the option of reading the .exrc file in the current directory. Initially, the flag is NOT set. That is, the vi/ex command will NOT read the .exrc file if it exists in the current working directory. You can modify this option by inserting the line

set exrc

or the abbreviation

set ex

in the $HOME/.exrc file which is read when one of these editors is executed if the EXINIT variable is not set in the .profile. If you want to set the EXINIT variable, add the following lines to your .profile:

EXINIT="set exrc"
export EXINIT

However, you should note that executing vi/ex as another user with su could result in your files being compromised, since certain variables in the environment are passed when su is executed without the "-u".

For more information, see the ex(1) manual page in the User's/System Administrator's Reference Manual.
Internationalization Software Notes

Below is a description of functionality differences that exist between 386 UNIX System V Release 3.1 international features and System V Release 3.0. The changes for each component are described along with any potential compatibility problems.

Functionality Changes

The following commands have functional differences in Release 3.1: awk, cat, cpio, date, ed, egrep, find, grep, ls, make, mount, pg, pr, sed, sort, and vi. These changes were made to implement 8-bit cleanup, date and time conventions, and character classification of different character sets. A few changes were also made to fix existing problems and for other enhancements. At the end of the section there is a list of new and changed environment variables.

Command Changes

cat 3.1

The cat command was changed to remove its dependency on the ASCII code set when used with the -v, -e and -t options. The output of cat depends on the character classification table used. All printable characters will be printed as the characters themselves. ASCII non-printable control characters will continue to be printed as ~C. Non-printable 8-bit characters will still be printed as M-x, where x is the character specified by the low order 7 bits. The environment variable CHRCLASS is used to determine if a character is printable.

This change should not present compatibility problems because the functionality change only occurs when the CHRCLASS environment variable is set.

cpio 3.1

The date output of the -vt option to cpio has changed, depending on what value the environment variable LANGUAGE has. In particular, the abbreviated month names will be displayed in the appropriate language.
This change should not cause compatibility problems because the changed behavior occurs only when the LANGUAGE variable is set.

**date 3.1**

The formatting software provided by the date utility has been moved to the cftime function in the C library, and date invokes the cftime routine with the appropriate arguments. date now supports date settings after the turn of the century and uses the environment variables CFTIME and LANGUAGE to define a default output format and to specify different languages.

These changes should not cause any compatibility problems if you do not use the internationalization features, because date functions differently only when the new environment variables are set, or a format string has extra characters for a century setting.

**ed 3.1**

ed no longer complains when a file has non-ASCII characters. The high-order bit is not stripped when reading in input and ed commands. If an encryption key is set, and the CHRCLASS environment variable is set to a non-null value and does not have the value ascii, ed now uses a different heuristic to determine if a file is encrypted. ed's 1 command now prints octal codes for non-printable 8-bit characters. Also, ed now allows 8-bit characters in regular expressions. See the section on regexp.h, and the -c and -x options on the ed(1), edit(1), ex(1), and vi(1) manual pages in the User's/System Administrator's Reference Manual.

The changed behavior occurs only when ed is used with 8-bit characters. A different heuristic for determining if a file is encrypted is used only when the CHRCLASS environment variable is set.

**egrep 3.1**

egrep now allows 8-bit characters in regular expressions and matches 8-bit characters in files correctly. Also, egrep uses the CHRCLASS environment variable to convert regular expressions and text to all lowercase for the -i option. This should not cause any problems because the functionality change occurs only when a new environment variable is set.
**expr 3.1**

`expr` now processes 8-bit characters in regular expressions. These changes are described in detail in the section on `regexp.h`.

**find 3.1**

`find` now processes 8-bit characters. A backslash is now used as an escape character, which allows you to escape the interpretation of metacharacters. For example,

```
find . -name '\**' -print
```

prints all filenames that begin with an `*`. Previously, `find` would print all files whose names began with a `\`, and there was no way to escape the interpretation of metacharacters.

The use of a backslash as an escape character only causes problems in a shell script that uses `find` to print the names of files that have backslashes in them.

**grep 3.1**

`grep` now processes 8-bit characters in regular expressions. These changes are described in detail in the section on `regexp.h`. `grep` also uses the `CHRCLASS` environment variable to convert regular expressions and text to all lowercase for the `-i` option. This change should not cause any problems because the functionality change occurs only when a new environment variable is set.

**ls 3.1**

The date in the long `ls -l` format output will change based on the value of the environment variable `LANGUAGE`. In particular, the abbreviated month names will be displayed in the appropriate language (see the section on `date`).

When `ls` is used with the `-q` or `-b` options, the determination of whether or not a character is printable is based on the character class table identified by the environment variable `CHRCLASS`.

These changes should not cause any compatibility problems because the changed functionality occurs only when new environment variables are set, or the environment variable `TZ` is unset or contains extra information for alternate time zones.
mount 3.1

If mount is invoked with no arguments, the date output will be based on the environment variable LANGUAGE. In particular, the abbreviated month and weekday names will be displayed in the appropriate language. The date and time output from mount is also affected differently by the TZ environment variable (see the section on date.)

These changes should not cause any compatibility problems because the functionality changes only when the LANGUAGE environment variable is set, or the TZ environment variable is unset or contains extra information for alternate time zones.

pg 3.1

pg now processes 8-bit characters in regular expressions (see the section on regexp.h). pg also correctly counts the number of printable characters on a line when it encounters 8-bit characters. This allows split lines that are longer than the screen width. The determination of whether or not a character is printable is based on the character class table identified by the environment variable CHRCLASS.

These functionality changes involve 8-bit characters and should not cause compatibility problems.

pr 3.1

The date output of pr will change based on the environment variable LANGUAGE. In particular abbreviated month names will be displayed in the appropriate language. These changes should not cause any compatibility problems because the functionality changes only when the LANGUAGE environment variable is set, or the TZ variable is unset or contains extra information for alternate time zones (see the section on date).

regexp.h 3.1

regexp.h is a header file used by ed, expr, grep, pg, and sed. It is also used by bfs, nl, csplit, and acctcom, but these commands are not part of Release 3.1 international features. regexp.h was modified to allow 8-bit characters in character classes. Previously, when a character call was compiled, the high-order bit would be stripped from 8-bit characters.
The changes to regexp.h for 8-bit characters should not cause any problems.

**sed 3.1**

sed now processes 8-bit characters in regular expressions (see the section on regexp.h) and sed's y command. The l command now uses the environment variable CHRCLASS to determine if an 8-bit character is printable, and it will print an octal code if an 8-bit character is not printable. Previously, sed would have tried to print an 8-bit character even if it was not printable.

These changes should not cause any compatibility problems because they only pertain to 8-bit characters.

**sort 3.1**

The ASCII dependency of the -d, -f, and -i options to sort has been removed. The output now depends on the character classification table identified with the environment variable CHRCLASS. When the command is used with the -M option, the output depends on the value of the environment variable LANGUAGE.

The changes for the -d, -f, and -i options only pertain to 8-bit characters and should not cause any other compatibility problems. The change to the -M option takes effect only when the environment variable LANGUAGE is set and should not cause any compatibility problems.

**vi 3.1**

vi no longer strips the high-order bit from 8-bit characters read in from text files, text insertion, and editing commands. It no longer looks for magic numbers of object files when reading in a text file. It also writes out text and displays text without stripping the high-order bit. Also, -L and -r options use the environment variables LANGUAGE and TZ.

vi now displays the octal codes of non-printable 8-bit characters in the text using the CHRCLASS environment variable to determine if a character is printable. It will also use the CHRCLASS environment variable to convert between uppercase and lowercase characters for the tilde command and for the ignorecase option.

vi uses the same heuristic as ed to determine if a file is encrypted. The heuristic functions differently when the CHRCLASS environment variable is set (see the section on ed).
The change involving magic numbers should not cause any problems. The code looking for object file magic numbers is outdated because the object file header has changed. Also, vi looks for the magic numbers 0177555 and 0177545 which are only seen in old archive files. The new archive files have a string header. Also, the magic numbers contain 0377 in the high-order byte. Except for null characters, vi can now display an object file, and there is no reason to forbid editing of object files.

The other changes should not cause any compatibility problems because they only occur for 8-bit characters or require that the CHRCLASS environment variable be set.
Environment Variables

CFTIME, CHRCLASS, and LANGUAGE are new environment variables in Release 3.1. Setting them will cause the functionality of commands and C library functions to change. Also, the TZ environment variable may be interpreted slightly differently. The following table lists the commands and library functions that function differently with these variables.

<table>
<thead>
<tr>
<th>Command/Function</th>
<th>Environment Variables</th>
</tr>
</thead>
<tbody>
<tr>
<td>cat</td>
<td>CHRCLASS</td>
</tr>
<tr>
<td>cpio</td>
<td>LANGUAGE, TZ</td>
</tr>
<tr>
<td>date</td>
<td>CFTIME, LANGUAGE, TZ</td>
</tr>
<tr>
<td>ed</td>
<td>CHRCLASS</td>
</tr>
<tr>
<td>egrep</td>
<td>CHRCLASS</td>
</tr>
<tr>
<td>grep</td>
<td>CHRCLASS</td>
</tr>
<tr>
<td>ls</td>
<td>CHRCLASS, LANGUAGE, TZ</td>
</tr>
<tr>
<td>mount</td>
<td>LANGUAGE, TZ</td>
</tr>
<tr>
<td>nawk</td>
<td>CHRCLASS</td>
</tr>
<tr>
<td>pg</td>
<td>CHRCLASS</td>
</tr>
<tr>
<td>pr</td>
<td>LANGUAGE, TZ</td>
</tr>
<tr>
<td>sed</td>
<td>CHRCLASS</td>
</tr>
<tr>
<td>sort</td>
<td>CHRCLASS, LANGUAGE</td>
</tr>
<tr>
<td>vi,ex,edit</td>
<td>CHRCLASS, LANGUAGE, TZ</td>
</tr>
</tbody>
</table>
Future Directions

awk, nawk, oawk

With Release 3.1 there is a new awk (nawk) (see "Features of 386 UNIX System V Release 3.1" in these Release Notes). For this release awk is linked to oawk and is the default when you type awk. In the next major release of UNIX System V, nawk will be the default and will be linked to awk, but oawk will still be available.

NOTE nawk must be used with the international version of the software.

Regular Expressions

In Release 3.1 the implementation of regular expressions (for example in ex, egrep, regexp.h, and sh) has been extended to support 8-bit characters. The semantics for the range notation, which currently uses ordinal values for the character, are the same as in previous releases. This permits ranges to include: 7-bit characters, 8-bit characters, and 7- and 8-bit characters.

In the next major release, UNIX System V may support multiple character sets. The semantics for range expressions across character sets may change when support for multiple character sets is provided. Range expressions that contain 7- and 8-bit characters should be avoided due to this direction.

getdents(2)

The implementation of getdents(2) does not match the description in the dirent(4) manual page. The field d_off in struct dirent does not contain the file offset of the current directory entry, but rather the file offset of the following entry. This will be corrected in the next major UNIX System V release. The correction may require the re-compilation or re-linking of programs using the directory-management library routines described in directory(3X)
Future Directions

(opendir, closedir, readdir, telldir, seekdir, rewinddir) and may require source changes to programs using the getdents(2) system call directly.
Documentation

Essential documentation is provided with the software when purchased. Separate sets of documentation for Foundation and Software Development are available and can be ordered. Consult with your AT&T Sales Representative/authorized dealer or see the Documentation Roadmap for more details. The Documentation Roadmap can be ordered separately by using the 9-digit ordering number 999-300-427. The Product Overview can be ordered separately by using the 9-digit ordering number 999-300-428.
Remote File Sharing Notes

Software Description

Remote File Sharing (RFS) Release 1.1 is a software package that allows computers running 386 UNIX System V Release 3.1 or later releases to share resources (directories containing files, subdirectories, devices, and named pipes) selectively across a network. Administrators for computers on an RFS network can choose directories on their systems they want to share and add them to a list of available resources on the network. From this list, they can choose resources from remote computers that they would like to use on their computers.

Each computer on a Remote File Sharing system can be grouped with others in a "domain" or can operate as an independent domain. The domain can provide a central point for administering a group of computers. Unlike other distributed file systems used with the UNIX Operating System, Remote File Sharing is built into the operating system. This approach has several advantages:

Compatibility

Once you mount a remote resource on your system, it will look to your users as though it is part of the local system. You will be able to use most standard UNIX System features on the resource. Standard commands and system calls, as well as features such as File and Record Locking, work the same on remote resources as they do locally. Applications should be able to work on remote resources without modification.

Security

Standard UNIX System file security measures will be available to protect your resources. Special means for verifying computers and restricting remote user permissions have been added for Remote File Sharing.

Flexibility

Since you can mount a remote resource on any directory on your system, you have a lot of freedom to set up your computer’s view of the world. You do not have to open up all your files to every host on the network. Likewise, you do not have to make all files on the network available to your computer’s users.
New Features

Two major features have been added to Remote File Sharing Release 1.1: client caching, and loop-back.

Client Caching

The client caching feature of RFS provides substantial performance improvements over non-caching systems by reducing the number of times data must be read across the network. Client refers to the computer that is using a remote resource, while caching refers to the client’s ability to store data in local buffer pools.

The first time a client process reads a block of data from a remote resource, it is placed in local buffer pools. Subsequent client processes reading a server file can avoid network access by finding the data already present in local buffers. This generally causes a large reduction in network messages, resulting in improved performance.

In order for client caching to work simply and reliably, the following features were built into it.

- Cache consistency. Checking mechanisms are used to ensure that the cache buffers accurately reflect the contents of the remote file the user is accessing.

- Transparency. The only difference users should see between caching and non-caching systems is improved response time. RFS-based applications do not have to be changed to run on a Remote File Sharing system that caches remote data.

- Administration. By default, client caching is on. However, options are available to turn off caching for an entire system or for a particular resource. (You would probably only do this if you have an application that does its own network buffering.) There are also some tunable parameters available to fine tune your system to the way you use RFS.

Loop-Back

The loop-back feature allows you to simulate the higher levels of Remote File Sharing (RFS) within one computer. For example, you could advertise a resource and mount it in a different location on the same machine.
Remote File Sharing Notes

The main use of the loop-back feature is to test application programs and give RFS demonstrations with only one computer. For information describing how to use this feature, see the -o option on the radmin(1M) manual page in the User's/System Administrator's Reference Manual.

Software Notes

This section describes problems that may occur with Remote File Sharing and, in some cases, workarounds to those problems.

acct

The accounting file passed to the acct(2) system call cannot be remote. This restriction applies to user software that uses the system call directly and to the software in the optional processing accounting package. RFS does not allow the acct system call; if passed a remote pathname, acct will return an errno of EINV.

chroot

If you use the chroot command to change to the root directory of a remote machine, the ps command will not work properly.

Client Caching

Reads and writes of block special files are not cached because they could duplicate other data in the cache (for example, data from a regular file residing on the block device defined by the special file). However, if you write to a block device special file on a server machine, the contents of regular files on the block device may be changed. Client-cache buffers associated with these files may be out of date. (Local disk accesses do not suffer from this problem, since block-device writes go through the local buffer pool).

Users who write to block special files on file systems that are advertised remotely in a way that affects the contents of regular files should turn off caching when the resource containing the device is first mounted. (See the -c option of mount(1M).)
Remote File Sharing Notes

df

If df is used without options, it will list each occurrence of a remote resource that is mounted on a system and place an asterisk next to the word blocks for the second and each subsequent resource that was advertised under the same remote file system (for example, /usr/mail and /usr/bin). This signifies that the identical block counts for the resources reside under the same file system.

The problem is that if df is used with multiple remote resources passed as arguments, the asterisk never appears. In this example, the two resources are on the same remote file system, though the asterisk does not appear:

```
$df USRMAIL USRBin
/mnt/(USRMAIL) 30402 blocks
/mnt/(USRBIN) 30402 blocks
```

fumount

The -w option to the fumount command allows a user to specify a grace period between warning clients that a resource is to be removed and actually removing the resource. The atoi subroutine [strtol(3C)] calculates the number of seconds. This routine looks for an initial numeric string and converts it to an integer. Any non-numeric character in the argument terminates the argument. For example, the argument -w 123abc gives a grace period of 123 seconds. Missing arguments and arguments without an initial numeric string produce an error message.

fuser

The fuser command does not find remote users with open local files. For example, mount a 1.2 Mbyte floppy file system as /tmp/a. Advertise the /tmp directory as resource TMP. Mount the TMP resource on a remote system and open a file residing in the TMP/a directory you just mounted. On the local system, run fuser /dev/dsk/f0q15d (the floppy). The remote user will not be found.

fuser may miss a process if that process gets a reference to the resource after fuser has begun its search. In this case, the offending process can be killed explicitly with the kill command. When all processes using the resource are gone, the resource can be unmounted.
**idload**

Many ID mapping features do not function properly with the loop-back function. Only use **global** blocks of information in mapping files (uid.rules and gid.rules). Within global blocks only **default transparent** works as intended. Specific mapping (map lines) or attempts to use host blocks will result in users and groups being mapped to 60002.

**labelit**

**labelit** performs a check to ensure that the destination device path begins with /dev/r?? for tape devices. However, remote tape devices on an RFS network are typically mounted under a local directory or in /dev. As a result, **labelit** cannot be used over RFS for remote tape devices.

**Logs**

These log files may contain information relating to RFS activities:

```
/usr/adm/rfuadmin.log
/usr/adm/log/rfuadmin.log
/usr/net/servers/rfs/rfs.log
/usr/net/nls/netspec/log
```

These files are for internal use only! Customers should not rely on the contents of these files because the information may change or the file may be deleted in future releases. Any tool written that takes advantage of the information contained in these files is not guaranteed to work in the future. (In the list above, netspec is replaced by the transport provider used by RFS. For the STARLAN NETWORK, the netspec is starlan.) The rfuadmin.log files are NOT automatically truncated. You may want to monitor them to make sure they do not get too large.

**Iseek**

Using **Iseek** with a negative offset on a remote file behaves differently than on a local file. On a local file, the call fails and returns **EINVAL**, as it always has. However, on a remote file the call succeeds and returns the negative offset.
mount

When a mount fails because of a password mismatch, the error message can be confusing. The following error messages result from a remote mount failure due to mismatched passwords:

- `negotiate: An event requires attention`
- `mount: negotiations failed`
- `mount: possible cause: machine password incorrect`
- `mount: could not connect to remote machine`

mount

When a remote resource is disconnected by a `fumount(1M)` command or a broken link, the default action in the client `rfuadmin` script is to try to remount the resource as it was mounted before. Therefore, if a resource that was originally advertised by the server as read/write is readvertised read-only, the client’s automatic mount will never succeed.

An administrator can always enter `mount` directly using the latest advertised mode.

mount

The programs that automatically remount remote resources (`/etc/rmount`, `/etc/rmountall`, and `/usr/nserve/rfuadmin`) do not recognize the `-c` option of the `/etc/mount` command as valid. The result is that when they attempt to remount a resource that was disconnected and that was originally mounted with client caching turned off [`mount(1M)` with the `-c` option], the resource is remounted with client caching turned on (the default).

One possible workaround for this problem is for the administrator to wait until the program that does the remount completes successfully, then to unmount and to remount the resource manually using the `mount -c` command. The administrator might also consider killing the process that does the remount, but there is no assurance that other actions that the process must perform will complete successfully.
mount

The **mount** command returns a different **errno** and error message than it did in UNIX System V Release 3.0. The error occurs when you try to mount a local device on a subdirectory of a Remote File Sharing resource you mounted from another machine. The previous errno was **EMULTIHOP** and the error message was:

```
mount: Multihop attempted
mount: cannot mount device
```

The new errno is **EREMOTE** and the error message is:

```
mount: Object is remote
mount: cannot mount device
```

Name Server

When the primary and secondary name servers are under heavy load, the normal passing of name server information between these machines may cause them to hang because the 1K Streams buffers have been depleted. There is one long-term and one short-term solution to the problem.

For the long term, you can increase the number of 1K Streams buffers in `/etc/conf/cf.d/mtune`. The parameter is **NBLK1024**. Increasing 2K Streams buffers (**NBLK2048** parameter) may also help because the system will try to use 2K buffers when 1K buffers are depleted.

The short-term solution is that you can stop Remote File Sharing on any secondary name server that is hung and then bring it back up again; that will clear the **NBLK1024** buffers.

nsquery

The resource list printed by **nsquery** does not always reflect the current state of the domain. If a resource is advertised and the server goes down, a subsequent **nsquery** from a client may still list the resource as being available, even though it is not. An attempt to mount the resource will fail, because it is unable to contact the server. You will have to wait for the remote machine to make the resource available again.
Remote File Sharing Notes

Programs

If a program creating remote directories or files loses its link to the remote machine, and the remote resource is unmounted, the program may begin to create local directories and files. For example, if you are using the `find` command piped to `cpio` to a remote machine and the link to the remote machine goes down and the resource is then unmounted, `cpio` may begin writing on the local machine—the target directory now looks just like an ordinary local directory.

Recovery

Remote File Sharing limits the size of a resource name to 14 characters. This is intended to limit only the resource name and not the domain name (if addressing a resource from another domain).

When a resource is mounted, the stated length of domain and resource name combinations can be used. However, when a link is broken to a server machine, the client’s recovery mechanism tries to unmount the server resources. In doing so, it calculates the size of the resource name including the domain and the "." that separates them. If this string exceeds 14 characters, recovery fails to unmount the resource and may not complete. The result is that users will not be able to access the resource, but the resources will still be listed when you print the names of locally mounted resources (`mount` command).

This problem can be solved in one of two ways. If the problem occurs, the administrator can manually unmount the resource that is posing the problem and then recovery will complete properly (`umount -d resource`). To avoid the problem altogether, the administrator should use domain names and resource names that, when concatenated together, will not exceed a total length of 14 characters.

Recovery

RFS recovery can appear to hang if too many `rmount` processes (automatic remount of disconnected resources) keep respawn ing, waiting for their resources to become available once again.

The problem is that `/etc/rmount`, as delivered, has a retry counter that is set to 0, which means retry forever. The solution is to edit that shell script and change the line where the variable `RETRIES` is set from 0 to a number that is more realistic (such as 5). Also, the variable `TIME` can be changed

86 RELEASE NOTES
from its default value of 60 to a longer time, such as 300 (5 minutes). This will allow the system a more reasonable time to wait for a resource to be re-advertised between mount attempts.

This solution will cause the `rmount` processes to stop respawning after a reasonable span of time.

**Recovery**

If a client loses its link to a server, any attempt to `umount` one of that server’s file systems from the client tree will fail until recovery runs. Recovery from a link failure is handled by `rfuadmin(1M)` and `rfudaemon(1M)`.

Recovery runs automatically when the link breaks, but not until someone tries to access the link or until at most 11 minutes have passed. (The 11-minute time interval applies if you are using STARLAN network. The time may be different for other transport providers.)

If the `umount` fails because the link is gone, the `umount` will start recovery. After recovery runs, a second `umount` will succeed.

**rfadmin**

The `rfadmin -p` command should not be used to pass name server responsibilities back to the primary while the RFS domain is in an unstable state. After RFS is restarted on the primary, the administrator on the secondary machine that is the current domain name server should wait until all requests to mount resources form the primary are completed. (When the primary went down `rmount` requests were placed in the background on all client machines that had mounted resources from the primary.) The secondary administrator can safely use `rfadmin -p` when the mount requests are completed.

**rfadmin**

The `-p` option of the `rfadmin` command returns a value of 2 if the command fails because no computer is available to take over primary name server responsibility.

In earlier releases, the `-p` option of the `rfadmin` command returned a 0 upon success or a 1 upon failure. In RFS Release 1.1, the `-p` option returns a 0 upon success, a 2 if the command fails because there is no computer available to assume primary name server responsibilities, or a 1 if the command fails for some other reason. This change gives the `rfstop(1M)` command the
ability to warn the administrator if the machine being stopped will cause the domain to be without a primary name server.

**rfmaster**

The acting domain name server is responsible for distributing important name service information to all other accessible (secondary) name servers that are serving the same domain, with no more than a 15-minute lag, so that if the acting name server should fail, another host could assume the name server role with a minimal loss of information. However, changes to the rfmaster file after rfstart has been run are not included in the information that is distributed in this way. Because the designation of hosts as primary and secondary name servers is made in the rfmaster file, this has the consequence of not allowing a change to the configuration of which hosts are the primary and secondary name servers for a domain without stopping and re-starting RFS on the affected hosts. For example, adding a new secondary name server to the rfmaster file will not take effect until RFS is taken down on all of the existing (primary and secondary) name servers, as well as the newly designated secondary and then re-started.

This limitation should not be confused with the temporary transfer of name server responsibility to another one of the hosts already listed in the rfmaster file as a primary or secondary name server; this temporary transfer is performed with the rfadmin -p command.

**rfmaster**

When rfstart is executed, the name server process (nserving) scans the rfmaster file and reports possible errors. However, if there is an entry in the rfmaster file that identifies a computer as a name server (either primary or secondary), but there is not a corresponding address entry in the rfmaster file for that computer, it is not reported as an error, even though it should be. In attempting to contact a name server (for example, during rfstart or rfadmin -p), name servers with no address listed will be silently skipped.
rfpasswd

The rfpasswd command is used to change the host password used for RFS, and it is intended to parallel the passwd command in the way it prompts for old and new passwords. However, if a host has no password (for example, it has a null password), the rfpasswd command will still prompt for the old password before asking for the new one, although it should ask only for the new one.

rfstart

This problem only happens when you are issuing an rfstart command from one terminal and some other RFS command from another terminal.

Sometimes when you run the rfstart command it will prompt for a password. RFS has actually started when the password prompt occurs, so an adv command from another terminal would be accepted. After the password is entered, rfstart completes, which includes clearing the advertise table. The result is that the domain advertise table would show the advertised resource as available, while the local advertise table would not.

This problem can be avoided by not issuing additional RFS commands until the rfstart completes and exits to the shell. If the problem does occur, use the adv -m or unadv commands to put the two advertise tables back in agreement.

rfudaemon

User-level recovery of resources that are disconnected gracefully (the remote system shuts down) may fail if the number of lost resources exceeds half of the value of the tunable parameter MAXGDP in /etc/master.d/du. By default, MAXGDP is 24. The failure is accompanied by one or more of the following messages:

rfs user-daemon queue overflow:
made sure rfudaemon is running

rmount

The /etc/rmount script loops forever if the system administrator manually mounts the remote resource on a different directory during the rmount sleep interval. If this occurs, do a ps -ef to get the process ID of rmount, and then kill the process manually by issuing kill -9 pid#.
Remote File Sharing Notes

**Sticky Bit**

If a program that has the sticky bit set is shared through RFS and is executed by a user on a client machine, it will become impossible to remove the program (for example, with the `rm` command). Attempts to remove the file will generate an error message of `text busy`, even though no one on any machine is currently running the program. Removing the sticky bit and trying to remove the program has no effect. A program with the sticky bit set that has never been run across RFS can be removed without complaints, even if it has been run locally.

To remove such a “stuck” sticky bit program, it is necessary to unmount the remote resource. This can be done either from the server, with the `fumount` command, or from the client, with the `umount` command.

**Streams**

The three system calls related to STREAMS—`getmsg`, `putmsg`, and `poll`—will not operate with a file descriptor associated with a remote file. If this is attempted, the system call will fail with `errno` equal to `ENOSTR`.

**swap**

Swap devices cannot be remote, including the swap device configured initially and any swap devices added using the `swap(1M)` command.

**System Calls**

When a Remote File Sharing server does not recognize an incoming request, it will return an `EREMOTE`. In Release 3.0, unrecognized system calls were ignored by the server.

**Transport Provider**

In 386 UNIX System V Release 3.1, RFS is protocol independent. In UNIX System V Release 3.0, RFS works on top of URP/STARLAN 1.0. With Release 3.1 RFS has been modified to work on any connection-oriented protocols that conform to the AT&T Transport Interface specification. To communicate between RFS on UNIX System V Release 3.0 and 386 UNIX System V Release 3.1, an upgraded release URP/STARLAN 1.1 is needed for running RFS Release 1.1. RFS 1.0 can use either URP/STARLAN 1.0 or 1.1.
Network Support Utilities Notes

Introduction

The Network Support Utilities (NSU) Release 1.1 package extends system capabilities to support networking applications. The package includes software support for STREAMS, the AT&T Transport Interface, and the Listener.

The Network Support Utilities package is required to take advantage of the following features of 386 UNIX System V Release 3.1: the Remote File Sharing package, STREAMS mechanisms and tools, the AT&T Transport Interface, the enhanced Basic Networking Utilities, and the Listener.

STREAMS

STREAMS is a general, flexible facility for developing UNIX System communication services. By defining standard interfaces for character input/output within the kernel, STREAMS supports development ranging from complete networking protocol suites to individual device drivers. The standard interfaces and associated tools enable modular, portable development and easy integration of network services and their components—these were used to develop protocol modules and device drivers for Release 3. STREAMS provides a broad framework that does not impose any specific network architecture. It implements a user interface consistent and compatible with the character I/O mechanism that is also available in the UNIX System.

The power of STREAMS resides in its modularity. The design reflects the layering characteristics of contemporary networking architectures. Each basic component (called a module) in a STREAMS implementation represents a set of processing functions and communicates with other modules via a standard interface. From the user level, kernel resident modules can be dynamically selected and interconnected to implement any rational processing sequence. No additional kernel programming, assembly, or link editing is required. Modularity allows for the following advantages.

- User-level programs (commands such as uucp) can be independent of underlying protocols and communications media so the programs need not be changed when new media or protocols between systems become available.
Network architectures and higher-level protocols are independent of underlying protocols, drivers, and media.

Higher-level services can be created by selecting and connecting lower-level services and protocols.

In addition to the standard interfaces, STREAMS provides a set of software tools that help source customers build modules and drivers.

**AT&T Transport Interface**

With Release 3, UNIX System V supports a Transport Interface based on the Transport Service Definition (Level 4) of the International Organization for Standardization (ISO) Open Systems Interconnection (OSI) reference model. The transport service supports two modes of transfer: connection mode and connectionless mode. Connection mode is circuit-oriented and supports data transfer over an established connection in a reliable, sequenced manner. The connectionless mode is message-oriented (datagrams) and supports data transfer in self-contained units with no logical relationship required among units.

The AT&T Transport Interface defines how a user accesses the services of a transport protocol, called a Transport Provider. An example of a Transport Provider is the Universal Receiver Protocol (URP). Application programs access the Transport Provider by using the Transport Interface routines in the new Network Services Library. These routines support access to a Transport Provider in a media and protocol-independent manner. The Transport Provider uses kernel level programs to send the information to the desired physical device, such as the STARLAN Network Access Unit (NAU). By using the AT&T Transport Interface, application programs will be able to access other Transport Providers that may be available in the future.


**Listener**

The "listener" is a program that can be used with Transport Providers on a system. The purpose of the listener is to receive requests for services from another system, interpret which service is needed, and start a process that has been named to provide the requested service. The listener then drops out of the communications path and continues to listen for new service requests.
For more information about the listener, see `nlsadmin(1M)` in the *User's/System Administrator's Reference Manual*. Also, see the Basic Network Utilities and Remote File Sharing sections of the *Operations/System Administration Guide* for information on how to set up the listener for these products.

**Software Notes**

This section describes problems that may occur with the Network Support Utilities and, in some cases, workarounds for these problems.

**Listener**

- In general, when the listener has trouble binding its listening addresses during its start sequence, the actual addresses bound will be chosen by the protocol and hence will differ from the ones requested. There are two cases where this can happen: when the address requested to be bound is either already bound for listening on that network, or when that address does not conform to whatever criteria the protocol is expecting for an address.

In both of these cases, the listener will output a message to the console stating the address returned by the protocol and

```
TLI bound a different name than requested.
```

Additionally, the requested address appears in the listener's log file along with the messages sent to the console. The listener will then exit.

- The listener process can now parse quoted strings in the command line in the database file.

In the previous release, the listener did not understand quoted strings as arguments to servers. If quoted strings were required, it was necessary to specify a shell script as the server and embed the real server and the quoted string arguments there. This change eliminates the need for that indirection.

- The listener process now runs as `root` and sets both the user and group ID as specified by the user ID in the database file.

In the previous release, the listener ran as user ID `listen`. Because of
this, servers that required special permissions needed to be owned by
the appropriate user with the setuser or setgroup ID bits set accord­
ingly. The proliferation of privileged servers was deemed to be a
potential security hazard. This change allows the administrator to
specify in the listener database file the user ID under which the server
should run:

```
nlsadmin -a svc -ccmd -wid -y comment netspec
```

The listener will perform `setuid` and `setgid` system calls based on this
ID before `exec`'ing the server.

**STREAMS**

**open**

A race condition exists in clone opens from different inodes. This problem
exists when two or more disk inodes with the major of the clone device and
equal minors are being opened at the same time. If the window is hit, then
another `open` after the first `open` may bypass the clone device entirely, thus
failing. For example, if `/dev/node1` was major 63 and minor 57, and
`/dev/node2` was also major 63 and minor 57, and if they were two different
inodes, then simultaneous opens of the two devices may result in failure of
the second open.

If two or more separate files are needed on disk, they should be created as
links to one disk inode, thereby closing the window. In the previous example,
`/dev/node2` should be linked to `/dev/node1` instead of being a separate inode.

**read**

For a STREAMS file, when a message containing one or more bytes is
read and the first message block of the message contains zero bytes, the mes­
sage will be erroneously treated as a zero byte message.

**write**

For a STREAMS file, if `write` is interrupted by a signal after some data
have been written, it should return the number of bytes written. However, it
returns -1 and sets `errno` to `EINTR`.

Even when there are no flow control restrictions, `write` will block if
STREAMS internal resources are not available, regardless of the state of
`O_NDELAY`.  

94 RELEASE NOTES
STREAMS Programmer’s Guide

In Appendix C, the initial lines of qenable should read as follows:

qenable – enable a queue

```
int qenable(q)
queue_t *q;
qenable places the queue pointed at by q ...
```

Transport Interface Library

Network Programmer’s Guide

The reference to tirdwr(7) in the caution note on page 5-2 should reflect the change (in the manual page) that popping the module from a stream no longer causes a disconnect.

_t_accept

The description of TBADDATA has been modified to clarify the condition under which it occurs.

[TBADDATA] The amount of user data specified exceeds the allowed amount as returned in the info.connect field by t_open or t_getinfo.

_t_connect

The description of TBADDATA has been modified to clarify the condition under which it occurs.

[TBADDATA] The amount of user data specified exceeds the allowed amount as returned in the info.connect field by t_open or t_getinfo.

_t_getstate

_t_getstate will not work following an exec(2)/t_sync(3N) call sequence because there is insufficient information available to determine the state. The problem arises because the state is maintained in a user process data space, but an exec will overwrite that data space and the state information will be lost. If _t_getstate is called in this case, it will fail with _t_error set to TSTATECHNG.
The processing of expedited transport data units (ETSDU) by t_rcv has been modified. The function will now set the T_EXPEDITED flag on return, if the data are part of an expedited message. The sentence in the fourth paragraph of the DESCRIPTION now reads:

Subsequent calls to retrieve the remaining ETSDU will have T_EXPEDITED set on return.

t_snddis

The description of TBADDATA has been modified to clarify the condition under which it occurs.

[TBADDATA] The amount of user data specified exceeds the allowed amount as returned in the info.discon field by t_open or t_getinfo.

t_sndudata

The paragraph describing the EPROTO error has been expanded as follows:

If t_sndudata is issued from an invalid state, or if the amount of data specified in udata exceeds the TSDU size as returned by t_open or t_getinfo, the provider will generate an EPROTO protocol error (see TSYSERR below.) However, the t_sndudata may not fail because EPROTO errors may not be reported immediately. In this case, a subsequent call that accesses the transport endpoint will fail with the associated TSYSERR.

t_sync

The description of the TBADF value t_errno has been changed to the following:

[TBADF] The specified file descriptor does not refer to a transport endpoint.
Appendix A: Installation Diskette Files

Base System, Diskette 1 of 8

/unix  
/INSTALL  
/INSTALL2  
/.profile  
/LABEL  
/readfloppy  
/bin  
/bin/chmod  
/bin/chgrp  
/bin/chown  
/bin/find  
/bin/stty  
/bin/cpio  
/bin/mkdir  
/bin/sync  
/bin/mv  
/bin/sh  
/bin/expr  
/bin/rm  
/bin/dd  
/bin/sleep  
/bin/cp  
/bin/ln  
/bin/-sh  
/bin/su  
/shlib  
/shlib/libnsLs  
/shlib/libc_s  
/dev  
/dev/console  
/dev/dsk  
/dev/dsk/0s0  
/dev/dsk/0s1  
/dev/dsk/0s2  
/dev/dsk/0s3  
/dev/dsk/1s0  
/dev/dsk/1s1  
/dev/dsk/1s2  
/dev/dsk/1s3  
/dev/dsk/1s4  
/dev/dsk/f0q15d  
/dev/dsk/f0q15dt  
/dev/mem  
/dev/null  
/dev/rdsk  
/dev/rdsk/0s0  
/dev/rdsk/0s1  
/dev/rdsk/0s2  
/dev/rdsk/0s3  
/dev/rdsk/0s4  
/dev/rdsk/1s0  
/dev/rdsk/1s1  
/dev/rdsk/1s2  
/dev/rdsk/1s3  
/dev/rdsk/1s4  
/dev/rdsk/f0q15d  
/dev/rdsk/f0q15dt  
/dev/syscon  
/dev/syssyscon  
/dev/tty  
/dev/swap  
/etc  
/etc/boot  
/etc/getclk  
/etc/group  
/etc/init  
/etc/fdisk  
/etc/inittab  
/etc/inittab2  
/etc/inittab3
Appendix: Installation Diskette Files

/etc/ioctl.syscon
/etc/mkfs
/etc/memsize
/etc/mount
/etc/umount
/etc/uadmin
/etc/mkpart
/etc/disksetup
/etc/passwd
/etc/labelit
/etc/fsck
/etc/fsck512
/etc/TIMEZONE
/etc/fixswap
/etc/mknod
/etc/emulator
/etc/mnttab
/mnt
/tmp
/yes
Appendix: Installation Diskette Files

Base System, Diskette 2 of 8

/bin/acsctcom
/bin/ar
/bin/basename
/bin/cat
/bin/cmp
/bin/date
/bin/df
/bin/diff
/bin/dirmame
/bin/du
/bin/echo
/bin/env
/bin/false
/bin/file
/bin/format
/bin/grep
/bin/i286emul
/bin/idas
/bin/idld
/bin/ipcrm
/bin/ipcs
/bin/kill
/bin/line
/bin/login
/bin/ls
/bin/mail
/bin/mesg
/bin/mt
/bin/newgrp
/bin/nice
/bin/nohup
/bin/passwd
/bin/pr
/bin/ps
/bin/pwd
/bin/rmdir
/bin/sed
/bin/setpgroup
/bin/sort
/bin/su
/bin/tee
/bin/time
/bin/tr
/bin/tty
/bin/true
/bin/uname
/bin/wc
/bin/who
/bin/write
/dev/cr
/dev/cram
/dev/dsk/0s5
/dev/dsk/0s6
/dev/dsk/1s5
/dev/dsk/1s6
/dev/dsk/f0d8d
/dev/dsk/f0d8dt
/dev/dsk/f0d9d
/dev/dsk/f0d9dt
/dev/dsk/f1d8d
/dev/dsk/f1d8dt
/dev/dsk/f1d9d
/dev/dsk/f1d9dt
/dev/dsk/f1q15d
/dev/dsk/f1q15dt
/dev/kmem
/dev/lp
/dev/lp0
/dev/lp1
/dev/lp2
/dev/prf
/dev/rdsk/0s5
/dev/rdsk/0s6
/dev/rdsk/1s5
Appendix: Installation Diskette Files

/dev/rdsk/ls6     /dev/sxt/025
/dev/rdsk/f0d8d   /dev/sxt/026
/dev/rdsk/f0d8dt  /dev/sxt/027
/dev/rdsk/f0d9d   /dev/sxt/028
/dev/rdsk/f0d9dt  /dev/sxt/029
/dev/rdsk/f1d8d   /dev/sxt/030
/dev/rdsk/f1d8dt  /dev/sxt/031
/dev/rdsk/f1d9d   /dev/sxt/032
/dev/rdsk/f1d9dt  /dev/sxt/033
/dev/rdsk/f1q15d  /dev/sxt/034
/dev/rdsk/f1q15dt /dev/sxt/035
/dev/root         /dev/sxt/036
/dev/rtc          /dev/sxt/037
/dev/swap         /dev/sxt/038
/dev/sxt          /dev/sxt/039
/dev/sxt/000      /dev/sxt/040
/dev/sxt/001      /dev/sxt/041
/dev/sxt/002      /dev/sxt/042
/dev/sxt/003      /dev/sxt/043
/dev/sxt/004      /dev/sxt/044
/dev/sxt/005      /dev/sxt/045
/dev/sxt/006      /dev/sxt/046
/dev/sxt/007      /dev/sxt/047
/dev/sxt/008      /dev/sxt/048
/dev/sxt/009      /dev/sxt/049
/dev/sxt/010      /dev/sxt/050
/dev/sxt/011      /dev/sxt/051
/dev/sxt/012      /dev/sxt/052
/dev/sxt/013      /dev/sxt/053
/dev/sxt/014      /dev/sxt/054
/dev/sxt/015      /dev/sxt/055
/dev/sxt/016      /dev/sxt/056
/dev/sxt/017      /dev/sxt/057
/dev/sxt/018      /dev/sxt000
/dev/sxt/019      /dev/sxt001
/dev/sxt/020      /dev/sxt002
/dev/sxt/021      /dev/sxt003
/dev/sxt/022      /dev/sxt004
/dev/sxt/023      /dev/sxt005
/dev/sxt/024      /dev/sxt006
### Appendix: Installation Diskette Files

<table>
<thead>
<tr>
<th>Diskette Files</th>
<th>Diskette Files</th>
</tr>
</thead>
<tbody>
<tr>
<td>/dev/sxt007</td>
<td>/dev/sxt008</td>
</tr>
<tr>
<td>/dev/sxt009</td>
<td>/dev/sxt008</td>
</tr>
<tr>
<td>/dev/sxt010</td>
<td>/dev/sxt011</td>
</tr>
<tr>
<td>/dev/sxt012</td>
<td>/dev/sxt013</td>
</tr>
<tr>
<td>/dev/sxt014</td>
<td>/dev/sxt015</td>
</tr>
<tr>
<td>/dev/sxt016</td>
<td>/dev/sxt017</td>
</tr>
<tr>
<td>/dev/tty</td>
<td>/dev/tty00</td>
</tr>
<tr>
<td>/dev/xt</td>
<td>/dev/xt/000</td>
</tr>
<tr>
<td>/dev/xt/001</td>
<td>/dev/xt/002</td>
</tr>
<tr>
<td>/dev/xt/003</td>
<td>/dev/xt/004</td>
</tr>
<tr>
<td>/dev/xt/005</td>
<td>/dev/xt/006</td>
</tr>
<tr>
<td>/dev/xt/007</td>
<td>/dev/xt000</td>
</tr>
<tr>
<td>/dev/xt001</td>
<td>/dev/xt002</td>
</tr>
<tr>
<td>/dev/xt003</td>
<td>/dev/xt004</td>
</tr>
<tr>
<td>/dev/xt005</td>
<td>/dev/xt006</td>
</tr>
<tr>
<td>/dev/xt007</td>
<td>/dev/xt000</td>
</tr>
<tr>
<td>/dev/xt001</td>
<td>/dev/xt002</td>
</tr>
<tr>
<td>/dev/xt003</td>
<td>/dev/xt004</td>
</tr>
<tr>
<td>/dev/xt005</td>
<td>/dev/xt006</td>
</tr>
<tr>
<td>/dev/xt007</td>
<td>/dev/xt007</td>
</tr>
<tr>
<td>/etc</td>
<td>/etc/brc</td>
</tr>
<tr>
<td>/etc/checklist</td>
<td>/etc/cleanup</td>
</tr>
<tr>
<td>/etc/dfspace</td>
<td>/etc/fboot</td>
</tr>
<tr>
<td>/etc/filesave</td>
<td>/etc/fsanck</td>
</tr>
<tr>
<td>/etc/fstyp.d</td>
<td>/etc/init.d</td>
</tr>
</tbody>
</table>
Appendix: Installation Diskette Files

/etc/init.d/README
/etc/init.d/RMTMPFILES
/etc/init.d/disks
/etc/init.d/firstcheck
/etc/ioctl.syscon
/etc/links
/etc/motd
/usr
/usr/bin
/usr/bin/adv.nf
/usr/bin/BDiff.nf
/usr/bin/nlsadmin.nf
Appendix: Installation Diskette Files

Base System, Diskette 3 of 8

/etc/.packagedate
/etc/bcheckrc
/etc/chroot
/etc/ckbupscd
/etc/clri
/etc/conf
/etc/conf/bin
/etc/conf/bin/idbuild
/etc/conf/cd
/etc/conf/cd/mdevice
/etc/conf/cd/mfsys
/etc/conf/cd/mtune
/etc/conf/cd/sassign
/etc/conf/cd/sdevice
/etc/conf/cd/sdevice.d
/etc/conf/cd/sdevice.d/cpyrt
/etc/conf/cd/sdevice.d/du
/etc/conf/cd/sdevice.d/fp
/etc/conf/cd/sdevice.d/gentty
/etc/conf/cd/sdevice.d/id0
/etc/conf/cd/sdevice.d/mem
/etc/conf/cd/sdevice.d/ntty
/etc/conf/cd/sdevice.d/osm
/etc/conf/cd/sdevice.d/prf
/etc/conf/cd/sdevice.d/sxt
/etc/conf/cd/sdevice.d/xt
/etc/cron
/etc/dcopy1K
/etc/dcopy2K
/etc/dcopy512
/etc/fsck
/etc/dumpsave
/etc/ff
/etc/fsck2K
Appendix: Installation Diskette Files

/etc/rc3
/etc/rc3.d
/etc/rstab
/etc/setmnt
/etc/shutdown
/etc/swap
/etc/sysdef
/etc/umountall
/etc/whodo
/lib
/lib/idcomp
/lib/idcpp
/usr/bin/cut
/bin/tail
/bin/touch
/usr/bin/join
/usr/bin/paste
/usr/bin/pg
/usr/bin/tr
Appendix: Installation Diskette Files

Base System, Diskette 4 of 8

```
/etc/conf/bin/idcheck
/etc/conf/bin/idconfig
/etc/conf/bin/idinstall
/etc/conf/bin/idmaster
/etc/conf/bin/idmkenv
/etc/conf/bin/idmkinit
/etc/conf/bin/idmknod
/etc/conf/bin/idmkunix
/etc/conf/bin/idreboot
/etc/conf/bin/idspace
/etc/conf/bin/idtune
/etc/conf/cf.d/init.base
/etc/conf/init.d
/etc/conf/mfsys.d
/etc/conf/mfsys.d/s5
/etc/conf/node.d
/etc/conf/node.d/asy
/etc/conf/node.d/lp
/etc/conf/pack.d
/etc/conf/pack.d/asy
/etc/conf/pack.d/asy/Driver.o
/etc/conf/pack.d/asy/space.c
/etc/conf/pack.d/clock
/etc/conf/pack.d/clock/space.c
/etc/conf/pack.d/cpyrt
/etc/conf/pack.d/cpyrt/Driver.o
/etc/conf/pack.d/cpyrt/space.c
/etc/conf/pack.d/cram
/etc/conf/pack.d/cram/Driver.o
/etc/conf/pack.d/disp
/etc/conf/pack.d/disp/Driver.o
/etc/conf/pack.d/disp/space.c
/etc/conf/pack.d/du
/etc/conf/pack.d/du/stubs.c
/etc/conf/pack.d/fd
/etc/conf/pack.d/fd/Driver.o
/etc/conf/pack.d/fp
/etc/conf/pack.d/fp/Driver.o
/etc/conf/pack.d/gentty
/etc/conf/pack.d/gentty/Driver.o
/etc/conf/pack.d/hd
/etc/conf/pack.d/hd/Driver.o
/etc/conf/pack.d/ipc
/etc/conf/pack.d/ipc/Driver.o
/etc/conf/pack.d/kd
/etc/conf/pack.d/kd/Driver.o
/etc/conf/pack.d/kd/space.c
/etc/conf/pack.d/kernel
/etc/conf/pack.d/kernel/io.o
/etc/conf/pack.d/kernel/locale.o
/etc/conf/pack.d/kernel/os.o
/etc/conf/pack.d/kernel/space.c
/etc/conf/pack.d/kernel/start.o
/etc/conf/pack.d/ld0
/etc/conf/pack.d/ld0/Driver.o
/etc/conf/pack.d/ld0/space.c
/etc/conf/pack.d/lp
/etc/conf/pack.d/lp/Driver.o
/etc/conf/pack.d/mem
/etc/conf/pack.d/mem/Driver.o
/etc/conf/pack.d/msg
/etc/conf/pack.d/msg/Driver.o
/etc/conf/pack.d/msg/space.c
/etc/conf/pack.d/msg/stubs.c
/etc/conf/pack.d/ntty
/etc/conf/pack.d/ntty/Driver.o
/etc/conf/pack.d/ntty/space.c
/etc/conf/pack.d/osm
/etc/conf/pack.d/osm/Driver.o
/etc/conf/pack.d/pic
/etc/conf/pack.d/pic/space.c
/etc/conf/pack.d/prf
/etc/conf/pack.d/prf/Driver.o
/etc/conf/pack.d/prf/space.c
```
Appendix: Installation Diskette Files

/etc/conf/pack.d/prf/stubs.c  /etc/conf/sdevice.d/rtc
/etc/conf/pack.d/rtc
/etc/conf/pack.d/rtc/Driver.o
/etc/conf/pack.d/s5
/etc/conf/pack.d/s5/Driver.o
/etc/conf/pack.d/s52k
/etc/conf/pack.d/s52k/Driver.o
/etc/conf/pack.d/s52k/stubs.c
/etc/conf/pack.d/s52k
/etc/conf/pack.d/s52k
/etc/conf/pack.d/s52k/stubs.c
/etc/conf/pack.d/s52k
/etc/conf/pack.d/sem
/etc/conf/pack.d/sem/Driver.o
/etc/conf/pack.d/sem/space.c
/etc/conf/pack.d/sem/stubs.c
/etc/conf/pack.d/sem
/etc/conf/pack.d/sem
/etc/conf/pack.d/sem
/etc/conf/pack.d/shm
/etc/conf/pack.d/shm/Driver.o
/etc/conf/pack.d/shm/space.c
/etc/conf/pack.d/shm/stubs.c
/etc/conf/pack.d/shm
/etc/conf/pack.d/shm
/etc/conf/pack.d/shm
/etc/conf/pack.d/sxt
/etc/conf/pack.d/sxt/Driver.o
/etc/conf/pack.d/sxt/space.c
/etc/conf/pack.d/sxt/stubs.c
/etc/conf/pack.d/vx
/etc/conf/pack.d/vx/space.c
/etc/conf/pack.d/vx/stubs.c
/etc/conf/pack.d/vx
/etc/conf/pack.d/vx
/etc/conf/pack.d/weitek
/etc/conf/pack.d/weitek/Driver.o
/etc/conf/pack.d/weitek/space.c
/etc/conf/pack.d/weitek/stubs.c
/etc/conf/pack.d/xt
/etc/conf/pack.d/xt/Driver.o
/etc/conf/pack.d/xt/space.c
/etc/conf/pack.d/xt/stubs.c
/etc/conf/pack.d/xt
/etc/conf/rc.d
/etc/conf/sd.d
/etc/conf/sdevice.d/asy
/etc/conf/sdevice.d/cram
/etc/conf/sdevice.d/fd
/etc/conf/sdevice.d/hd
/etc/conf/sdevice.d/ ipc
/etc/conf/sdevice.d/kd
/etc/conf/sdevice.d/lp
/etc/conf/sdevice.d/msg
Base System, Diskette 5 of 8

/etc/idrc.d
/etc/idsd.d
/etc/link
/etc/magic
/etc/mvdir
/etc/setclk
/etc/unlink
/usr/adm
/usr/adm/acct
/usr/adm/acct/fiscal
/usr/adm/acct/nite
/usr/adm/acct/sum
/usr/adm/sa
/usr/bin/asa
/usr/bin/at
/usr/bin/awk
/usr/bin/banner
/usr/bin/batch
/usr/bin/bc
/usr/bin/cal
/usr/bin/calendar
/usr/bin/cancel
/usr/bin/captoinfo
/usr/bin/checkeq
/usr/bin/chrtbl
/usr/bin/cpset
/usr/bin/crontab
/usr/bin/ct
/usr/bin/cu
/usr/bin/dc
/usr/bin/disable
/usr/bin/enable
/usr/bin/factor
/usr/bin/getopt
/usr/bin/graph
/usr/bin/id
/usr/bin/infocmp
/usr/bin/logname
/usr/bin/lp
/usr/bin/mailx
/usr/bin/oawk
/usr/bin/tar
/usr/bin/tic
/usr/bin/tplot
/usr/bin/tput
/usr/include
/usr/include/sys
/usr/include/sys/fs
/usr/lib/uuep
/usr/lib/uuep/Deveonfig
/usr/lib/uuep/Devices
/usr/lib/uuep/Dialcodes
/usr/lib/uuep/Maxuuseheds
/usr/lib/uuep/Maxuuxqts
/usr/lib/uuep/Permissions
/usr/lib/uuep/Poll
/usr/lib/uuep/SetUp
/usr/lib/uuep/Sysfiles
/usr/lib/uuep/Systems
/usr/lib/uuep/Uutry
/usr/lib/uuep/remote.unknown
/usr/lib/uuep/uuchek
/usr/lib/uuep/uucico
/usr/lib/uuep/uucleanup
Appendix: Installation Diskette Files

Base System, Diskette 6 of 8

/bin/crypt.nf
/etc/diskadd
/usr/bin/ismpx
/usr/bin/jterm
/usr/bin/jwin
/usr/bin/layers
/usr/bin/lpstat
/usr/bin/mcs
/usr/bin/news
/usr/bin/sag
/usr/bin/sar
/usr/bin/shl
/usr/bin/spline
/usr/bin/tabs
/usr/bin/timex
/usr/bin/units
/usr/bin/uucp
/usr/bin/uulog
/usr/bin/uuto
/usr/bin/uux
/usr/bin/xargs
/usr/bin/xtd
/usr/bin/xts
/usr/bin/xtd
/usr/include/agent.h
/usr/include/pn.h
/usr/include/sys/acct.h
/usr/include/sys/adv.h
/usr/include/sys/alttbl.h
/usr/include/sys/asy.h
/usr/include/sys/at_ansi.h
/usr/include/sys/bitmasks.h
/usr/include/sys/buf.h
/usr/include/sys/callo.h
/usr/include/sys/cdUMP.h
/usr/include/sys/cirmgr.h
/usr/include/sys/clock.h
/usr/include/sys/clockcal.h
/usr/include/sys/cmn_err.h
/usr/include/sys/comm.h
/usr/include/sys/conf.h
/usr/include/sys/cram.h
/usr/include/sys/crtct1.h
/usr/include/sys/debug.h
/usr/include/sys/debugreg.h
/usr/include/sys/dirent.h
/usr/include/sys/dma.h
/usr/include/sys/elog.h
/usr/include/sys/erec.h
/usr/include/sys/err.h
/usr/include/sys/errno.h
/usr/include/sys/fblk.h
/usr/include/sys/fcntl.h
/usr/include/sys/fd.h
/usr/include/sys/fdisk.h
/usr/include/sys/file.h
/usr/include/sys/filsys.h
/usr/include/sys/flock.h
/usr/include/sys/fp.h
/usr/include/sys/fs/prfcnt1.h
/usr/include/sys/fs/s5dir.h
/usr/include/sys/fs/s5fblk.h
/usr/include/sys/fs/s5filsys.h
/usr/include/sys/fs/s5inode.h
/usr/include/sys/fs/s5macros.h
/usr/include/sys/fs/s5param.h
/usr/include/sys/fs/sid.h
/usr/include/sys/fs/statyp.h
/usr/include/sys/gate.h
/usr/include/sys/gdpstr.h
/usr/include/sys/gdmisc.h
/usr/include/sys/gdparam.h
/usr/include/sys/gdtab.h
/usr/include/sys/gdtime.h
Installation Diskette Files

<table>
<thead>
<tr>
<th>File Path</th>
<th>File Path</th>
</tr>
</thead>
<tbody>
<tr>
<td>/usr/include/sys/getpages.h</td>
<td>/usr/include/sys/rbuf.h</td>
</tr>
<tr>
<td>/usr/include/sys/hd.h</td>
<td>/usr/include/sys/rdebug.h</td>
</tr>
<tr>
<td>/usr/include/sys/hetero.h</td>
<td>/usr/include/sys/recover.h</td>
</tr>
<tr>
<td>/usr/include/sys/idtab.h</td>
<td>/usr/include/sys/reg.h</td>
</tr>
<tr>
<td>/usr/include/sys/immu.h</td>
<td>/usr/include/sys/region.h</td>
</tr>
<tr>
<td>/usr/include/sys/inline.h</td>
<td>/usr/include/sys/rfsys.h</td>
</tr>
<tr>
<td>/usr/include/sys/ino.h</td>
<td>/usr/include/sys/rtc.h</td>
</tr>
<tr>
<td>/usr/include/sys/inode.h</td>
<td>/usr/include/sys/seg.h</td>
</tr>
<tr>
<td>/usr/include/sys/ioctl.h</td>
<td>/usr/include/sys/sem.h</td>
</tr>
<tr>
<td>/usr/include/sys/ipc.h</td>
<td>/usr/include/sys/sema.h</td>
</tr>
<tr>
<td>/usr/include/sys/ipl.h</td>
<td>/usr/include/sys/shm.h</td>
</tr>
<tr>
<td>/usr/include/sys/istk.h</td>
<td>/usr/include/sys/stat.h</td>
</tr>
<tr>
<td>/usr/include/sys/ivlab.h</td>
<td>/usr/include/sys/statfs.h</td>
</tr>
<tr>
<td>/usr/include/sys/jioctl.h</td>
<td>/usr/include/sys/stermio.h</td>
</tr>
<tr>
<td>/usr/include/sys/kd.h</td>
<td>/usr/include/sys/stream.h</td>
</tr>
<tr>
<td>/usr/include/sys/ld0.h</td>
<td>/usr/include/sys/strlog.h</td>
</tr>
<tr>
<td>/usr/include/sys/lhdr.h</td>
<td>/usr/include/sys/stropts.h</td>
</tr>
<tr>
<td>/usr/include/sys/lock.h</td>
<td>/usr/include/sys/strstat.h</td>
</tr>
<tr>
<td>/usr/include/sys/log.h</td>
<td>/usr/include/sys/swap.h</td>
</tr>
<tr>
<td>/usr/include/sys/lp.h</td>
<td>/usr/include/sys/sxt.h</td>
</tr>
<tr>
<td>/usr/include/sys/macro.h</td>
<td>/usr/include/sys/sys386.h</td>
</tr>
<tr>
<td>/usr/include/sys/map.h</td>
<td>/usr/include/sys/sysinfo.h</td>
</tr>
<tr>
<td>/usr/include/sys/message.h</td>
<td>/usr/include/sys/sysmacros.h</td>
</tr>
<tr>
<td>/usr/include/sys/mount.h</td>
<td>/usr/include/sys/sysm.h</td>
</tr>
<tr>
<td>/usr/include/sys/msg.h</td>
<td>/usr/include/sys/termio.h</td>
</tr>
<tr>
<td>/usr/include/sys/nami.h</td>
<td>/usr/include/sys/tdhr.h</td>
</tr>
<tr>
<td>/usr/include/sys/nserve.h</td>
<td>/usr/include/sys/times.h</td>
</tr>
<tr>
<td>/usr/include/sys/ntty.h</td>
<td>/usr/include/sys/timod.h</td>
</tr>
<tr>
<td>/usr/include/sys/open.h</td>
<td>/usr/include/sys/tiuser.h</td>
</tr>
<tr>
<td>/usr/include/sys/page.h</td>
<td>/usr/include/sys/trace.h</td>
</tr>
<tr>
<td>/usr/include/sys/param.h</td>
<td>/usr/include/sys/trap.h</td>
</tr>
<tr>
<td>/usr/include/sys/pfdat.h</td>
<td>/usr/include/sys/tss.h</td>
</tr>
<tr>
<td>/usr/include/sys/pic.h</td>
<td>/usr/include/sys/ttold.h</td>
</tr>
<tr>
<td>/usr/include/sys/pit.h</td>
<td>/usr/include/sys/tty.h</td>
</tr>
<tr>
<td>/usr/include/sys/poll.h</td>
<td>/usr/include/sys/tuneable.h</td>
</tr>
<tr>
<td>/usr/include/sys/proc.h</td>
<td>/usr/include/sys/types.h</td>
</tr>
<tr>
<td>/usr/include/sys/que.h</td>
<td>/usr/include/sys/uadmin.h</td>
</tr>
<tr>
<td>/usr/include/sys/queue.h</td>
<td>/usr/include/sys/user.h</td>
</tr>
<tr>
<td>/usr/include/sys/ramd.h</td>
<td>/usr/include/sys/ustat.h</td>
</tr>
</tbody>
</table>
Appendix: Installation Diskette Files

/usr/include/sys/utsname.h
/usr/include/sys/v86.h
/usr/include/sys/var.h
/usr/include/sys/vt.h
/usr/include/sys/vtoc.h
/usr/include/sys/weitek.h
/usr/include/sys/xt.h
/usr/include/sys/xtproto.h
/usr/lib
/usr/lib/libp
/usr/vmsys
/usr/vmsys/OBJECTS
### Base System, Diskette 7 of 8

<table>
<thead>
<tr>
<th>Directory Path</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>/etc/volcopy</td>
<td></td>
</tr>
<tr>
<td>/etc/wall</td>
<td></td>
</tr>
<tr>
<td>/usr/bin/300</td>
<td></td>
</tr>
<tr>
<td>/usr/bin/300s</td>
<td></td>
</tr>
<tr>
<td>/usr/bin/4014</td>
<td></td>
</tr>
<tr>
<td>/usr/bin/450</td>
<td></td>
</tr>
<tr>
<td>/usr/bin/backup</td>
<td></td>
</tr>
<tr>
<td>/usr/bin/greek</td>
<td></td>
</tr>
<tr>
<td>/usr/bin/hp</td>
<td></td>
</tr>
<tr>
<td>/usr/bin/installpkg</td>
<td></td>
</tr>
<tr>
<td>/usr/bin/message</td>
<td></td>
</tr>
<tr>
<td>/usr/bin/pwdmenu</td>
<td></td>
</tr>
<tr>
<td>/usr/bin/removepkg</td>
<td></td>
</tr>
<tr>
<td>/usr/bin/restore</td>
<td></td>
</tr>
<tr>
<td>/usr/bin/xtract</td>
<td></td>
</tr>
<tr>
<td>/usr/lib/Install.sh</td>
<td></td>
</tr>
<tr>
<td>/usr/lib/fsinfo</td>
<td></td>
</tr>
<tr>
<td>/usr/lib/installpkg.r</td>
<td></td>
</tr>
<tr>
<td>/usr/lib/removepkg.r</td>
<td></td>
</tr>
<tr>
<td>/usr/lib/ATT455</td>
<td></td>
</tr>
<tr>
<td>/usr/lib/ATT473</td>
<td></td>
</tr>
<tr>
<td>/usr/lib/ATT475</td>
<td></td>
</tr>
<tr>
<td>/usr/lib/ATT__.sf</td>
<td></td>
</tr>
<tr>
<td>/usr/lib/accept</td>
<td></td>
</tr>
<tr>
<td>/usr/lib/acct</td>
<td></td>
</tr>
<tr>
<td>/usr/lib/acct/acctcmsg</td>
<td></td>
</tr>
<tr>
<td>/usr/lib/acct/acctcon1</td>
<td></td>
</tr>
<tr>
<td>/usr/lib/acct/acctcon2</td>
<td></td>
</tr>
<tr>
<td>/usr/lib/acct/acctdisk</td>
<td></td>
</tr>
<tr>
<td>/usr/lib/acct/acctdusg</td>
<td></td>
</tr>
<tr>
<td>/usr/lib/acct/acctmerge</td>
<td></td>
</tr>
<tr>
<td>/usr/lib/acct/accton</td>
<td></td>
</tr>
<tr>
<td>/usr/lib/acct/acctprc1</td>
<td></td>
</tr>
<tr>
<td>/usr/lib/acct/acctprc2</td>
<td></td>
</tr>
<tr>
<td>/usr/lib/acct/acctwtmp</td>
<td></td>
</tr>
<tr>
<td>/usr/lib/acct/chargefee</td>
<td></td>
</tr>
<tr>
<td>/usr/lib/acct/ckpacct</td>
<td></td>
</tr>
<tr>
<td>/usr/lib/acct/diskusg</td>
<td></td>
</tr>
<tr>
<td>/usr/lib/acct/dodisk</td>
<td></td>
</tr>
<tr>
<td>/usr/lib/acct/fwtmp</td>
<td></td>
</tr>
<tr>
<td>/usr/lib/acct/holidays</td>
<td></td>
</tr>
<tr>
<td>/usr/lib/acct/lastlogin</td>
<td></td>
</tr>
<tr>
<td>/usr/lib/acct/monacct</td>
<td></td>
</tr>
<tr>
<td>/usr/lib/acct/nulladm</td>
<td></td>
</tr>
<tr>
<td>/usr/lib/acct/prctmp</td>
<td></td>
</tr>
<tr>
<td>/usr/lib/acct/prdally</td>
<td></td>
</tr>
<tr>
<td>/usr/lib/acct/ptecms.awk</td>
<td></td>
</tr>
<tr>
<td>/usr/lib/acct/ptelus.awk</td>
<td></td>
</tr>
<tr>
<td>/usr/lib/acct/remove</td>
<td></td>
</tr>
<tr>
<td>/usr/lib/acct/runacct</td>
<td></td>
</tr>
<tr>
<td>/usr/lib/acct/shutacct</td>
<td></td>
</tr>
<tr>
<td>/usr/lib/acct/startup</td>
<td></td>
</tr>
<tr>
<td>/usr/lib/acct/turnacct</td>
<td></td>
</tr>
<tr>
<td>/usr/lib/acct/wtmpfix</td>
<td></td>
</tr>
<tr>
<td>/usr/lib/calprog</td>
<td></td>
</tr>
<tr>
<td>/usr/lib/cron</td>
<td></td>
</tr>
<tr>
<td>/usr/lib/cron/.proto</td>
<td></td>
</tr>
<tr>
<td>/usr/lib/cron/at.allow</td>
<td></td>
</tr>
<tr>
<td>/usr/lib/cron/at.deny</td>
<td></td>
</tr>
<tr>
<td>/usr/lib/cron/cron.allow</td>
<td></td>
</tr>
<tr>
<td>/usr/lib/cron/cron.denys</td>
<td></td>
</tr>
<tr>
<td>/usr/lib/cron/queuedefs</td>
<td></td>
</tr>
<tr>
<td>/usr/lib/diffh</td>
<td></td>
</tr>
<tr>
<td>/usr/lib/getoptcvt</td>
<td></td>
</tr>
<tr>
<td>/usr/lib/layersys</td>
<td></td>
</tr>
<tr>
<td>/usr/lib/layersys/lsys.8;7;3</td>
<td></td>
</tr>
<tr>
<td>/usr/lib/layersys/relogin</td>
<td></td>
</tr>
<tr>
<td>/usr/lib/layersys/set_enc.j</td>
<td></td>
</tr>
<tr>
<td>/usr/lib/layersys/wtmp</td>
<td></td>
</tr>
<tr>
<td>/usr/lib/layersys/wtmp</td>
<td></td>
</tr>
<tr>
<td>/usr/lib/lib300.a</td>
<td></td>
</tr>
<tr>
<td>/usr/lib/lib300s.a</td>
<td></td>
</tr>
<tr>
<td>/usr/lib/lib4014.a</td>
<td></td>
</tr>
</tbody>
</table>
Appendix: Installation Diskette Files

/usr/lib/lib450.a
/usr/lib/libgen.a
/usr/lib/libplot.a
/usr/lib/libvt0.a
/usr/lib/libwindows.a
/usr/lib/lpadmin
/usr/lib/lpmove
/usr/lib/lpsched
/usr/lib/lpshut
/usr/lib/mailx
/usr/lib/mailx/mailx.help
/usr/lib/mailx/mailx.help.
/usr/lib/mailx/rmmmail
/usr/lib/mv_dir
/usr/lib/reject
/usr/lib/sa
/usr/lib/sa/sa1
/usr/lib/sa/sa2
/usr/lib/sa/sadc
/usr/lib/t300
/usr/lib/t300s
/usr/lib/t4014
/usr/lib/t450
/usr/lib/terminfo
/usr/lib/terminfo/A
/usr/lib/terminfo/A/AT386
/usr/lib/terminfo/A/AT386-M
/usr/lib/terminfo/a
/usr/lib/unittab
/usr/lib/uucp/uudemon.admin
/usr/lib/uucp/uudemon.cleanup
/usr/lib/uucp/uudemon.hour
/usr/lib/uucp/uudemon.poll
/usr/lib/uucp/uugetty
/usr/lib/uucp/uuschd
/usr/lib/uucp/uuxq
/usr/lib/vplot
/usr/mail
/usr/mail/:saved
/usr/news

/usr/spool
/usr/spool/cron
/usr/spool/cron/atjobs
/usr/spool/cron/crontabs
/usr/spool/cron/crontabs/adm
/usr/spool/cron/crontabs/root
/usr/spool/cron/crontabs/sys
/usr/spool/locks
/usr/spool/lp
/usr/spool/lp/class
/usr/spool/lp/interface
/usr/spool/lp/member
/usr/spool/lp/model
/usr/spool/lp/model/att435S
/usr/spool/lp/model/att455
/usr/spool/lp/model/att470
/usr/spool/lp/model/att473
/usr/spool/lp/model/att474
/usr/spool/lp/model/hplaserjS
/usr/spool/lp/pstatus
/usr/spool/lp/qstatus
/usr/spool/lp/request
/usr/spool/uucp
/usr/spool/uucp/.Admin
/usr/spool/uucp/.Corrupt
/usr/spool/uucp/.Log
/usr/spool/uucp/.Log/uucico
/usr/spool/uucp/.Log/uucp
/usr/spool/uucp/.Log/uux
/usr/spool/uucp/.Log/uuxqt
/usr/spool/uucp/.Old
/usr/spool/uucp/.Sequence
/usr/spool/uucp/.Status
/usr/spool/uucp/.Workspace
/usr/spool/uucp/.Xqtdir
/usr/spool/uucppublic
/usr/tmp
**Base System, Diskette 8 of 8**

/etc/.cpiopc
/etc/.useradm
/etc/adduser
/etc/deluser
/etc/stdprofile
/etc/termcap
/usr/bin/displaypkg
/usr/bin/menu_add
/usr/bin/menu_del
/usr/bin/passmgmt
/usr/bin/pwconv
/usr/include/shadow.h
/usr/include/sys/bootinfo.h
/usr/lib/libsec.a
/usr/vmsys/OBJECTS/Menu.admin
/usr/vmsys/OBJECTS/Menu.office
/usr/vmsys/OBJECTS/Text.exit
/usr/vmsys/OBJECTS/Text.sysinf
/usr/vmsys/OBJECTS/backup
/usr/vmsys/OBJECTS/backup/Menu.backup
/usr/vmsys/OBJECTS/backup/Text.history
/usr/vmsys/OBJECTS/backup/Text.privil
/usr/vmsys/OBJECTS/backup/private
/usr/vmsys/OBJECTS/backup/private/Form.backdir
/usr/vmsys/OBJECTS/backup/private/Menu.backdv2
/usr/vmsys/OBJECTS/backup/private/Menu.backdv3
/usr/vmsys/OBJECTS/backup/private/Menu.backpri
/usr/vmsys/OBJECTS/backup/private/Menu.device2
/usr/vmsys/OBJECTS/backup/private/Menu.device3
/usr/vmsys/OBJECTS/backup/system
/usr/vmsys/OBJECTS/backup/system/Form.backsel
/usr/vmsys/OBJECTS/backup/system/Menu.backdv2
/usr/vmsys/OBJECTS/backup/system/Menu.backdv3
/usr/vmsys/OBJECTS/backup/system/Menu.backdv4
/usr/vmsys/OBJECTS/backup/system/Menu.backpri
/usr/vmsys/OBJECTS/backup/system/Menu.backsys
/usr/vmsys/OBJECTS/backup/system/Menu.device2
/usr/vmsys/OBJECTS/backup/system/Menu.device3
Appendix: Installation Diskette Files

/usr/vmsys/OBJECTS/backup/system/Menu.users
/usr/vmsys/OBJECTS/date
/usr/vmsys/OBJECTS/date/Form.d1
/usr/vmsys/OBJECTS/date/Form.date
/usr/vmsys/OBJECTS/date/Text.datecfm
/usr/vmsys/OBJECTS/date/Text.datpriv
/usr/vmsys/OBJECTS/date/Text.dcfm
/usr/vmsys/OBJECTS/dskopts
/usr/vmsys/OBJECTS/dskopts/Menu.dskopts
/usr/vmsys/OBJECTS/dskopts/Text.ac
/usr/vmsys/OBJECTS/dskopts/Text.csf
/usr/vmsys/OBJECTS/dskopts/Text.fcwarn
/usr/vmsys/OBJECTS/dskopts/Text.fdf
/usr/vmsys/OBJECTS/dskopts/Text.ftfc
/usr/vmsys/OBJECTS/dskopts/Text.fwwarn
/usr/vmsys/OBJECTS/dskopts/Text.rdf
/usr/vmsys/OBJECTS/dskopts/Text.rff
/usr/vmsys/OBJECTS/dskopts/Text.rsf
/usr/vmsys/OBJECTS/dskopts/Text.wdf
/usr/vmsys/OBJECTS/fs
/usr/vmsys/OBJECTS/fs/Form.crtfil
/usr/vmsys/OBJECTS/fs/Form.dsk
/usr/vmsys/OBJECTS/fs/Form.mt
/usr/vmsys/OBJECTS/fs/Menu.filsys
/usr/vmsys/OBJECTS/fs/Menu.seldev
/usr/vmsys/OBJECTS/fs/Text.busy
/usr/vmsys/OBJECTS/fs/Text.cf
/usr/vmsys/OBJECTS/fs/Text.crfcfm
/usr/vmsys/OBJECTS/fs/Text.fmtfd
/usr/vmsys/OBJECTS/fs/Text.hmount
/usr/vmsys/OBJECTS/fs/Text.hmtfd
/usr/vmsys/OBJECTS/fs/Text.hmountw
/usr/vmsys/OBJECTS/fs/Text.hmounttw
/usr/vmsys/OBJECTS/fs/Text.insert
/usr/vmsys/OBJECTS/fs/Text.mininsert
/usr/vmsys/OBJECTS/fs/Text.mount
/usr/vmsys/OBJECTS/fs/Text.mountw
/usr/vmsys/OBJECTS/fs/Text.mtd
/usr/vmsys/OBJECTS/fs/Text.mtdir
/usr/vmsys/OBJECTS/fs/Text.mtwrt
/usr/vmsys/OBJECTS/fs/Text.nmt
/usr/vmsys/OBJECTS/fs/Text.priv
/usr/vmsys/OBJECTS/fs/Text.rename
/usr/vmsys/OBJECTS/fs/Text.umount
/usr/vmsys/OBJECTS/fs/Text.wait
/usr/vmsys/OBJECTS/fs/Text.wrt
/usr/vmsys/OBJECTS/hwset
/usr/vmsys/OBJECTS/hwset/Form.232
/usr/vmsys/OBJECTS/hwset/Form.comp
/usr/vmsys/OBJECTS/hwset/Form.mod
/usr/vmsys/OBJECTS/hwset/Form.other
/usr/vmsys/OBJECTS/hwset/Form.par
/usr/vmsys/OBJECTS/hwset/Form.ps
/usr/vmsys/OBJECTS/hwset/Form.ser
/usr/vmsys/OBJECTS/hwset/Menu.hard
/usr/vmsys/OBJECTS/hwset/Menu.port2
/usr/vmsys/OBJECTS/hwset/Menu.psetup
/usr/vmsys/OBJECTS/hwset/Text.add
/usr/vmsys/OBJECTS/hwset/Text.boot1
/usr/vmsys/OBJECTS/hwset/Text.boot2
/usr/vmsys/OBJECTS/hwset/Text.busy
/usr/vmsys/OBJECTS/hwset/Text.cmpl
/usr/vmsys/OBJECTS/hwset/Text.compcfm
/usr/vmsys/OBJECTS/hwset/Text.conerr
/usr/vmsys/OBJECTS/hwset/Text.conuse
/usr/vmsys/OBJECTS/hwset/Text.inuse
/usr/vmsys/OBJECTS/hwset/Text.noncfm
/usr/vmsys/OBJECTS/hwset/Text.nop
/usr/vmsys/OBJECTS/hwset/Text.noport
/usr/vmsys/OBJECTS/hwset/Text.not
/usr/vmsys/OBJECTS/hwset/Text.privil
/usr/vmsys/OBJECTS/hwset/Text.prterr
/usr/vmsys/OBJECTS/hwset/Text.rscfm
/usr/vmsys/OBJECTS/hwset/Text.sercfm
/usr/vmsys/OBJECTS/hwset/Text.set
/usr/vmsys/OBJECTS/hwset/Text.termcfm
/usr/vmsys/OBJECTS/hwset/Text.tscfm
/usr/vmsys/OBJECTS/logins
/usr/vmsys/OBJECTS/logins/Form.add
/usr/vmsys/OBJECTS/logins/Form.change
Appendix: Installation Diskette Files

/usr/vmsys/OBJECTS/logins/Form.delete
/usr/vmsys/OBJECTS/logins/Form.delfile
/usr/vmsys/OBJECTS/logins/Form.display
/usr/vmsys/OBJECTS/logins/Menu.logins
/usr/vmsys/OBJECTS/logins/Text.addcfm
/usr/vmsys/OBJECTS/logins/Text.addpriv
/usr/vmsys/OBJECTS/logins/Text.chgcfm
/usr/vmsys/OBJECTS/logins/Text.chgpriv
/usr/vmsys/OBJECTS/logins/Text.delcfm
/usr/vmsys/OBJECTS/logins/Text.delpriv
/usr/vmsys/OBJECTS/logins/Text.privil
/usr/vmsys/OBJECTS/mailset
/usr/vmsys/OBJECTS/mailset/Form.mail1
/usr/vmsys/OBJECTS/mailset/Form.maila
/usr/vmsys/OBJECTS/mailset/Form.mailc
/usr/vmsys/OBJECTS/mailset/Form.maild
/usr/vmsys/OBJECTS/mailset/Form.maildel
/usr/vmsys/OBJECTS/mailset/Form.new
/usr/vmsys/OBJECTS/mailset/Menu.mail
/usr/vmsys/OBJECTS/mailset/Menu.mailot
/usr/vmsys/OBJECTS/mailset/Text.a1
/usr/vmsys/OBJECTS/mailset/Text.c1
/usr/vmsys/OBJECTS/mailset/Text.del
/usr/vmsys/OBJECTS/mailset/Text.mail2
/usr/vmsys/OBJECTS/mailset/Text.malpriv
/usr/vmsys/OBJECTS/mailset/Text.new2
/usr/vmsys/OBJECTS/mailset/Text.nodel
/usr/vmsys/OBJECTS/mailset/Text.none
/usr/vmsys/OBJECTS/mailset/Text.noned
/usr/vmsys/OBJECTS/prt
/usr/vmsys/OBJECTS/prt/Form.prst
/usr/vmsys/OBJECTS/prt/Menu.propt
/usr/vmsys/OBJECTS/prt/Menu.queue
/usr/vmsys/OBJECTS/prt/Text.pqcfm
/usr/vmsys/OBJECTS/prt/Text.privil
/usr/vmsys/OBJECTS/prt/Text.restart
/usr/vmsys/OBJECTS/prt/Text.rsterr
/usr/vmsys/OBJECTS/prt/Text.status
/usr/vmsys/OBJECTS/restore
/usr/vmsys/OBJECTS/restore/Menu.restore
Appendix: Installation Diskette Files

/usr/vmsys/OBJECTS/restore/Text.privil
/usr/vmsys/OBJECTS/restore/private
/usr/vmsys/OBJECTS/restore/private/Form.overwr
/usr/vmsys/OBJECTS/restore/private/Form.overwr2
/usr/vmsys/OBJECTS/restore/private/Menu.device
/usr/vmsys/OBJECTS/restore/private/Menu.device2
/usr/vmsys/OBJECTS/restore/private/Menu.private
/usr/vmsys/OBJECTS/restore/private/Menu.select
/usr/vmsys/OBJECTS/restore/system
/usr/vmsys/OBJECTS/restore/system/Form.overwr
/usr/vmsys/OBJECTS/restore/system/Form.overwr2
/usr/vmsys/OBJECTS/restore/system/Menu.device
/usr/vmsys/OBJECTS/restore/system/Menu.device2
/usr/vmsys/OBJECTS/restore/system/Menu.select
/usr/vmsys/OBJECTS/restore/system/Menu.system
/usr/vmsys/OBJECTS/shutdown
/usr/vmsys/OBJECTS/shutdown/Form.grace
/usr/vmsys/OBJECTS/shutdown/Text.privil
/usr/vmsys/OBJECTS/shutdown/Text.privil2
/usr/vmsys/OBJECTS/shutdown/Text.shut
/usr/vmsys/admin
/usr/vmsys/admin/.chkuser
/usr/vmsys/admin/Pclear.sh
/usr/vmsys/admin/Shutdown
/usr/vmsys/admin/backup
/usr/vmsys/admin/backup/dirchk
/usr/vmsys/admin/backup/filechk
/usr/vmsys/admin/backup/filemsg
/usr/vmsys/admin/backup/owrchk
/usr/vmsys/admin/backup/runback
/usr/vmsys/admin/backup/runback2
/usr/vmsys/admin/backup/runback3
/usr/vmsys/admin/backup/runback4
/usr/vmsys/admin/backup/runstore
/usr/vmsys/admin/backup/runstore2
/usr/vmsys/admin/backup/setquote
/usr/vmsys/admin/backup/uidchk
/usr/vmsys/admin/date
/usr/vmsys/admin/date/Date
/usr/vmsys/admin/date/dt
Appendix: Installation Diskette Files

/usr/vmsys/admin/date/hours
/usr/vmsys/admin/dskopts
/usr/vmsys/admin/dskopts/Ccopyin.sh
/usr/vmsys/admin/dskopts/Ccopyout.sh
/usr/vmsys/admin/dskopts/Cdensity
/usr/vmsys/admin/dskopts/Cfwrite
/usr/vmsys/admin/dskopts/Fcopyin.sh
/usr/vmsys/admin/dskopts/Fcopyout.sh
/usr/vmsys/admin/dskopts/Chkmnt.sh
/usr/vmsys/admin/errmsg
/usr/vmsys/admin/filsys
/usr/vmsys/admin/filsys/diskumount
/usr/vmsys/admin/filsys/hcheck
/usr/vmsys/admin/filsys/labelfsname
/usr/vmsys/admin/filsys/makefsys
/usr/vmsys/admin/filsys/mname
/usr/vmsys/admin/filsys/mountfsys
/usr/vmsys/admin/filsys/part
/usr/vmsys/admin/filsys/part2
/usr/vmsys/admin/filsys/permit.ck
/usr/vmsys/admin/filsys/unmountfsys
/usr/vmsys/admin/hw
/usr/vmsys/admin/hw/Printers.sh
/usr/vmsys/admin/hw/RS232.sh
/usr/vmsys/admin/hw/RSfree.sh
/usr/vmsys/admin/hw/chkcon
/usr/vmsys/admin/hw/chkport
/usr/vmsys/admin/hw/chktype
/usr/vmsys/admin/hw/dvc
/usr/vmsys/admin/hw/getname
/usr/vmsys/admin/hw/mkdvc
/usr/vmsys/admin/hw/modlist
/usr/vmsys/admin/hw/plist
/usr/vmsys/admin/hw/pnum
/usr/vmsys/admin/hw/port2ad
/usr/vmsys/admin/hw/port2rm
/usr/vmsys/admin/hw/pports
/usr/vmsys/admin/hw/slist
/usr/vmsys/admin/hw/speed
Appendix: Installation Diskette Files

/usr/vmsys/admin/hw/sports
/usr/vmsys/admin/hw/used
/usr/vmsys/admin/logins
/usr/vmsys/admin/logins/cdlog
/usr/vmsys/admin/logins/cglog
/usr/vmsys/admin/logins/chgfull
/usr/vmsys/admin/logins/chguser
/usr/vmsys/admin/logins/confirm
/usr/vmsys/admin/logins/delcfm
/usr/vmsys/admin/logins/dellog
/usr/vmsys/admin/logins/full
/usr/vmsys/admin/logins/homechk
/usr/vmsys/admin/logins/homedir
/usr/vmsys/admin/logins/lgid
/usr/vmsys/admin/logins/lgname
/usr/vmsys/admin/logins/privgid
/usr/vmsys/admin/logins/seeklg
/usr/vmsys/admin/logins/userid
/usr/vmsys/admin/mailset
/usr/vmsys/admin/mailset/check.date
/usr/vmsys/admin/mailset/chg.Systems
/usr/vmsys/admin/mailset/looksys
/usr/vmsys/admin/mailset/mail.check
/usr/vmsys/admin/mailset/nuucp.pwd
/usr/vmsys/admin/pcheck
/usr/vmsys/admin/prtq
/usr/vmsys/admin/prtstat
/usr/vmsys/admin/qcheck
/usr/vmsys/admin/qpermit
/usr/vmsys/admin/rst.name
/usr/vmsys/admin/sysinfo
/usr/vmsys/admsets
/usr/vmsys/admsets/base-adm
/usr/vmsys/bin
/usr/vmsys/bin/adm
/usr/vmsys/bin/fmli
/usr/vmsys/bin/initial
/usr/vmsys/bin/suspend
Appendix: Installation Diskette Files

Editing Package, Diskette 1 of 1

/Size  /hlistb
/vi     /hlista
/unpack  /spell
/spellprog  /dircmp
/spellin  /diff3
/hashmake  /spellhist
/hashcheck  /Remove
/exrecover  /Install
/expreserve  /Files
/diff3prog  /Name
/uniq
/split
/sdiff
/pack
/nl
/newform
/fgrep
/egrep
/deroff
/ctags
/csplit
/comm
/col
/bfs
/bdiff
/sum
/od
/paste
/touch
/tail
/join
/tr
/pg
/cut
/exstrings
/compress
/hstop
Network Support Utilities, Diskette 1 of 1

/Size
/cloned/Master
/cloned/System
/cloned/Driver.o
/log/Master
/log/System
/log/Node
/log/Space.c
/log/Driver.o
/timod/Master
/timod/System
/timod/Space.c
/timod/Driver.o
/tirdwr/Master
/tirdwr/System
/tirdwr/Space.c
/tirdwr/Driver.o
/libnsL_s.a
/listen
/listen.h
/libnls.a
/nlsadmin
/strace
/strerr
/strclean
/Files
/Name
/Install
/Remove
/tiuser.h
Remote File Sharing, Diskette 1 of 1

/Size
/du/System
/du/Space.c
/du/Driver.o
/dufst/Master
/dufst/System
/dufst/Mfsys
/dufst/Sfsys
/dufst/Driver.o
/sp/Master
/sp/System
/sp/Node
/sp/Space.c
/sp/Driver.o
/adv
/dname
/fumount
/fusage
/idload
/nserve
/nsquery
/rfadmin
/rfpasswd
/rfsetup
/rfstart
/rfstop
/rfadmin
/rfudaemon
/rmntstat
/rmount
/rmountall
/rmountall
/unadv
/fmounts
/rfs
/rmounts
/adv.init

/files
/Name
/Install
/Remove
Remote Terminal Package, Diskette 1 of 1

/install
/install/Rlist
/install/UNINSTALL
/install/INSTALL
/install/setup
/install/adds.ti
/install/annarbor.ti
/install/ansi.ti
/install/att.ti
/install/beehive.ti
/install/cdc.ti
/install/colorsctan.ti
/install/contel.ti
/install/datamedia.ti
/install/dec.ti
/install/diablo.ti
/install/fortune.ti
/install/general.ti
/install/hardcopy.ti
/install/hazeltine.ti
/install/hds.ti
/install/heath.ti
/install/homebrew.ti
/install/hp.ti
/install/lis.ti
/install/microterm.ti
/install/misc.ti
/install/pc.ti
/install/perkinelmer.ti
/install/print.ti
/install/special.ti
/install/sperry.ti
/install/tekntronix.ti
/install/teleray.ti
/install/television.ti
/install/ti.ti
/install/tymshare.ti
/install/visual.ti
/new
/new/usr
/new/usr/lib
/new/usr/lib/terminfo
/new/usr/lib/tabset
/new/usr/lib/tabset/3101
/new/usr/lib/tabset/beehive
/new/usr/lib/tabset/std
/new/usr/lib/tabset/teleray
/new/usr/lib/tabset/vt100
/new/usr/lib/tabset/xerox1720
/new/usr/options
/new/usr/options/terminfo.name

Installation Diskette Files A-27
Appendix: Installation Diskette Files

Security Administration Package, Diskette 1 of 1

/Size
/makekey
/crypt
/libcrypt.a
/Remove
/Files
/Install
/Name
Appendix:  Installation Diskette Files

Two Kilobyte File System Package, Diskette 1 of 1

/Size
/Driver.o
/Master
/Mfsys
/Sfsys
/System
/Space.c
/Stubs.c
/Name
/Files
/Install
/Remove
/fsba
Appendix B: Documentation Updates

Documentation Updates

This update involves the following actions:

1. **ACTION:** Replace existing pages 2-13 through 2-26 with the new pages.

2. **ACTION:** Replace existing pages 4-31 through 4-34 with the new pages.

3. **ACTION:** Replace existing pages 4-73 and 4-74 with the new pages.

4. **ACTION:** Replace existing pages 4-165 and 4-166 with the new pages.

5. **ACTION:** Replace existing pages 4-171 through 4-174 with the new pages.

6. **ACTION:** Replace existing pages 6-23 through 6-26 with the new pages.

7. **ACTION:** Replace existing pages 8-39 and 8-40 with the new pages.

8. **ACTION:** Replace existing pages 9-11 through 9-14 with the new pages.
the UNIX system partition is being scanned, you will see the following message:

Checking cylinder: nnn

where "nnn" is updated for each cylinder scanned.

5. If, during the identification of defective blocks, the table overflows, you will see the following message, the installation aborts, and you will be left at the UNIX system prompt:

Error: Your UNIX System partition has too many bad blocks. A UNIX System cannot be installed on it. Please restart the installation procedure by typing "INSTALL" at the UNIX System prompt and repartition your hard disk so that the UNIX System partition is located elsewhere.

Installation aborted.

If this occurs, restart the installation process by typing INSTALL. When you perform the fdisk procedure ("Partition the Hard Disk") on Page 2-6, manually place the UNIX system partition somewhere else. If this does not clear the problem, contact the manufacturer regarding the integrity of the hard disk.
Create UNIX System File Systems

1. When the surface analysis is completed, the system calculates the optimal amount of space on your hard disk for swap, user, and/or root(/) file systems.

You will see a message that begins as follows:

The UNIX System partition has nnn cylinders assigned to it. nn cylinders will be used for alternate sectors. This leaves nnn cylinders (nnnnn bytes) available.

The following seems like a reasonable partitioning of your UNIX System disk space:

NOTE Throughout this procedure, "nnn" depends on the partition size.
2. You will see the file system size selections in the following messages:

   If no separate /usr file system was called for, based on your partition size you will see the following message:

   A combined root/user filesystem of nnn cylinders (nnnnnnn bytes),

   Otherwise, you will see the following message:

   A root filesystem of nnn cylinders (nnnnnnn bytes),
   a user (usr) filesystem of nnn cylinders (nnnnnnn bytes),

   In either case, if an additional /usr2 file system were called for, you will see the following message:

   an extra user filesystem (/usr2) of nnn cylinders (nnnnnnn bytes),

   The message will be completed with:

   and a swap/paging area of nnn cylinders (nnnnnnn bytes).
3. You will see the following prompt message:

   **Is this allocation acceptable to you (y/n)?**

   If this allocation is acceptable, type `y`, strike `Enter` and then proceed to Step 11.

   If not acceptable, type `n` and strike `Enter`.

4. You will see the following message:

   **Do you wish to have separate root and user filesystems (y/n)?**

   If you wish to have separate root and user file systems, type `y` and strike `Enter`.

   If you wish that root and user be combined, type `n` and strike `Enter`.

   **NOTE** With many users and limited hard disk space (less than 68 megabytes), it is advisable to create separate root and user file systems.
5. You will see the following message:

Do you want an additional /usr2 filesystem (y/n)?

Generally, an additional /usr2 file system is not necessary. However, if you wish to have an additional /usr2 file system, type y and strike Enter.

If you do not want the above, type n and strike Enter.

Go on to the next step.
6. You will see the following message:

You will now be given the opportunity to specify the size, in cylinders, of each filesystem. (One megabyte of disk space is approximately nn.n cylinders).

followed by the prompt message:

How many cylinders would you like for swap/paging (1-nnn)?

NOTE In the above message, nnn is the maximum legal size of swap space, calculated as the total space (UNIX system partition less cylinders reserved for alternates) minus 20 megabytes.

Enter the desired parameter and strike Enter. If your answer was not in the given range, you will see the following message and Step 6 will be repeated:

Illegal value: nnn; try again.

If only one file system was selected, you will see the following message:
The remaining nnn cylinders will be assigned to root/usr.

NOTE
In the above message, nnn is the remaining space after swap space is subtracted.

If only one file system was selected, then proceed to Step 9.

7. You will see the following message:

How many cylinders would you like for root (1-nnn)?

NOTE
In the above message, nnn is the maximum legal size for root (/), which is the amount of space remaining after swap space has been subtracted.

Enter the desired parameter and strike [Enter].

If your answer is not in the given range, you will see the following message and Step 7 will be repeated:

Illegal value: nnn; try again.
Install Base System Diskette Number 1

If there is no space left after subtracting the root(\) file system, only one file system shall be used, you will see the following message, then proceed to Step 9.

No space remaining for a user filesystem.
Assuming single root/usr filesystem.

If an additional /usr2 file system was not selected, you will see the following message, then proceed to Step 9.

The remaining nnn cylinders will be assigned to /usr.

If an additional /usr2 file system was selected, you will see the following message:

The remaining nnn cylinders will be assigned to /usr2.
8. You will see the following message:

```
How many cylinders would you like for /usr (1-nnn)?
```

**NOTE** In the above message, nnn is the maximum legal size for /usr, which is the amount of space remaining after root(/) has been subtracted.

Enter the desired parameter and strike `Enter`.

If the answer is not in the given range, you will see the following message and this step will be repeated:

```
Illegal value: nnn, try again.
```

If there is no space after subtracting the /usr file system, only two file systems shall be used, you will see the following message, and the process continued with Step 9:

```
No space remaining for a /usr2 filesystem. Assuming just root and /usr filesystems.
```
Install Base System Diskette Number 1

Otherwise, you will see the following message:

The remaining nnn cylinders will be assigned to /usr2.

9. You will see the following message:

You have specified the following disk allocation:

Return to Step 2.

10. For each selected file system, you will see the following message:

A xxx filesystem will now be created on your hard disk...

NOTE In the above message, xxx is either "root(/)", "/usr", or "/usr2."

Skip Step 11 and go directly to the procedure to "Verify Successful File System Creation."
11. You will see the following message:

UNIX System file system(s) will now be created on your hard disk...
Verify Successful File System Creation

1. If an error was encountered in the creation of any of the UNIX system file system(s), you will see the following message:

   An error has occurred while setting up your hard disk. Strike ENTER to install again.

   If you see this message, strike Enter to install again and go back to the procedure to "Partition Your Hard Disk."

   NOTE If you strike Break, the Abort Message and procedure on Page 2-2 will be deployed.

2. If the mkfs commands were performed successfully, you will see the following message:

   UNIX System file system(s) have been created in your active UNIX System partition.

   A UNIX System will now be installed on your hard disk.....

   Go on to the next step.
Complete Installation of Diskette Number 1

1. The root(1) file system is mounted. If this fails, you will see the following message (up to three times):

   Mounting root file system failed, trying again ...

   If the mount attempt fails three times, you will see the following message:

   Cannot mount the root file system.
   Please notify your AT&T services representative for further assistance.

   If the above occurs, the installation will abort and you will be left in single-user mode with the hard disk unmounted.

2. All files on diskette number 1 are copied to their respective directories on the hard disk.
3. You will see the following message:

Please standby.

When you are prompted to reboot your system, remove the floppy disk from the diskette drive, and strike CTL-ALT-DEL.

Please wait for the prompt.

When the system is ready to be rebooted, you will see the following message:

Reboot the system now.

NOTE: If you strike any key other than Ctrl, Alt, and Del simultaneously, it will be ignored by the system and not echoed.

4. Strike Ctrl, Alt, and Del simultaneously to reboot the system.
Change Password

A password is a code word that should be known only by its creator. The password secures your login so no unauthorized person can enter the computer and have access to your files. Once a password has been assigned, it must be entered with your login when you want to use the computer.

After you have responded to the login: prompt, the password prompt appears on the screen. When you type in your password, it will not appear on the screen. This protects your password from being seen by someone else.

Each password is required to be at least six characters or longer. The password must have two alphabetic characters and at least one numeric character in the first eight characters.

Choose a password that is not common and is hard to guess. Your password should be changed from time to time to safeguard its secrecy.

You can only change passwords for your own login name. See Appendix C for a procedure to change the password for other users.
Change Password

Changing Your Own Password

To change the password that is associated with your login, use the following procedure.

1. Log in using the login name associated with the password you want to change.

2. From the Administration menu, highlight Change Password and strike Enter. The screen clears and the UNIX system passwd command is executed. At the top of the screen, the following message is printed.

Strike Break or Del to return to AT&T Administration without changing your password.

3. When prompted for your current password (Old password:) type the password you used when you logged in.

   CAUTION
   If you strike Del or Ctrl Break before you type your old password, your password will be unchanged. If you strike Del or Ctrl Break after you type your old password, your password will be locked and you will not be able to log on to the system.

4. When prompted for the new password (New password:), enter the new password you want.

   The password you enter will not be displayed on the screen.
   You will receive an error message in the following circumstances:
   • If you enter the old password incorrectly
   • If the new password is not six character long
   • If the new password does not have two alphabetic characters and at least one special character in the first eight
- If the password resembles the login name by being a reverse or circular shift
- If the new password does not differ from the old password by three or more characters
- If the new password includes a space or a ":"
- If you enter the new password incorrectly the second time.

5. When prompted to repeat the new password (Reenter new password:), type your new password again.

If the two password entries are the same, the password is assigned. If the two password entries do not match, the message

   They don't match; try again.
   New password:

appears. If this message appears, type the new password again and then reenter the new password again.

6. After you reenter the new password, you will be prompted to:

   Strike RETURN to continue with AT&T Administration.

   Strike Enter to return to the Administration menu.
Date and Time

The system clock can be changed using the Date and Time feature. You must have system administration privileges to change the date and time. You can use the Date and Time form to display the current setting, change a value as desired, and verify that the appropriate changes have been made.

Before re-setting the date or time, notify all users that the date is being reset. Changing the date and/or time may disrupt make, cron, a compile, or any applications that rely on the current date thus disturbing other users' work. To set the UNIX system clock, use the following procedure.

1. From the Administration menu, highlight Date and Time and strike Enter. The Date and Time form appears as follows:

   AT&T Administration
   Change Date and Time

   Date: Aug 17, 1987
   Time: 1:35
   AM/PM: PM
   Time Zone: Eastern
   Is Daylight Savings time ever used? Yes

   Strike the CHOICES function key to change. Strike SAVE when you complete the form.

   CANCEL  CHOICES  CONT   PREV-FRM   NEXT-FRM

2. Use the arrow keys to move the cursor to the field to be changed.

3. Once the cursor is resting on the field to be changed, strike the CHOICES key. If only two or three choices are available, the choice will toggle when the CHOICES key is stricked. If more than three choices are available for you to enter into each field a pop-up menu will appear. Use the arrow keys to move to the appropriate entry and select by striking Enter.
Mail Setup

The Mail Setup menu item enables you to give your computer a node name so that other systems can send mail to your system and you can send mail to other systems.

To exchange electronic mail or files with another computer, you and the other computer user must set up and exchange some information. In general, you must do the following:

- Physically install the communication line to be used. You must network (connect) the computers together in some fashion such as a:
  
  modem

  direct connection

  data switch.

  You should exchange system names, mail names, passwords, data phone numbers if using a modem, and data communication line speeds.

- Assign your computer a mail name.

- Assign the mail login a password so that only other trusted systems can log in to your computer.

- Provide the users of other systems with your mail name and mail login password.

- Enter information about the computer receiving your mail (information you receive from the other computer administrator/operator).

If you have not installed the Electronic Mail feature, you can still use the basic UNIX system mail feature. Refer to the UNIX System V User Guide in the section "Communication Tutorial" for some instructions in using UNIX system mail. This document will not be included with the documents you are
Mail Setup

provided. To obtain a copy, consult your Documentation Roadmap for ordering information.

CAUTION
If you strike [Del] or [Ctrl] Break while assigning a password, the mail login for your system will be locked and you will not be able to receive mail.

Set Up the Communication Line

Before sending or receiving mail, you must configure the port using the Serial Ports Setup function from the Peripherals Setup menu in this chapter.

Configure the serial port for a modem, computer, or other (e.g., data switch) depending on the physical connection between your system and the other system.

Set Up This System to Receive Mail

To set up your system to receive mail from other systems, use the following procedure:

1. Highlight Mail Setup and strike [Enter] from the Administration menu. The Mail Setup menu appears as follows:
AT&T Administration

Login Name: jas
Full Name: Jane A. Smith
Login ID Number: 101
HOME Directory: /usr/jas
System Administration Privilege: No

Strike CONT to confirm, or CANCEL to cancel without adding this user.

9. Strike CONT if the information is correct. If the information is not correct, strike CANCEL.

10. After you strike CONT, you will be prompted for a password. The screen clears and appears as follows:
**User Logins**

To return to the interface without changing the password, strike `Break` or `Del`.

New password:

If you strike `Del` or `Ctrl Break` while assigning a password, the login will be locked and the user will not be able to log on to the system using the login.

Respond to the prompts to assign a password for the new user login name. When you have assigned a password, you will be prompted as follows:

Strike RETURN to continue.

Strike `Enter`.

A password is required for every login. If you do not provide a password, the users login will not be installed.

11. After you provide a password, you will return to the User Logins menu.
Delete User Logins

When you no longer need a login, it should be removed.

1. From the Administration menu, highlight User Logins and strike [Enter].

2. From the User Logins menu, highlight Delete and strike [Enter].

NOTE

A validation check determines if you have special system administration privileges. If you do not, you will receive a warning message. You must have special system administration privileges to delete a login even if it is your own.

The Delete Login Name form appears as follows:

AT&T Administration

Delete Login Name

Login Name: jab

Strike the CHOICES function key. Strike SAVE when you complete the form.

CANCEL CHOICES CONT PREV-FRM NEXT-FRM

3. While the cursor is resting on the Login Name field, strike CHOICES and select the desired login from the pop-up menu of user logins. Move to an item with the arrow keys and strike the [Enter] key to select. The login to be deleted is now entered into the Login Name field.
You can also type in the login name that you want to delete. If you type the login name incorrectly, the following message appears:

That is not a valid login name.
Strike CHOICES for valid choices.

If the user you select is currently logged on, the deletion is denied and the following message appears:

"username" is currently logged in.
You may not delete this user at this time.

4. When you delete a user login name, you must decide if you want to remove all the files from the user's HOME directory. The Remove Files of <user login> form appears as follows:

AT&T Administration

Remove Files of jab

Should the files in /usr/jab
be removed? Yes

Strike the CHOICES function key. Strike SAVE when you complete the form.
5. The default decision is Yes. Strike CHOICES (Yes or No) until your decision appears (toggle). When you complete the form, strike SAVE.

If you responded Yes to delete the files the following confirm frame appears:

```
AT&T Administration

Confirm Delete jab

User jab will be deleted and jab's files will be moved to /lost+found.

Strike CONT to confirm, or strike CANCEL to cancel delete user.

CANCEL CONT PREV-FRM NEXT-FRM
```

The files will be stored temporarily in /lost+found. Then they will be deleted.

Responding "No" to delete the files causes the following confirm frame to appear:
User Logins

AT&T Administration

Confirm Delete jab

User jab will be deleted, but jab files will not be deleted.

Strike CONT to confirm or strike CANCEL to cancel delete user.

CANCEL CONT PREV-FRM NEXT-FRM

6. Strike CONT if you’re sure you want to delete the login. If you do not want to delete the login, strike CANCEL.

When you strike CONT the Confirm message closes and the User Logins menu becomes active. You can remove another login if desired or go to another menu.
Recovery From Major Hard Disk Damage

When the file system becomes corrupted to the point where the system is inoperable, try the following:

1. Insert the first floppy disk of the Foundation Set into the floppy disk drive and strike RESET.

2. When you see the message asking if you are ready to install the UNIX system, strike [Ctrl] and [Break] at the same time.

3. Run `fsck` from the `root` prompt by typing:

```
# /etc/fsck /dev/rdsk/0s1  Enter
```

The `fsck` command will either run with no errors or will request action from the user on repairing the file system. Most of the time answering yes to the questions ask by `fsck` will be sufficient, but be aware this could remove some files.

4. Remove diskette and reboot the system by striking RESET.
Bad Block Handling

Recovery of the UNIX System

There may be a time when booting up the computer you will see the message /unix is missing or corrupted. If this should occur, you'll need to replace /unix with the default /unix. When /unix is corrupted, the results are unpredictable. In either case, try the following:

1. Insert the first floppy disk of the Base Foundation Set into the floppy disk drive and strike RESET.
2. When you see the message asking if you are ready to install the UNIX system, break out by striking Ctrl Break at the same time.
3. Run fsck from the root prompt by typing:
   
   # /etc/fsck /dev/rdsk/0s1
   
   4. Mount the device 0s1 by typing:
      
      # /etc/mount /dev/dsk/0s1 /mnt
      
   5. Copy the /unix directory by typing:
      
      # cp /unix /mnt/unix
      
   6. Unmount the device 0s1 by typing:
      
      # /etc/umount /dev/dsk/0s1
      
   7. Reboot the system by striking RESET.

**Warning:** When you get the UNIX system prompt, make sure you are logged in as root. You should then reinstall all drivers previously installed. This can be done via the idbuild command. See the manual page idbuild in the UNIX system V Release 3.1 User's/System Administrator's Reference Manual.

Alternate Recovery of the UNIX System

If you have added device drivers or changed configuration, you may want to use this alternate recovery procedure. You can copy /unix to /unix.orig and reboot from /unix.orig. The advantage of doing this is that you do not have to reinstall all drivers previously installed. The disadvantage is that /unix.orig will occupy additional disk space.
If you see the message `/unix` is missing or corrupted, replace `/unix` with the backup `/unix.orig`. When `/unix` is corrupted, try the following:

1. Insert the first floppy disk of the Base Foundation Set into the floppy disk drive and strike RESET.

2. When you see the message asking if you are ready to install the UNIX system, break out by striking Ctrl Break at the same time.

3. Run `fsck` from the root prompt by typing:
   ```
   # /etc/fsck /dev/rdsk/0s1 Enter
   ```

4. Mount the device `0s1` by typing:
   ```
   # /etc/mount /dev/dsk/0s1 /mnt Enter
   ```

5. Copy `/unix.orig` to `/unix` by typing:
   ```
   # cp /unix /mnt/unix Enter
   ```

6. Unmount the device `0s1` by typing:
   ```
   # /etc/umount /dev/dsk/0s1 Enter
   ```

7. Reboot the system by striking HARDWARE RESET. When the boot prompt is
Creating Backup Copies and Recovering Lost Files

The value of backing up a file is sometimes not appreciated until it's too late and data is lost. Backing up a system takes time, but recovering data that was not backed up takes much longer. The purpose of system backup is to back up your software on floppy disks (or tape) so that you will have it in case data is lost.

Refer to Chapter 4, System Administration for procedures on disk backup and restore.
Poll File

The Poll file (/usr/lib/uucp/Poll) contains information for polling specified machines. Each entry in the Poll file contains the name of the remote machine to call, followed by a TAB character, and finally the hours the machine should be called. The entry:

```
eagle  0 4 8 12 16 20
```

will provide polling of machine eagle every 4 hours.

**NOTE** It should be understood that uudemon.poll does not actually perform the poll, it merely sets up a polling work (C.) file in the spool directory that will be seen by the scheduler, started by uudemon.hour. Refer to the discussion on uudemon.poll.

Maxuxqts File

The Maxuxqts (/usr/lib/uucp/Maxuxqts) file contains an ASCII number to limit the number of simultaneous uuxqt programs running. This file is delivered with a default entry of 2. This may be changed to meet local needs. If there is a lot of traffic from mail, it may be advisable to increase the number of uuxqt programs that will run to reduce the time it takes for the mail to leave your system. However, keep in mind that the load on the system increases with the number of uuxqt programs running.

Maxuuscheds File

The Maxuuscheds (/usr/lib/uucp/Maxuuscheds) file contains an ASCII number to limit the number of simultaneous uusched programs running. Each uusched running will have one uucico associated with it; limiting the number will directly affect the load on the system. The limit should be less than the number of outgoing lines used by UUCP (a smaller number is often desirable). This file is delivered with a default entry of 2. Again, this may be changed to meet the needs of the local system. However, keep in mind that the load on the system increases with the number of uusched programs running.
remote.unknown

The remote.unknown program (/usr/lib/uucp/remote.unknown) is a C program that is executed when a remote site that is not in the Systems file calls in to start a conversation. The C program will append the name of the system and the time of the call to the file /usr/spool/uucp/.Admin/Foreign.

Administrative Tasks

There is a minimum amount of maintenance that must be applied to your computer to keep the files updated, to ensure that the network is running properly, and to track down line problems. When more than one remote machine is involved, the job becomes more difficult because there are more files to update and because users are much less patient when failures occur between machines that are under local control. The uustat program provides you with information about the latest attempts to contact various machines and the age and number of jobs in the queue for remote machines. The following sections describe the routine administrative tasks that must be performed by someone acting as the UUCP administrator or are automatically performed by the UUCP daemons (demons).

The biggest problem in a dialup network like UUCP is dealing with the backlog of jobs that cannot be transmitted to other machines. The following cleanup activities should be routinely performed.
Setting Up RFS

After installing RFS, as described in Chapter 2, Software Installation, the basic RFS configuration and reconfiguration can be handled using the tasks in this section.

These tasks are run from the shell. They should be run initially in the order described.

Once these tasks are completed, go to the “Starting/Stopping RFS” section for information on starting RFS.

Prerequisites

Before you begin setting up RFS, the following must be installed and running: UNIX System V Release 3.1 (or later) software, Remote File Sharing Utilities, Networking Support Utilities, and transport provider software. (See the Remote File Sharing Release Notes and the transport provider manuals that accompany the product for installation instructions.)

You must also log in as root.

Set Node Name

Changing the node name of your computer requires careful coordination with all machines that communicate with yours using Remote File Sharing or other communications packages that rely on node name.

Check to see if your computer’s node name is set to the name you want (uname -n). If it’s not, set it by typing:

    uname -S nodename

You will be asked to type in your computer’s node name. A node name that is valid for Remote File Sharing can consist of up to eight characters of letters (uppercase and lowercase), digits, hyphens (-), and underscores (_). Some networks, such as the STARLAN network, require that every node name in the network be different. Remote File Sharing, however, only requires that every node name in a domain be different.
**Set Up Network Listener**

If you have installed the Networking Support Utilities, the AT&T implementation of the STARLAN network, and Remote File Sharing in the order described in the Chapter 2, Software Installation, you can skip this task. The listener will already be installed and set up to run automatically and Remote File Sharing will be listed as an available service.

If you are using another transport provider, or suspect that your STARLAN network listener is set up improperly, this task will show how to manually set up the listener. In the following example the STARLAN network is used. To set up the listener for other networks compatible with the AT&T Transport Interface, you should replace `starlan` with the name of the network (network specification) you are installing. [For more details, see the `nlsadmin(1M)` manual page.]

To determine if the listener is properly installed and set up for use by RFS, type the following:

```
nlsadmin -v starlan
```

If service code 105 is listed, then the listener is configured to be used for Remote File Sharing.

Run the following commands if the listener is not properly set up. If you run any of these commands and they have already been run, you will receive a message telling you so. This will not harm your listener configuration. Type:

```
nlsadmin -i starlan
```

to initialize the files needed for the listener process for the network specified, in this case `starlan`.

Next, type:

```
nlsadmin -a 105 -c /usr/net/servers/rfs/rfsetup -y "rfsetup" starlan
```

to add the Remote File Sharing service (`rfsetup`) to the list of services available to the `starlan` listener.
Use the following command line to report the status of the **starlan** listener process installed on this machine (ACTIVE or INACTIVE):

```
nlsadmin -x
```

Next, type:

```
nlsadmin -l "nodename.serve" -t "nodename" starlan
```

to register the network addresses of your machine. The listener will listen for requests for these addresses on the network. Only the `-l` address is required by Remote File Sharing. The `-t` address is used only for terminal services and may not be needed on all networks.

To start the listener, type:

```
nlsadmin -s starlan
```

Normally, it will be started automatically when your machine enters multiuser mode (init 2).

**Set the Domain Name**

Set the domain name by typing:

```
dname -D domain
```

where *domain* is replaced by the domain of which your machine will be a member. The domain name must:

- Contain no more than 14 characters
- Consist of any combination of letters (uppercase or lowercase), digits, hyphens, and underscores
- Be different from the name of any other domain used on the network if there is more than one domain on your network.

You can check the current domain name by typing:

```
dname
```
Set the Transport Provider

To identify the network, you must tell Remote File Sharing the network (transport provider) it should use. (In our example, this is starlan for the STARLAN network.)

dname -N starlan

This command indicates the device, relative to the /dev directory, that is used for the transport provider.

Create rfmaster File

The rfmaster file should only be created manually on the primary. If your machine is not the primary, you should skip this task; the rfmaster file for your domain will automatically be placed on your machine the first time you start RFS (rfstart -p primary_addr).

If you are on the primary, you can create an rfmaster file in the /usr/nserv directory using any standard file editor. The contents of this file will define:

- The primary name server for your domain
- Secondary name servers for your domain
- Network addresses for each of these machines.

(See the section on "Multiple Domain Name Service" in this chapter for a description of other information you may want to put into the rfmaster file.)

Here is an example of an rfmaster file for a domain called peanuts, whose primary and secondary name servers' node names are charlie, linus, and lucy. Adding each machine's domain name (peanuts) to its node name, separated by a period, forms its full Remote File Sharing machine name. Each line of the example translates as follows.

- For domain peanuts the primary is peanuts.charlie.
- For domain peanuts a secondary is peanuts.linus.
- For domain peanuts another secondary is peanuts.lucy.
This update involves the following actions:

1. **ACTION:** Replace AT(1) page 1/AT(1) page 2 with the new page.

2. **ACTION:** Replace AWK(1) page 3/BACKUP(1M) page 1 with the new page.

3. **ACTION:** Replace CRASH(1M) page 4/CRASH(1M) page 5 with the new page.

4. **ACTION:** Replace CRASH(1M) page 6/CRASH(1M) page 7 with the new page.

5. **ACTION:** Replace FACTOR(1) page 1/FDISK(1M) page 1 with the new page.

6. **ACTION:** Replace IDLOAD(1M) page 3/IDMKINIT(1M) page 1 with the new page.

7. **ACTION:** Add new page IDMKINIT(1M) page 2/blank side.

8. **ACTION:** Replace MKDIR(1) page 1/MKFS(1M) page 1 (Base System) with the new page.

9. **ACTION:** Replace MKFS(1M) page 2 (Base System)/MKFS(1M) page 1 (2K File System) with the new page.

10. **ACTION:** Replace MKFS(1M) page 2 (2K File System)/MKNOD(1M) page 1 with MKFS(1M) page 2 (2K File System)/MKFS(1M) page 3 (2K File System).
11. **ACTION:** Add new page MKNOD(1M) page 1/blank side.

12. **ACTION:** Replace REMOVEPKG(1) page 1/RESTORE(1M) page 1 with the new page.

13. **ACTION:** Replace TAR(1) page 2/TEE(1) page 1 with the new page.

14. **ACTION:** Replace RTC(7) page 1/STREAMIO(7) page 1 with the new page.
NAME
at, batch – execute commands at a later time

SYNOPSIS
at time [ date ] [ + increment ]
at -r job ...
at -l [ job ... ]
batch

DESCRIPTION
The at and batch commands read commands from standard input to be executed at a later time. at allows you to specify when the commands should be executed, while jobs queued with batch will execute when system load level permits. at may be used with the following options:

-r Removes jobs previously scheduled with at.
-l Reports all jobs scheduled for the invoking user.

Standard output and standard error output are mailed to the user unless they are redirected elsewhere. The shell environment variables, current directory, umask, and ulimit are retained when the commands are executed. Open file descriptors, traps, and priority are lost.

Users are permitted to use at if their name appears in the file /usr/lib/cron/at.allow. If that file does not exist, the file /usr/lib/cron/at.deny is checked to determine if the user should be denied access to at. If neither file exists, the file /usr/lib/cronfat.deny is checked to determine if the user should be denied access to at. If at.deny is empty, global usage is permitted. The allow/deny files consist of one user name per line. These files can only be modified by the super-user.

The time may be specified as 1, 2, or 4 digits. One-and two-digit numbers are taken to be hours, four digits to be hours and minutes. The time may alternately be specified as two numbers separated by a colon, meaning hour:minute. A suffix am or pm may be appended; otherwise a 24-hour clock time is understood. The suffix zulu may be used to indicate GMT. The special names noon, midnight, now, and next are also recognized.

An optional date may be specified as either a month name followed by a day number (and possibly year number preceded by an optional comma) or a day of the week (fully spelled or abbreviated to three characters). Two special “days”, today and tomorrow are recognized. If no date is given, today is assumed if the given hour is greater than the current hour and tomorrow is assumed if it is less. If the given month is less than the current month (and no year is given), next year is assumed.

The optional increment is simply a number suffixed by one of the following: minutes, hours, days, weeks, months, or years. (The singular form is also accepted.)
Thus legitimate commands include:

- `at 0815am Jan 24`
- `at 8:15am Jan 24`
- `at now + 1 day`
- `at 5 pm Friday`

`at` and `batch` write the job number and schedule time to standard error.

The `at -r` command removes jobs previously scheduled by `at` or `batch`. The job number is the number given to you previously by the `at` or `batch` command. You can also get job numbers by typing `at -l`. You can remove only your own jobs unless you are the super-user.

**EXAMPLES**

The `at` and `batch` commands read from standard input the commands to be executed at a later time. `sh(1)` provides a different way of specifying standard input. Within your commands, it may be useful to redirect standard output.

This sequence can be used at a terminal:

```
bash
batch
   sort filename > outfile
   <control-D> (hold down 'control' and depress 'D')
```

This sequence, which demonstrates redirecting standard error to a pipe, is useful in a shell procedure (the sequence of output redirection specifications is significant):

```
bash
batch <<<!
   sort filename 2>&1 > outfile 
   mail loginid
```

To have a job reschedule itself, invoke `at` from within the shell procedure, by including code similar to the following within the shell file:

```
echo "sh shellfile" 1 at 1900 thursday next week
```

**FILES**

- `/usr/lib/cron` main cron directory
- `/usr/lib/cron/at.allow` list of allowed users
- `/usr/lib/cron/at.deny` list of denied users
- `/usr/lib/cron/queue` scheduling information
- `/usr/spool/cron/atjobs` spool area

**SEE ALSO**

- `cron(1M)`, `kill(1)`, `mail(1)`, `nice(1)`, `ps(1)`, `sh(1)`, `sort(1)`.

**DIAGNOSTICS**

Complains about various syntax errors and times out of range.
Add up first column, print sum and average:

```awk
{ s += $1 }
END { print "sum is", s, " average is", s/NR }
```

Print fields in reverse order:

```awk
{ for (i = NF; i > 0; --i) print $i }
```

Print all lines between start/stop pairs:

```awk
/start/, /stop/
```

Print all lines whose first field is different from previous one:

```awk
$1 != prev { print; prev = $1 }
```

Print file, filling in page numbers starting at 5:

```awk
/Page/ { $2 = n++; }
{ print }
```

command line: `awk -f program n=5 input`

SEE ALSO

`grep(1), sed(1), lex(1), printf(3S)` in the *Programmer's Reference Manual.*

BUGS

Input white space is not preserved on output if fields are involved. There are no explicit conversions between numbers and strings. To force an expression to be treated as a number, add 0 to it; to force it to be treated as a string, concatenate the null string (" ") to it.
NAME
backup – performs backup functions

SYNOPSIS
backup [-t] [-p] [-c] [-f] <files> | -u "<user1> [user2]"
     -d <device>

backup -h

DESCRIPTION
-h produces a history of backups. Tells the user when the last complete and incremental/partial backups were done.
-c complete backup. All files changed since the system was installed are backed up.
-p incremental/partial backup. If an incremental/partial backup was done, all files modified since that time are backed up, otherwise all files modified since the last complete backup are backed up. A complete backup must be done before a partial backup.
-f backup files specified by the <files> argument. File names may contain characters to be expanded (i.e., *, .) by the shell. The argument must be in quotes.
-u backup a user’s home directory. All files in the user’s home directory will be backed up. At least one user must be specified but it can be more. The argument must be in quotes if more than one user is specified. If the user name is "all", then all the user’s home directories will be backed up.
-d used to specify the device to be used. It defaults to /dev/rdsk/f0q15d (the 1.2M floppy).
-t used when the device is a tape. This option must be used with the -d option when the tape device is specified.

A complete backup must be done before a partial backup can be done. Raw devices, rather than block devices should always be used. The program can handle multi-volume backups. The program will prompt the user when it is ready for the next media. The program will give you an estimated number of floppies/tapes that will be needed to do the backup. Floppies MUST be formatted before the backup is done. Tapes do not need to be formatted. If backup is done to tape, the tape must be rewound.

SEE ALSO
qt(7).
file [-e] [-w file] [[-p] table_entry ...]
   Alias: f.
   Print the file table.

findaddr [-w file] table slot
   Print the address of slot in table. Only tables available to the size
   function are available to findaddr.

findslot [-w file] virtual_address ...
   Print the table, entry slot number, and offset for the address
   entered. Only tables available to the size function are available to
   findslot.

fs [-w file] [[-p] table_entry ...]
   Print the file system information table.

gdp [-e] [-f] [-w file] [[-p] table_entry ...]
   Print the gift descriptor protocol table.

gdt [-e] [-w file] [[-p] table_entry ...]
   Print the global descriptor table.

help [-w file] function ...
   Print a description of the named function, including syntax and
   aliases.

idt [-e] [-w file] [[-p] table_entry ...]
   Print the interrupt descriptor table.

inode [-e] [-f] [-w file] [[-p] table_entry ...]
   Alias: i.
   Print the inode table, including file system switch information.

kfp [-w file] [value]
   Print the frame pointer for the start of a kernel stack trace. If the
   value argument is supplied, the kfp is set to that value.

lck [-e] [-w file] [[-p] table_entry ...]
   Alias: l.
   Print record-locking information. If the -e option is used or table
   address arguments are given, the record lock list is printed. If no
   argument is entered, information on locks relative to inodes is
   printed.

ldt [-e] [-w file] [-s process] [[-p] table_entry ...]
   Print the local descriptor table for the given process, or for the
   current process if none is given.

linkblk [-e] [-w file] [[-p] table_entry ...]
   Print the linkblk table.

map [-w file] mapname ...
   Print the map structure of the given mapname.

mbfree [-w file]
   Print free streams message block headers.
mblock [-e] [-w filename] [[-p]table_entry ...]
Print allocated streams message block headers.

mode [-w file] [mode]
Set address translation of arguments to virtual (v) or physical (p) mode. If no mode argument is given, the current mode is printed. At the start of a crash session, the mode is virtual.

mount [-e] [-w file] [[-p]table_entry ...]
Alias: m.
Print the mount table.

nm [-w file] symbol ...
Print value and type for the given symbol.

Alias: rd.
Print count values starting at the start address in one of the following formats: character (-c), decimal (-d), hexadecimal (-x), octal (-o), ASCII (-a), or hexadecimal/character (-h), and one of the following modes: long (-l), short (-t), or byte (-b). The default mode for character and ASCII formats is byte; the default mode for decimal, hexadecimal, and octal formats is long. The format -h prints both hexadecimal and character representations of the addresses dumped; no mode needs to be specified. When format or mode is omitted, the previous value is used. At the start of a crash session, the format is hexadecimal and the mode is long. If no count is entered, 1 is assumed.

panic
Print the latest system notices, warnings, and panic messages from the limited circular buffer kept in memory.

pcb [-w file] [process]
Print the process control block (TSS) for the given process. If no arguments are given, the active TSS for the current process is printed.

The page descriptor table of the designated memory section and segment is printed. Alternatively, the page descriptor table starting at the start address for count entries is printed. If no count is entered, 1 is assumed.

pfdat [-e] [-w file] [[-p]table_entry ...]
Print the pfdata table.

proc [-e] [-f] [-w file] [[-p] table_entry ... #procid ...]
or
proc [-f] [-w file] [-r]
Alias: p.
Print the process table. Process table information may be specified in two ways. First, any mixture of table entries and process ids may be entered. Each process id must be preceded by a #.
Alternatively, process table information for runnable processes may be specified with the runnable option (-r). The full option (-f) details most of the information in the process table as well as the region table for that process.

**qrun** [-w file]
Print the list of scheduled streams queues.

**queue** [-e] [-w file] [[-p]table_entry ...]
Print streams queues.

**quit**
Alias: q.
Terminate the crash session.

**rcvd** [-e] [-f] [-w file] [[-p]table_entry ...]
Print the receive descriptor table.

**redirect** [-w file] [-c]

or

**redirect** [-w file] [file]
Used with a file name, redirects output of a crash session to the named file. If no argument is given, the file name to which output is being redirected is printed. Alternatively, the close option (-c) closes the previously set file and redirects output to the standard output.

**region** [-e] [-w file] [[-p]table_entry ...]
Print the region table.

**sdt** [-e] [-w file] [-s process] section

or

**sdt** [-e] [-w file] [-s process] [-p] start_addr [count]
The segment descriptor table for the current process is printed.

**search** [-p] [-w file] [-m mask] [-s process] pattern start_addr count
Print the long words in memory that match pattern, beginning at the start address for count long words. The mask is anded (&) with each memory word and the result compared against the pattern. The mask defaults to 0xffffffff.

**size** [-w file] [-x] [structure_name ...]
Print the size of the designated structure. The (-x) option prints the size in hexadecimal. If no argument is given, a list of the structure names for which sizes are available is printed.

**sndd** [-e] [-f] [-w file] [[-p]table_entry ...]
Print the send descriptor table.

**srmount** [-e] [-w file] [[-p]table_entry ...]
Print the server mount table.

**stack** [-w file] [process]
Alias: s.
Dump stack. If no arguments are entered, the kernel stack for the current process is printed. The interrupt stack and the stack for the
current process are not available on a running system.

stat [-w file]
Print system statistics.

stream [-e] [-f] [-w file] [[-p]table_entry ...]
Print the streams table.

strstat [-w file]
Print streams statistics.

trace [-w file] [-r] [process]
Alias: t.
Print kernel stack trace. The kfp value is used with the -r option.

ts [-w file] virtual_address ...
Print closest text symbol to the designated address.

tty [-e] [-f] [-w file] [-t type [(-p)table_entry ...]]
Valid types: kdb and asy.
Print the tty table. If no arguments are given, the tty table for the
console is printed. If the -t option is used, the table for the single
tty type specified is printed. If no argument follows the type
option, all entries in the table are printed. A single tty entry may be
specified from the start address.

user [-f] [-w file] [process]
Alias: u.
Print the ublock for the designated process.

var [-w file]
Alias: v.
Print the tunable system parameters.

vtop [-w file] [-s process] start_addr ...
Print the physical address translation of the virtual start address.

FILES
/dev/mem system image of currently running system
NAME
factor – obtain the prime factors of a number

SYNOPSIS
factor [ integer ]

DESCRIPTION
When you use factor without an argument, it waits for you to give it an integer. After you give it a positive integer less than or equal to $10^{14}$, it factors the integer, prints its prime factors the proper number of times, and then waits for another integer. factor exits if it encounters a zero or any non-numeric character.

If you invoke factor with an argument, it factors the integer as described above, and then it exits.

The maximum time to factor an integer is proportional to $\sqrt{n}$. factor will take this time when $n$ is prime or the square of a prime.

DIAGNOSTICS
factor prints the error message, "Ouch," for input out of range or for garbage input.
NAME
fdisk – create or modify hard disk partition table

SYNOPSIS
fdisk [device]

DESCRIPTION
This command is used to create and modify the partition table that is put in
the first sector of the hard disk. This table is used by DOS and by the
first-stage bootstrap to identify parts of the disk reserved for different
operating systems, and to identify the partition containing the second-stage
bootstrap (the active partition). The optional argument can be used to
specify the raw device associated with the hard disk; the default value is
/dev/rdsk/c0s0.

The program displays the partition table as it exists on the disk, and then
presents a menu allowing the user to modify the table. The menu, ques­
tions, warnings, and error messages are intended to be self-explanatory.

If there is no partition table on the disk, the user is given the option of
creating a default partitioning or specifying the initial table values. The
default partitioning allows 10% of the disk for MS-DOS and 90% for the
UNIX system, and makes the UNIX system partition active. In either case,
when the initial table is created, fdisk also writes out the first-stage bootstrap
code [see hd(7)] along with the partition table. After the initial table is
created, only the table is changed; the bootstrap is not modified.

Menu Options
The following are the menu options given by the fdisk program:

Create a partition
This option allows the user to create a new partition. The max­
imum number of partitions is 4. The program will ask for the type
of the partition (MS-DOS, UNIX system, or other). It will then ask
for the size of the partition as a percentage of the disk. The user
may also enter the letter c at this point, in which case the program
will ask for the starting cylinder number and size of the partition in
cylinders. If a c is not entered, the program will determine the
starting cylinder number where the partition will fit. In either case,
if the partition would overlap an existing partition, or will not fit, a
message is displayed and the program returns to the original menu.

Change Active (Boot from) partition
This option allows the user to specify the partition where the first­
age bootstrap will look for the second-stage bootstrap, otherwise
known as the active partition.

Delete a partition
This option allows the user to delete a previously created partition.
Note that this will destroy all data in that partition.

Exit This option writes the new version of the table created during this
session with fdisk out to the hard disk, and exits the program.
cannot map by remote name in global blocks.)

NOTE: idload will always output warning messages for map all, since password files always contain multiple administrative user names with the same id number. The first mapping attempt on the id number will succeed; each subsequent attempt will produce a warning.

Remote File Sharing doesn't need to be running to use idload.

EXIT STATUS
On successful completion, idload will produce one or more translation tables and return a successful exit status. If idload fails, the command will return an exit status of zero and not produce a translation table.

ERRORS
If (1) either rules file cannot be found or opened; (2) there are syntax errors in the rules file; (3) there are semantic errors in the rules file; (4) host password or group information could not be found; or (5) the command is not run with super-user privileges, an error message will be sent to standard error. Partial failures will cause a warning message to appear, though the process will continue.

FILES
/etc/passwd
/etc/group
/usr/nserve/auth.info/domain/nodename/[user | group]
/usr/nserve/auth.info/uid.rules
/usr/nserve/auth.info/gid.rules

SEE ALSO
mount(1M).
"Remote File Sharing" chapter of the System Administrator's Guide for detailed information on ID mapping.
NAME
idmkinit – reads files containing specifications

SYNOPSIS
/etc/conf/bin/idmkinit

DESCRIPTION
This command reads the files containing specifications of /etc/inittab entries from /etc/conf/init.d and constructs a new inittab file in /etc/conf/d.d. It returns 0 on success and a positive number on error.

The files in /etc/conf/init.d are copies of the Init modules in device Driver Software Packages (DSP). There is at most one Init file per DSP. Each file contains one line for each inittab entry to be installed. There may be multiple lines (i.e., multiple inittab entries) per file. An inittab entry has the form (the id field is often called the tag):

id:rstate:action:process

The Init module entry must have one of the following forms:

action:process
rstate:action:process
id:rstate:action:process

When idmkinit encounters an entry of the first type, a valid id field will be generated, and an rstate field of 2 (indicating run on init state 2) will be generated. When an entry of the second type is encountered only the id field is prepended. An entry of the third type is incorporated into the new inittab unchanged.

Since add-on inittab entries specify init state 2 for their rstate field most often, an entry of the first type should almost always be used. An entry of the second type may be specified if you need to specify other than state 2. DSP’s should avoid specifying the id field as in the third entry, since other add-on applications or DSPs may have already used the id value you have chosen. The /etc/init program will encounter serious errors if one or more inittab entries contain the same id field.

Idmkinit determines which of the three forms above is being used for the entry by requiring each entry to have a valid action keyword. Valid action values are as follows:

off
respawn
ondemand
once
wait
boot
bootwait
powerfail
powerwait
initdefault
sysinit
The `idmkinit` command is called automatically upon entering init State 2 on the next system reboot after a kernel reconfiguration to establish the correct `/etc/inittab` for the running `/unix` kernel. `Idmkinit` can be called as a user level command to test modification of `inittab` before a DSP is actually built. It is also useful by installation scripts that do not reconfigure the kernel, but need to create `inittab` entries. In this case the `inittab` generated by `idmkinit` must be copied to `/etc/inittab`, and a `telinit q` command must be run to make the new entry take affect.

The command line options are:

- `-o` directory  `Inittab` will be created in the directory specified rather than `/etc/conf/cf.d`.
- `-i` directory  The ID file `init.base`, which normally resides in `/etc/conf/cf.d`, can be found in the directory specified.
- `-e` directory  The Init modules that are usually in `/etc/conf/init.d` can be found in the directory specified.
- `-#`  Print debugging information.

**ERROR MESSAGES**

An exit value of zero indicates success. If an error was encountered, `idmkinit` will exit with a non-zero value, and report an error message. All error messages are designed to be self-explanatory.

**SEE ALSO**

`idbuild(1), idinstall(1m), idmknod(1m), init(1m), inittab(4)` in the *Programmer’s Reference Manual*.
NAME
mkdir – make directories

SYNOPSIS
mkdir [-m mode] [-p] dirame ...

DESCRIPTION
The `mkdir` command creates the named directories in mode 777 [possibly altered by `umask(1)`].

Standard entries in a directory (e.g., the files ., for the directory itself, and .., for its parent) are made automatically. `mkdir` cannot create these entries by name. Creation of a directory requires write permission in the parent directory.

The owner ID and group ID of the new directories are set to the process’s real user ID and group ID, respectively.

Two options apply to `mkdir`:

- `-m` This option allows users to specify the mode to be used for new directories. Choices for modes can be found in `chmod(1)`.

- `-p` With this option, `mkdir` creates `dirname` by creating all the non-existing parent directories first.

EXAMPLE
To create the subdirectory structure `ltr/jd/jan`, type:

```bash
mkdir -p ltr/jd/jan
```

SEE ALSO
`chmod(1), sh(1), rm(1), umask(1)`.


DIAGNOSTICS
The `mkdir` command returns exit code 0 if all directories given in the command line were made successfully. Otherwise, it prints a diagnostic and returns non-zero. An error code is stored in `errno`. 
NAME
mkfs – construct a file system

SYNOPSIS
/etc/mkfs special blocks[:i-nodes] [gap blocks/cyl]
/etc/mkfs special proto [gap blocks/cyl]

DESCRIPTION
The mkfs command constructs a file system by writing on the special file using the values found in the remaining arguments of the command line. The command waits 10 seconds before starting to construct the file system. During this 10-second pause, the command can be aborted by entering a delete (DEL).

If the second argument is a string of digits, the size of the file system is the value of blocks interpreted as a decimal number. This is the number of physical (512-byte) disk blocks the file system will occupy. If the number of i-nodes is not given, the default is the number of logical (1024-byte) blocks divided by 4. mkfs builds a file system with a single empty directory on it. The boot program block (block zero) is left uninitialized.

If the second argument is the name of a file that can be opened, mkfs assumes it to be a prototype file proto, and will take its directions from that file. The prototype file contains tokens separated by spaces or new-lines. A sample prototype specification follows (line numbers have been added to aid in the explanation):

1. /stand/ diskboot
2. 4872 110
3. d---777 3 1
4. usr d---777 3 1
5. sh ---755 3 1 /bin/sh
6. ken d---755 6 1
7. $
8. b0 b---644 3 1 0 0
9. c0 c---644 3 1 0 0
10. $
11. $

Line 1 in the example is the name of a file to be copied onto block zero as the bootstrap program.

Line 2 specifies the number of physical (512-byte) blocks the file system is to occupy and the number of i-nodes in the file system.

Lines 3-9 tell mkfs about files and directories to be included in this file system.

Line 3 specifies the root directory.

Lines 4-6 and 8-9 specifies other directories and files.

The $ on line 7 tells mkfs to end the branch of the file system it is on, and continue from the next higher directory. The $ on lines 10 and 11 end the process, since no additional specifications follow.
File specifications give the mode, the user ID, the group ID, and the initial contents of the file. Valid syntax for the contents field depends on the first character of the mode.

The mode for a file is specified by a 6-character string. The first character specifies the type of the file. The character range is `-bcd` to specify regular, block special, character special and directory files respectively. The second character of the mode is either `u` or `-` to specify set-user-id mode or not. The third is `g` or `-` for the set-group-id mode. The rest of the mode is a 3-digit octal number giving the owner, group, and other read, write, execute permissions [see `chmod(1)`].

Two decimal number tokens come after the mode; they specify the user and group IDs of the owner of the file.

If the file is a regular file, the next token of the specification may be a path name whence the contents and size are copied. If the file is a block or character special file, two decimal numbers follow which give the major and minor device numbers. If the file is a directory, `mkfs` makes the entries and .. and then reads a list of names and (recursively) file specifications for the entries in the directory. As noted above, the scan is terminated with the token `$`.

The final argument in both forms of the command specifies the rotational gap and the number of blocks/cyl. The following gap size values are recommended:

<table>
<thead>
<tr>
<th>CONTROLLER</th>
<th>GAP SIZE</th>
</tr>
</thead>
<tbody>
<tr>
<td>ST506 3:1</td>
<td>2</td>
</tr>
<tr>
<td>ST506 1:1</td>
<td>22</td>
</tr>
<tr>
<td>ESDI 3:1</td>
<td>4</td>
</tr>
<tr>
<td>ESDI 1:1</td>
<td>10</td>
</tr>
</tbody>
</table>

If the gap and blocks/cyl are not specified or are considered illegal values, default values of gap size 7 and 400 blocks/cyl are used.

FILES

/etc/vtoc/*

SEE ALSO

`chmod(1)`,
`dir(4), fs(4)` in the *Programmer's Reference Manual*.

BUGS

With a prototype file, it is not possible to copy in a file larger than 64K bytes, nor is there a way to specify links. The maximum number of i-nodes configurable is 65500.
NAME
mbks – construct a file system

SYNOPSIS
/etc/mkfs special blocks[:i-nodes] [gap blocks/cyl] [-b blocksize]
/etc/mkfs special proto [gap blocks/cyl] [-b blocksize]

DESCRIPTION

mbks constructs a file system by writing on the special file using the values
found in the remaining arguments of the command line. The command
waits 10 seconds before starting to construct the file system. During this
10-second pause the command can be aborted by entering a delete (DEL).

The -b blocksize option specifies the logical block size for the file system.
The logical block size is the number of bytes read or written by the operat­
ing system in a single I/O operation. Valid values for blocksize are 512,
1024, and 2048. The default is 1024. A block size of 2048 may be chosen
only if the 2K file system package is installed. If the -b option is used it
must appear last on the command line.

If the second argument to mbks is a string of digits, the size of the file sys­
tem is the value of blocks interpreted as a decimal number. This is the
number of physical (512 byte) disk blocks the file system will occupy. If the
number of i-nodes is not given, the default is approximately the number of
logical blocks divided by 4. mbks builds a file system with a single empty
directory on it. The boot program block (block zero) is left uninitialized.

If the second argument is the name of a file that can be opened, mbks
assumes it to be a prototype file proto, and will take its directions from that
file. The prototype file contains tokens separated by spaces or new-lines. A
sample prototype specification follows (line numbers have been added to
aid in the explanation):

1. /stand/ diskboot
2. 4872 110
3. d---777 3 1
4. usr d---777 3 1
5. sh ----755 3 1 /bin/sh
6. ken d---755 6 1
7. $
8. b0 b---644 3 1 0 0
9. c0 c---644 3 1 0 0
10. $
11. $

Line 1 in the example is the name of a file to be copied onto block zero as
the bootstrap program.

Line 2 specifies the number of physical (512 byte) blocks the file system is
to occupy and the number of i-nodes in the file system.

Lines 3-9 tell mbks about files and directories to be included in this file sys­
tem.
Line 3 specifies the root directory.

Lines 4-6 and 8-9 specifies other directories and files.

The $ on line 7 tells `mkfs` to end the branch of the file system it is on, and continue from the next higher directory. The $ on lines 10 and 11 end the process, since no additional specifications follow.

File specifications give the mode, the user ID, the group ID, and the initial contents of the file. Valid syntax for the contents field depends on the first character of the mode.

The mode for a file is specified by a 6-character string. The first character specifies the type of the file. The character range is `-bcd` to specify regular, block special, character special and directory files respectively. The second character of the mode is either `u` or `-` to specify set-user-id mode or not. The third is `g` or `-` for the set-group-id mode. The rest of the mode is a 3 digit octal number giving the owner, group, and other read, write, execute permissions [see `chmod(1)`].

Two decimal number tokens come after the mode; they specify the user and group IDs of the owner of the file.

If the file is a regular file, the next token of the specification may be a path name whence the contents and size are copied. If the file is a block or character special file, two decimal numbers follow which give the major and minor device numbers. If the file is a directory, `mkfs` makes the entries `. and ..` and then reads a list of names and (recursively) file specifications for the entries in the directory. As noted above, the scan is terminated with the token $.

The `gap` and `blocks/cyl` arguments in both forms of the command specifies the rotational gap and the number of blocks/cylinder. The following gap size values are recommended:

<table>
<thead>
<tr>
<th>CONTROLLER</th>
<th>FILESYSTEM SIZE</th>
<th>GAP SIZE</th>
</tr>
</thead>
<tbody>
<tr>
<td>ST506 3:1</td>
<td>1K</td>
<td>2</td>
</tr>
<tr>
<td>ST506 1:1</td>
<td>1K</td>
<td>22</td>
</tr>
<tr>
<td>ESDI 3:1</td>
<td>1K</td>
<td>4</td>
</tr>
<tr>
<td>ESDI 1:1</td>
<td>1K</td>
<td>10</td>
</tr>
<tr>
<td>ST506 3:1</td>
<td>2K</td>
<td>4</td>
</tr>
<tr>
<td>ST506 1:1</td>
<td>2K</td>
<td>24</td>
</tr>
<tr>
<td>ESDI 3:1</td>
<td>2K</td>
<td>40</td>
</tr>
<tr>
<td>ESDI 1:1</td>
<td>2K</td>
<td>12</td>
</tr>
</tbody>
</table>

FILES

/etc/vtoc/*

SEE ALSO

`chmod(1)`,
BUGS
With a prototype file, it is not possible to copy in a file larger than 64K bytes, nor is there a way to specify links. The maximum number of i-nodes configurable is 65500.
NAME
mknod - build special file

SYNOPSIS
/etc/mknod name b | c major minor
/etc/mknod name p

DESCRIPTION
The mknod command makes a directory entry and corresponding i-node for a special file.

The first argument is the name of the entry. The UNIX System convention is to keep such files in the /dev directory.

In the first case, the second argument is b if the special file is block-type (disks, tape) or c if it is character-type (other devices). The last two arguments are numbers specifying the major device type and the minor device (e.g., unit, drive, or line number). They may be either decimal or octal. The assignment of major device numbers is specific to each system. The information is contained in the system source file conf.c. You must be the super-user to use this form of the command.

The second case is the form of the mknod that is used to create FIFO's (a.k.a named pipes).

WARNING
If mknod is used to create a device in a remote directory (Remote File Sharing), the major and minor device numbers are interpreted by the server.

SEE ALSO
NAME
    removepkg – remove installed package

SYNOPSIS
    removepkg [software_package]

DESCRIPTION
    The removepkg command will remove the software package specified as an arguement to removepkg or will remove the software package the user selects if no argument is given to removepkg.

    If an argument is specified, removepkg will search the list of previously installed packages and remove the first name it matches. If no name is matched, the user is given an error message.

    If no argument is specified, removepkg will query the user, via a menu, which package to remove.

    You will need to be root to remove some packages.

LIMITATIONS
    You must envoke removepkg on the console.

SEE ALSO
    displaypkg(1), installpkg(1).
NAME
  restore - restore file to original directory

SYNOPSIS
  restore [-c] [-i] [-o] [-t] [-d <device>] [pattern [pattern]...]

DESCRIPTION
- `c`  complete restore. All files on the tape are restored.
- `i`  gets the index file off of the medium. This only works when the archive was created using backup. The output is a list of all the files on the medium. No files are actually restored.
- `o`  overwrite existing files. If the file being restored already exists it will not be restored unless this option is specified.
- `t`  indicates that the tape device is to be used. MUST be used with the `-d` option when restoring from tape.
- `d`  `<device>` is the raw device to be used. It defaults to `/dev/rdsk/f0q15d` (the 1.2M floppy).

When doing a restore, one or more patterns can be specified. These patterns are matched against the files on the tape. When a match is found, the file is restored. Since backups are done using full pathnames, the file is restored to its original directory. Metacharacters can be used to match multiple files. The patterns should be in quotes to prevent the characters from being expanded before they are passed to the command. If no patterns are specified, it defaults to restoring all files. If a pattern does not match any file on the tape, a message is printed.

When end of medium is reached, the user is prompted for the next media. The user can exit at this point by typing "q". (This may cause files to be corrupted if a file happens to span a medium.) In general, quitting in the middle is not a good idea.

If the file already exists and an attempt is made to restore it without the `-o` option, the file name will be printed on the screen followed by a question mark. This file will not be restored.

In order for multi-volume restores to work correctly, the raw device MUST be used.

SEE ALSO
  qt(7), sh(1).
the function letter. With the t function, v gives more information about the tape entries than just the name.

w What. This causes tar to print the action to be taken, followed by the name of the file, and then wait for the user's confirmation. If a word beginning with y is given, the action is performed. Any other input means "no". This is not valid with the t key.

f File. This causes tar to use the device argument as the name of the archive instead of /dev/rmt/0s0. If the name of the file is -, tar writes to the standard output or reads from the standard input, whichever is appropriate. Thus, tar can be used as the head or tail of a pipeline. tar can also be used to move hierarchies with the command:

    cd fromdir; tar cf - | (cd todir; tar xf -)

l Link. This tells tar to complain if it cannot resolve all of the links to the files being dumped. If l is not specified, no error messages are printed.

m Modify. This tells tar not to restore the modification times. The modification time of the file will be the time of extraction.

o Ownership. This causes extracted files to take on the user and group identifier of the user running the program, rather than those on tape. This is only valid with the x key.

FILES
/dev/rmt/*

SEE ALSO
    cpio(1), ls(1).

DIAGNOSTICS
    Complaints about bad key characters and tape read/write errors.
    Complaints if enough memory is not available to hold the link tables.

BUGS
    There is no way to ask for the n-th occurrence of a file.
    Tape errors are handled ungracefully.
    The u option can be slow.
    The current limit on file name length is 100 characters.
    tar doesn't copy empty directories or special files.
NAME
tee – pipe fitting

SYNOPSIS
tee [ -i ] [ -a ] [ file ] ...

DESCRIPTION
The tee command transcribes the standard input to the standard output and makes copies in the files. The
-i ignore interrupts;
-a causes the output to be appended to the files rather than overwriting them.
NAME
rtc – real time clock interface

DESCRIPTION
The rtc driver supports the real time clock chip, allowing it to be set with
the correct local time, and allowing the time to be read from the chip.

Ioctl Calls
RTCRTIME
This call is used to read the local time from the real time clock chip.
The argument to the ioctl is the address of a buffer of RTCNREG
unsigned characters (RTCNREG is defined as <sys/rtc.h>). The
ioctl will fill in the buffer with the contents of the chip registers.
Currently, RTCNREG is 14, and the meanings of the byte registers
are as follows:

<table>
<thead>
<tr>
<th>Register</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Seconds</td>
</tr>
<tr>
<td>1</td>
<td>Second alarm</td>
</tr>
<tr>
<td>2</td>
<td>Minutes</td>
</tr>
<tr>
<td>3</td>
<td>Minute alarm</td>
</tr>
<tr>
<td>4</td>
<td>Hours</td>
</tr>
<tr>
<td>5</td>
<td>Hour alarm</td>
</tr>
<tr>
<td>6</td>
<td>Day of week</td>
</tr>
<tr>
<td>7</td>
<td>Date of month</td>
</tr>
<tr>
<td>8</td>
<td>Month</td>
</tr>
<tr>
<td>9</td>
<td>Year</td>
</tr>
<tr>
<td>A</td>
<td>Status register A</td>
</tr>
<tr>
<td>B</td>
<td>Status register B</td>
</tr>
<tr>
<td>C</td>
<td>Status register C</td>
</tr>
<tr>
<td>D</td>
<td>Status register D</td>
</tr>
</tbody>
</table>

For further information on the functions of these registers, see your
hardware technical reference manual.

RTCSTIME
This call is used to set the time into the real time clock chip. The
argument to the ioctl is the address of a buffer of RTCNREGP
unsigned characters (RTCNREGP is defined as <sys/rtc.h>). These
bytes should be the desired chip register contents. Currently,
RTCNREGP is 10, representing registers 0–9 as shown above.
Note that only the superuser may open the real time clock device
for writing, and that the RTCSTIME ioctl will fail for any other
than the superuser.

FILES
/dev/rtc
NAME
streamio — STREAMS ioctl commands

SYNOPSIS
#include <stropts.h>
int ioctl (fildes, command, arg)
int fildes, command;

DESCRIPTION
STREAMS [see intro(2)] ioctl commands are a subset of ioctl(2) system calls
which perform a variety of control functions on streams. The arguments
command and arg are passed to the file designated by fildes and are inter­
preted by the stream head. Certain combinations of these arguments may be
passed to a module or driver in the stream.

fildes is an open file descriptor that refers to a stream. command determines
the control function to be performed as described below. arg represents
additional information that is needed by this command. The type of arg
depends upon the command, but it is generally an integer or a pointer to a
command-specific data structure.

Since these STREAMS commands are a subset of ioctl, they are subject to the
errors described there. In addition to those errors, the call will fail with
errno set to EINVAL, without processing a control function, if the stream
referenced by fildes is linked below a multiplexer, or if command is not a
valid value for a stream.

Also, as described in ioctl, STREAMS modules and drivers can detect errors.
In this case, the module or driver sends an error message to the stream head
containing an error value. This causes subsequent system calls to fail with
erro set to this value.

COMMAND FUNCTIONS
The following ioctl commands, with error values indicated, are applicable to
all STREAMS files:

L_PUSH
Pushes the module whose name is pointed to by arg onto the
top of the current stream, just below the stream head. It then
calls the open routine of the newly-pushed module. On
failure, errno is set to one of the following values:

[EINVAL] Invalid module name.
[EFAULT] arg points outside the allocated address space.
[ENXIO] Hangup received on fildes.

L_POP
Removes the module just below the stream head of the stream
pointed to by fildes. arg should be 0 in an L_POP request. On
failure, errno is set to one of the following values:

[EINVAL] No module present in the stream.
[ENXIO] Hangup received on fildes.