Changes in the 2.11BSD kernel

November 1, 1990

Steven M. Schultz

Contel Federal Systems
Westlake Village, California 91359-5027
wlbr!wlv!sms; sms@wlv.imsd.contel.com

This document summarizes changes in PDP-11↑ UNIX‡ between this release and the January 1989 2.10.1BSD distribution.

It is intended to provide sufficient information that those who maintain the kernel, have local modifications to install, or who have versions of 2.10.1BSD modified to run on other hardware should be able to determine how to integrate this version of the system into their environment. As always, the source code is the final source of information, and this document is intended primarily to point out those areas that have changed.

Most of the changes between 2.10.1BSD and 2.11BSD fall into one of several categories. These are:

- bug fixes,
- performance improvements,
- addition of 4.3BSD system calls and application programs,
- removal of features no longer supported in the 4.3BSD release,
- new protocol and hardware support.

The major changes to the kernel are:

- the implementation of the 4.3BSD on disk directory structure,
- a rewrite of the kernel I/O interface using the 4.3BSD UIO/IOVEC mechanism,
- the addition of system calls,
- replacement of even more of the high kernel with portions of the 4.3BSD kernel,
- the addition of the 4.3BSD namei cache,
- additional restructuring of the kernel into the 4.3BSD structure.

The major purpose of this document is to summarize the changes between this and the January 1989 release of 2.10BSD. The latter was designated as 2.10.1BSD. Although the changes are fairly simple to describe, they cover large portions of the distribution. Most will not be visible to either users or administrators; in some cases no recompilation is necessary. However, since the on disk directory structure and password file format have changed, programs which deal with directories or the password file will need to be recompiled. Administrators should be aware that the 4.3BSD disk quota system is available and works quite well.

The major change, and the reason for the release, is an **extensive** reworking of the kernel to implement the *readv* and *writev* system calls and the 4.3BSD on disk directory structure. Filenames are no longer limited to 14 characters in length. At present, MAXNAMELEN is set at 63 (one fourth of the maximum path name length specified by MAXPATHLEN).

In application land, many of the 4BSD changes/enchancements released over the Internet have been ported to 2.11BSD, most notably the shadow password file (with password aging), FTP with the *restart*

↑ DEC, PDP-11, QBUS, and UNIBUS are trademarks of Digital Equipment Corporation.
‡ UNIX is a trademark of Bell Laboratories.

capability, rshd and rlogind with security fixes. **Many** other programs have had fixes applied, in particular, the line printer spooling sub-system.

This document is not intended to be an introduction to the kernel, but assumes familiarity with prior versions of the kernel, particularly the 2.10.1BSD release. Other documents may be consulted for more complete discussions of the kernel and its subsystems. It cannot be too strongly emphasized that 2.11BSD resembles 4.3BSD even more than 2.10.1BSD did. The tradition of remaining bug-for-bug compatible with 4.3BSD has been continued. It is **STRONGLY** recommended that 4.3BSD manuals be consulted when using this system. 2.9BSD manuals are not even close to being correct. Online documentation is as complete and correct as time permitted.

This release is not supported, and it is definitely **NOT** an official Berkeley release. It was called 2.11BSD because of the number of changes (including the file system) and because a coldstart from tape is required. The ‘‘bugs’’ address supplied with this release (as well as with the 4BSD releases) will work for some unknown period of time; **make sure** that the ‘‘Index:’’ line of the bug report indicates that the release is ‘‘2.11BSD’’. See the *sendbug*(8) program for more details. All fixes that I make, or that are sent to me, will be posted on *USENET*, in the news group ‘‘comp.bugs.2bsd’’.

The author would like to thank the following people for their contributions to the development of this release. His employer for providing the environment and equipment used in the creation of this release. To Paul Taylor (taylor@oswego.oswego.edu) for doing the 11/73 testing in spite of Q22 being undefined. To Terry Kennedy (terry@spcvxa.spc.edu) for being eclectic enough to beta-test a 2.11BSD system alongside the VMS and RSTS systems.

Finally, much credit should go to the authors of 4.2BSD and 4.3BSD from which even more was stolen than had previously been the case for 2.10.1BSD.

1. Organizational changes

The directory organization and file names are almost identical to that of 2.10.1BSD. This document summarizes the changes between 2.10.1BSD and 2.11BSD. For a description of the differences between 2.9BSD and 2.10.1BSD refer to the documentation in /usr/doc/2.10/kchanges.2.10.

The following files in /sys/h have changed in 2.11BSD.

uio.h The 4.3BSD *uio* and *iovec* structures were ported. scatter/gather i/o is implemented in 2.11BSD. The compatibility routines have been replaced with system calls.

buf.h The device number was included in the buffer hashing function. The **SMALL** conditional was removed.

file.h The *fileops* structure from 4.3BSD was added.

dir.h The 4.3BSD version was ported, the old V7 directory structure is gone.

mbuf.h Change made to the mbuf allocation macro to call the drain routines if out of mbufs. This is useful when a large number of connections are in TIME_WAIT states.

namei.h The 4.3BSD *namei* argument encapsulation and inode cache structures were ported.

proc.h The **SMALL** conditional was removed because it is now standard.

user.h Many changes made to the *user* structure. The members *u_offset*, *u_count*, *u_segflg*, and *u_base* do not exist now. The *u_nd* and *u_ncache* are either new or changed.

msgbuf.h The 4.3BSD kernel error logger (*/dev/klog*) is now present in 2.11BSD, the message buffer is now external to the kernel.

inode.h Both the on disk and incore inode structures have changed. The incore inode has had *i_id* added as part of the *namei* cache support. The on disk structure defines block numbers as **daddr_t** now, the old 3 byte packing scheme is obsolete at last.

1.1. /sys/sys

The following files were changed for 2.11BSD.

tty.c Changes to use **uio/iovec** instead of fixed offsets in user structure.

vm_text.c Use new *rdwri* kernel I/O routine.

init_sysent.c Added *readv* and *writev*.

init_main.c Use *rdwri* and *namei* argument encapsulation.

ufs_namei.c Totally rewritten from the 4.3BSD source. Implements the argument encapsulation and maintains the name translation cache.

kern_exec.c Use *rdwri* instead of *readi*.

sys_pipe.c Use *rdwri*. Defines a *fileops pipeops* table for use by sys_generic.c in dispatching i/o requests.

kern_acct.c *namei* encapsulation when looking up accounting file name.

sys_inode.c Defines *fileops inodeops*, use *rdwri/rwip*, **QUOTA** check ignores **PIPES** now. Remove a couple extraneous *saveseg5* calls.

kern_clock.c Autonice long running processes like 4.3BSD does. Programs in an endless loop impact the system less if this is done.

kern_descrip.c Uses the externally *fileops* tables to dispatch file requests. *closef* had the extra argument *nouser* removed because it was no longer used and the 4.3BSD sources did not refer to it any longer.

ufs_fio.c Use the *namei* argument encapsulation.

ufs_inode.c Extra *saveseg5* call removed. *igrab* ported from 4.3BSD. **SMALL** conditional definition of HASHSIZE removed, the smaller value made the default.

sys_generic.c *readv* and *writev* implemented. *fileops* and *uio/iovec* changes.

kern_sig.c *namei* argument encapsulation changes in *core*. The core file written using *rdwri* instead of *writei*.

kern_subr.c *uiomove* rewritten to use *uio/iovec* mechanism.

kern_synch.c **SMALL** conditional definition of **SQSIZE** removed, the smaller value made the default.

subr_prf.c *log* defined to provide the kernel interface to **syslogd**. The message buffer is now 4kb and resides external to the kernel.

ufs_alloc.c Extra include files needed due to changes in some of the */sys/h* files.

tty_pty.c Changes to use *uio/iovec* mechanism.

vm_swp.c *physio* and *physstrat* ported from 4.3BSD and now use the *uio/iovec* mechanism.

tty_tb.c *uio/iovec* changes.

ufs_mount.c *namei* argument encapsulation changes. Prevent *mount* on a directory which is already itself a *mount* point.

ufs_bmap.c Additional include files needed due to changes in */sys/h*.

tty_tty.c *uio/iovec* changes.

ufs_syscalls.c *rename* ported fresh from 4.3BSD, *namei* argument encapsulation changes, directory syscalls changed to handle the new directory structures. Many changes in this file.

uipc_socket.c *uio/iovec* changes.

uipc_mbuf.c Changes to call the drain routines on a mbuf shortage. This is useful if many sockets are in the TIME_WAIT state at once due to something like a *ftp mget/mput* transferring many files in a short period of time. The storage for the starting addresses (both physical click and UNIBUS) of the DMA I/O region are declared here, see *sys_net.c* for their use.

uipc_usrreq.c *uio/iovec* changes.

quota_kern.c *rdwri* used instead of *readi*.

sys_net.c The ACC LH/DH-11 (if_acc.c) and Proteon proNET (if_vv.c) network interfaces added. *uiomove* rewritten to use *uio/iovec* mechanism. *putchar* redefined as *_pchar* and a macro written to call the kernel *putchar* routine. This causes networking error messages to be passed thru the kernel logger to *syslogd*. A missing third argument added in the *SKcall* to *putchar*. To greatly reduce the number of UMRs consumed by network interface drivers the DMA I/O region is mapped once using the minimal number of UMRs required. The starting click address and UNIBUS virtual address are saved for use in *pdpif/if_uba.c*.

sys_kern.c *namei* argument encapsulation changes.

subr_log.c The kernel logger ported from 4.3BSD.

uipc_syscalls.c *uio/iovec* changes, *sendmsg*, etc ported fresh from 4.3BSD to handle scatter/gather i/o correctly.

1.2. /sys/conf

The following files were changed for 2.11BSD.

GENERIC **SMALL** was moved to the always defined category to save kernel D space.

Make The Make.* files were modified to add the kernel logger and to reflect the changes in some of the kernel file names. With file names greater than 14 characters supported, kern_resrce.c becomes kern_resource.c, etc.

checksys.c There were several bugs fixed in the program's calculation of how much memory the system would occupy.

1.3. /sys/pdpuba

Almost **ALL** of the files in this directory were modified. The modifications were small, usually nothing more than passing an extra *uio* argument in the *xxread* and *xxwrite* functions on thru to *physio*.

tmscp.c This driver is new to 2.11BSD. At present it has only been tested on an 11/73 with a TK50.

In addition the *sys/OTHERS* directory has had several ''new'' device drivers added to it that may or may not work. A cursory pass was made thru this directory to add the *uio/iovec* changes - no guarantee is made that all necessary changes were made, or that the changes made will work. It is extremely probable that they do not handle the new ioctl protocols and it is also likely that they do not correctly map buffers in and out of kernel space correctly. For more information regarding the installation of device drivers, see the file *sys/OTHERS/README*. This is a rambling, disjointed ''must'' for the driver hacker. You should also see this directory if you are having problems with a driver that's currently in place in 2.10BSD. There are several different versions of drivers, bug fixes etc. that we just didn't have time to install and/or test out. A great deal of work has been done on the *ra* and *xp* drivers. They are known to work, and work reliably. You should use them, if at all possible.

1.4. /sys/pdpmba

1.5. /sys/pdp

The following files were changed in 2.11BSD:

conf.c Entries added in the *cdevsw* and *bdevsw* tables for */dev/klog* and *tmscp* tapes.

seg.h Changed the **normalseg5** macro to not depend on **QUOTA**. This is safe since quota manipulation only occurs in the high kernel.

scb.s Define vectors for ACC LH/DH-11 and Proteon proNET network interfaces.

machparam.h Definitions from files in /usr/include/OLD were moved into this file. Almost all of /usr/include/OLD has been removed.

trap.c Changes made to save more information if the network crashes and to prevent further corruption from happening.

machdep2.c Allocate memory for the external kernel error message buffer and the *namei* cache.

mch_click.s Extra **mov** instructions removed (to save I space) and the loop count doubled. Never did know why memory was copied in 4 loops of 8 **mov** instructions instead of 8 loops of 4 **mov** instructions.

mem.c Ported fresh from 4.3BSD. Uses the *uio/iovec* mechanism now.

cons.c Changes to use the *uio/iovec* mechanism.

mch_dump.s Save area for extra information on network crash allocated.

mch_xxx.s Same changes as made to mch_click.s and the *delay* routine is no longer conditional on a networking system being defined.

mch_var.s The flag *ubmap* was made an *int* instead of a *char* to force even alignment. Even alignment is required for use by the *mfkd* function from the networking code. The networking no longer has its own private/wired copy of *ubmap*, instead the kernel version is examined in exactly one place:*pdpif/if_uba.c*.

net_mac.h *putchar* macro defined for the supervisor mode networking to use when calling the kernel *putchar* routine. The *NETUBAA* macro was modified to be a 0 if not on a UNIBUS system, this allows code to be written which checks *ubmap* and references NETUBAA for the UNIBUS case but returns earlier for the Qbus case.

net_trap.s Network device interrupts now included in system interrupt counts.

tmscp.h This file is new and contains the definitions for the TMSCP driver.

1.6. /sys/mdec

ALL of the bootstraps have been modified to read the new on disk directory structure. The changes to read the more complicated directory format necessitated the removal of prompting from all bootstraps. If **/boot** can not be found you are in deep trouble.

1.7. /sys/netinet

The following were changed in 2.11BSD:

raw_ip.c Changes to support the *traceroute* utility.

tcp_subr.c Changes to support the *traceroute* utility.

1.8. /sys/pdpstand

The following were changed in 2.11BSD:

maketape.c Changes made to use *mtio*(4) functions to write tapemarks instead of doing a open/close/open sequence on the non-rewind tape. This change was necessitated by TU81s at 1600BPI.

sys.c New directory reading routine written.

ra.c Error checking corrected.

tmscp.c This is a newly ported standalone TMSCP driver (TK50/TU81).

1.9. /sys/netimp

The following were changed in 2.11BSD:

if_imphost.h

if_imp.c

if_imphost.c Porting to the supervisor mode networking, some changes to fix compiler errors, other changes to fix bugs.

1.10. /sys/pdpif

The following were changed in 2.11BSD:

if_vv.h

if_vv.c Under 2.10.1BSD these files were accidentally placed in the supported directory when they would not even successfully compile. For 2.11BSD these were ported from 4.3BSD and the changes necessary to operate in the supervisor mode networking were made.

if_acc.c Changes to run in supervisor mode were made as well as several bugs (missing arguments, etc) being fixed.

if_uba.c The routine *ubmalloc*() was rewritten to compute addresses using the pre-allocated UMRs which map the DMA I/O region. *ubmalloc*() does not allocate UMRs now. Also, the calling convention of *uballoc* and *ubmalloc* has changed.

if_de.c

if_de.h The calls to *uballoc* and *ubmalloc* had to be changed. The *dereset* routine was removed for two reasons: 1) The concept of a UNIBUS being reset without a system reboot is meaningless on a PDP-11, 2) *dereset* wouldn't work even if it was called due to resource exhaustion.

if_il.c The second call to *uballoc*() in *ilinit*() was removed. The UNIBUS resources are allocated at attach time, the second call was allocating (and wasting) UMRs which were not needed.

if_dmc.c The reset routine was removed for the same reasons it was removed in *if_de.c*.

if_qe.c The previous version of the driver was flaky and would hang at unpredictable time. The current version is marginally slower, but is a fresh port of the 4.3BSD driver which no longer hangs or grabs and holds 20 mbufs. This driver will statically allocate a fixed number of buffers from main memory after first using the the memory dedicated to the network DMA arena.

1.11. /sys/net

The skeletal support for PUP has been removed.

2. Changes inside the kernel

1) The user structure *u* has undergone several changes. The members *u_offset*, *u_count*, *u_segflg* and *u_base* have been removed. Drivers and programs which refer to them will no longer compile. The kernel now allocates a *struct uio* dynamically on the stack and refers to the *uio_offset*, *uio_iovec*, *uio_segflg* members, exactly as 4.3BSD does.

   The *u_ncache* structure was modified to remove the timestamp member to save space - there were no references anywhere to it.

   The changes to the *u* structure make old core images unusable by the debugger *adb*.

2) The *namei* function has a new calling convention with its arguments, associated context, and side effects encapsulated in a single structure. *namei* has been extensively modified to implement the name cache and to cache directory offsets for each process. It may now return ENAMETOOLONG when appropriate, and returns EINVAL if the 8th bit is set on one of the path name characters.

   The automatic and silent truncation of file names to 14 characters is not performed - a name longer than MAXNAMELEN (currently 63) will produce the error ENAMETOOLONG.

   A table of recent name-to-inode translations is maintained by *namei*, and used as a look-aside cache when translating each component of each file path name. Each *namecache* entry contains the parent directory's device and inode, the length of the name, and the name itself, and is hashed on the name. It also contains a pointer to the inode for the file whose name it contains. Unlike most inode pointers, which hold a ''hard'' reference by incrementing the reference count, the name cache holds a ''soft'' reference, a pointer to an inode that may be reused. In order to validate the inode from a name cache

↑ UNIX is a Trademark of Bell Laboratories.

reference, each inode is assigned a unique ''capability'' when it is brought into memory. When the inode entry is reused for another file, or when the name of the file is changed, this capability is changed. This allows the inode cache to be handled normally, releasing inodes at the head of the LRU list without regard for name cache references, and allows multiple names for the same inode to be in the cache simultaneously without complicating the invalidation procedure. An additional feature of this scheme is that when opening a file, it is possible to determine whether the file was previously open. This is useful when beginning execution of a file, to check whether the file might be open for writing, and for similar situations.

3) Compatibility with previous versions of PDP-11 UNIX↑ is limited to partial binary compatibility with 2.10.1BSD. 2.10.1BSD programs which do not read directories or the password file should run without change. It is **strongly** recommended that applications be recompiled to obtain the benefit of the changes to the system libraries.

   Because the file system has changed with 2.11BSD old file systems can not be mounted. There is a version of *dump* available in /usr/src/old/dump which understands the old directory structure of 2.10.1BSD filesystems.

   Old *dump* tapes from 2.10BSD or 2.10.1BSD are automatically converted by 2.11BSD's *restor* on input.

4) As mentioned earlier, the *uio/iovec* method of kernel I/O has been implemented. If you have local drivers, they will require some work. There are many working examples in */sys/pdpuba*. The routines *readi* and *writei* do not exist any longer, having been replaced with the general 4.3BSD *rdwri* routine.

5) The ''real-time'' features of 2.9BSD should probably go away, but for now they have been left in place, and, should work as they always have, with one major exception. The *fmove*() routine has not been fixed to be interruptible. See the *copy*() routine for examples of what needs to be done to make it behave correctly. This, however, will be fairly difficult. I suggest that if you want to use **CGL_RTP** that you comment out the use of *uiofmove*() in *uiomove*().

6) The 4.3BSD kernel logger */dev/klog* has been implemented. Kernel messages are placed in the message buffer and are read from there through the log device */dev/klog*. The *log* routine is similar to *printf* but does not print on the console, thereby suspending system operation. *Log* takes a priority as well as a format, both of which are read from the log device by the system error logger *syslogd*. *Uprintf* was modified to check its terminal output queue and to block rather than to use all of the system clists; it is now even less appropriate for use from interrupt level. *Tprintf* is similar to *uprintf* but prints to the tty specified as an argument rather than to that of the current user. *Tprintf* does not block if the output queue is overfull, but logs only to the error log; it may thus be used from interrupt level. Because of these changes, *putchar* and *printn* require an additional argument specifying the destination(s) of the character. The *tablefull* error routine was changed to use *log* rather than *printf*. Some of the other drivers *dh* and *br* also have been modified to use *log*.

   The message buffer is now 4kb in size and is external to the kernel. The message buffer is mapped when data is written or read from the buffer. This obsoletes *dmesg*(8) which has been modified to find the external message buffer on the off chance anyone still wants to run it.

7) Most of the conditional compilation defines in the 2.9BSD kernel have been removed because the features they controlled are now either standard The following table lists *#defines* that are now a standard part of 2.11BSD.

define name	feature	comment
SMALL	use smaller hash table and queues to save D space	

8) Directory truncation now is performed the same way as 4.3BSD does it, and directories are always a multiple of 512 bytes. The old method of truncating directories with 10 or more trailing empty slots has disappeared. There is a new version of *fsck* which can automatically create and extend (up to the number of direct blocks allowed, currently 4) the *lost+found* directory.

9) Again, it must be mentioned that this document summarizes the changes from 2.10.1BSD to 2.11BSD. If you are upgrading from a earlier version than 2.10.1BSD or 2.10BSD you will want to format and read the documentation in /usr/doc/2.10/kchanges.2.10 and /usr/doc/2.10/setup.2.10.

3. Changes in application programs and libraries.

The application directories have been rearranged slightly in an effort to follow the 4.3Tahoe based updates released over the Internet. Many programs which used to be single .c source files are now subdirectories with their own make files (*login*(1) for example). As in the kernel, the goal was to remain as true to the 4.3BSD sources as possible. In some of the source directories there are either or both of the special directories ''PORT'' and ''OLD''. PORT contains copies of 4.3BSD source which ought to be ported to 2.11BSD, but due to time constraints had to be left undone. The number of PORT directories included with 2.11BSD is smaller than with 2.10.1BSD for two reasons: 1) to keep the distribution from being more than two reels of tape and 2) the amount of new/ported software has increased greatly. Copies of the missing PORT directories are available from either a cooperating 4.3BSD system or the network archives around the world. OLD contains copies of 2.9BSD source that, while we have ported the 4.3BSD version of the source code, we're unsure enough of it that we wanted to provide a backup copy. The only OLD directory of significance included in 2.11BSD is /usr/src/new/OLD which contains miscellaneous bits of trivia such as *lisp11* and so on.

The portable/ASCII *ar*(1) file format from 4.3BSD has been implemented, this is described in the paragraphs below.

The following paragraphs are a description of several things that have changed outside of the kernel.

1) Since the C compiler still only recognizes seven significant characters in external names, several standard library names had to be changed to prevent name collisions. However, to prevent portability problems in your programs you should use the standard names. All known collisions in the standard include files or the C library have been handled either in the include file itself or in the include file *include/short_names.h*. This works because we're using the 4.3BSD C preprocessor, which has flexnames. Networking programs almost always need to include *short_names.h*. See *src/bin/mail.c* and *src/bin/login/login.c* for examples of long name work arounds. The C library itself with only a couple exceptions is a port of the 4.3BSD C library.

2) Files ported from 4BSD systems that have more than MAXNAMLEN characters will no longer port correctly. Since MAXNAMLEN has been raised to 63 this should not be a problem.

3) The directory reading routines are a fresh port from 4.3BSD and are part of stdio. The old V7 directory structure does **NOT** exist any longer in *dir.h*. There are only two programs in 2.11BSD which know what the old directory structure looked like: src/old/*dump* and *restor*(8). The define ''DIRSIZ'' has been ported (and fixed, there was a rounding error) from 4.3BSD. MAXNAMLEN is now 63. It would be possible (and certainly easier) to raise this now than it was to implement the new filesystem initially, but once raised the limit can not be lowered.

4) The 4.3BSD manual pages for *sigblock*(2), *sigpause*(2), and *sigsetmask*(2) are deceptive. They indicate that signal masks are integers, but, they must be able to hold 32 bits. Typically you'll see code like:

```
int omask;
omask = sigsetmask(0);
...
sigsetmask(omask);
```

which should be translated to:

```
long omask;
omask = sigsetmask(0L);
...
```

sigsetmask(omask);

In general, the fact that 4.3BSD thinks an ‘‘int’’ is 32 bits is the worst porting problem that you’ll run into; finding ‘‘ints’’ that should be ‘‘longs’’ is an arcane art. Routines that I look for as a matter of habit are any one of the *seek*(2) routines, *ftell*(3), *time*(2), the various *signal*(2) routines, *select*(2) and the *truncate*(2) routines.

The functions *sigblock* and *sigsetmask* are defined as returning a long result in *h/signal.h*, which should ease some of the porting problems. The lint libraries have also been updated. In general, though, you’ll have to scan any source you plan on porting for calls to *sigblock*, *sigpause*, or *sigsetmask* that take an int as a parameter or store their return value in an int.

To give an indication of the subtlety the long/int problem can take on, consider the following code fragment taken from /sys/sys/tty.c:

newflags = (tp->t_flags&0xffff0000) | (sg->sg_flags&0xffff));

where *newflags* and the fields *t_flags* and *sg_flags* are all longs. The problem occurs with "sg->sg_flags&0xffff". The intent is fairly obvious, but in **long op int** the **int** ("0xffff") is sign-extended to **long** ("0xffffffff") before the operation as per K&R. The resulting operation in this case is a no-op! The fix is fairly simple in this instance and yields the following:

newflags = (tp->t_flags&0xffff0000) | (sg->sg_flags&0xffffL));

5) The PDP-11 *setjmp*(2) implementation contains a subtle bug that occurs when a routine containing a *setjmp* has *register* variables. The bug sometimes causes those variables to be given invalid values when a longjmp is made back to the routine. This is probably impossible to fix in a reasonable manner, and it’s much simpler to simply avoid register variables in such routines.

6) The optional ’#’ character is still not supported by *printf*(3).

7) The DEC MXV11 bootstrap ROM, for the RL’s, TU’s, RX’s, TK’s and RD’s among others, *requires* that deadstart blocks begin with an 0240 and a branch. This has already been implemented in *rauboot.s* and *rluboot.s* in *sys/mdec* as well as *sys/pdpstand/tmscpboot.s*.

8) To port the 4.3BSD *make*(1) program, several of its table sizes had to be reduced. Make is very unforgiving of Makefiles that are too large. If make drops core for no reason that you can think of, try reducing the size of the Makefile. Also, don’t run make depend in the system application directories, make can’t handle it.

9) Don’t set *csh* history too high; it eventually runs out of space and logs you out. *csh* now is overlaid. The additional code added by the shadow password file routines pushed *csh* over the edge - no surprise there, *csh* was within a couple hundred bytes to begin with. Since *csh* had to be overlaid, *limits* were enabled.

10) The games directory under 2.10.1BSD was largely untested and nothing has been done to change this.

11) The C compiler actually handles bit fields (but generates atrocious code), identically named global structure elements, and lots of other stuff. The generated code is not terrible overall but not exactly great either, and the optimizer does very little to correct the situation. It *doesn’t* handle any of the old assignment operators, and, not only doesn’t handle them, but **produces bogus code**. It is **STRONGLY** recommended that you read the file *src/lib/ccom/TEST/README*. It goes into the problems with this compiler in more depth and contains some other Extremely Important Information.

12) *readv*(2) and *writev*(2) under 2.11BSD are implemented as actual system calls rather than compatibility routines. At present very little in the system aside from *perror*(3) and *syslog*(3) use scatter read or write.

13) There is a define, ‘‘SEPFLAG’’, in many Makefiles, that governs compilation for separate and non-separate I/D machines. If you have a non-separate I/D machine, set it to ‘‘-n’’. If you have a separate I/D machine, set it to ‘‘-i’’. This should really go away since the chances of 2.11BSD ever running on a non-split machine are not distinguishable from zero at this time. Too many capabilities have been added, programs such as *csh* and sendmail have to be overlaid even on split I/D machines!

14) The directory */usr/src/new* is a compendium of programs moved in from the 4.3BSD directory *usr/src/new*, assorted programs ported to 2.11BSD from various places around the Internet, and remnants from the 2.9BSD *usr/src/local* and *usr/src/contrib* directories. Most of the programs in */usr/src/new* have been in production use for many months, while others are less well tested.

15) New versions of *ftp* and *ftpd* are present. The stream restart capability is present allowing an aborted transfer to be restarted if the remote server also provides the *restart* command. Also a number of bugs were fixed, some were simple long vs. int problems, in other cases both *ftp* and *ftpd* insisted on freeing memory which had never been allocated (in a couple cases part of the stack would be free’d!).

16) *mkpasswd*, *chpass*, *passwd*, *vipw*, *lock* were ported from the 4.3BSD shadow password implementation. The password file format has changed, and unlike the Vax version, the new password files are not binary compatible with old programs. Any program which uses the password routines will have to be relinked with the new libraries.

    The maximum length of a login name has been increased from 8 to 15 under 2.11BSD. This was dictated by having to share development facilities with production systems where the length had been increased. If this change is not acceptable, then the *utmp.h* file will have to be edited and the system libraries and applications recompiled.

17) *rlogind*, *rshd*, *rcp*, *fingerd*, *syslogd*, *finger* are all ported from Internet distributed 4BSD bugfix releases. These programs include many security enhancements, and in the process of porting them to 2.11BSD the long vs. int bugs were fixed as well.

18) *sendmail* had a fatal memory leak in alias processing. A string extraction method is used (thank you Cyrus) to reduce sendmail’s D space requirements by about 5kb - there is now the file */usr/share/misc/sendmail.sr* used to hold much of sendmail’s string data.

    ctimed is a program which moves the time zone/daylight savings time tables to a separate process, this saves approximately 2kb of D space in sendmail. The *ctime*(3) entry points are replaced in *sendmail/src/ctime.c* with calls to *ctimed*. At present only *sendmail* uses *ctimed*, but the method may be used by any process which needs to save about 2kb of D space.

19) *vmstat* was modified to print out the namei cache statistics. A typo was corrected which caused *vmstat* to omit print the DZ pseudo dma count.

20) The assembler *as* underwent **MAJOR** changes to permit it to run split I/D (the text being sharable speeds loading). The buffer sizes *as* uses to read and write files was doubled from 512 to 1024 bytes.

21) *csh* was overlaid because the password file library routines grew in size. The overlay scheme is optimal, the only time an overlay switch is incurred is when doing a ~ expansion. *limits* are now enabled.

22) *fsck*, *icheck*, *dcheck*, *ncheck*, *mkfs*, *mkproto* were all ported from 4.3BSD to handle the new on disk directory structure. The 30000 inode/file limit in *dcheck* has been removed. *mkfs* no longer has the ability to populate a filesystem with files, this capability has been moved to *mkproto*. Note: *mkproto* can only handle files up to single indirect (about 256kb) in length.

    fsck can now recover from a disconnected root inode! also, *fsck* can create and dynamically expand *lost+found* up to the number of direct blocks allowed to an inode. This limit is comparable to the number of empty slots created by the obsolete *mklost*(8). *mklost*(8) has been renamed *mklost+found*(8) and uses 63 character file names to create the empty slots.

23) *make* has been fixed to correctly handle library archive members The man page was correct, *make* just had a bug.

24) *ld* had a bug in computing the size of the symbol table. This resulted in debuggers not being able to find symbols. Other bugs in the loadmap and trace options were fixed. Modifications were also made to accommodate the new *ar* archive file format.

25) *ranlib* has the -t option from 4.3BSD now. This allows the internal time on an object archive to be updated (typically used after a copy which changed the timestamp on the file but not that within the archive). *ranlib* was modified to handle the new *ar* archive file format.

26) *login* is a new version ported from the 4.3BSD shadow password file release. This version of *login* performs the password and account aging checks.

27) *chroot* is a utility ported from 4.3Tahoe, used to execute a command after issuing a *chroot*(2) system call.

28) Both *write* and *wall* had long vs. int bugs.

29) *dump* and *restor* were modified to handle the new directory structure. *restor* can read old 2.10.1BSD *dump* tapes.

30) *mkhosts* now correctly handles a host having multiple addresses. Although host table usage is discouraged in this era of domain name service, if you want to use host tables they should at least work.

31) The **hayes** dialer in the *tip* program was totally broken. The correct solution would have been to rewrite it to behave like the *uucp* **hayes** dialer, but time did not permit this. The fixed version will work, but it's not pretty.

32) The spooling system (*lpd*) was rife with long vs. int bugs, in particular the free space check could cause the connection to be dropped. Another problem, that of removing print jobs has been fixed by implementing long file names - the concatenation of the hostname to a queue id previously would result in mismatches of file names when trying to remove remote requests. A small change was made to permit printing of the new ASCII *ar* archive files.

33) The pascal subsystem had a flaw in the *clock*() function, the floating point conversions of the timeofday were wrong.

34) *f77* had a bug in the **equivalence** handling caused by a long vs. int problem.

35) *rn*, *zmodem*, *notes*, *kermit* have been ported. The overlay schemes for *rn* and *notes* are definitely not optimal, but the programs will run. There were numerous long vs. int problems corrected, hopefully all were found, but there might be some lurking about.

    kermit is a fairly modern version (4F(89)), the DEBUG option had to be left disabled to fit the D space available. The comment about one of the modules crashing optimizers is correct - it has to be compiled manually so that the remaining modules can use the -O flag.

36) A version of the Network Time Protocol has been ported. Typical usage is to use *ntp* to synchronize the PDP-11 with a master system. The daemon *ntpd* does run and will use *adjtime*(2) to adjust the system clock.

37) The *traceroute* program has been ported, the kernel modifications are included in 2.11BSD as well.

38) *getpwent*(3) could fail to recognize when a rewind of the password file was necessary.

39) *qsort*(3) could fail when sorting large arrays, a missing 'unsigned' was the cause.

40) *signal(3)* misdeclared the saved signal masks as **int** rather than **long**.

41) *strdup*(3), *strsep*(3), *readv*(2), *writev*(2), *strtok*(3) are new to 2.11BSD.

42) *puts*(3) is implemented in C rather than assembly. The assembly version is still present on the system, but has a fatal flaw in programs which run out of D space. The port of **puts.s** from the Vax assembly is correct, but the PDP-11 can fail to *malloc*(3) a buffer where the Vax would succeed, the failure to allocate the buffer causes file corruption when *puts* is called.

43) *perror*, *getusershell*, *rcmd* are new versions ported from 4.3BSD Internet releases.

44) *gethostent*, *gethnamadr* have been rewritten to support multiple addresses per host.

45) *syslog* had a long vs. int bug corrected in a *signal* mask.

46) *readdir* was ported from 4.3BSD to handle the new directory structure.

47) *readv* and *writev* are system calls rather than compatibility routines.

48) *Mail* incorrectly handled the editing of multiple messages, only the first one was processed.

49) *gcore*,*pstat* and *fstat* were modified to deal with the new *u* structure.

50) *rdist* ported from a 4.3BSD Internet release and the *long* vs. *int* bugs corrected.

51) *lastcomm* could not handle large (>2048) uids. also, a large amount of memory was wasted with static tables for the uid to name mapping. The uid to name mapping logic from *ls* was adopted.

52) *mkstr* had a error in an ascii-octal to binary conversion. This could result in erroneous *string* file being constructed.

53) *tcopy* incorrectly checked the number of arguments passed.

54) *getty* failed to handle adjacent colons in the *gettytab* file which would cause skipping of a field.

55) *sa* incorrectly handled uids which are signed. The fix was to use *uid_t* instead of int.

56) *rwhod* created files unreadable by *rwho*. Changed *rwhod* to create publicly readable files.

57) *testnet* is a new program used by */etc/rc* to test for a networking kernel.

58) *telnet* has had several bugs in the open and close handling fixed.

59) *ar* was ported from 4.3BSD. Archive files now use a portable ASCII format rather than embedding binary information in the archive headers. If printable files ar *ar*'d together, the resulting archive is printable.

60) *file* was ported from 4.3BSD.

61) *nm* was heavily modified to deal with the new *ar* file format. At the same time changes were made which permit *nm* to list the symbol table of */unix* without running out of memory.

62) The *man* pages for *ranlib*, *ar*, *file* have been updated or copied from 4.3BSD.

63) *arcv* was ported from 4.3BSD, it is used to convert (in place) old format *ar* archives to the new portable format. ALL system libraries have been converted, *arcv* will only be needed to convert local archives. See */usr/src/old/arcv* for the source and man page. The executable is already installed in */usr/old*.