

PLEXUS SOFTWARE SUPPORT NEWSLETTER

Summer 1987

PLEXUS SOFTWARE SUPPORT NEWSLETTER

Summer 1987

Editor:

Harry K. Wong

Associate Editor:

Lynn Macey

Editorial Review Board:

Ron Green

Ric Urrutia

Contributing Authors:

Jeff Huntington

John Maddox

Mickey Mestel

Stephen Palmer

Theresa Ruebenson

John Saxon

Sheryl Truby

Diana Yee

CDB is a trademark of Third Eye Software, Inc.

DEC, VT220 and VT100 are trademarks of Digital Equipment Corporation.

IBM and AT are trademarks of International Business Machines Corporation.

INFORMIX is a trademark of Informix Software, Inc.

MC68000, MC68010, MC68020 and MC68881 are trademarks of Motorola, Inc.

Multiplex is a trademark of Network Innovations Corporation.

Plexus, XDP, XDP System and XDE are trademarks of Plexus Computers.

RM/COBOL is a trademark of Ryan-McFarland Corporation.

UNIX is a trademark of AT&T Bell Laboratories.

PLEXUS SOFTWARE SUPPORT NEWSLETTER Summer 1987

DEPARTMENTS

New Product Announcements	1
Customer Satisfaction	5
Support Services Available	7
Plexus Customer Education News	10
Technical Publications Update	11
Feature Terminfo: The Tektronix 4105/4107	12
Public Domain Software Spotlight	15
Technical Tips for Your Plexus System and Software	17

FEATURE

Using FSCK—A Guide to the UNIX File System Check Program	20
--	----

APPENDIX

Software Limitations in UNIX S/M/R 1.5	36
Correction to Sys5 S/M/R 1.5 Release Notice	37
Third Party Software Limitations	38
Software Patches and Updates	43
Current Software Releases	45
Current Documentation Releases	52

A Note from the Editor

Welcome to the Summer 1987 edition of Plexus' Software Support Newsletter! Continuing on our theme from the Spring Newsletter, we are addressing yet another subject that has remained a mystery to many system administrators — the FSCK file system check program. Author Michael Saxon has written a simple-to-understand explanation of the fsck program. He outlines in a step-by-step fashion what happens while fsck is being run. In addition, be sure to read the column on Customer Satisfaction as it outlines changes we have made to improve customer support.

One section not to miss is the appendix, which lists the latest software and documentation available for your Plexus computer. Please check in the appendix to see if you are fully up to date on the latest levels of Plexus software and documentation. For those customers who are running on UNIX release S/M/R 1.5, please be sure to check the section on Software Patches and Updates.

This publication is designed to help you take full advantage of the capabilities of your Plexus computer. Each issue provides information on the latest software available on Plexus computers as well as handy information on bug fixes and current updates on our latest software.

We hope that you will find this Plexus Software Support Newsletter to be helpful. The Plexus Software Support Newsletter is a supplement to Plexus Documentation and assumes that the reader has a working knowledge of system administration on UNIX machines. If you have any comments, helpful hints or articles that may be useful to Plexus users, please let us know by sending your comments via email to harry@FE while on the Plexus Software Support Bulletin Board.

Subscribing to the Plexus Software Support Newsletter

The Software Support Newsletter is published four times a year by the Software Services Group at Plexus Computers. You may obtain this newsletter by purchasing an annual Software Support Contract from Plexus Computers through your local Plexus Sales Office.

PLEXUS INTRODUCES THE P/90 A HIGH PERFORMANCE ENTRY-LEVEL SYSTEM

Plexus Computers, Inc. announces the addition of a high-performance, entry-level system to its family of departmental data processing systems, the Plexus P/90. The P/90 runs under the UNIX operating system and is designed to support up to 64 simultaneous users for departmental computer applications requiring high performance at a lower cost. In addition, the system is ideally suited to act as an entry-level processor for the *Plexus XDP Extended Data Processing System*, which was introduced in March of this year. The *XDP system* is the first commercial computer which provides "mixed mode" data processing, enabling users to effectively integrate and manage images, text and alphanumeric data.

"The P/90 provides an excellent development platform for XDP and UNIX applications," said Paul Klein, president and chief executive officer of Plexus. "In addition, we have had a high level of acceptance of the P/90 in the VAR marketplace due to its low entry price and configuration flexibility."

The P/90 is based on the 32-bit MC68020 microprocessor running at 25 megahertz and executing approximately three million instructions per second, the most powerful microprocessor currently available in the Motorola family. The new computer provides expansion from two to 16 megabytes of main memory and up to 3.5 gigabytes of high-speed magnetic disk storage. It also features advances in cache memory, system memory, and use of the industry standard Small Computer Systems Interface (SCSI) for peripheral input/output (I/O).

Advanced Architecture

The high performance of the Plexus P/90 is due to its advanced multiprocessor architecture optimized for the UNIX operating system. The P/90 uses a VMEbus to link the 32-bit job processor with dedicated I/O processors for disk, communications and terminal activity. During operation, the job processor performs data processing functions. A Communications Processor handles terminal I/O, while the SCSI bus controls disk and tape I/O activity.

A sophisticated cache design and a dedicated high-speed RAM bank enhance computational performance. An industry-standard SCSI bus makes it possible to attach a variety of peripheral devices to the system. The eight-slot VMEbus extends this flexibility further to meet sophisticated original equipment manufacturers' (OEMs) requirements.

The Plexus Sys5 UNIX implementation, along with industry-standard programming languages, makes the P/90 compatible with a variety of existing applications. Additionally, it is easy to transport applications from one Plexus system to another because the P/90 is object-code compatible with the full line of Plexus 68020-based systems.

Expandability, Capacity and Price

The performance of the P/90 can be accelerated for computationally intensive arithmetic operations with the addition of an optional MC68881 floating point coprocessor.

The base system supports eight users and can be expanded to support 64 simultaneous users. The system can be configured with two to 16 megabytes of memory on a single board. The 5 1/4-inch Winchester disk drives on the P/90 offer formatted capacities of 67, 135 and 310 megabytes. Total system storage can be

expanded to 3.5 gigabytes with the addition of optional peripheral storage cabinets.

Quantity one retail pricing for the P/90 begins at \$29,500 for a base system, which includes two megabytes of main memory, 67 megabytes of magnetic disk storage and a 60 megabyte cartridge tape drive.

Sales and Service

The Plexus P/90 computer system is available immediately through Plexus sales offices in the U.S. and Western Europe, and via distributors throughout the world. Plexus provides on-site service and training for all its products through its field sales organization.

Plexus has a direct sales force with extensive knowledge and experience solving a range of computing problems, from traditional departmental computing to complex mixed-mode processing. It works closely with value-added resellers (VARs), OEMs and large-volume end users in developing applications for Plexus products.

CARTRIDGE DRIVE IMPROVEMENTS for P/35's, P/55's, and P/75's

Plexus Computers has now developed a series of new products to improve backup performance on cartridge drives. These products consist of firmware, software, or a combination of the two, with an enhanced version of software also available. Code named 'Turbo Specials', these products are now available for immediate shipment.

Earlier this year Plexus introduced an upgrade capability for cartridge drive systems called M&M firmware. This new firmware greatly improves the performance and reliability of the Archive tape driver. In some cases, Plexus has benchmarked performance improvements by some 300%. This firmware is not available for P/15 or P/20 systems.

The current version of fbackup is being deleted from future products as new streaming drivers are developed. The desire for faster and more efficient backup methods has resulted in number of new improvements:

- The IMSP driver has been overhauled and improved
- A new caching driver has been developed
- An "IBACKUP" program has been generated that features:
 - backup and restore parameter definition
 - single command for multiple logical partitions
 - multi-volume (up to 36 tapes) can be used in one backup
 - Backups can be performed in multi-user or single user mode (though there are some restrictions on multi-user)
 - automatic tape labeling
 - verification of tape contents
 - tape block sizes can be specified
- New software will work with or without M & M firmware

All of these specials may be ordered through your local sales representative.

FSUK-3 M & M UPGRADE NA

This product has been obsoleted and replaced by FSUK-7

FSUK-5 BASIC TURBO \$125

Consists of new software for the IMSP driver and the cache driver. No M & M firmware is required or included. Software is licensed by Plexus Computers to each machine by serial number and software may

not be duplicated. Volume discounts apply for quantity purchases.

FSUK-6 IBACKUP **\$200**

Consists of new software which serves as a front-end facility using UNIX drivers. Ibackup is similar to fbackup functionality but does not have a direct performance impact. This may be purchased independent of other backup packages. Software is licensed by Plexus Computers to each machine by serial number and software may not be duplicated. Volume discounts apply for quantity purchases.

FSUK-7 ENHANCED TURBO **\$650**

Consists of new hardware and software which includes:

- M & M firmware on the cartridge drive controller board
- Updated IMSP board - if the board is upgradeable. Some earlier versions may need to be replaced. Call FE for details.
- Basic Turbo software

Volume discounts apply for quantity purchases. The price does not include installation (time and materials at \$90 per hour).

FSUK-8 SUPER TURBO **\$850**

Includes Enhanced Turbo plus Ibackup software. Volume discounts apply for quantity purchases. Price does not include installation (time and materials at \$90 per hour).

Orders in a quantity of 10 or more are offered a 10% discount. VAR's are additionally entitled to Class II discounts.

**Plexus Computers, Inc.
Software Support Center**

Customer Satisfaction

Those of you who have called for software support in the last few weeks should have noticed that we have changed the manner in which we handle calls. Our goal is to be more responsive to your needs and close open calls more quickly.

What we have done is to create a position called "Response Analyst". The position is staffed from 6:30 AM until 11:30 AM Pacific time. When you call for software support, the Customer Service Center will ask for information about your system as has been done in the past and then pass the call to the Response Analyst.

The Response Analyst is charged with determining the actual nature of the call; closing it if possible, or collecting enough information for an analyst to research and provide you with an answer on a return call. To do this, the Response Analyst is limited to a maximum of 10 minutes on that call when any other calls are waiting. If the Response Analyst is on a call, your call will be queued and the Analyst will call you right back. The goal is that you receive a call back within 30 minutes.

Some things that you can do to get service quickly are:

- Have your machine serial number and contract number at hand when you call. Remember that the contract goes with a machine and is non-transferable.
- If you do not have a Comprehensive Software Support Contract (CSS), you will need to provide us with a purchase order number or an American Express number. You will not be able to talk to an analyst until the billing arrangements are in place.
- It may be necessary to dial into your machine. If special arrangements are necessary to permit access to the machine by phone are required, please make them before you need service.
- Call the Customer Service Center (previously known as the Dispatch Center) number rather than calling an analyst directly. A call that goes outside the system cannot be tracked nor do we have the flexibility to assign the call to another analyst.

As with any dynamic organization, there are changes. To bring you up to date, the following people are no longer with Plexus: Bob Gough, Ellen Richards and Mark Carlos. Theresa Ruebenson has transferred from Software Support to Marketing.

New people in the organization are: Mickey Mestel and Alan Henley, both of whom are Software Support Analysts. Ric Urrutia transferred from Software Quality Assurance and is supervising the Response Center.

Lynn Macey is the manager of the department and is responsible for providing quality support in a timely manner. If you have any problems with the support that you receive or the way it was given, please call him so that those problems can be addressed.

If you are especially pleased with the support that you received, Lynn would like to hear that as well.

Here are the results of the last quarter response and closure time for A and B calls:

<u>Month</u>		<u>Average Call Response Time</u>	<u>Average Call Closure Time</u>	<u>Closure Exceptions</u>
March	A	.8 hours	2.3 hours	2
	B	1.3 hours	6.8 hours	7
April	A	.8 hours	4.3 hours	0
	B	1.0 hours	5.5 hours	5
May	A	1.0 hours	3.5 hours	2
	B	1.3 hours	13.5 hours	11
June	A	1.5 hours	4.5 hours	7
	B	1.5 hours	12.5 hours	10

While still within our published goals, the trend has been in the wrong direction. The implementation of the Response Center is the result of our constant efforts to become more responsive to our customers.

1987 Support Services Available

Plexus Software Services provides several levels of software support as outlined below:

C.S.S. Comprehensive Software Support - based on system classification. Includes:

- (800) telephone hotline (limited to 2 hrs. per month); two authorized individuals per CPU. Phone services provided in excess of 2 hrs. per month will be billed at the rate of \$80/hour.
- Access to the Electronic Bulletin Board, with the ability to read information and offload programs/files
- Automatic distribution of software product and documentation updates
- Automatic distribution of quarterly newsletters
- Ability to report software bugs (verbally or written)
- A six-hour phone allowance for installation support services (for new systems only, if CSS is purchased at time of system order)
- Support for Plexus-approved peripherals
- Reduced consulting rates

S.U.S. Software Update Service - based on system classification. Includes:

- Access to the Electronic Bulletin Board, with the ability to read information only
- Automatic distribution of software product updates and documentation updates
- Automatic distribution of quarterly newsletters
- Ability to report software bugs (written only)
- Reduced consulting rates

B.S.S. Basic Software Support - intended for out-of-production products only, based on volume of CPU's purchased. Includes:

- (800) telephone hotline (limited to 2 hrs. per month); two authorized individuals per CPU. Phone services provided in excess of 2 hrs. per month will be billed at the rate of \$80/hour.
- Automatic distribution of quarterly newsletters

- Workaround support for Sys3 software bugs

Block Support - a choice of two quantities of telephone consultation time by a dedicated analyst:

A = 10 calls or 5 hours at \$375, whichever is reached first

B = 20 calls or 10 hours at \$650, whichever is reached first

Software Installation Service - Customers who purchase a CSS contract at the time of the system order will get 6 hours of free installation service. Otherwise, a choice of two programs are available (new system sales only):

- Telephone Service - \$480 for up to 6 hours
- On Site Service - \$500/day, plus expenses

Time & Materials - On-site consulting services, authorized by purchase order or American Express card

- for existing CSS, SUS, or BSS customers - \$80/hour
- for non-contract customers - \$100/hour

Installation Service for UNIX-based products

Customers who purchase a CSS contract at the time of the computer system order will receive 6 hours of free installation service. This service is intended to provide customers with telephone support while installing software and connecting peripherals (terminals, printers, modems) to work with the software.

All other customers may still obtain Installation Services on their computer systems during the first 90 days following shipment from the factory. There are two programs available:

Telephone Service - \$480 for up to 6 hours

On Site Service - \$500/day, plus travel & lodging expenses

Software Duplication

Software duplication services are available to produce 9-track or cartridge tapes from customer-supplied originals with prior Plexus management approval. The cost per duplicate is \$50 plus shipping. Contact Plexus Field Engineering to discuss duplication suitability.

Blank Media

Blank 9-track or cartridge media is available for \$35 each. Tapes used in the software duplication service will be billed to the customer if blank media is not supplied with the software original.

Priority Shipping

Plexus normally ships all software and documentation by UPS or US Mail, with an average delivery time of 5-8 days (domestic) or 15-21 days (international). At the customer's request, Plexus will ship via Federal Express (next day delivery, including Saturdays) or UPS Blue Label (48 hour delivery), but the cost of expediting shipments will be billed to the customer. Currently, Federal Express shipments average around \$25, and UPS Blue Label shipments average around \$15. If you require immediate receipt of software and/or documentation, inform the Plexus software support analyst you're talking to when the need for shipment is determined.

Plexus Customer Education News

The new Customer Education Course Catalogs for July - December have been mailed out and you may have noticed the addition of several new courses. These include:

UNIX for Programmers - As an intermediate course focusing on shell programming and including UNIX tools, process environment and the operating system kernel, this is an excellent course for those requiring a more powerful interface to the UNIX system.

C Programming - This course covers the entire C language and is an ideal starting point for programmers who will be working in C.

Advanced UNIX for C Programmers - For those UNIX programmers who want to take advantage of the full potential of the UNIX operating system, we offer this course that covers Make & SCCS, UNIX C Libraries, and UNIX system calls.

If you presently do not receive our catalog, and wish to, please contact the Customer Education group at 408/943-2237.

Due to popular demand, our Hardware Maintenance course has been modified from 8 days, as listed in the catalog, to (2) five day courses to allow students to concentrate on a specific type of system. The first week covers the P/35, P/55, P/60, & P/75 systems and the second week covers the P/90 & P/95 systems.

Our schedule over the next few months is as follows:

<u>Dates</u>	<u>Course</u>	<u>Location</u>
Aug 26 - 27	Introduction to UNIX	San Jose, CA
Aug 31 - Sep 4	System Administration	San Jose, CA
Sep 28 - 29	UUCP Implementation	San Jose, CA
Sep 30 - Oct 2	System Tuning	San Jose, CA
Oct 5 - 6	Introduction to UNIX	St. Louis, MO
Oct 7 - 9	UNIX for Programmers	St. Louis, MO
Oct 12 - 16	System Administration	St. Louis, MO
Oct 19 - 23	C Programming	St. Louis, MO
Oct 26 - 28	Adv UNIX for C Programmers	St. Louis, MO
Oct 19 - 23	Hardware Maintenance	San Jose, CA
Oct 26 - 30	P/90-95 Maintenance	San Jose, CA

Remember, classes fill up fast, so register early. If there are any other courses you would like to see offered, please let us know. We would appreciate hearing from you.

Technical Publications Update

UNIX Sys5.2 Documentation Updates

The following is a listing of documentation updates for Plexus version Sys5 manuals. Please note that each subsequent update is built upon its predecessor; for example, you cannot insert version E unless you have versions A, B, C, and D.

Customers covered by a software support agreement have been automatically sent one copy of each of these updates. If you would like additional sets or individual items, or if you have not received your automatic shipment, please contact Linda Allread at Plexus Field Engineering at (408) 943-2250.

If you are not covered by a software support agreement, you may purchase a complete set or individual items. Contact Linda Allread at Plexus Field Engineering at (408) 943-2250 to place your order or for more information.

Complete Set of All Sys5 Documentation Updates \$125.00

BOOK No.	TITLE	Version From - To	PUP No.	Price
98-05076.1	Sys5 UNIX Administrators Guide	A to B	98-05076.11	\$10.00
		B to C	98-40105.11	5.00
		C to D	98-40105.12	5.00
98-05080.1	Sys5 UNIX Programmer's Guide	A to B	98-40109.11	5.00
98-05081.1	Sys5 UNIX Support Tools Guide	A to B	98-40110.11	5.00
		B to C	98-40110.12	10.00
98-05083.1	Sys5 UNIX User's Guide	A to B	98-40112.11	5.00
98-05084.1	Sys5 UNIX Admin. Reference Manual	A to B	98-05084.11	15.00
		B to C	98-40113.12	5.00
		C to D	98-40113.13	15.00
		D to E	98-40113.14	5.00
		E to F	98-40113.15	5.00
98-05085.1	Sys5 UNIX Prog. Ref. Manual	A to C	98-40114.12	5.00
		B to C	98-40114.12	5.00
		C to D	98-40114.13	5.00
		D to E	98-40114.14	5.00
		E to F	98-40114.15	5.00
98-05086.1	Sys5 UNIX User's Ref. Manual	A to B	98-05086.11	10.00
		B to C	98-40115.12	5.00
		C to D	98-40115.13	5.00
		D to E	98-40115.14	5.00
		E to F	98-40115.15	5.00
98-05077.2	Sys5 UNIX Error Messages Manual	.1 to .2	98-40106.11	5.00

Feature Termino: The Tektronix 4105/4107

This month's feature terminfo entries are for the Tektronix 4105 and the Tektronix 4107 terminals. Compilation of this terminfo source file will result in creation of the terminfo files *tek4105* and *tek4107*. They may then be used as your TERM variable to permit operation of visual programs such as vi and programs relying on the **curses(3x)** features. For a full description of the meaning of each of the capabilities listed below, see **terminfo(4)** in the **Sys5 UNIX Programmer's Reference Manual**.

To compile the entry, log in as root and issue the command: **tic <filename>** where **<filename>** is the name of the terminfo source file. This will create the new entry *tek4105* and/or *tek4107* in the directory */usr/lib/terminfo*.

```
# Has sysline at top of screen.
# Has 1024 line dialog scrolling (149 for 4105)

# This termcap will work with all factory settings, except that 'code'
# must be set to ansi.

# All colors are shown in comments below. Standard color indices are:
# 0 clear, 1 white, 2 red, 3 green, 4 blue, 5 cyan, 6 magenta, 7 yellow
# Color codes for standout mode (SGR command):
# 40 clear, 41 red, 42 green, 43 yellow, 44 blue, 45 magenta, 46 cyan,
# 47 white, 49 clear
# Note: these colors change if you change your dialog area map.

#following are lines to change:
# sysline color - default green -----+
#          :ts=\E%!0\ELF7~ @\EKA0\EMG0\EMT3\E%!0:\
# screen colors - character, character background, erase
# default - cyan, blue, clear -----+++
#          :is=\E[m\E[4\E[?6;7h\E[J\E[2;32r\   |||
#          \E%!0\ENF3\ELBA@0\ELLA\?\ETD27\ELI540\E%!1:\
# alpha cursor indices - red/yellow -----++
# Begin standout mode - background color set to red (41)
# -----+++
#          :so=\E[41m:\
# End standout mode - background color set to blue (44)
# -----+++
#          :se=\E[44m:\
# Make changes in actual termcap entries below.

4107-s|tek4107-s|Tektronix 4107 with sysline,
da, db, hs, mir, msgr, xon,
cols#80, lines#31,
bel=^G, blink=\E[5m, bold=\E[32m, cbt=\E[Z,
clear=\E[H\E[J,
cnorm=\E%!0\EKDH(0\EKDH'0\EKDH*0\EKDH)0\E%!1, cr=\r,
cub=\E[%p1%dD, cub1=\b, cud=\E[%p1%dB, cud1=\E[B,
cuf=\E[%p1%dC, cuf1=\E[C, cup=\E[%i%p1%d;%p2%dH,
cuu=\E[%p1%dA, cuu1=\E[A,
cvvis=\E%!0\EKDH(1F;\EKDH'1F<\EKDH*1F:\EKDH)1F8\E%!1,
```

```

dch1=\E[P, dim=\E[36m, dl=\E[%p1%dM, dl1=\E[M,
dsl=\E%!0\ELF7~\s@\EKA0, ed=\E[J, el=\E[K,
fsl=\E%!0\EKA1\E%!1, ht=\t, il=\E[%p1%dL, il1=\E[L,
ind=\ED, invis=\E[34m,
is2=\E[m\E[4\E[?6;7h\E[J\E[2;32r\t\s\s\s\E%!0\ENF3\ELBA@0\ELLA?\ETD27\ELI540\E%!1,
kbs=\b, kcub1=\E[D, kcud1=\E[B, kcu1=\E[C,
kcuu1=\E[A, khome=\EOP, lf0=f1, lf1=f2, lf2=f3,
lf3=f4, lf4=f5, lf6=f7, lf7=f8, lf8=F1, lf9=F2,
rc=\E8, rev=\E[7m, ri=\EM,
rmcup=\E%!0\ENF3\ELBA@0\E%!1\E[J, rmir=\E[4l,
rmso=\E[44m, rmul=\E[24m, sc=\E7, sgr0=\E[m,
smcup=\E%!0\ENF0\ELBA?\E%!1, smir=\E[4h, smso=\E[41m,
smul=\E[4m, tsl=\E%!0\ELF7~\s@\EKA0\EMG0\EMT3\E%!0,

```

#

4105-s|tek4105-s|Tektronix 4105 with sysline,

```

da, db, hs, mir, msgr, xon,
cols#80, lines#29,
bel=^G, blink=\E[5m, bold=\E[32m, cbt=\E[Z,
clear=\E[H\E[J,
cnorm=\E%!0\EKDH(0\EKDH'0\EKDH*0\EKDH)0\E%!1, cr=\r,
cub=\E[%p1%dD, cub1=\b, cud=\E[%p1%dB, cud1=\E[B,
cuf=\E[%p1%DC, cuf1=\E[C, cup=\E[%i%p1%d;%p2%DH,
cuu=\E[%p1%DA, cuu1=\E[A,
cvvis=\E%!0\EKDH(1F;\EKDH'1F<\EKDH*1F:\EKDH)1F8\E%!1,
dch1=\E[P, dim=\E[36m, dl=\E[%p1%dM, dl1=\E[M,
dsl=\E%!0\ELF7~\s@\EKA0, ed=\E[J, el=\E[K,
fsl=\E%!0\EKA1\E%!1, ht=\t, il=\E[%p1%dL, il1=\E[L,
ind=\ED, invis=\E[34m,
is2=\E[m\E[4\E[?6;7h\E[J\E[2;30r\t\s\s\s\E%!0\ENF3\ELBI5\ELLA=\ETD27\ELI540\E%!1,
kbs=\b, kcub1=\E[D, kcud1=\E[B, kcu1=\E[C,
kcuu1=\E[A, khome=\EOP, lf0=f1, lf1=f2, lf2=f3,
lf3=f4, lf4=f5, lf6=f7, lf7=f8, lf8=F1, lf9=F2,
rc=\E8, rev=\E[7m, ri=\EM,
rmcup=\E%!0\ENF3\ELBA@0\E%!1\E[J, rmir=\E[4l,
rmso=\E[44m, rmul=\E[24m, sc=\E7, sgr0=\E[m,
smcup=\E%!0\ENF0\ELBA=\E%!1, smir=\E[4h, smso=\E[41m,
smul=\E[4m, tsl=\E%!0\ELF7~\s@\EKA0\EMG0\EMT3\E%!0,

```

#

Below are the termcap entries for those applications that require them. To install the termcap entries, simply use your favorite text editor to add them onto your existing */etc/termcap* file. A good idea is to put these entries near the top of the */etc/termcap* file so that applications will find the tek4105 or tek4107 termcaps quicker.

4107-s|tek4107-s|Tektronix 4107 with sysline:\

```

:al=\E[L:AL=\E[dL:bs:bt=\E[Z:cd=\E[J:ce=\E[K:cl=\E[H\E[J:\
:cm=\E[%i%d;%dH:co#80:da:db:dc=\E[P:dl=\E[M:DL=\E[%dM:\
:do=\E[B:DO=\E[%dB:ts=\E%!0\ELF7~ @\EKA0\EMG0\EMT3\E%!0:\
:fs=\E%!0\EKA1\E%!1:hs:ds=\E%!0\ELF7~ @\EKA0:ei=\E[4l:im=\E[4h:
is=\E[m\E[4\E[?6;7h\E[J\E[2;32r\t\s\s\s\E%!0\ENF3\ELBA@0\ELLA?\ETD27\ELI540\E%!1:\

```

```

:kb=^H:ku=\E[A:kd=\E[B:kh=\EOP:kl=\E[D:kr=\E[C:\
:I0=f1:I1=f2:I2=f3:I3=f4:I4=f5:I6=f7:I7=f8:I8=F1:I9=F2:\
:IA=F3:IB=F4:IC=F5:ID=F6:IE=F7:IF=F8:LE=\E[%dD:li#31:\
:mb=\E[5m:md=\E[32m:me=\E[m:mh=\E[36m:mi:mk=\E[34m:mr=\E[7m:\
:ms:nd=\E[C:pt:rc=\E8:RI=\E[%dC:sc=\E7:se=\E[44m:sf=\ED:so=\E[41m:\
:sr=\EM:te=\E%\!0\ENF3\ELBA@0\E%\!1\E[J:ti=\E%\!0\ENF0\ELBA?\E%\!1:\
:ue=\E[24m:up=\E[A:UP=\E[%dA:us=\E[4m:\
:ve=\E%\!0\EKDH(0\EKDH'0\EKDH*0\EKDH)0\E%\!1:\
:vs=\E%\!0\EKDH(1F;\EKDH'1F<\EKDH*1F\:\EKDH)1F8\E%\!1:\

```

#

4105-s|tek4105-s|Tektronix 4105 with sysline:\

```

:al=\E[L: :AL=\E[%dL:bs:bt=\E[Z:cd=\E[J:ce=\E[K:cl=\E[H\E[J:\
:cm=\E[%i%d;%dH:co#80:da:db:dc=\E[P:dl=\E[M:DL=\E[%dM:\
:do=\E[B:DO=\E[%dB:ts=\E%\!0\ELF7'~ @\EKA0\EMG0\EMT3\E%\!0:\
:fs=\E%\!0\EKA1\E%\!1:hs:ds=\E%\!0\ELF7'~ @\EKA0:\
:ei=\E[4l:im=\E[4h:\
:is=\E[m\E[4l\E[?6;7h\E[J\E[2;30r\
  \E%\!0\ENF3\ELB15\ELLA=\ETD27\ELI540\E%\!1:\
:kb=^H:ku=\E[A:kd=\E[B:kh=\EOP:kl=\E[D:kr=\E[C:\
:I0=f1:I1=f2:I2=f3:I3=f4:I4=f5:I6=f7:I7=f8:I8=F1:I9=F2:\
:IA=F3:IB=F4:IC=F5:ID=F6:IE=F7:IF=F8:\
:LE=\E[%dD:li#29:\
:mb=\E[5m:md=\E[32m:me=\E[m:mh=\E[36m:mi:mk=\E[34m:\
:mr=\E[7m:ms:nd=\E[C:pt:rc=\E8:RI=\E[%dC:\
:sc=\E7:se=\E[44m:sf=\ED:so=\E[41m:sr=\EM:\
:te=\E%\!0\ENF3\ELBA@0\E%\!1\E[J:\
:ti=\E%\!0\ENF0\ELBA=\E%\!1:\
:ue=\E[24m:up=\E[A:UP=\E[%dA:us=\E[4m:\
:ve=\E%\!0\EKDH(0\EKDH'0\EKDH*0\EKDH)0\E%\!1:\
:vs=\E%\!0\EKDH(1F;\EKDH'1F<\EKDH*1F\:\EKDH)1F8\E%\!1:\

```

#

Public Domain Software Spotlight

by Lynn Macey

In this issue, the Public Domain Spotlight shines on something a little different. Working with dates in UNIX is always a challenge and this library package makes it a little easier. The demo program for the package is *phoon* and it will show you approximately what the moon looks like if you were outside. Given that UNIX types are often kept in inside offices and dark basements, this could be a real advantage.

Another demo program is *delttime*. It calculates the difference between two dates. You were wondering how old you are in days, how long since your last cup of coffee? Well then, this is the program for you.

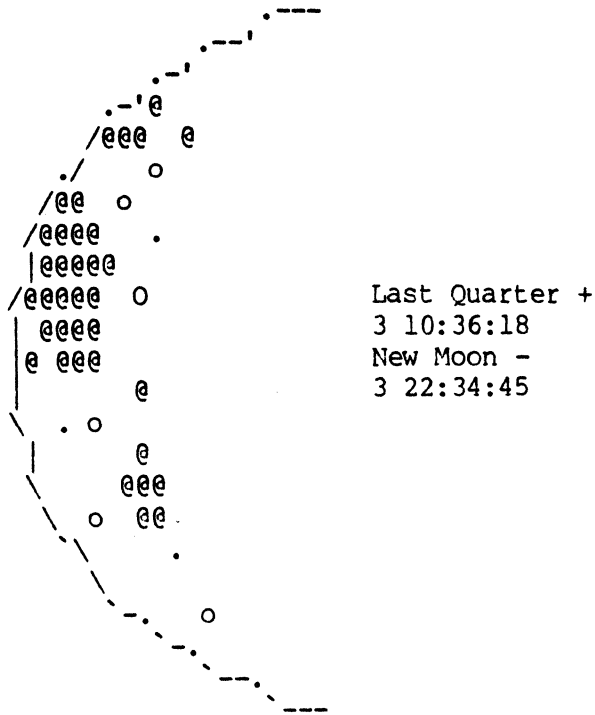
The library itself allows the programmer to parse dates into an internal format and then manipulate it. A distinct advantage is the fact that the package understands dates prior to 1970. Routines are included to convert UNIX date structures to internal form, GMT, Julian or printable string. Date strings, in very flexible formats, can be parsed into internal form. Date structures can be copied and compared.

The author of the package is Jef Poskanzer of UniSoft Systems in Berkeley. The package was distributed via Usenet but some bugs were found and fixed.

The software is packaged as a compressed tar file and can be downloaded as *phoon.tar.Z* located in the directory */usr/spool/uucppublic*. It contains the following files:

README	the author's introduction
Makefile	builds the package
delttime.c	source for time subtraction tool
delttime.man	manual for time subtraction tool
dtime.c	source for most of the time routines
dtimep.lex	source for time-parsing routine
lexedit.sed	script to modify output of lex
lexstring.c	front end for time-parsing routine
libtws.man	manual for time library
parsetime.c	source for test program
phoon.c	source for phase of moon program
phoon.man	manual for phase of moon program
tws.h	include file for time library

A quick preview of phoon's output:



Tips and Techniques for Your Plexus System & Software

compiled by the Staff at Plexus Software Support

1. *How can I have my own personal default printer destination ?*

The **LPDEST** environment variable may be set by the user to cause printouts to be directed to some printer other than the system default for that user. Example,

```

LPDEST=lp2
export LPDEST                # Bourne shell
or
setenv LPDEST lp2           # C shell
```

will cause this user's printouts to go to printer **lp2** instead of the system default printer. This is handy for users who do a lot of special application work which must be sent to particular printers.

2. *What is a neat way of removing files who have tricky file names ?*

The **-q** option to **ls(1)** causes non-printing characters in file names to be represented by the '?' character, making it much easier to remove such files.

3. *How can I prevent against removing files accidentally ?*

The **-i** option to **rm(1)** causes **rm** to prompt the user for a yes or no response before removing a file. All users are encouraged to set this option at all times:

```

alias rm /bin/rm -i          # C shell
or
rm () /bin/rm -i $*         # Bourne shell
```

Will cause **rm** to prompt you every time you wish to delete a file.

4. *How can I have certain commands done in the Bourne shell at logout time ?*

Users of the C shell have always been able to have the shell perform certain functions at logout time (such as clearing the screen and printing a fortune). Users of the Bourne shell may now have the same functionality by using the **trap** command. Place the following statement in your *.profile*, and any statements appearing in your *.logout* file will be executed when you log out:

```
trap "trap 0; . $HOME/.logout; exit 0" 0
```

5. *My terminal doesn't handle lines correctly in vi when they get longer than 80 characters.*

If your terminal has a problem with wrap-around in an editor such as **vi**, disabling the automargin (*am*) capability in terminfo/termcap will take care of this. Try typing in 81 characters on a line, and then deleting the first character. If the line disappears and you need to do a **^L** to get it back again, then this will take care of the problem. DEC terminals and many that emulate DEC are set up so that when the 80th character is typed on a line, the cursor remains in that position until the

next character is typed. At that point, a new line is inserted before the character is displayed and the cursor is moved to the next line. This works fine when cat'ing a file, for instance, but *vi* uses autowrap for redisplay and depends on the cursor being on the next line for autowrap after a wrap. This will clear up the problem for any DEC ANSI type terminals.

6. *Is there an easy way to clean out directories that accumulate temporary files?*

The **uuclean(1M)** command is quite useful for this purpose. It's initial function is to periodically clean out the */usr/spool/uucp* directory, but by using the **-d** *directory* option, you can specify directories other than the default spool directory. The **-ppre** deletes files with *pre* as the file prefix, and **-ntime** specifies files which are older than *time* hours. Other options available are listed in the Administrator's Reference Manual entry. This command should be set up to run from **cron(1M)**.

7. *How can I keep users from logging on when I need to do system work in multi-user mode?*

A clean way to do this is to create a file in */etc* with a name like *.nologin*. Then in the system startup files */etc/cshprofile* and */etc/profile*, the first thing you do is test to see if this file exists. If it does, and the person is not either superuser or someone you designate to be able to log in, then you can kill the process immediately, i.e. 'kill \$\$'. This will allow you to do work in multi-user mode without worrying about someone logging on without your knowledge. Of course you must remember to remove the *.nologin* file when you are done, or your users will not be able to get back on at all. Also, make sure you set up the startup files correctly, or you yourself will not be able to get on the system!

8. *How can I customize my mailx environment?*

Along with the global start-up file for mailx, */usr/lib/mailx/mailx.rc*, you can add a file in your home directory called *.mailrc*. This way you can customize your own environment beyond what (if anything) is set for global defaults. The most common are setting a PAGER command, such as */usr/bin/more*, so your messages scroll nicely, setting the 'crt' variable so that any messages with more than crt lines are sent through the PAGER, and setting your mailx prompt. There exist a wide variety of options under the **mailx(1)** command for setting your mailx environment to what is most efficient and comfortable for you.

9. *How can I tell what files are on a cpio tape?*

Assuming that you created the tape using cpio with the following parameters: `cpio -odumBv > /dev/rrm/0m`, you can read the directory of the files stored on tape with the following command:

```
cpio -itvB < /dev/rrm/0m .
```

If you wish the file listing to be saved into a file named *tapelist*, so you can view it later, execute the following command:

```
cpio -itvB < /dev/rrm/0m > tapelist
```

10. *How can I send binary files via cu?*

The **uuencode(1)** and **uudecode(1)** commands allow encoding and decoding of binary files for transmission using cu. The user on the sending side executes the following command:

```
uuencode file1 file2 > file3
```

file1 is the binary file. **file2** is what the destination file will be named after **uudecode** is run on **file3**. **file3** is the ASCII file created from **file1**.

The user can then use a **put** to send **file3** to the remote site. The user on the remote side then executes the following command:

```
uudecode file3
```

This will create a binary destination file named **file2** which is now identical to your original source. **file3** during this process will not be altered or removed by **uudecode**.

11. *There is a bug in DBLOAD for INFORMIX in the following products: INFORMIX-SQL Releases 2.00.05 and earlier; INFORMIX-4GL releases 1.00.05 and earlier; and INFORMIX-ESQL/C release 2.00.05 and earlier. This problem occasionally causes core dumps in DBLOAD if more than 32 columns are being loaded.*

The general workaround to this problem is to split the table into subtables of 32 columns or less. Multiple work tables, each containing part (no more than 32 columns) of the schema definition of the table to be loaded, must first be created via the RDSQL CREATE TABLE statement. The first work table contains columns 1-32 of the master table, the second work table columns 33-64, and so forth. Care must be taken to ensure that the work tables are newly created before use, this procedure will fail if any rows have ever been present in either table before being loaded using DBLOAD.

Once the work tables are loaded, the desired rows are inserted into the master table by selecting all columns from the two work table in sequence and joining upon the rowid. This column is a "hidden" column internal to the table and contains the physical record sequence number of the row in the table's data file **.dat**. The resulting table from this select is then inserted into the master table to complete the load. After the load is satisfactorily completed, the work tables may be dropped.

12. *How do I get my INFORMIX menus to work using the C shell?*

In the MENU ENTRY FORM at the "Program to Execute" field, you must specify the absolute path name of the INFORMIX program you wish to execute (i.e. **/usr/informix/bin/sacego -q mail**). You do not need to specify absolute path names if you are using the Bourne shell.

13. *How do I get a HP Laserjet printer to do "landscape" printing in Q-one?*

Make a copy of the laser printer printcap files in the **/usr/qlib/files** directory giving it a different name (i.e. **HPLASERL.pc**). Edit the file to give it the appropriate escape codes for wide documents. Then define that printer to Q-one as **HPLASERL**, and call that printer when you want to do landscape printing.

Using FSCK— A Guide to the UNIX File System Check Program

by Michael S. Saxon

The UNIX File System Check program, *fsck*, checks the disk to make sure a proper environment exists in which to create and maintain files. This is called "consistency". If the file system is inconsistent, corrective action must be taken before the file system can be used.

This article is basically a rewrite of the article "FSCK—The UNIX/TS File System Check Program" by T. J. Kowalski of Bell Laboratories. While Kowalski's article provides a complete portrait of *fsck*, it is not written in a fashion which is understandable by the non-technical personnel who are now called upon to use this life saving program.

In some instances, general information not provided in Kowalski's article has been added to make some portions of *fsck* more clear. In other cases, some more esoteric items have been left out. For UNIX purists, the original article remains the "Bible" on *fsck*. This article attempts to fill the gap between the dangerous ignorance with which *fsck* is often used today and the gospel version which precedes it.

Using fsck

fsck should be used whenever the UNIX operating system is rebooted. UNIX maintains the integrity of the file system on its own while it is running.

When *fsck* checks a file system, it does so in multiple passes. It is because of the multi-pass nature of *fsck* that the file system must be quiescent when *fsck* is run. If this is not the case, the running computer system could undo changes being made by *fsck* or could crash as a result of changes made by *fsck*.

Lastly, as certain information concerning the file system is kept in memory at all times—see the discussion of the superblock, below—it may be necessary to reset the computer immediately after running *fsck*. This is because if a 'sync' operation is performed, the old version of the superblock will be rewritten to the disk, thus overwriting the newly corrected version.

When a UNIX system is in multiuser mode, the sync operation is usually performed every 30 seconds by the *init* process. This means that changes made to a system in multiuser mode by *fsck* could be undone by the update process. This is another reason why *fsck* should always be run in single-user mode.

With the exception of the root filesystem on Plexus machines, *fsck* can only be run on unmounted file systems to prevent these problems from occurring. This is because the superblocks of mounted file systems are updated by UNIX while it running. If it is impossible to unmount the file system in question, the system must be reset after *fsck* is run if any changes were made. *fsck* will warn you that this is necessary with the message "BOOT UNIX NO SYNC".

If *fsck* finds an inconsistency in a file system, it reports it immediately to the operator and asks the operator whether or not to take action. The reason why *fsck* does not take any action on its own is that there may be multiple courses of action available. Also, the operator may be working on a known bad file system and may not wish to take any steps which would do further damage while in the process of trying to save

files.

fsck must create a number of large tables in order to check a file system. Usually these tables are created in the main memory. If there is not enough memory space available for all the tables, *fsck* will prompt the operator for a place to create a scratch file. Alternatively, the *-t* option (described below) can be used to specify which file will be used.

Remember that each file system on a computer is an individual unit. Therefore, *fsck* must be run for each file system.

FILE SYSTEM CONSISTENCY

In order to maintain the consistency of a file system, UNIX is very careful about how and when the disk is updated. When this process is disturbed, corruption of the file system can occur. How does this happen?

The most common reason why a file system becomes corrupted is that it is improperly shut down. This can often be result of a power failure. It can also be due to operator error, such as pressing the reset button while the system is running or forgetting to use the 'sync' command prior to resetting the system or turning it off.

Any actual hardware failure, especially of the disk, itself, or its controller, can also result in corruption of a file system.

Another thing to remember about corrupt file systems is that they become worse with time. A corrupt file system, if not repaired, will undoubtedly become more corrupt. The results can be disastrous.

WHAT CAN BECOME CORRUPTED?

One of the most common places for corruption to occur is the superblock. Every file system has a superblock which contains the most basic parameters about the file system which UNIX needs to have on hand at all times. Some of these parameters include things like the size of the file system, the number of free blocks and the number of free inodes. The superblock is stored in block 1 of the file system.

As UNIX needs the information in the superblock at all times, a copy of it is kept in the main system memory. In order to keep the version of the superblock on the disk up to date, the superblock is periodically copied from memory to the disk. This operation is called 'sync'.

On a system running UNIX in multiuser mode, you will notice that a process called */etc/init* is always running. One of the tasks of *init* is to perform a 'sync' operation every thirty seconds to update the superblock.

The superblock is changed so often that if a system is powered down without performing a final 'sync' there is a good chance that the version of the superblock in main memory—the current version, will not be the same as the version on the disk which is now out of date. This is the most common form of corruption.

Any file, directory or device on a UNIX system is described by an *inode*. The inode is a block which describes the file and contains information such as the size of the file and

last time of modification. Additionally, the inode contains information about the location of the file's contents on disk.

An individual inode, or the main list of inodes for the file system, can also become damaged. For example, each inode must be a file, a directory, a character device or a block device. If an inode does not have one of those four types, it is considered invalid.

Each inode which is allocated (in-use) must be linked to one or more directory entries. The link count, the number of directory entries for an inode, is one of the things which is stored in the inode and must match the number of actual directory entries found in the file system for that inode.

There is a list of blocks associated with every inode. If a block appears in the block list for more than one inode, it is considered to be a 'duplicate' block. This is really a poor name because the block, itself, is not duplicated. In reality, the block is claimed by more than one inode.

Another possible problem in a file system is a block with a bad number - either lower than the lowest block number or higher than the highest. The list of free blocks on the system can also be checked for such an inconsistency.

The size of a file can also be checked. Each file has a 32-bit integer which contains the number of bytes in the file. This can be checked to see the number of bytes is consistent with the number of blocks allocated to that file. Also, UNIX directory files can be checked to ensure that their size in bytes is a multiple of 16.

Directories can be further checked to insure that correct links exist from the directory to itself and from the directory to its parent. Directories can also be checked for 'connectivity'. This means that all files in the file system are accessible from the file system through its network of directories. If a file or a directory cannot be accessed, it is considered to be "orphaned".

THE lost+found DIRECTORY

If *fsck* finds a file or directory which has been orphaned, it needs a location where it can link that file or directory back into the file system. This is the "*lost+found*" directory.

The *lost+found* directory must always be created in the root directory of the file system. In the case of the first file system on a computer, this would be the system root directory ("/"). However, in the case of other file systems, the *lost+found* directory must be located in the root directory of that file system.

For example, if the file system */dev/dk2* is mounted as the directory *u* then a *lost+found* directory must be created under */u* because it is the root directory of *dk2*.

The *lost+found* directory is made using the *mkdir* command. Then some files are created under that directory and then removed. This is because it is advisable that *fsck* never have to write on virgin areas of the disk (you may be working on a bad disk).

If some files are created within the *lost+found* directory and then removed, the space taken up by those directory entries (16 bytes per file) is not released. That space remains a part of the directory, itself. Therefore, if an orphaned inode is linked into that directory, the 16 byte directory entry is written into a directory entry slot which has

already been allocated but not currently in use. This prevents the possibility of further corruption were it necessary for *fsck* to claim new space for the entry.

The following procedure is recommended for creating the *lost+found* directory.

login as root

```
# cd / #(or cd /u, etc...)  change to the root directory
# mkdir lost+found         # make the lost+found directory
# cd lost+found
# for i in a b c d e f g h i j k l m n o p
> do
> > $i                    # use a shell loop to create a
> done                    # few null files
> rm *
```

This procedure should always be performed on a newly created file system.

If *fsck* finds an orphaned file and links it back into the file system, it has to give the file a unique name. Therefore, it simply uses the inode number, which is always unique, as the file name. For example, if inode number 986 is recovered on file system *dk0*, it will be accessible when *fsck* is finished as */lost+found/000986*.

If a whole directory has been orphaned, it is relinked into *lost+found* as a directory and the file names under that directory will be correct.

fsck RUN-TIME OPTIONS

fsck has a number of run-time options. However, some are extremely dangerous and should be used with care.

-b option:

The *-b* option when set will automatically reboot your system when you get the message "BOOT UNIX: NO SYNC!". This option does not work on Plexus P/15's and P/20's and z8000-based machines.

-y option:

The *-y* option tells *fsck* to assume that the answer to all operator questions will be "yes". This is an extremely dangerous option as *fsck* will move unconditionally through the file system.

The *-y* option should be used with EXTREME CAUTION.

-n option:

The *-n* option tells *fsck* to assume that the answer to all operator questions will be "no". This can be used safely as a "no" answer to all questions means that the disk will not be written on at all, only read. It can be useful to use this option in order to see the extent of the damage to the file system before

making any changes.

-s option:

The -s option will force a new free list to be created from scratch for the file system. Special arguments to this option can permit optimal organization of the free list. The -s option should not be used except by qualified personnel.

-S option:

The -S option is similar to the -s option except that the free list will only be recreated if NO discrepancies are found in the file system. This is used for forcing the free list to be reorganized if the file system is operating fine. It also assumes the -n option so that no operator responses are necessary. If any discrepancies are found in the file system, the free list is not reorganized. There is usually no need for this option to be used. Special arguments to this option can also permit optimal organization of the free list.

-t option:

fsck uses main memory to keep the large tables that it needs in order to check the file system. If there is not enough memory space available, *fsck* will prompt the operator for a place to create a scratch file. If the -t option is used, the file named as the next argument will be used as the being checked. Also, if the file is a plain disk file which did not exist before, it will be removed when *fsck* is finished.

file system:

Following the above options, the user can specify the names of the file systems to be checked. If no list of file systems is given, *fsck* will use the list found the file */etc/checklist*.

fsck ERROR MESSAGES - Initialization Stage

The possible error messages from *fsck* can be broken down into various stages. The errors in the initialization stage result from bad parameters or hardware problems.

C option?

An option 'C' was given which is not one of the valid options (-b, -y, -n, -s, -S, -t). See the section on run-time options. If this error occurs, *fsck* will terminate immediately.

Bad -t option

The -t option was used but was not followed by a file name. See the section on run-time options. If this error occurs, *fsck* will

terminate immediately.

invalid -s argument, defaults assumed

The -s option was not suffixed by a "3","4" or values for the number of blocks per cylinder and the number of blocks to skip. The values which were used when the file system was created are assumed. If no such values were given, the values of 400 blocks per cylinder and 9 blocks to skip are used (400:9). See **UNIX User's Reference Manual fsck(1)** or **fsck(8)** for more information.

Incompatible options: -n and -s

It is not possible to change the free list without modifying the file system. Therefore, the -s and -n options can not be used at the same time. See the section on run-time options. If this error occurs, *fsck* will terminate immediately.

Can't get memory

fsck requested memory in order to create its tables. The request failed. This error should never occur. If it does, contact Plexus Software Support. If this error occurs, *fsck* will terminate immediately.

Can't open checklist file: <file name>

The checklist file could not be opened for reading. This is usually /etc/checklist, and the file should be set to allow read access. If this error occurs, *fsck* will terminate immediately.

Can't stat root

One of the first things which *fsck* tries to do is to get information on the root directory. This message occurs if the attempt fails. This error should never occur unless you are trying to run *fsck* on a disk which has no file system. If it does, contact Plexus Software Support. If this error occurs, *fsck* will terminate immediately.

<file name> is not a block or character device

The name which you have given to *fsck* as the name of the file system to be checked is not correct. It is a plain file rather than a device, If this error occurs, *fsck* will ignore this device name and go on to the next one in the checklist.

Can't open <file name>

The named file system could not be opened for reading. If this error occurs, *fsck* will ignore this device name and go on to the

next one in the checklist. The permissions of the file system should be changed to allow read access.

Size check: `fsize <file system size> isize <inode list size>`

fsck has found that the inode list has more blocks in it than the whole of the file system. Otherwise, there are more than 65,535 inodes in the file system. If this error occurs, *fsck* will ignore this device name and go on to the next one in the checklist.

Can't create <file name>

fsck tried to create a scratch file for its tables with the above file name. If this error occurs, *fsck* will ignore this device name and go on to the next one in the checklist. The access mode of the file should be checked.

CAN NOT SEEK: BLK <block number> (CONTINUE ?)

fsck tried to move to the specified block in the file system and failed. This error should never occur. If it does, contact a Plexus Software Support. If the error does occur, you can try to continue by answering YES or stop by answering NO. If you continue, the file system may not be completely checked and *fsck* should be run again to insure file system consistency. However, if the error occurred in certain sensitive parts of the disk, *fsck* will terminate with the message "Fatal I/O error".

CAN NOT READ: BLK <block number> (CONTINUE ?)

fsck tried to read the specified block in the file system and failed. This error should never occur. If it does, contact Plexus Software Support. If the error does occur, you can try to continue by answering YES or stop by answering NO. If you continue, the file system may not be completely checked and *fsck* should be run again to insure file system consistency. However, if the error occurred in certain sensitive parts of the disk, *fsck* will terminate with the message "Fatal I/O error".

CAN NOT WRITE: BLK <block number> (CONTINUE ?)

fsck tried to write the specified block in the file system and failed. This means that the disk is write-protected. If this error occurs, contact a Plexus software Support. If the error does occur, you can try to continue by answering YES or stop by answering NO. If you continue, the file system may not be completely checked and *fsck* should be run again to insure file system consistency. However, if the error occurred in certain sensitive parts of the disk, *fsck* will terminate with the message "Fatal I/O error".

fsck ERROR MESSAGES - Phase 1: Check Blocks and Sizes

In this phase, *fsck* checks all inodes. A type check is performed on each inode to insure that it has a valid file type (file, directory, blocked device or character device). Also, inode block numbers are examined for bad or duplicate blocks, the size of each file is checked and the format of each inode is checked. Lastly, a table is set up for inodes which have no links to them.

UNKNOWN FILE TYPE I=<inode number> (CLEAR ?)

This inode did not have one of the four valid file types (see above). A YES answer will clear this inode. Note that this will also result in an "unallocated" error condition during Phase 2 for any directory entry which points to this inode. A NO answer will ignore the error condition and leave the file with it's invalid type.

LINK COUNT TABLE OVERFLOW (CONTINUE ?)

When the *fsck* program is created, a maximum size is set internally for the table which contains the number of inodes which have no links to them. If it happens that there are more orphaned inodes than the table can hold, this error message will be printed.

Since it is not possible for the user to re-create the *fsck* program, the best option is to answer YES. However, a complete check of file system is not guaranteed and *fsck* will have to be run again until all of the zero-link files can be dealt with. A reply of NO will cause the *fsck* program to terminate.

<block number> BAD I=<inode number>

This inode contains the given illegal block number. This means that the block number is either lower than the first data block in the file system or higher than the last one. There is nothing which can be done about this error and if there are a lot of bad blocks, the "excessive bad blocks" error will occur. Also, any block found to be bad will result in the "bad/dup block" error message in phase 2 and phase 4.

EXCESSIVE BAD BLKS I=<inode number> (CONTINUE ?)

There are more than 10 blocks with bad block numbers (see above) associated with the given inode number. A response of YES will ignore the rest of the blocks in this inode and continue checking the file system with the next inode. However, a complete check of the file system is not guaranteed and *fsck* should be run again. A NO answer will terminate the program.

<block number> DUP I=<inode number>

This inode contains the given block number which is also associated with another inode. There is nothing which can be

done in response to this message and if there are a lot of duplicate blocks, the "excessive dup blocks" error will occur. If any bad blocks are found, phase 1B will be run so that *fsck* can complete the list of duplicate blocks. This error will also result in the "bad/dup" error message in phase 2 and phase 4.

EXCESSIVE DUP BLKS I=<inode number> (CONTINUE ?)

There are more than 10 duplicate blocks associated with the given inode number. A response of YES will ignore the rest of the blocks in this inode and continue checking the file system with the next inode. However, a complete check of the file system is not a guaranteed and *fsck* should be run again. A NO answer will terminate the program.

DUP TABLE OVERFLOW (CONTINUE ?)

When the *fsck* program is created, a maximum size is set internally for the table which contains the list of duplicate blocks. If it happens that there are more such blocks than the table can hold, the above error message will be printed. Since it is not possible for the user to recreate the *fsck* program, the best option is to answer YES. However, a complete check of the file system is not guaranteed and *fsck* will have to be run again until all of the duplicate blocks can be dealt with. A reply of NO will cause the *fsck* program to terminate.

POSSIBLE FILE SIZE ERROR I=<inode number>

The size of this file does not match the actual number of blocks which have been allocated to it. This is only a warning and no action is required. This error often results from files which have been created in "random order", meaning that open space may exist in the middle of the file and that no blocks have yet been allocated for that space. Therefore the actual file size is larger than the number of blocks would indicate. While this is not dangerous, the message can usually be eliminated by copying the file into another file and removing the original.

DIRECTORY MISALIGNED I=<inode number>

This size of this directory is not a multiple of 16, the size of an individual directory entry. This is only a warning and does not require operator action.

PARTIALLY ALLOCATED INODE I=<inode number> (CLEAR ?)

This inode is neither allocated (in use) or unallocated (in the free inode list). A YES answer will properly de-allocate the inode by setting its contents to zero and adding it to the free inode list. A NO answer will cause the error to be ignored.

ERROR MESSAGES - Phase 1B: Rescan For More Dups

Whenever an inode is scanned, all of its block numbers are added to the block number table. If an inode is found at a later date which contains a block of the same number, the block is considered to be duplicated and is added to the duplicated block table. However, it is necessary to then rescan the file system to identify the inode which was first found to claim that block.

Consequently, whenever a duplicate block is found, Phase 1B is automatically run to rescan the file system. This will insure that all of the inodes associated with a specific bad block are found. Only the duplicate block error condition can result from Phase 1B. You can manually determine from Phase 1 and Phase 1B which inodes contain overlapping blocks.

`<block number> DUP I=<inode number>`

This inode contains the given block number which is also associated with another inode. There is nothing which can be done in response to this message and if there are a lot of duplicate blocks, the "excessive dup blocks" error will occur.

ERROR MESSAGES - Phase 2: Check Pathnames

In this phase, *fsck* removes directory entries related to bad inodes and then checks all directories for references to inodes which are out of range or not allocated. It also checks the root directory for correct mode and status bits.

ROOT INODE UNALLOCATED. TERMINATING

The root directory inode (usually inode number 2) is not correctly marked as allocated. As this should never happen, *fsck* will automatically terminate.

If this error should occur, contact Plexus Software Support.

ROOT INODE NOT DIRECTORY (FIX ?)

The root directory inode (usually inode number 2) was found not to be of the directory type. A YES answer will cause *fsck* to try to fix the type of the root inode. However, if the data blocks associated with the root inode are not found to have directory entries, a VERY large number of error messages will follow.

A NO answer will cause *fsck* to terminate.

DUPS/BAD IN ROOT INODE (CONTINUE ?)

Duplicate or bad blocks were found in the root inode (usually inode number 2) during Phase 1 or Phase 1B. A YES answer will cause *fsck* to ignore the DUPS/BAD error condition and to attempt to continue checking the file system. However, if the root inode is not correct, a large number of other error conditions may result. A NO answer will cause *fsck* to terminate.

<inode number> OUT OF RANGE I=<inode number> NAME=<file name> (REMOVE ?)

A directory entry with the given name has this inode number which is out of range, either smaller than the first inode number in the file system or bigger than the last one. In other words, directory entry is useless as it refers to a file which does not exist. A YES answer will cause the directory entry to be ignored.

UNALLOCATED I=<inode number> OWNER=<user id> MODE=<mode bits>
 SIZE=<size in bytes> MTIME=<time last modified>
 NAME=<file name> (REMOVE ?)

A directory entry has been found with the given name, owner, mode, size and last time of modification. The inode, however, has no allocation bits, meaning it is currently unallocated. This can happen when the system has only partially created a file when the system crashes. A YES answer will clear the directory entry which is in error. A NO answer will ignore the error.

DUP/BAD I=<inode number> OWNER=<user id> MODE=<mode bits>
 SIZE=<size in bytes> MTIME=<time last modified>
 DIR=<file name> (REMOVE ?)

Duplicate or bad blocks were found in the directory entry for this directory inode during Phase 1 or Phase 1B. This means that the directory entry is probably corrupted. A YES answer will remove the damaged directory entry. A NO answer will not.

DUP/BAD I=<inode number> OWNER=<user id> MODE<mode bits>
 SIZE=<size in bytes> MTIME=<time last modified>
 FILE=<file name> (REMOVE ?)

Duplicate or bad blocks were found in the directory entry for this file inode during Phase 1 or Phase 1B. This means that the directory entry is probably corrupted. A YES answer will remove the damaged directory entry. A NO answer will not.

ERROR MESSAGES - Phase 3 - Check Connectivity

In this phase, *fsck* checks to make sure that all files and directories are properly connected to the file system. It will find unreferenced directories and, if possible, reconnect them to the *lost+found* directory.

UNREF DIR I=<inode number> OWNER=<user id> MODE=<mode bits>
 SIZE=<size in bytes> MTIME=<time last modified> (RECONNECT ?)

A directory was found with this inode number which is not connected to the file system. A YES answer will result in the directory being reconnected to the *lost+found* directory. The inode number will be used as the file name. This may result in the "CONNECT" error below.

SORRY. NO *lost+found* DIRECTORY

There is no *lost+found* directory in the root directory of the file system or the permission on the *lost+found* directory do not allow reading and writing. You have just made a request to reconnect a directory in *lost+found*. Your request will be ignored and the directory will remain unconnected. You will have to run *fsck* again with a proper *lost+found* directory in order to reconnect this directory. This error will also result in the "UNREF" error condition in Phase 4.

SORRY. NO SPACE IN *lost+found* DIRECTORY

There is no space to add another entry to the *lost+found* directory. You have just made a request to reconnect a directory in *lost+found*. Your request will be ignored and the directory will remain unconnected. You should either remove unnecessary entries from the *lost+found* directory or make *lost+found* larger by creating empty files and removing them (see the section on *lost+found*). When this has been done, you will have to run *fsck* again in order to reconnect this directory. This error will also result in the "UNREF" error condition in Phase 4.

DIR I=<inode number> CONNECTED. PARENT WAS I=<inode number>

This message advises you that a directory inode was successfully connected to the *lost+found* directory. Under the reconnected directory, the parent directory (".") has been linked to the *lost+found* directory. The second inode number given is the inode number of the former parent directory of this directory.

ERROR MESSAGES - Phase 4: Check Reference Counts

During Phase 4, *fsck* checks the link count table which was created during Phase 2 and Phase 3 and identifies incorrect link counts for files, directories and special files. It also locates unreferenced files.

**UNREF FILE I=<inode number> OWNER=<user id> MODE=<mode bits>
SIZE=<size of bytes> MTIME=<last modification time>
(RECONNECT ?)**

The file with this inode number exists but is not connected to a directory entry (it is orphaned). If you respond YES, the inode will be connected to the file system in the *lost+found* directory. If you respond NO, error will be ignored. This will result in the "CLEAR" error condition below.

SORRY. NO *lost+found* DIRECTORY

There is no *lost+found* directory in the root directory of the file system or the permission on the *lost+found* directory do not allow reading and writing. You have just made a request to reconnect a file in *lost+found*. Your request will be ignored

and the file will remain unconnected. You will have to run *fsck* again with a proper *lost+found* directory in order to reconnect this file. This error will also result in the "CLEAR" error condition below.

SORRY. NO SPACE IN *lost+found* DIRECTORY

There is no space to add another entry to the *lost+found* directory. You have just made a request to reconnect a file in *lost+found*. Your request will be ignored and the file will remain unconnected. You should either remove unnecessary entries from the *lost+found* directory or make *lost+found* larger by creating empty files and removing them (see the section on *lost+found*). When this has been done, you will have to run *fsck* again in order to reconnect this file. This error will also result in the "CLEAR" error condition below.

(CLEAR ?)

Your request to reconnect this inode (which just failed) could not be completed. If you respond YES, the inode will deallocated and added to the free inode table. If you wish to make another attempt with *fsck* to save the inode, you should respond NO, and the error condition will be ignored.

```
LINK COUNT FILE I=<inode number> OWNER=<user id> MODE=<mode bits>
      SIZE=<size in bytes> MTIME=<last modification time>
      COUNT=<link count> SHOULD BE <link count> (ADJUST ?)
```

This file inode has an incorrect count of the number of links to it. A YES answer will modify the link count with the correct value. A NO answer will ignore the error.

```
LINK COUNT DIR I=<inode number> OWNER=<user id> MODE=<mode bits>
      SIZE=<size in bytes> MTIME=<last modification time>
      COUNT=<link count> SHOULD BE <link count> (ADJUST ?)
```

This directory inode has an incorrect count of the number of links to it. A YES answer will modify the link count with the correct value. A NO answer will ignore the error.

```
LINK COUNT <file name> I=<inode number> OWNER=<user id>
MODE=<mode bits> SIZE=<size in bytes> MTIME=<last modification time>
      COUNT=<link count> SHOULD BE <link count> (ADJUST ?)
```

This file has an incorrect count of the number of links to it. A YES answer will modify the link count with the correct value. A NO answer will ignore the error.

```
UNREF FILE I=<inode number> OWNER=<user id> MODE=<mode bits>
SIZE=<size in bytes> MTIME=<last modification time> (CLEAR ?)
```

A file was found with this inode number which is not connected

with the file system and could not be reconnected. A YES answer will result in the inode being deallocated and the blocks allocated to that inode being released. A NO response will cause the error to be ignored.

```
UNREF DIR I=<inode number> OWNER=<user id> MODE=<mode bits>
SIZE=<size in bytes> MTIME=<last modification time>          (CLEAR ?)
```

A directory was found with this inode number which is not connected with the file system and could not be reconnected. A YES answer will result in the inode being deallocated and the blocks allocated to that inode being released. A NO response will cause the error to be ignored.

```
BAD/DUP FILE I=<inode number> OWNER=<user id> MODE=<mode bits>
                SIZE=<size in bytes> MTIME=<last modification time>
                (CLEAR ?)
```

Duplicate or bad blocks were found in this inode during Phase 1 or Phase 1B. If a file contains bad blocks, you may wish to try to read and salvage the file before removing it. If a file contains blocks also claimed by another inode, you may want to try and do the same. If two inodes claim the same block, usually only one of them is corrupted. This is usually the inode with the oldest date of modification. You should be selective in removing inodes under this error condition. A YES answer will deallocate the inode. A NO answer will not.

```
BAD/DUP DIR I=<inode number> OWNER=<user id> MODE=<mode bits>
                SIZE=<size in bytes> MTIME=<last modification time>
                (CLEAR ?)
```

Duplicate or bad blocks were found in this directory during Phase 1 or Phase 1B. If a directory contains bad blocks, you may wish to try to read and salvage the directory before removing it. If a directory contains blocks also claimed by another inode, you may want to try and do the same. If two inodes claim the same block, usually only one of them is corrupted. This is usually the inode with the oldest date of modification. You should be selective in removing inodes under this error condition. A YES answer will deallocate the inode. A NO answer will not.

```
FREE INODE COUNT WRONG IN SUPERBLK (FIX ?)
```

The actual count of free inodes in the file system does not match the count in the superblock. This is often because you have just removed some inodes and the free count has been changed. If you reply YES, the count in the superblock will be replaced with the correct value. A NO answer will cause the error to be ignored.

ERROR MESSAGES - Phase 5: Check Free List

In this phase, *fsck* checks the free block list to ensure that all of the block numbers are within range, there are no duplicate blocks in the list, and that the count of free blocks is correct.

EXCESSIVE BAD BLKS IN FREE LIST (CONTINUE ?)

There are more than 10 blocks in the free list with a block number which is less than the maximum block number for this file system or more than the maximum block number. A YES response will cause *fsck* to ignore the rest of the free block list and continue. This will also result in the *Bad Blocks in Free List* error below. A NO response will terminate *fsck*.

EXCESSIVE DUP BLKS IN FREE LIST (CONTINUE ?)

There are more than 10 blocks in the free list which are claimed by an inode or are duplicated within the list. A YES response will cause *fsck* to ignore the rest of the free block list and continue. This will also result in the *Dup Blocks in Free List* error below. A NO response will terminate *fsck*.

BAD FREEBLK COUNT

The count of free blocks in the free block list in the superblock is greater than 50 or less than 0. There is no operator response for this condition and the condition will result in the *Bad Free List* condition below.

<count> BAD BLKS IN FREE LIST

This number of blocks were found in the free block list with block numbers which are less than the minimum block number for this file system or more than the maximum block number. There is no operator responses for this condition and the condition will result in the *Bad Free List* condition below.

<count> DUP BLKS IN FREE LIST

This number of blocks were found in the free block list which are claimed by an inode or are duplicated within the list. There is no operator response for this condition and the condition will result in the *Bad Free List* condition below.

<count> BLK(S) MISSING

This number of blocks which are unused by the file system were not found in the free block list. There is no operator response for this condition and the condition will result in the *Bad Free List* condition below.

FREE BLK COUNT WRONG IN SUPERBLOCK (FIX ?)

The actual count of free blocks in this file system does not match the count of free blocks in the superblock of the file system. A YES answer will replace the old value with the correct number. A NO response will cause the error to be ignored.

BAD FREE LIST (SALVAGE ?)

During this phase, bad or duplicate blocks were found in the free block list or blocks were found to be missing from the file system. A YES answer will cause the free block list in file system to be replaced with the correct list. The new free block list will also be ordered to reduce the time spend waiting for the disk rotate into position. A NO response will cause this error to be ignored.

ERROR MESSAGES - Phase 6: Salvage Free List

In this phase, the free block list is reconstructed. The only error message from this phase is caused by bad values for "blocks to skip" and "blocks per cylinder" variables.

Default free-block list spacing assumed

This message advises you that the "blocks to skip" value is greater that the "blocks per cylinder", either value is less than one or the number of "blocks per cylinder" is greater than 500. The default values of 9 "blocks to skip" and 400 "blocks per cylinder" will be used.

fsck ERROR MESSAGES - Cleanup

Once the file system has been checked by *fsck*, a few cleanup functions are performed. The following messages may be displayed.

```
<count> files <count> blocks <count> free
```

This message indicates the the file system which was just checked has the given number of files and blocks. The number of free blocks is also shown.

```
***** BOOT UNIX (NO SYNC!) *****
```

This message indicates that the root file system or a mounted file system was modified by *fsck*. If UNIX is not rebooted immediately, the work which *fsck* has done may be undone by UNIX (see the earlier section on "Using *fsck*"). After this message has been displayed, you must reboot UNIX. This is usually accomplished by resetting the computer.

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

This message indicates that the file system was modified by *fsck*. If this file system is currently mounted or if it is the root file system UNIX must be rebooted immediately or the work which *fsck* has done may be undone by UNIX (see the earlier section on "Using *fsck*"). This is usually accomplished by resetting the computer.

Software Limitations in UNIX S/M/R 1.5

In addition to the known problems listed in the S/M/R 1.5 Release Notice and in the Spring SSC Newsletter, the following problems have been found to exist in the S/M/R 1.5 Release of Sys5:

Utilities:

1. Uucico is affected by the longjmp bug (see below) in an adverse way. Order patch F from the Software Support Center to overcome this problem.

C library:

1. **longjmp()** does not guarantee a non-zero return value to **setjmp(3C)** as stated in the manual. If **longjmp()** is called with a second argument of 0, or with no second argument, **setjmp** will behave incorrectly. The workaround is to guarantee that **longjmp** return a non-zero value to **setjmp** by passing a non-zero value as the second argument to **longjmp**.
2. The **plock(2)** system call always returns -1 and sets **errno** to **EINVAL**. This is due to a problem in the kernel system call interface. There is a simple workaround. Contact the Software Support Center for details.

Warnings:

1. The **access(2)** system call determines file accessibility with regard to the caller's REAL user id. Very often, this is not what the caller had in mind. To determine accessibility relative to the callers EFFECTIVE user or group id, use the **stat(2)** system call.
2. Programs should not depend on **creat()** when a lockfile scheme is to be implemented, as the **creat()** system call will ALWAYS succeed when the caller has a user id of 0 (root) -- use

`open(filename, O_WRONLY|O_CREAT|O_EXCL)` instead.

Correction to S/M/R1.5 Release Notice

The "Plexus Sys5 UNIX S/M/R1.5 Release Notice" incorrectly implies that a system can be upgraded from Sys3 to Sys5 simply by loading the new operating system software over the existing Sys3 operating system. This should not be done. Instead, the user should follow the procedures as documented in Chapter 3 "Loading S/M/R1.5 Onto A New System". The correct steps to follow are:

1. Backup all system software.
2. Backup the Sys3 root file system using the dump routine.
3. Copy special files to tape or onto another file partition. The files that should be saved are documented in Chapter 5 on pages 5-1 and 5-2. Read Chapter 5 carefully to see which files are specific to Sys5 and not Sys3. Also save those files specific to the system (i.e. /usr/lib/uucp/L.sys or any user files).
4. Run dconfig for the correct root and swap sizes.
5. Run mkfs for the root file partition.
6. Run fsck on the root file partition.
7. Execute the restor as documented on page 3-17 to read down the minimal root file system.
8. As documented on page 3-18, boot from disk and do the srcheof 21. Then do the cpio to read down the other files needed to complete the upgrade.
9. Check /tmp/cpio.err for error messages to verify a correct upgrade.
10. Cd /tmp and rm -rf release as documented.
11. Run fsck on the new root file partition to verify that the software has been correctly installed.
12. Skip to chapter 5 of the release notice. Do the tape srcheof 23 and then the cpio step to read down the files that need to be manually integrated. Copy in the special files that you have previously saved on tape/disk to a temporary directory. Run a diff on the two versions of the files as documented on page 5-3. Read Chapter 5 carefully to determine what modifications need to be made to the files.
13. Run a shutdown and then reboot the system. You have now completed the upgrade to 1.5

Third Party Software Limitations

The following sections contain information about limitations/bugs found in third party software.

EXCELAN TCP/IP

1. **Issue:** The `close` system call, when applied to socket file descriptors, returns immediately, with the socket in the process of being closed. A `socket` call requesting the same address while the `close` is in progress will fail, often causing the EXOS board to hang.

Workaround: Active (client) connections need never specify an IP port number for a socket. Prior to calling `socket`, set the port number to 0, to indicate that the system should assign a port address from the group of available ports. The socket can then be used to connect to any address on the server machine. To avoid conflicts where the server and client are on the same machine, always have the server `accept` connections on a port number less than the value of `IPPORT_RESERVED`.

PHILON FAST BASIC

1. Unable to use ADDS-3+ terminal types with this product.
2. If LPRINT command creates a line printer file, a subsequent 'rm' command to clear its contents will also remove the ability to create another file the next time the LPRINT command is executed.

UNIFY

1. ALL UNIFY EXECUTABLES: All Unify executables must be started with standard input, standard output, and standard error already open! Starting a Unify executable with any of these files not open may result in catastrophic corruption of the database or data dictionary. When starting programs using Unify HLI features from `/etc/inittab` or `crontab` files, always redirect standard input, standard output and standard error to ensure that they really are open!
2. DBLOAD: Long path names on command line (e.g. 5-directory path) may result in a core dump.
3. DBLOAD: In version 3.2-850727.1, (check the first line of your UNIFY executable, this is usually `/usr/local/unify`), default mode does not update existing records when a duplicate key is found in the data file. This has been fixed in subsequent releases.
4. DBLOAD: In version 3.2-850727.1, AMOUNT fields which are part of a COMB will not be inserted properly in all cases. This has (presumably) been fixed.
5. AFA/DBLOAD: Key fields with a specified 'unique' afa value that are also indexed with a no-dups btree will receive the appropriate default value, but will not be entered into the hash table correctly (so direct access to them is not possible). There is no workaround. This has been fixed in the upcoming release.

6. RECONF: If there is not enough space on the default file system for the reconfigure to complete, major corruption of the database may result (depending on what phase of reconstruction the program is in at the time of the failure). ALWAYS make a BUDB backup of the database before attempting to reconfigure.
7. HLI Functions: Sys5 output buffering (see release notice).
8. HLI Functions: Attempting to make a record current by calling `addrec` with an existing key will succeed but will result in btree problems on that record. Always test using `acckey` to determine if a record with that key already exists, then call `addrec` only if appropriate. This has been fixed in a later release.
9. HLI Functions: Numeric screen fields of length 1-4 not associated with a database field will be output incorrectly by `ptube` (and consequently, `outbuf()`). The solution is to use a LONG variable, or to create a dummy database field to associate this screen field with. Thus, all NUMERIC screen fields not associated with the database are of type LONG, regardless of the display length of the field. This is only of interest if you call `outbuf` or `ptube` to a NUMERIC 1-4 screen field not associated with database field.
10. HLI Executables: If you neglect to include a `main()` in a C executable, and you don't include the source for the `menucall` table, (page 1-21 in the Programmers' Manual), the program will fail with no error messages, and in certain conditions will just print `FOO` on the screen.
11. AFA: AFA processing is written to the current directory (not the DBPATH directory). The system administrator should always be in the DBPATH directory when modifying AFAs.
12. SCHENT: Under certain circumstances, it is impossible to lower the expected number of records for a given record type (of course, never less than the actual number present in the database). This value represents the "high-water" mark achieved some time in the past for this record type. The solution is to dump the data for that record out of the database using SQL, delete the record, reconfigure the database, add the record exactly as the old one except with the lower expected record number, reconfigure again, and then load the data back in with `DBLOAD` or SQL. This might be easier and quicker than the procedure on page 2-18 of the Reference Manual if you have a lot of records. The upcoming release of Unify will make this easier by permitting the administrator to dump the data out in an internal binary form (which allows it to be reloaded much more reliably and quickly).

QUADRATRON

Q-Office

1. Install script causes problems if specific user-defined files are not saved prior to installation. Requires redefinition and/or recustomizing. Specific requirements for files to be resident in both root directory and each user's home directory.
2. Install script causes problems with `crontab` (QDCRON); displays error messages to root.
3. Backspace key does not work per `termcap` entry for QVT102 terminal.
4. In Q-Form cannot enter new data on form after defining database and form. Message "unable to open form.f" is displayed. If `form.f` is copied to

/usr/qlib/qimenu, then new data can be entered.

5. In Q-Form cannot search/inquiry on form after defining database and form. If form.f is copied to /usr/qlib/qimenu, then new data can be viewed.
6. In Q-Form cannot move box, line, etc. when defining forms, though this capability is described in the documentation.

Q-One

1. Install script causes problems if specific user-defined files are not saved prior to installation. Though documentation spells this out fairly well, there is really no need for this if installation script were written differently.
2. Backspace key does not work per termcap entry for QT102 terminal.
3. On parallel printers set with multi-pass on, printing a Q- One document, then changing the printer setting to off, does not release the mode. All printing (from Q-One to UNIX) remains in multi-pass mode.
4. Multiple edits of headers and/or footers within the same edit session causes terminal hangs and file corruption.
5. Documentation procedures for headers and footers is incorrect.
6. Conversion programs for files from Fortune to Word do not work accurately for files with multiple format lines, multiple columns, and/or special character attributes (boldface, underscore, double underscore, overstrike, or a combination of these).
7. When Q-One exits, the high intensity is left on with tvi970 terminals.
8. Q-One files converted to "external" files still contain quadratron control characters.
9. Key repeats and other means of terminal input overflow cause immediate exits to Q-One menu from the editor. Screens also seem to hang.
10. Use of tabbed indentation print incorrectly, especially when multiple lines are tabbed, causing tabs and lines to be printed in the left margin.
11. When a print job is killed in the middle of printing a document, the printer queue "jams" so that the status of waiting documents cannot be changed.

GREEN HILLS C COMPILER

The following is a list of things to watch out for when using the Green Hills C Compiler.

1. Order of evaluation of an assignment is not specified by the C language and cannot be relied upon to be performed in a given order. Previous C compilers evaluated the right-hand side of the expression first, while Green Hills C evaluates the left-hand side. This is only significant when the left hand side of the assignment has side effects which will affect the right hand side. Example:

```
*ptr++ = toupper(*ptr)
```

Assuming that ptr is (char *), this will work in earlier versions of the standard sys5/sys3 C compiler, but will not work under Green Hills in the same fashion. Assignments of this form should not be used in ANY C compiler, as the side effects of such operations are implementation dependent.

2. Certain arguments to the compiler front end (cc) are not accepted. Argument processing by the Green Hills C version of cc conforms to that found under sys3. Specifically, arguments of the form -Wx,xx (valid only under sys5) will not be processed correctly by the Green Hills C version of cc. This may require extensive changes to your make files.
3. The dash character '-' in an argument name confuses the Green Hills C version of cc. File or directory names which are passed to this version of cc must not contain the '-' character.
4. Compiling with the -g option may not be totally effective unless the -X18 option is used as well. The -X18 option prevents Green Hills C from performing register optimizations (and thus hiding local variables from the programmer).
5. Type and line number information for modules compiled with the -g option may be incorrect when dealing with local variables which are in registers. This results in local variables allocated to registers to be hidden when debugging with a source level debugger. Green Hills is fixing this serious problem.
6. Deep recursive functions, functions that make large numbers of function calls, and functions which pass very large arguments to function calls may cause memory faults. This is due to an optimization which the Green Hills C Compiler performs. The optimization causes the stack to grow (and never shrink) during the life of a block (from opening brace to closing brace). Exit from the block causes the stack to be adjusted. This optimization may be defeated with little or no performance impact by compiling with the -X23 option.
7. Users of systems with the 68881 floating point coprocessor should contact the Software Support Center to obtain versions of the C and Math libraries compiled with the -81 option (floating point support) to Green Hills C.
8. You cannot preprocess non-C code as you could with the old C compiler preprocessor.
9. SCCS "what" strings are removed by the built-in optimizer if they are not used by the program.
10. The option -X36 does not work well. Avoid using this option.
11. #include statements cannot have comments which extend beyond one line, otherwise compilation errors will result.
12. The -O and -I options must be passed to the compiler separately. The syntax -OI<path> which worked with the old C compiler is not allowed with the Green Hills compiler.
13. Non-optimal code may be generated in two cases. There may be branches generated within switch statements, and there may be extraneous code generated at function entry which loads unused arguments into registers. These problems will be solved in the next release of this product.

14. Assembler output files (.s files) are kept in the working directory rather than in a temporary directory. If you have any existing .s files you wish to keep, you must rename them.

R:Office and R:Word

1. Version 3.0.3 as installed does not permit users to transfer to the shell by typing `sh`. The fix is to extract the file `/RWORD/rmenu` from the release tape and copy it to the `/RWORD` directory on your system. Use the command:

```
cpio -idumBv RWORD/rmenu </dev/tapedrive
```

Software Patches and Updates

This column is designed to notify users of new updates and bug fixes to software as they become available.

CDB Update

A new version of the C debugger is available. Version 6.1 of CDB can now be purchased from the support center. As update support is not provided for this product, existing customers with a CDB license must pay a nominal upgrade fee.

TCP/IP Update

A new release of the Excelan TCP/IP network software is now in production. This version is labeled 3.2Q, and includes numerous bug fixes to all aspects of the network software. It also includes enhancements to permit up to 16 network logins (via telnet or rlogin) onto each system on the network. The previous limit was eight network logins. All network application programs should be completely recompiled under this version for better reliability and speed.

NOS

A Software Update is currently available to Plexus Sys5 NOS users. This update prevents system crashes with the message "smfree: mem already free" and includes improvements in Sys5 NOS interprocess communications.

Hayes Modem Dialer Program

A new version of `/usr/plx/dialh`, the Hayes modem dialer program is now available on the Plexus Bulletin Board. This new version `fhayes.c` no longer dumps core and provides the capability to dial 2400 baud Hayes modems.

ACP Patch "C"

The patch "C" version of the ACP download modules improves ACP reliability. It includes corrections to the DMA handling, improves the "flush" capability, and provides additional diagnostic data. Currently, patch "C" is only functional on level S1.5, 68020 based systems. 68000 compatability has been added.

Sys5 S1.5 Patch "E"

A new version of the file "copy.s" is provided in this patch. This patch resolves a problem causing core dumps to occur randomly in some I/O intensive programs on 68020 machines. This patch is available on the bulletin board as the file `/usr/spool/uucppublic/4300/1.5/e.z` in packed cpio format. Patch "S1.5E" is also provided on the "S1.5" patch tape.

Sys5 S/M1.5 Patch "F"

A new version of the file `/usr/lib/uucp/uucico` is provided in this patch. This patch is available on the bulletin board as the file `/usr/spool/uucppublic/4300/1.5/f.z` in packed cpio format. Patch "S1.5F" is available on both the "S1.5 patch tape" and the "M1.5 Patch tape."

NOTE: The S1.5 patch tape currently contains Patches E and F while the M1.5 patch tape contains Patch F. The naming of these patches are arbitrary. For example, there is no S/M/R 1.5 patch "A", "B" or "D".

If you are experiencing any of the symptoms mentioned above, or feel you could benefit from any of these patches, please contact the Plexus Software Support Center for details.

PLEXUS COMPUTERS INC.

Current Software Releases

as of 7/31/87

OPERATING SYSTEMS					
Part #	# of Users	Model	Systems	Software Product	Current Release
Sys3					
67-00294	16	4100AC	P/25	Sys3-Z8000-cartridge	1.2
67-00343	16	4100AC	P/25	Sys3 Update	1.21
67-00295	16	4100AT	P/40	Sys3-Z8000-9-track	1.2
67-00344	16	4100AT	P/40	Sys3 Update	1.21
67-00296	32	4100BT	P/40	Sys3-Z8000-9-track	1.2
67-00345	32	4100BT	P/40	Sys3 Update	1.21
67-00297	64	4100CT	P/40	Sys3-Z8000-9track	1.2
67-00346	32	4100BT	P/40	Sys3 Update	1.21
67-00327	16	4200AC	P/35	Sys3-68000-cartridge	3.3
67-00353	16	4200AC	P/35	Sys3 Update	3.31
67-00328	16	4200AT	P/60	Sys3-68000-9-track	3.3
67-00354	16	4200AT	P/60	Sys3 Update	3.31
67-00329	32	4200BT	P/60	Sys3-68000-9-track	3.3
67-00355	32	4200BT	P/60	Sys3 Update	3.31
67-00330	64	4200CT	P/60	Sys3-68000-9-track	3.3
67-00356	64	4200CT	P/60	Sys3 Update	3.31
Sys5					
Domestic					
67-00748	80	4300AT	P/60	Sys5-68000-9-track	1.5
67-00749	80	4300AC	P/35	cartridge	
67-00750-01	32	4300GC	P/35	20mb cartridge vol. 1	
67-00750-02	32	4300GC	P/35	20mb cartridge vol. 2	
67-00754	16	4300DC	P/15-20	Sys5-68010-cartridge	1.5
67-00756	80	4302AT	P/60-75	Sys5-68020-9-track	1.5
67-00757	80	4302AC	P/35-55-75	cartridge	
67-00758-01	32	4302GC	P/35	20mb cartridge vol. 1	
67-00758-02	32	4302GC	P/35	20mb cartridge vol. 2	
67-00840	128	4302CT	P/95	Sys5-68020 - 9-track	1.6
67-00846	128	4302CC	P/95	cartridge	1.6
Sys5					
International					
67-00751	80	4300IT	P/60	Sys5-68000 - 9-track	1.5
67-00752	80	4300IC	P/35	cartridge	
67-00753-01	80	4300IG	P/35	20mb cartridge vol. 1	
67-00753-02	80	4300IG	P/35	20mb cartridge vol. 2	
67-00755	16	4300IF	P/15-20	Sys5-68010-cartridge	1.5

67-00759	80	4302IT	P/60-75	Sys5-68020 - 9-track	1.5
67-00760	80	4302IC	P/35-55-75	cartridge	
67-00761-01	80	4302IG	P/35	20mb cartridge vol. 1	
67-00761-02	80	4302IG	P/35	20mb cartridge vol. 2	
67-00841	128	4302IT	P/95	Sys5-68020 - 9-track	1.6
67-00847	128	4302IC	P/95	cartridge	1.6
NOS					
67-00543		4310AC	P/35-55-75	Sys5 - NOS - cartridge	1.4
67-00544		4310AT	P/60-75	- 9-track	
67-00545		4310DC	P/20	Sys5 - NOS - cartridge	1.4

PLEXUS COMPUTERS INC.
Current Software Releases
as of 6/30/87

COMMUNICATIONS				
Part #	Model	Systems	Software Product	Current Release
RJE/HASP				
67-00319	4211AC	P/35	68000-Sys3-cartridge	3.1
67-00320	4211AT	P/60	9-track	3.1
67-00599	4311AT	P/60-75	68000/20-Sys5 - 9-track	4.0
67-00600	4311AC	P/35-55-75	cartridge	4.0
BATCH 2780/3780				
67-00333	4231AC	P/35	68000-Sys3-cartridge	3.2
67-00334	4231AT	P/60	9-track	3.2
67-00623	4331AC	P/35-55-75	68000/20-Sys5-cartridge	4.0
67-00624	4331AT	P/60-75	9-track	4.0
TCP/IP				
67-00625	4371AT	P/60-75	68000/20-Sys5 - 9-track	3.2
67-00626	4371AC	P/35-55-75		
67-00833	4371DC	P/20	68010-Sys5-cartridge	3.2
67-00824	4371DC	P/95	68020-Sys5 - 9-track	3.2
67-00825	4371CC	P/95	cartridge	3.2
Multiplex				
67-00699	4374DF	P/15-20	68010-IBM PC floppy disk	1.1
67-00730	4374AC	P/35-55-75	68000/20-Sys5-cartridge	1.1
67-00731	4374AT	P/60-75	9-track	1.1
67-00732	4374DC	P/15-20	68010-Sys5-cartridge	1.1
3270 SNA				
67-00715	4378AT		mSNA/3270 - 9 track	2.0
67-00716	4378AC		cartridge	2.0

PLEXUS COMPUTERS INC.
Current Software Releases
as of 7/31/87

APPLICATIONS				
Part #	Model	Systems	Software Product	Current Release
UNIFY				
67-00250	4218AC	P/35	68000-Sys3-cartridge	3.1b
67-00251	4219AC	P/35	(Runtime) cartridge	3.1b
67-00252	4218AT	P/60	9-track	3.1b
67-00253	4219AT	P/60	(Runtime) 9-track	3.1b
67-00504	4318AC	P/35-55-75	68000/20-Sys5-cartridge	3.2
67-00505	4318AT	P/60-75	9-track	3.2
67-00507	4319AC	P/35-55-75	(Runtime)-cartridge	3.2
67-00508	4319AT	P/60-75	(Runtime) 9-track	3.2
67-00506	4318DC	P/15-20	68010-Sys5-cartridge	3.2
67-00509	4319DC		(Runtime) 9-track	3.2
67-00702	P/35-55-75		Unify-RM/Cobol Interface-cartridge	
67-00703	P/60-75		-9-track	
67-00704	P/15-20		-cartridge	
20/20				
68-00423	4320DC/AC	P/20-35-55-75	68000/10/20-Sys5-cartridge kit	1.13
68-00424	4320AT	P/60-75	9-track kit	1.13
Q-Office				
67-00564	4222AC	P/35	68000-Sys3-cartridge	1.94
67-00565	4222AT	P/60	9-track	1.94
67-00561	4322AC	P/35-55-75	68000/20-Sys5-cartridge	1.94
67-00562	4322AT	P/60-75	9-track	1.94
67-00563	4322DC	P/15-20	68010-Sys5-cartridge	1.94
Q-One				
67-00659	4221AC	P/35	68000-Sys3-cartridge	1.94
67-00660	4221AT	P/60	9-track	1.94
67-00656	4321AC	P/35-55-75	68000/20-Sys5-cartridge	1.94
67-00657	4321AT	P/60-75	9-track	1.94
67-00658	4321DC	P/15-20	68010-Sys5-cartridge	1.94
R:Office				
67-00818	4332AC	P/35	68000/20-Sys5-cartridge	3.0.3
67-00819	4332AT	P/60	9-track	3.0.3
67-00820	4332DC	P/15-20	68010-Sys5-cartridge	3.0.3
R:Word				
67-00821	4333AC	P/35	68000/20-Sys5-cartridge	3.0.3
67-00822	4333AT	P/60	9-track	3.0.3

67-00823	4333DC	P/15-20	68010-Sys5-cartridge	3.0.3
R:Desk				
67-00475	4234AC	P/35	68000-Sys3-cartridge	3.0.2
67-00476	4234AT	P/60	9-track	3.0.2
67-00469	4334AC	P/35	68000-Sys5-cartridge	3.0.2
67-00470	4334AT	P/60	9-track	3.0.2
67-00453	4334DC	P/15-20	68010-Sys5-cartridge	3.0.2
67-00575	4344AC	P/35-55-75	68020-Sys5-cartridge	3.0.2
67-00576	4344AT	P/60-75	9-track	3.0.2
INFORMIX SQL				
67-00772	4345AC	P/35	SQL-Sys5-cartridge	2.00
67-00803	4345BC	P/55	"	2.00
67-00804	4345CC	P/75	"	2.00
67-00773	4345BT	P/60	9-track	2.00
67-00805	4345CT	P/75	"	2.00
67-00774	4345DC	P/15-20	cartridge	2.00
INFORMIX 4GL				
67-00775	4346AC	P/35	4GL-Sys5-cartridge	1.00
67-00806	4346BC	P/55	"	1.00
67-00807	4346CC	P/75	"	1.00
67-00776	4346BT	P/60	9-track	1.00
67-00808	4346CT	P/75	"	1.00
67-00777	4346DC	P/15-20	cartridge	1.00
INFORMIX ESQL/C				
67-00778	4347AC	P/35	ESQL/C-Sys5-cartridge	2.00
67-00809	4347BC	P/55	"	2.00
67-00810	4347CC	P/75	"	2.00
67-00779	4347BT	P/60	9-track	2.00
67-00811	4347CT	P/75	"	2.00
67-00780	4347DC	P/15-20	cartridge	2.00
INFORMIX File-It!				
67-00781	4349AC	P/35	File-It!-Sys5-cartridge	3.00
67-00812	4349BC	P/55	"	3.00
67-00813	4349CC	P/75	"	3.00
67-00782	4349BT	P/60	9-track	3.00
67-00814	4349CT	P/75	"	3.00
67-00783	4349DC	P/15-20	cartridge	3.00
INFORMIX C-ISAM				
67-00784	4350AC	P/35	C-ISAM-Sys5-cartridge	3.00
67-00815	4350BC	P/55	"	3.00
67-00816	4350CC	P/75	"	3.00
67-00785	4350BT	P/60	9-track	3.00
67-00817	4350CT	P/75	"	3.00
67-00786	4350DC	P/15-20	cartridge	3.00

PLEXUS COMPUTERS INC.
Current Software Releases
as of 6/30/87

LANGUAGES				
Part #	Model	Systems	Software Product	Current Release
RM/COBOL				
67-00487	4104AC	P/25	Z8000-Sys3-cartridge	2.1A
67-00488	4104AT	P/40	9-track	2.1A
67-00489	4105AC	P/25	(Runtime) cartridge	2.1A
67-00490	4105AT	P/40	(Runtime) 9-track	2.1A
67-00483	4204AC	P/35	68000-Sys3-cartridge	2.1A
67-00484	4204AT	P/60	9-track	2.1A
67-00485	4205AC	P/35	(Runtime) cartridge	2.1A
67-00486	4205AT	P/60	(Runtime) 9-track	2.1A
67-00477	4304AC	P/35-55-75	68000/20-Sys5-cartridge	2.1A
67-00478	4304AT	P/60-75	9-track	2.1A
67-00480	4305AC	P/35-55-75	(Runtime) cartridge	2.1A
67-00481	4305AT	P/60-75	(Runtime) 9-track	2.1A
67-00479	4304DC	P/15-20	68010-Sys5-cartridge	2.1A
67-00482	4305DC	P/15-20	(Runtime) cartridge	2.1A
PHILON COBOL				
67-00501	4336AC	P/35-55-75	68000/20-Sys5-cartridge	1.32
67-00502	4336AT	P/60-75	9-track	1.32
67-00503	4336DC	P/15-20	68010-Sys5-cartridge	1.32
PHILON FAST/BASIC-M				
67-00794	4337AC	P/35-55-75	68000/20-Sys5-cartridge	1.35
67-00795	4337AT	P/60-75	9-track	1.35
67-00796	4337DC	P/15-20	68010-Sys5-cartridge	1.35
SMC BASIC				
67-00419	4209AC	P/35	68000-Sys3-cartridge	6.5.8
67-00420	4209AT	P/60	9-track	6.5.8
67-00707	4309AC	P/35-55-75	68000/20-Sys5-cartridge	6.5.10
67-00708	4309AT	P/60-75	9-track	6.5.10
67-00709	4309DC	P/15-20	68010-Sys5-cartridge	6.5.10
CDB C-Debugger				
67-00325	4223AC	P/35	68000-Sys3-cartridge	1.0
67-00326	4223AT	P/60	9-track	1.0
67-00826	4323AC	P/35-55-75	68000/20-Sys5-cartridge	6.1
67-00827	4323AT	P/60-75	9-track	6.1

67-00828	4323DC	P/15-20	68010-Sys5-cartridge	6.1
SVS PASCAL				
67-00400	4215AC	P/35	68000-Sys3-cartridge	2.4
67-00401	4215AT	P/60	9-track	2.4
67-00837	4315AC	P/35-55-75	68000/20-Sys5-cartridge	2.6
67-00838	4315AT	P/60-75	9-track	2.6
67-00839	4315DC	P/15-20	68010-Sys5-cartridge	2.6
SVS FORTRAN				
67-00425	4225AC	P/35	68000-Sys3-cartridge	2.4
67-00426	4225AT	P/60	9-track	2.4
67-00834	4325AC	P/35-55-75	68000-Sys5-cartridge	2.6
67-00835	4325AT	P/60-75	9-track	2.6
67-00836	4325DC	P/15-20	68010-Sys5-cartridge	2.6
67-00789	4326AC	P/35-55-75	68020-Sys5-cartridge	2.6
67-00790	4326AT	P/60-75	9-track	2.6
67-00787	4327AC	P/35-55-75	68020 w/68881-cartridge	2.6
67-00788	4327AT	P/60-75	9-track	2.6
Greenhills C Compiler				
67-00745	4303AT	P/60-75	68000/20-Sys5-9-track	1.8.01
67-00746	4303AC	P/35-55-75	-cartridge	1.8.01
67-00747	4303DC	P/15-20	-cartridge	1.8.01

PLEXUS COMPUTERS INC.
Current Documentation Releases
 as of 7/31/87

SYSTEMS DOCUMENTATION	
Part #	Description
SYSTEMS MANUALS	
98-05075.1 - Ver. A	P/15-20 Users Manual for Sys5 - 68010
98-05014.7	P/25 Users Manual for Sys3 - Z8000
98-05043.7 - Ver. C	P/35 Users Manual for Sys3 - 68000
98-05001.6 -	P/40 Users Manual for Sys3 - Z8000
98-05041.6 - Ver. C	P/60 Users Manual for Sys3 - 68000
 OPERATING SYSTEMS	
Sys3	
98-05045.8	UNIX Programmer's Manual - Vol. 1A
98-05046.8	UNIX Programmer's Manual - Vol. 1B
98-05036.5	UNIX Programmer's Manual - Vol. 2A
98-05037.5	UNIX Programmer's Manual - Vol. 2B
98-05047.2	UNIX Programmer's Manual - Vol. 2C
98-05054.3	UNIX Programmer's Manuals - 5-volume set
98-40092.2 - Ver. A	Sys3 - 1.2 Release Notice for Z8000
98-40135.2 - Ver. A	Sys3 - 1.21 Update Notice for Z8000
98-40099.1 - Ver. B	Sys3 - 3.3 Release Notice for 68000
98-40140.1 - Ver. B	Sys3 - 3.31 Update Notice for 68000
 Sys5	
98-05082.1 - Ver. A	Sys5 Description
98-05079.0 - Ver. A	Sys5 Operator's Guide
98-05077.2 - Ver. A	Sys5 Error Guide
98-05078.1 - Ver. A	Sys5 Graphics Guide
98-05081.1 - Ver. C	Sys5 Support Tools Guide
98-05076.1 - Ver. D	Sys5 Administrator's Guide
98-05083.1 - Ver. A	Sys5 User's Guide
98-05084.1 - Ver. D	Sys5 Administrator's Reference Manual
98-05100.2 - Ver. C	Sys5 Administrator's Set (Guide & Reference)
98-05080.1 - Ver. B	Sys5 Programmer's Guide
98-05085.1 - Ver. D	Sys5 Programmer's Reference Manual
98-05086.1 - Ver. D	Sys5 User Reference Manual
98-05096.2 - Ver. B	Sys5 Documentation (14 volume set)
98-05091.1 - Ver. A	Sys5 Documenter's Workbench - Introduction
98-05092.1 - Ver. A	- Text Formatter
98-05093.1 - Ver. A	- Work Macro
98-05094.1 - Ver. A	- Work Preprocessor

98-05101.1 - Ver. A	- Complete Set
98-05103.1 - Ver. A	Sys5 UNIX Run-time Manual
98-05133.1 - Ver. A	Sys5 Administrator's Handbook
98-05133.1 - Ver. A	Sys5 Administrator's Handbook for P/95
98-40199.4 - Ver. D	Sys5 - S/M/R 1.5 Release Notice
98-40199.5 - Ver. B	Sys5- 1.6 Release Notice for P/95

NOS

98-05107.2 - Ver. D	NOS Users Manual for Sys5 (incl. Release Notice)
98-40142.2 - Ver. D	NOS Users Manual for Sys5
98-40044.1	Using Plexus NOS
98-40046.1	Installing NOS Hardware
98-40212.0 - Ver. A	NOS 1.2 Patch for P/15-20

PLEXUS COMPUTERS INC.
Current Documentation Releases
as of 7/31/87

APPLICATIONS DOCUMENTATION	
Part #	Description
Unify	
98-40077.2 - Ver. A	Unify 3.1b Release Notice (for Sys3 or Sys5)
98-40161.1 - Ver. B	Unify 3.2 Release Notice (for Sys5 only)
98-05052.3 - Ver. A	Unify 3.2 Reference Manual
98-05053.4 - Ver. A	Unify 3.2 Tutorial Manual
98-05116.1	Unify 3.2 Programmer's Manual
Informix	
98-05152.1	C-ISAM Reference Manual
98-40261.1	C-ISAM Release Notice
98-05151.1	ESQL-C Reference Manual
98-40262.1	ESQL-C Release Notice
98-05150.1	SQL Reference Manual
98-40263.1	SQL Release Notice
98-05153.1	File-It! Reference Manual
98-40264.1	File-It! Release Notice
98-05149.1	4GL 3-volume set (Reference Manuals & User Guide)
98-40265.1	4GL Release Notice
20/20	
98-05117.1	20/20 User's Manual (includes Release Notice)
Quadratron	
98-05099.2 - Ver. A	Q-Office 1.94 User's Manual
98-05099.3 - Ver. A	Q-Office 1.94 Set (incl. Q-One W/P Manual)
98-40145.2 - Ver. A	Q-Office 1.94 Release Notice
98-05061.0	Q-One Word Processing Manual
98-05064.1	Q-One Tutorial Guide
98-05065.1	Q-One Reference Guide
98-05065.3 - Ver. A	Q-One Set (both Tutorial and Reference Guide)
98-40094.3 - Ver. A	Q-One Release Notice
R Systems	
98-05108.1 - Ver. A	R:Office Operator's Manual
98-40165.3 - Ver. A	R:Office 3.0.3 Release Notice (includes R:Word)
98-05109.1 - Ver. A	R:Word Operator's Manual
98-05110.1 - Ver. A	R:Desk Operator's Manual

**LANGUAGE
SOFTWARE DOCUMENTATION**

RM/COBOL

98-05066.2 - Ver. A
98-40163.1 - Ver. A
98-40163.2 - Ver. A

RM/COBOL 2.0c Reference Manual
RM/COBOL 2.1A Release Notice (Sys3 - Z8000)
RM/COBOL 2.1A Release Notice (Sys3 & 5 - 68000/10/20)

SVS FORTRAN

98-05097.3 - Ver. A
98-40234.2

(includes use with MC68881 co-processor)
SVS FORTRAN 2.6 Reference Manual
SVS FORTRAN 2.6 for 68020 Release Notice

SVS PASCAL

98-05057.2 - Ver. A
98-40159.1
98-40158.1

SVS PASCAL 2.3 Reference Manual
SVS PASCAL 2.4 Release Notice (Sys3)
SVS PASCAL 2.4 Release Notice (Sys5)

SMC BASIC

98-05051.1
98-40102.4 - Ver. A
98-40102.5 - Ver. A

SMC BASIC Thoroughbred Manual
SMC BASIC 6.5.8 for Sys3 Release Notice
SMC BASIC 6.5.10 for Sys5 Release Notice

Philon FAST/BASIC-M

98-40225.1 - Ver. A
98-05122.1 - Ver. A

PHILON BASIC-M 1.35 Release Notice
PHILON BASIC-M Programmer's Reference Manual

Philon COBOL

98-40188.1
98-05119.1

PHILON COBOL 1.32 Release Notice
PHILON COBOL Programmer's Reference Manual

Greenhills C Compiler

98-40240.1 - Ver. C.
98-05144.1 - Ver. B.

GREENHILLS C COMPILER Release Notice
GREENHILLS C COMPILER Users Manual

CDB C-Debugger

98-40155.3

CDB 6.1 Release Notice

**COMMUNICATIONS
SOFTWARE DOCUMENTATION**

BATCH 2780/3780

98-40059.3
98-40200.1

BATCH 2780/3780 3.2 Release Notice
BATCH 2780/3780 4.0 Release Notice

RJE/HASP

98-40201.1 - Ver. B

RJE/HASP 4.0 Release Notice

TCP/IP

98-40211.3
98-05131.1

TCP/IP 3.2 Release Notice
Excelan EXOS - 8010 Reference Manual

MULTIPLEX

98-05124.2	Multiplex User's Manual
98-40244.1	Multiplex Host Installation Guide
98-40245.1	Multiplex PC Installation Guide
98-30029.1	One Page Notice

mSNA/3270

98-40223.1 - Ver. A	mSNA/3270 User's Manual
98-40226.1	mSNA/3270 2.0 Release Notice
98-40226.2	mSNA/3270 2.0 Release Notice for P/95
