DECUS U.S. CHAPTER SIGs NEWSLETTERS

July 1987 Volume 2, Number 11

DECUS

Digital Equipment Computer Users Society

DECUS, U. S. Chapter
219 Boston Post Road, (BP02)
Marlboro, MA 01752

U. S. Activities (617) 480-3259 or 3302
Finance & Admin. (617) 480-3634
Library (617) 480-3521

Dear Subscribers,

Our editors are among the hardest working groups within this great volunteer/technological giant called DECUS. Not only have they committed to a monthly task of tracking-down and editing their SIG's contributions within an implacable deadline but the fact that the product they product is a tangible publication subject to monthly public scrutinizing makes their job one of immense importance to the Society. Indeed, to some of you the newsletter represents your primary contact with DECUS; the Newsletter represents the fulfilling of the mission of the society, promotion of information exchange among its members. And this task has fortunately fallen on the shoulders of some very talented and dedicated people. People who write not only for readers like Steve Lacativa in New York but also the fine people I meet in old York, England at a recent DECUS symposium. Not only for Joel Snyder in the desert of Arizona, but also for Eric in the desert of Kuwait. Our newsletter editors (and your articles) have an impact world-wide.

Recently the Communications Committee has directed a pilot project managed by Don Stern, DATATRIEVE/4GL editor. You've seen the results featured in the gray area over the past few issues. The outstanding feature is, of course, the change to portrait mode. We've actually been able to compress the type so well that the entire contents of a landscape page (sideways) can be placed on a regular (up & down) page.

At the Nashville symposium enthusiasm was running high and a "challenge" was issued by Clyde Poole, Large Systems editor, who offered to try to duplicate the DECpage style with LaTeX. You can see his results in this issue.

What does all this mean? It means the Communications Committee is committed to providing its editors with the best possible tools to accomplish their important job. It means we are committed to providing you a quality product at a reasonable price.

But no matter how well we package our columns, no matter how well we train & equip our editors, they can't do their job unless there's something to publish. And, from where does material for the newsletter come? From YOU! Readers, subscribers, members ... YOU.

I know what you're thinking, some of you. I've got an article of my own right now sitting on the shelf behind me: "Is it good enough? Does it need more polish? Oh, maybe they don't need something on that topic, too much has been written about it already. It's such a tiny idea, It's not really in good enough shape yet. The editor won't even read it."

Wrong, wrong, wrong! We encourage your ideas and articles. I don't know of one editor here who wouldn't read your article and advise you, if necessary. Try it. It's an opportunity to contribute to the society; to plow something back in. It's an opportunity to see your name in print ... and a by-line is a thrill. It's an opportunity to participate in the real reason you're subscribing, the sharing of technical know how one person has acquired with someone else.

Of course, our thanks go out to the many fine authors who contribute on a regular basis. They are the core and the corps that this publication couldn't go without!

Keep contributing and keep subscribing. Check out the many ways to transit your article usually listed in the editor's corner. If all else fails, type and mail a hardcopy to the most appropriate SIG and fill out that re-subscription form. Your support ... that is what will make the Newsletter great!

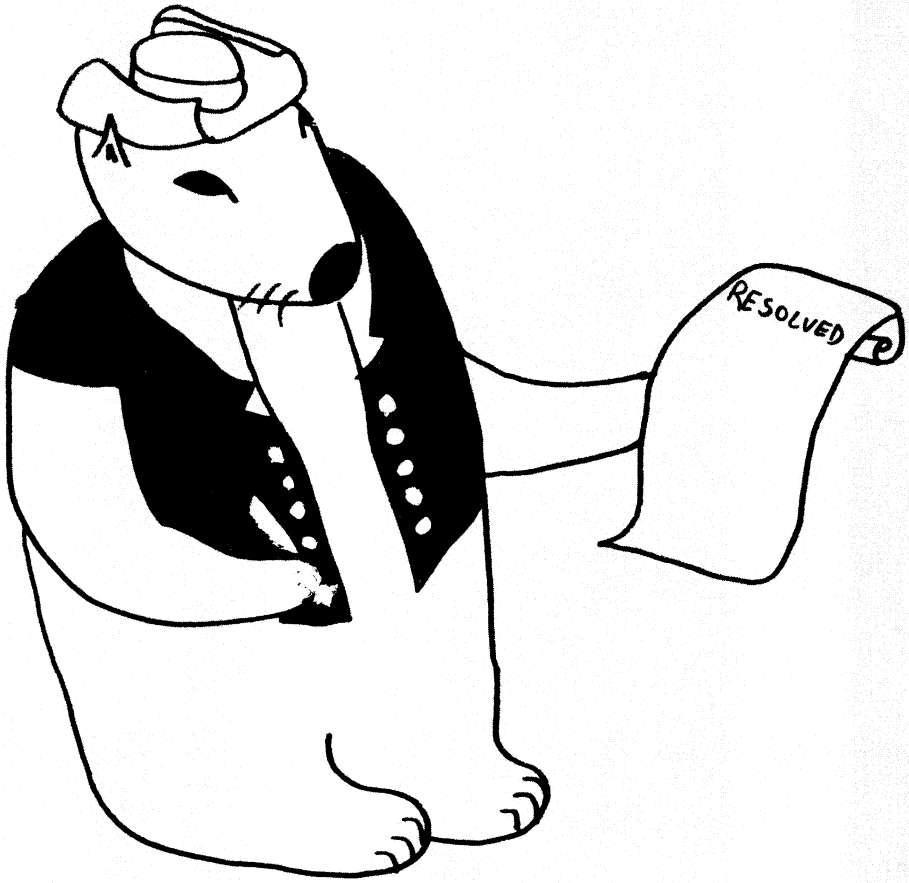
Beverly J. Welborne
Communications Committee Chair

The Wombat

EXAMINER and 4GL Dispatch

"Increases the Circulation of Anyone in America" *Volume 8* *Number 11*

DTR



Contributions

Submissions to this newsletter are constantly sought. A submission can be an article, a letter to the Wombat Wizard, a technical tip, or anything of interest to people using or considering the use of Datatrieve or any 4GL product. Submissions on magnetic media are preferred but almost any type will be considered.

Contributions for the newsletter can be sent to either of the following addresses:

Editor, DATATRIEVE Newsletter
c/o DECUS U.S. Chapter
219 Boston Post Road, BP02
Marlboro, MA 01752

Donald E. Stern, Jr
Warner Lambert Company
10 Webster Road
Milford, CT 06460

Table of Contents

DECUS U.S. Chapter SIG Newsletters, Volume 2, No. 11, July 1987
Wombat Examiner and 4GL Dispatch, Volume 8, No. 11

User Defined Function to Generate SIXEL Files

Donald E. Stern, Jr., Warner Lambert Co., Milford, CT

Introduction

In the May 1986 issue of this newsletter, I described a standalone program which could be used to generate a SIXEL graphics file using a ReGIS source file. This SIXEL graphics file could then be printed on a queued printer such as a Digital LN03 laser printer. This original program has been modified into a FUNCTION subroutine. This routine has been linked with the Datatrieve shareable image as a User Defined Function (UDF).

Description

Like its standalone cousin, the CONVERT_SCREEN_TO_SIXEL function uses the firmware of a VT240 or VT241 to perform the conversion. (Readers are directed to the May 1986 issue for a more complete description of the conversion mechanism, a discussion of the SIXEL protocol, etc.) The actual code for the function is contained in Appendix I. Basically, the original SIXEL program was modified such that the parameters normally passed to the routine on the command line are now passed in an argument list. The arguments are passed as follows:

WL_SIXEL(filespec,mode)

where filespec is a mandatory argument containing a valid RMS file name and mode is an optional argument.

To incorporate the function into the Datatrieve shareable image, one must (a.) compile the routine, (b.) incorporate it into the Datatrieve object library, and (c.) create the UDF by adding the following code fragment to DTRFND.MAR.

```
; WL_SIXEL - Convert a screen into a sixel file
;
; Output is a SIXEL File
; Input is a valid file specification
; Input is Sixel Mode - Choice of rotated, compressed or expanded


$DTR$FUN_DEF WL_SIXEL, CONVERT_SCREEN_TO_SIXEL, 2
    $DTR$FUN_OUT_ARG  TYPE = FUN$K_STATUS
    $DTR$FUN_NOVALUE
    $DTR$FUN_NOOPTIMIZE
    $DTR$FUN_IN_ARG  TYPE = FUN$K_DESC, DTYPE = DSC$K_DTYPE_T, ORDER = 1
    $DTR$FUN_IN_ARG  TYPE = FUN$K_DESC, DTYPE = DSC$K_DTYPE_T, ORDER = 2
$DTR$FUN_END_DEF
```

Usage

To use the function, one might execute a compound statement such as the following.

```
DTR> PLOT X_Y LOA, PRICE OF YACHTS THEN WL_SIXEL("YACHTS.SIX","")
```

This compound statement will (a.) plot a scatter diagram of yacht prices as a function of length on the terminal screen and (b.) produce a corresponding SIXEL file called YACHTS.SIX in the current default directory. YACHTS.SIX can then be printed on a queued printer capable of printing SIXEL graphics, such as an LN03. The graph shown in figure 1 was produced using this compound statement.

```
C             SET_TERMINAL_CHARACTERISTICS   - Internally developed routine
                                  to set terminal characteristics
C             SYS$QIOW              - Synchronous queued I/O system service
C             SYS$ASSIGN            - System service to assign a channel
                                        number to a device
C             LIB$SET_LOGICAL       - RTL routine to create a logical name
C;;           LIB$GETDVI            - RTL routine to get device information
C
C       I/O Operations Definitions
        INCLUDE       '($IODEF)'

C       Terminal I/O Operation Modifiers
        INCLUDE       '($TTDEF)'
        INCLUDE       '($TT2DEF)'

C       System Service Definitions
        INCLUDE       '($SSDEF)'

C       Device Information Definitions
        INCLUDE       '($DVIDEF)'
        INCLUDE       '($DCDEF)'

C       SYS$QIO Structures
        INTEGER*2      CHANNEL              !I/O Channel
        INTEGER       CODE,                !Type of I/O operation
        1             INPUT_BUFFER_SIZE,     !Input buffer size
        2             INPUT_SIZE            !Size of input as read
        PARAMETER      (INPUT_BUFFER_SIZE=1)

C       Terminal Characteristics Buffers
        BYTE          CLASS,
        1             TYPE
        INTEGER*2      WIDTH
        INTEGER*4      BASIC,
        1             EXTENDED
        INTEGER*4      OLD_BASIC,            !Old basic char.
        1             OLD_EXTENDED          !Old extended char.

C       QIO Status Block
        STRUCTURE      /IOSTAT_BLOCK/
             INTEGER*2 IOSTAT,       !Return Status
        1             TERM_OFFSET,   !Location of Line terminator
        2             TERMINATOR,    !Value of terminator
        3             TERM_SIZE      !Size of terminator
        END STRUCTURE
        RECORD /IOSTAT_BLOCK/ IOSB

C       Subprograms
        INTEGER*4      SYS$ASSIGN,
        1             SYS$QIOW,
        2             LIB$SET_LOGICAL,
        3             LIB$GETDVI


C       Status codes returned by this program
        INTEGER*4      STATUS               !Return status
        INTEGER*4      SIXEL_READ_ERROR       !Read error status code
```

```
      INTEGER*4       CRT_OPEN_ERROR         !Open sys$output error,
      INTEGER*4       SIXEL_OPEN_ERROR        !Open error status code
      PARAMETER       (SIXEL_READ_ERROR = 11)!Just set status code to
C                                             these values, arbitrary
      PARAMETER       (SIXEL_OPEN_ERROR = 13)
      PARAMETER       (CRT_OPEN_ERROR   = 15)


      INTEGER*4       DEVICE_CLASS           !Device class returned
                                            !by LIB$GETDVI
C     Terminal Escape Sequences
      BYTE          ENTER_REGIS(4)          !Enter Regis mode
      BYTE          EXIT_REGIS(2)           !Exit Regis
      BYTE          LOCK_KEYBOARD(4)         !Lock keyboard
      BYTE          UNLOCK_KEYBOARD(4)       !Unlock keyboard
      BYTE          GRAPHICS_TO_HOST(5)      !Graphics to host
      BYTE          GRAPHICS_TO_PRINTER(5) !Graphics to printer
      BYTE          EXPANDED_PRINT(6)        !Expanded graphics
      BYTE          ROTATED_PRINT(6)         !Rotated graphics
      BYTE          COMPRESSED_PRINT(6)      !Compressed graphics


      CHARACTER*2     EXIT_GRAPHICS          !Same as EXIT_REGIS
      CHARACTER*13    MAKE_HARDCOPY          !Graphics hardcopy
      CHARACTER*(*)   FILESPEC               !SIXEL output filespec
      CHARACTER*(*)   SIXEL_MODE             !Mode of file to print
      CHARACTER*256   BUFF                  !Data Buffer

      LOGICAL       QUIT

      EQUIVALENCE (EXIT_REGIS(1),EXIT_GRAPHICS)

      DATA ENTER_REGIS /27,80,49,112/              ! <ESC>P1p
      DATA EXIT_REGIS /27,92/                     ! <ESC>\
      DATA LOCK_KEYBOARD /27,91,50,104/             ! <ESC>[2h
      DATA UNLOCK_KEYBOARD /27,91,50,108/           ! <ESC>[2l
      DATA GRAPHICS_TO_HOST /27,91,63,50,105/       ! <ESC>[?2i
      DATA GRAPHICS_TO_PRINTER /27,91,63,48,105/     ! <ESC>[?0i
      DATA EXPANDED_PRINT /27,91,63,52,51,104/       ! <ESC>[?43h
      DATA ROTATED_PRINT /27,91,63,52,55,104/       ! <ESC>[?47h
      DATA COMPRESSED_PRINT /27,91,63,52,51,108/     ! <ESC>[?43l
      DATA MAKE_HARDCOPY /'S(H(P[00,0]))'/

C-------------------------------------------------------------------------------
C     All constants, parameters, variables are now defined.
C     We must make checks on our input and output devices to see if we
C     are capable of doing the conversion.
C-------------------------------------------------------------------------------

      STATUS = LIB$SET_LOGICAL('SIXEL$DEVICE',FILESPEC,,,)   !Assign logical
                                                            !to file spec
C     If error, then return with error status
      IF (.NOT. STATUS) THEN
            CONVERT_SCREEN_TO_SIXEL = STATUS
            RETURN
      ENDIF
```

```
C       Make sure that SYS$OUTPUT is a terminal device with a call to
C       LIB$GETDVI.  If the call fails or if SYS$OUTPUT is not a terminal then
C       return with error status.
        CODE = DVI$_DEVCLASS
        STATUS = LIB$GETDVI (CODE,,'SYS$OUTPUT',DEVICE_CLASS,,)
        IF (.NOT. STATUS) THEN                  !GETDVI failed
              CONVERT_SCREEN_TO_SIXEL = STATUS
              RETURN
        ENDIF
        IF (DEVICE_CLASS .NE. DC$_TERM) THEN    !Invalid characteristics
              CONVERT_SCREEN_TO_SIXEL = SS$_IVCHAR
              RETURN
        ENDIF



C       Assign a channel number to SYS$OUTPUT.  If the assignment fails then
C       return with the proper status code.
        STATUS = SYS$ASSIGN('SYS$OUTPUT',CHANNEL,,)
        IF (.NOT. STATUS) THEN                  !ASSIGN failed
              CONVERT_SCREEN_TO_SIXEL = STATUS
              RETURN
        ENDIF



C       Get the existing terminal characteristics and save them
        CALL GET_TERMINAL_CHARACTERISTICS (CHANNEL,CLASS,
        1       TYPE,WIDTH,BASIC,EXTENDED)
        OLD_BASIC    = BASIC
        OLD_EXTENDED = EXTENDED

C       If the terminal does not support Regis return with an error status
        IF((EXTENDED .AND. TT2$M_REGIS) .EQ. 0) THEN    !Invalid chars.
              CONVERT_SCREEN_TO_SIXEL = SS$_IVCHAR
              RETURN
        ENDIF


C-------------------------------------------------------------------------------
C       Everything looks good so far, now change I/O port set-up, CRT set-up.
C       We tell the CRT to send the contents of the screen back to use in SIXEL
C       format. The screen could have been painted with FMS, TDMS, DECGRAPH,
C       DECSLIDE, VT200 escape sequences, or ordinary text we don't care. We
C       will read the SIXEL code from the CRT and store it in our file.
C-------------------------------------------------------------------------------

C       Associate LUN 7 with SYS$OUTPUT and allow for large records
        OPEN(UNIT=7,NAME='SYS$OUTPUT',RECL=1024,STATUS='OLD',ERR=1200)

C       Lock the keyboard to avoid extraneous input during conversion
        WRITE(7,18)LOCK_KEYBOARD
18      FORMAT('$',4A1)

C       Open a new file to contain the SIXEL information
```

```
      OPEN(UNIT=41,NAME='SIXEL$DEVICE',ACCESS='SEQUENTIAL',STATUS='NEW',
     1      FORM='UNFORMATTED',RECORDTYPE='VARIABLE',
     2      CARRIAGECONTROL='LIST',ERR=1400)


C     Set the required terminal characteristics: HOSTSYNC, NOBRDCAST,
C     NOECHO, TTSYNC, NOWRAP, PASTHRU
      BASIC = BASIC .OR.
     1         TT$M_HOSTSYNC .OR.
     2         TT$M_NOBRDCST .OR.
     3         TT$M_NOECHO   .OR.
     4         TT$M_TTSYNC
      BASIC = IBCLR(BASIC,TT$V_WRAP)
      BASIC = IBCLR(BASIC,TT$V_EIGHTBIT)7 bit codes
      EXTENDED = EXTENDED .OR. TT2$M_PASTHRU  .OR.
     1             TT2$M_XON

      CALL SET_TERMINAL_CHARACTERISTICS (CHANNEL,CLASS,
     1      TYPE,WIDTH,BASIC,EXTENDED)

C     Send the appropriate escape sequence to SYS$OUTPUT
C     to set it up to print graphics

101   FORMAT('+',2A1,5A1,4A1,A13,$)
102   FORMAT('+',6A1,$)


C     Check for the optional SIXEL_MODE argument
      IF (SIXEL_MODE .EQ. 'ROTATED')    WRITE(7,102)ROTATED_PRINT
      IF (SIXEL_MODE .EQ. 'COMPRESSED') WRITE(7,102)COMPRESSED_PRINT
      IF (SIXEL_MODE .EQ. 'EXPANDED')   WRITE(7,102)EXPANDED_PRINT

            WRITE(7,101)EXIT_REGIS,GRAPHICS_TO_HOST,
     1             ENTER_REGIS,MAKE_HARDCOPY

            N    = 1                      !Position in BUFF
            NESC = 0                      !Number of <ESC>'s detected
            QUIT = .FALSE.                !Quit code
C-------------------------------------------------------------------------------
C     We now read a character at a time until we get three <ESC>.
C     Three <ESC> means we are finished getting the Sixel code.
C     For every 255 characters that are read, one output record is formed.
C-------------------------------------------------------------------------------

200         STATUS = SYS$QIOW(,
     1                %VAL(CHANNEL),              !SYS$OUTPUT
     2                %VAL(IO$READVBLK),          !Read virtual block
     3                IOSB,,,                    !I/O Status block
     4                %REF(BUFF(N:N)),            !Input Buffer
     5                %VAL(INPUT_BUFFER_SIZE),,,,) !Buffer Size

            IF (.NOT. STATUS) GOTO 1000           !Processing Error
            IF (.NOT. IOSB.IOSTAT) GOTO 1000       !IO status error
            IF (QUIT) GOTO 250                    !We have the final ESC,
                                                 !done.
```

```
            IF(BUFF(N:N) .EQ. CHAR(27))NESC=NESC+1        !Check for <ESC>
            IF(NESC .EQ. 3)QUIT = .TRUE.                  !Escape on the
                                                          !byte after the
                                                          !third <ESC>

            IF (N .EQ. 255) THEN          !Our output record is full,
                  WRITE(41)BUFF(1:255)    !Flush the buffer
                  N=0                     !Reset the character counter
            ENDIF

            N = N+1                 !Increase the character counter for
                                    !every character read
            GOTO 200                 !We keep processing


C-------------------------------------------------------------------------------
C     When we get here after we have read all the data from the CRT.
C     The buffer is flushed, file is closed, and the CRT is set back to normal
C     mode, keyboard is unlocked.
C-------------------------------------------------------------------------------

250         WRITE(41)BUFF(1:N)             !Write out the final sequence
            CLOSE(UNIT=41)                 !Close the file

            WRITE(7,252)EXIT_GRAPHICS,UNLOCK_KEYBOARD,GRAPHICS_TO_PRINTER
252         FORMAT('+',A2,4A1,5A1)

C     Reset terminal characteristics and return with success status
      CALL SET_TERMINAL_CHARACTERISTICS (CHANNEL,CLASS,
      1     TYPE,WIDTH,OLD_BASIC,OLD_EXTENDED)

      CONVERT_SCREEN_TO_SIXEL = SS$_NORMAL
      RETURN
C-------------------------------------------------------------------------------
C     The following lines are for processing errors
C     First reset terminal characteristics, if necessary,
C      then return with error status.
C-------------------------------------------------------------------------------

1000  WRITE(7,252)EXIT_GRAPHICS,UNLOCK_KEYBOARD,GRAPHICS_TO_PRINTER
      CALL SET_TERMINAL_CHARACTERISTICS (CHANNEL,CLASS,
      1     TYPE,WIDTH,OLD_BASIC,OLD_EXTENDED)

      CONVERT_SCREEN_TO_SIXEL = SIXEL_READ_ERROR  !Error reading SIXEL file
      RETURN
1200  CONVERT_SCREEN_TO_SIXEL = CRT_OPEN_ERROR    !Error opening SYS$OUTPUT
      RETURN
1400  CONVERT_SCREEN_TO_SIXEL = SIXEL_OPEN_ERROR   !Error opening Sixel file
      RETURN
      END
```

Please Note

The subroutines SET_TERMINAL_CHARACTERISTICS and GET_TERMINAL_CHARACTERISTICS were included in the May 1986 issue of the *Wombat Examiner.*

DTR/4GL Notes from Nashville

Many thanks to Larry Kilgallen for forwarding the DTR/4GL related material from the VAXnotes conference conducted during the Spring Symposium in Nashville. The following was excerpted from that material.

DTR-11 TO DTR-32 HELP!!

How can I convert a Datatrieve-11 Query Dictionary to VMS CDD format??? Please help I have a method which requires basically an ASCII transfer "SHOW object-name" to a file which is not very efficient.

Response #1 - Moving DTR 11 Dictionaries

Larry Jasmann

It isn't very difficult. In DTR11 you can EXTRACT all to put the contents of the Query Dic into a command file. (at one time there used to be a special utility to do it but I THINK that they put the functionality in the product. After this, shift the ASCII file over to the VAX, create the dictionary node and @<filename> will load the contents of the dictionary into the CDD. Almost everything will work OK.

Response #2 - QXTR does it

Bert Roseberry

Actually what Larry said was more or less correct BUT try this first while on your PDP system to convert the QUERY.DIC . When I converted long, long time ago there was a program called QXTR . This should be in your system directory, I guess [1,1] ? Anyway RUN QXTR and it will prompt for an input dictionary name. I think one of the things this methodology will do is is add a record definition clause that says ALLOCATION IS LEFT-RIGHT which is needed depending on how your data is brought over. Larry's method does not add this clause. Try QXTR and see what it does. - Bert Roseberry (202) 267-2624

DTR Graphics output devices

Can DTR graphics be done on any of the following hardware:

a. LA100 ?

b. PRINTRONIX LXY22 ?

c. COMPAQ DeskPro 286 with a CGA interface ?

Response #1 - DTR Graphics output devices

Pat Scopelliti

a) Well.. you can send sixel files to the LA100. You can convert a ReGIS file to sixels by either typing in the program from the WOMBAT Examiner a couple of months back or if you have DECslide just use $ SLIDE/NOINT/SIXEL=sixel-filename regis-filename

b) You can convert sixel to LXY format.. its not terribly tough. Essentially, each sixel character represents 6 bits in a vertical column, LXY graphics represents 6 horizontal bits in a single character. Not really that hard to do. You'll need any DEC terminals manual that describes sixel graphics and your LXY manual that describes LXY graphics. (Don't forget to do a print/nofeed when printing the LXY graphics image!)

c) Dunno...can others help here?

Response #2 - Not much out there...

R. Hassinger
The file translator software for the new LN03R is said to convert ReGIS to Postscript so we may be able to convert output from DTR graphics to print on the new printer at least. In general there has been very little in the way of conversion software and I have not seen any sign of DEC providing different graphics output formats for DTR to date - they should at least give us GKS. Has anyone else heard of any code that parses ReGIS? With a good front end the various conversions should not be too bad to do.

Response #3 - WOMBATS get graphical - GKS??

Al Sorrell, Westinghouse Balto.
What DTR really needs is a hook into GKS - DEC Seems to have defined this as their long-range graphics philosophy. Since it supports user-written drivers (and the doc's pretty well written too!) this would eliminate many of the problems of "How do I output DTR on my IDGET-1000".

Callable DTR from Allin1

Callable DTR 4.0 from Allin1 v2.1 doesn't seem to let the up cursor recall the last command. DTR from DCL does. Is this a problem with Allin1 or callable DTR (or me)?

No Response - Forwarded to the Wombat Wizard

DTR PLOT-caused system deaths

I have a problem with DTR - users collect 300 records or so, do either a multi-line or multi-bar plot, DTR blows itself up to lots of megs and my entire system locks up and goes to lunch with OPA0: messages about page file space being critical. Even <ctrl-C> does not stop this. If I leave the system alone it comes back in 5 or 10 minutes if I am lucky. I don't like it when nonpriv users can effectively kill my system with an innocent command on a standard product. Anybody else have this problem or know a fix?

Response #1 - -< PAGEFILE(s) too small

Denny Thury, Texas Instrument Incorp

What the error message told you, as system manager, is that your pagefile is too small! According to metrics I've gotten from DEC's performance mgt seminars (not necessarily at this DECUS), the PAGEFILE(s) usage should not exceed 55-60% utilization; nor should the SWAPFILE usage exceed 70-75% utilization!

More Info

Yes, I realize that. What I should have made clearer is that DTR blows up to about a 6-meg task size. Just how incredibly huge should I make my paging file anyway? If somebody tries to plot 1200 records do I need a 40-meg page file to support one user?

Response #2 - Should Not Happen

Pat Scopelliti

You have apparently discovered a bug. What version of DTR are you using? And are the records stored in RMS files? I have successfully plotted tens-of-thousands of records using DTR and it's PLOT statement, so your problem is a bit unexpected. Also, was the problem repeatable? Did it happen on a quiet system?

Help with DTR graphics

I'm relatively new to the DATATRIEVE world, and our group is about embark on a project that will rely on this project. Basically the project involves the following:

1. Using monitor on all of our systems to gather performance/capacity data.
2. Once a month, this data is transmitted to a central processing system.
3. This data is then processed by a collection of DTR procedures
4. Finally several plots of the data are produced for review by all levels of management. We see most (if not all) of these procedures taking place in essentially a batch environment.

Here's my problem; The copies need to be hardcopy! I know DTR can direct the plots to a file. But these plots use ReGIS! What kind of printers/plotters are available for ReGIS graphics? Does anyone know of anyway to translate ReGIS graphics to say an HP 72xx pen-plotter format? (We're looking at bar & stacked-bar charts, as well as "simple" line graphs)! Perhaps a more fundamental question is: Is what we're planning a practical application of DTR?
Thanks in advance

Denny Thury
Texas Instruments Incorporated
(214)952-2066

Response #1 - DEC has a REGIS Printer

Bob Graham, Dow Chemical

DEC sells an inkjet printer called an LCG01 (formerly the LCP01) which handles REGIS. We've had our for about 1-1/2 years now with reasonably good results. Users are sending output to it from DTR, DECgraph, DECslide, SAS/Graph and a few other odds and ends.

The only real problems have been the price (start thinking about 5 digit numbers...) and the reliability. The print engine is a Tektronix (mumble) which runs fine unless a DEC field service rep touches it, then its out of commission for at least a week. After several repeats, we convinced the local FS office to just call Tektronix FS and send one of their guys over to work on it.

Response #2 - Look at the LN03R Postscript Printer

R. Hassinger

The new LN03R uses a software package that does conversion from several formats to Postscript. REGIS is one of the formats it handles. I assume your DTR plots could go through that too.

Ref. Response #2

Is the reference software package part of the LN03R product? If NOT where/how do I get a copy? Thanks, Denny

Response #3 - ReGIS to HPGL Translation

Tony Scandora

There was a ReGIS to HPGL converter called something like RTHP on a recent RSX or VAX SIG tape. I can't remember any more right now, but if you are interested in HP plotters, contact me and I'll find it for you.

Tony Scandora,
Argonne National Lab, CMT 205
Argonne, IL 60439
312-972-7541

Response #4 - You could do it (maybe), but it would be wrong

Saul Tannenbaum

Having done this sort of thing, my advice, in the strongest possible terms, is to get something like SPSSX or SAS, that can do _real_ data analysis and graphics. SPSSX has an option to interface to monitor data (and I believe the new SAS version does, too). We drag performance files (SPM actually, but monitor stuff would work as well) into SPSSX with it's GET DATATRIEVE facility, massage them as needed, and produce spectacular full color plots on HP gear with SPSS graphics. Management (who wouldn't know real performance data if it hit them in the head) eats the stuff up.

DEFINE PROCEDURE BUG

I have found a nuisance problem. When I try to define a procedure SP I get an error. This is what happens:

```
DTR> DEFINE PROCEDURE SP

%CDD-E-ILLNAMCHR, a given name contains a character other than A - Z,
    0 -9, $, _

DTR>
```

Response #1 - Logical Names Can Cause It.

B. Dooley

Every time that I have encountered this bug it was because some CDD element that I was trying to reference (record definition or domains usually) had the same name as a logical name that was defined in my system. So when I typed READY DOMAIN, what the system was really seeing was READY DISK1:[USER_DIR]FILE.EXT.

Response #2 - Getting Around Logical Names

Tony Scandora

If you have VAX-11 RSX installed on your system, it probably assigns the logicals SY, LB, SP, and WK that RSX likes to know about. These are in the system table, so they're hard to ignore. The easiest fix is to not use them or anything else you see in a show logicals listing. If you really like the name SP, you can use the name _SP to get the domain, as in

```
DTR> ready _sp
DTR> print all a,b,c in _sp with b > 100    or whatever.
```

There are times when DTR doesn't want the underscore, and it will tell you when you get it wrong.

DTR040_FILES.DAT - Huh?

Following the installation of DTR040, I found files like this in SYS$SYSTEM - it implies dire consequences if I delete it! Could somebody enlighten me on the why's and wherefore's of this file?

(Obviously it is a list of the files added to my system - is this just a convenience for the next update of DTR - [also noticed a file fro CDD] - if so, can I expect to see one of the files for EVERY product on my system??)

I must admit that it is handy having a file which documents when the product was installed, etc.

```
!  File:       SYS$COMMON:[SYSEXE]DTR040_FILES.DAT
!  Product:    VAX DATATRIEVE V4.0-1
!  Installed: 18-MAR-1987 17:15
!*********************************************
!  This file is associated with the above product.
!  It must NOT be moved, edited, or deleted.
!*********************************************
!  Flags: (for DIGITAL use only)
!     $D = Delete file when deleting product
!     $I = File inserted into IMAGELIB.OLB
!     $K = Keep file when deleting product
!     $M = Multiple version/product file
!-----------------------------------------------
SYS$COMMON:[DTR]ACCOUNTS.DAT /$M
SYS$COMMON:[DTR]ANNUAL.DAT /$M
SYS$COMMON:[DTR]BUY.DBS /$M
SYS$COMMON:[DTR]BUY.SNP /$M
SYS$COMMON:[DTR]DTR040.DAT /$D
SYS$COMMON:[DTR]DTRFIND.COM /$M
SYS$COMMON:[DTR]DTRFND.MAR /$D
              .
              .
              .
SYS$COMMON:[SYSTEST.DTR]FORMS.DTR /$M
SYS$COMMON:[SYSTEST.DTR]PLOTS.DTR /$M
SYS$COMMON:[SYSTEST.DTR]RDB.DTR /$M
```

No Response - Forwarded to the Wombat Wizard

Bert Roseberry's First Annual Best DECUS Awards

*** Best Threat by a Digital Employee**

This lady employee from Digital was waiting in line for the free glass mug in the shape of a boot. When some man stepped in front of her she told him, "If you don't move I'm going to goose you." He moved.

*** Best Enhancement to Datatrieve**

After eating lunch at the same resturaunt as Minnie Pearl, Datatrieve developer Doug Cropper has modified source code so that the command READY YACHTS now prints out the word, "How-DEE!"

*** Best side-step by a Digital Employee**

When Lew Lasher was asked by a member of the audience as to the possible time frame for new enhancements to Rally.

Audience - "All I want to know is will I be sunning myself on the beach or skiing down the slopes when this comes out ?"

Lew Lasher - "That depends on whether you are in the northern or southern hemisphere."

*** Best Advise from One Digital Employee to Another**

"Hey take off your DEC ribbon when you go around the exhibit hall and you'll get more stuff."

*** Best Shock**

"NOW she tells me she's married and her husband is a policeman with a large gun collection."

*** Best Known VIA Product Developer with the Least Known Name**

Dan Efemmess

*** Best Line in a Nashville Bar**

"Rocky Top Tennessee? Don't know if I have that. Do you know who did it ?"

*** Best Lie Given to an Attractive Hostess in a Nashville Restaurant**

"I really don't notice any accent."

*** Best Battle**

Joe Angelico and Chris Wool holding back people before the reception Sunday.

*** Best Kept Secret**

How to get to the hotel before the continental breakfast disappeared.

*** Best Dressed**

Joyce from Indiana (I figued if I mentioned her maybe she would call me up if she ever came to the Washington, DC area)

*** Best News of All**

Next DECUS symposium is only five months away.

Converting from User-defined Units to Device-dependent Units

Steven Cordiviola, Kentucky Geological Survey, Univ. of Kentucky, Lexington, KY

Have you ever been in the situation where you want to do graphics in DATATRIEVE, but you where hindered by the lack of a graphics terminal or printer or simply did not have any graphics devices? Any device has the potential to print characters anywhere on the physical portion of the device by using control codes. VT100 terminals, for instance use a control code with a row and column value to position the cursor anywhere on the screen.

One of the first problems users encounter is "How do I map my data, which are in units meaningful to my application, onto a device which uses its own units?" For example, we use geographic coordinates in locating our data points. If I want to use a non-graphics terminal to show the spatial relationships of the points, I must convert the geographic coordinates, which range from 0 - 1,000,000 meters in both x and y directions, into 1 - 80 columns and 1 - 24 rows.

An easy solution is a formula which converts from user units to device units. This formula, which was provided in Hewlett-Packard's plotting manuals, can be used on any device with direct pen or cursor positioning capabilities or a more involved program can be written to place characters on the device. In DATATRIEVE the formula can be used with user's data to provide "poor-man's" DATATRIEVE graphics. The following example illustrates the formula and its use in VAX DATATRIEVE.

Figure 1 represents a potential plotting area (ex. terminal screen):

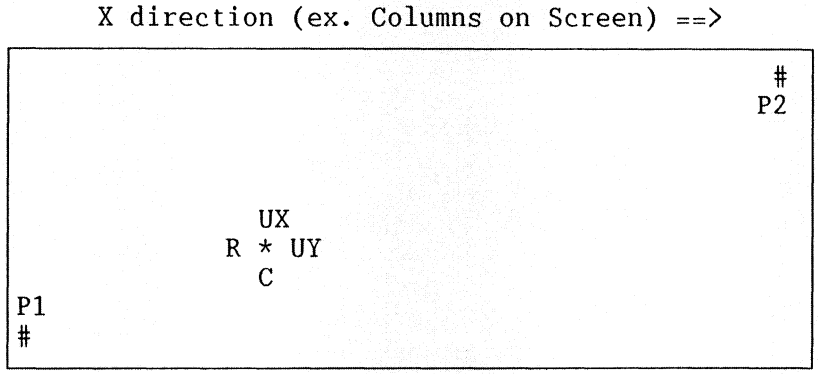


Figure 1 - Example of a Plotting Area

where:

P1 the lower left-most corner of the area the graph is to fit.

For P1 the following information is needed:

- Dy1 is the Y Coordinate in DEVICE UNITS
- Dx1 is the X Coordinate in DEVICE UNITS
- Ux1 is the X coordinate in User Units at P1 usually the origin of plot
- Uy1 is the Y coordinate in User Units at P1 usually the origin of plot

P2 the Upper right-most corner of the area the graph is to fit.

For P2 the following information is needed:

- Dy2 is the Y Coordinate in DEVICE UNITS
- Dx2 is the X Coordinate in DEVICE UNITS
- Ux2 is the X coordinate in User Units at P2 usually the Maximum point to be plotted
- Uy2 is the Y coordinate in User Units at P2 usually the Maximum point to be plotted

UX is the X coordinate of point to be mapped (in User Units)

UY is the Y coordinate of point to be mapped (in User Units)

R is the Y coordinate which is calculated and given in DEVICE Units for the point of interest.

C is the X coordinate which is calculated and given in DEVICE Units for the point of interest.

In DATATRIEVE, a user can create a procedure or record definition which will calculate C and R using COMPUTED BY fields with data stored in a file. To plot the data, the user would proceed to PRINT the necessary device dependent commands using the calculated coordinates on the appropriate device.

To Calculate C:

$$C = \left[\frac{Dx2 - Dx1}{Ux2 - Ux1}\right] * UX * Dx1 * Uc1 * \left[\frac{Dx2 - Dx1}{Ux2 - Ux1}\right]$$

To Calculate R :

$$R = \left[\frac{Dy2 - Dy1}{Uy2 - Uy1}\right] * UY * Dy1 * Ur1 * \left[\frac{Dy2 - Dy1}{Uy2 - Uy1}\right]$$

On a terminal the ANSI escape sequences can be used to position the cursor at the appropriate Row and Column, then printing the information desired. For example:

```
DEFINE PROCEDURE Formula_PU_to_UU

        ! Declare the variables to calculate device x and y using
        !  the formulas published in the H-P plotting manuals.
        !
        !       will assume X_COORD and Y_COORD will be supplied by
        !       appropriate RSE...
        !
        !       For Lower Left...
        !                       Dy1     Lower left Y Coordinate in DEVICE UNITS
        !                       Dx1     Lower left X Coordinate in DEVICE UNITS
        !                       Ux1     Lower Left X coordinate in USER UNITS
        !                                  usually the origin of plot
        !                       Uy1     Lower Left Y coordinate in USER UNITS
        !                                  usually the origin of plot
        !       For Upper right....
        !                       Dy2     Upper Right Y Coordinate in DEVICE UNITS
        !                       Dx2     Upper Right X Coordinate in DEVICE UNITS
        !                       Ux2     Upper Right X coordinate in USER UNITS
        !                                usually the Maximum point to be plotted
        !                       Uy2     Upper Right Y coordinate in USER UNITS
        !                                usually the Maximum point to be plotted
        !
        !       UX      is the X coordinate of point to be mapped (User Units)
        !
        !       UY      is the Y coordinate of point to be mapped (User Units)
        !
        !       ROW  = Y dimension of plot in DEVICE UNITS
        !       COLM = X dimension of plot IN DEVICE UNITS

                DECLARE DX1 USAGE REAL.
                DECLARE DX2 USAGE REAL.
                DECLARE DY1 USAGE REAL.
                DECLARE DY2 USAGE REAL.
        !
                DECLARE UX1 USAGE REAL.
                DECLARE UX2 USAGE REAL.
                DECLARE UY1 USAGE REAL.
```

```
            DECLARE UY2 USAGE REAL.
        !
        DECLARE PART_X COMPUTED BY (DX2 - DX1) / (UX2 - UX1 ).

        DECLARE PART_Y COMPUTED BY (DY2 - DY1) / (UY2 - UY1 ).

        DECLARE COLM COMPUTED BY
                ( PART_X * X_COORD ) + DX1 - ( UX1 * PART_X )
                EDIT_STRING ZZZZZZ.    ! can be anything device expects

        DECLARE ROW COMPUTED BY
                ( PART_Y * Y_COORD ) + DY1 - ( UY1 * PART_Y )
                EDIT_STRING ZZZZZZ.    ! can be anything device expects

END_PROCEDURE


DEFINE PROCEDURE TERMINAL_GRAPHICS

        ! This procedure will plot a character at the proper
        ! TERMINAL ROW and COLUMN from user supplied coordinates
        !
        ! It uses the procedure Formula_PU_to_UU to declare the variables:
        !       ROW and COLM from user supplied variables X_COORD and Y_COORD.
        !
        ! This procedure uses the VTxxx cursor placement escape sequences
        !       <ESC>[ row ; column   to position cursor.
        !
        !   User must supply variables X_COORD and Y_COORD, which are
        !   coordinates of the desired point in USER_UNITS.  SYMBOL
        !   is user supplied and will be printed at the proper
        !   terminal location
        !
        ! DEFINE THE ESCAPE CHARACTER (NOTE use the editor to insert
        ! the "<ESC>" by pressing the ESCAPE KEY twice)
        !
                DECLARE ESCAPE COMPUTED BY "<ESC>".
        !
        !       call procedure Formula_PU_to_UU to declare variables...
        !
                :Formula_PU_to_UU
        !
        ! Set the known variables Dx1, Dx2, Dy1, Dy2 for a terminal screen
        !
                DX1 =  0.0
                DX2 = 80.0
                DY1 = 20.0
                DY2 =  0.0
        !
        !  Now prompt user for Ux1, Ux2, Uy1, Uy2 to determine how their
        !  data will map onto screen window.  This could be automated with
        !  MAX and MIN statements...
        !
        UX1 = *." X coordinate of Lower Left position of map"
        UY1 = *." Y coordinate of Lower Left position of map"
        UX2 = *." X coordinate of Upper Right position of map"
        UY2 = *." Y coordinate of Upper Right position of map"

END_PROCEDURE
```

Now the user can call the procedure TERMINAL_GRAPHICS prior to "Printing" the data to the screen. The statement would look something like this:

```
DEFINE RECORD FOO USING
    01 FOO.
        .
        .
        .
        05 X_COORD PIC IS 9(9).
        05 Y_COORD PIC IS 9(9).
        05 SYMBOL PIC IS X.      ! CHARACTER TO BE PRINTED
        .
        .
        . ;

:TERMINAL_GRAPHICS

FOR rse
BEGIN
!
! For VT100 type terminals the " ESCAPE [ row ; column " sequence will
! position the cursor at the appropriate row and column.
!
      PRINT ESCAPE || "[" || ROW || ";" || COLM || "f" || SYMBOL
END
```

Note in the above print statement, ROW and COLM may be formatted differently than what the device expects. For example, most terminals expect integer data and DATATRIEVE prints floating point values. Format statements or the built in functions (i.e. FN$NINT) help eliminate any problems.

MACHINE BITES WOMBAT

(or, How to stump the "experts")

Bart Lederman, ITT, New York, NY

At the recent Nashville symposium, as has been done for many years, the Datatrieve/Fourth Generation Languages SIG Suite/Campground was open for people to come in and ask questions, have problems solved, or just rest from the hubbub of the symposum. One such user came in and posed a problem: when he tried to define a procedure called "SP", Datatrieve told him the name was invalid as it contained non-alphanumeric characters. As both the letters "S" and "P" are alphanumeric, he could not understand why it would not work. We could not understand why either, as SP is not a reserved word: we also tried defining a procedure named SP on the Micro-VAX in the campground and had no trouble doing so. The user went away dissatisfied, but not discouraged. He later found out what the problem was using the VAX Cluster in the display area, and I was fortunate enough to run into him again Saturday morning on the way to the airport and found out what the problem was.

On his machine, and on the VAX Cluster at the Symposium, there was a system logical name defined which translated "SP" to "LPA0:". Users of VAX-11/RSX will recognise this as a way to direct the "spooled" print device to a real printer. Because the logical name tranlsation contains a non-alphanumeric character (":") Datatrieve rejects it as a procedure name. The Micro-VAX in the campground did not have this logical name defined, so the procedure name was accepted.

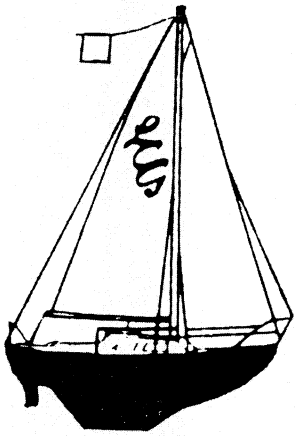
The moral: in addition to checking for reserved words, logical name translations should also be checked when DTR rejects a name.

DECUS

LARGE SYSTEMS SIG DECUS US CHAPTER

DIGITAL EQUIPMENT COMPUTER USERS SOCIETY

LS



AT LARGE

The Newsletter of the Large Systems SIG

Contributions

Contributions and suggestions for this newsletter are constantly needed. Articles, letters, technical tips, or anything of interest to our SIG are greatly appreciated. The editor prefers submissions be made electronically, but magnetic tape and hard copy will be accepted.

Send your contributions to:

ARPAnet/CSnet: ctp@sally.utexas.edu
UUCP: ctp@ut-sally.uucp ({harvard,ihnp4,seismo}!ut-sally!ctp)
CIS: 75226,3135
BITNET: use the Wisconsin Gateway

or if you must, use the U. S. Mails:

Clyde T. Poole
The University of Texas at Austin
Department of Computer Sciences
Taylor Hall 2.124
Austin, Texas 78712-1188

Table of Contents

Chairperson's Article

Leslie Maltz, Stevens Institute of Technology, Hoboken, New Jersey

Because last month's publication deadline fell during symposium season, this article will touch on a variety of topics that have occurred during the last few months. Beginning with the things that happened at the Spring Symposium is probably a good start. The overall view was that the Nashville Symposium was a huge success. The sessions went very well, the facilities seemed good, and the weather was quite pleasant. Importantly, there did not seem to be any monumental faux pas or catastrophic announcement as we had seen last year at the Fall Symposium. There were Digital presentations on the latest iteration of the software license transfer policy. But no surprises this time.

The Large Systems SIG sponsored our usual series of sessions. Digital sponsored some evening activities that complemented our regular schedule and gave us all a chance to get acquainted. Our group is growing as the number of sites acquiring large VAX systems increases. We welcome the new faces and ideas, and encourage ongoing participation.

The SIG sponsored some Birds-of-a-Feather sessions that are worth noting. One such session was on supercomputing. Some DECUS members are employed at sites that have supercomputers as well as DEC equipment. Still others have access to supercomputers such as ones associated with regional or statewide networks. A growing number of people are gaining access to the supercomputer facilities at the national centers funded by the National Science Foundation. Quite a few of our members have expressed an interest in sessions that concentrate on the support, software compatibility and tools, interconnection, and use of such facilities in relation to our DEC equipment. The level of attendance and discussion at the Birds-of-a-Feather session verified this interest. The SIG will be sponsoring an expanded set of activities at the Fall Symposium in Anaheim on this topic.

Another B-O-F session held in Nashville pertained to the use of Ultrix/Unix on DEC's large VAX systems. This session was also very interesting and it appears that interest in this area will be growing over time. The SIG will continue to pursue this topic at future symposia, and will be coordinating such efforts with UNISIG to best meet the needs of the users of such systems.

The Large Systems SIG has had some changes in recent months. Due to changes in their work environments and/or non-DECUS lives, some of the SIG leadership have had to say farewell to their DECUS roles. Specifically, Carla Rissmeyer, Chuck Bacon, and Dave Edwards have expressed their regrets, but will not be continuing on the Steering Committee. Carla, Chuck, and Dave have been active members of the Steering Committee and we will miss their efforts and their presence. We hope to see them back at some time in the near future, and we are most grateful to have worked with them. The SIG benefited much from their efforts and we wish them much luck in their endeavors.

On a related note, with the resignation of some of the SIG leadership we appear to have some openings on the Steering Committee that need filling. The current openings are for the Coordinators for Systems Software, the SIG Menu, and Languages and Tools. If you are interested in any of these positions or perhaps aren't sure what might be involved, please contact me. Involvement is usually painless, and the experience can be very positive. In addition to the above positions, we are seriously looking for volunteers for a variety of positions ranging from session chairs to assistant coordinators for the Steering Committee positions. DECUS can't function without volunteer help. How about starting now? We need your ideas and your help.

From the INFO-VAX Mailing List

Abstracted by: Betsy Ramsey, American Mathematical Society, Providence, RI

The following messages are selections taken from the INFO-VAX interest group, which is a mailing list maintained on the DARPA Internet. These items appear for information purposes only. Neither DECUS nor the authors assume any responsibility regarding the usefulness or accuracy of the information herein. By convention, lines beginning with ">" are extracts from previous mail messages that are included for clarity.

Date: 9 APR 87 08:38-CST
From: Mark Moore <MOORE%UTHSCSA.BITNET@WISCVM.WISC.EDU>
Organization: The University of Texas, Health Science Center, San Antonio, Texas
Subject: Software installation

Well it's me again. I have another question about policy at different sites. My last question of this type netted one response, but I will try again.

We have four VAXen in a cluster with 9 RA81s that are dual ported with two HSC50s. When we install software on the cluster (e.g. new versions of FORTRAN or Datatrieve) we will disable logins on all four machines, since anything the installation procedure touches is accessible to all of the machines. We don't have any real schedule for software installation. We will generally wait until two or three products have arrived so we can get them all done at once.

Management is complaining that it is absurd to disable all four machines for the installation of software. And that at least some of the machines should be up at all times. I could take one machine down and install the software, but...

I would like to establish a software installation policy and put this one to bed. You can help me do this. Please tell me what your software installation policy is. A couple of lines will do. I just want some ideas.

Date: Thu, 9 Apr 87 20:51:09 MST
From: cetron%ced@cs.utah.edu (Ed Cetron)
Organization: Center for Engineering Design, Univ. of Utah
Subject: Re: Software installation

I have over and over found the only reliable way is to shut everything down and do it at once..... While my management is as bewildered as yours, mine lost almost 2 days' work when he demanded to keep running while I installed and tested a new BRU (RSX's sorta equivalent to backup) - and while it recreated his entire directory, every file was empty.....I don't get questioned much any more and can often be found doing updates Saturday night at 3am....

Date: Fri, 10 Apr 87 13:21:10 PST
From: Frank Nagy <nagy%43198.hepnet@LBL.ARPA>
Organization: Fermilab Research Division EED/Controls
Subject: RE: Software installation

Being a relatively fearless system manager, I've installed "minor" items like new compiler versions on live systems on a VAXcluster and then just logged into the "other" system and used

INSTALL to REPLACE the shared images.

Datatrieve and others were done on semi-live systems due to the length of time required for the install and other factors (more complex products).

Factors in installing on live systems are:

- updating the shared images on the other nodes (can be easily done with INSTALL REPLACE).
- a user might interfere with the install by using the HELPLIB at the time the installation attempts to put a new module into the library.
- new DCLTABLES can be installed on other nodes by the same INSTALL trick as above. But users must logout and back in to use the new tables.
- others I've probably forgotten.

Date: Mon, 13 Apr 87 12:17:19 EDT
From: tedcrane@tcgould.tn.cornell.edu (Ted Crane)
Organization: Program of Computer Graphics, Cornell University
Subject: Re: Software installation

As a rule, we have been able to do most installations online. We fearlessly (foolishly) answer yes to the "following users are logged in," "DECnet is active," and "Satisfied with Backup" questions. VMSINSTAL proceeds happily.

The following kits cannot always be installed online:

VMS (this one would be foolish to try)

DECnet (ditto)

LSE (may require rebuilding of user's section files)

Compilers are a pretty safe bet.

The tricks mentioned in previous responses (Using INSTALL on other cluster members to update DCLtables, etc.) are mandatory if you want to keep your head on your neck after the online installation.

Date: Mon, 13 Apr 87 15:53 ADT
From: Aidan Evans <AE%DAL.BITNET@WISCVM.WISC.EDU>
Organization: Dalhousie University
Subject: VMS volume sets

We will shortly be installing a VAX 8800 with 11 RA81 disks on an HSC70; it is planned to increase the number of RA81's to 22. It has been suggested that the RA81's should be configured in volume sets so that

(a) the users have fewer disks to know about;

(b) we can have files bigger than one disk (450 megabytes);

(c) disk space utilization will be greater because whole disks (1 to number in the set minus 1) can be filled instead of having wasted space at the "end" of each disk.

The basic arguments against volume sets seem to be

(a) if one disk in a set fails, the whole set is out of commission;

(b) online backup of volume sets might interfere more with users than single disk online backups;

(c) while it is easy to add disks to a set, it is difficult to break a set up.

I know that the SYSTEM disk must not be put in a set. I am interested in comments on the arguments I have listed and in additional ideas for or against using volume sets.

Date: Tue, 14 Apr 87 04:52:18 PST
From: Frank Nagy <nagy%43198.hepnet@LBL.ARPA>
Organization: Fermilab Research Division EED/Controls
Subject: RE: Volume Sets

Some points for/against volume sets:

1. Too many volumes in a volume set is probably hazardous to a system manager's health. Especially in a software development or normal timesharing environment where there are lots of little files on the disks...

2. Free space on members of a volume set are used in parallel by writing a new file on the member with the greatest amount of free space instead of completely filling member #1 then filling member #2, etc.

3. Volume sets to improve performance by bring more seek arms to play on the file system.

4. In my past life as a system manager, I'd only played with dual-disk volume sets: 2 pairs of RA81s and a pair of RM80s as 3 separate volume sets. No problems beyond the usual. Online backups proceeded as before and did NOT affect the users any more adversely than online backups did before I installed the volume sets.

The "great" argument for volume sets and for making ALL your file system one giant volume set is make it easy for users to locate the files of other users since they don't have to remember which disk user X is on. This argument is actually rather specious under VMS V4.x: the solution to the "problem" is to define a logical name which is a search list of all the mounted disk volumes in the public file structures:

```
$ define usr$disks usr$disk1:,usr$disk2:,usr$disk3:,usr$disk4:
```

Thus anyone can locate anyone elses file by using the usr$disks logical name as the "device" specification. This assumes that a person only has a single top-level directory on the set of disks listed in usr$disks; this does not seem to be much of a restriction.

Date: 14 Apr 87 10:59 EST
From: John Child <JOHNC%CAD2.decnet@GE-CRD.ARPA>
Organization: General Electric, Aircraft Engines, Lynn MA
Subject: Vol Sets and Search-list Logicals

More on Volume Sets:

Several writers have discussed the pros of volume sets: larger files, more spindles, fewer device names to remember, less confusion about "where" another user's directory is. Several have also mentioned some cons: longer backups, more data (the whole volume set) unavailable when a disk fails, difficult to break up into separate volumes w/out "n" free volumes, where "n" is the size of

the volume set.

I haven't read anything about what I believe is the worst problem with volume sets: When a disk fails not only is all data on the volume set unavailable, but the entire volume set has to be restored from backup when the failed disk is repaired or replaced. Other vendors (like IBM or Honeywell) don't make you do this: when a disk in a pool fails then any file which is on that disk (all or part of the file) is unavailable, but the rest of the pool functions normally. When the disk is replaced or repaired it is restored from tape; but only the files which were actually on that volume have to be reloaded.

If volume sets worked better I would certainly use them - but the failure rate of RA81's being what it is I don't want to put more than the disk's 456 meg of data at risk of a single device failure.

> ...the solution to the "problem" is to define a logical name which is a
> search list of all the mounted disk volumes in the public file structures:
>
> $define usr$disks usr$disk1:,usr$disk2:,usr$disk3:,usr$disk4:
>
> Thus anyone can locate anyone elses file by using the usr$disks logical
> name as the "device" specification. This assumes that a person only has
> a single top-level directory on the set of disks listed in usr$disks;
> this does not seem to be much of a restriction.

Nope. This doesn't work correctly either. Search lists do their thing fine for read access; but if you want to write to, for example, usr$disks:[sample]xxx.dat it will fail unless the [sample] directory is in the first volume of the volume set. I'd use this, too, if it worked better. Does anyone from DEC know whether this unfortunate behavior is going to be altered in a future release?

DECUS Europe Large Systems SIG Interests and Activities

Rolf Nordhagen, University of Oslo Computing Centre, Oslo, Norway

---

The impact of high end systems in the VAX product line is creating a growing awareness within DECUS of the need to serve and support users of large systems. Users of high end systems have specific requirements which presently needs to be covered.

Traditionally the mainframe perspective within DECUS has been covered by the 10/20 SIG. The large installations who are migrating to new Digital systems, are bringing to the VAX environment a strong background in precisely the same issues which will have to be resolved in order to make the high end VAX products a viable choice in large system computing. The natural consequence is for the high end VAX users to join with the 10/20 users to form a "Large Systems SIG".

What is a "Large System"?

The term "Large System" cannot be explicitly defined. It can be seen to cover a diffuse area and only some guidelines can be given:

- The system is VAX based and is probably a Cluster, or a 10/20 system.
- It supports a "General Purpose" computing function.

- It has more than 100 users.
- The user community may be distributed over a wide area.
- It has a variety of network communication facilities.
- The purchase price of the system exceeds $1 million.

If your system meets 3 or more of these criteria then it can be defined as a "Large System". Typically it is the type of system found in a computer centre of a company, university or research centre.

What are the areas of interest?

The areas of interest to those responsible for Large Systems differ considerably from those which concern the users and managers of dedicated or smaller systems. Obviously all have in common the VMS operating system, the networks and most of the DEC hardware. The differences lie in the management, the policy making requirements and the extra facilities. Some examples are given below:

- VMS Accounting does not provide the facilities needed from a computer centre and lacks many of the required analysis features. In order to carry out cost centre and project related charging extensive modifications are needed. In addition facilities to permit network utilization to be charged are missing.
- Allocation of user resources is inflexible and requires too much effort to make simple adjustments.
- Better on-line back-ip and disk re-organization utilities are needed.
- Security facilities are an area of particular concern as these systems frequently have dial-in or X.25 access to the PTT networks and are thus more vulnerable.
- Data security and protection facilities including encryption systems are important for many installations.
- Large systems often have large networks and thus network management tools are needed.
- The ability to have the VAX system as a part of a heterogeneous system closely coupled to mainframe computers. One sees, today, many examples where large VAX Clusters are acting as interactive "front-ends" to IBM and Cray systems.
- The issues of software licensing and software costs for a broad spectra are causing increasing concern.
- The RAMP characteristics of the systems, the suppliers ability to support over an extended lifetime are key investment considerations.
- The issues of Field Service support at large sites.
- Batch features are lacking.
- The scheduler is too simple for such Large Systems.
- File base maintenance problems. These is no real archival tool and the disk quota system is too inflexible.

From this list, which is not exhaustive, it can be seen that the interests of the "Large System" community are very different from those of others SIGs.

Activities of the Large Systems SIG

The activities of the European Large Systems SIG are summarized below:

- To provide a separate Menu/SIR activity aimed at high end issues. This includes coverage of issues relating to VAX based systems and also for an iterim period 10/20 issues. This activity is the main vehicle to seek improvements from DEC and the influence through the large purchasing power of LSSIG members gives considerable leverage.
- To provide an active programme of events for the European Symposium.
- To ensure interaction with and support from the Digital organization in the Large Systems area. This is particularly important for Symposium Programs and Exhibits, and in providing support for National Chapter activities.
- To provide a newsletter which is available to the national SIGs.
- To promote and support the formation and activities of the national SIGs.
- For existing 10/20 users activities must be kept at a sufficiently high level to ensure on-going support for these systems during the remaining years.

Relationships with the VAX SIG

An important consideration is the potential overlapping of the Large Systems SIG activities with that of the VAX SIG. It is clear that a close relationship should exist between the two SIGs and that the programmes should be co-operative. The VAX SIG is clearly the place where the detailed operating system related issues are discussed and were the System Manager of dedicated and Large Systems gets his VMS information. In the formation of the Large Systems SIG, therefore it is essential that liaison with the VAX SIG is maintained from the outset.

Further Information

If you need further information please contact:

Rolf Nordhagen,
University of Oslo Computing Centre
P.O. Box 1059 Blindern
0316 Oslo 3
Norway

OFFICE AUTOMATION

IN THIS ISSUE...

FROM THE EDITOR...

Welcome to the Office Automation newsletter. Our newsletter is an open forum for exchanging ideas and information with other OA users. It is also a vehicle for the OA SIG to keep you abreast of our activities and events. We invite you to contact us with comments, ideas, suggestions and of course.... articles.

Many thanks to all of our new readers who subscribed during the Nashville subscription drive!! I am pleased to announce that for new subscriptions turned in during the week, the OA SIG sponsored the most new subscribers.

For those of you who had the opportunity to hear a preview of the up-coming OASIS system, we will be publishing the particulars in the next month or two. For the rest of you - the OA SIG will be sponsoring a microvax with the NOTES facility for you to share ideas and ask questions of other OA users. (there will be no charge other than your phone costs to access the system) Sign-up information will be forthcoming.

We are well into our production of the newsletter with the DecPage format. I am still interested in your reactions...what do you think? Is the format more consistent and easy to follow, do you prefer it one way or the other.

Our featured technical article this month is about DecMate 'Secrets' for using fonts, command blocks and some other neat things. We also have our regular Notes on Notes column back. Watch the August issue for Digital's response to the TOP 50 SIRs presented to them last Fall; and a list of the new SIRs for you to vote on from Nashville.

For those of you who have submitted articles to me in a paper format, they will be in the September issue. We are still trying to work out a standard submission format.

Regards,

Therese LeBlanc

OA Newsletter Editor
275 London Place
Wheeling, IL 60090
(312) 459-1784

DECMATE SECRETS REVEALED

Bob Hassinger, Liberty Mutual Research Center, Hopkinton MA 01748

At the Nashville Symposium DEC made available some DECmate Technical Notes on undocumented and under documented features in DECmate WPS V2.1 and 2.2. There are features in the software that none of us knew about. Because these features give us tools to do things that there is no other way to accomplish in WPS it seems worthwhile to pass them on here.

I have adapted the following sections from the Technical Application Notes material so all of us can share in this very interesting information. The actual illustrations of results in the applications notes can not be reproduced here because they only work on DECmate WPS with output directly to an LN03. The Newsletter is produced with DECpage on a VAX. The information and examples included should allow you to try it for yourself however.

DEC has not included this information in the regular documentation because it is relatively complex and can only be used effectively by users who are familiar with the details of programming printers. **Note the disclaimers.** The Atlanta Hot Line *does not support* these features.

First, we have information on the complete list of command words. Many of these are included in the regular documentation but some are not. This is a convenient list that brings the information together in one place.

A LIST OF KEYWORDS USED FOR PRINTER APPLICATIONS

This application note lists the various keywords that are used in the control blocks for printing applications. The details on how to use most of these keywords are spelled out in the WPS documentation set. These keywords are understood by the printer software but not all printers can take advantage of all of the commands. For instance, trying to select a tray on a printer without a sheet feeder is meaningless and so is trying to select a color on a printer that only has one color.

DEC standards have defined a number of printer commands that fall into the category of "SGR's" (Select Graphic Rendition codes). The software has incorporated many of these commands even though the current printers may not support all of the features. This allows for the possibility of newer printers to be developed and the software will be ready to support some of the new features. Listed below are the currently defined Keywords and the function they perform:

TOP LEVEL COMMAND WORDS

PRINTER	Defines this to be a PRINTER Control Block
MULTI	Defines this to be a MULTICOLUMN Control Block
RESET	Defines this to be a RESET Command Control Block
SECTION	Defines this to be a SECTION Command Control Block
TOP	Defines this to be a HEADER (Key 1) Control Block
HEADER	Defines this to be a HEADER (Key 2) Control Block
BOTTOM	Defines this to be a BOTTOM (Footer) Control Block

KEYWORDS USED WITH THE PRINTER COMMAND WORD

SELECT	Sheet Feeder Tray Selection
CHRSET	Character Set Selection
FONT	Font Selection
BOLD	Select Attribute for BOLDED Text
UNDER	Select Attribute for UNDERLINED Text
FORMAT	Select LN03 Page Orientation
PITCH	Draft Printer Pitch Selection
COLOR	Select one of the COLOR SGR Sequences
TRANS_MIT	Transmit an ESCAPE Sequence to the Printer
GRAPH	Enter SIXEL Graphics Mode
COMMENT	COMMENT LINE - Ignore the Text on the Rest of Line

KEYWORDS USED WITH PRINTER SELECT COMMAND WORDS

LET	Selects the LETTER Tray of the ASF-02 Sheet Feeder
ENV	Selects the ENVELOPE Tray of the ASF-02 Sheet Feeder
REAR	Selects the REAR Tray of the ASF-01 or ASF-02 Sheet Feeder
FRONT	Selects the FRONT Tray of the ASF-01 or ASF-02 Sheet Feeder

KEYWORDS USED WITH PRINTER CHRSET AND PRINTER FONT COMMAND WORDS

NORMAL	Selects an Attribute for ALL of the Text
NONBOL	Selects an Attribute for ONLY UNBOLDED Text
BOLD	Selects an Attribute for ONLY BOLDED Text

KEYWORDS USED WITH PRINTER BOLD AND PRINTER UNDERLINE COMMAND WORDS

DOUBL	Select the DOUBLE-UNDERLINE SGR Sequence
STRIK	Select the STRIKE-THROUGH SGR Sequence
UNDER	Select the UNDERLINE SGR Sequence
ITALI	Select the ITALIC SGR Sequence
FAINT	Select the FAINT SGR Sequence
BOLD	Select the BOLD SGR Sequence

KEYWORDS USED WITH PRINTER FORMAT COMMAND WORDS

A	Select the "A" Paper Size
A4	Select the "A4" Paper Size
PORTR	Select PORTRAIT Page Orientation
LANDS	Select LANDSCAPE Page Orientation

KEYWORDS USED WITH PRINTER PITCH COMMAND WORDS

16	Print at a Pitch of 16.5
15	Print at a Pitch of 15
13	Print at a Pitch of 13.2
12	Print at a Pitch of 12
10	Print at a Pitch of 10
8	Print at a Pitch of 8 or 8.25
7	Print at a Pitch of 6.6
6	Print at a Pitch of 6
5	Print at a Pitch of 5

KEYWORDS USED WITH PRINTER COLOR COMMAND WORDS

TRANS	Select SGR 37 TRANSPARENT
CYAN	Select SGR 36 CYAN
MAGENTA	Select SGR 35 MAGENTA
BLUE	Select SGR 34 BLUE
YELLOW	Select SGR 33 YELLOW
GREEN	Select SGR 32 GREEN
RED	Select SGR 31 RED
BLACK	Select SGR 30 BLACK

KEYWORDS USED WITH PRINTER TRANS-MIT COMMAND WORDS

US	ASCII value of 037	CR	ASCII value of 015
RS	ASCII value of 036	FF	ASCII value of 014
GS	ASCII value of 035	VT	ASCII value of 013
FS	ASCII value of 034	LF	ASCII value of 012
ESC	ASCII value of 033	TAB	ASCII value of 011
SUB	ASCII value of 032	HT	ASCII value of 011
EM	ASCII value of 031	BS	ASCII value of 010
CAN	ASCII value of 030	BEL	ASCII value of 007
ETB	ASCII value of 027	ACK	ASCII value of 006
SYN	ASCII value of 026	ENQ	ASCII value of 005
NAK	ASCII value of 025	EOT	ASCII value of 004
DC4	ASCII value of 024	ETX	ASCII value of 003
DC2	ASCII value of 022	STX	ASCII value of 002
DLE	ASCII value of 020	SO	ASCII value of 001
SI	ASCII value of 017	NUL	ASCII value of 000
SO	ASCII value of 016		

SENDING ESCAPE SEQUENCES TO THE PRINTER

There is a technical feature in the Released Versions of DECmate/WPS Software that is not described in the standard documentation set. This feature will allow a user to send ESCAPE Sequences to a printer from within a document and is available in Versions 2.0, 2.1, and 2.2. If you believe that you have a printing problem that can only be solved by sending some selected escape sequences to the printer from within a document, then this feature may be for you.

Escape sequences can be sent to the printer from within a document by enclosing them within a "Control Block" and using some special keywords. The keyword for DECmate/WPS Version 2.0 is "XMT" and the keyword for Version 2.1 and 2.2 is "TRANS_MIT". These keywords operate in two modes, a numeric mode and a text mode. The "SLASH" character is used to switch between each mode. The syntax of the command is as follows:

```
------------------------------- START COMMAND -----------------------------
PRINTER  TRANS_MIT  000  001  NUL  SOH  /Any text characters/  STX  002  ...
-------------------------------  END   COMMAND -------------------------------
```

A hard RETURN or a SOFT-WRAP will terminate the transmit command. Multiple commandscanbeentered into one control block as follows:

```
---------------------------- START COMMAND ----------------------------
PRINTER
TRANS_MIT  000  001  002  003  004  . . . . .
TRANS_MIT  NUL  SOH  STX  EOT  ENQ  . . . . .
TRANS_MIT  /TEXT - any text characters/  . . . . .
----------------------------  END   COMMAND ------------------------
```

In numeric mode, the system looks for groups of three digit characters separated by a "SPACE" character. The system expects the number to be entered in the "OCTAL" numbering system and will translate the group of three digits into the octal character it represents and send it to the printer. The system can also translate the basic two or three letter sequences used in the lower part of the ASCII chart. Listed below are the sequences that can be translated and the OCTAL value they generate:

```
NUL = 000 SOH = 001 STX = 002 ETX = 003 EOT = 004 ENQ = 005 ACK = 006
BEL = 007 BS  = 010 HT  = 011 TAB = 011 LF  = 012 VT  = 013 FF  = 014
CR  = 015 SO  = 016 SI  = 017 DLE = 020 DC2 = 022 DC4 = 024 NAK = 025
SYN = 026 ETB = 027 CAN = 030 EM  = 031 SUB = 032 ESC = 033 FS  = 034
GS  = 035 RS  = 036 US  = 037
```

Listed below are the ASCII character codes and their OCTAL values:

```
040 = SPACE 041 = !  042 = "  043 = #  044 = $  045 = %  046 = &  047 = '
050 = (     051 = )  052 = *  053 = +  054 = ,  055 = -  056 = .  057 = /
060 = 0     061 = 1  062 = 2  063 = 3  064 = 4  065 = 5  066 = 6  067 = 7
070 = 8     071 = 9  072 = :  073 = ;  074 = <  075 = =  076 = >  077 = ?

100 = @     101 = A  102 = B  103 = C  104 = D  105 = E  106 = F  107 = G
110 = H     111 = I  112 = J  113 = K  114 = L  115 = M  116 = N  117 = O
120 = P     121 = Q  122 = R  123 = S  124 = T  125 = U  126 = V  127 = W
130 = X     131 = Y  132 = Z  133 = [  134 = \  135 = ]  136 = ^  137 = _

140 = `     141 = a  142 = b  143 = c  144 = d  145 = e  146 = f  147 = g
150 = h     151 = i  152 = j  153 = k  154 = l  155 = m  156 = n  157 = o
160 = p     161 = q  162 = r  163 = s  164 = t  165 = u  166 = v  167 = w
170 = x     171 = y  172 = z  173 = {  174 = |  175 = }  176 = ~
```

NOTE: This feature of the DECmate/WPS Software is a TECHNICAL feature that requires a Sophisticated and Knowledgeable individual to use it correctly. Sending escape sequences to the printer can change the way a document is printed and the result may NOT be what the user really wanted to obtain. Many of the escape sequences that the printers use work differently from printer to printer and the various escape sequences may interfere with each other and with the escape sequences sent to the printer from within the WPS software.

Printing applications using this feature are not supported by the Atlanta Hot Line. The level of support required to answer questions about this feature goes far beyond the technical charter of the Atlanta Hot Line group. If you use this feature, you may have to use a lot of trial and error effort to get the results you are trying to obtain. Some results may be impossible to obtain because of correcting escape sequences issued from within the WPS software. If you use this feature, you own any problems encountered in its use.

DRAWING LINES WITH THE LN03 PRINTER

AN EXAMPLE USING THE PRINTER TRANS-MIT COMMAND

There is an Escape Sequence described in the LN03 Programmer Reference Manual that will allow you to draw horizontal or vertical lines with length and width (DECVEC). The fact that this command draws lines without modifying the active printing position makes it particularly useful in DECmate applications. The syntax of the command is as follows:

```
ESC [ Pn1 ; Pn2 ; Pn3 ; Pn4 ; Pn5 ! |
Pn1 = Draw line, 0 = Horiz. line, 1 = Vert. line, Other = Ignore command
Pn2 = Selects the X start position - Default value is 0
Pn3 = Selects the Y start position - Default value is 0
Pn4 = Select the line length - Default value is 1 pixel
Pn5 = Select the line width  - Default value is 1 pixel

---------------------------------- START COMMAND----------------------------------
PRINTER TRANS_MIT ESC/[2 I/ESC/[0;200;200;5400;30!|/ESC/[0;200;7400;5400;30!|/
TRANS_MIT  ESC/[1;200;200;7200;30!|/ESC/[1;5600;200;7230;30!|/ESC/[7  I/
---------------------------------- END   COMMAND ----------------------------------
```

The above escape sequence when sent to an LN03 will cause it to print a nice border around your text. It is a set of four escape sequences that will draw the four sides of a box. The example above has had all of the spaces compressed out of the line so that more escape sequences can be placed on one line. In the next example the escape sequences have been separated so that only one escape sequence is on each line and spaces have been added for clarity. Line numbers have also been added for reference in the following text. The example (without the line numbers) will work correctly.

```
---------------------------- START COMMAND --------------------------------
(1) PRINTER
(2) COMMENT  -  This comment line is IGNORED in a printer control block
(3) TRANS_MIT  ESC  /[2  I/
(4) TRANS_MIT  ESC  /[0;200;200;5400;30!|/
(5) TRANS_MIT  ESC  /[0;200;7400;5400;30!|/
(6) TRANS_MIT  ESC  /[1;200;200;7200;30!|/
(7) TRANS_MIT  ESC  /[1;5600;200;7230;30!|/
(8) TRANS_MIT  ESC  /[7  I/
---------------------------- END   COMMAND --------------------------------
```

Line 1 The PRINTER keyword defines the function of this control block. It can stand alone or be combined on a single line with another command. It is only used once in the block.

Line 2 The COMMENT keyword allows for comments within a PRINTER control block. The purpose of the block can be described for later reference. The keyword must precede each line that is to be a comment.

Line 3 The TRANS_MIT keyword defines the function of sending escape sequences to the printer. This escape sequence is made up of the keyword ESC (which is equivalent to the Octal value of 33) followed by the text string "[2 I". The purpose of this escape sequence is to select the size unit to be used for drawing the line (SSU). The value of 2 means to select the Decipoint point size which is 720ths. of an inch.

Line 4 This escape sequence draws the top line.

Line 5 This escape sequence draws the bottom line.

Line 6 This escape sequence draws the left line.

Line 7 This escape sequence draws the right line.

Line 8 This escape sequence re-selects the proper size unit for the DECmate to use in continuing the print process. The value of 7 means to select the Pixel point size which is 300ths. of an inch.

With a moderate amount of work someone could draw some very complicated looking forms. The lines on the form could be drawn using the above escape sequences as examples and then the form could be labeled using the normal printing process. It is possible to construct a form document for list processing that will generate a printed form and then use the list processing field values to fill in the form.

DESIGNING A FORM WITH THE LN03 PRINTER

AN EXAMPLE USING THE PRINTER TRANS-MIT COMMAND

For this application we will design a simple form to contain some information about the employees of a company. The following information must be recorded: Badge Number, Name, Hire Date, Phone Number, Address, Job Code, City, State, and Zip Code. The first step was to identify what information must be on the form and the second step is to decide how big the form should be. This form will be used in a card file index box so it should fit on a 3" by 5" card.

The third step is to rough out a design on paper for the arrangement and placement of information on the form. Enough space must be allocated so that the information areas on the form can be labeled. Line width and spacing is important in deciding what has to go where. Also the way the form is to be filled out plays a part in determining line spacing. A form that will be filled in by hand needs more space than a form that will be filled in by a typewriter. This form will be typed and so we need to insure that the line spacing is 6 lines to the inch.

The forth step is to type the text that is to be on the form into a document. Using the techniques spelled out in the DECmate "Using the LN03" user guide, select the LN03 fonts to be used (either the built in fonts or the fonts from a font cartridge) and set up the tab positions in the document ruler. Print the document and begin the process of making adjustments in the printer font commands or the ruler settings until the text prints correctly as desired.

The fifth step is to use the LN03 to draw lines around the text to make a completed form. That is what this application note is all about. The LN03 Select Size Unit (SSU) command gives the user two options for specifying size units: Pixels which are in 300ths. of an inch or Decipoints which are in 700ths. of an inch. For this application, we choose the pixel size because it is divisible by 6 which we need for the 6 lines to the inch spacing. Since the DECmate uses the pixel size for printing there is no need to use the size unit selection.

Temporarily add the following control block to the start of the form document and print it again. This control block will cause two lines to be drawn on the form: one in the horizontal direction and one in the vertical direction. These lines are the reference points from which measurements can be made to determine the position of the actual lines of the form.

```
------------------------------ START COMMAND -----------------------------
PRINTER  TRANS_MIT  ESC  /[0;0;0;2400;3!|/  ESC  /[1;0;0;3200;3!|/
------------------------------- END   COMMAND -----------------------------
```

Manually draw lines with a pencil and ruler on the copy of the printed text of the form in the approximate locations that the lines will occupy on the completed form. Mark each line that is drawn on the form with a letter of the alphabet and record the following information for each line: Line Direction (either vertical or horizontal), Line Origin (the upper leftmost position of the line measured first in reference to the vertical line [the X value] and then the horizontal line [the Y value]), Line Length and Line Width (either thin, medium, or thick).

Use a calculator to convert the measurements to Pixel units (1 inch = 300, 1/2 inch = 150, 1/4 inch = 75, 1/8 inch = 38, 1/16 inch = 19). For our form we want to draw a medium line around the outside to use as a border. Since the form will be placed onto a 3 by 5 card, the borders should be 2 3/4 by 4 3/4 inches long and these line lengths work out to be 825 by 1445. Using the reference lines locate the origin point for the upper left corner of the form and calculate the pixel distance to the vertical line and then the pixel distance to the horizontal line. Use these two numbers to fill in the sequence to draw the top and left side of the form.

Every time you calculate the origin for a line you determine X and Y values that can be used for other lines. To draw the bottom line, use the same X value as the top line. The Y value is the sum of the top line Y value plus the length of the left side line minus the width of the bottom line. If we do not subtract the line width value, the line will be lower than we want it to be. To draw the right side line, find the X value by adding the top line X value plus the length of the top line minus the width of the right side line. The Y value is the same Y value as the top line.

Once you develop the knack for placing a line where you want it to be, form designing will become a snap. Designing a form can take a lot of time and paper so be patient. It is very important to place the text on the page first and then draw the lines around the text. Doing it the other way around is much more difficult. After the form is completed, remove the escape sequences that caused the reference lines to be drawn. Such a form could be used in List Processing to both draw the form and to fill it out. To do that, put the field names in the form text. The actual escape sequences used to draw a completed form for this example are shown below:

```
------------------------------ START COMMAND -----------------------------
PRINTER
COMMENT - Draw the Horizontal and Vertical reference lines.
TRANS_MIT  ESC /[0;0;0;2400;3!|/ ESC /[1;0;0;3200;3!|/
COMMENT - Draw the top, left, bottom, and right sides of the form.
```

```
TRANS_MIT  ESC /[0;480;340;1445;10!|/ ESC /[1;480;340;825;10!|/
TRANS_MIT  ESC /[0;480;1165;1445;10!|/ ESC /[1;1915;340;825;10!|/
COMMENT - Draw the four horizontal inside lines.
TRANS_MIT  ESC /[0;480;470;1445;5!|/ ESC /[0;480;620;1445;5!|/
TRANS_MIT  ESC /[0;480;760;1445;5!|/ ESC /[0;480;900;1445;5!|/
COMMENT - Draw the four vertical inside lines.
TRANS_MIT  ESC /[1;770;470;430;5!|/ ESC /[1;1680;470;150;5!|/
TRANS_MIT  ESC /[1;1530;760;140;5!|/ ESC /[1;1680;760;140;5!|/
------------------------------ END  COMMAND ----------------------------
```

DECMATE WORD PROCESSING TECHNICAL NOTEBOOK

At the Nashville Symposium I had a chance to see a preliminary draft of a new manual entitled Word Processing Technical Notebook. It contains a great deal of very technical information about DEC's WPS systems, particularly DECmate. Things like the details of the terminal emulations, disk and file formats and the DX formats and protocols. This information is generally quite hard to come by so anyone needing detailed technical information in the WPS area should consider ordering this manual. The draft I saw was dated June 1987 and the number was AA-J356C-TK.

****

Notes on Notes

- Discussions on VAX Notes - Volume 1, Number 7

by Mark H. Hyde and C.J. Trayser, Vax Notes Support Specialists, Digital Equipment Corporation

Whew, it's good to be home. But we really had a blast in Nashville, meeting many of you and talking about Notes as much as we could. Our first DECUS was an invaluable experience and we look forward to many more. In the rush to prepare our Notes presentations for the symposium the deadline for our article last month passed, so we'll try to get back on track. This time, let's talk a little about using VAX Notes in batch or in a command file via a spawned subprocess. What? I thought Notes was an interactive tool! Yes, Notes is primarily for interactive electronic conferencing, but there are several places where using Notes in batch or in a subprocess could come in handy. For example, suppose you've had Notes running for a while and you now have quite a few conferences in your notebook and you begin to get annoyed at how long it takes to UPDATE the entire notebook. Here is an excellent opportunity to use Notes in a subprocess. As part of your login process, you could spawn a very small command procedure (we'll call it NOTES_UPDATE.COM) that issues one of the DCL level Notes commands to update your notebook. The following command in your LOGIN.COM will do the trick:

```
$ SPAWN/NOWAIT/OUTPUT=NL: @NOTES_UPDATE
```

This command will fire off the following command file:

```
$! NOTES_UPDATE.COM
$! Command procedure to update entire VAX Notes notebook.
$! To be spawned as part of the login process.
$!
$  NOTES /UPDATE /CLASS=*
$!
$! Note that the command can be modified to only update a
$! specific class by modifying the /CLASS=* part of the command.
$!
$  EXIT
```

This command procedure could also be submitted as a batch job as part of your login process. Updating the Notebook is one of the most popular uses of Notes in batch or a subprocess.

The popular circumstance where you will find batch Notes handy is for following some types of conferences. If you have a lot of conferences in your notebook you (or your boss) may feel that you need to reduce the time spent Noting. Or, if you have one or more conferences that you read regularly but find that you do not contribute to very often, extracting the unseen notes in batch is a great way to follow them. A command procedure, submitted in batch, can open a conference and extract all the unseen notes to a file which you can then print and read, or read with an editor at your leisure. There are many different approaches to accomplishing this. We will present a fairly simple example and leave you to develop the more complex ones on your own, because each site will probably have different parameters to work within. As usual a picture is worth a thousand words, so here is a picture of a command procedure with lots of comments.

```
$! NOTES_EXTRACT.COM
$!
$! This command procedure will open a VAX Notes conference called
$! VMSNOTES that might exist in your notebook and extract all the
$! unseen notes in that conference to a text file.
$!
$! If your profile is set to do an automatic next unseen command
$! (this is the default) then the first note will wind up in
$! the batch log file.  So we disable any automatic stuff
$! with the /NOAUTO switch on the Notes command line.
$!
$! Perform an extract command that extracts all (/ALL) unseen
$! notes (/UNSEEN), marks the ones extracted as seen (/SEEN),
$! and put them in a file called VMSNOTES.EXTRACT
$!
$! CLOSE the conference and EXIT from Notes.
$!
$  notes /noauto VMSNOTES extract/seen/unseen/all VMSNOTES.EXTRACT
   close exit
$!
$! At this point, some people like to mail themselves the
$! extract file, others like to just mail a notification
$! that it is done, and some don't do anything.
$! We will mail ourselves the extract.
$!
$ mail VMSNOTES.EXTRACT HYDE /subject="VMSNOTES extract"
$!
$ exit
```

This procedure is pretty simple and is geared toward the extraction of a single conference. If you have more than one to follow then you could submit a procedure like that for each one, but that's not very practical. We will outline one approach for dealing with many conferences and let you develop it more fully for your own needs.

1. It can be useful to create a [.NOTES] subdirectory to keep all this stuff in.
2. Create a file to contain a list of the Notebook entries that you want extracted, (i.e. NOTES_EXTRACT.DATA). As you change your mind about the conferences that you want to extract, it will be easier to maintain this simple file, rather than a complex command procedure.
3. The command procedure will open NOTES_EXTRACT.DATA and loop through the following steps:
   a. read the first conference (entry) name from NOTES_EXTRACT.DATA.
   b. open the entry name with the /NOAUTO switch on the Notes command.
   c. EXTRACT/SEEN/UNSEEN the conference to a file.
   d. Mail the extracted results, or mail yourself a notice that extracts are done, etc.

A simple example might be:

```
$   open/read entrylist [.notes]notes_extract.data
$   next_conference:
$   read/end=clean entrylist next$entry
$   notes/update 'next$entry' /class=*
$   notes/noauto 'next$entry'
extract/seen/unseen notes_extract.tmp *.*
close
exit
$   mail notes_extract.tmp TRAYSER /sub="''next$entry' extract"
$   goto next_conference
$   clean:
$   delete notes_extract.tmp;*
```

This basic process can be expanded and bug-proofed in a number of ways.

Another use for Notes in batch is searching for information. There are the various forms for DIRECTORY command to look fir notes by title strings, author names, keywords, etc. Instead of taking up interactive terminal time waiting for the directory commands to complete, you can submit them in a batch job. If you use the /NOTIFY qualifier on the SUBMIT command, you will know when the job is finished and can scan the log file for the results.

```
$! TITLE_SEARCH.COM
$!
$! Search the FY87_FINANCE conference for all notes
$! that might pertain to the budget process.
$!
$! submit/noprint/notify TITLE_SEARCH.COM
$!
$  Notes /noauto FY87_FINANCE DIR /ALL /TITLE="budget" 1-last exit
$!
$  exit
```

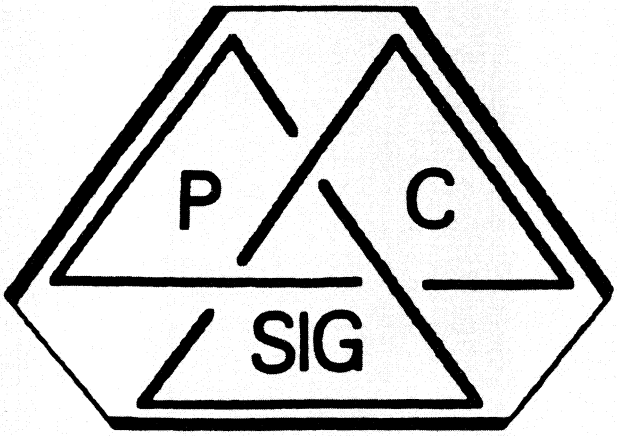
TITLE_SEARCH.LOG will contain a list of all the notes with titles
that contain the string "budget".

There is much more that Notes can do with sub processes, such as keeping it in a subprocess and attaching to it when we want it. But that as well as other topics will have to wait for a future article.

Happy Noting :^)

Personal Computer Special Interest Group

PC



Newsletter

Table of Contents

DECUS PC SIG Newsletter Volume 2, Number 11

Two Low-Cost Optimizer Options

After deciding that your disk needs to be optimized, read this article to find out about a couple of public domain programs to optimize your disk and save you money.

DEC Reportedly Dumps MS-DOS V3.1 for the Rainbow

By now, you already know that Digital has made their official announcement of the end of the Rainbow... Now, they've also announced the cancellation of the Network Integration Kit that was promised last year.

Joint Request for Windows and MS-DOS V3.1

Both the U.S. DECUS PC Sig and the European DECUS PC Sig are disappointed with Digital's decision to scrap the Network Integration Kit. In response to that announcement, the two Sigs have joined forces to ask Digital to deliver the parts of the Integration Kit that could still be useful to the Rainbow user.

Update for Version 3.1 of P/OS and More

Tom Hintz, the PRO Working Group chairman has offered to distribute the P/OS programs DEC donated to DECUS in Nashville.

Hard Disks and Controller Boards for the PRO

If you are at all interested in adding more disk storage to your PROfessional computer, this will be a valuable article for you.

Tailoring P/OS to Save Disk Space

The other solution to disk space problems is to reduce the amount of space being used. Perhaps this article will be of some use to you.

Welcome New Volunteers!

The PC SIG would like to welcome the following new volunteers. We are happy to have each of you as a contributing member of the SIG and look forward to getting better acquainted. Anyone wishing to become a member of a working group or help in Anaheim, please call Pierre Hahn; volunteer coordinator.

Stuart Labovitz
USAF
(513)255-7680

Larry L. Alber
F.D.A.
(312)353-9766

Mark Ciampa
Volunteer State Community College
(615)452-8600

Roger Davis
Alexandria Daily Town Talk
(318)487-6303

C. M. Devine
Federal Home Loan Bank Board
(202)377-6025
VAXmates

Kalpen Shah
CIBA-GEIGY
(201)277-5361
VAXmates

Fred Jerger
EMS
(414)359-9800 ext.228
VAXmates

Aaron Phelps

Nancy J. Fallon
Brookhaven National Lab
(516)282-4530 or 4207
Data bases, LAN

Jan C. Ostendarp
Massachusetts Financial Service
(617)536-3640
VAXmate Software

Robert L. Hardester
Ralston Purina
(314)425-9548
DECmate, WPSPLUS, All-in-1

Christine Iwanowski
Lever Research Inc.
(201)943-7100 ext 2214
DECmate Working Group

Kathy Winston
U.S.D.A.
(816)926-6336
IBM to DEC

Robert Ingram
Sandia National Labs
(505)846-9717
IBM PC & MicroVAX

Angela B. Drago
O.H. Materials Corp.
(419)423-3526
PCs, Networks, OA

John M. Benda
The BOC Group, Inc.
(201)771-1856
VAXmates, OA & PCs

Velitc Williams
Federal Home Loan Bank Board
(202)377-6175
VAXmates

Mark Kalicki
Personal Products Co.
(201)524-7607
Rainbows, VAXmates

(202)778-2563
VAXmates

Bob Andrus
Linear Corp.
(619)438-7055
VAXmates

Phillip Perry
U.S. Treasury
(202)633-7921
WPSPLUS, PCs, Workstations

Jack L. Best
U.S. Treasury
(202)566-7963
WPSPLUS,VAXmates, Rainbows,DECmates

Rudie Schughter
U.S. Treasury
(202)566-9355
Rainbows

Dave Gudenicz
Abbott Labs
(312)937-8227
VAXmates, Rainbows

Robert Nevins
Millsaps College
(601)354-5201
Rainbows, Apple II (?)

Ted V. Stelling
Computer Applications Corp.
(312)789-8681
Rainbows

Joseph M. Daylor
Computer Applications Corp.
(312)789-8681
Rainbows

Brenda Tucker
Digital Equipment Corporation
(617)467-4051
Rainbows

Try a Few New Newsletters

Reprinted from the Denver Rainbow User's Group Newsletter

The Silicon Valley DEC PC Users Group
Newsletter Editor: Carl Neiburger, 408-294-6497 (h)
Membership: Dennis Boodt 408-756-9866 (w) 415-853-1562 (h)

The Denver Rainbow Users' Group (DRUG)
Treasure: John C. Forster 303-526-0088

The digital:Bellevue Users' Group (d:BUG)
Newsletter Editor: Otto Pearson 206-931-3258 (w), 206-226-7379 (h)
Membership: Otto Pearson

The University of Pennsylvania, DEC Rainbow Users' Group
Newsletter Editor: Chad Graham 215-898-8509 (w) 215-896-7053 (h)

WARUG's newsletter has evolved into a *de facto* national newsletter for Rainbow users, called *Rainbow News*. Subscriptions are $20 a year from IRUG, Box 1940, Vienna, Va. 22180.

Ted Needleman, who writes the "Rainbow Corner" column for *Hardcopy* magazine, has started his own national newsletter for DEC microcomputer users. Subscriptions are $36 a year. Write to *DEC MicroLetter*, Box J, New City, N.Y. 10956.

Those wishing sample copies of the *Rainbow News* and Ted Needleman's new *DEC MicroLetter* can get them from **Suitable Solutions** (408)-725-8944.

Rainbow Chair's Column

Lynn Jarrett, Vice Chair and Rainbow Working Group Chair

Well, we all survived another DECUS. It was interesting and it was informative, as usual, although I believe this DECUS was a historic one for Rainbow owners.

Ron Gemma, Digital's Rainbow Product Manager, was the bearer of bad news when he spoke at the Rainbow Working Group session in Nashville. The bad news he carried was two-fold. He announced that Digital would no longer be taking Rainbow orders after June 30, and shipping of Rainbows would cease as of December. He also announced the cancellation of the Rainbow Integration Kit, that which would give Rainbow users Windows along with MS-DOS 3.1 and allow the Rainbows to be networked to VAX's.

Digital will make parts available through Field Service for maintenance contracts to repair the Rainbows for the next several years. Software may still be purchased through DEC Direct, but there are certainly fewer packages to be obtained from Digital for the Rainbows than ever before.

There's still a lot of development going on through third party vendors and we hope to see a continuance of this. I will be publishing the Rainbow Wish List again next month with a few changes in the hopes that the third party vendors are still looking at it.

Through the joint efforts of this SIG and the European DECUS PC SIG, we are hoping to convince Digital to release MS Windows and MS-DOS 3.1 to the users since they don't intend to market these products.

VAXmate MS-DOS V3.10

Mitch Lichtenberg, DEC, Personal Computer Systems Group (PCSG)

The following pages have been compiled from a set of session notes given to the Sig by Mitch Lichtenberg. Mitch gave an excellent session in Nashville about the VAXmate MS-DOS operating system. Unfortunately, however, his session notes didn't make it into the session note packet. Therefore, we are reprinting them here. If you have a tape of his session, turn it on and follow along with these notes, here.

MS-DOS: The VAXmate's Operating System

- The VAXmate uses MS-DOS v3.10
  - RD31 (20MB) support
  - RX33 support
    - Read/Write 1.2MB diskettes
    - Read/Write Rainbow RX50 diskettes
    - Read 360K diskettes
  - Extended memory support
  - International character sets, keyboards
  - Network support
- Rainbow uses MS-DOS v2.11

MS-DOS 3.10 from the viewpoint of a VAX/VMS user

- MS-DOS is a single-tasking operating system
- Hierarchical file structure
- All files are "stream" files
- Rudimentary batch commands
- Device drivers defined in CONFIG.SYS
- System startup file in AUTOEXEC.BAT
- Does not support "true" record-locking
- Size of MS-DOS is very small (under 60K bytes)

MS-DOS Components

- Boot Block
- IO.SYS file on boot device
- MSDOS.SYS file on boot device
- COMMAND.COM file on boot device
- Utility programs

MS-DOS Components: Boot Block

First part of MS-DOS to be loaded

Located on first sector of boot device
(for floppies, this is track 0, sector 1)

Contains a parameter table for the boot device called the BIOS Parameter Block, which describes the shape of the disk.

Contains a seal to identify it as an MS-DOS boot sector

Contains an OEM name to identify the version of MS-DOS that formatted the disk

Contains the code to read in the next MS-DOS component, the file IO.SYS

MS-DOS Components: IO.SYS

First file in root directory

Must be contiguous

Contains machine-specific code

Contains device drivers for:
console (CON)
RS-232 (AUX)
Printer (PRN)
Real-time clock (CLOCK$)
Disk device drivers

Initializes MS-DOS parameters

Loads MS-DOS.SYS in high memory

Using DOS to access device files, loads drivers right after IO.SYS

Initializes DOS buffers and caches

Relocates MS-DOS.SYS to be right after IO.SYS and device drivers

Loads and executes COMMAND.COM

Many parts of IO.SYS are "throw-away" code used only for system initialization

MS-DOS Components: MSDOS.SYS

Must be second file in root directory

Not necessary for it to be contiguous

Contains machine-independent code, such as:

File system

Memory manager

Console interface routines

Dispatcher

Hooks for network

MS-DOS Components
COMMAND.COM

COMMAND.COM is MS-DOS's command-line interpereter program (CLI)

Reads commands from keyboard

Executes built-in commands, such as DIR, DELETE, COPY, etc.

Built-in commands are part of COMMAND.COM

Initiates execution of external programs

Searches along PATH variable for external program files

Executes AUTOEXEC.BAT on first invocation

Handles I/O redirection and pipes

Special file extensions

The following special file extensions are handled by COMMAND.COM

BAT - "Batch" file (similar to DCL "COM" files)

EXE - Executable image with relocation table

COM - Executable image without relocation table (program size limited to 64K)

MS-DOS Components
Utility Programs

Some of the MS-DOS utilities include:

FORMAT.EXE - prepares blank disks

EDLIN.EXE - a simple line-oriented text editor

FONT.COM - loads display fonts

MDRIVE.SYS - RAM disk device driver

FDISK.EXE - Partitions fixed disks into logical DOS drives

MODE.EXE - Controls screen and printer device drivers

SYS.EXE - Transfers MS-DOS system to another disk

DISKCOPY.EXE - Copies entire disks, track-by-track

CHKDSK.EXE - Checks & repairs file systems

MS-DOS Utilities

DEC Modified MS-DOS Components

BACKUP RESTORE	MS-DOS backup program. BACKUP was modified to allow backups to network drives.
DISKCOPY	MS-DOS disk-to-disk copy program. DISKCOPY was modified to allow all RX33 supported media to be copied.
FORMAT	MS-DOS disk initialization routine. FORMAT reads the factory bad-block table off DEC RDxx winchesters to lock out bad sectors.
SORT	MS-DOS sorting filter. SORT now has international features for modifying the sort order table depending on which country files are loaded.

DEC-Only MS-DOS Utilities

FDISK	Fixed Disk Utility program. Used for partitioning disks.
FONT	Loads text-mode font files into video character generator
GRAFTABL	Loads graphics-mode font files into BIOS tables
KEYB	Loads keyboard mapping files
LCOUNTRY	Loads all files for a particular country
MDRIVE	Memory-based disk emulator program

VAXmate MS-DOS supports the Industry-Standard, MCS, ISO, and NRC fonts.

VAXmate System Startup

80286 Processor begins execution in ROM at 0FFFF0H

Power-up self-test executes

Memory is sized, and various system configuration data structures are set up

Hardware subsystems are initialized

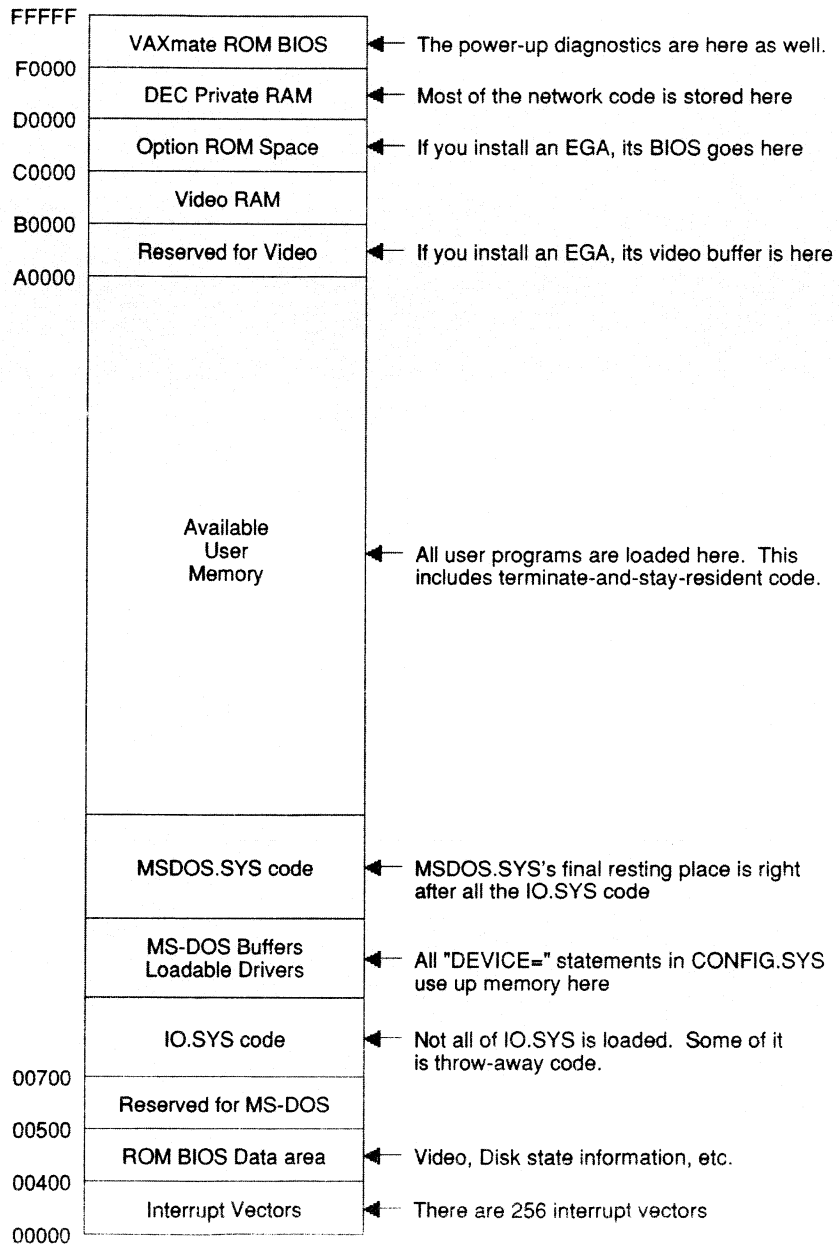
Control is passed to the ROM BIOS

ROM BIOS sets up software interrupt vectors

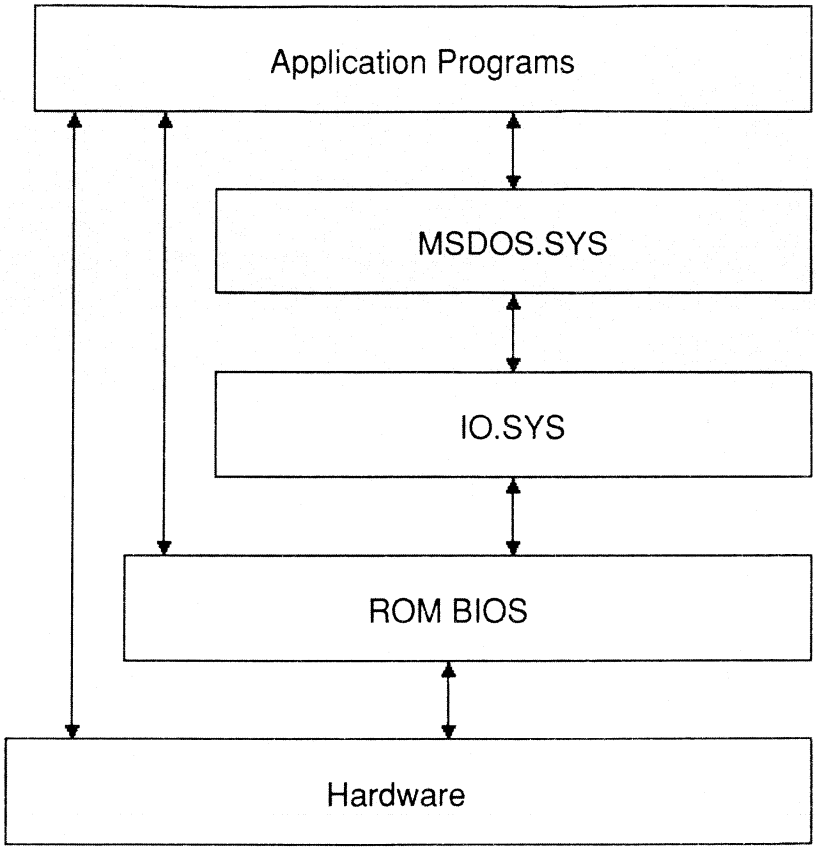
"INT 19" software interrupt starts the boot process

System tries different boot devices in this order:
Diskette drive "A"
Hard disk drive "C"
Network

VAXmate System Memory Map



Hierarchy of MS-DOS Modules

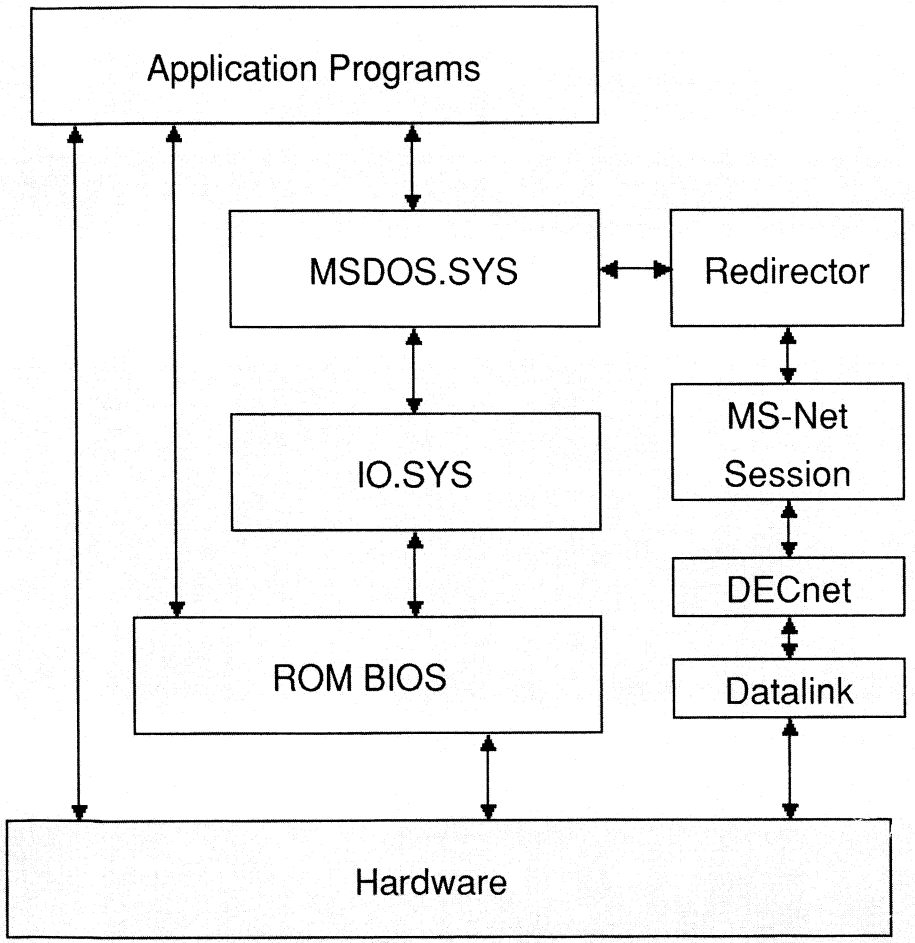


Most MS-DOS applications fall into two categories:

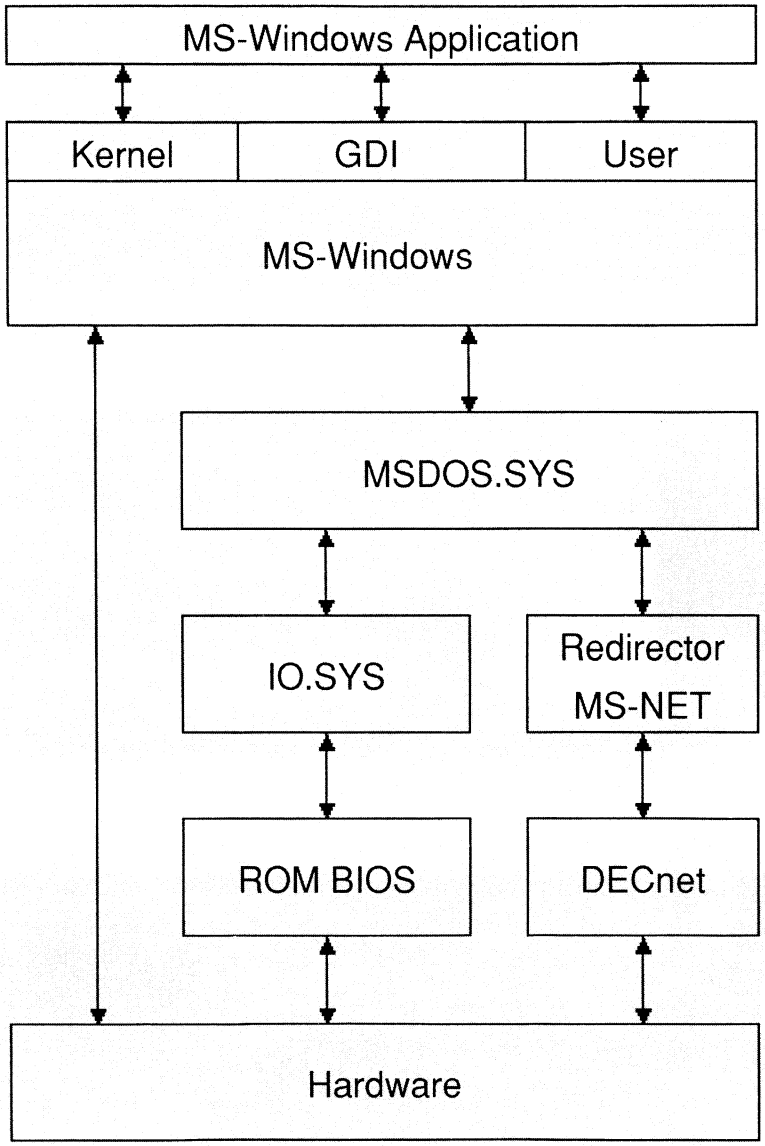
"Well-Behaved" applications use only MS-DOS and the ROM BIOS for their I/O.

Other applications access screen and keyboard hardware directly, mostly to use hardware features not supported by the BIOS, or for increased execution speed.

Hierarchy of MS-DOS Modules With network software loaded

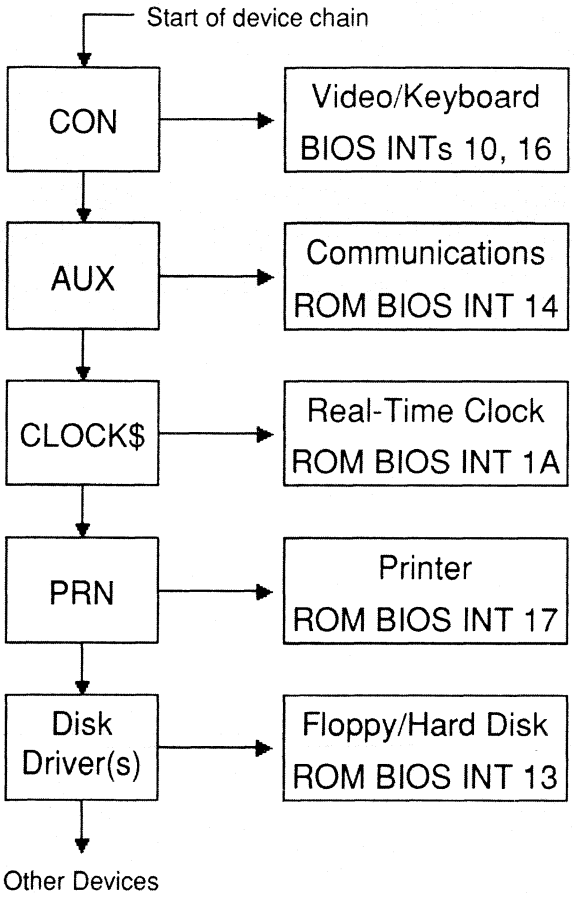


MS-Windows and MSDOS



MS-DOS Device Drivers

MS-DOS has at least five device drivers:



- Block device drivers do not have names. They are assigned to MS-DOS drive "letters" in the order they are initialized
- Device drivers are collected in a chain managed by MS-DOS. Additional drivers (those that are not in IO.SYS) are added through "DEVICE=" statements in the CONFIG.SYS file

MS-DOS Device Drivers

Character Devices

Serial (stream-oriented) devices
Are named like files
To access device, open devices name as a file
The console (CON), printer (PRN) and clock (CLOCK$) are character devices

Block Devices

Are used to construct MS-DOS file systems
Random-access devices such as Disk Drives
Must understand BIOS parameter blocks
Transfers data in blocks (logical disk sectors)
One block device driver may define multiple units (this is how VAXmate disk partitioning is done)

VAXmate disk BIOS

- Accesses hardware through ROM BIOS INT 13
- Basic functions: Read, Write, Verify, Format, Status
- Supports two fixed disks and two floppy disks

  Fixed disks may have up to four partitions

  DEC fixed disks supported: RD31, RD32

  Floppy disk drive supported: RX33

  Can read/write RX33 media, read/write RX50, and read RX31 (360K) media.
- DEC Extended functionality:

  Function 0xD0: Read 256 byte sector
  (used to read factory bad block table in FORMAT)

VAXmate Clock BIOS

- Accessed through ROM BIOS INT 1A
- Clock is implemented using a timer tick interrupt in INT 8 (IRQ0) at about 18.2 Hz
- INT8 routine also calls INT 1C so user may attach special timer-tick routines
- ROM BIOS INT 1A allows user to set/get time (seconds elapsed since midnight)
- VAXmate Clock driver uses INT 1A to maintain current time and date

VAXmate Printer BIOS

- Accesses hardware through ROM BIOS INT 17
- Supports up to 4 printers
- Output, initialize, and status functions
- DEC Extended features:
  - Redirect parallel printer to serial printer (0D0H)
  - Set/Get Printer Type (0D1H)
  - Set retries for parallel printers (0D2H)
- DEC Printers Supported: LA75, LN03+

VAXmate Serial Port BIOS

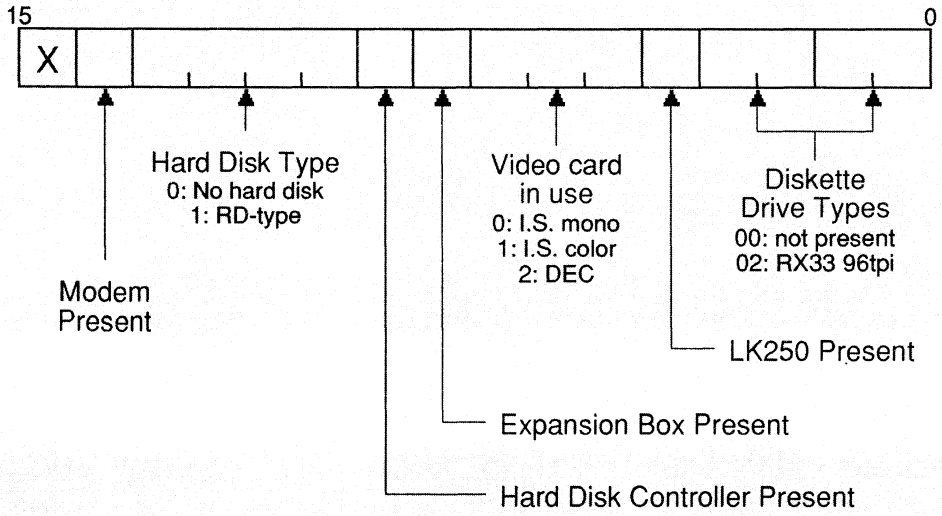
- Accesses hardware through ROM BIOS INT 14
- Initialize, send, receive, status functions
- Supports up to two comm ports plus the internal serial printer port
- DEC Extended functions:
  - Buffered serial I/O with flow control (0D0H)
  - BREAK signal control (0D1H)
  - Modem signal control (0D2H)
  - Retry-on-timeout control (0D3H)
  - Extended baud rates and split baud rates (0D4H)

VAXmate Video BIOS

- Text modes supported:
  - 40x25 monochrome, color
  - 80x25 monochrome, color
- Industry-Standard Graphics modes supported:
  - 320x200 monochrome
  - 320x200x4 color
  - 640x200 monochrome
- DEC Extended Graphics modes:
  - 640x400x2 color
  - 640x400x4 color
  - 800x252x4 color
- Reads, writes screen memory, controls cursor, scrolls screen, sets graphics modes.
- Loadable character sets in both text and graphics modes
- DEC Extended functions:
  - Enable/Disable graphics fonts
  - Load font RAM
- Screen-saver feature

DIGITAL Configuration Word

The DIGITAL configuration word contains information about the VAXmate that is not available in the industry-standard configuration word. The I.S. configuration is accessed via INT 11H, and the DEC configuration is accessed via INT 15H, function 0D0H



Differences between DEC MS-DOS and other MS-DOS implementations

- MS-DOS Calls implemented via INT 21H
- Function 30H: Get Version & OEM number

  DEC MS-DOS returns OEM number 16H
- Function 38H: Get/Set Country information

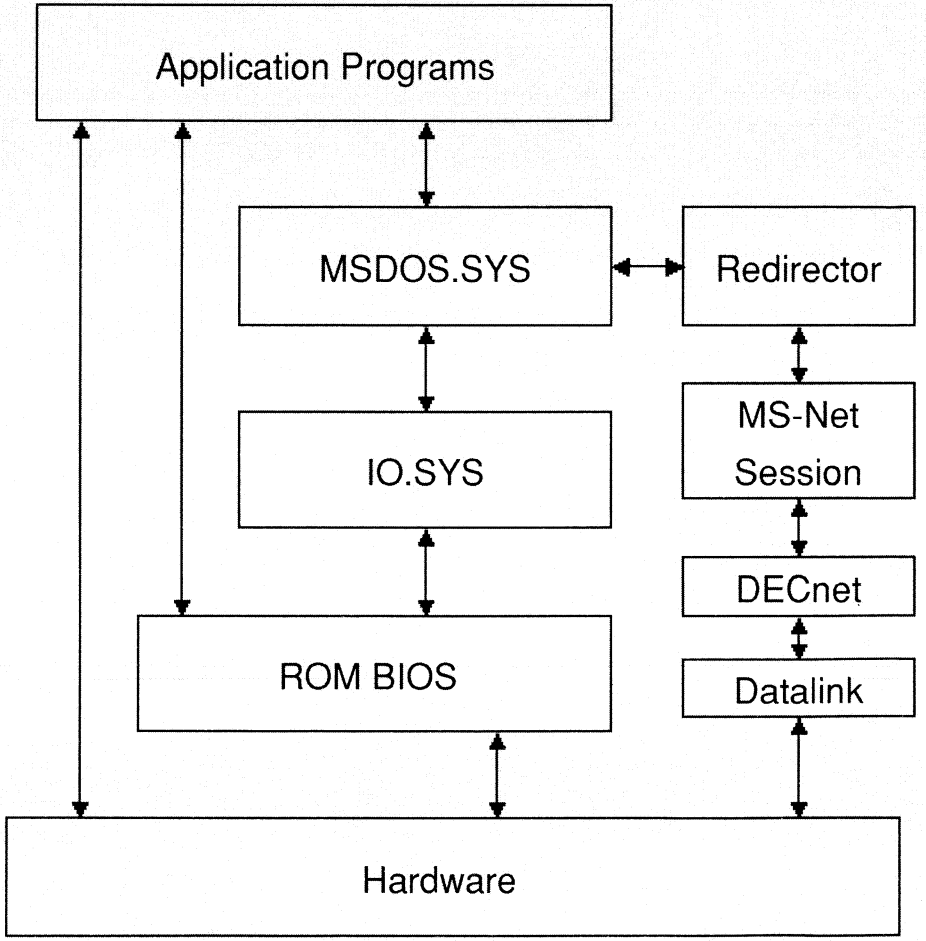
  More country codes, support for keyboard and display font switching

  Sets/Gets default keyboard map and font files

Font File Format

Position	Contents
00-03	Contains the string "F0$N" to identify file as a DEC font file
04	Number of bytes per character
05	Number of columns per character
06	Number of rows per character
07-08	Total number of characters in the file
09-10	Starting character location to load
11-12	Ending character location to load
13-131	Reserved
132-4228	Character bitmaps

Hierarchy of MS-DOS Modules With network software loaded



80286 Real-Mode Register Layout

AX	BP
BX	SI
CX	DI
DX	SP

General Purpose Registers

CS
DS
ES
SS

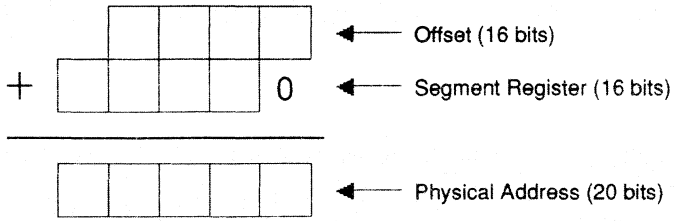
Segment Registers

The 80286 in "real mode" is essentially just a fast 8086. "Protected mode" is not used by MS-DOS ver 3.X

IP
Flags

System Registers

Real-Mode Memory Addressing



A physical address is calculated by adding the offset of the object you wish to access to the segment register value shifted left four bits. Segment boundaries are often called "paragraphs."

MS-DOS System Calls

MS-DOS System calls are divided into four categories:

- Character Device I/O Calls

  Read/write to console, keyboard status, auxillary (serial) port, printer port

- "FCB" file management calls

  Old-style (DOS 1.x) file calls (open,close,read, write,etc.) using File Control Blocks, similar to those used in CP/M.
  These calls are obsolete, but retained for backward compatibility.

- File management calls

  Provides file management calls similar to UNIX and XENIX.
  File-handle oriented (no control blocks to manage)

- Memory Management calls

  Allocate, Deallocate memory
  Adjust size of allocated memory blocks

- Miscellaneous calls

  Program terminate, Get Time, etc.

Making MS-DOS System Calls

Set up registers, then issue software interrupt 21H. On return, the carry flag will be set if there were errors.

Register	Contents
AH	MS-DOS Function number to call
AL	Sub-Function number
DS:DX	Input buffer address
ES:BX	Output buffer address
BX	File handle
CX	Number of bytes to transfer

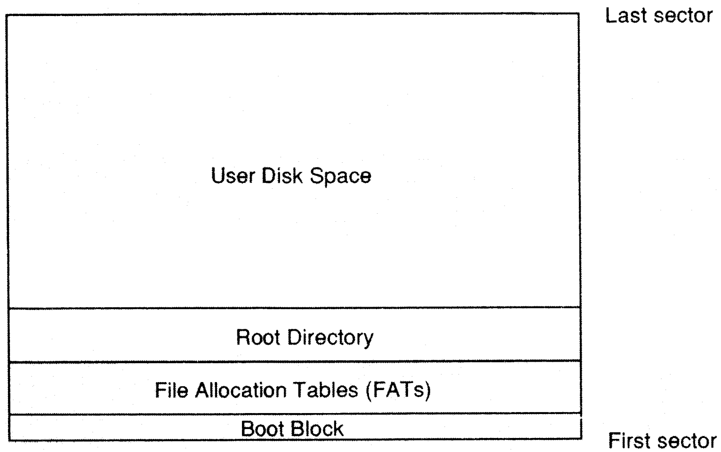
Not all of the MS-DOS functions use all of the registers. This is only the general rule.

VAXmate Keyboard BIOS

- Accesses keyboard via ROM BIOS INT 16H
- Read keyboard, status functions
- DEC Extended features:

  Install/Deinstall keyboard notification routines
  Return number of characters in keyboard buffer
  Install extended keyboard buffer
  Enable LK250 extended functions
  Request keyboard ID (revision, etc.)
  Send command to keyboard
  Change keyboard mapping tables

MS-DOS Disk Structure



The Boot Block contains the BIOS Parameter Block (BPB) which describes the shape of the disk to MS-DOS

MS-DOS allocates disk space in clusters, which is a number of disk sectors stored in the BPB. Floppy disks are typically one sector per cluster, and hard disks are usually 4 sectors per cluster.

The File Allocation Table (FAT) contains the file system structure and the in-use/available status for each cluster of the file
There may be multiple FATs for increased reliability.

FATs are indexed by cluster number, and each FAT entry is the cluster number of the next cluster in a file, or 0FFFFH for the EOF.

Zero FAT entries denote free disk space, an entry of 0FFF8 indicates a bad spot on the disk.

The root directory is a fixed size. Subdirectories are files with the "directory" attribute, so they may be any size.

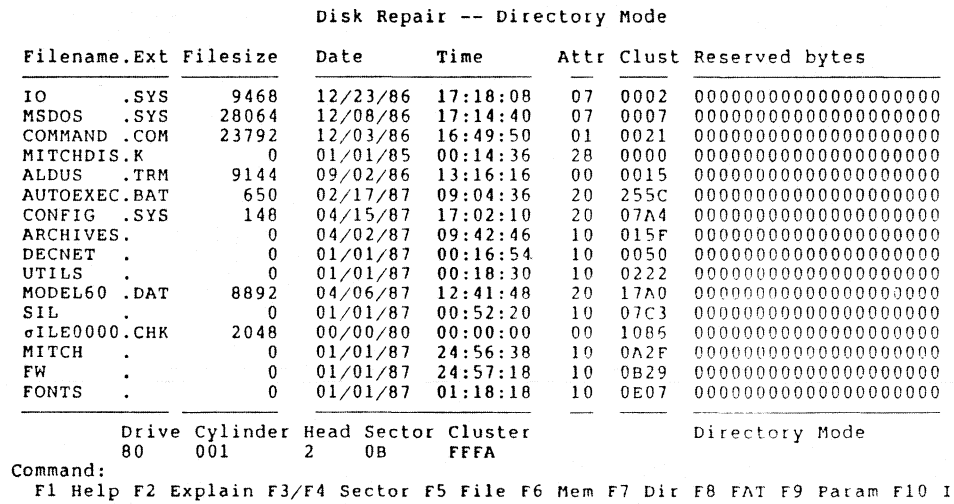
MS-DOS Disk Structure

File Allocation Table

```
                  Disk Repair -- File Allocation Table Mode
Cluster                         +4                       +8
-------    ---------------------------------------------------------------------
0000       FFF8 FFFF 0003 0004   0005 0006 FFFF 0008   0009 000A 000B 000C
000C       000D 000E 000F 0010   0011 0012 0013 0014   FFFF 0016 0017 0018
0018       0019 FFFF 001F FFFF   FFFF FFFF FFFF 0020   002D 0022 0023 0024
0024       0025 0026 0027 0028   0029 002A 002B 002C   FFFF 0035 002F 0030
0030       0031 FFFF FFFF 0044   0037 0036 0043 0038   0039 003A 003B 003C
003C       003D 003E 003F 0040   0041 0042 0049 0047   0045 FFFF 01C2 0048
0048       00D0 00C2 FFFF FFFF   0060 FFFF 004F 0051   FFFF FFFF 0053 FFFF
0054       FFFF 0056 FFFF FFFF   0059 005A FFFF 005C   FFFF FFFF 005F FFFF
0060       0061 0068 FFFF FFFF   FFFF FFFF FFFF FFFF   0069 006A 006B 006C
006C       006D 006E 006F 0070   FFFF 0072 0073 0074   0075 0076 FFFF FFFF
0078       FFFF 007A 007B 007C   007D FFFF FFFF FFFF   0081 0082 0083 0084
0084       0085 0086 0087 0088   0089 008A 008B 008C   FFFF FFFF 008F 0090
0090       0091 0092 0093 0094   0095 0096 00A0 0098   0099 009A 009B 009C
009C       009D 009E 009F FFFF   00A1 00A2 00A3 00A4   00A5 00A6 00A7 00A8
00A8       00A9 00AA 00AB 00AC   00AD 00AE 00AF 00B0   00C3 00B2 00B3 00B4
00B4       00B5 00B6 00B7 00B8   00B9 00BA FFFF 00BC   00BD 00BE 00BF 00C0
-------    ---------------------------------------------------------------------
           FAT Clu FAT Copy  Drive Cylinder Head Sector   F A T Mode
           0000    0         80    000      1    03
Command:
  F1 Help F2 Explain F3/F4 Sector F5 File F6 Mem F7 Dir F8 FAT F9 Param F10 INT
```

MS-DOS Disk Structure

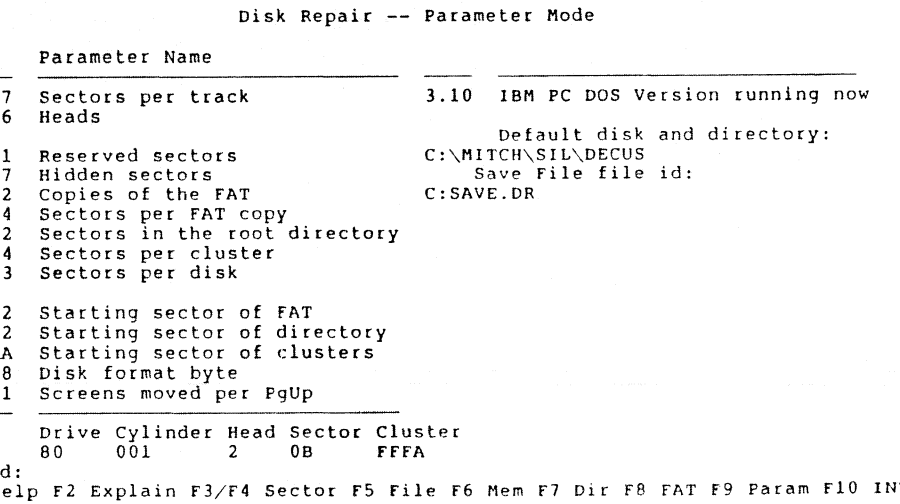
Directory Structure



```
                    Disk Repair -- Directory Mode

Filename.Ext  Filesize   Date       Time       Attr Clust Reserved bytes
IO      .SYS      9468   12/23/86   17:18:08   07   0002  0000000000000000000
MSDOS   .SYS     28064   12/08/86   17:14:40   07   0007  0000000000000000000
COMMAND .COM     23792   12/03/86   16:49:50   01   0021  0000000000000000000
MITCHDIS.K           0   01/01/85   00:14:36   28   0000  0000000000000000000
ALDUS   .TRM      9144   09/02/86   13:16:16   00   0015  0000000000000000000
AUTOEXEC.BAT       650   02/17/87   09:04:36   20   255C  0000000000000000000
CONFIG  .SYS       148   04/15/87   17:02:10   20   07A4  0000000000000000000
ARCHIVES.            0   04/02/87   09:42:46   10   015F  0000000000000000000
DECNET  .            0   01/01/87   00:16:54   10   0050  0000000000000000000
UTILS   .            0   01/01/87   00:18:30   10   0222  0000000000000000000
MODEL60 .DAT      8892   04/06/87   12:41:48   20   17A0  0000000000000000000
SIL     .            0   01/01/87   00:52:20   10   07C3  0000000000000000000
σILE0000.CHK      2048   00/00/80   00:00:00   00   1086  0000000000000000000
MITCH   .            0   01/01/87   24:56:38   10   0A2F  0000000000000000000
FW      .            0   01/01/87   24:57:18   10   0B29  0000000000000000000
FONTS   .            0   01/01/87   01:18:18   10   0E07  0000000000000000000

        Drive Cylinder Head Sector Cluster          Directory Mode
        80    001      2    0B     FFFA
Command:
  F1 Help F2 Explain F3/F4 Sector F5 File F6 Mem F7 Dir F8 FAT F9 Param F10 INT
```


MS-DOS Disk Structure

BIOS Parameter Block



```
                    Disk Repair -- Parameter Mode

      Parameter Name                          3.10  IBM PC DOS Version running now
 17   Sectors per track
  6   Heads                                       Default disk and directory:
                                           C:\MITCH\SIL\DECUS
  1   Reserved sectors                          Save File file id:
 17   Hidden sectors                         C:SAVE.DR
  2   Copies of the FAT
 64   Sectors per FAT copy
 32   Sectors in the root directory
  4   Sectors per cluster
65263 Sectors per disk

0012  Starting sector of FAT
0092  Starting sector of directory
00AA  Starting sector of clusters
 F8   Disk format byte
  1   Screens moved per PgUp

      Drive Cylinder Head Sector Cluster
      80    001      2    0B     FFFA
Command:
  F1 Help F2 Explain F3/F4 Sector F5 File F6 Mem F7 Dir F8 FAT F9 Param F10 INT
```

The New US Robotics Courier HST Modem

Jim Turner

The modem arrived and it has been installed on my Rainbow. The modem is especially suited for the bulletin board world. It will accommodate all the customary baud rates of 300, 1200, and 2400 plus it will also do 9600 baud. The lower three baud rates work well because of the established standards at those speeds. Hayes and US Robotics have been major players in setting the standards in the dial up modem industry. Many other modem manufactures have followed their lead.

Up until now the only standard, if you will, for the higher baud rate of 9600 was a standard called V.32 set by CCITT, the communications protocol standards committee. But, there are two draw backs. The modems that used the V.32 standard are both very expensive and have no lower speeds for the BBS world within the same modem.

Successful communications over the dial up network at the higher baud rates (4800 baud plus) requires some rather sophisticated use to technologies like "echo cancellation", "Trellis Coded Modulation (TCM)", "multiple carriers", "adaptive equalization", "full duplex", half duplex", "asynchronous" and "synchronous". USR decided to create a multi-speed modem including their own proprietary protocol made up of TCM and their own error control algorithms, which they have called High Speed Technology (HST). To communicate at 9600 baud with their modem you will have call another USR HST modem I have been told. USR has decided to offer these modems to the system operators of bulletin boards around the country at a special price hoping to establish their own standard for the 9600 baud rate. Otherwise the modem will talk to the other modems at one of the slower speeds. If you are a little skeptical about USR's capability to pull of this standards quest then you may want to wait it out a while longer. I took the plunge and bought one.

Sysops can order the modem directly through USR's modem numbers 312- 982-5092 or 312-982-5274. They will call your bulletin board to verify that you are a sysop then send you the modem. They will call you board a couple of weeks later to establish a 9600 baud connection. The rest of the modem users here in the Denver area can purchase one through R-Squared, Inc. USR's suggested list is $995, but R-Squared has been known to cut that price by 20% so if your interested try making a low ball offer for the modem and see if they accept. With the HST, when the modems connect the HST will take a few extra seconds to figure out the highest communications baud rate then it will tell the user what that baud rate is and turn control of the terminal/computer back over to the user. Some details about the HST. It is physically larger than other modems. It's foot print is 3/4 sq. ft. It requires a RS-232 shielded cable, pins 1-8,12,20,22 connected straight through, with male/female connectors to connect it up to the Rainbow. The female connector's hood should be flush mounted if possible. It is fully Hayes compatible, plus offers an extensive extended command set that offers a host of other capabilities.

Four features that I like are. 1) Not sure what the command is to do something, just type a simple command and the modem will send to your screen several help screens full of available commands. 2) Want to know how long that last call was, just ask the modem to tell you. 3) The modem gives call progress information on the screen, telling you things like it is 'dialing', phone number is 'busy', a voice answered, no answer, etc. and 4) It has a automatic redial feature for those times you are trying to log on to a very busy system. Just issue the command when you get a busy number and the modem takes over calling the number repeatedly.

DECBBS.LST

List of DEC Rainbow bulletin boards — 15 October — Copyright (c) 1986 by Rob Elliott

"+" 2400 bps, "*" 9600 bps RE: restricted, DA: daily, WE: weekend, WK: weekday.

Net/Node	System name	BPhone number	Location	Sysop	Flags
TOPS-20	DEC Market	617-467-7437	Marlboro MA	Bernard Eiben	Login: @LOG
13/15	Med Tech Fido	+716-897-0504	Buffalo NY	Bill Hliwa	WK:4p-10a
14/610	Omaha Fido	402-348-7603	Omaha NE	Jim Singer	WK:6p-6a
15/1002	UofA CAG	602-621-2097	Tucson AZ	Robert MacArthur	WK:10a-8a
18/7	Charleston Fido	803-763-4920	Charleston SC	Ed Ridpath	
18/13	TABS Fido	601-634-2365	Vicksburg MS	Don Bach	
19/503	Over the Rainbow	713-493-7483	Houston TX	John Lellis	RE:
100/16	Mike's Board	+314-726-3448	St. Louis MO	Mike Mellinger	
100/22	PC LUG	+314-576-2743	St. Louis MO	Ken Kaplan	
100/51	DECUS Central	+314-576-4067	St. Louis MO	Ken Kaplan	
101/14	WayStar	+617-481-7147	Marlboro MA	Kevin Porter	
101/27	Dave's Fido	+617-632-1861	Gardner MA	David Rene	
101/45	Midnite DEC	+617-787-3033	Brighton MA	David Strickler	
101/111	Truk Bord	+617-631-3304	Marblehead MA	Mark Bornstein	
101/125	Davy Jones Locker	+617-865-3290	West Millbury MA	Kichard Kenadek	
101/310	Dave's Annex	617-874-4325	Westminster MA	David Rene	WK:2p-7a
101/367	Rainbow Engineer	617-486-2285	Littleton MA	Bruce Gibson	
101/625	The Doctor	617-879-3714	Framingham MA	Herbie Cohen	WK:10p-6a
102/101	Rainbow Data	+213-204-2996	Culver City CA	Don Brauns	
102/104	King James II	+213-618-8454	Torrance CA	Glenn Bowes	DA:10p-4p
102/109	Rainbow Brite	+213-644-1963	Hawthorne CA	Bruce Headley	
102/110	Long Island RB	+213-370-4113	Los Angeles CA	George Dalhco	
102/111	Beach City RB	+213-376-9567	Redondo Beach CA	Dan Tanna	
102/113	Rainbow West	+213-305-8303	Los Angeles CA	Jay Rosenberg	
102/603	CompuLink	+805-494-3350	Westlake Village CA		Eric Daymo
102/701	Oberon Systems	+805-643-0982	Ventura CA	Scott Johnson	
103/301	SD Rainbow LUG	+619-488-2116	San Diego CA	Rick Eliopoulos	DA:7a-7p
103/501	King James RB	+714-537-7355	Garden Grove CA	Mike Hamilton	RE:
103/508	Medic	+714-964-0454	Huntington Beach CA		Phil St. Er
107/17	DEC-House	+609-866-9481	Cherry Hill NJ	Brian Sietz	DA:12m-5p
107/31	Rainbow Corner	+914-425-2613	Spring Valley NY	Ted Needleman	
107/723	HitchHikers Guide	315-589-7361	Williamson NY	Fritz Howard	
109/74	The Bear's Den	+703-671-0598	Falls Church VA	Kurt Reisler	
109/483	Wash-A-RUG	703-359-6549	Fairfax VA	Kurt Reisler	
109/601	Beauty Board	+301-725-7510	Laurel MD	John S Raum	
109/625	Catt House	+717-794-5268	Blue Ridge PA	Bob Catt	
109/645	Rainbow News	+703-280-2878	Fairfax VA	Caroline Mack	
114/3	Rainbow BBS	602-952-8520	Phoenix AZ	Jim Koshner	RE:
115/100	Illini Data RB	+312-759-5402	Bolingbrook IL	Rob Elliott	
115/123	Chicago DECUS	312-490-9206	Schaumburg IL	Chuck Garrett	
124/10	Big D Fido	+214-392-1121	Dallas TX	Dennis Forcier	
129/15	DEC-User	412-469-2468	Pleasant Hills PA	John Vukovic	

130/120	Hubbin's Board	+302-239-3969	Hockessin DE	Van Olmstead	
132/107	M'Cycle Bytes	+603-889-3366	Amherst NH	Robert Nilson	
132/225	SeaCoast Fido	207-439-9367	Kittery ME	Bill Thomas	
132/620	Spaghetti Heaven	603-635-7771	Pelham NH	Victor Coppola	DA:10p-7a
132/615	Wizard's Castle	+603-883-1596	Windham NH	Paul Gilberti	
134/3	Husky's Board	+403-743-4900	Ft. McMurray AB	Don Thompson	
134/101	Nemesis Fido RB	+403-355-3881	Faust AB	Benoit Guay	
134/103	NotQuiteAFullDEC	+403-791-1486	Ft. McMurray AB	Don Thompson	
138/32	Glacier Peak RB	+206-644-8431	Bellevue WA	Garry Stebbins	WK:12m-5p
138/34	Arctic Net	+206-581-7003	Steilacoom WA	Rob Barker	
141/320	Rainbow Surprise	203-795-0339	Orange CT	David Hecht	RE:
141/491	Naugy Net	+203-729-7569	Naugatuck CT	Vince Perriello	
154/600	Tele-Post RB	+414-964-4046	Milwaukee WI	John Spiegel	
154/777	Large DEC	414-354-5941	Milwaukee WI	Mark Buda	
161/613	Rainbo Works	+209-832-1002	Manteca CA	Andre Coltrin	
166/3	ConservEnergy	+714-645-7747	Costa Mesa CA	Rick Ellis	
No Node	DRUG bbs	*303-671-0801	Denver, CO.	Jim Turner	WK:7p-7a WE

Three books for the Rainbow

Julie Starr

In our local bookstore in Palo Alto, I found three books specifically for the Rainbow:

- *Writing with WordStar on the DEC Rainbow.*
- *Going Online — Communications on the DEC Rainbow.*
- *Controlling Financial Performance with Lotus 1-2-3 on the DEC Rainbow.*

All of these were published under the DECBooks label, and the cover indicates this is "a division of Digital Press."

Digital Press is DEC's publishing arm for general interest and commercial technical publications, and I believe they have a pretty good reputation for putting out quality books. The address is 30 North Ave., Burlington, Mass. 01803.

The books are glossy paperbacks priced about $— to $— each. I quickly skimmed through them, and it appears that what they did was take, for instance, a book on writing with WordStar and add two or three Rainbow-specific chapters. I did not have time to get a good sense of the quality of the information they contained.

Curing Disk Fragmentation

Rick Lorenzen, Denver LUG

Disk Optimizer from **Softlogic Solutions** (it is distributed on double- sided IBM disks) is a disk utility program that solves a problem we all experience and probably aren't even aware of: disk fragmentation.

Although the program is intended for the IBM PC family, it works successfully on the Rainbow when run under Code Blue, but not DIBEM, the shareware version. By successfully, I mean that I have tried it on several deliberately fragmented floppies, and then, with much trepidation, on a badly fragmented 10 megabyte hard disk.

The smallest unit of storage on a disk under MS-DOS is called a cluster. Its size depends on the operating system On Rainbow hard disks under MS-DOS version 2.11 a cluster is 2048 bytes.

This means that all files are stored and take up space in multiples of this cluster size. Even though DOS says a file might be 137 bytes in length, it is really takes up one cluster, 2048 bytes, on the disk.

The 137 is just the actual number of real data bytes in the file. If you add up the number of bytes of all files stored on a completely full floppy, for example, you will rarely get a sum equalling the 396K bytes or so that is the capacity of the disk.

When a file is written to a blank disk, DOS starts writing data at some starting cluster and uses as many clusters as are needed to save the file. It uses sequential clusters, that is, clusters that are physically next to each other on the disk surface. (This is not technically true due to a thing called "interleaving," but the effect is the same).

When a disk has had lots of activity with lots of files being created and deleted, there are lots of empty clusters between the files stored on the disk. These empty clusters form holes like missing teeth in the overall pattern of full clusters on the disk.

When DOS writes a file to a normal disk, one with scattered files and holes, it locates the first available cluster and starts to write the file. If the file needs more than one cluster, the next available clusters are used. This chain of clusters may or may not be next to the starting cluster; in fact none of the clusters may be next to each other.

Consequently, a single file, using several clusters, may be scattered in pieces over the disk — fragmented. A very easy way to generate a fragmented file is to create a first file, create other files, then add information to the first file. The new files will surely cut off the chain of clusters forming the first file, forcing it to chain to a different place on the disk.

Why do fragmented files matter, especially since they are so common? Information is read most efficiently from the disk, at least theoretically, when the chain of clusters forming a file are contiguous. If the heads of the disk drive need to wander all over the disk grabbing clusters for a badly fragmented file, it takes longer to read the file, especially if it is large.

Ordinarily the only way to unfragment a disk is to copy all the files on the disk to some backup medium, erase all the files on the disk, then copy all files from the backup medium back to the disk. Because the disk now starts out blank, all files written to it will be contiguous.

This practice is common at mini- and mainframe computer sites and generally takes many hours. It is also the safest way to unfragment a disk. It would be nice if the disk could be "packed" or "compacted" more easily. This is where Disk Optimizer comes in handy.

Disk Optimizer completely rearranges the contents of a disk so that all multicluster files are written to contiguous clusters. In addition, frequently used directories may be placed at the beginning of the master disk directory for faster access, and classes of files, such as .EXE or .DBF, may all be placed at the beginning or end of the disk for faster access.

This is accomplished without writing anything to external media except RAM memory and is quite efficient compared to backing up a large disk to floppies. Softlogic claims that only a single free cluster need be present on the disk to do the job, although things run more slowly then. Moreover, it is allegedly possible to pull the power plug at any time during optimization.

Well . . .

Is it safe? Yes and no. Disk Optimizer is in its second version with a large number of users. Only a very few bugs were found in version 1, all dealing with certain copy-protection schemes which fail if a file is moved. This problem has apparently been fixed. No reports of untoward occurrences have been heard.

Disk Optimizer appears to be a very robust and successful product.

However the operations carried out by Disk Optimizer are difficult, and the smallest error probably produces an unusable disk. Consequently it is very important that all files on the disk to be optimized are backed up. This is advisable all the time and is good practice in general, and is violated by everyone.

What about using Disk Optimizer on the Rainbow, for which it was not written? I called Softlogic some time ago and asked. The response was not "A what?" but "No."

I mentioned the availability of an IBM emulator which could take care of funny screen and keyboard differences. My expressed concern was that all disk operations be done through the operating system rather than by accessing the hardware directly. I was assured that the operating system was used for all disk input and output.

This was encouraging, but version 1 was copy-protected and could not be used on the Rainbow anyway, although for extra dollars an unprotected version was available. But version 2 is "not" copy-protected and just might work.

My copy of version 2 arrived on a double-sided IBM disk. I used an infernal machine to copy all files to a single-sided IBM disk and tried to install the software on my Rainbow using the supplied install program. Install failed when I was told to put the original disk in the floppy drive. I thought it was not copy-protected!

I next installed the software on infernal machine, copied all installed files to a single-sided IBM disk and tried to install software on Rainbow. Success!

What was the problem? A mysterious file on the distribution disk showed a size of 0 and just as mysteriously disappeared after the install procedure. Apparently this file really isn't of length zero. Its contents are checked against some expected information apparently. Clever but pointless.

I tested it by DISKCOPYing the distribution disk and installing the copy on the infernal machine. The copy installed, so there really is no copy-protection.

The problem with getting the software onto a Rainbow is that a single- sided disk may not be created from a double-sided disk with DISKCOPY but only through COPY, which will not copy 0 length files. Also, file transfer via communications programs will not send more than 0 bytes of a 0 byte file. Grrrr.

So what is to be done? The only solution I can come up with is to install Disk Optimizer on its favorite computer and then copy all files to a single-sided floppy, or transfer them, via a communications programs, to a Rainbow. Presumably this will not violate the license agreement as long as the software is not used on both machines.

The above nonsense is apparently worth the trouble. I gingerly tried to optimize a deliberately fragmented floppy without using Code Blue or DIBEM and got Interrupts Off. I then tried Disk Optimizer with DIBEM, and again got Interrupts Off.

Lastly I tried Disk Optimizer with Code Blue with no switches set, and ran successfully, including the successful unfragmentation of the floppy. The optimize program does not use all 25 lines of a 25-line display, so no display problems occurred.

I then very gingerly optimized a nearly full and very badly fragmented 10 megabyte hard disk. (Actually I didn't do anything gingerly: I just tried not to look). Expecting the worst, I executed all executable programs with no difficulty. None were copy-protected however.

I looked at a number of text files, and all appeared OK. I ran CHKDSK and Norton's disk test successfully. The disk seems to be OK and is noticeably quicker in accessing files.

With the above caveat about backing up your files routinely, I would certainly recommend the product. Disk Optimizer still needs to be tried on a disk with a copy-protected program installed. I don't (and won't) have any.

What about performance of an optimized disk? Softlogic claims that disk read-write times can be cut by up to two-thirds. This is significant for such input-output bound applications as databases. For much routine use, such as loading programs and working with not large word processing files, the time savings may be in the range of seconds rather than minutes.

By the way, how did I know my disks were fragmented? Two ways, actually.

Disk Optimizer comes with an analyzer program that tells you how badly your files are fragmented. This program does assume a 25-line display, and some directory options are not accessible due to scrolling of the display. It is always possible to get an analysis of the entire disk by simply pressing return at this point however. The program runs quickly and is easily stopped in any case.

A second method that anyone can use involves the MS-DOS program CHKDSK. Execute **CHKDSK <filename.ext**, where **<filename.ext>** is any valid path/file name combination, including wild card characters. At the end of its display, CHKDSK will list each file which matches filename.ext and is fragmented, and will tell you how many fragments the file is broken into. Another undocumented feature of the world's most popular toy "operating system."

The Disk Optimizer disk also includes some other utilities of interest. LOCK and UNLOCK will encrypt and decrypt files using a password so that others may not use them. Both must be run under Code Blue.

They both work well, except that while LOCK erases your password from the screen after you have entered it, UNLOCK does not. I had some trouble encrypting an executable program, then trying to execute it. It failed to execute with an **Insufficient Memory** error, which was fine, but subsequent execution of UNLOCK crashed the machine. On rebooting, UNLOCK successfully decrypted the program. I don't think these are intended to encrypt executable programs.

Another utility called FILEPEEK allows you to inspect files as you would with DEBUG or Norton's utility NU. This must be run with the /v switch under Code Blue, but due to scrolling of the 25th line it is basically unusable. No matter; DEBUG and Norton are of equal utility.

Two low-cost optimizer options

Carl Neiburger

Rainbow owners who want to eliminate disk fragmentation have two less expensive alternatives to Disk Optimizer, both of which work without Code Blue: A free program named RBREFMT.ARC and a "user-supported" program called DOG101.ARC

I have tried both of these programs briefly, using each successfully on one floppy disk, but have not had the courage nor the inclination to risk either on a hard disk.

Judging by the information included with both programs, DOG — short for Disk OrGanizer — is the more thoroughly tested of the two. Indeed Tony Camas, who ported RBREFMT to the Rainbow and is also the author of WUTIL, the Winchester utility, suggests using DOG, which apparently came out after RBREFMT.

RBREFMT comes only with the program itself, compiled in Turbo Pascal, and the source code. No instructions.

DOG was written by G. Allen Morris III of Oakland. It comes with 11 pages of instructions, but they're written so confusingly that I have never tried to come to terms with them. The program appears, however, to allow a variety of options in reorganizing data on disk drives to optimize performance.

The instructions also state that DOG has been tested extensively. They do warn, however, against reformatting a disk containing copy-protected programs.

Camas says he has used DOG a number of times with no ill effects.

Morris asks that commercial and government users pay a $— licensing fee. He asks private users to pay "whatever you think it is worth."

I would certainly recommend that anyone seeking to reformat disks take a look at DOG. It would also be nice if someone would accept the challenge of translating its manual for the rest of us.

DEC reportedly dumps MS-DOS 3.1 for Rainbow

Reprinted from the Silicon Valley Newsletter, May 1987

BULLETIN: DEC has dumped plans to release the Rainbow Network Integration Kit, MS-DOS 3.1 and MS-Windows for the Rainbow, according to reports from the DECUS meeting in Nashville, Tenn.

MS-DOS 3.1 would permit use of larger hard disk drives and increase the number of files permitted in one directory.

The network kit would permit Rainbows to be tied into DEC's new system to link personal computers with VAXes, allowing transparent communication between them. DEC reportedly still plans to issue the network kit for IBM PCs.

The Nashville announcement appears to renege on DEC's February announcement that — although it was discontinuing manufacture of Rainbows — the network kit, MS-DOS 3.1 and Windows would be released.

One report said DEC attributed the decision to economic reasons, and problems in getting the network products to market in a timely manner.

Editor's note: Following the announcement of the above decision of Digital's to discontinue the Rainbow Network Integration Kit, the DECUS PC Sig wrote a letter to Digital requesting that the individual components of the NI Kit be unbundled and released. Large numbers of Rainbow users have expressed their concern and intention to write similar letters. The following letter is what the DECUS PC Sig has sent to DEC: if you are interested in the future of the Rainbow, please compose a similar letter and let Digital know of your concern.

Joint Request for Windows and MS DOS 3.1

Barbara A. Maaskant

The announcement of the "death" of the Rainbow at DECUS was certainly a disappointment to many of us, though not unexpected. Cancellation of the Network Integration Kit, albeit a GOOD business decision as presented by Ron Gemma of Digital, compounded that disappointment. Many attendees expressed a desire for parts of the NI as well as support for third party solution. In response to this,our SIG and the European PC SIG counterpart representation hurriedly formulated the following letter and mailed it the last day of symposia. So far there has been NO response.

To reinforce this request correspondence from you is needed. Please send a letter to John Rose or Ron Gemma at the same address and reference the PC SIG letter dated May 1, 1987. Hopefully we will see something positive in the near future.
decpage: literal

John Rose
PCSG Group Manager
30 Porter Road
Littleton, Mass.

Dear John,

Digital's recent decision to cancel the Network Integration Kit development fosters a new problem onto the Rainbow user community. Observations at the Spring DECUS Symposia confirmed that there is a strong interest in parts of the NI kit that would prolong the usefulness of the Rainbow: specifically MS-DOS 3.1 and MS Windows.

This interest was originally observed in the Fall DECUS in San Francisco last year when our membership spoke out in unison for unbundling the NI. Cancellation of the project has not diminished this desire.

On behalf of the United States and European Personal Computer Special Interest Groups we urge you to engage in a cooperative agreement with Microsoft to release these two programs.

Your responsiveness to this issue would be greatly appreciated by your customers.

Barbara A. Maaskant
Chair, PC SIG United States

Holberg Masson
Chair, PC Sig Europe

Lynn Jarrett
Vice Chair/Rainbow WG Chair

Paul Sawyer
DECUS Symposia Committee Chair, Europe

Update For Version 3.1 of P/OS and More

Thomas R. Hintz

At the 1987 spring DECUS symposium in Nashville, the DEC developers for the PRO-300 computer made several diskettes available to the PC SIG for duplication.

The first diskette contained PRO/VLINK V2.0-03 for P/OS V2.0 & V3.0. For those of you not familiar with this program, it is designed for the less experienced users of the PRO/Toolkit. To create a running program the user is required to create the program, a task builder command file, and an overlay descriptor file. Although not terribly difficult for the experienced programmer, it can be a formidable obstacle for the beginning users of P/OS. This program relieves the programmer from having to create these two complex files until he better understands how it is done. The user can simply create, compile, link and run the application under development.

The second TWO diskettes are updates to the P/OS V3.1 operating system. If you already have V3.1 of P/OS, you might be interested in this upgrade. They provide support for the two newest printers, the LA75 and the LN03-PLUS. They are bootable diskettes that will automatically upgrade your version of P/OS V3.1 ONLY.

For those of you not able to attend the DECUS meeting in Nashville but would like copies of these diskettes, do the following:

1. Send three RX50 formatted diskettes
2. With suitable protection
3. With self addressed, and stamped envelope
4. Label each diskette

Send all of the above to:

Thomas R. Hintz
University of Florida
IFAS Computer Network Building
120 Gainesville, FL 32611

REMEMBER, I will NOT replace bad diskettes, add postage, write addresses, print labels, or provide additional packing. What I get sent to me is what you get back after my best attempt at copying the diskettes.

Hard Disks and Controller Boards For the PRO

Thomas Hintz, Univ. of FL, Inst. of Food & Agr. Sc., Gainesville, FL

Larger capacity hard disks for the PROfessional 300 computer have become increasingly important as owners add application programs to their single user systems. Although the 5MB disk may be adequate for the most basic configuration or when connected into the server environment, the need for bigger and faster disks will continue to grow.

The PRO supports Digital's RD line of 5 1/4 inch winchester hard disks. The RD disks currently range from 5MB to 159MB, however, not all are supported by P/OS or the disk controller. Table 1 lists the currently available RD drives along with pertinent information.

PART NUMBER	SIZE (MB)	MANUFACTURER	I/O HEADS	P/OS* VERSION	CONTROLLER REV. LEVEL
RD-50	5	Seagate ST-506	2	1.0	0
RD-51	10	Seagate ST-412	2	1.0	0
RD-31	20	Seagate ST-225	4	3.0	0
RD-52	33	Quantum Q-540	8	2.0A	1
RD-53	67	Micropolis 1325D	8		
RD-54	159		12	no support	

* lowest P/OS version providing support

Table 1. Information on various RD hard disk drives that fi tinto DEC PROfessional computers.

You will notice in table 1 that the RD53 does not indicate support by P/OS and the RD54 shows no support. According to DEC, the existing controller board does not support 12 I/O heads. The number of I/O heads that can be controlled by the controller board is the limiting factor. Consequently, the RD54 will not be released for support on the PRO with this controller board. We can only speculate at this time, since the RD53 has only 8 I/O heads (like the RD52 which is supported), it might be supported in future releases of P/OS.

Identification of the hard disk drive is relatively easy. But, the controller board required to support the drive is more difficult to identify. Several versions of the board have been sold by DEC. After having looked at a number of controllers it is not always clear just what revision (REV) level you have. All boards contained the number 54-15134 etched as part of the printed circuit. They also contain many other letters and numbers either etched, stamped or pasted on the board that indicate the REV level.

Their appears to be two types of REV levels that are important to determine the capabilities of your board. The first type is a letter designator for the board type. REV level D or F and greater (i.e. D2, F1 or H1) can support the larger capacity disks with the correct ROMs. REV level C or E will not work with the larger capacity drives even with the newer ROMs.

The second REV type is a number that appears to indicate the ROM support level type. The only boards I have seen of this type have a 01 stamped after the etched board number (i.e. 54-15135-01). The etched board number may appear more than once on the board, but the version number 01 may only appear after one of them. These boards should contain the ROMs needed for the larger capacity disks. Table 2 lists the important numbers found on the board and the ROMs.

Table 2. Numbers found on various components of hard disk controller board.

CONTROLLER	OLD (REV. 0)	NEW (REV. 0)	??? (REV. 0)	NEW (REV. 1)
board no.	54-15134	54-15134	54-15134	54-15134-01
ROM #1	014B2	014B2	014B2	073B2
ROM #2	013B2	013B2	013B2	072B2
ROM #3	012B2	012B2	012B2	071B2

The ROMs of importance are in the upper right hand corner of the controller board. The chips are arranged according to the diagram in figure 1. Each chip can have up to four numbers stamped on them. Only the numbers ending with B2 are important for determining the REV level. Compare them with the numbers in table 2 to determine the REV level of your board.

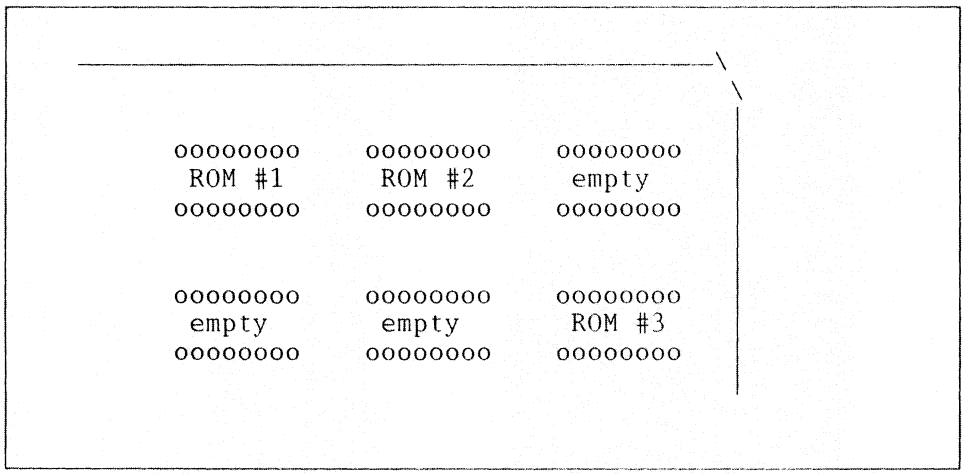


Figure 1. Upper right corner of hard disk controller board.

Although it has been stated in some articles that V3.0 of P/OS can control either revision of the controller board for the higher capacity disks, this may not be completely accurate. I have been told by DEC that it may appear to work in some cases, but its reliability is questionable if it works at all. They also state that all three ROMs need to be upgraded, not just ROM #3, if reliability is desired.

Consequently, to support larger disk capacities on a PRO that has the REV 0 controller board, it will be necessary to purchase a REV 1 controller board or acquire the newer chip set if your existing board is upgradeable. These should be available from DEC or from other 3rd party vendors such as Horizon Computer Services in Edison, NJ (201-420-5888) or American Digital Systems in Sudbury, MA (617-443-7711).

Tailoring P/OS To Save Disk Space

A Spring '87 DECUS Symposium Report by Thomas R. Hintz

At the fall 1986 DECUS symposium in San Francisco, the P/OS development group from DEC presentented an excellent session that discussed various methods to conserve disk space with P/OS V3. The results of that session were reported in an earlier newsletter article by Gary Rice (Reclaiming Disk Space After Installing P/OS V3, January 1987, Vol 2, No. 5, pp PC-2 to PC-6).

At the Q & A following this DECUS presentation it was noted by many members of the audience that at least half of those present still use V2 of P/OS. Because of this, DEC was requested to repeat the session at the next symposium and to include information useful to V2 of P/OS and PRO/COMMUNICATIONS.

As requested, DEC provided the types of information needed by developers and/or users for tailoring their system to make the most efficient use of the limited disk space. The following information was taken from notes and session handouts provided at the spring 1987 DECUS symposium in Nashville.

One comment made during the session refered to the earlier published article. The comment was to change all references of DW2: to LB0:. Other than that minor change, the information was correct.

PRO/COMMUNICATIONS files that can be deleted:

Can be deleted if one is not using the LAT feature of PRO/COMM:

```
[ZZSYS]LCP.TSK              (65 blocks)
[ZZSYS]XLDRV.STB            (4 blocks)
[ZZSYS]XLDRV.TSK            (31 blocks)

(User directory NODE$$PROVOLUME$USER:[ZZCOMM])
NODE$$PROVOLUME$USER:
```

Can be deleted in all cases:

```
[ZZCOMM]COMIN2.TSK          (15 blocks)
[ZZCOMM]COMIN3.TSK          (18 blocks)
```

Can be deleted if you are not going to use the phonebook feature of PRO/COMM:

```
[ZZCOMM]PHONEBOOK.PBK (V3)  (7 blocks)
[ZZCOMM]PHONEBOOK.DAT (V2)
```

Can be deleted in all cases:

```
[ZZCOMM]COMIN5.TSK          (18 blocks)
```

Can be deleted if one is not using DECMATE to PRO communications:

```
[ZZCOMM]DXFLX.TSK           (83 blocks)
[ZZCOMM]DXLPT.TSK           (96 blocks)
```

These can be deleted if you are not using PRO to WPS file transfer capabilities:

```
[ZZCOMM]PROAXT.TSK          (73 blocks)
[ZZCOMM]WPSQIO.MSG          (29 blocks)
[ZZCOMM]WPS11.MSG           (10 blocks)
```

This file can be deleted if one uses only "standard" PRO/COMM terminal emulation:

```
[ZZCOMM]STE.TSK             (42 blocks)

(System Directory NODE$$PROVOLUME:[ZZFILEX])
```

Can be deleted if one is not using PRO to PRO file transfer:

```
[ZZFILEX]XFH.TSK            (79 blocks)
```

Can be deleted if one is not using HOST to PRO file transfer:

```
[ZZFILEX]XFS.TSK            (72blocks)
```

** Groups of Optional Files for P/OS V2.0-V2.0A **

Runtime libraries/Callable System Services
Drivers (Comm port and TMS)
Help files
Graphics libraries

P/OS V2.0 Contents of Distribution Kit, Diskettes

"OPTIONAL" SYSTEM FILES COPIED TO DISK FOR P/OS V2.0-V2.0A SYSTEM INSTALLATION VIA PROSYSTEMV2:[1,54]SCRIPT.COM
(Files are copied to DW2:[directory])

Key to special codes in this listing are:

S following the filename indicates removing the file is "Safe", although precautions should always be taken.

D following the filename indicates removing the file is "Drastic", meaning that it can be removed in certain circumstances but may result in the system being inoperative at a later date, or problems may occur that are difficult to diagnose.

X* indicates file is part of the following system function:

R* = Runtime libraries/Callable System Services
D* = Drivers (Comm port and TMS)
H* = Help files
G* = Graphics libraries

```
FILE NAME SIZE IN BLOCKS
!
! From volume label PROSETUPV2
!
[ZZSYS]CTEX.MSG
[ZZSYS]CMAIN.TSK
[ZZSYS]ACCNT.TSK
[ZZSYS]CSUTL.TSK
[ZZSYS]SERSAP.TSK
[ZZSYS]SETUP.MSG
H* [ZZSYS]SETUP.HLP S 12.
[ZZSYS]CAINS.TSK
[ZZSYS]CAREM.TSK
R* [ZZSYS]FCSRES.TSK D 31.
[ZZSYS]CSTART.TSK
D* [ZZSYS]XKDRV.TSK S 18.
D* [ZZSYS]XTDRV.TSK S 18.
[ZZSYS]COMLIB.TSK
[ZZSYS]COMSETUP.DAT
[ZZSYS]CGLGRT.TSK
G* [ZZSYS]GIFILE.TSK S 50.
G* [ZZSYS]GIBITM.TSK S 59.
!
! From volume label PRODISPATV2
!
[ZZSYS]STARTUP.TSK
[ZZSYS]CTEX.TSK
[ZZSYS]CBUTL.TSK
```

```
[ZZSYS]CFUTL.TSK
[ZZSYS]CDUTL.TSK
[ZZSYS]CPRNT.TSK
[ZZSYS]CBTLOG.TSK
[ZZSYS]CPUTL.TSK
[ZZSYS]CVUTL.TSK
[ZZSYS]COPY.TSK
!
! From volume label PROLIBRARV2
!
[ZZSYS]DAPRES.TSK
[ZZSYS]RMSERR.MSG
[ZZSYS]BRUTL.HLP
[ZZSYS]BRUTL.MNU
[ZZSYS]BRUTL.MSG
[ZZSYS]CTEX.HLP
[ZZSYS]DUTL.MNU
[ZZSYS]DUTL.HLP
[ZZSYS]DUTL.MSG
[ZZSYS]FUTL.MNU
[ZZSYS]FUTL.HLP
[ZZSYS]FUTL.MSG
[ZZSYS]SYSTEM.MNU
[ZZSYS]PRINT.MNU
[ZZSYS]PRINT.HLP
[ZZSYS]PRINT.MSG
H* [ZZSYS]MESBRD.HLP S 3.
[ZZSYS]INSAPPL.SYS
[ZZSYS]MSGBOARD.SYS
[ZZSYS]INIT.MSG
[ZZSYS]NSHODIR.TSK
!
! From volume label PROUTILV2
!
[ZZSYS]PROSE.MNU
[ZZSYS]CET.TSK
[ZZSYS]PROSE.HLP
[ZZSYS]PROSE.MSG
[ZZSYS]PROSE.UDK
R* [ZZSYS]C81LIB.TSK D 34.
R* [ZZSYS]PBESML.TSK D 34.
R* [ZZSYS]PBFSML.TSK D 34.
R* [ZZSYS]PROF77.TSK D 34.
R* [ZZSYS]DBLPRO.TSK D 30.
[ZZSYS]PASRES.TSK
R* [ZZSYS]PROSORT.TSK D 26.
[ZZSYS]PROSORT.SYS
[ZZSYS]CGLEIS.TSK
[ZZSYS]CGLFPU.TSK
G* [ZZSYS]GIHPGL.TSK S 47.
R* [ZZSYS]DBLRES.TSK D 35.
R* [ZZSYS]FMSRES.TSK D 23.
[1,2]PASERR.MSG
[1,2]DIBOLERR.MSG
[1,2]PROF77.MSG
[1,2]C81RTE.TXT
[1,2]BASIC2.ERR
[1,2]FMSERR.MSG
[1,2]C81DBG.HLP
```

** Groups of Optional Files for P/OS V3.0-V3.1 **

FONTs and Definitions for printing
Pro/Sight Support
Color Monitor Setup
Requested Printing/Printers
Positional Device Drivers/library.
PRO & callable/Communications.
Setting Defaults
Miscellaneous Graphics files
Help for System Services, and error reporting.
Other Miscellaneous

Account and Workstation management
Copy/Install, Delete/Remove application
Runtime libraries/Callable System Services
Language runtime systems

V3.0 Contents of Distribution Kit, Diskettes

"OPTIONAL" SYSTEM FILES COPIED TO DISK FOR P/OS V3.0-V3.1 SYSTEM INSTALLATION VIA PROSYSTEMV3:[1,54]SCRIPT.COM OR V31UPDATE1:[1,54] SCRIPT.COM

(Files copied to LB:[directory] or LB:[root.directory], LB: = DW2:)

Key to special codes in this listing are:

S following the filename indicates removing the file is "Safe", although precautions should always be taken.

D following the filename indicates removing the file is "Drastic", meaning that it can be removed in certain circumstances but may result in the system being inoperative at a later date, or problems may occur that are difficult to diagnose.

-Un indicates newer files found on V31UPDATEn: diskettes.

X* indicates file is part of the following system function:

A* = Setting Defaults, protection, and general setup
W* = Account and Workstation management
P* = Requested Printing/Printers
I* = Copy/Install, Delete/Remove application
R* = Runtime libraries/Callable System Services
L* = Language runtime libraries and misc. language files
D* = Positional Device Drivers/library.
C* = PRO & callable/Communications.
S* = Pro/Sight Support
K* = Color Monitor Setup
H* = Help for system Services and error reporting.
F* = FONTs and Definitions for printing
G* = Miscellaneous Graphics files
M* = Miscellaneous

NOTE: *Help files that do NOT have an H* next to them are files that CANNOT be deleted if the function is deleted (ex. if the file PRINT.HLP is deleted but print services is still installed, an unRESUMable error will occur if print services is chosen.*

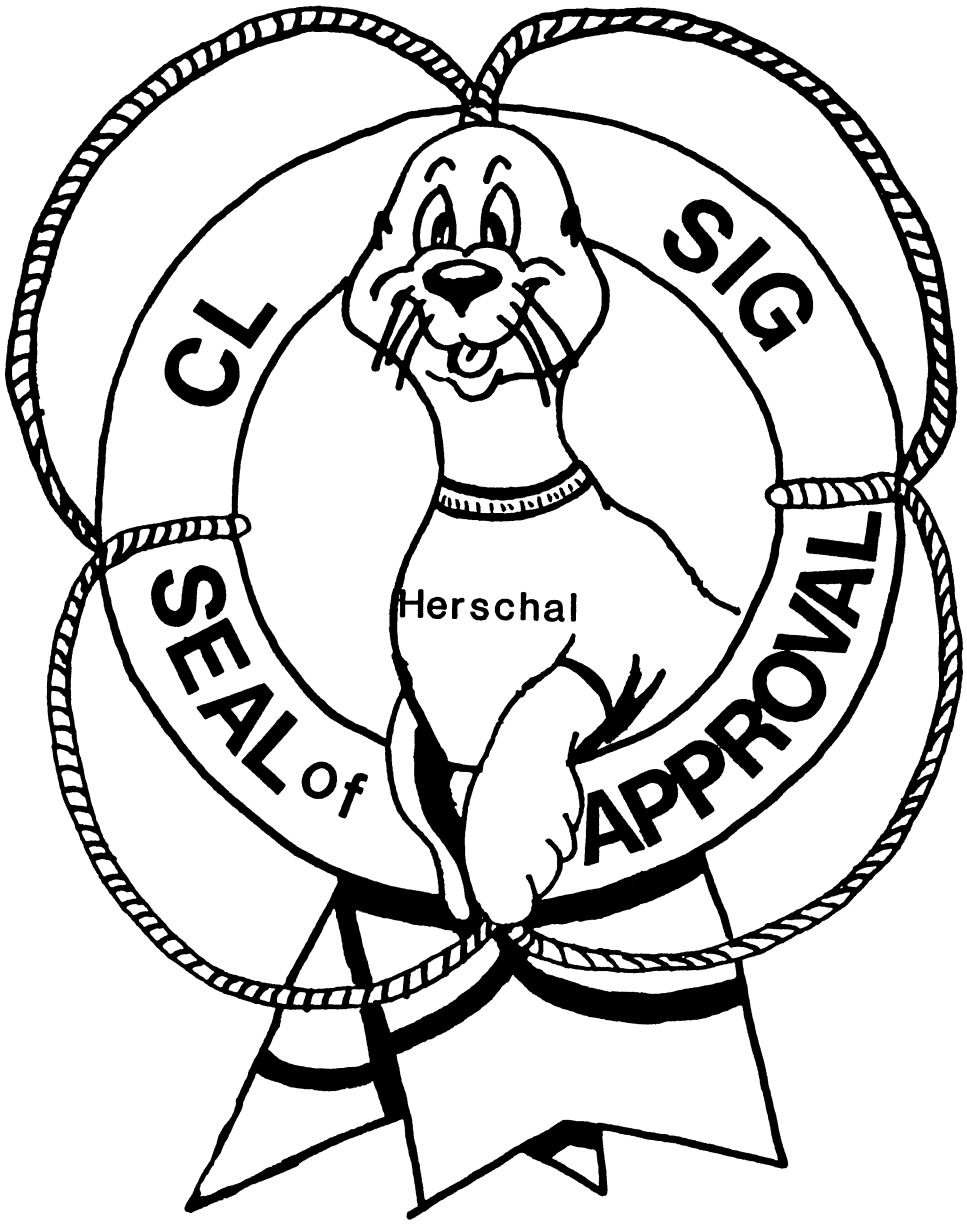
```
FILE NAME SIZE IN BLOCKS
!
! From volume label PROSETUPV3:
!
W* [ZZSYS]AMTMAN.TSK D 121.
M* [ZZSYS]BASECOM.STR S 1.
H* [ZZSYS]CBTERR.TSK S 21.
R* [ZZSYS]CET.TSK D 122.
K* [ZZSYS]COLOR.TSK S 31.
P* [ZZSYS]CPR.TSK S 41.
A* [ZZSYS]SETDIR.TSK S 39.
!
! From volume label PROCTABV3:
!
I* [ZZSYS]AREM.TSK D 65.
W* [ZZSYS]DELALL.TSK D 20.
A* [ZZSYS]DFLACN.TSK S 63.
P* [ZZSYS]DSPL.TSK -U2 S 74.
A* [ZZSYS]FIRSTAPP.TSK S 34.
A* [ZZSYS]FPROT.TSK S 39.
I* [ZZSYS]INSTALL3.TSK D 109.
R* [ZZSYS]PROSORT.TSK D 66.
P* [ZZSYS]QMG.TSK -U2 S 20.
P* [ZZSYS]QMGCOM.TSK -U2 S 42.
!
! From volume label PROLIBRARV3:
!
L* [ZZSYS]BP2V23.TSK D 34.
L* [ZZSYS]C23LIB.TSK D 34.
L* [ZZSYS]C81LIB.TSK D 34.
C* [ZZSYS]COMLIB.TSK D 13.
P* [ZZSYS]CPUTL.TSK -U2 S 116.
L* [ZZSYS]DBLPRO.TSK D 30.
L* [ZZSYS]DBLRES.TSK D 35.
L* [ZZSYS]PASCLU.TSK D 33.
L* [ZZSYS]PASRES.TSK D 34.
D* [ZZSYS]PDL.TSK S 8.
L* [ZZSYS]PBESML.TSK D 34.
L* [ZZSYS]PBFSML.TSK D 34.
L* [ZZSYS]PROF77.TSK D 34.
!
! From volume label PROGRAPH1V3:
!
F* [ZZFONT]DGM15.TSK S 14.
F* [ZZFONT]DGM1A.TSK S 18.
F* [ZZFONT]DGM1F.TSK S 22.
F* [ZZFONT]DGM5K.TSK S 25.
F* [ZZFONT]DGMZZ.TSK S 46.
S* [ZZFONT]FONT08.TSK S 7.
S* [ZZFONT]FONT09.TSK S 15.
S* [ZZFONT]FONT10.TSK S 15.
S* [ZZFONT]SIGHT.FDF S 1.
G* [ZZSYS]SVGRAF.TSK S 51.
P* [ZZFONT]DBULTN.FDF -U2 (V3.1 only) 2.
G* [ZZSYS]GIHPGL.TSK S 54.
```

```
G* [ZZSYS]GIPAL.TSK S 62.
P* [ZZSYS]GIBITM.TSK S 59.
G* [ZZSYS]GIFILE.TSK S 51.
!
! From volume label PRODRIVERS:
!
C* [ZZSYS]XKDRV.STB S 2.
C* [ZZSYS]XKDRV.TSK S 20.
C* [ZZSYS]XTDRV.STB S 2.
C* [ZZSYS]XTDRV.TSK S 19.
D* [ZZSYS]DTDRV.STB S 8.
D* [ZZSYS]DTDRV.TSK S 19.
C* [ZZSYS]YQAUTO.TSK -U2 S 6.
!
! From volume label PROMENUV3:
!
L* [001002]BASIC2.ERR D 8.
L* [001002]C81DBG.HLP D 13.
L* [001002]C81RTE.TXT D 10.
AH* [001002]DEFACN.HLP S 18.
A* [001002]DEFACN.MNU S 4.
A* [001002]DEFACN.MSG S 16.
L* [001002]DIBOLERR.MSG D 5.
MH* [001002]LOGIN.HLP S 11.
M* [001002]LOGIN.MNU S 2.
L* [001002]PASERR.MSG D 8.
L* [001002]PROF77.MSG D 7.
WH* [ZZSYS]AMT.HLP S 16.
W* [ZZSYS]AMT.MSG D 15.
W* [ZZSYS]AMTMAIN.MNU D 3.
IH* [ZZSYS]AREM.HLP S 7.
I* [ZZSYS]AREM.MSG D 11.
H* [ZZSYS]CBTERR.MSG S 11.
KH* [ZZSYS]COLOR.HLP S 4.
K* [ZZSYS]COLOR.MSG S 3.
H* [ZZSYS]MESBRD.HLP S 3.
P* [ZZSYS]PRINT.HLP -U2 S 35.
P* [ZZSYS]PRINT.MNU S 3.
P* [ZZSYS]PRINT.MSG -U2 S 37.
R* [ZZSYS]PROSE.HLP D 66.
R* [ZZSYS]PROSE.MNU D 3.
R* [ZZSYS]PROSE.MSG D 7.
R* [ZZSYS]PROSORT.SYS D 26.
P* [ZZSYS]QMGERR.MSG S 16.
I* [ZZSYS]REMINS.HLP D 12.
I* [ZZSYS]REMINS.MNU D 3.
I* [ZZSYS]REMINS.MSG D 16.
H* [ZZSYS]SUTERM.HLP S 30.
!
! From volume label PROACCOUNTV3:
!
P* [ZZSYS]DEFPCHAR.DAT -U2 S 2.
W* [ZZSYS]SPCOPY.TSK D 27.
M* [001002]APPLFIX.CMD S 15.
M* [001002]RELEASE.DOC -U2 S 32.
P* [ZZOPRINT]PPSETUP.DAT S 1.
P* [ZZOPRINT]QUEUE.SYS S 6.
W* [ZZUSER]UFLIST.DAT D
```

*** The rest of these template files all add up to appox. 156 blocks.

```
{ Root = [ZZUSER.] } Prototype user account (template)

W*  [ZZSYS]CLSETUP.DAT          D
W*  [ZZSYS]COMSETUP.DAT         D
W*  [ZZSYS]LOGIN.INI            D
W*  [ZZSYS]LOGOUT.INI           D
W*  [ZZSYS]MSGBOARD.SYS         D
W*  [ZZSYS]PRVSETUP.DAT         D
W*  [ZZSYS]USERMENU.SYS         D
W*  [001002]FMSERR.MSG          D

{ End Root = [ZZUSER.] }

W*  [ZZWS]GRPLST.DAT            D
W*  [ZZWS]RSX11.SYS             D
W*  [ZZWS]WSLIST.DAT            D

{ Root = [ZZWS.] }              Prototype Workstation Template

W*  [ZZDECNET]NETLINE.DAT       D
W*  [ZZSYS]PSETUP.DAT           D
W*  [ZZSYS]SPSETUP.DAT          D

{ End Root = [ZZWS.] }
```

Other files are rooted in [ZZLOCAL.] and [SYSTEM.] relating to Accounts, Pos'l Dev., and Printing are optional but may be too risky for hand removal. The expert/knowledgeble user could replace/edit the SCRIPT.COM files on copies of the bootable installation diskettes (commenting out the optional files) and reinstall the system. This would also provide a less fragmented disk. In any case file/default protections should be considered.



COBOL BASIC DIBOL RPG

COBOL Working Group

by

Bill Leroy, Chairman

To all users of the DEC COBOL language:

This is the FIRST of a new project -- hopefully a newsletter of sorts -- to people who use **COBOL** in the Digital Equipment Corporation (DEC) world. The intent is that it might help those of us who use **COBOL** to keep up with each other. It might also point out hints & kinks of using the various DEC **COBOL** compilers, and any other tools that are available, such as the VAX **COBOL** Language Sensitive Editor (LSE), and the VAX **COBOL** Code Generator.

As such, it is being sponsored by the **COBOL** Working Group of the Commercial Languages SIG (possibly Languages & Tools SIG by the time you read this). Each month it will be submitted to the SIG Newsletter Editor for publication in the combined SIGs monthly Newsletter. As in any volunteer newsletter, this one will be **USER** driven. This means that I will be using what **YOU** submit.

At least initially, it will also be sent directly to each person on the **COBOL** Working Group mailing list. If you would like to join the **COBOL** Working Group, please send your name, address, and telephone number to me, Bill Leroy, Chairman, at the address at the end of this article. I would also appreciate information on the application you support, as well as the hardware and software you use.

At this time, I am not sure what a **COBOL** Working Group does, having only agreed to be Chairman in Nashville, and being in the middle of the CLSIG asking to be merged into the L&T SIG. However, there will be a **COBOL** Working Group, no matter which SIG it attached to.

At a **too brief** organizational meeting held at the Nashville Spring Symposium, we discussed trying to sponsor a Pre-Symposium Seminar, possibly by Jerome Garfinkel, who wrote **"The COBOL 85"**, ISBN 0-471-80461-4. Bruce Gaardner is working on this. We will also work with the SIG Symposia Coordinator to sponsor **COBOL** sessions at each Symposium. If you have any ideas, or would like to present a session, please let me know.

(As symposia sessions begin to freeze 4 months ahead of time, by the time you read this, the SIGs will be looking for sessions for next spring in Cincinnati.)

This first issue will include a showcase on one of the **COBOL** Working Group members, one answer to a problem raised in Nashville, a write-up of an old program used many years ago to reduce hardware errors, and a current list of the **COBOL** Working Group members.

- -

Showcase on members - Bill Leroy

I have a "Compact" Microvax II (from DEC's CSS Group) and use VAX **COBOL** for development and support of programs for home offices of life insurance companies, using color terminals, LN-03 laser printers, and ISAM files. Common Data Dictionary and DECreporter are on the system, but I have not used them yet. However, I plan to do so. I also have a PDP-11/73, and support clients using programs written using COBOL-Plus (tm), running under TSX-Plus (tm) [multi-user RT-11], both from S&H Computer Systems, Inc., in Nashville, TN. I also use IBM DOS-VSE COBOL and CICS, and have (but do not use) a **COBOL** compiler on my Rainbow 100A.

- -

Create ISAM file using FDL

In Nashville, a question was raised as to how to use **COBOL** to create an ISAM file using FDL. The way I do it is as follows:

(1) Create an FDL file from the existing ISAM file.

```
$ analyze/rmsfile/fdl XXXXXX.ISM

$ edit/fdl/analysis=XXXXXX/script=optimize XXXXXX
```

(2) Create a sequential **COBOL** data file, FILE.SEQ.

(3) Use the following command in a **COBOL** program (or VMS DCL).

```
call "LIB$SPAWN" using by descriptor
"$ CONVERT/FAST/FDL=XXXXXX/PAD=%D32/STATISTICS FILE.SEQ XXXXXX.ISM".
```

(4) You will then have **quite quickly** an ISAM file.

- -

Computer Program Virtuously Eliminates Machine Errors

Spokesmen for a local electronics firm have announced a computer program that -- through fresh application of an old technique -- virtually eliminates lost time due to malfunction of components. Called OREMA (oh-RAY-ma, from the Latin 'oremus', meaning 'let us pray'), the program offers prayers at selected intervals for the continued integrity of memory units, tape transports, disk drives, and other elements subject to depravity.

Basically liturgical in structure, OREMA uses standard petitions and intercessions stored in subroutines in Latin, Hebrew, and English. It holds regular Maintenance Services thrice daily on an automatic cycle; and operator intervention is required only for making responses, such as 'And with thy spirit', on the console.

Prayers in English may be entered directly on the console, but normally, the Hebrew, Latin and English prayers are called from subroutines.

Although manufacturer-supplied prayer subroutines cover all machine troubles known today, there is the ability to add additional subroutines after the final existing AMEN section. Classified prayer subroutines are available for government installations.

In trials on selected machines, OREMA reduced by 98.2% the average downtime due to component failure. The manufacturer's spokesman emphasized, however, that OREMA presently defends only against malfunctions of hardware. Requestor errors and other human blunders will continue unchecked until completion of a later version, to be called SIN-OREMA.

Rewritten by Bill Leroy from a version credited to:

Dr. William S. (Bill) Minkler
Nuclear Engineer and columnist
NUCLEAR NEWS

- -

Known Users of DEC COBOL

Anthony, Kathryn
Programmer Analyst
Organon, Inc.
West Orange NJ 07052
201/325-4879

Bair, James F.
Mgr, Tech Support
Grinnell Infosystems, Inc.
Grinnell IA 50112-0795
515/236-6121

Batzel, Keith N.
Crowe, Chizek & Company
South Bend IN 46624
219/232-3992

Bersack, Barbara
Bexcom
Belmont MA 02178
617/484-9236

Buckley, John
V.P. Data Processing
Professional Underwriters Life
San Juan PR 00936
809/721-7500

Cruz, Sandy
Handley Computer Corp
Boulder CO 80307
303/494-5412

Error, Mike
Programming/Ops
Croatian Fraternal Union Of Am
Pittsburgh PA 15235
412/351-3909

Farler, Gregory K.
Programmer/Analyst
Libbey-Owens-Ford Co.
Laurinburg NC 28352
919/277-2203

Feerick, Mary Anne
RDBS Inc.
Kernersville NC 27285-0644
919/299-5037

Gaardner, Bruce L.
President
The Gaardner Group, Inc.
Saint Paul MN 55116
612/699-0870

Ginn, Rita
Programmer
Amex Travel Related Svcs Co
Atlanta GA 30329
404/320-3667

Handley, Bruce
Handley Computer Corp
Boulder CO 80307
303/494-5412

Heil, Ted
Educational Service Unit #3
Omaha NE 68137
402/330-2880

Kessler, Judy A.
Mitre - Tactical Communication
Ft. Huachuca AZ 85613
602/458-0863

Leroy, Bill
President
The Software House, Inc.
Atlanta GA 30355-0661
404/231-1484

Machi, Mario
Mgr, Program Develop
Maher Terminals
Jersey City NJ 07306
201/963-2100

Matthews, Herbert J, IV "Matt"
Logicon - C.R.W.
Alexandria VA 22314
703/836-7120

McGee, John
Dir, D P Development
Abbott, Jordan & Koon
LaGrange GA 30240
404/882-9226

Patton, Tom
E. I. DuPont
Old Hickory TN 37138

Piraino, Robert J.
Mgr, Info Services
Organon, Inc.
West Orange NJ 07052
201/325-4874

Pryor, Charles R.
Systems
Croatian Fraternal Union Of Am
Pittsburgh PA 15235
412/351-3909

Seifert, Wayne E.
Evansville Data Processing Cor
Evansville IN 47714
812/479-6951

Skypek, Mike
Programming
Skypek & Associates
Covington GA 30209

Wallis, Barry L.
Systems Analyst
Fleetwood Enterprises, Inc.
Riverside CA 92523
714/351-3682

Welborne, Jim
Crowe, Chizek & Company
South Bend IN 46624
219/232-3992

Wilson, Jim
Programmer/Analyst
Pfizer Quality Control Div.
Terre Haute IN 47808
812/299-2121

- -

And finally, please send any questions, hints, kinks, wish list items, bugs, "puff" sheets on yourself to be showcased in a future issue, new members, or whatever -- on pieces of paper, or if a long article, in VAX/VMS format, on 1600 bpi magnetic tape, or TK-50 -- to me at:

Bill Leroy (404) 231-1484
Chairman, **COBOL** Working Group
The Software House, Inc.
P. O. Box 52661
Atlanta, GA 30355-0661

(OR)

Bill Leroy (404) 231-1484
Chairman, **COBOL** Working Group
The Software House, Inc.
2964 Peachtree Road, N.W. #300
Atlanta, GA 30305-2120

THE DEVIAS LETTER

Contents:

CONTRIBUTION GUIDELINES

Contributions for the newsletter should be sent to:

Frank R. Borger
Michael Reese Medical Center
Department of Radiation Therapy
Lake Shore Drive at 31st St
Chicago, IL 60616

Contributions of letters, articles, important SPR's etc will be accepted in any form, (including notes jotted in pencil on gravy-stained tablecloths.) Contributions will be much more graciously accepted in one of the following formats:

1. Non machine readable sources, (SPR's etc,) should be reasonably dark to insure good photocopying. Text whatever should be the equivalent of 66 lines at 6 lpi, with 4-line top margin, 5-line bottom margin, left-margin 10, right margin 74 at 10cpi. If using a DEC LN03 for output, use left-margin 8. right margin 72.

2. Machine readable sources may be submitted on 9-track Mag-tape, (800,1600, or 6250 BPI,) DEC-tape II, DecMate floppies, or whatever. We're not fussy, we'll even accept paper tape or cards. Preferred format is DOS or BRU for tapes, Files-11 for DEC-tape II.

3. 1200 baud dial-up modems are available on our IAS system and our VAX, with KERMIT servers available. Give the editor a call at (312)-791-2515 (preferably later in the day,) to obtain access information, etc.

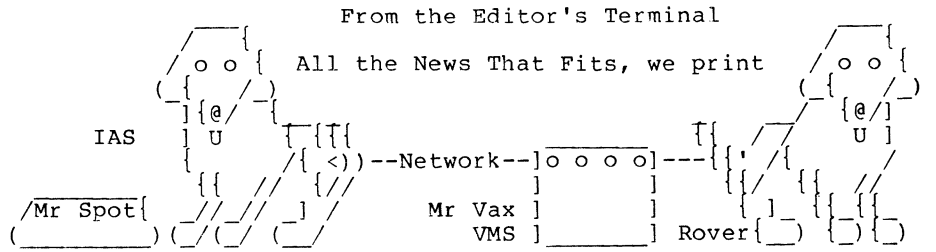
4. If long distance dialout is not possible on your system, we'll be willing to call your system and do the work, (unless you want to transfer the entire manual set at 300 baud.)

Any media sent to us will be promptly returned.

ASK THE DEVIAS WIZARD

If you have a problem you would like to submit to the Devias wizzard, write a letter or fill out a copy of a standard SPR and send it to the Editor at the above address. Answers to problems from members (or anyone) should also be sent to the Editor.

From the Editor's Terminal



Its heading for 90 degrees today, and the air-conditioning is spasmodic, and spring DECUS is a fond memory, (or mostly fond, nobody has fond memories of early morning committee meetings.) In other words, SNAFU, and its time to get down to the business of running this newsletter. As usual, when things go wrong, they really go wrong. Hans is fighting a spasmodic error in an RM03 drive positioner thats crashed 2 heads and 3 disks, and our normally rock solid hardware is crashing around us like flies. Even our LN03 is down, (the DEC repairman is working on it now,) If it gets repaired in time I won't have to use the old daisy wheel printer. (The serviceman just told me the LN03 spares kit is missing a board.) Oh well, another day, another deadline. If Mr Spot and Rover in the masthead above look a little wierd, blame DEC service.

This issue marks editorial number 12 for your humble servant. It doesn't seem that a year has passed since I took over the editorship. Tempus fugit, (but Tempus always was a little wierd anyway.)

For those of us that were at Spring DECUS, it was a time of mixed emotions. You still saw quite a few IAS people, but they tended to be going to more and more VAX session. I think in part this is because their IAS systems, although still running, are basically stable systems running established operations, with less intensive development. We did meet a couple of version 6 users, (from when 11D and IAS were different names for the same thing, and DEC was herding the 11D users to IAS.) A message to you people running early versions of IAS, or 11D. There still is an old guard of users who have kept their distribution tapes, etc from years back. If you have problems, etc. send a note to the Editor and we'll get the DeVIAS Demon to tackle the problem.

This issue includes an updated IAS SIG operating procedures. There are a couple of changes proposed by the SIG council. All interested members are invited to comment on the procedures.

Summer is only 2160000 clock ticks away, (1800000 in Europe.)

Report on the Nashville Symposium

This rather narrative report on the Nashville Symposium is your editor's humble attempt to compress a lot of discussions, presentations, (and non-presentations) etc into a capsule description of the symposium. We will also prepare material gathered at sessions for presentation in the newsletter, but also having to spend our time at more mudane things such as our regular job, will schedule these for the next couple of issues, (when the summer doldrums normally cut our page count to a minimum.)

Update D

Update D is scheduled to go to the field test users soon after you receive this newsletter. Among the things included in update D are:

1. BBR (Bad Block Replacement) support for MSCP devices.
2. RMS version 2.0 (yes Virginia, there really is version 2 of RMS)
3. SORT11, version 3.0.
4. Support for MSCP devices as a crash dump device.
5. Support for VT3XX terminals.
6. Various bug fixes.

DEC has made an obvious effort to get a wide assortment of sites as field test sites. Your editor hopes those sites will make a similar effort to really test this update, and hopefully send information to the newsletter concerning their experiences.

New IAS Librarian

Michael Robitaille announced that Ted Smith, of the University of Pennsylvania Hosptial, has agreed to accept the position of Program Librarian. The members of the Steering Committee extend a heartfelt "Welcome Aboard" to Mr. Smith. Your Editor's first DECUS membership card listed on the back 3 "objectives", one of which was, "Establish standards and provide channels to facilitate the free exchange of computer programs among members". As such, the Program Librarian coordinates a vital function of our SIG. Thanks a lot, Ted.

MSCP Presentation Cancelled

Much to the dismay of many attendees, DEC announced that the session on writing a device driver for MSCP class devices had to be cancelled, due to "The confidentiallity of the information contained in the session". Although this restriction is understandable, given DEC's commitment to controlling the number of "Clones" manufacturing add-on devices to DEC hardware, it was also pointed out that much of the information that would have been in the presentation is available to end users, and that sources for handlers are included in the distribution, so that the information, although "Confidential", is really available to anyone.

Informal WHIMS List

At the User's Forum, the user's presented the following "wish list" items to the IAS developers.

1. Support I & D space.
2. Modify DEMO to show node usage in SENPAR.
3. Provide warnings in the SPD for a new version or update when the new version drops support for a device or layered product, or requires new hardware or software. The SPD should be out long enough to let the user make a decision to buy the update.
4. Unprivileged users should not be able to link or run a privileged task. (An unprivileged user can write a program that finds his pud and sets the Privileged bit in the PUD.)
5. There should be a BATCH option to notify the submitter when the batch job is done.
6. Support LQP02, LN03 as spooled devices. (This would require QUE to optionally send data to the devices with null carriage control and with pass all characters, and possibly download a font file to the LN03.)
7. Modify INI to emit status on exit.
8. Provide a method to checkpoint a region.
9. Provide a switch to batch to not print the log file, and to rename the log file on exit.
10. Support WPS-Plus.

(Note that feedback to DEC from the users has been very minimal. If you have something you would like to see in IAS, PLEASE send in a WHIMS form. Send a note to the editor too, we'll put it in the newsletter to see what other users have to say about it.)

Ten Years Ago Today

The July 1977 Multi-Tasker contained:

An editorial by Mark F. Lewis, occasioned by the start of volume 8 of the Multi-Tasker, recalled the early days of RSX. He reminisced that;

"On the morning in 1973 when the SIG was first organized (at the San Francisco symposium), approximately 30 users of RSX-11D signed up (along with 12 to 15 DEC employees). These early users were sufficiently motivated (and if you recall RSX11D V2, you know why) that I did not have to wait long for submissions to the SIG newsletter to reach a level that required doubling the frequency of publications.

The early symposia, though haphazardly organized, reflected the motivation level of the users. Though frequently chaotic, these meetings were anything but quiet. Just about every user present participated in the discussions (perhaps 'arguments' is a better descriptor) with Digital personnel over the directions RSX-11D was taking and the persistency of certain problems in recurring in each release after release after release..."

(Your editor has been involved with RSX/IAS since early 1974 and is missing only Volume 1 number 1 of the newsletter, so he can attest to how chaotic things were for early users. Mercifully we did not have to work with version 2 of RSX11D, only because out distribution tape was bad, and by the time DEC replaced it, version 4 was out. Dec had the habit, up till IAS and RSX11M, of keeping odd numbered versions as in-house test versions, and only releasing even numbered versions to the public.)

The newsletter contained a user written version of ACCRPT, the Accounting report program. (The DEC system accounting report generator generated unbelievably verbose reports, easily running up to 10 or 20 pages per user per day. Since many users were still running small disks,and it could take an hour or or more to process a day's activity, most users tried accounting once and turned it off on their systems.) This program provided a minimum accounting which could be easily processed for billing purposes. (Accounting had some other unnice bugs. If your editing system took a long time and EDI exceeded any CPU or elapsed time quotas, your EDIT was terminated, without closing the output file. Nice going, DEC.)

The newsletter also published the original bylaws of the RSX-11/IAS Special Interest Group, as adopted at the spring 1977 symposium.

Program of the Month

The following program is a simple program to do graphs on a VT100 terminal using escape sequence cursor control. Its a quick and dirty package, but suffices for most simple data. Resolution is on the order of 2% vertical, 1% horizontal.

NOTE: Written in REESE BASIC, and uses the break command which dose a write pass all with null carriage control.

Could be converted to other versions of basic, as long as they incorporate an equivalent to the break function.

```
2000 ! do horizontal axis on vt100
2001 ! xl = min x value, xh=max x value
2002 ! nx=number of x tic marks 3,4,5,6,10
2003 ! uses variables xi,xp,xv,jj
2005 print chr$(27);"[22;10H";
2006 print "+-----------------------------------------------------
-------+"; : break
2010 xi=60/(nx-1) : xp=10
2020 for jj=1 to nx-2
2030 xp=xp+xi : print chr$(27);"[22;";frmt$(xp,2);"H+"; : break
2040 next jj
2050 xp=7 : xv=xl
2060 print chr$(27);"[23;";frmt$(xp,1);"H";frmt$(xv,6,2); : break
2070 xp=xp+xi : xv=xv+(xh-xl)/(nx-1)
2080 for jj=2 to nx
2090 print chr$(27);"[23;";frmt$(xp,2);"H";frmt$(xv,6,2); : break
2100 xp=xp+xi : xv=xv+(xh-xl)/(nx-1)
2110 next jj
2120 return
3000 ! do vertical axis on vt100
3001 ! yl = min y value, yh=max y value
3002 ! ny=number of y tic marks 3,5,6,11
3003 ! uses variables yi,yp,yv,jj
3010 for jj=2 to 22
3020 if jj<10 then print chr$(27);"[";frmt$(jj,1);";10H|"; : break
3030 if jj>9 then print chr$(27);"[";frmt$(jj,2);";10H|"; : break
3040 next jj
3050 yi=int(20/(ny-1)) : yp=22 : yv=yl
3060 for jj=1 to ny
3070 if yp>9 then print chr$(27);"[";frmt$(yp,2);";10H+"; : break
3075 if yp<10 then print chr$(27);"[";frmt$(yp,1);";10H+"; : break
3080 if yp>9 then print chr$(27);"[";frmt$(yp,2);";3H";frmt$(yv,6,
2); : break
3085 if yp<10 then print chr$(27);"[";frmt$(yp,1);";3H";frmt$(yv,6
,2); : break
```

```
3090 yv=yv+(yh-yl)/(ny-1) : yp=yp-yi
3095 next jj : return
4000 ! put cursor at 23rd line for prompt
4005 print chr$(27);"[23;1H"; : break
4010 return
4100 ! put cursor at top
4105 print chr$(27);"[1;1H"; : break
4110 return
4200 ! clear screen
4205 print chr$(27);"[2J"; : break
4210 return
4300 ! display lb$ on top of screen
4310 zz=(80-len(lb$))/2+1
4320 if zz<10 then print chr$(27);"[1;"frmt$(zz,1);"H";lb$;: break
4330 if zz>9 then print chr$(27);"[1;"frmt$(zz,2);"H";lb$; : break
4340 return
5000 ! display data point xv,yv using ch$ character
5001 ! uses xp,yp
5010 xp=int((xv-xl)/(xh-xl)*60+10.5) : yp=int(22.5-(yv-yl)/(yh-yl)
*20)
5020 print chr$(27);"[";
5030 if yp<10 then print frmt$(yp,1);";";
5040 if yp>9 then print frmt$(yp,2);";";
5050 if xp<10 then print frmt$(xp,1);"H";ch$; : break
5060 if xp>9 then print frmt$(xp,2);"H";ch$; : break
5070 return
```

Normal operating proceedure is to keep a copy of the above program in your default account. You then edit in the appropriate lines of the specific graph required. The following will generate a sample graph.

```
10 ! sample graph plot on VT100
15 if end then 1000
204200
25 xl=0 : xh=100 : nx=6 : gosub 2000
30 yl=0 : yh=100 : ny=6 : gosub 3000
35 dim lb$[70]v : lb$="Sample of VT100 graphics" : gosub 4300
40 ch$="o" : for i=1 to 10 : xv=i*10 : yv=xv : gosub 5000 : next i
45 ch$="x" : for i=1 to 100 : xv=i : yv=100-xv : gosub 5000 : next i
50 gosub 4000 : input "X, Y, Character ";xv,yv,ch$
55 gosub 5000 : goto 500
60 gosub 4000 : exit
```


IAS SIG
OPERATING PROCEDURES

Article I
Name

1.0 The name of the organization is the IAS Special Interest Group, (IAS SIG).

Article II
Purpose

2.0 The IAS SIG is established under the bylaws of the DECUS U.S. Chapter to:

1. Provide a forum for users of the IAS Operating System to exchange ideas, programs, and any other items of common interest.

2. Provide feedback to Digital Equipment Corporation (DEC) on all matters concerning the IAS operating system, related software products, services, policies, and all DEC manufactured computers, peripheral equipment, and other products and services.

Article III
Membership

3.0 Membership requirements:

1. Any DECUS member is qualified to be a member.

2. Any person qualified to be a member will be accepted as a member either upon submitting a request to the DECUS U.S. chapter, or personally indicating interest at a Symposium.

3.1 Rights of Members:

1. Members shall be eligible to participate in SIG activities and be members of the Steering Committee.

2. Five or more members of the IAS SIG may, by written petition, bring a motion before a meeting of the SIG Steering committee.

3. Any member that believes that they are being denied participation in SIG activities shall have recourse by petition to the DECUS U.S. chapter SIG council.

Article IV
Steering Committee

4.0 General

1. The IAS SIG shall be administered by the Steering Committee.

2. The Steering Committee shall consist of four officers and up to four at large members and up to four past officer members.

3. There will also be a non-voting, ex officio member of the IAS SIG Steering Committee appointed by the RSX SIG to be the RSX liason with the IAS SIG.

4. Any member of the IAS SIG may be on the Steering Committee and the Steering Committee shall be composed solely of members.

5. The Chairman may act independently on all matters, and shall inform and consult with the Steering Committee as (s)he sees fit. A majority vote of the remaining members shall be required to override decisions of the Chairman.

4.1 Steering Committee Officers

1. The Steering Committee officers shall serve for two years, and be elected by the Steering Committee.

2. The officers are the Chairman, the Newsletter Editor, the Symposia Coordinator, and the Program Librarian.

4.2 At Large Members

1. The Chairman may appoint up to two at-large members of the Steering Committee.

2. Up to two at large members may be appointed by a majority vote of the Steering Committee.

4.3 Past Officer Members

1. To assist in the transfer of responsibiities to new officers, past officers, when eligible, will remain as members of the Steering Committee (non-voting) for one year.

4.4 Duties of the Chairman

1. The Chairman is the chief executive officer of the SIG, and shall chair all Steering Committee meetings and be responsible for directing all activities of the SIG between meetings of the Steering Committee. The Chairman is subject to the review of the Steering Committee, or recall by a majority vote of the members.

2. The Chairman shall appoint one of the Steering Committee to be the liason with the RSX SIG.

3. The Chairman will propose, and recommend for adoption, to the Steering Committee, a resolution to dissolve the SIG when interest in the SIG's activities becomes too small to justify the SIG's existence.

4. The Chairman will represent the IAS SIG's interest in the DECUS U.S. Chapter SIG council.

4.5 Duties of the Newsletter Editor.

1. The Newsletter Editor shall edit and publish the SIG's newsletter called "The DeVIAS Letter".

2. In the event that the position of Chairman becomes vacant, the Newsetter Editor shall temporarily assume all duties of the Chairman except that of Steering Committee appointments until a permanent Chairman is elected by remaining members of the Steering Committee.

3. The Newsletter Editor will represent the IAS SIG's interests in the DECUS U.S. Chapter Communications Committee.

4.6 Duties of the Symposia Coordinator

1. The Symposia Coordinator is responsible for the planning and scheduling of IAS sessions at DECUS symposia.

2. In the event that both the position of Chairman and Newsletter Editor become vacant, the Symposia Coordinator shall assume all duties of the Chairman except that of Steering Committee appointments until a permanent Chairman is elected.

3. The Symposia Coordinator will represent the IAS SIG's interests in the DECUS U.S. Chapter Symposia Committee.

4.7 Duties of the Program Librarian.

1. The Program Librarian shall be responsible for the collection and distribution of IAS specific software via the DECUS tape copy program, with the RSX SIG.

2. In the event that the positions of Chairman, Newsletter Editor and the Symposia Coordinator become vacant, the Program Librarian shall temporarily assume all duties of the Chairman except that of Steering Committee appointments until a permanent Chairman is elected.

3. The Program Librarian will represent the IAS SIG's interests in the DECUS U.S. Chapter Library Committee.

4.8 Vacancy in Office

1. Should the Chairman vacate his(her) office by resignation, disability, or ineligibility or impeachment, a new Chair-

man shall be elected by a majority vote of the remaining members of the Steering Committee.

2. Should any other officer vacate his(her) office by resignation disability, or ineligibility or impeachment, the Chairman shall appoint a replacement.

Article V
Elections

5.0 Election of Officers

1. Officers shall be elected by the Steering Committee at the first Steering Committee meeting in every even numbered year, but not less than two years from the recog- of the SIG.

5.1 Appointment of the At-Large Members

1. The two at-large members appointed by the Chairman are intended to assist the chairman in discharging his or her duties. They will serve until discharged by the current Chairman.

2. The two at-large members appointed by vote of the Steering Committee shall be elected by the officers at the first Steering Committee meeting in every odd-numbered year, but not less than two years from the recognition of the SIG.

5.2 Past Officer members

1. Outgoing officers, unless ineligible or elected as an officer or appointed as an at-large member, or impeached from office shall become a Past Officer member of the Steering Committee for one year.

5.3 Impeachment of Officers

1. In accordance with Article III, the Steering Committee will accept any motion to remove an officer of the IAS SIG. The motion will be presented in the next Newsletter along with the comments of the remaining Steering Committee members and a request that members file a vote on the motion within 30 days. Should a majority of the respondents, comprising at least 1/4th of the membership at the time of the Newsletter's distribution agree to the removal, the officer is impeached, and must be replaced by election by the members, as described below.

5.4 Nominations

1. Should an officer be impeached, or all Steering Committee officer positions become simultaneously vacant, nomina-

tions for that (those) position(s) will be accepted by the Newsletter Editor, or the person designated to function in that capacity. Nominations will be signed by three members and accompanied by a letter indicating willingness to serve. All members may vote by returning a ballot published in the Newsletter. The nominee receiving the most votes will be elected and take office immediately.

Article VI
Meetings

6.0 General Meetings

1.0 Business meetings shall be scheduled at the Spring and Fall DECUS U.S. Chapter Symposia.

6.1 Steering Committee meetings

1. The Steering Committee shall meet by phone prior to each general meeting, or at the Chairman's request, and shall also meet at each DECUS U.S. Chapter Symposium.

Article VII
Amendments

7.0 Ammendments to these Operating Procedures

1. Amendments to these operating procedures shall be made by a majority vote of the Steering Committee. An amendment will be proposed at one meeting, and voted on in a future meeting with the proposed amendment published in the Newsletter in the interim.

IAS SIG
OPERATING PROCEDURES

Article I
Name

1.0 The name of the organization is the IAS Special Interest Group (IAS SIG)

Article II
Purpose

2.0 The IAS SIG is established under the bylaws of the DECUS U.S. Chapter to:

1. Provide a forum for users of the IAS Operating System to exchange ideas, programs, and any other items of common interest.

2. Provide feedback to Digital Equipment Corporation (DEC) on all matters concerning the IAS operating system, related software products, services, policies, and all DEC manufactured computers, peripheral equipment, and other products and services.

Article III
Membership

3.0 Membership Requirements:

1. Any DECUS member is qualified to be a member.

2. Any person qualified to be a member will be accepted as a member either upon submitting a request to the DECUS U.S. Chapter, or personally indicating interest at a Symposium.

3.1 Rights of Members:

1. Members shall be eligible to participate in SIG activities and be members of the Steering Committee.

2. Five or more members of the IAS SIG may, by written petition, bring a motion before a meeting of the SIG Steering Committee.

3. Any member that believes that they are being denied participation in SIG activities shall have recourse by petition to the DECUS U.S. Chapter SIG Council.

Article IV
Steering Committee

4.0 General

1. The IAS SIG shall be administered by the Steering Committee.

2. The Steering Committee committee shall consist of four officers and up to four at large members and up to four past officer members.

3. There will also be a non-voting, ex officio member of the IAS SIG Steering Committee appointed by the RSX SIG to be the RSX liason

with the IAS SIG.

4. Any member of the IAS SIG may be on the Steering Committee and the Steering Committee shall be composed solely of members.

5. The Chairman may act independently on all matters, and shall inform and consult with the Steering Committee as (s)he sees fit. A majority vote of the remaining members shall be required to override decisions of the Chairman.

4.1 Steering Committee Officers

1. The Steering Committee officers shall serve for two years, and be elected by the Steering Committee.

2. The officers are the Chairman, the Newsletter Editor, the Symposia Coordinator, and the Program Librarian.

4.2 At-Large Members

1. The Chairman may appoint up to two at-large members of the Steering Committee.

2. Up to two at large members may be appointed by a majority vote of the Steering Committee.

4.3 Past Officer Members

1. To assist in the transfer of responsibilities to new officers, past officers, when eligible, will remain as members of the Steering Committee (non voting) for one year.

4.4 Duties of the Chairman

1. The Chairman is the chief executive officer of the SIG, and shall chair all Steering Committee meetings and be responsible for directing all activities of the SIG between meetings of the Steering Committee. The Chairman is subject to the review of the Steering Committee, or recall by a majority vote of the members.

2. The Chairman shall appoint one of the Steering Committee to be the liason with the RSX SIG.

3. The Chairman will propose, and recommend for adoption, to the Steering Committee, a resolution to dissolve the SIG when interest in the SIG's activities becomes too small to justify the SIG's existence.

4. The Chairman will represent the IAS SIG's interests in the DECUS U.S. Chapter SIG Council.

4.5 Duties of the Newsletter Editor

1. The Newsletter Editor shall edit and publish the SIG newsletter called "The DeVIAS Letter".

2. In the event that the position of Chairman becomes vacant, the Newsletter Editor shall temporarily assume all duties of the Chairman except that of Steering Committee appointments until a permanent Chairman is elected by remaining members of the Steering Committee.

3 The Newsletter Editor will represent the IAS SIG's interests in the DECUS U.S. Chapter Communications Committee.

4.6 Duties of the Symposia Coordinator

1. The Symposia Coordinator is responsible for the planning ond scheduling of IAS sessions at DECUS symposia.

2. In the event that both the position of Chairman and Newsletter Editor become vacant, the Symposia Coordinator shall temporarily assume all duties of the Chairman except that of Steering Committee appointments until a permanent Chairman is elected.

3 The Symposia Coordinator will represent the IAS SIG's interests in the DECUS U.S. Chapter Symposia Committee.

4.7 Duties of the Program Librarian

1. The Program Librarian shall be responsible for the collection and distribution of IAS specific software via the DECUS tape copy program, with the RSX SIG.

2. In the event that the positions of Chairman, Newsletter Editor and the Symposia Coordinator become vacant, the Program Librarian shall temporarily assume all duties of the Chairman except that of Steering Committee appointments until a permanent Chairman is elected.

3 The Program Librarian will represent the IAS SIG's interests in the DECUS U.S. Chapter Library Committee.

4.8 Vacancy in Office

1. Should the Chairman vacate her(his) office by resignation, disability, or ineligibility or impeachment, a new Chairman shall be elected by a majority vote of the remaining members of the Steering Committee.

2. Should any other officer vacate his(her) office by resignation, disability, or ineligibility or impeachment, the Chairman shall appoint a replacement.

Article V
Elections

5.0 Election of Officers

1. Officers shall be elected by the Steering Committee at the first Steering Committee meeting in every even numbered year, but not less than two years from the recognition of the SIG.

5.1 Appointment of the At-Large Members

1. The two at-large members appointed by the Chairman are intended to assist the Chairman in discharging his or her duties. They will serve until discharged by the current Chairman.

2. The two at-large members appointed by vote of the Steering Committee shall be elected by the officers at the first Steering Committee meeting in every odd-numbered year, but not less than two years from the recognition of the SIG.

5.2 Past Officer members

1. Outgoing officers, unless ineligible or elected as an officer or appointed as an at-large member, or impeached from office shall become a Past Officer member of the Steering Committee for one year.

5.3 Impeachment of Officers

1 In accordance with Article III, the Steering Committee will accept any motion to remove an officer of the IAS SIG. The motion will be presented in the next Newsletter along with the comments of the remaining Steering Committee members and a request that members file a vote on the motion within 30 days. Should a majority of the respondents, comprising at least 1/4th of the membership at the time of the Newsletter's distribution agree to the removal, the officer is impeached, and must be replaced by election by the members, as described below.

5.4 Nominations

1. Should an officer be impeached, or all Steering Committee officer positions become simultaneously vacant, nominations for that (those) position(s) will be accepted by the Newsletter Editor, or the person designated to function in that capacity. Nominations will be signed by three members and accompanied by a letter indicating willingness to serve. All members may vote by returning a ballot published in the Newsletter. The nomimee receiving the most votes will be elected and take office immediately.

Article VI
Meetings

6.0 General Meetings

1. Business Meetings shall be scheduled at the Spring and Fall DECUS U.S. Chapter Symposia.

6.1 Steering Committee meetings

1. The Steering Committee shall meet by phone prior to each general meeting, or at the Chairman's request, and shall also meet at each DECUS U.S. Chapter Symposium.

Article VII
Amendments

7.0 Amendments to these Operating Procedures

1. Amendments to these operating procedures shall be made by a majority vote of the Steering Committee. An amendment will be proposed at one meeting, and voted on in a future meeting with the proposed amendment published in the Newsletter in the interim.

Date: 17-Oct-1986 03:58 EST
From: Michael Robitaille
ROBITAILLE
Dept: IAS SIG Steering Ct'ee
Tel No: (703) 556-7400 x431

TO: See Below

Subject: Revised Operating Procedures for the IAS SIG

Bill:

Attached to this memo is my chop at revising the IAS SIG's Operating Plan to conform to my understanding of the guidelines of the SIG openness Task Force of the SIG Council.

Since I was not present at the SIG Council Meeting in Seattle - I wasn't even a SIG chair then - I would appreciate any comments you may have concerning it.

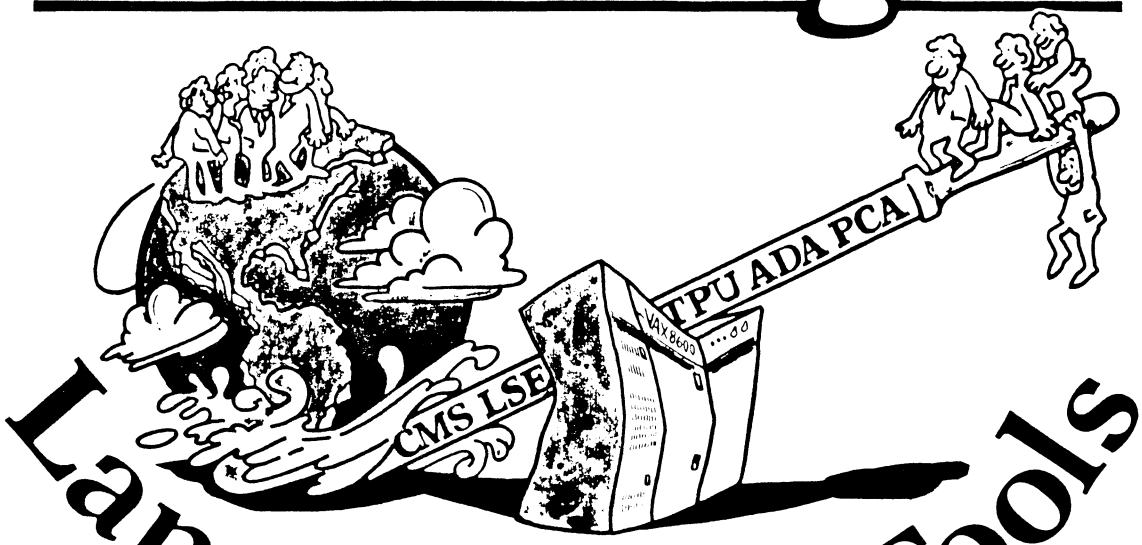
Be advised that the last section of the Operating Plan is unchanged and provides for a mechanism to ratify the changes by the IAS SIG Steering Committee. I will also (by copying this memo to the IAS SIG Steering Committee) request that they review it as well in order that a) other issues that may necessitate changes be effected now, and b) the amendments be approved by a DCS vote after publication in DeVIAS.

Yr Hmbl Srvt,
Mike R.

Distribution:

TO:	Bill Brindley	(BRINDLEY)
CC:	Bob Curley	(CURLEY)
CC:	Robert B. Mack	(MACK_B)
CC:	Michael Reilly	(REILLY_M)
CC:	D.I. Reno	(RENO)
CC:	Mike Robitaille	(ROBITAILLE)
CC:	John Roman	(ROMAN)
CC:	Tim Leisman	(LEISMAN)

Leverage



Languages & Tools

TABLE OF CONTENTS

EDITOR'S NOTES

Just a few submissions for this issue of *Leverage*, but I believe they may be of great interest. First is the long-awaited L&T SIG Masters Directory. These are knowledgeable people who have volunteered their time to help you with questions. If you can use their assistance, please read the section preceeding the list carefully, and then take advantage of this service. If you feel that **you** can help as a master, a Master's Application is included at the back of the Newsletters.

Also at the back of the newsletters is a Wishlist form for L & T. Please note that there is a new Wishlist coordinator, and a new address to submit items. Please give it some thought, and make some submissions to the Wishlist.

The second submission for this issue is Wayne Sewell's trip report from the Nashville symposium. Since I did not make it to the Nashville symposium, I found this to be a great overview of the activities there. Wayne submitted his report from the San Francisco symposium also, and I would like to thank him for taking the time to share his information with the rest of the SIG members who don't always attend symposia.

Finally, we have another installment of Earl Cory's *Fun With DCL*. It's a long one but a goody. Hopefully I've retyped it correctly, but once again, any DCL mistakes are probably mine.

Al Folsom
Leverage Editor

TECO WORKING GROUP

The Languages and Tools SIG is pleased to announce that a TECO working group was formed at the Nashville Symposium. It will be chaired by Mark J. Hyde. He is actively seeking TECO freaks to represent all DEC operating systems, including 36-bit,12-bit, and PC. Those wishing to join can contact him on DCS (HYDE) or at the address listed under the MUMPS steering committee.

FUN WITH DCL

This column is a feature of the Languages and Tools section of the combined newsletter. Each month I will present some DCL commands or command procedures that have been found to be useful by me (or you) in software development. Useful hints and tricks that may be done with DCL will be included.

I am not restricting this to VAX/VMS. RT-11, RSTS/E, RSX-11M, all have DCL to some level. I encourage each of you to send us any DCL procedures, symbols, one-liners, etc. that you find to be useful. Address your suggestions to me or Al Folsom, the *Leverage* newsletter editor.

In VAX/VMS V4 we were given the ability to change our prompt. I am sure that by this time everyone has a favorite prompt that they use.

This feature of VMS may be used in many helpful ways. For instance, I use it on my systems to display the user's DECnet node, device, and default directory.

When a user changes default directory, his prompt is also changed. I call this procedure CD.COM, like UNIX's change directory. The definition:

```
$ CD :== @[directory]:CD.COM
```

must be placed in the user's LOGIN.COM file or SYLOGIN.COM.

Following is the is the command procedure, CD.COM, in full. Following the procedure are notes for the various sections of code. The procedure is long. It includes several sections that truncate the prompt in case it is too long. If you do not need the node name or device included in the prompt, remove the section of code from note (C) up to (K).

```
    $!                          Title: CD.COM
    $
    $       V = F$verify(0)
(A) $       If P1 .NES. "" the SET DEFAULT 'P1'
    $
(B) $       ESC[0,8]            = 27
    $       Bold_on             = ESC+"[1m"
    $       Bold_off            = ESC+"[0m"
    $       Max_length          = 23
(C) $       Node                = F$Getsyi("NODENAME")
    $       Device              = F$trnlmn("SYS$DISK")
    $
(D) $ CHECK:
    $       If F$length("'device") .LT. 6 then GOTO CONCAT
    $
(E) $       Place               = F$locate("$", device)
    $       If Place .EQ. F$length(Device) then GOTO NEXT
    $
```



```
(F) $       Temp                = F$extract(Place+1,F$length(device),device);
    $
(G) $ UP_THERE:
    $       If F$length("'temp") .GT. 6 then GOTO THERE
    $
(H) $       Device              = Temp
    $       GOTO CONCAT
    $
(I) $ THERE:
    $       Place               = F$locate("$", temp)
    $       If Place .EQ. F$length(temp) then GOTO NEXT
    $
    $       Temp                = F$extract(Place+1,F$length(temp),temp)
    $       GOTO UP_THERE
    $
(J) $ NEXT:
    $       Device              = F$extract(0,F$locate(":",device),device)
    $       Device              = F$trnlmn("'device")
    $       GOTO check
    $
(K) $ CONCAT:
    $       First_part          = Node+"::"+"'device"
(L) $       Cur_dir             = First_Part+F$directory()
(M) $       If F$length(Cur_Dir) .LE. Max_length then -
                                GOTO CHANGE_PROMPT
    $
(N) $       Abbrev_dir          = Cur_dir
    $       Abv_Length          = 'Max_length'-'F$length(First_part)'-4
    $
(O) $ FIND_PERIOD:
    $       Period              = F$locate(".", Abbrev_dir)
(P) $       If Period .EQ. F$length(Abbrev_dir) then -
                                GOTO ABBREV_PROMPT
    $
(Q) $       Abbrev_dir = F$extract(Period+1, -
                                F$length(Abbrev_dir)-1, -
                                Abbrev_dir)
(R) $       If F$length(Abbrev_dir) .GT. 'Abv_length' then -
                                GOTO FIND_PERIOD
    $
(S) $ ABBREV_PROMPT:
    $       If 'F$length(Abbrev_dir)' .GT. 'Abv_length' then -
                                Abbrev_dir = F$extract(0,-
                                Abv_length-1,-
                                Abbrev_dir)+"]"
(T) $       Cur_dir             = First_part+"[..."+"'Abbrev_dir"
    $
(U) $ CHANGE_PROMPT:
    $       Set Prompt="'Bold_on'"'Cur_dir' "'Bold_off"
    $       V                   =F$verify(V)
```

Following are comments on the above procedure.

(A) The first thing the procedure does is check P1. If P1 is not null, the user's default directory is changed. If it is null, then only the prompt will be changed to reflect the current directory.

(B) Next, define constants to be used.

(C) Find the node and disk name.

(D) Parse out the device name, allowing only a maximum of six characters.

(E) Locate the "$" In HSC Disks names. If it isn't an HSC disk go to the next part.

(F) Take the non HSC part of the device name and place it in a temporary string.

(G) Check its length. If it is greater than six characters, go to label THERE to continue parsing.

(H) Else assign the temporary string to the device name and continue.

(I) Continue parsing until the device name is six characters long.

(J) Use only up to the ":" for the device name, translate it if it is a logical name, and go up to CHECK to check it's length.

(K) Concatenate the node name and device to form the first part of the prompt.

(L) Determine the current directory and concatenate it to the node and device name.

(M) Check the length of the prompt string. If it is less than the maximum, then we are finished.

(N) Since the prompt is too long, we will have to delete part of it. To do this we will start taking out directory names starting at the root. First determine the number of characters we are allowed.

(O) First, find the first "." in the directory string.

(P) If there are no "."s in the directory string, it will have to be truncated.

(Q) Extract everything from the first "." onward.

(R) Check the length. If it is still too long, go back and find the next ".".

(S) Now that we have a directory string that is short enough, pu in a "[" and "]" to make it look lie a directory.

(T) Concatenate that to the device and node name.

(U) Now change the prompt to our new string.

As an example, I have just logged on to the system and my prompt is the familiar "$". I type:

```
$ CD
```

My prompt becomes:

```
EATON::DJA3:[CORY.COM]
```

To change to another directory:

```
EATON::DJA3:[CORY.COM] CD [GEORGE]
EATON::DJA3:[GEORGE]
```

After setting my default directory to [GEORGE], my prompt is changed to reflect my new default directory. The Bold_on and Bold_off characters inserted into the prompt help to delineate it.

Earl S. Cory

EATON Corporation
31717 La Tienda Drive
Westlake Village, California 91359

LANGUAGES & TOOLS SIG

MASTERS DIRECTORY

May 22, 1987

Those listed below have agreed to answer questions from users, normally by telephone, on the products or subjects listed beside their names. Expertise is generally in VAX software, unless otherwise noted. A State Code in braces {} follows each name to help the user locate Masters in appropriate geographical areas. Complete addresses and phone numbers appear in an alphabetical list following this directory.

The list will become fuller as time goes on.[1] We have included in this list all the L&T products (and several other L&T areas of interest), even where no Master has yet volunteered. We hope this encourages volunteers, especially in those areas. Not all DEC products are listed here, of course. Other Special Interest Groups cover software not included in the Languages & Tools Masters List, although a few non-L&T products are mentioned to accommodate individual Masters with interests broader than L&T's. Mumps is included by special request of the Mumps SIG, as a service to Mumps users.

The expertise of these volunteer Masters overlaps; you may find it necessary to call more than one. Please remember that these Masters can provide you with brief assistance, not with long-term support. Some Masters are professional consultants who have agreed to donate their time and talent in their areas of expertise; it is not L&T's intent to provide a reference service for consultants, and any instance of unwanted commercialism should be reported to the L&T Masters Coordinator (see below). Neither L&T nor DECUS make any claim that the information you receive will necessarily be correct or complete.

Please also notify the L&T Masters Coordinator of any errors in the entries in this Directory, or if you experience real difficulty in your effort to obtain help through this list. **Please note that this list expires three months from the date appearing above. After that time, please consult a more recent issue of the Newsletter for a current list.**

If you can participate as a Master yourself, please fill out the Masters Program application which you will find in the back of the Newsletter, or, at Symposium, in the L&T Information Folder or in the Campground. Submit it to the L&T Campground Host during symposium or mail it to:

Dena Shelton, L&T Masters Coordinator, Cullinet Software, Inc., 2860 Zanker Road, Suite 206, San Jose, CA 95134; (408)434-6636

[1] Especially in the commercial languages—Cobol, Basic, Dibol, and RPG—which have only recently been included under the Languages & Tools SIG.

•Ada

Donald E. Amby {WI}	•Ada •C •CMS •EDT •LSE •MMS •Pascal •Runoff & DSR •TPU
William Graham {AZ}	•Ada •CMS •Debug •Fortran •Runoff & DSR •TPU
Richard Wallace {OH}	•Ada •C •Fortran •Pascal

•APL

Richard Golden {IL}	•APL (& non-DEC APL)

•Basic

•Basic Plus 2

Joel Garry {CA}	•Basic Plus 2
Christopher Thorn {NY}	•Basic Plus 2 •EDT •Kermit •Runoff & DSR

•Bliss

Lawrence J. Kilgallen {MA}	•Bliss •TECO

•C

Donald E. Amby {WI}	•Ada •C •CMS •EDT •LSE •MMS •Pascal •Runoff & DSR •TPU
Fred Avolio {MD}	•C
Dale Hites {IL}	•C •Macro
Jim Maves {CA}	•C •CMS •LSE
Teri McNamara {MN}	•C (under CP/M, First Systems, & VAX) •Config Mgmt
Joseph A. Pollizzi, 3rd {MD}	•C •CMS •MMS •SCAN
Richard Wallace {OH}	•Ada •C •Fortran •Pascal

•CMS

Donald E. Amby {WI}	•Ada •C •CMS •EDT •LSE •MMS •Pascal •Runoff & DSR •TPU
Earl Cory {CA}	•CMS
Joel Finkle {IL}	•CMS •Fortran •MMS •Pascal •Test Manager •TPU
William Graham {AZ}	•Ada •CMS •Debug •Fortran •Runoff & DSR •TPU
Jim Gursha {NY}	•CMS •MMS •SCA
Howard Holcombe {NJ}	•CMS •MMS •Project Mgmt •Fortran
Jim Maves {CA}	•C •CMS •LSE
G. Del Merritt {NJ}	•CMS •Config Mgmt •Emacs •MMS •Fortran
Joseph A. Pollizzi, 3rd {MD}	•C •CMS •MMS •SCAN
George Scott {NJ}	•CMS •Config Mgmt

Jay Wiley {CA} •CMS •LSE •Test Manager •Fortran

•Cobol

•Cobol Generator

•Config Mgmt

Mark Kidwell {TX} •Config Mgmt
Teri McNamara {MN} •C (under CP/M, First Systems, & VAX) •Config Mgmt
G. Del Merritt {NJ} •CMS •Config Mgmt •Emacs •MMS •Fortran
George Scott {NJ} •CMS •Config Mgmt

•Debug

William Graham {AZ} •Ada •CMS •Debug •Fortran •Runoff & DSR •TPU

•Dibol

•Document

•EDT

Donald E. Amby {WI} •Ada •C •CMS •EDT •LSE •MMS •Pascal •Runoff & DSR •TPU
James Meeks {TN} •EDT •EVE •TPU
Christopher Thorn {NY} •Basic Plus 2 •EDT •Kermit •Runoff & DSR
Allen Watson {NJ} •EDT •EVE •Runoff & DSR •TECO •TPU •VAX Notes

•Emacs

G. Del Merritt {NJ} •CMS •Config Mgmt •Emacs •MMS •Fortran

•EVE

Jef Kennedy {OH} •EVE •TPU
David Medvedeff {NY} •EVE •Fortran •TPU •VAX Notes
James Meeks {TN} •EDT •EVE •TPU
Dennis Thury {TX} •EVE •Pascal •TPU
Allen Watson {NJ} •EDT •EVE •Runoff & DSR •TECO •TPU •VAX Notes

•Forth

John Lundin, Jr {VA} •Forth (under CPM, CPM86, MSDOS, VAX)

•Fortran

Donna Calhoun {TN} •Fortran •Runoff & DSR
Joel Finkle {IL} •CMS •Fortran •MMS •Pascal •Test Manager •TPU
William Graham {AZ} •Ada •CMS •Debug •Fortran •Runoff & DSR •TPU
Howard Holcombe {NJ} •CMS •MMS •Project Mgmt •Fortran
Noah Kaufman {MA} •Fortran (& F77)
Scott Krusemark {OH} •Fortran
David Medvedeff {NY} •EVE •Fortran •TPU •VAX Notes
G. Del Merritt {NJ} •CMS •Config Mgmt •Emacs •MMS •Fortran
John Miano {NJ} •Fortran
Paul Plum {PA} •Fortran (& F77)
Andrew Potter {NY} •Fortran •Runoff & DSR •VAX Notes
Lindsay Todd {NY} •Fortran •PL/I
Richard Wallace {OH} •Ada •C •Fortran •Pascal
Jay Wiley {CA} •CMS •LSE •Test Manager •Fortran

•Kermit

Christopher Thorn {NY} •Basic Plus 2 •EDT •Kermit •Runoff & DSR

•LaTeX

Barbara Beeton {RI} •TeX •LaTeX
Kent McPherson {MI} •LSE •TeX •LaTeX

•LSE

Donald E. Amby {WI} •Ada •C •CMS •EDT •LSE •MMS •Pascal •Runoff & DSR •TPU
Jeff Boes {MI} •LSE •MMS •SCAN
Jim Maves {CA} •C •CMS •LSE
Kent McPherson {MI} •LSE •TeX •LaTeX
Lyle Sutton {MD} •LSE •Test Manager
James Thompson {WA} •LSE
Jay Wiley {CA} •CMS •LSE •Test Manager •Fortran

•Macro

Dale Hites {IL} •C •Macro

•MMS

Donald E. Amby {WI} •Ada •C •CMS •EDT •LSE •MMS •Pascal •Runoff & DSR •TPU
Jeff Boes {MI} •LSE •MMS •SCAN

Joel Finkle {IL}	•CMS •Fortran •MMS •Pascal •Test Manager •TPU
Jim Gursha {NY}	•CMS •MMS •SCA
Howard Holcombe {NJ}	•CMS •MMS •Project Mgmt •Fortran
G. Del Merritt {NJ}	•CMS •Config Mgmt •Emacs •MMS •Fortran
Joseph A. Pollizzi, 3rd {MD}	•C •CMS •MMS •SCAN

•Modula II

Jack Davis {TN}	•Modula II

•MS-DOS

Wayne Sewell {TX}	•MS-DOS •Pascal (including realtime use) •Web

•Mumps

Mark V. Berryman {CA}	•Mumps

•Pascal

Donald E. Amby {WI}	•Ada •C •CMS •EDT •LSE •MMS •Pascal •Runoff & DSR •TPU
Joel Finkle {IL}	•CMS •Fortran •MMS •Pascal •Test Manager •TPU
Thomas Lane {TX}	•Pascal
Scott Sewall {MN}	•Pascal
Wayne Sewell {TX}	•MS-DOS •Pascal (including realtime use) •Web
Dennis Thury {TX}	•EVE •Pascal •TPU
Richard Wallace {OH}	•Ada •C •Fortran •Pascal

•PCA

•PL/I

Steven Duff {CA}	•PL/I
Matthew Madison {NY}	•PL/I
David K. Ream {OH}	•PL/I •SCAN
Lindsay Todd {NY}	•Fortran •PL/I

•Project Mgmt

Howard Holcombe {NJ}	•CMS •MMS •Project Mgmt •Fortran

•RPG

•Runoff & DSR

Donald E. Amby {WI}	•Ada •C •CMS •EDT •LSE •MMS •Pascal •Runoff & DSR •TP
Donna Calhoun {TN}	•Fortran •Runoff & DSR
William Graham {AZ}	•Ada •CMS •Debug •Fortran •Runoff & DSR •TPU
Andrew Potter {NY}	•Fortran •Runoff & DSR •VAX Notes
Christopher Thorn {NY}	•Basic Plus 2 •EDT •Kermit •Runoff & DSR
Allen Watson {NJ}	•EDT •EVE •Runoff & DSR •TECO •TPU •VAX Notes

•SCA

Jim Gursha {NY}	•CMS •MMS •SCA

•Scan

Jeff Boes {MI}	•LSE •MMS •SCAN
Joseph A. Pollizzi, 3rd {MD}	•C •CMS •MMS •SCAN
David K. Ream {OH}	•PL/I •SCAN

•Software Project Mgr

•TECO

Lawrence J. Kilgallen {MA}	•Bliss •TECO
Allen Watson {NJ}	•EDT •EVE •Runoff & DSR •TECO •TPU •VAX Notes
Phil Wettersten {OH}	•TECO

•Test Manager

Joel Finkle {IL}	•CMS •Fortran •MMS •Pascal •Test Manager •TPU
Lyle Sutton {MD}	•LSE •Test Manager
Jay Wiley {CA}	•CMS •LSE •Test Manager •Fortran

•TeX

Barbara Beeton {RI}	•TeX •LaTeX
Kent McPherson {MI}	•LSE •TeX •LaTeX

•TPU

Donald E. Amby {WI}	•Ada •C •CMS •EDT •LSE •MMS •Pascal •Runoff & DSR •TPU
Joel Finkle {IL}	•CMS •Fortran •MMS •Pascal •Test Manager •TPU
William Graham {AZ}	•Ada •CMS •Debug •Fortran •Runoff & DSR •TPU
Jef Kennedy {OH}	•EVE •TPU
David Medvedeff {NY}	•EVE •Fortran •TPU •VAX Notes
James Meeks {TN}	•EDT •EVE •TPU

Steven Szep {NY}	**•TPU**
Dennis Thury {TX}	**•EVE •Pascal •TPU**
Allen Watson {NJ}	**•EDT •EVE •Runoff & DSR •TECO •TPU •VAX Notes**

•VAX Notes

David Medvedeff {NY}	**•EVE •Fortran •TPU •VAX Notes**
Andrew Potter {NY}	**•Fortran •Runoff & DSR •VAX Notes**
Allen Watson {NJ}	**•EDT •EVE •Runoff & DSR •TECO •TPU •VAX Notes**

•Web

Wayne Sewell {TX}	**•MS-DOS •Pascal** (including realtime use) **•Web**
JR Westmoreland {UT}	**•Web**

L&T MASTERS ALPHABETICAL LISTING

Donald E. Amby, Delco Systems Operations, PO Box 471 M/S 1A21, Milwaukee, WI 53201; Phone: (414)768-2682
Fred Avolio, 8300 Professional Place, M/S DCO/913, Landover, MD 20785; Phone: (301)-731-4100
Barbara Beeton, American Mathematical Society, PO Box 6248, Providence, RI 02940; Phone: (401)272-9500
Mark V. Berryman, Plessy Peripheral Systems, 17466 Daimler Ave, Irvine, CA 92714; Phone: (714)261-9945
Jeff Boes, Lear Siegler, 4141 Eastern SE, M/S 121, Grand Rapids, MI 49508; Phone: (616)241-8157
Donna Calhoun, Computer Engineering, 704 S. Illinois Ave, PO Box 3174 Oak Ridge, TN 37831; Phone: (615)483-0000
Earl Cory, Eaton Corporation, 31717 Latienda Drive, Westlake Village, CA 91359; Phone: (818)706-5385
Jack Davis, Philips Home Interactive Systems, 1111 North Shore Dr, Knoxville, TN 37919; Phone: (615)558-5206
Steven Duff, Software Factory, 2401 E. 17th St, Suite 190, Santa Ana, CA 92701; Phone: (714)542-9155
Joel Finkle, G.D. Searle, 4901 Searle Parkway, Skokie, IL 60077; Phone: (312)982-8010
Joel Garry, Beck Computers, 5372 Long Beach Blvd, Long Beach, CA 90805; Phone: (213)428-2894
Richard Golden, Systems Manager, 300 S. Riverside Plaza, Suite 1054, Chicago, IL 60606; Phone: (312)-930-9800
William Graham, U.S.A.F., OC-ALC/De t. MMECZA, PO Box 11037 Tucson IAP, Tucson, AZ 85734; Phone: (602)573-2391
Jim Gursha, VP & Director of Information Services, Financial Strategies Group, Goldman Sachs & Co, 85 Broad St, 25th Floor, New York, NY 10007; Phone:

(212)902-3009
Dale Hites, Health Chicago, 1624 Bay Ct (home), Naperville, IL 60565; Phone: (312)961-0825
Howard Holcombe, Front & Cooper St, Camden, NJ 08105; Phone: (609)338-4946

Noah Kaufman, JEOL (U.S.A.), Inc. 11 Dearborn Road, Peabody, MA 01960; Phone: (617)535-5900
Jef Kennedy, PBR & Assoc, 6549 Park N A-1, Solon, OH 44139; Phone: (216)349-5877
Mark Kidwell, Software Design Engineer, Texas Instruments, PO Box 869305, M/S 8435, Plano, TX 75086; Phone: (214)575-3547
Lawrence J. Kilgallen, Box 81, MIT Station, Cambridge, MA 02139-0901
Scott Krusemark, Systems/Software Consultant, 8473 Daisywood Ave. NW, North Canton, OH 44720; Phone: (216)499-6251
Thomas Lane, Software Engineer, 1310 Electronics Dr, Carrellton, TX 75006
John Lundin, Jr, Technical Asst., Academic Computing, University of Richmond, Richmond, VA 23173; Phone: (804)289-8652
Matthew Madison, RPI Center for Interactive Computer Graphics, Rensselaer Polytechnic Institute, Troy, NY 12180-3590; Phone: (518)271-2606; Net Addr: MADISON@RPICICGE.BITNET
Jim Maves, Eaton Corp, IMSD, 31717 Latienda Dr, Box 5009, Westlake Village, CA 91359; Phone: (818)706-5375
Teri McNamara, Senior Software Engineer/System Manager, Data Card Corp, 111 Bren Road West, Minnetonka, MN 55343; Phone: (612)931-1792
Kent McPherson, 4141 Eastern Ave, SE M/S 121, Grand Rapids, MI 49508; Phone: (616)241-7458
David Medvedeff, Rochester Institute of Technology, One Lomb Memorial Dr, Rochester, NY 14623; Phone: (716)475-2810; Net Addr: DJMACC@RITVAX.BITNET

James Meeks, Trevecca College, 333 Murfreesboro Road, Nashville, TN 37210; Phone: (615)248-1236
G. Del Merritt, Senior Member Technical Staff, Computer Sciences Corp, 304 W. Rt 38, Moorestown, NJ 08057; Phone: (609)234-2595
John Miano, Berlex Laboratories, 110 East Hanover Ave, Cedar Knoll, NJ 07927
Paul Plum, Consultant, PO Box 1208, Lansdall, PA 19446; Phone: (215)275-7216
Joseph A. Pollizzi, 3rd, Space Telescope Science Institute, 3700 San Martin Dr, Baltimore, MD 21218; Phone: (301)338-4901
Andrew Potter, Rochester Institute of Technology, One Lomb Memorial Dr, PO Box 9887, Rochester, NY 14623-0887; Phone: (716)475-6994; Net Addr: AWPSYS@RITVAXD.BITNET
David K. Ream, LEXI-COMP, 26173 Tallwood, N. Olmsted, OH 44070; Phone: (216)777-0095
George Scott, Senior Computer Scientist, Computer Sciences Corp, PO Box N, Moorestown, NJ 08057; Phone: (609)234-1100
Scott Sewall, College of St. Catherine, 2004 Randolph St. St. Paul, MN 55105; Phone: (612)690-6405

Wayne Sewell, Software Engineering Specialist, E-Systems, Garland Div, M/S 53730, Box 660023, Dallas, TX 75266-0023; Phone: (214)272-0515 Ext 3553
Lyle Sutton, Space Telescope Science Institute, 3700 San Martin Dr, Baltimore, MD 21218; Phone: (301)338-4509
Steven Szep, Chase Manhattan Bank, New York Plaza, New York, NY 10081; Phone: (212)676-2085
James Thompson, Applications Analyst, 20000 Cypress Way, Alderwood Manor, WA 98036; Phone: (206)775-8471
Christopher Thorn, Elias Sports Bureau, 500 Fifth Avenue, Suite 2114, New York, NY 10110-0297; Phone: (212)869-1530
Dennis Thury, Texas Instruments Incorporated, Box 801 M/S 8006, McKinney, TX 75069; Phone: (214)952-2066; Net Addr: CSNet: THURY MCCORE@TI-EG
Lindsay Todd, Rensselaer Polytechnic Institute, Troy, NY 12180-3590; Phone: (518)276-6751; Net Addr: TODD@RPICICGE.BITNET
Richard Wallace, AFWAL/AADE, WPAFB, OH 45433; Phone: (513)255-4448 Ext. 8654
Allen Watson, Watson Consulting, Inc., 3 River St, Suite 30, Little Ferry, NJ 07643; Phone: (201)641-2468/(212)902-2984
JR Westmoreland, Custom Software Products, Utah Power & Light, 1407 West North Temple, Annex 6/208, Salt Lake City, UT 84116; Phone: (801)535-4784
Phil Wettersten, Borden, Inc, E. Broad St, Columbus, OH 43215
Jay Wiley, Manager, Advanced Technology Group, Bechtel Power Corp, 12440 E. Imperial Highway, Norwalk, CA 90650; Phone: (213)807-4016

Trip Report for DECUS Spring Symposium, Nashville

E. W. (Wayne) Sewell
Software Engineering Specialist
E-Systems, Garland Division
Mail Stop 53730, P.O. Box 660023
Dallas, Texas 75266-0023

May 14, 1987

Contents

1 Introduction

During the week of 27 April-1 May, I attended the Digital Equipment Computer User's Society (DECUS) Spring Symposium, which was held at the Opryland Hotel in Nashville. This is one of the nicest facilities we have visited in recent years. There were plenty of large meeting rooms and there were no problems with overcrowding that I was aware of. Only a few sessions had to be repeated and there was no difficulty finding a room in which to reschedule them.

This trip report is written from the viewpoint of one person, with all of the biases and personal interests that implies. Most of the sessions I actually attended were sponsored by the Languages and Tools SIG, the rest falling primarily under the VAX SIG.

2 The Source Code Analyzer / Language Sensitive Editor

The SCA is a new product announced by DEC at the San Francisco Symposium, although it is only now becoming available. It is used for interactive analysis of program source files and is especially useful in the case of an executable which is built of many different modules in different source files in different languages.

One of the major functions of the SCA is cross-referencing symbols used within one or more programs. With one command, the user can display all occurrences of a variable in the entire image. Each instance is displayed on a line by itself and contains the module in which the variable appears, the line number, and whether it is a read or write reference (such as an assignment to the variable). Procedures and functions can be displayed the same way, with indication of whether the instance is a definition or a call. Wild cards are accepted in symbol names, allowing a search for multiple symbols at the same time. Multiple queries can be in progress simultaneously. The VIEW CALL_TREE command can be used to show the caller/callee relationship between procedures. For instance, it can indicate that Procedure X calls Procedure Y which calls Procedure Z. The call tree can be inverted, showing that Procedure Z is called by Procedure

Y which is called by Procedure X. Recursive procedures are identified. For procedures and functions with parameters, the SCA will verify that the formal parameters on the procedure definition match the actual parameters on the call. This cross-check is inherent in strongly-typed languages like Pascal (except for **EXTERNAL** procedures in another language), but not in most other languages. For most languages, errors of this type are not found until run-time, and are often hard to diagnose even then.

The SCA can also display include and environment file dependencies.

Several of the compilers (including Pascal) have been modified to generate the binary files which contain the information SCA uses to do its job. The information created by the compilers is stored in a library maintained by SCA, which SCA later reads to perform the analysis. Multiple libraries can be used with a search list. For instance, there could be a project- or group-wide library with a personal library for each developer.

The Language Sensitive Editor (LSE) has been modified (Version 2) to work with the SCA. Operation of the LSE with SCA is similar to that of the Review function of LSE. When the SCA function to show cross-references is used from within the LSE, the list of references appears in a window. When a particular reference is selected with the cursor (using the **GOTO SOURCE** command), the source file containing that reference is automatically pulled in by the LSE and displayed in another window with the cursor sitting on the variable being referenced. The window comes up in a read-only state, but the user can make it writable with a **SET MODIFY** command. If the code displayed is not the desired reference, the user can move back to the other window, where the list of references is still displayed, and select another reference. The new reference, which may be in another source file, will also be retrieved and displayed. When looking at a variable, the **GOTO DECLARATION** command can be used to transfer directly to the source line where the variable is defined.

In addition to SCA support, there are several features which have been added to LSE for Version 2. Comment support is one of the most useful additions. For stand-alone comment blocks (as opposed to end-of-line comments), the LSE can reformat the comments to a more readable form without interfering with the delimiters. If a program contained the following comment block

```
(* this comment block has *)
(* been *)
(* edited many times *)
```

and the LSE fill command were used, the block would change to

```
(* this comment block has *)
(* been edited many times *)
```

An interface to CMS has been added so that modules can be reserved, etc., directly from the editor. A EVE keypad has been added (the current version uses an EDT keypad). Editor templates have been added for the VMS System Services, making the calls to these procedures much less of a hassle, since you no longer have to remember or look up every last parameter.

For more information, see the Languages and Tools Session Notes, page 7, 102, 140, 250, and 345.

3 VAX Project Manager

As its name implies, this new product is a tool for project management. All of the standard methodologies appear to be supported (Gantt charts, PERT charts, etc.). Since I am not familiar with project management myself, I can't really evaluate the product. Unfortunately, I wasn't able to bring back any information on it. Digital usually doesn't submit product announcements to the Session Notes (I suppose because the notes are submitted a month or so before the Symposium and can't be recalled if they decide not to make the announcement for some reason), and when I went down to the System Software booth at the exhibit hall, the DEC rep told me that the brochures for the product were supposed to have been there, but didn't get loaded onto the truck due to a logistics error.

4 Documentation Systems

4.1 TEX/LATEX

There wasn't much on either TEX or LATEX this time, especially compared to the last symposium. The only session I am aware of that even mentioned TEX was a session sponsored by the Unix SIG describing a program that creates a permuted index of the hundreds of commands present in PLAIN TEX. It was only a half-hour session and was very easy to miss in the SAG[1]. I didn't realize it was there until afterward.

4.2 VAX Document

A documentation preparation system used internally by Digital to produce the VMS document set has been described at previous symposia under the name Standard Digital Markup Language (SDML). This system is now available as a product called VAX Document. Basically it is a markup language which provides a higher level of abstraction than a typesetting language such as TEX. Rather than raw formatting commands as in TEX, it provides a structure to a document. In this sense, it is closer to LATEX in philosophy, although the command syntax is simpler.

Formatting commands in VAX Document, called *tags*, are identified in the text by angle brackets (as in `<title>`).

Unlike LATEX documents, which can only be printed on a bitmapped device, rough drafts generated by VAX Document can be printed on a regular line printer because the high-level source is internally converted to Runoff source if the target device is a line printer. If the target is a laser printer or other raster device, the conversion is to TEX source rather than Runoff.

Unfortunately, the TEX portion of VAX Document cannot be used in a standalone mode; it is embedded too deeply within the system. It would be nice to be able to transfer TEX source or DVI files created by VAX Document to other TEX systems on other types of computers, but apparently this is not possible. First, the internal TEX system has been modified to an unknown degree. Second, a package of proprietary TEX macros is integral

[1]Sessions at a Glance—tabular listing of all of the sessions on a particular day

to the system. Possibly the intermediate TEX source created by VAX Document could be intercepted and transferred to another TEX system (possibly not; it could be passed in a mailbox), but without the macro definitions, the other TEX system could not process the document. The macros are not provided in source form; it is highly likely that they have been converted to binary form and preloaded into the executable image.

If that weren't enough of an inhibition, VAX Document doesn't use the standard Computer Modern fonts normally supplied with TEX. The only TEX fonts it uses are the math fonts.

Oh, well. At least the documents created should be TEX-quality (at least the positioning of the text—the non-CM fonts are an unknown quantity), even if you are restricted in what you do with them.

5 Upcoming Changes to the VMS Exec

5.1 Image Structure

The structure of the VMS Executive image is being changed drastically for Version 5. Basically it is being broken up into smaller executables, which are to be installed as privileged shareable images. The base image will ultimately become nothing but a vector table pointing to the actual procedures in the smaller images. The intent is for the base image to never change again while VMS is on the face of the earth. For more information, see the VAX Session Notes, page 157.

5.2 Synchronization

Version 5 contains massive changes in the way synchronization is done in VMS. This especially affects things like device drivers, which must synchronize with things like the I/O data base. Presumably, all of the DEC-supplied drivers will track the changes, since they are part of VMS, but any drivers that are user-written or supplied by a third-party will not and will be completely inoperative until modified. At the very least, changes will have to be made to the driver source code; possibly the logic may have to change.

The changes are being implemented to support full multiprocessing. In the current version of VMS, synchronization is done with two basic mechanisms: MUTEXes and raising IPL (this discussion does not consider higher-level mechanisms like the lock manager). The method normally used by drivers is to raise IPL. Each of the various resources used in the driver has a particular IPL used to lock it. When a driver wishes to lock a resource, such as the I/O data base, it raises its IPL to the level designated for the resource, effectively locking out all other processes which wish to use the same resource. When the exclusive access is complete, the IPL is lowered, allowing other processes to execute.

The problem with IPL as a synchronization method in a multiprocessing environment is that raising IPL on one CPU will lock out other processes on the same CPU but has no effect on other CPUs. It is for this reason that the multiprocessor machines currently implement a master-slave relationship rather than one in which all CPUs are equal. Only the master machine is allowed to run in privileged modes at all, and the slave is locked into user mode only. Since all processes have to use kernel mode at some point, only a CPU-bound type of process will spend enough time in user mode to really utilize the slave processor.

DEC considered other synchronization techniques, such as interrupts between processors and message-passing schemes, but the overhead of such methods was prohibitive.

The method finally implemented is referred to as a "spinlock". A spinlock is a data structure in a particular memory location which is used to control access to a particular resource. It is accessed via the interlocked TEST BIT SET AND SET instruction (BBSSI), which is uninterruptable. When a processor attempts to lock a resource, the BBSSI instruction is used on the spinlock. Either the lock is successful and the processor has exclusive access to the resource, or the acquisition fails and the processor hangs in a loop retrying the lock (this is the "spin" part—also known as "busy waiting", which is generally associated with more primitive operating systems).

On the plus side, spinlocks are more flexible than IPL levels regarding the particular resource locked. There are a fixed number of IPL levels, but since the spinlocks are only structures in memory, each entity to be locked can have a dedicated spinlock. For instance, IPL 8 is used for many different

things in the current version of VMS, including the entire I/O database. In Version 5, the I/O database is broken up into several modular parts, each of which has its own spinlock. Instead of locking all of the database, only the part actually needed is locked. Also, each physical device and/or controller has its own spinlock and can be locked independently of all other devices. The greater granularity provided by spinlocks would hopefully allow more overlapping operations and reduce the number of resource conflicts.

As far as existing software is concerned, the bulk of the changes to support the new synchronization method would be in replacing the code currently used to change IPL (occurances of DSBINT, ENBINT, SETIPL) with the new macros for manipulating spinlocks. It might be possible to continue using the old macros (I vaguely remember the comment that the old macros would generate the new spinlock code, but this would not take advantage of the greater granularity of the spinlocks, forcing the driver to continue locking the entire I/O data base instead of just a portion of it. The new macros are:

- LOCK — Acquires a spinlock by resource index
- UNLOCK — Releases a spinlock by resource index
- DEVLOCK — Acquires a spinlock by device index
- DEVUNLOCK — Releases a spinlock by device index

The code generated by LOCK and DEVLOCK is virtually identical—the difference is in how the parameter identifying the spinlock is evaluated. Additionally, since a spinlock is just a memory location, there is probably no reason why user-defined spinlocks cannot be used. Of course, a new pair of macros would have to be generated, since the LOCK and UNLOCK macros would not be able to find the user spinlock.

The spinlock is not just a bit in memory, although that is all that would be required for the interlock itself. It is actually a data structure containing information about the resource being locked and who the current owner is. The fields contained in the spinlock structure were not itemized in detail, but one significant field is the ID of the locking CPU. This is a number rather than just a bit, indicating that more than two CPUs are supported,

an important fact when the fabled n-processor machines in future systems are taken into account.

While device drivers are the most obvious victims of the synchronization changes, *any* kernel mode code which uses IPL for synchronization will be affected.

6 VAXCluster Failure and Recovery

There was a good session on what occurs during cluster transitions. This presentation had been made in the past for Australian DECUS, but had never before appeared at an American Symposium.

The discussion covered basic VAXcluster concepts, including the components that make up the cluster (both the regular CI version and the NI, or Ethernet, versions were discussed), the events that occur during cluster operation, failure and recovery strategies, and failure scenarios.

The hardware components of VAXclusters include the CI communications port and the HSC storage system. The software part consists of the Lock Manager, the Configuration Manager, System Communication Services (SCS). the disk class driver (DUDRIVER), and the port driver (PADRIVER).

Failures are detected by periodic polling, port timeout, the port sanity timer, virtual circuit timeout, and a node stop ("last gasp") datagram if the failing machine is able to send one. Prior to CI microcode Rev 7 and with VMS V4.3, the interval before failure detection was over 100 seconds; after V4.4 and Rev 7, the interval should be 10–30 seconds.

The sequence of events for several scenarios was discussed:

- Addition of a member into the cluster
- Loss of a member
- Loss of a HSC

For more information, see the VAX Session Notes, page 131.

7 DCL Internals

I attended an entertaining session on the internal operation of DCL. Several of the data structures DCL builds in the P1 space (process-permanent data, per-command data, indirect command file structure, command token pointer, symbol table, SPAWN parameter block, and SPAWN context block) were described, plus the mechanics of symbol substitution. The handling of the SYS$INPUT and SYS$OUTPUT files for nested command procedures and the handling of Control-Y, SPAWN and ATTACH were also discussed. Instructions for looking at DCL structures with SDA were provided.

The DCL symbol table is organized as a 256-entry hash table, and each entry is the head of a doubly-linked list (queue) which contains all of the symbols which hash to that value. Each queue is sorted first by symbol name, then by local/global indicator (local symbols come first), finally by procedure nest level (inner levels precede outer levels). The address of the hash table is stored in the process permanent structure. For more information, see the VAX Session Notes, page 320.

8 Multiprocessing

There was a good session on parallel processing techniques. The particular algorithms presented will work on the current dual-processor machines and also on the future n-processor machines that Version 5 will make possible (see Section 5.2).

The thrust of the session was on program decomposition, which doesn't mean that the program is rotten and decomposing (although this may be the case for some programs), but instead refers to the procedure of reorganizing the program into a form that can take advantage of parallel processing. Under the current version of VMS, only number-crunching programs can gain any benefit from multi-processor systems. The basic technique is to break up the task to be performed into chunks which can be processed independently. For instance, in some matrix operations the calculation to be performed on a particular element can be done independently of the other elements. The next step is to create a master process and multiple

slave processes with a shared global section containing the data they will be working on concurrently. Initialization processing is performed by the master, then the parallel processing begins. In the case of an operation performed on elements of an array or matrix, allocation of processing is accomplished by a function that returns the index of the next unprocessed element. All of the processes execute a loop that processes one element during each iteration. At the beginning of the loop the allocation function is executed to provide the index of the element to be processed. In the meantime, the other processes are executing the same loop, but processing different elements. The simplest way to do the allocation is with an interlocked increment instruction.

The ratio of the number of instructions used for switching to those used in the processing itself is critical, because the other processors are locked out while the switching code is being executed. If the ratio is too large or too small, there is a lot of idle time, since the processors can't interleave properly, and total throughput for the program suffers. According to the speaker, a machine instruction ratio of 20 to 1 produced optimal results in tests (or was it 10 to 1?—I can't remember). If the per-element algorithm is too simple to provide the correct ratio (the processing code is not much longer than the switching code), the algorithm can be reworked to perform more processing during each iteration. For instance, each iteration can process a row of the matrix rather than just one element.

9 Pascal Wizards' Session

The Wizard Session contained some interesting techniques for making VAX Pascal do things it doesn't officially support. Some of the "tricks" discussed were:

- Using VAR Class-S parameters
- Using immediate-mode parameters
- Using features supported only for EXTERNAL procedures with regular native Pascal procedures
- Opening a file by File ID rather than by name

- Special tricks for AST procedures
- ISAM files with variable-length records (ISAM is supported only for fixed-length records)

For more information, see the Languages and Tools Session Notes, page 328.

10 Using Digital Tools

There were a couple of sessions relating user experiences with the Digital tools for program development. The pros and cons of each of the tools were discussed, based on actual project usage. For more information, see the Languages and Tools Session Notes, page 285 and 341.

11 Extended Pascal Standard

There was a BOF[2] on the work currently in progress by ANSI to define the new standard for Extended Pascal. It was basically a status report presented by John Reagan, Project Leader for VAX Pascal and Digital's representative on the ANSI Committee. Most of the people present received a copy of the working draft. The comment period for the draft ended May 1st, and the committee will meet again this summer. If no technical changes are made, then a formal document will be developed. If technical changes to the standard are made, a new draft will be made and the review cycle will begin again.

The current version of the draft contains some interesting extensions that would make standard Pascal a viable language (while VAX Pascal is one of the best implementations in existence, the current version of *Standard* Pascal is less than useless). This is only the draft, *not* the approved standard. Some of the new features being proposed are:

- Conformant arrays, dynamically-generated types, and other dynamic allocation

[2]Birds of a Feather—informal session for people interested in a particular topic, scheduled dynamically during the Symposium

- Separate compilation with interface modules, similar to the definition modules of Modula-2 (it is important to note that Extended Pascal, like Modula-2 and Ada, does full type-checking across module boundaries).
- Enhanced string handling (almost any form of string handling would be "enhanced" when compared to Standard Pascal). Some of the string enhancements are:
  - Variable length strings.
  - Comparison of strings, with or without padding.
  - Substring handling.
  - **readstr** and **writestr** (**read** and **write** to/from a string variable rather than a text file)
  - Concatenation.
- Extended file support.
  - Binding of file variables to external filenames in a generic way (virtually all compilers do this already by necessity; it's just not in the current standard).
  - Direct access file handling.
- Structured constants.
- Structured function results.
- Automatic variable initialization within any block, not just the outer level.
- Date and time primitives.
- Ranges and **otherwise** clause for **case** statement (finally!!!)
- Underscore in identifiers.
- Complex numbers.

- Type inquiry (make a variable have the same type as a parameter, constant, or another variable).
- Non-decimal-based numbers (hex, octal, etc).
- Extensions to set operations.

Most of these new features are already present in VAX Pascal, in some cases implemented exactly the same. Others will be added if they are present in the final standard, since Digital is committed to supporting the standard.

12 Pre-Symposium Seminar

I attended the pre-Symposium seminar entitled "Advanced Device Driver Techniques", which was sponsored by the VAX SIG. There was a lot of useful information disclosed in the seminar, especially regarding communications drivers. The standard driver model for VMS expects to have only one I/O operation in progress at a time (witness the "busy" flag in the Unit Control Block), but communications drivers, especially those implementing OSI network protocols, cannot be restricted that way.

The following topics were among those discussed:

- the status of multiple concurrent I/O operations can't be stored in the UCB as is normally done; instead the IRP itself can be used.
- many of the standard macros and procedures will not work with alternate fork blocks; the user must write his own
- how to extend the UCB and IRP
- cloned UCBs
- buffer management
- smart devices
- multiple queues for waiting I/O requests

- network protocols
- "state machine" drivers
- timer queue elements (TQEs)
- exception handling (including timeout)
- debugging techniques

Mention was made of the VAX BI and the special problems of implementation it represents.

13 Tape Copy Project

There was a session which discussed the contents of the VAX SIG tape created from the submissions received at this symposium. Each of the attendees who had contributed something to the tape said a few words about his submission. The tape includes, among other things, the latest and greatest versions of the Kermit[3] communications program, GNU[4] EMACS, a VAX disassembler, an object file manipulation program, DCL access to $ENQ/$DEQ, structured programming macros for MACRO, a DCL compiler, lots of TPU procedures, and much more. Unfortunately, the tape won't be available for several months, since it takes that long for it to go through the distribution channels. (The tape for the San Francisco Symposium is just now becoming available to the LUGs.)

For the first time, the Languages and Tools SIG has produced its own Symposium tape. It contains submissions from the Dallas and San Francisco Symposia. The distribution channels will be basically the same as for the other SIG tapes. Included on the Languages and Tools tape is a full implementation of TEX and LATEX (including fonts for the LN03), GNU EMACS, EVE Plus, DEPROC (TEX and LATEX support for the DECUS Proceedings), LSE templates for LATEX, TPU and LSE itself, the ICON language, index and glossary generators for LATEX, EDT Plus, and much more.

[3]No, it's not a acronym—it really *is* named after the frog.

[4]GNU's Not Unix, a recursive definition if there ever was one.

14 Working Groups

The working groups are a mechanism which has developed within DECUS over the last several years, and continue to increase in importance. A working group is a collection of people who have an interest in a particular topic and are willing to expend some effort in supporting that topic. The VAX SIG has had working groups for years, but Languages and Tools is only now starting to create some. The goal is for each language to have a working group, plus groups for certain topics pertaining to tools, such as configuration management and tools integration.

A working group acts as a clearinghouse for issues related for the particular topic in which it is interested and interfaces with Digital to represent the needs of the user base. Ideas for new sessions and Pre-Symposium Seminars are cultivated, as are wish list items.

I am the chair of the new Pascal Working Group, which was created during the Symposium. Because of the last-minute scheduling of the BOF (and probably because it was held on Friday), Denny Thury of DFWLUG was the only person to attend the meeting itself and sign up for the working group, although I recruited a few others later. More details on the working group and the list of members will be published separately in Leverage (Languages and Tools newsletter). Anyone who has ideas for a session relating to Pascal at future Symposia or who wishes to join the working group, is invited to contact me.

15 Masters Program

There was a BOF about the Languages and Tools Masters Program, which is a pool of expertise available to the DECUS membership on various Digital and public domain products. The L & T Masters are individuals who have knowledge of a particular language or tool and are willing to answer questions about its use. The Masters are not necessarily gurus, although they are considered experts. Long term support is not the function of the Masters Program; one-shot questions are the intended use of the resource, since the masters are volunteers. The Masters Directory will be published in the newsletter as soon as all of the new recruits from Nashville are compiled

and added to the list. As a fallout of the BOF, a decision was made to define a second level of expertise, tentatively referred to as Journeyman. The journeymen don't have extensive knowledge of a product, just a basic understanding sufficient for simple questions. Normally, an individual will be listed as a master in one or more categories and as a journeyman in several others.

16 VT300 Series Terminals

Quite a few of the terminals in the Exhibit Hall available for logging onto the various machines were of the new VT300 series. I never got a chance to use one myself, but from what I heard, the graphics were about five times faster than the 200 series.

17 Digital's Lemons

One extremely popular session was a list of Digital products that should never have existed. I didn't actually attend the session, so I don't know which products were listed, but apparently it was an entertaining and informative session (obviously, this was a user-sponsored session, rather than one presented by Digital). Rather than simply DEC-bashing, the speaker presented persuasive arguments as to exactly what was wrong with the products and why they were half-baked when inflicted on an unsuspecting public.

18 The Rest

As usual, there were a great number of sessions that I wanted to attend, but could not due to scheduling conflicts or sheer exhaustion. These included DECNET internals, Ethernet, GKS, Autogen, Kermit,VAXELN, HSC Architecture, disk driver and disk server internals, Local Area VAXclusters, VAX Performance Advisor, terminal servers, paging adjustment, SPM and system tuning, network performance, the DEC/SNA Gateway, VAX security, and many others. Many of these are included in the session notes.

GLOBAL ACCESS

MMP

MUMPS SIG DECUS U.S. CHAPTER

"If you don't want it in the paper, don't let it happen."

DECUS

G L O B A L A C C E S S

Volume 1, No. 1 July, 1987

MUMPS means you never have to say you're sorting.

$VIEW(Editor)

Welcome to GLOBAL ACCESS, the latest go-round of the MUMPS SIG newsletter (we're going to try it this way for a while, and see if it will stick this time...). My name is Mark J. Hyde, and it will be my duty to keep turning the crank (or pumping the bellows, or even--maybe--pounding the keyboard) until a newsletter comes out the other end. I am a long-time DECUS member, with roots in the 12-/18-/36-bit universe. However, my current major diversion is MUMPS systems: specifically, a VAX and a PDP-11 running InterSystems MUMPS, and numerous "industry standard" PCs running various implementations. I am thus ready and able to keep you up to date in the world of MUMPS.

I would immediately like to abscond with this opportunity to mitigate a confusion factor which currently exists in the Big One: there is another Mark Hyde writing the *Notes on Notes* column for Office Automation. He is not me. His middle initial is "H," and I will be very careful to use mine ("J") in this newsletter and whenever my byline appears in other newsletters. Thusly can you, dear reader, tell us apart and appropriately direct your cheers or jeers.

$DATA

In a significant departure from the time-honored tradition of interpretive MUMPS, Greystone Technology Corporation has introduced a MUMPS compiler for the VAX. The company states that the compiler conforms to the 1984 standard, and is a true optimizing compiler which generates native VAX machine code. Other features include MUMPS extensions to allow access to VMS system services, a set of source language debugging tools, and a library of powerful utility routines. Greystone Technology is located at 8 Lakeside Office Park, Wakefield, MA 01880.

$HOROLOG

July 24	Submission deadline for Sept. newsletter
Dec. 7-11	Fall '87 Symposium; Anaheim, CA
Feb. 8-12, 1988	Canadian '88 Symposium; Toronto
May 16-20, 1988	Spring '88 Symposium; Cincinnati, OH

$ORDER("Symposium")

This month's topic is usage of the word *symposium.* In an organization which is so dependent upon constant use of this word, it is shameful that so few people use it correctly. Symposium is the singular form; *symposia* is the plural. A simple trick for determining which to use in any given situation is to substitute the word *congress* (which follows more typical English rules). If congress fits, use symposium. If (and only if) *congresses* is required, use symposia. Some examples:

> I attended both U. S. Symposia [congresses] last year.
> We will have a regional symposium [congress] this year.
> I missed the Fall '85 Symposium [congress].

Please note that reference to a specific date (Fall '85, last spring's, ...) always requires the singular. This is the most common misuse, and is glaringly obvious if one tries congress (no one would say "the Fall '86 congresses"). The case of the *Symposium Committee* is less clear. It can be argued that the plural is correct because the committee deals with more than one symposium. However, it is apparent that this usage contributes to the confusion among the membership, and thus should be amended to "Symposium Committee" for their benefit.

As an aside, it is also the case that English is generally moving away from "latinate" plurals. *Antennas* is gradually replacing *antennae,* and *condominiums* never had anything to replace. Thus, it seems that we may be moving toward the era of (heaven forfend!) *symposiums.*

[In the interest of greater clarity in writing, and precision in communication, this column will address itself to examples of grammar or usage that I find to be egregiously poor, sloppy, trendy, or corpocratic.--Ed.]

$NEXT

At press time, preparations are in full swing for the 1987 MUMPS Users' Group Conference, to be held in Atlanta, June 8-12. Since most of the SIG steering committee (excepting, alas, your Editor) will be in attendance, the September newsletter should contain both a report on the conference, and a synopsis of news from the MUMPS Development Committee.

Longer term plans for articles include MUMPS benchmarking, a comparison of the multiplicitous implementations in the MS-DOS world, and an evaluation of the Greystone compiler.

```
$NEXT($ORDER)="Product"
```

$RANDOM

If recent economic news has you down, just remember that the downswing in the upturn only means that things are getting less worse more slowly.

RSX

MULTI-TASKER

The RSX Multi-Tasker
June, 1987

"We've Never Been Proud"

Fine Realtime Commentary Since 1975

Table of Contents

Food for Thought

"Pitiless indeed are the processes of Time and Creative Thought and Logic; they respect the convenience of none nor the love of things held sacred; agony attends their course. Yet their work is the increasing glory of a world - the production of psychic light - the growth of knowledge - the advancement of understanding - the enlargement of human life - the emancipation of Man."

- *Cassius J. Keyser*
Mathematical Philosophy

The Editor's Corner

Bruce R. Mitchell

As we go to press, the Spring Symposium is only two days away. For obvious reasons, we don't have any reports from the Symposium in this issue. But hang in there, they're coming.

We have a pretty fair selection of articles this month. Something that we've all wanted to see for a long time is here - an analysis of weak points in the RSX security system. Read it. Apply it. If you don't - don't bitch about your system being penetrated.

Jim Preciado returns with "It's In the Code" this month, to discuss some seldom-seen but very useful options in the M-Plus Executive code. There is a patch to the M-Plus Data Cache Manager. From Gary Maxwell comes an article on removing the seldom used and mostly useless FMS code from Indirect.

And Justin L. Hewser, infamous RSX system programmer and man about town, shows up again with another of his gems of wisdom in "The Notebooks of Justin L. Hewser".

The Editor is pretty tired this month, what with prepping for the Spring Symposium. The Editorial is short. But you, the users, asked for this one specifically. So here it comes.

----- New Tricks -----

The RSX SIG has been around for a long time. It is one of the oldest SIGs, if not the oldest SIG, within DECUS. Next year the SIG celebrates its 15th anniversary.

A few years back (Multi-Tasker article: "Whither Goest the SIG?") I said that ten years is a long time for a computer architecture to endure, much less an operating system. Well, we're looking at our fifteenth anniversary next year. DEC's competitors have never done as well, despite all their "hot, fast, new industry standard hardware to be announced sometime next year".

Each year at the "Woods" meetings the SIG leadership examines the directions for the SIG's future. Those directions have become somewhat circumscribed recently, but there's still a lot of room for us. "Small" (by VMS standards) multitasking systems are still needed in many applications.

Certainly we are not ready - as some have suggested - to amalgamate all PDP-11 users into a "16-bit SIG". Remember what happened to the 12-bit SIG? "What 12-bit SIG?" you ask? Right.

Users ask me: "Where is RSX going?" I can only answer: "Where are you going?" Usually the response to that is - "To VMS."

VAXes and VMS are not the answer to all problems, contrary to what DEC keeps trying to cram down our throats. They never were and still are not a real-time answer. Look at the size of the VMS Executive. It still takes time to execute code, and more code takes more time to execute. This is one of the reasons that RSX is faster than VMS for real-time applications.

Even so, it would be a canard to state that RSX is still the universal solution. There was a time when this was true. It is no longer so. But RSX systems still front-end the plant VAX clusters and control the machinery out on the floor. Why? Because they're good at it. They're inexpensive. A lot of proven and available software exists.

If Digital would wake up and decide to sell PDP-11s as a commodity, not as systems, they would kick 68000s right out of the real-time control market. Can a $3,000 11/73 be built and sold? Yes. It would be the death knell for VMEbus systems. Will Digital do it? Probably not. Why not? Because there's no market for it. Why is there no market for it? Because Digital isn't making, selling and pushing such a machine.

"Does RSX have a future?", you ask me ...

RSX has a future. How long a future, and what kind of a future, is not apparent yet. Ask me in another 10 years, when V8.0 of M-Plus comes out.

Where is RSX going? That depends on DEC and on us. As long as we keep buying PDP-11s, RSX has a future.

----- Submitting Articles to the Multi-Tasker -----

Please submit machine readable media if you can. RX01, RX02, RX50, or 9 channel magtape at 800 or 1600 BPI are best. Any RSX volume format is acceptable except ROLLIN or PRESRV. ANSI, BRU and DOS FLX formats are well-liked by the Editor's tape drive.

The Editor can now Kermit articles into the Multi-Tasker host. The reverse is unfortunately not true; the Multi-Tasker host is normally an isolate. If you want to submit via Kermit, call beforehand with (1) phone number, (2) login for the host machine and (3) system uptimes.

Submissions which aren't machine readable take longer to get into print. The editor is lazy and types mass quantities only once a month when progress reports are due.

If you preformat a submission in RUNOFF, please set page size 58,80; left margin 10; right margin 75; and, when changing margins, use incremental changes rather than absolute. The editor blesses you for the consideration.

Send all submissions to:

Bruce R. Mitchell
Machine Intelligence and Industrial Magic
PO Box 816
Byron, MN 55920
(507) 775-6268

----- And That's The Way Things Are -----

... this month in Pool Lowbegone, where opposition to VAX/Elan is strong, real switch register front panels are good-looking, and the number of systems that come in on time and under budget is above average.

It's in the Code

James J. Preciado
Crossfield - Composition Systems, Inc.

Some RSX-11M-Plus Executive Debugging Hooks

Even Digital must debug code. Astonishing thought, but true. Sometimes they even make it easy on us users.

Recently I discovered some hooks in the RSX-11M-Plus Executive that are extremely useful to device driver writers and other developers of privileged code. One of these hooks allows for the the detection of double forks by drivers or other privileged code. Another provides for monitoring of a specific location in the Executive and halting of the system when it changes.

Usually when confronted by problems such as these, one writes a separate program to grab some POOL, load in a trap routine to catch the particular problem at hand, and point a piece of the Executive to this little mousetrap. This becomes a bit more complicated when dealing with an I and D space Executive where these code fragments must reside in ICB POOL and the Executive code itself is write protected.

In M-Plus V3.0, hooks exist in the Executive code for detection of Executive data changes as well as catching the addition of an item to the Executive fork list when the same fork block is already there (a double fork). These hooks are assembled based on conditional assembly flags. SYSgen does not define these flags, but users can easily supply them by pausing during SYSgen and editing [11,10]RSXMC.MAC. If you are generating environments for driver and other privileged code development, it makes sense to turn these hooks on so that they can be called upon when needed.

The Double Fork Test

Fork handling is done in Executive module SYSXT, in routine $QFORK. All routines that need to add a fork block to the end of the Exec's fork list end up in $QFORK.

In this modules is code conditionalized on the flag R$$FRK. What the conditional code does is look through the existing fork queue and ensure that the block being added is not already in the fork block list. If it is found, a double fork is being attempted and the system is bugchecked. Yes - RSX has a bugcheck facility but that, as they say, is another story. When the system is bugchecked, register 4 points to the offending fork block.

Watchpoint Support

RSX-11M-Plus has a watch point facility. This facility allows monitoring of a location in the Executive for a specific value and breakpointing the processor when its value changes.

Again in SYSXT are two pieces of code conditionalized on the flag R$$WPT. The first is in the routine $DIRSV which is called by the DIRSV$ macro from an Exec directive or switch to system state (SWST$, SWSTK) call.

The first piece of watchpoint related code is a one-liner as follows:

```
MOV     (R5), $WPLST            ; SAVE ADDRESS OF LAST SYSTEM ROUTINE
```

What's happening here is that the address of the system state routine which called $DIRSV is saved. In this way we have a record of where the system came from.

Next, at the label $DIRXT we have the actual watchpoint code.

```
        CMP     $WPVAL, @$WPADR ;;;WATCHPOINT STILL OK ?
                                ;;;ONE OF THE FOLLOWING BRANCHES
                                ;;;SHOULD BE NO-OPED TO START THE
                                ;;;WATCHPOINTS
$WPBR:  BEQ     1$              ;;;IF EQ STILL OK
        BNE     1$              ;;;IF NE STILL OK
        BPT                     ;;;OOOOOOPPPPPSSS!!!
1$:                             ;;;REFERENCE LABEL
```

If you are wondering where you find the data items being used in the above instructions, you find them in system common (module SYSCM) right before the Executive common APR table. They are:

```
$WPLST::.WORD   0               ;LAST SYSTEM STATE ROUTINE CALLED
$WPVAL::.WORD   0               ;VALUE TO WATCH FOR OR AGAINST
$WPADR::.WORD   $WPVAL          ;PLACE TO WATCH FOR IT
```

What we have here is a way of monitoring a location in Executive data space for a change in a specific value. This location is checked each time the system attempts to leave system state and exit to the user level.

In order to do this, RSX must empty the fork queue. The watchpoint code is executed upon return from the system state routine called and before each entry of the fork queue is removed and called.

This allows us to see if the system state routine just called or one of the entries removed from the fork queue trashed the location we are interested in.

Since the watchpoint code is driven from a table in SYSCM, all that has to be done is to load the address we want to watch into location $WPADR and the value to check against into $WPVAL. A NOP instruction is then deposited at either $WPBR, to catch the system when the location changes to the value in $WPVAL, or at $WPBR+2 to catch the system when the location changes from the value in $WPVAL.

Since these data locations and the instruction labels are global Executive symbols, their values can be obtained from the Executive map and loaded using the MCR OPEN command. No special program is required! If you are dealing with an I and D space Executive, then the watchpoint data table should be opened in data space (/KNLD) and the branch instruction should be opened in instruction space (/KNLI).

How to Enable These Hooks

As said, all of this code already exists in the M-Plus Executive. We just have to supply the necessary conditional assembly flags. During SYSGEN, answer YES when asked if you wish to pause to edit any files prior to assembly. The edit [11,10]RSXMC.MAC and add the following conditionals before the executive MACROs section:

```
R$$FRK=0                ;CATCH DOUBLE FORKS
R$$WPT=0                ;WATCHER LOCATION
```

Now continue with the SYSGEN and the support will be included. RSX-11M users aren't so lucky. They would have to add this code.

Watchpoint support is a fairly low overhead operation. However, scanning the fork queue at each fork block insertion can have a negative impact on system performance. Adding a branch around this code at a global Executive label that could be NOPed would be nice. In this way, the double fork check could be enabled by the OPEN command in a manner similar to turning on the watchpoint code. (This last item was not my idea but I liked it when I heard it.)

Also, user code may reload $WPLST with more meaningful information during the course of its execution. A piece of key input data or a bit mask showing which portions any suspected privileged code executed are just some useful items that come to mind as being useful information to have available when the system is breakpointed.

RSX-11M-Plus Data Cache Manager Change

James P. Tvrdik
Analysts International Corporation
650 Woodfield Drive, Suite 300
Schaumburg, IL 60195

While installing RSX-11M-Plus Version 3.0 Update D, I noticed (to my great dismay) that disk data caching does not operate in conjunction with the Volume Valid Monitor Program (VVC) published by Gary Maxwell.

What occurs is that VVC causes disk data caching to be deactivated for the system disk. The reason behind this is that the data cache manager (DCM) responds directly to <u>any</u> QIO issued with the IO.STC (Set Characteristics) function code. Since the VVC program uses the IO.STC function to set the volume valid bit, the first issuance of the QIO$ IO.STC directive causes the cache to be purged and deactivated for the system device.

According to Gary, the intent of VVC is to use the IO.STC as a control function (<u>not</u> a transfer function) for avoiding excessive disk head movement.

To correct this problem, I patched the Disk Cache Manager controller source module ([11,10]DCCTL.MAC). The code change causes the requestor taskname to be checked. If it is VVC, then the IO.STC request is queued directly to the driver, thus bypassing the deactivation code.

Note that there are alternate solutions such as using the DCM's bypass subfunction bits in the QIO, but since these depend too much on DEC, they are avoided.

Create the following correction file in UFD [11,40], insuring that you are patching Version 01.04 of DCCTL.MAC:

```
SY:[11,10]DCCTL.MAC;2/AU/-BF=SY:[11,10]DCCTL.MAC;1
-3,3,/;JPT001/
        .IDENT  /01.04A/
-25,25,/;JPT001/
;
;       J. P. Tvrdik    01 April 1987   01.04A
;       AiC, JPT001
;
;       Install checks to verify the requestor task
;       as the Volume Valid Monitor (VVC) program
;       (from DECUS). If it is, the IO.STC function
;       should be passed directly to the driver and
;       NOT interpreted by the Disk Cache Manager
;       (DCM11M).
```

```
;
%
-73,73,/;JPT001/
        MOV     I.TCB(R1), R2           ; Pick up TCB of requestor
        CMP     T.NAM(R2), #^RVVC       ; Is it VVC ?
        BEQ     20$                     ; If EQ, yes
/
```

Now apply the correction file to the source file using the SLP source patcher, and assemble the patched module:

```
SET /UIC=[11,10]
SLP @[11,40]DCCTL.COR
SET /UIC=[11,24]
MAC DCCTL,[11,34]DCCTL/-SP=[1,1]EXEMC/ML,-
-> [11,10]RSXMC/PA:1,DCPRE/PA:1,DCCTL
```

Replace the module into [1,24]RSX11M.OLB, saving the old one if you desire, then re-Taskbuild the Disk Cache Manager (DCM11M):

```
SET /UIC=[1,24]
LBR DCCTL.OBJ=RSX11M.OLB/EX:DCCTL
LBR RSX11M.OLB/RP/-EP/SZ=[11,24]DCCTL
TKB @[200,200]DCM11M
```

Now create and VMR a new system image file:

```
SET /UIC=[1,54]
PIP RSX11M.SYS/NV/CO/BL:1026.=RSX11M.TSK
VMR @SYSVMR
```

Since the Disk Cache Manager is installed as a directive common, a new SYSgen is <u>not</u> required. Simply BOOt the new image, SAVe it and write the boot block.

This change has been implemented on a client's host computer and appears to work quite well. My thanks to Gary Maxwell for his time and assistance.

Removing FMS From INDirect

Gary Maxwell
U.S. Geological Survey
Office of Earthquakes, Volcanoes and Engineering
345 Middlefield Road, M977
Menlo Park, CA 94025

How many people use the FMS-11 interface available with Indirect (...AT., the @ command processor)? For the eight of you who raised your hands, the following won't concern you. But for the rest of us, here is a way to get back some space in your system.

A lot of people don't realize that FMS-11 builds into the task image of Indirect, and that it is big. Really big. But you can remove all the FMS code very simply and reduce the size of the task image to something reasonable.

Create the following SLP correction files using your favorite text editor:

```
[1,40]ICMBLD.COR:

ICMBLD.BLD;2/-AU=[1,20]ICMBLD.BLD;1
-/EXTBUF/,.
.;      .SETN   EXTBUF  4000    ! Space for FMS-11 buffer
        .SETN   EXTBUF  0       ! No space for FMS-11 buffer
-/$FORMS/,.
.;.IFT $MINST   .DATA GBLDEF=$FORMS:0    ; No FMS-11 present
                .DATA GBLDEF=$FORMS:0    ; No FMS-11 present
/


[1,40]ICPCOMBLD.COR:

ICPCOMBLD.BLD;2/-AU=[1,20]ICPCOMBLD.BLD;1
-/ICM:/,.
;ICM:    .FCTR   ROOT-IGTN-IL-GTKN-EXED-FORM-NETW,FDRV
ICM:     .FCTR   ROOT-IGTN-IL-GTKN-EXED-EGCML-NETW
-/ICMFSL:/,.
;ICMFSL:.FCTR    ROOT-IGTN-IL-GTKN-EXED-FORM-NETW,FDRV
ICMFSL:  .FCTR   ROOT-IGTN-IL-GTKN-EXED-EGCML-NETW
-/ICMRES:/,.
;ICMRES:.FCTR    ROOT-IGTN-IL-GTKN-EXED-FORM-NETW,FDRV
ICMRES:  .FCTR   ROOT-IGTN-IL-GTKN-EXED-EGCML-NETW
;
EGCML:   .FCTR   ECM1-ECM2-ECM3
/
```

Now perform the following commands to apply the patches and

```
rebuild Indirect:

SET /DEF=[1,20]
SLP @[1,40]ICMBLD.COR
SLP @[1,40]ICPCOMBLD.COR
SET /DEF=[200,200]
@SYSGEN
```

Go to the non-privileged task build section of SYSgen and build your favorite version(s) of Indirect (ICM, ICMRES, or ICMFSL). You will be pleasantly surprised to find that the new Indirect is 61% smaller than before.

Editor's note: If you feel bold, apply these patches before you do SYSgen. Then you don't have to redo the non-priv task build section.

Response to February "Bag of Tricks"

Editor's note: The following is a response to the February "Bag of Tricks" article on the Extend Task (EXTK$) directive. More comments follow the letter.

sentrol

Sentrol Systems Ltd.
1 - 6325 - 12th Street S.E.
Calgary, Alberta
Canada T2H 2K1
(403) 253-8848
Telex: 03-825784

Postmarked 20 March 87 Alberta C
Rec'd 25 Mar

March 20, 1987

James A. McGlinchey
5 Skyline Drive
Essex Junction, VT
U.S.A. 05452

Dear Sir:

I am disappointed with a statement that you have published in the RSX Multitasker for February, 1987. In the "Bag of Tricks" Section, you stated that a task will load more slowly if it is installed with a maximum increment.

This was true in RSX-11M 3.2/E, it was repaired a long time ago in RSX-11M 4.0 and works properly in recent RSX-11M/PLUS systems. When I say properly, I mean that the LOADR task uses the non-extended size to perform loading. Of course, a hole in the tasks partition must be found which is large enough to hold the entire task first.

In actual fact, a task which is loaded with no increment will be physically loaded into memory in the same amount of time. However, its demand for a maximum increment which cannot be satisfied by merely stretching the top of the task into free memory will cause checkpointing and reloading of the task. Clearly, the task will respond more slowly and cause unnecessary disk activity.

Having taken your "RSX-11M/M+ System Performance Management" seminar, I am surprised by your recommendation to the RSX community.

Yours truly,

SENTROL SYSTEMS LTD.

R. H. Ingles

Roger H. Ingles

RI/ks

The Editor responds:

Yes, it's true. In current RSX systems the Task Loader loads the task using the unextended size, and allocates the additional space. It takes significantly more time for a task to extend using the Extend Task directive than it does to load it already extended.

If a task always requires extra buffer space, it should obviously be built into the task and not obtained through extensions of any kind. If much free memory is available on the host system, and the task often benefits by being installed with an extension, this is preferable to doing runtime extends.

However, it is never safe to assume for a task to assume that it is already pre-extended. It must find out. That's why the GTSK$ and GPRT$ directives exist. Using GTSK$ and GPRT$, the task can determine if it is already extended to its limit. If it is not, it can extend itself as far as necessary.

The February issue was very short of submissions, and Mr. McGlinchey very kindly agreed to let me ghostwrite the original article using his by-line. The Editor is not as expert as some users in the intricacies of RSX, and made an incorrect statement.

I apologize for any inconvenience this may have caused the readership.

Security of the RSX-11 Operating Systems

Thomas R. Wyant III
E. I. DuPont de Nemours & Co., Inc.
Textile Fibers Department
Richmond, VA 23261

This article had its origin as a "hacker's cookbook" for RSX. Therefore, the various routes of entry are discussed in some detail. There are some who believe that this kind of thing gives people ideas, and should not be done. I disagree. I feel that any responsible system manager wants to know what the threats to his system are, so he can counter them.

I have collected here some of the more notable security issues - and non-issues - that I have encountered (and occasionally exploited) in the RSX operating systems. It is not intended to be exhaustive - but then Godel's Incompleteness Theorem implies that no discussion of security can ever be exhaustive. There are, however, two principles that lie behind almost all the issues raised here:

- Know what your users are doing. The best logical security in the world is no good if the users are doing some of the things mentioned here.
- Protect the physical system. This includes CPU, media, and so on. If you don't have physical security, you don't have any other kind, either.

Backup Media

There is an obvious problem with backup media: it is normally off line. Unless it is guarded in some way, it can be spirited out, copied, and returned, leaving no trace of tampering. The situation can be worse if the backup is of the system pack, as the account file resides there.

The possessor of system backup media can read accounts and passwords from it, and thereby gain privileged access to the system. Whether this is a problem depends on how the backup was created, as follows:

- PIP - Account file is present only if explicitly copied.
- PRESRV - Account file always present.
- DSC - Account file always present.
- BRU - Account file is present only if the whole system disk was copied, or if a selective BRU copied it explicitly.

One might think that the M+ user would not be at risk from loss of passwords by this route, as the passwords are encrypted in the file. However, encryption is not complete protection. The passwords can still be obtained by the brute force method of encrypting potential passwords, and comparing the output of the encryption algorithm to what is actually in the account file. This is discussed more fully under "Brute Force".

The only solution is to be aware of which backup media have the account file on them, and control those media.

Front Panel

The front panel (whether switches, pushbuttons, or console processor) available on most PDP-11s represents a very crude but unrestricted means of entry to the system. Even an unsophisticated person can bring a system down simply by pushing the "HALT" button. A sophisticated user can log his terminal on or grant it privileged status in a matter of seconds. The procedure for this is given below:

Taskbuild the system's symbol table using the command

```
>TKB ,TI:/-WI=LB:[1,54]RSX11M.STB
```

By default, any user has the required access.

In the Taskbuilder map, find the Unit Control Block (UCB) address of the terminal to be made privileged. For terminal TTnn:, this address appears as the value of the symbol ".TTnn". For an RSX-11M system, or an RSX-11M+ system without I and D space support, this value is the physical address of the UCB, and you can skip the next step.

For an RSX-11M+ system with I and D space support, the symbol $EXEND appears in the Taskbuilder output. Compute the physical UCB address by:

- Round $EXEND UP to next 32 word boundary (i.e., add 77 octal, then make the last 2 digits zero).
- Subtract 20000 octal (4KW for the ICB pool, which is mapped by both I and D space APR 0).
- Add the value of the symbol ".TTNN".

The result is the physical address of the UCB for TTnn:.

Calculate the physical address of the second unit status word for TTnn: by adding 12 octal to the physical address of the UCB, calculated above.

Halt the processor and examine the second unit status word using console ODT.

Calculate a new value for the second unit status word, which is the old value, but with bit 3 (10 octal) set, to grant privilege. Also clear bit 8 (400 octal) to log the terminal in if necessary. In other words,

new value = ((old value)&177377)!10

Patch complete. Continue the processor.

When we tried this on an 11/70 running RSX-11M+, the CPU was halted for less than 10 seconds. This outage might not be noticed on a lightly loaded system, and on a heavily loaded one, might be attributed to CPU load. The only lasting side effect was that the system's time of day clock was off by the amount of time spent halted; this would not be noticed in most shops.

There are two ways to block this means of entry:

- Restrict access to the front panel. Lock it when it's not in use.
- Protect [1,54]RSX11M.STB as you would the account file. This includes watching backup media, and reprotecting it so that the world has no access.

System Console Terminal

Sometimes, the console terminal is left logged on and privileged. If it is left unattended, anyone can use it for their own purposes. I have seen this problem dealt with by doing a

SET /SLAVE=TI:

at the start of the login command file, and

BYE

at the end. Another approach is to set the console to a custom CLI.

However, this still leaves a window when the system is booted. At system boot, someone at TT0: can issue a control/C as soon as the system comes up, followed by the command

MCR>REM ...AT.

Of course, this means the startup command file does not get run, and the console terminal is left logged on and privileged.

There are two ways to block this means of entry: Restrict access to the console terminal, and/or slave the console in the system image by:

VMR>SET /SLAVE=TT0:

In this case, the console terminal comes up slaved. Note, however, that if you log the console off, it becomes unslaved again.

DECnet

The main problem with DECnet is the DAP utility set (NFT/FAL). If you have enabled access verification nowhere else, enable it for FAL (DECnet object number 17) and you will sleep sounder. Remote task control (object 15) is another good candidate.

Remote Command File Submission

If remote command file submission is available on the local node, the user can submit a command file that issues a

>SET /PRIV=

command on his terminal. This works if the .CMTS. task submits the command file directly to the indirect command processor, and the terminal used is logged on as a privileged terminal. In some M+ systems, it also works if .CMTS. submits the command file to the batch processor, and the submitter specifies a UIC that does not

exist.

Any of the following will help reduce this problem:

- Don't install .CMTS., or remove it after DECnet installs it.

  If you must use .CMTS., use it with the batch processor if possible (and if you are not running the vulnerable version of .CMTS. mentioned above).

- If you must have .CMTS. spawn command files to a terminal, have them executed on a nonprivileged terminal. The TI: of the spawned command file is controlled by a LUN assignment on .CMTS..

Access Control in Global Aliases

If access control information (account and/or password) is embedded in any global aliases, any user can issue the

```
>NCP SHOW ALL ALIASES
```

command, and select the alias which best suits his needs. It is true that DECnet refuses to display the password associated with an alias, but the account number or user name (which is displayed) is enough to tell if a given alias is of interest.

Global aliases with privileged access control are particularly to be avoided. Such aliases can be used (among other things) to read the account file for the designated node with commands similar to:

```
>NFT ACNT.TMP=alias::[0,0]RSX11.SYS
>DMP TI:=ACNT.TMP/AS/OC
```

Avoid the use of global aliases that have access control associated with them.

Worms

In case anyone is still unacquainted with the terminology, a "worm" is a program or other procedure that copies itself from node to node around a network, getting itself executed on each node it visits, and possibly doing (by chance or design) deleterious things along the way.

Unlike VMS, there is no way under RSX for a person off-node to either install a task, or run an uninstalled task other than use remote command file submission, as discussed above. Assuming this rat hole is plugged, I can't get terribly excited about this as a security threat. However, the loading of a Trojan Horse (discussed below, under "Guile and Stealth") is still a possibility.

account file in milliseconds rather than seconds per attempt.

The best way to counter this is to be careful with the account file itself. Longer passwords can also help, as even milliseconds can be stretched to millennia by requiring enough of them.

Guile and Stealth

There are, unfortunately, better ways than brute force to obtain a password. The literature and personal experience suggest that several things can cause problems.

Non-random Passwords

There is a tendency for naive users to select trivial passwords. First names, initials, license numbers, and so forth are easy to guess.

The only solution here is the education of users. A periodic audit of dumb passwords might catch the worst offenders.

Hardcopy Passwords

HELLO allows the password to be specified on the same line as the uic as:

```
>HEL uic/password
```

This is bad news anywhere, but particularly on a hardcopy terminal. The literature suggests that passwords are commonly lost in this way, when someone with an eye out for 14.5 x 11 paper plows through the right dumpster.

There are two possible solutions. One is to educate the user. The other is to patch HELLO (see old SIG tapes) to overprint the login line when this form of login is used.

If DEC would disable this form of login, allow the system manager the option of disabling it, or overprint the line, the problem would be relieved.

Automated Logins

I have seen people who save themselves the trouble of remembering their passwords by programming the login command into a terminal's answerback message. Unfortunately, this allows anyone to use it.

Educate the users. Or cut the terminal's answerback enable

jumper. This practice is fairly easily audited for.

Password Grabbers

Since RSX allows any user access to an unowned device, a nonprivileged user can write a program that simulates the behavior of ...HEL. Being nonprivileged, he cannot of course log his victim in, but he can record the victim's password and then issue an appropriate error message and exit his program. The user then has access to the real ...HEL and can log in normally, believing (the hacker hopes) that he made a typo on his password.

The only general solution is vigilance on the part of the user. Perhaps DEC will consider the option of having logged out terminals allocated to CO:, though after doing this comes the problem of how to get ...HEL to run.

Trojan Horses

Like many hacker's tools, a "Trojan Horse" is a program or command procedure with undocumented functionality that is possibly deleterious to your system. The distinguishing feature of a Trojan Horse is that it requires the unwitting cooperation of someone on the victimized system, as (unlike a worm) it has no means of securing its own execution.

The main sources for unscrutinized code are load media and networks. The prudent system manager is cautioned to be familiar with the origin and content of anything loaded onto his system, and to allow network access only under the same conditions as login access would be allowed. Like all the issues raised under "Guile and Stealth", informed and cooperative users are vital to the defense against this attack.

Things You Haven't Thought of Yet

Some users are sophisticated enough to do unanticipated things to mess you up, or are unsophisticated enough to do unanticipated things to mess themselves up. You may be better off restricting their actions to a list you control.

There are several ways of controlling which programs the user has access to:

A "captive account" can be created simply by setting up the account to be logged on slaved, and using the person's LOGIN.CMD to provide the command interface. Capture can be made almost complete by issuing a

.ENABLE CONTROL-Z

before the first query issued, to prevent "Control/Z" bailout on a .ASK directive. If you omit this, the user will still have a slaved terminal when LOGIN.CMD exits, and you will have to get him straight.

A subtle point in this approach is that the user should have read-only access to both his own login command file and the directory in which it resides - otherwise, a number of seemingly harmless applications (such as mail) allow him to modify his own LOGIN.CMD. I can think of two ways to approach this:

- Protect the user's home directory and the files in it read-only for owner and group. If the user has need to create files, set his default to some other directory in his LOGIN.CMD.
- Have SYSLOGIN.CMD chain to some file other than LOGIN.CMD for a terminal that is logged on slaved.

It is possible to write a CLI that restricts the user to a subset of the full list of MCR commands. A quick way to do this is to obtain CCL (from the KMS/Fusion package, on several recent SIG tapes) and modify it:

- Remove the functionality that spawns unrecognized commands to MCR.
- Clean out the memory-resident command table.
- Remove the search of SYSCLI.CCL for commands.
- Modify the open operation on USERCLI.CCL so that this file can be stored in a protected directory.

The Notebooks of Justin L. Hewser

Gary Maxwell
U.S. Geological Survey
Office of Earthquakes, Volcanoes and Engineering
345 Middlefield Road, M977
Menlo Park, CA 94025

While sitting near the back of a recent Symposium Q&A session I observed a commotion occuring both at the microphone and at the panelists' table of the panelists. Someone named "Justin L. Hewser" was insisting that the RSX group should build the M-Plus batch processor against RMSRES with DAPRES support, so he could batch across the net.

Well, he didn't get the answer he or I wanted, for it sounded

like a good idea to me, too. When I discussed it with Mr. Hewser, he pushed up his dark glasses, looked about nervously, gave me a floppy disk, mumbled something about having to get out of town fast, and hurried away without giving me an explanation of the contents of the disk.

After returning home, I eagerly looked on the disk to see what I had. The following is a condensation of the contents.

*S006. -- Linking the Batch Processor to RMSRES

We have up to twelve batch processors active. BPR links against RMS. Why not RMSRES? Why not DAPRES? Too much of a resident library is really a good thing.

The solution is really easy. Create the following correction file in UFD [1,40]:

```
BPRBLD.BLD;2/-AU=[1,20]BPRBLD.BLD;1
-/PAR=/
        .DATA ;
        .DATA ; Link to RMSRES/DAPRES
        .DATA ;
        .DATA CLSTR=RMSRES,DAPRES:RO
-/GBLDEF=D$ELPS/,.
        .DATA GBLDEF=D$ELPS:1130
-/@BPRRMS.ODL/,.
.;      .DATA @BPRRMS.ODL
        .DATA @LB:[1,1]DAPRLX.ODL
/
```

Change the default job CPU time limit from the incredibly short three minutes to 10 hours by changing the global definition of D$ELPS. You choose any limit in minutes, convert to octal, and use that instead of "1130" above.

Next, patch everything by performing the following steps:

```
>SET /DEF=[1,20]
>SLP @[1,40]BPRBLD.COR
>SET /DEF=[200,200]
>@SYSGEN
```

You can then redo the privileged task builds to build a new BPR and batch across the net.

You can't link it as a multi-user task. No such luck. Everything is in the blank PSECT. Thanks a lot, DEC.

Ah, yes.

Where shall we begin here? Justin's fundamental idea is sound. Assuming that you already use RMSRES for other tasks in the system, linking the batch processor to RMSRES trims about 3.5 Kwords from the task image (a 35% reduction) and trims the same amount from the total physical memory demands on the system. So far, pretty good.

Batching across DECnet is an interesting if not bizarre idea. However, there is more to DECnet remote file access than linking to the Data Access Protocol resident library, DAPRES. Alas, as the reader may discover, all other batch components speak FCS, not RMS. Network file specifications cannot be entered into QUEUE.SYS.

There is also another consideration: Why is Justin mapping RMSRES as a user-mode library? This is M-Plus, remember. His system probably supports supervisor mode. Mapping RMSRES in supervisor mode peels out most of the RMS overhead in the task and often allows "flattening" of a task through reductions in overlay complexity.

And a 10 hour default CPU time limit seems excessive when the default can be overridden at submission time.

There are gains available in the batch processor, but this is not the best way to do it. If you want to build the batch processor against RMSRES in non-supervisor mode, replace the lines:

```
.DATA CLSTR=RMSRES,DAPRES:RO
.DATA @LB:[1,1]DAPRLX.ODL
```

in Justin's procedure above with the following:

```
.DATA LIBR=RMSRES:RO
.DATA @LB:[1,1]RMSRLX.ODL
```

and pick some default CPU time limit which produces a reasonable result.

There is a lot of other information on the floppy. I have just now begun to examine it in detail. Here are some candidates for future columns:

- Cacheing Memory-Resident Disks to Decrease Access Time
- Intertask Communication using CINT$ and PIRQ
- Building Unmapped RSX-11M-Plus for the 11/03

The RSX Multi-Tasker

July, 1987

"By Hook, by Crook, Or by Cytoplasmic Streaming"

Fine Realtime Commentary Since 1975

Table of Contents

Food for Thought

"If you ask me to name the proudest distinction of Americans, I would choose - because it contains all the others - the fact that they were the people who created the phrase 'to *make* money.' No other language or nation had ever used these words before; men had always thought of wealth as a static quantity - to be seized, begged, inherited, shared, looted or obtained as a favor. Americans were the first to understand that wealth has to be created. The words 'to make money' hold the essence of human morality."

- *Ayn Rand*
Atlas Shrugged

The Editor's Corner

Bruce R. Mitchell

I made a big boo-boo last month. By now you probably all know about it. There was indeed a June issue of The Multi-Tasker, but it went in after the deadline for submission. It's not clear whether it made it into the June issue of the SIG newsletters, and I won't know until after the deadline for this issue. If it missed, you are now reading both June and July in the same issue of the SIG newsletters.

Love those symposia ... As usual, RSX presented a full slate of presymposium seminars, technical sessions, meetings, and raised a row as well. VMS users who attended their Magic session will not soon forget that RSX rules at symposia. By the end of the week, most other SIGs were arriving at the RSX suite to present tribute ... or possibly they looked at it as "protection".

You may wonder why the SIG makes such a major effort to become so "visible" at symposia. That is a reasonable question. Aside from

it being a lot of fun, it's easier to get and hold someone's attention if they know who you are, and think you can back it up.

Linda Ziegler and Karen Noel presented sessions at Nashville that deserve special attention and thanks from the entire RSX user community. Linda, of course, is the VAX/RSX implementer. Karen is a new member of the RSX team, the implementer of a shiny new RSX product - Coprocessor RSX.

"So what's coprocessor RSX?" you ask. Coprocessor RSX is probably the most exciting thing to come down the pike since the 11/74 multiprocessor. Take a KXJ11 card. Plug up to 4 of them into the backplane of a MicroVAX II. Make a few additions to M-Plus, add a few drivers on both systems, and what do you have? You have a VAX that can run native mode RSX applications *concurrently* with VMS. Peer to peer. In the same backplane.

Do we need this? Yes! Think of the applications for it! In most industrial applications, the front-end PDP-11s feed process information into VAXes handling a workcell or process line. Why put in separate 11s with cabinets and disks when you can put them into the backplane of the VAX? Let the two systems share peripherals. It cuts the cost of a typical process control system by 50 percent, at a guess.

We need this. Not just as the RSX SIG, but as system implementers. It's exciting. It's got possibilities nobody has thought of yet. It's not expensive. And it promises to extend the lifetime of the PDP-11 for a long, long time to come.

All of us owe Karen and Linda a big vote of thanks for the time and effort they invested bringing this project to fruition. I hope that I may convince them to write an article about it for a future issue of the newsletter.

Thanks are due to *all* the members of the RSX implementation team who attended Nashville. For some reason we seem to take their ongoing efforts and attendance at symposia for granted. We're footing the bill for them, you say? That is true ... but remember this: True RSX wizardry is getting scarce. And, sure as the carp flop upriver to spawn (it was a dry spring in Minnesota) it's not easy to keep that type of people around when DEC offers jobs that promise more recognition, higher pay, and state of the art equipment.

For all the people at DEC who support RSX and fight to keep it alive - Cathy Ziegelmiller, Dick Day, Brian McCarthy, and their entire group - on behalf of the RSX SIG, thank you. We don't say it often enough.

Everything I want to talk about won't fit into this column, so I guess I'll cut it off. But not without a few editorial comments first. No, you don't have to get out the fire extinguishers - no flaming this month (well, not much ...)

----- By-Laws and By-Ways -----

At Nashville, the SIG distributed petitions to request a U.S. Chapter referendum on an amendment to the DECUS U.S. Chapter By-Laws. As most of you know, the Chapter By-Laws govern the way that DECUS is run.

To make a long story short, what we propose is a change in the section of the By-Laws which allows members to propose changes to the By-Laws. In other words, we want to amend the section which lets us amend sections.

The current wording of the section that we suggest be changed is shown below:

> 11.0.1 Initiation of Amendments
>
> The Board of Directors may propose an amendment to the Bylaws by a two-thirds (2/3's) vote of the Board of Directors voting members. Chapter Members may propose an amendment by submitting a petition signed by ten percent (10%) of the chapter membership total as of the first day of the current year. Such petition will be delivered to the President of the Board of Diretors. Within sixty (60) days of receipt by the Board, the signatures will be validated and the ballots distributed to the Chapter Membership so that they might vote on the proposed amendment.

The change to this section that we propose is minor. It is a single sentence change. The change is italicized below:

> Chapter Members may propose an amendment by submitting a petition signed by *Chapter Members numbering at least ten percent (10%) of the vote in the most recent election of the Board of Directors.*

This amendment will give active DECUS members more control over the Society. The numbers necessary to amend will change from around 5,000 (10% of 50,000 members) to 3,500 (10% of 35,000 votes in the most recent election).

Certainly both are huge numbers of signatures. Neither quantity is easy to obtain, but it is easier to find 10 percent of the active members - those who take the time to vote - than 10 percent of all members, including "inactive" ones.

There would be a fighting chance for collection of enough

signatures at one symposium to propose amending the By-Laws.

At last count we had a few hundred signatures obtained at Nashville. We need more. Lots more. More from the people who attend symposia. More from the people who don't attend symposia. More from LUGs. More from individual members.

I have petition forms. Write me or call me and I will be happy to send them out. If you have petition forms from Nashville and you're waiting to get them filled up - don't wait! Get everybody you know to sign them, then send them back. They don't have to be full up to be "acceptable".

Every signature counts. Just drop me a note stating that you support this petition, with your name, signature, and DECUS number (if you know it). That puts us one closer to those 5,000 signatures.

Take control of your future. The Board election turned out well for us. Now help us put even more control of DECUS back in the hands of you, the members at large.

----- Submitting Articles to the Multi-Tasker -----

Please submit machine readable media if you can. RK05, RK07, RL01, RL02, RM03, RX01, RX02, RX50, TK25, TK50 or 9 channel magtape at 800 or 1600 BPI are all acceptable. I can read paper tape too. Anything I can't read can be converted; don't let that stop you from submitting. All RSX volume formats are acceptable except ROLLIN or PRESRV.

The Editor can now Kermit articles into the Multi-Tasker host. The reverse is unfortunately not true; the Multi-Tasker host is normally an isolate. If you want to submit via Kermit, call beforehand with a phone number, login for the host machine and system uptimes.

Submissions which aren't machine readable take longer to get into print. The editor is lazy and types mass quantities only once a month when progress reports are due.

If you preformat a submission in RUNOFF, please set page size 58,80; left margin 10; right margin 75; and, when changing margins, use incremental changes rather than absolute. The editor blesses you for the consideration.

Send all submissions to:

Bruce R. Mitchell
Machine Intelligence and Industrial Magic
PO Box 816
Byron, MN 55920
(507) 775-6268

----- And That's The Way Things Are -----

... this month in Pool Lowbegone, where the Field Circus representatives are strong, the coprocessor RSX implementers are mighty good-looking, and the backplane voltages are above average.

RSX BBS Announcement / Call for Hardware

James Bostwick and Bruce Mitchell
RSX SIG Steering Committee

The RSX SIG has perennially considered establishing an "RSX users only" computer bulletin board. Whenever the topic came up, one problem was always insurmountable: Lack of funding for the necessary hardware. Well, that's all changed.

An RSX bulletin board system is now reality, thanks to the generous donation of a PDP-11/24 from Digital Basics of Burnsville, Minnesota. We are pleased to announce that the basic system is available for use.

The BBS modem phone number is (612) 777-7664. The line supports Bell 103 protocol (110/300 baud) and Bell 212 protocol (1200 baud). Considering the type of traffic that we see on DEC-oriented bulletin board systems, it's hardly surprising to find that an alpha mnemonic for that number is: (612) SPR-PONG. (Now you can play, too!)

To obtain an account on the BBS, call it up and log in using the account name ACCOUNT, password REQUEST. This gets you into a slaved procedure which requests your name, address, telephone number and DECUS membership number. No accounts are given out without this information, so dig out that membership card or look on your newsletter subscription mailing label to find your DECUS number.

On verification of the submitted information, we mail an account notice letter to the address given in the request. This letter contains the account name, virgin password, and a copy of the user procedures. With this information in hand, you are welcome to log in at any time.

We must point out that the software for the system is still under development; the behavior of the system is therefore subject to change without much notice. As we start up, functions available include back issues of the Multi-Tasker, general news, user-to-user MAIL and outgoing KERMIT of all of the above. We hope to have conferencing up before the end of the year.

Call for Hardware!

We need hardware for the bulletin board system. What we have is the 11/24 with 128 Kw of memory and an RL01. We need more memory, disk drives and packs, but anything and everything is welcome.

So - root through those spares cabinets and back rooms. Pull out what you can give us. Q-bus, Unibus, PDP-8, PDP-11, VAX, whatever; you name it, we'll take it. Front panels, side skins, power supplies, anything. If we can't use it on the BBS, we'll flog it for something that we can use. No donation is too small; single cards are just as welcome as complete 11/83s.

Contact your friendly Multi-Tasker editor at the address and number in the editorial section to arrange for us to take that useless 11/751 off your hands!

Private LUNs for Low-Level I/O Routines

Alan E. Frisbie
Flying Disk Systems, Inc.
4759 Round Top Drive
Los Angeles, CA 90065
(213) 256-2575

This article describes how the writer of low-level I/O routines can reserve private LUNs for error messages, special devices, etc., even when a high-level language allocates its own LUNs.

In my business, I write device drivers for some pretty weird devices. Some of those devices have more functions and options than a Chinese menu. This is no problem for me, of course; a good driver can map a QIO function code to almost any device function.

The problem comes when the customer says, "Fine, now write us some programmer-friendly subroutines to make it easy for us to issue QIOs to our bizarre device. And by the way, we're using the Mongolian Software Pascal-Zero compiler for all of our development."

What the customer doesn't say is that his programmers know nothing but Pascal-Zero, have no concept of QIOs, LUNs or TKB options, and have never used RSX before. They also think that status values returned from subroutines never need to be checked.

If a programmer-friendly (even to Pascal-Zero) routine is to issue a QIO to the XX: device, it needs a private LUN. The question is, which one? Also, since the application programmer never bothers to check the error status, we want to print a message on the terminal if the device dies or the programmer screws up. Again, what LUN do we use for the error message?

In an all-Macro environment it is easy to say, "LUN 1 is assigned to device XX: and LUN 5 is assigned to the terminal." The assignments can be made either at Taskbuild time with the ASG option or at runtime with the ALUN$ directive.

It is not that simple with "high-level" languages like Pascal-Zero, or with programmers that forget that your device needs a LUN of its very own.

Newer high-level languages generally mask the details of LUN assignment by assigning them dynamically. The first file opened is assigned LUN 1, the second is assigned LUN 2, etc. If they are opened in a different order, the LUN assignments are different. As soon as you think you are safe by using LUN 10 for your QIOs, some bozo opens 10 files simultaneously. Likewise, you can never depend on LUN 5 being assigned to the terminal, either.

What to do?

Fortunately, the Taskbuilder comes to the rescue by providing up to nine "secret" LUNs beyond the number specified by the user in the TKB "UNITS=" option. Five of these are reserved for DEC's use and another one is semi-reserved. This leaves us with three LUNs that are all ours, plus one that we can share!

How does the Taskbuilder do this and how do we use it?

The Taskbuilder builds the Logical Unit Table ("LUT") for your task and assigns device names to each LUN. It then checks for the existence of certain "reserved" symbols in your object files. If one of these reserved symbols is found, the word at that location is filled with a "private" LUN beyond the numbers already assigned. The LUT is extended by one entry and a default device name is assigned to that LUN.

The reserved symbols and their default device assignments are as follows:

.NLUNS		Filled with the number specified by the TKB "UNITS=" option. The following assignments receive numbers above "n" in this order, assuming that they are defined.
.MOLUN	TI:	"Message output" LUN used by Fortran IV, Fortran-77 and ??? for error message output to TI:. Perfect for error messages.
.MBLUN	SY:	"Mailbox" LUN used by RMS for DECnet applications.

```
.PTLUN  SY:  "Plotter" LUN for plotter/graphics software.  If you are
             not using DEC's graphics software, it's fair game.

.USLU1  SY:  "User" LUN 1.  Reserved for you!

.USLU2  SY:  "User" LUN 2.  Reserved for you!

.NOVLY  OV:  Overlay handler LUN.  Hands off.

.ODTL1  TI:  ODT terminal LUN.  Hands off, unless you want ODT to use
             another terminal for I/O.  (This can be useful.)

.ODTL2  CL:  ODT listing LUN.  Used with ODT's "L" command for
             dumping to the optional listing device.

.TRLUN  CL:  Used by the TRACE debugging aid for its listing.
```

P/OS systems have one additional reserved symbol (.SUML1) for the standard utility module. Its usage and position in the table is left as an exercise for the reader. This list can also be found in Appendix E of the Taskbuilder manual.

The example shown below uses both .MOLUN and .USLU1 for error messages and custom device I/O, respectively. Only the details essential for illustrating the principles are shown. Code specific to a particular high-level language calling sequence is not shown.

```
        .TITLE  LDEMO  LUN Assignment Demo

; Demonstration of "private" LUN use in low-level I/O routines.
; This demo performs some QIO operation on device XX:  We use
; .USLU1 (User reserved LUN  1) for I/O to XX:.

LDEMO:: TST   UL1FLG                    ; Is LUN already assigned?
        BGT   20$                       ; If >0, Yes - skip to 20$
        BEQ   10$                       ; If =0, No - assign it

; ALUN$ failure.  Print error message on LUN specified by .MOLUN.
; TKB has already assigned it to TI:, so we don't have to do it.

        MOV   .MOLUN, ERRQIO+Q.IOLU     ; Set TI: LUN in QIO DPB
        DIR$  #ERRQIO                   ; Print the error message
        <Return status as required by the high-level language>

        RETURN          ; Return to high-level code


; Attempt to assign the first user LUN to the XX: device

10$:    MOV   .USLU1, ALUNXX+A.LULU     ; Put private LUN in DPB
        MOV   .USLU1, XXQIO+Q.IOLU      ; And also into QIO DPB
```



```
        DIR$  #ALUNXX                   ; Assign LUN to device XX:
        MOV   $DSW, UL1FLG              ; Save the ALUN$ status
        BR    LDEMO                     ; And go test the status

; Here we do whatever must be done before issuing the QIO...

20$:    <code continues to set up for the I/O, then suddenly...>
        DIR$  #XXQIO                    ; Do QIO to device XX:
        <Return status as required by the high-level language>

        RETURN                          ; Return to user


; >>> THE FOLLOWING TWO WORDS ARE FILLED BY THE TASKBUILDER:  <<<

.MOLUN:: .WORD  0       ; Filled by TKB with LUN for TI:
.USLU1:: .WORD  0       ; Filled by TKB with our private LUN
UL1FLG:  .WORD  0       ; User LUN 1 assigned flag

ALUNXX:  ALUN$  0, XX, 0
XXQIO:   QIOW$  IO.WLB, 0, 24,, XXIOSB,, <FOOBUF, FOOLEN>
ERRQIO:  QIOW$  IO.WVB, 0, 24,,,, <ERRMSG, ERRLEN, 40>

ERRMSG:  .ASCII /Error assigning LUN to device XX:/<CR><LF>
         .ASCII /Jim's fly is probably open./
ERRLEN = . - ERRMSG

        .END
```

The above code works with either Macro or high-level language callers. When using it with Fortran-77 and some other languages, however, the symbol .MOLUN may be multiply defined. This is because F77 uses .MOLUN for its own error messages to TI:.

In this case you might simply choose not to define .MOLUN in your code, and use the copy that F77 defines. This causes it to be UNdefined if you ever use that routine without Fortran.

The cleanest solution is to create a one-word object module that defines .MOLUN. This module is then inserted into SYSLIB, from whence it will be extracted if no other module defines .MOLUN.

```
        .TITLE  MOLUNX  Definition of .MOLUN

; This definition of .MOLUN is written to be inserted in SYSLIB.
; Thus, .MOLUN is automatically defined regardless of whether the
; using program already defines it (as with FOR or F77).
;
; The Taskbuilder fills it in with a LUN and assigns it to TI:

.MOLUN:: .WORD   0
         .END
```

Assemble it, and insert the resulting object module into SYSLIB:

```
LBR LB:[1,1]SYSLIB.OLB=MOLUNX.OBJ/IN
```

By use of these techniques the innocent high-level applications programmer is insulated from the vulgarity of LUN allocation and assignment, Taskbuilder options and the like. Even better, the low-level I/O routine writer (i.e., you) no longer must spend countless hours explaining these subjects to each new applications programmer.

Now, if I could only do something similar with event flags...

The Hitchhiker's Guide to RSX, Part I

A-to-Z Base Product Marketing
Digital Equipment Corporation
Continental Boulevard, MKO1-2/E25
Merrimack, NH 03054

Through the kind intervention of a Digital employee, the Multi-Tasker is pleased to present "The Hitchhiker's Guide to RSX". This is probably the best overall coverage of the current RSX environment available. It is also a valuable reference for all application programmers, not just business application developers.

This document is being serialized due to its length. This is the first of three parts, covering chapters 1 through 4.

An Introduction to RSX for Business-Application Developers

Revision 0.0

Disclaimer

September, 1986

Table of Contents

D O N ' T P A N I C

Chapter 1

Introduction

This guide introduces the RSX operating system to the Small Business Application developer. It contains information about aspects of RSX which are of particular interest and importance to application developers considering or in the process of migrating commercial applications to RSX, or creating new ones. Much information is also of use to developers considering a move to VAX/VMS, since the two systems share many characteristics.

The guide describes general principles for developing a new application or adapting an existing one to RSX. It also covers certain areas of special importance to small business application designers.

RSX is a mature operating system and there are many case histories of successful use in a small business environment. Unfortunately, there are also instances in which developers encountered difficulty in making the transition from simpler environments. In many cases the effort required to correct the problems proved to be trivial. All that was lacking was information.

There is nothing about RSX making it unsuitable as a base for small business software. Developers who invest the time and energy in developing an understanding of the strengths and weaknesses of RSX maximize their chances at a smooth, successful and inexpensive conversion.

1.1 The Purpose of this Guide

Operating systems are all different. The most frequent problem in migrating an application *from* one system to another is moving a set of programs designed around the strengths and weaknesses of the original system without considering the target's characteristics.

In many cases, such oversights are the result of time pressure. In others, they are due to lack of information about which areas need the most attention and how to attain the most improvement for the least amount of work. This guide helps point the way.

This is not a substitute for the various developer's materials available for RSX, but is rather a summary intended to assist third

party software developers in assessing the desirability of moving to RSX, and in minimizing the number of pitfalls into which such a developer might fall. If you are new to RSX it is of assistance during those first days in which adjusting to any new system is difficult.

The principles and techniques described are largely independent of the implementation language to be used. An efficient RMS file design improves the performance of an application program whether it is written in Fortran, Basic, or Pascal. Many persons reading this guide do so in preparation for a migration of an existing application from CTS-300 or CTS-500 to RSX and VMS, and some special notice is taken of this path.

1.2 Intended Audience

This guide is addressed to people with a variety of backgrounds. You might be a third party software developer considering a move to RSX. You might have already decided to give RSX a try, and are wondering how to plan migration of your software. You could be part way through a conversion and finding difficulties in locating the information you need. You might be facing a particular problem and looking for a quick solution.

It is assumed that you know something of commercial application development and have a favorite programming language. If you are coming from a CTS-300 environment you may find that many of the facilities and concepts available with RSX are complex. If you are coming from VMS on the other hand, RSX seems more familiar.

1.3 On the Synergy of Operating Systems and Applications

Operating systems are alike in that they serve as a mechanism by which system resources are allocated to various objects, terminals, or programs.

They are different, however, in the overall design which determines the type of expected user application. Some are designed to react as quickly as possible to an event, some are designed for security against power or component failures, and some are optimized for a particular language.

In all cases there is synergy between an application and the operating system. Since each system has a distinct set of strengths and weaknesses, it behooves the application developer to make changes that take advantage of strengths and avoid weaknesses. Problems do not usually reside exclusively in the operating system or the application, but most often result from attempts to treat one operating system as though it had the properties of another.

The importance of an application designer's understanding the characteristics of the chosen operating system cannot be overemphasized. The return on such an investment can be staggeringly large. In one case a simple change of subroutine linkages taking 20 minutes resulted in a twofold performance improvement. In another case, optimization of a single subroutine resulted in tenfold improvement of a critical module.

1.4 Elements of Application Tuning

There are two distinct aspects of design and tuning of applications in a commercial environment. The first is the number of resources required by a particular program, such as the number of files, CPU cycles or terminal operations. The second is speed and efficiency with which the operating system can deliver those resources.

Identifying the contributions that an application makes to system load is difficult for many developers. A module may work well with small numbers of users, but under full load the overall system may deteriorate quickly.

All too often this condition does not occur until the product is actually installed at a customer site where there is less time and fewer opportunities to investigate the problem. The purpose of this document is to help you anticipate such problems and to apply corrective action as early and inexpensively as possible.

When a performance problem occurs, the solution must be to reduce the demand for resources or increase the speed with which the requests are satisfied. Since delivery of a resource can never be instantaneous, the first effort should be to eliminate the requirement. The sections following describe ways you may reduce or eliminate demands on an operating system by your application.

Chapter 2

Overview of RSX

The 1970 announcement of the PDP-11 resulted in an industry demand for a high quality, high performance and comprehensive multi-tasking operating system. The RSX family began with an initial adaptation from the PDP-15 and grew with additions of RSX-11A, -11B, -11D, -11S, -11M, IAS, P/OS, -11M-Plus and, preserving the family ties with the 32 bit big brother, VAX-11 RSX. Many VMS concepts can be traced to previous work on RSX.

2.1 Historical Overview

A number of operating systems were developed for the PDP-11 including RT-11 as a 16-bit follow-on to OS/8, DOS-11 for larger machines and more sophisticated customers, and RSTS for the education market. DEC mythology has it that at one time or another there were 27 different PDP-11 operating system projects funded and underway.

In the early years, the scientific/industrial market was of primary importance to Digital and RSX was the flagship product. Most early minicomputers were used for process control and data collection, and it was essential that the system software offer the greatest possible degree of flexibility. Hardware was expensive and if a month or two of bending and squeezing the software meant that memory requirements could be reduced by 4KW it was well worth the effort.

Furthermore, there was no end to the variety of configurations which could be assembled. OEMs and end users were willing and able to create their own hardware which could range from a simple parallel digital interface to DMA communication ports. And, since their world was primarily real time, they required that response times to external events be both extremely fast and highly predictable. RSX was designed to satisfy the most exacting real-time requirements and could be tailored to almost any configuration of off the shelf and custom hardware.

Every feature has an associated cost. The modularity of RSX meant that the process of creating a working system was complex, even if the hardware configuration was comparatively simple. Users who didn't have critical realtime requirements were nevertheless obliged to deal with a human interface designed for operating four or five processes at once.

This was particularly true for the less technically oriented commercial users who were beginning to experiment with minicomputers as an alternative to batch processing. Multi-user protection was not supported in the early versions. Early MCR bore an unfortunate resemblance to certain UNIX shells. Memory was divided up into partitions which always seemed to be mismatched with the job at hand, the system simply stopped if a particular resource was exhausted, there was no data management other than block I/O, and the only implementation languages were Macro and Fortran. Terminals seemed to be supported only as an afterthought.

But even scientists tire of creating record management routines, and over the years RSX was improved in ways attractive to non-technical users. As underlying hardware became less expensive it was possible to ease the requirements for partitions in memory and provide more facilities in the Executive.

RMS finally appeared with multi-key ISAM. Partitions and POOL management became dynamic; round-robin scheduling and swapping support were added. DCL was provided (albeit only as an "alternative" to MCR). DECnet was put in place and, wonder of wonders, the terminal driver was made full duplex and could support type-ahead, trap CTRL/Cs and process escape sequences.

Still, the SYSgen problem remained, and it was finally decided to produce a pre-SYSgenned, bounded version of RSX - Micro/RSX - with commercial features which could be handled by a computer naive customer.

The Micro PDP-11 was the first PDP-11 intended for installation by a first time user. Since the hardware was user-installable it was felt that the software should be as well. Moreover, it was considered that the Micro customer represented a new class of RSX user - a person not concerned with the traditional scientific/realtime features of RSX, but simply desiring a system which could quickly and easily be put to work.

Rather than require these users to describe the target hardware configuration in detail, the Micro/RSX software accommodates itself to whatever is available.

2.2 RSX Version 3.0 Family

Micro/RSX was the *first* member of what is now the RSX Version 3.0 family and has been joined by RSX-11M-Plus in both pregenned and full kit form. Micro/RSX and pregenned M-Plus eliminate the SYSgen process and adapt themselves to any legal hardware configuration during system startup. M-Plus still requires a SYSgen to move from the baseline (distribution) Executive to full timesharing, but even this process has been considerably simplified.

The features available with RSX V3.0 include:

- Full support for all of today's PDP-11 high performance CPUs, memory to 3.8 MB and I/D hardware where available.

- Support of all modern disks and peripherals.
- Disk structure, named directories and logical names compatible with VMS.
- RMS including multi-key ISAM compatible with VMS.
- Terminal emulation and remote file access via DECnet.
- Sharable, clustered and supervisor-mode libraries.
- Sharable, checkpointable common regions.
- Multiple stream print and batch queues.
- Implicit device spooling.
- Synchronous and asynchronous I/O on up to 256 channels per task (not supported by all languages).
- DCL with extended help and command line continuation.
- Disk data caching.
- Password encryption.
- Multi-volume backup to disks and tape.

Application implementation languages and tools available under RSX include:

- BASIC-Plus-2
- COBOL-11
- COBOL-81
- Datatrieve-11
- DBMS-11
- DECnet
- DIBOL-83
- FMS-11
- FORTRAN-IV
- FORTRAN-77

- MACRO-11
- Pascal
- RPG II
- Sort/Merge

With the availability of Version 3.0, RSX has become a high performance, high functionality operating system base for commercial applications.

Chapter 3

Getting Started

This section is for users who have never installed or used RSX. It is intended to get you "on the air" as quickly as possible beginning with that uncertain moment your system has been installed and checked-out, and you're about to open the first box containing the software. It also outlines the general process by which the software is installed and customized so that you may estimate the amount of time required.

3.1 The Three Faces of RSX

Beginning with the Version 3.0 release, RSX comes in three highly compatible "flavors". (There are also two other members of the RSX family, RSX-11M and -11S but these have become special purpose products and are only of interest to the traditional RSX customer.)

3.1.1 Micro/RSX

The low-end member of the family is Micro/RSX, the first version of RSX with a full set of commercial features. Micro/RSX was specifically tailored for the MicroPDP-11 and is the first version which can be taken out of the box, installed and used by a non-technical person. Micro/RSX served as the basis for A-to-Z V1.0 and is usually the system of choice for Micro-11 configurations.

3.1.2 Pregenned M-Plus

The intermediate family member is the "Pre-SYSgenned" or RL02 version of M-Plus. This is variant of M-Plus which is configured so that it runs on a wide variety of hardware configurations including the MicroPDP-11. It is self tailoring and, with the exception of certain features specifically selected through SYSgen, is identical to the full M-Plus system. This is the most appropriate system for commercial users on configurations other than the Micro-11.

This version is only distributed on RL02 disk packs, and there are restrictions about which disk types may be used on the target system; otherwise, it is usable as soon as it is copied to the system disk. Unless you require special hardware support or Executive features which are not part of this version, you should use this version of M-Plus.

Due to size constraints of the RL02, some parts of the full M-Plus system are missing. If any of these components are required on your system they may be copied from an M-Plus kit.

3.1.3 RSX-11M-Plus

M-Plus is the high end, full functionality system and must be tailored (SYSgenned) before it can be used. It is distributed in source form along with a baseline system (minimal Executive) with which a SYSgen may be performed.

SYSgen is a question and answer process in which the user describes the exact combination of Executive features and device support desired. The Executive and associated utilities are then assembled and linked together to form a bootable system image. This image is further tailored by setting up partitions, installing tasks, etc. until a runnable system is finished and ready to be installed. More on SYSgen later.

3.2 Installation and Startup

Once the three versions of RSX are installed there are few differences between them. The three versions do, however, differ in the way they are shipped to the customer and in how they are installed. These differences are due to which assumptions can be made about the underlying hardware.

3.2.1 Installing Micro/RSX

Installing Micro/RSX on a MicroPDP-11 requires only that the software be copied onto the system disk. The distribution media consist of either a set of RX50 diskettes or a single TK50 cartridge tape.

In either case, the first medium contains a bootable, standalone version of BACKUP. All that is necessary to install the software is to insert the first media unit into the appropriate drive and turn the machine on. Once the PDP-11 completes its self-check it notices that a medium is in place and bootstraps BACKUP into memory.

BACKUP announces itself and then determines the processor type. The distribution kit actually contains two complete Executives. One of the system images is intended for use on processors such as the 11/73 and 11/83 that support separate instruction and data space mapping. The other is for processors such as the 11/23 which do not support I/D space mapping.

The drive to be used as the system disk is initialized (and cleared of all existing data!); then the system software is copied into place. When the copy operation is complete, you are directed to remove the media and restart the system.

The restart requests the time and date and then runs automatically through the remainder of the startup procedure. When startup completes, the console terminal logs itself out to prevent an unauthorized person from gaining access to your system. The system is now fully operational and ready for use. The entire process requires less than an hour.

Micro/RSX was designed around systems with limited disk capacity and consequently there are some optional software packages which are distributed separately. These options are primarily those required for software development. If you have these options they may be installed right away or at some future time as needed. The procedures for installing optional software on Micro/RSX are described below.

3.2. Installing Pregenerated M-Plus

The pregenerated M-Plus kit is the quickest and easiest way to get RSX-11M-Plus running on your system and, for most commercial customers, it is the most appropriate combination of SYSgen parameters. Installation involves copying the software onto the target system disk, selecting the appropriate system image file and installing the system disk handler. This process is not as automated as that for Micro/RSX and the dialog appears somewhat mysterious, but the procedure is only slightly more complicated.

The entire installation is described, step by step, in a special chapter in the "RSX-11M-Plus System Generation and Installation Guide". The process does not require any great degree of computer experience as long as you have this guide handy when you begin the installation.

The kit consists of a single RL02 which contains two ready-to-run system images with all necessary system files and utilities. The RL02 pack can be bootstrapped and (except for the lack of free space on the distribution volume) used straight away.

One of the system images is intended for use on PDP-11 processors such as the 11/73 and 11/83 which suppor separate instruction and data space mapping. The other is for use on processors such as the 11/23 which do not support I/D space mapping. The non-I/D Executive can also run on I/D systems and it is, in fact, the system that runs when you first boot the distribution disk.

Installation begins with bootstrapping the distribution disk on the target system. When the startup procedure asks for the time and date, the procedure aborts and a special standalone Backup/Restore is bootstrapped from the RL02. This special system allows initializing the target disk, locating and marking any bad blocks, and copying all contents of the distribution kit onto the target disk.

At this point all files are in place, but the target disk is not

bootable. What remains is to choose the appropriate Executive (I/D or non-I/D), configure it for the target system, and then make the system image bootable.

These procedures are largely automated, though they do require you to type in some very strange commands that produce even stranger output on the console. It is from this type of dialog that SYSgen acquired its evil reputation. As of Version 3.0, however, it is no longer necessary to understand the process in any detail. You are not required to make any decisions other than selecting an Executive based on your processor type. Simply follow the instructions carefully.

The last step begins with rebooting the kit disk. Interrupt the startup procedure and, following the instructions provided in the SYSgen and Installation guide, load the driver for the system disk, bring it on line, and set default to the desired system area. Selecting the proper system is the only decision you have to make and, unless you have good reason to do otherwise, it is determined by your hardware as described above.

You then create a copy of the system image and invoke VMR which performs an automatic procedure to tailor the selected Executive to your system disk. You then make the image bootable by booting the image just tailord and "saving" it in such a way that the bootstrap code can locate the image file.

The installation is now complete. The entire installation process requires less than two hours.

3.2.3 Installing M-Plus

Installing M-Plus is a two stage process. First, you copy the distribution kit to the target system disk. Second, you do SYSgen to build an Executive tailored to your specific hardware configuration and software needs. This process can be performed on an existing RSX system or on a new system.

Compared to previous versions, SYSgen for RSX V3.0 is painless and largely automatic. If your hardware configuration is standard and you have no special requirements of the system, you can be finished in a few hours. If you are working on the target system you can direct SYSgen to "Autoconfigure" your hardware; it uses the results as the default answers to the most difficult questions.

Furthermore, you can direct SYSgen to create a "full functionality" Executive thereby eliminating a large number of intricate questions about Executive options. If you accept all the default answers, the dialog section of SYSgen is largely a matter of confirming that little used options are, in fact, not desired.

If you are an inexperienced M-Plus user, you are strongly

advised to Autoconfigure on the target and accept all the defaults. In this way you get your system up and running as quickly as possible. Once you have more experience you may repeat SYSgen to create an Executive more closely tailored to your needs. Most commercial users find that the full functionality Executive suits their needs perfectly.

On the other hand, if you have an unusual hardware configuration or have special needs, you may wish to contract for enough support to get you through the installation. RSX can be tailored to fit almost any hardware configuration manufactured, but the process of discovering all the ins and outs of CSR and vector assignments is a complex one. This is no time to play the hero. If you have doubts about your capabilities, call for help.

Details of the installation and SYSgen process are beyond the scope of this section, and are described in the "RSX System Installation and Generation Guide". There are many different paths depending on what existing systems are available but the simplest is installation on new hardware. The sequence of events for installing on a new system is roughly as follows:

Boot the distribution media. This starts up standalone Backup (actually an RSX-11/S system image) which you use to format the target disk and scan it for bad blocks.

Copy the first of two backup save sets from the distribution media to the target disk. The result is a hardware bootable disk containing a baseline system which runs on almost any hardware and contains just enough functionality to complete a SYSgen.

Boot the target disk. The baseline system automatically copies the second backup save set from the distribution kit to the target disk.

Apply any necessary Updates. Unless you have obtained your software shortly after a major release of RSX there are likely one or more Update kits which must be applied to the target disk before you proceed with the SYSgen. This process is largely automatic. Invoke the UPDATE command procedure to apply the corrections.

Invoke SYSgen. Again, there are many paths and options but the best plan for a beginner is to go straight through from the beginning to the end, accepting defaults wherever possible. While the simplest path does not require you to answer many questions, a copy of the dialog is useful if anything goes wrong. If you do not have a hardcopy terminal as the system console it is wise to connect a printer so that all output is saved.

SYSgen asks a number of questions about how the Executive is to be built. Unless you know better you should always take the default. The exception is the question labeled SU100 which asks if you wish to run Autoconfigure. If you are running on the target hardware you

should answer YES. Doing so results in an operation in which the system interrogates all the devices it can find and configures your peripherals accordingly.

The only other questions you are asked concern hardware which was expected but is missing (why SYSgen asks if you have a card reader is a mystery) or about options that are costly or infrequently used (modem support on terminal lines).

After about 15 minutes of dialogue SYSgen assembles and links the Executive and all the privileged tasks. Watch the dialog as it unfolds and compare it to the example in appendix D of the SYSgen guide. The remainder of SYSgen takes about three hours. The process is automatic but you are asked an occasional question, so it's a good idea to check in from time to time to see how things are progressing.

When SYSgen completes you have a fresh copy of the Executive which is not hardware bootable. Follow the directions in the Guide to software boot the new Executive and enter the time of day. This action is only a sanity check to see that everything worked properly.

SAVe the Executive. SAV causes the new Executive to be rebooted and begin the standard system startup procedure. If everything still looks all right, abort the startup as directed and SAVe the system again, this time with the /WB (Write Bootstrap) switch. This final step changes the bootstrap blocks of the target disk to point to the newly generated Executive. The system restarts automatically and you are on the air.

3.3 Logging In and Other First Day Tasks

Once your system is installed and operational the first thing you should do is log in as the system manager and make any additional adjustments you feel are necessary. Typical chores involve post-installation clean-up, selecting appropriate startup options, creating user accounts, configuring terminals and printers, and installing layered products.

3.3.1 Logging Into the Prebuilt Accounts

RSX is distributed with a System account and a User account. You can log into either one from an unused terminal (such as the console terminal once startup has completed. Press the Return key until you get the system prompt, which may be either ">" or "$". Type "HELLO" and press Return. You are prompted for a username and password. When these are entered and verified against the system accounting file, login completes.

The prebuilt account username/passwords are SYSTEM/SYSTEM (or MICRO/RSX on Micro/RSX) and USER/USER. Once the login task validates your account it extracts other information about your account from

the authorization file and then executes two command files on your behalf.

The first of these, LB:[1,2]SYLOGIN.CMD, (if it exists) is executed for all users and, in its most common form, determines your terminal type with SET TERMINAL/INQUIRE and then chains to the LOGIN.CMD file in your default directory.

If LOGIN.CMD exists, it is also executed. In the case of the USER account it prints out a welcome message and some introductory information.

These two login files can be used to control the accounts in your system. In the case of A-to-Z they are used to direct the user directly from login into the appropriate A-to-Z menu.

RSX supports two Command Line Interpreters (CLIs). MCR is the traditional CLI still favored by RSX old timers, while DCL is probably more familiar to newcomers. The account entry in the authorization file determines which of the two is established for a particular user during login. You may switch from MCR (usually identified by the "›" prompt) by typing

```
› SET /DCL=TI:
```

You may switch from DCL (usually identified by the "$" prompt) by typing

```
$ SET TERMINAL/CLI=MCR
```

Most users learn one CLI and stick with it. You should modify the authorization file entry for each account according to the user preference.

3.3.2 Post Installation Cleanup

The Pregenerated (RL02) version of M-Plus contains both an I/D and the non-I/D Executive. Once installation is complete you may delete the unused files to recover disk space. A command procedure is provided for this purpose. The System Generation and Installation guide tells how to proceed.

The procedure is completely automatic. It determines which Executive is in use and deletes the unused files. At its conclusion the procedure reports the number of disk blocks recovered. You may postpone this task if you have sufficient disk space for the early days of usage.

3.3.3 Customizing Startup

The Micro and RL02 systems are shipped to you with certain system parameters, such as the number of batch and print queues, set to values appropriate for an "average" configuration. These parameters are stored in a file (LB:[1,2]SYSPARAM.DAT) which is readable by non-technical users and which is interpreted by the system startup procedure.

Statements in this file determine such things as the amount of checkpoint space and secondary POOL allocated, what combination of batch and print queues are to be started, what to do about error logging, and so on. Provision is made for controlling every commonly used element of the system. If you can use an editor, you should have little difficulty modifying this file to your own taste. It is unlikely that you must do anything further.

If you have chosen M-Plus, you are supplied with a skeleton system startup control file, LB:[1,2]STARTUP.CMD. This sample file is sufficient to get you on the air but it is expected that you want to modify it and know how to do so.

Regardless of the system, if you have added A-to-Z and are running primarily A-to-Z applications you find that A-to-Z manages all of the usual system startup tasks. If you have requirements beyond those managed by A-to-Z you probably know how to take care of them.

3.3.4 Configuring Terminals and Printers

When RSX configures your system it can detect the presence of the various device controllers on the system, but in the case of serial and parallel ports it makes no attempt to determine the type of device, if any, which is attached. You have several options:

If the devices are all terminals, you can let SYSLOGIN.CMD do the work with SET TERMINAL/INQUIRE which polls the terminal to determine its type whenever a user logs in.

If the devices are a mix of printers and terminals and you are running Micro or pregenerated RSX, you can edit the parameter file LB:[1,2]SYSPARAM.DAT and designate each of the lines or ports with the appropriate device type and line speed. The next time your system is re-started the devices are identified and brought on line.

If you don't want to be bothered, or if you are installing a system for a customer who may not want to be bothered, you should consider adding A-to-Z to the system. A-to-Z includes a process which runs at system startup which polls all available ports and determines the device type and speed of each line. This information is stored in a table where it is available for inspection or modification by the customer. The advantage of this is that if a terminal is moved or the speed of a device is inadvertently changed, the customer can deal with the problem directly without any special training.

3.3.5 Creating User Accounts

The procedures for adding, changing, and removing accounts are identical for all three systems. A utility (ACNT) may be invoked by any privileged user to make changes to the system accounting file (LB:[0,0]RSX11.SYS). Take care that this file is not inadvertently deleted. If the file is deleted you will have difficulty regaining control of your system.

Before you enter a new account you should know certain things required by ACNT. You are asked to supply a username, typically the user's last name, which must be unique on the system, a password which is a "hard to guess" word of up to 39 characters, a User Identification Code (UIC) which classifies the user according to group and access privileges, a default CLI (DCL is easiest for most new users to learn), and a disk and directory for the user's files. The items are entered one at a time and the default directory is created if it does not already exist.

The new user may log in immediately after you have entered the information for the account. There are several other items which you may enter including a session identifier, first name, and so on. You may also modify an existing account if you wish to change any parameter. Note that beginning with Version 3.0 of RSX you may opt for password encryption on an account by account basis. Such passwords may not be examined once they have been entered, so they must be specified anew if the original has been lost.

If you have A-to-Z you may wish to manage your accounts with the facilities provided for the A-to-Z Manager. Unless you have special needs you should find that the facilities provided by A-to-Z are adequate and are much easier and safer to use.

3.4 Installing Layered Products

Installation on any version of RSX is largely a matter of copying files into the appropriate directories, possibly taskbuilding certain modules, and adding the necessary initialization commands to the system startup procedures. On M-Plus this is usually accomplished by a command file, supplied with the layered product, which conducts a dialogue with the system manager to determine which options are to be selected and then performs all the necessary tasks. Some products require that the system manager manually edit the system startup file and provide the appropriate commands.

Micro/RSX has a more automatic installation architecture which makes it possible for a person with little or no computer expertise to manage the installation. A-to-Z has combined elements of both of these architectures. If you are installing A-to-Z layered products you may do so via an option on the A-to-Z Manager's Menu. The procedure is the same regardless of which version of RSX lies underneath.

3.4.1 Installation on Micro/RSX

The Micro/RSX user does nothing more than choose the product to be installed (in case there is more than one product module in the kit) and handle the media. The architecture and the layered product developer do the rest including making decisions about the underlying hardware, product specific installation and startup procedures, and integration with other components on the system.

To install a product the user logs into the system manager's account (or the A-Z manager's account) and invokes the installation procedure OPTION.CMD (or selects the application installation option from the A-to-Z manager's Auxiliary Functions Menu).

System software then requests that the installer identify the appropriate device drive according to the media type, insert the media, select the application to be installed, and then insert more media as required. While the particular product may require some customization there is nothing further required to get it running on the underlying system.

3.4.2 Installation on M-Plus

Layered product installation on M-Plus (both the Pregenerated and full SYSgen flavors) assumes a more sophisticated system manager with particular requirements. In this case the layered product developer supplies a kit containing all the components of the product and, with the nicer products, a command procedure with which the manager picks and chooses from the different combinations.

The developer is still required to be sensitive to aspects of the product which may change from system to system such as re-linking privileged tasks or relocation of files and modules across disks. The developer is also responsible for providing information regarding the installation, usually in the form of an installation guide. The installation guide and the release notes are the basis from which the system manager decides between various options or, in some cases, redesigns the installation procedure to satisfy specific system conditions.

There is no set pattern for installation on M-Plus. Generally, the Installation Guide provided with the product directs the manager to log in and create a temporary directory. The guide also describes conditions that must be set up before installation can proceed. Certain tasks may have to be installed or optional software may have to be moved into place.

The manager then mounts the media, usually a magnetic tape or a disk pack, and copies the installation control file (traditionally INSTALL.CMD) into the temporary directory. The command file is invoked and, through a dialogue with the manager, determines what actions to take. Once the dialog phase is completed, the command

file performs the remainder of the installation: copying and converting files, linking tasks, and setting up the database. The installation guide also usually provides the manager with information about what modifications must be made to the system startup procedure in order for the product to operate properly.

Products for M-Plus differ considerably in the degree to which the installation is automated. Some come with a list of instructions for the system manager including general directions on what to add to the system startup command file. Others are more automatic. The trend is towards automation as available hardware resources become more plentiful and squeezing the last free byte out of a file becomes less important.

An example of such automation is the A-to-Z Base System which not only controls its own installation, but carries with it elements of the Micro/RSX layered product architecture. This makes it much easier for a computer naive system manager to install and manage an application mix. It also makes it easier on the product developers since the kitting procedure for A-to-Z applications on Micro/RSX and M-Plus is the same. Only the media are different.

Chapter 4

Terminals

Terminal I/O is a prominent aspect of modern commercial applications on timesharing systems. It is, in fact, one of the primary reasons that the old batch processing systems have been replaced. It is also, unfortunately, a major source of system overhead.

Problems related to this aspect of application design are particularly insidious because the cost of moving characters to and from a terminal is so easy to overlook. Most programmers understand that opening and closing files is costly. Far fewer consider what is happening when a screen is being painted.

4.1 Cost of I/O

The cost of a particular I/O operation can be divided into two general categories. The first is the number of instructions executed to move data from the program to the device controller (overhead). The second is time the device requires to move the data between the controller and the outside world (latency).

With disks, much of the time required to fulfill an I/O request is latency due to the mechanical movement of the head and spindle. Once the disk head is moved to the proper cylinder, large quantities of data can be transferred very quickly, usually without CPU intervention. More on disks and files in the next section.

4.2 Specific Cost of Terminal I/O

Terminals are another problem altogether. There is a short latency period while a character is shifted into or out of a UART register but the real problem is that there are ever so many more I/O operations involved in moving a block of characters to or from the terminal. A disk can transfer thousands of characters in a single operation whereas most terminals make the transfers one byte at a time.

4.2.1 Interrupt Service

Terminals can cause almost 2000 CPU interrupts per second, and each interrupt requires the Executive to suspend processing to service the device. Even worse, terminals are unpredictable on the input side. Operating system software can keep track of the head position of a disk drive and can make a guess at how long a given operation takes. This permits operating systems to optimize the

queue of waiting I/O requests. But there is no way that the system can predict how long it takes an operator to press a character key - the computer simply has to wait for the interrupt.

4.2.2 Scheduler Overhead

RSX considers completion of any I/O request to be a "significant event". Such events signal to the system scheduler that there may have been a change in various lists of tasks waiting to execute, e.g. I/O completion may have unblocked a task of higher priority than the current task, thereby requiring a context switch. With even a small number of commercial applications painting screens, the scheduler may be executing hundreds of times per second.

Excessive context switching (thrashing) is a factor in almost every system performance problem encountered so far. The worst possible circumstance is when interactive tasks competing for memory are being checkpointed to disk. In this case each character moving to or from a program causes several far more costly disk reads and writes to take place.

So the cost of moving a single character is the sum of the code to actually move the character to or from the device plus any program overlays which may have occurred plus the interrupt service routines plus the context switching. When many hundreds of characters are being moved back and forth every second the system overhead is considerable.

4.3 Minimizing Costs

There is little you can do to reduce the number of characters which must be moved to and from the terminal. You can, however, do a great deal to reduce the operating system overhead associated with terminal intensive applications by reducing the number of I/O requests necessary to move the requisite characters.

4.3.1 Blocking Output Requests

Whether or not DMA hardware is available, it is very important to gather small groups of characters together before sending them to the terminal. Throughput improvements of 10 to 1 are reported when scattered output requests are converted to calls to a central routine which assembles character strings into a single buffer.

The amount of processing to move the characters to the terminal is the same in either case, but if the number of I/O requests is reduced there is a corresponding reduction in scheduler overhead.

It is not always easy to determine just how an implementation language such as BASIC or DIBOL performs terminal I/O, but making the

effort to find out is well repaid. For example, the following three DIBOL code fragments produce output that looks exactly the same to the operator:

```
WORST,
      DISPLAY (CHAN,'A')
      DISPLAY (CHAN,'B')
      DISPLAY (CHAN,'C')
```

This results in three I/O requests and some unnecessary interpreter overhead.

```
BETTER,
       DISPLAY (CHAN,'A','B','C')
```

This eliminates some of the interpreter overhead but there are still three I/O operations (and consequently three calls to the scheduler, etc.) involved.

```
BEST,
     DISPLAY (CHAN,'ABC')
```

Best of all. It is admittedly unlikely that the output data is so easy to assemble. Most often it is necessary to feed the data fragments to a subroutine along with a flag which serves as a "Do It Now" or "More Coming" signal. But it's worth it.

4.3.2 Blocking Terminal Input

It wasn't long ago that many operating systems were designed to only accept data from an operator in line mode - a series of characters terminated with the Return key. No sane Fortran programmer wants to see characters one at time or, Heaven forbid, a five character escape sequence.

Operating system people were glad to avoid the headaches of having to worry about mapping all the ANSI escape sequences into terminator codes. They preferred to tell any developer who wanted to know whether an input string had been terminated with Up-Arrow or Down-Arrow to figure it out themselves.

But terminals kept getting smarter and even people working in laboratories type with their elbows now and again, so eventually operating systems began to support commercial features, beginning with single character input and some going so far as providing elementary block mode facilities.

There is, however, a certain amount of holdover code in language processors put in place in the good old days to give operating systems the appearance of block mode input and, in some cases, this code can get in the way.

Thus, it is once again important for the designer to understand how a high level language translates input statements into RSX directives. Some languages, DIBOL for example, convert what appears to be a field level request ("READS") into a series of requests for single characters.

The latest versions of such languages may, however, provide a means of control over such actions. This is important because grouping input characters or data has the same benefit that blocking output has. It often makes the job of the developer more difficult, and sometimes impossible, if the functionality of an application is going to be preserved.

If you cannot avoid processing the input stream a byte at a time, so be it. But if it is possible to change the character-at-a-time to post entry validation you enjoy considerable improvement in overall system efficiency.

4.3.3 Use DMA Terminal Hardware

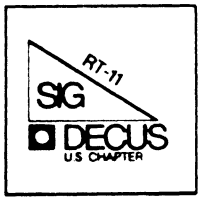
DMA terminal interface hardware reduces the number of interrupts the Executive must service to move a block of characters to a device. The terminal driver is able to treat the output side of the device much as it does a disk. The output operation is initiated and, except for the bus cycles stolen by the interface to fetch the next character, the CPU is free to perform other tasks.

However, there is no great improvement with such an interface if characters are fed to it individually or in small groups. The greatest gain is seen when output data is gathered into a single block and sent to the device with a single I/O request. It is possible to output an entire screenful of information including escape sequences for formatting and setting character attributes.

THE mini• tasker

DECUS

RT-11 SIG NEWSLETTER



RT-11 MINITASKER
July, 1987

From the editor:

In a moment of silliness at Nashville, I volunteered to take on the job of your Minitasker Newsletter editor. (I think Bill Leroy actually slipped something in my drink.) So here goes with my first issue. I promise to try to maintain the standards of quality that Bill and his predecessors established.

The quality of the articles you'll find here is mostly up to you. So please send me your submissions. Don't be shy. If you have a problem looking for a solution, or a solution in dire need of a problem, or just anything you think might be of interest to the RT-11 community, send it in. Use any means you like - machine readable floppies (RT-format preferably), neatly typed, photo-ready hardcopy, or crayon on a brown paper bag (no toilet paper please).

Send your stuff to:
John M. Crowell
RT-11 Newsletter Editor
Multiware, Inc.
2121-B Second St. Suite 107
Davis, CA 95616

==========

This month's issue is devoted entirely to the RT-11 Wish List. On the following pages you will find a compilation of wished-for features for RT-11 gleaned from Newsletter submissions, DECUS symposia in the U.S., Europe, and Australia (Let's hear more from you Canadians.), and notes found on park benches. My thanks to Bob Walraven for compiling this list and submitting it to the Minitasker.

You'll find the RT-11 Wish List Survey response form in the back of this volume of newsletters. Please take the time to read the the wish list and fill out the form. Give each item a score. The scoring is simple:

4 - Very important (you really want it)
3 - Mildly important (you would like to have it)
2 - Mildly undesirable (you would prefer not to have it)
1 - Very undesirable (you really don't want it)

If you don't care one way or another about a particular item, don't score it. Just leave it blank.

If you know of other RT-11 users who don't get the newsletter, please copy the wish list and the form for them, and ask them to respond. The more ammunition we have, the better.

Send your completed surveys to:
RT-11 Wish List Survey
Multiware, Inc.
2121-B Second St. Suite 107
Davis, CA 95616

Pencils ready? ... Begin!

RT-11 WISH LIST

May 21, 1987

This is a collection of RT-11 Wish List Items contributed from RT-11 users in the U.S., Europe, and Australia. The RT-11 SIG Steering Committee would like to know your opinion on how important each of these items is. This is what we would like you to do: read the wish list items and decide which items are significant in your RT-11 or TSX-Plus environment(s). Then follow the procedures described at the end of the list for rating the importance of the items and send your ratings to us.

We will submit the list to the RT-11 Development Team and ask them to rate each item on how easy it would be for them to implement it. Combining this information with your ratings will allow us to determine which items we should most actively campaign for. For example, for those items that are both easy to implement and in high demand we will insist that they be implemented in the next release of RT-11. For those items that are difficult to implement but are in high demand, we will continue to bug the RT-11 Development Team about them. And for items in low demand or in questionable demand, we will not actively campaign.

(Some wish list items have been deleted because they have already been implemented or will be provided in the next release of RT-11. Other wish list items were eliminated because they overlaped with other items or were included within more general items.)

KMON COMMANDS

1.1 ASSIGN

Allow a filename to be assigned to a logical name, so that output to the logical name will be directed to a file rather than a device. (TSX-Plus already does this.)

1.2 D (DEPOSIT)

Allow the specified value to be ORed, ANDed, or XORed with the value currently in the addressed memory location. E.g.,

```
D aaaaaa=&nnnnnn        to AND
D aaaaaa=!nnnnnn        to OR
D aaaaaa=^nnnnnn        to XOR
```

1.3 F77

Add F77 command with appropriate switches.

1.4 INITIALIZE

Provide a switch to format a device before initialization.

1.5 INSTALL

Allow handlers to be installed even if they have a suffix different from that for which the monitor was sysgened. This would allow rarely used devices or handlers that use common hardware (e.g., serial lines) to be installed only when required, thereby not inadvertently munging a currently loaded device. For example,

```
.INSTALL MTZ        (becomes "MT" on the system)
.LOAD MT
```

1.6 R

Allow parameters (even if this processes the command with CCL rather than purely at a monitor level).

1.7 RENAME

a. When a device is specified on the first filename but no device is specified on the second filename, assume the device on the first filename is to be used for the second. That is, allow commands of the form **RENAME LD2:OLDNAM.TXT NEWNAM.TXT**.

b. When /SETDATE is used to change the date of a file, don't require the filename to be specified twice. That is, allow commands of the form **RENAME/SETDATE MYFILE.DAT**.

1.8 RESET

Provide a switch that allows a hardware reset to be performed also.

1.9 SET

a. Add **SET xx SHOW** to display handler set options.

b. Provide set options for the FORTRAN command to imply specific compilers and libraries. E.g.,

```
SET FORTRAN F77     --> SY:FORF77.SAV & SY:FORF77.OBJ
SET FORTRAN FIS     --> SY:FORFIS.SAV & SY:FORFIS.OBJ
SET FORTRAN EIS     --> SY:FOREIS.SAV & SY:FOREIS.OBJ
SET FORTRAN THR     --> SY:FORTHR.SAV & SY:FORTHR.OBJ
SET FORTRAN LD3:NHD --> LD3:FORNHD.SAV & LD3:FORNHD.OBJ
```

c. Allow file names to be specified instead of only device names. This would allow handlers such as XM to be set up under SJ or FB, and other useful operations. For example,

```
.SET SY:DUX.SYS PART=0,UNIT=3
```

d. Provide a generalized SET interface to any program.

1.10 SETDATE

Provide a SETDATE command for changing a filename date.

1.11 SETUP

Expand SETUP support of the VT100, VT200, and LA100.

1.12 SHOW

Add /OUT:filename switch so results can be sent to file or printer.

1.13 /WAIT

Make the /WAIT switch on copy, rename, etc. position dependent because it is not always necessary to change the volume on both devices.

1.14 /^X

Use ^Xn (with no CR) to access jobs instead of requiring the job name to be typed.

2 MONITOR

2.1 BACKWARD COMPATABILITY

Remove any backward compatible baggage from the monitor for versions before V5.0. This will make the monitors smaller but may require some rewritting of old codes and will certainly require rebuilding of executable programs.

2.2 COMMAND FILES

If a command file aborts, restore the QUIET/NOQUIET TT status and ERROR LEVEL to their values before the command file started.

2.3 COMPLETION ROUTINES

Allow completion routines to appear to the user as "soft interrupts". That is, allow the monitor to enter a completion routine with the code

```
.MFPS   -(SP)          ; Fake int. PS=PSW
.MTPS   #340           ; Go to priority 7
CALL    @USER.C        ; Enter user comp.
```

It would be the user's responsibility to handle the completion properly, using .INTEN, .SYNC, or .FORK, and returning with RTI, RTT, or RTS.

2.4 CONFIGURATION

Provide a minimal configured FB monitor with only 1 job capability.

2.5 CSI

For files on the right of the "=" (being LOOKUPed or ENTERed), allow an extended file spec notation. For example,

```
*x.tmp=du1:[dir1,subdir]fred.dat
```

2.6 DATA CACHING

Provide generalized data caching (including VM) similar to the functionality provided in TSX-Plus.

2.7 DATE

Extend date support beyond 40 years.

2.8 DCL

For all utilities, make ALL available CCL switches as DCL options.

2.9 DEFAULT EXTENSIONS

Allow the PRINT and TYPE command to look for .TXT and .DOC extensions if .LST is not found.

2.10 FILE TIMES

When a file is made permanent, set the time of day in the directory word that is not used after the file is made permanent. TSX already does this - use the same format. Add support for this feature in utilities.

2.11 GLOBAL REGIONS

XM global region directives should allow for control of the PDR access mode bits - specifically the "read-only" bits.

2.12 I & D SPACE

Add support for separate Instruction and Data space. (NOTE: RSX already does this; it is a way to effectively extend the amount of memory available to a job.)

2.13 LOGICAL DEVICE NAMES

Allow special Logical Device Names (that are used if they exist) for:

- Command files (current = DK , proposed = AT)
- FRUN (current = DK , proposed = FK)
- SRUN (current = SY , proposed = SK)
- Default library device (current = SY for SYSLIB and DK for ODT, proposed = LB)

2.14 MEMORY PARITY SUPPORT

Remove memory parity support from the monitor and replace it with an MP handler.

2.15 MULTI TERMINAL SUPPORT

Provide an optional protection mechanism for Multi Terminal Support, so that if a terminal is used with an auto-answer modem then files could not be modified without password verification. Also SIPP and DUMP would not be runable from a remote terminal without password verification.

2.16 SIZE OF MONITORS

Do not let the monitors grow in size (say by 10%) in the future.

2.17 SJ, FB, XM COMPATIBILITY

Add a pseudo JOB impure area to the SJ monitor so user programs can have the same structure for SJ, FB, and XM. Add appropriate fixed offsets to the Resident Monitor.

2.18 SPECIAL CHAINED EXIT

When a chained .EXIT is performed, allow a "@" file to be in positions other than the last line

2.19 SPECIAL DIRECTORY DEVICES

Add support for special directory wildcard operations that utilities such as PIP and DIR can use.

2.20 SUPERVISOR MODE LIBRARIES

Support Supervisor Mode Libraries. (NOTE: RSX already does this; it is a way of effectively extending the amount of memory available to a job.)

2.21 TERMINAL LOGGING

Provide a terminal logging capability where output to the terminal can also be directed to a disk file. TSX-Plus already does this (implement it in the same way).

2.22 UNDERGROUND JOB

Support an underground job that runs at lower priority than the background.

2.23 VIRTUAL JOBS

Provide mechanism so that jobs run under VBGEXE can connect to an interrupt vector.

2.24 VIRTUAL MAPPING IN XM

Map the monitor out of user space by default (ala TSX-PLUS). Also eliminate PAR1 and PAR2 restrictions.

3 UTILITIES

3.1 BINCOM

Tidy up messages on *.* comparisons (where an apparently random number of line-feeds appear between pairs of filenames).

3.2 BUP

a. Provide a /NOVERIFY that does no verification of the media at all (for file transfer rather than actual backup).

b. Provide a /VERIFY that verifies by comparison AFTER the data have been written. If errors are then found, allow insertion of new media.

c. Provide a /COMPRESS to make a compressed backup. (The VAX DECUS tape has a nice set of utilities LZCMP and LZDCM for compressing/decompressing.)

d. Allow specification of a new output device in addition to "Y" in reply to the request to insert the next volume. This would, for example, allow switching between DY0 and DY1 to speed backup.

e. Provide a /DIRECTORY to show a directory of files within a backup set rather than the backup set names themselves. This is needed for individual file retrieval.

3.3 DIR

a. Provide an ORDER switch to list the directory in alphabetical blocks. I.e.,

```
AAAAAA.FIL blkcnt date  ABAAAA.FIL blkcnt date
ACAAAA.FIL blkcnt date  ADAAAA.FIL blkcnt date
AEAAAA.FIL blkcnt date

BAAAAA.FIL blkcnt date  BBAAAA.FIL blkcnt date

CAAAAA.FIL blkcnt date  CBAAAA.FIL blkcnt date
CCAAAA.FIL blkcnt date
```

Currently, if the directory is more than one page long, you have to read one column all the way to the last page, then return to the first page for the second column.

b. Provide the ability to search for a file through multiple logical devices. For example, **DIR LD*:XYZ.*** would check LD0 through LD7 for any occurance of XYZ.

c. Use the low byte of a directory entry status word (currently unused) as a version number so that multiple backup versions of files can appear in the directory. A specific version of a file could be referenced in the same way as in VMS, i.e., **filename.ext;version**.

d. Use one of the currently unused bits in a directory entry status word to indicate that a file has been backed up. When the file is modified, the bit can can be turned off.

3.4 FILEX

a. Provide interchange for MAINDEC programs (both directions).

b. Allow FILEX to work with all the current DEC storage formats and with all supported devices.

c. Distribute the documented source of FILEX so users can add functionality.

3.5 FORMAT

a. Provide a switch for doing an automatic INITIALIZE after formatting.

b. Provide a SET option for FORMAT that causes user written formatting routines to be called when a device that is not supported by DEC is specified. Alternatively, have FORMAT invoke a command file FORMAT.CMD for calling FORMAT.SAV. The user could modify this command file as desired to invoke special device formatters.

3.6 IND

a. Make the extension for IND command files ".CMD", so that the two types of command files can be distinguished.

b. Make IND command files compatible with RSX IND command files. (E.g., read and ignore RSX UIC)

c. Provide hooks in IND (and RESORC) so that IND can determine what file an LDn unit is associated with and which LDn unit a logical disk file is associated with.

d. Let **@filnam**
first look for filnam.COM
if found, execute as if **$@filnam**
else look for filnam.CMD
if found, execute as if **$IND filnam**.

e. Provide a way to express ASCII characters in IND files without actually using the character (e.g. 33% => <ESC>).

f. Allow KMON commands to be issued from IND while IND has other files open.

g. Provide capability to pass strings or values from user program to IND when the user program exits.

3.7 KED, K52, KEX

a. Allow multiple tabs to be set and provide a command line switch to specify them. E.g., /T:nn:nn:nn:nn

b. Allow "SET WRAP nnn" to be set in the command line. E.g., /W:nnn[:NO]

c. Allow right margin justification to be enabled.

d. Allow the left margin to be specified.

e. Allow multiple learn sequences to be defined and be associated with any key preceeded by a <GOLD>.

f. Allow library searches for external files, where the libraries have the same syntax as the HELP.MLB file.

g. For specification of printer escape sequences, allow the sequence <ESCAPE><CTRL-CHAR> to appear as <GOLD>"0<n<32"<SPEC INS>. Alternatively, allow definition of printer escape sequences in a single character replacement table, or the ability to define short escape sequences that expand to larger sequences in the file.

h. Do not include in the wrap count sequences that begin with an <ESCAPE> or <CTRL-CHAR> and that are enclosed in brackets.

i. Allow the ability to have search mask control constructs in search for occurance (see TECO).

j. Allow a particular section to be cleared without having to clear the paste buffer first.

k. Provide optional journaling of an editing session so that work can be recovered if the system crashes.

l. Allow specification of a startup initialization command file (KED.INI) that is automatically called when the editor starts. Allow all SET commands to be included in this file plus a LEARN sequence and a default filespec.

m. Provide 8-bit support.

n. Allow DELWORD to delete only to the next non-alphabetic character so that all separators are allowed, not just spaces.

o. Provide one or more 16-bit counters that are inserted by a command and incremented/decremented each time they are inserted.

p. Provide a fully functional word processor for RT-11.

q. Do not delete the carriage return when the cursor is at the end of a line.

r. Implement the BACKSPACE key in KED so that goes to beginning of current line (the same as EDT).

s. Provide full VT220 support, including the additional keys.

t. Provide a REPLAY command that will execute a series of keystrokes from an external file.

u. Make the screen scrolling more efficient.

v. Allow the user to define the number of lines on the screen (useful for slow modem work).

w. Allow <gold><find> to search for special characters. (They could be inserted in the find string with <gold><spcins>.)

x. Implement <ctrl-H> to mean "move to start of current line".

y. Provide a way to go to an absolute line number.

z. Provide an overstike as well as an insert mode.

aa. FILL should put two spaces after a full stop (exclamation and question marks included).

bb. Provide a columnar cut and paste.

cc. Implement a SET REGION HOLD so that after using a marked region its definition is not lost (for example, after a WRITE SELECT).

dd. Make the command notation consistent with VAX EDT.

ee. Provide split screen (2-window) editing.

3.8 LIBR

a. Currently the only way to exclude a module name from a library is to extract all the modules and rebuild the library. A switch is needed for removing a module name from a library directory.

b. Allow conditionals for macro library building. For example,

```
.IF cnd,<expression>
.MACRO DIBOL
   DIBOL    ...
   ...
.ENDM
.ENDC
```

in HELP.MLB would allow the help file to be built with only those commands that are available at sysgen time. This feature would save storage space and edit time for special macro libraries.

c. Allow /UPDATE to appear on multi-line commands.

3.9 LINK

a. Provide a customization patch so that the /G switch could be made permanent.

b. Provide a DCL interpretation of the /G and /F switches.

c. Automatically enable the /G and /F switches whenever the /FORTRAN switch is used in COMPILE or EXECUTE.

d. Allow each overlay region of an overlaid program to have an optional INCLUDE switch that indicates that unsatisfied globals for that region are to be extracted from a common overlay library (specified on the first line).

e. Display the size of each overlay region in the link map, e.g.:

```
OVERLAY REGION nnn SIZE = nnnnn.   WORDS, SEGMENT
```

f. Provide a /SHORT option that causes an abbreviated map with perhaps only overlay section numbers, subroutine names, and subroutine sizes listed, or with symbols preceeded by a dollar sign deleted.

g. Support ISD object records so that debuggers for high level languages can easily pick up symbol information from the compilers. (MACRO and FORTRAN-77 already generate ISD records, but the linker just throws them away.)

h. Provide a switch option so that PARTICULAR modules from a library can be included in an overlay.

i. When an Undefined Global message is output, include the name of the module that referenced it.

j. Allow a library path to be specified so that the linker will search through a specified string of default directories. (This might be implemented through a .INI file.)

k. Allow linking with routines in a common sharable library that resides in a permanent global region. These routines would not have to be linked with the .SAV file.

l. Allow library modules to be put in the high memory portion of an XM job.

3.10 MACRO

a. Provide a switch that allows the error list ONLY to go to the line printer.

b. Allow HEXadecimal Radix to be specified.

c. Allow conditional directives such as .IF, .ENDC to have an optional name argument similar to the name argument for .MACRO, .ENDM, so that multi-level conditionals would be more readable.

d. Let a ";;" in a macro definition denote a local comment that would not be listed in macro expansions.

e. Provide a reversed .IRP list mode (perhaps called .IRPR). For example,

```
.macro  fcall name,list
.$$. = 0
   .irpr $$$,<list>
      .if nb,<$$$>
         .$$.=.$$.+1
         mov  $$$,-(sp)
      .endc
   .endr
   mov     .$$.,-(sp)
   mov     sp,r5
   call    name
   .globl name
   add     #<.$$.*2>+2,sp
.endm
```

would allow

```
...src: fcall   ijcvt  <#isrc,#jres>
```

to expand to

```
mov     #jres,-(sp)
mov     #isrc,-(sp)
mov     #2,-(sp)
mov     sp,r5
call    ijcvt
.globl ijcvt
add     #6,sp
```

for a nice way to interface to a FORTRAN-callable subroutines.

f. Make ".MCALL .MODULE" implicit.

g. Allow multiple modules in a single source, so that modules that are to be combined into a library do not have to be maintained in separate source files. For example, each module could begin with .MODULE name [,...] and end with .MEND.

h. Provide the capability of inserting ASCII text into a program in image mode. This might be done with .TEXT followed by the text enclosed in delimiters. For example,

```
.TEXT %
This is text to be inserted in ASCII
including escapes, CRs, LFs,
etc. %
```

i. Allow the date and time of assembly and linking (NOT module release level) to be included within a program. For example, .TIMTAG might create two 3-word data structures:

```
.WORD   date assembled
.WORD   (high) time assembled
.WORD   (low)  time assembled
.WORD   date linked
.WORD   (high) time linked
.WORD   (low)  time linked
```

j. Provide an .ASCDAT macro to insert the current date at that point in the program in the form **.ascii /dd-mmm-yy/**. (It should not have any additional spaces, and the "dd" should not be padded with a left zero!)

k. Include source file line number when an error messaage is output to the terminal.

l. Provide hex support for both input and listings.

m. Provide an alternative to .PRINT to avoid a conflict with the SYSLIB macro.

n. Allow the numbers 8 and 9 even if a terminating "." is not specified.

3.11 PIP

a. Allow the file name to be displayed BEFORE the copy function starts instead of after it is done.

b. Allow 8-bit support in ASCII mode copies

3.12 SEARCH

Provide a supported SEARCH utility.

3.13 SIPP

a. Provide a backward mode that is more convenient than the sequence <shift-6><RETURN> or <shift-6><LINE-FEED>, such as <ctrl-H>, which is implemented as the BACKSPACE key on most terminals.

b. Allow lower-case input with the command ";A".

c. Allow a string input mode.

d. Allow a single <ctrl-C> to abort the search and verify process rather than aborting SIPP completely.

3.14 SPOOL

Provide a switch (and internal page counter) so that a banner page can be specified to begin on either an even or odd page number.

3.15 SYMBOLIC DEBUGGER

Provide a symbolic debugger similar to the RSX FORTRAN-77 Symbolic Debugger that supports ISD object records and that can be used to debug using symbolic names in MACRO and FORTRAN sources.

3.16 TECO

Put it on the distribution kit.

3.17 VTCOM/TRANSF

a. Allow wildcard specifications.

b. Provide switch to control transmit speed.

c. Allow command file capability (i.e., specification of files that contain several lines for performing such things as logon procedures).

d. Provide a fixed-packet-size option so that "halving" algorithm is bypassed.

e. Restore packet size after two or three successes.

f. Allow arguments to be on the same line as the command.

3.18 MISCELLANEOUS

Provide lower case support for "Y" answers in PIP, DUP and BUP.

3.19 NEW UTILITIES

a. Provide an ARCHIVE utility (ala MS-DOS) with the ability to put into an archive library ANY file, including some form of compression with ECC correction, and with the ability to update a library from a working disk area with any files that were updated since they were last put in the library.

b. Provide a MAINT utility (ala MS-DOS on the Rainbow). MAINT brings up a listing of a directory and allows the to move among the file names with the cursor keys marking files for inspection, copying, deletion, renaming, etc.

c. Put UCL+ on the distribution.

4 PROGRAMMED REQUESTS

4.1 .CLOSE

Allow for an additional (optional) argument that specifies the size to make a new file when it is made permanent.

4.2 .DRSET

a. Add a <CMD> argument to mode field of .DRSET that that causes the address of the command part of a SET command to be passed in R0, and the unit number in R1, and ignores the VAL parameter. This would allow handlers to process the command line directly.

b. Allow .SAV and .REL files to use to .DRSET option so that the SET command could be used to enter data into a user's program. Allow an additional block to be added to .SAV and .REL files to make room for the setting code.

4.3 .DSTATUS

Include more returned information, such as
- o Address of device interface
- o Address of device vector(s) (table)

4.4 .GTLIN

a. Allow a completion mode so that the job is not blocked.

b. Provide full "SL" compatibility.

4.5 .PRINT

a. Provide a WAIT mode so that the message can be assured of being printed out when using mixed .TTYOUT and .PRINT. For example, to wait until .PRINT is done:

```
.PRINT  #message
.PWAIT
BCS     .-2
```

and to wait until .TTYOUT is done:

```
.PWAIT
BCS     .-2
.PRINT  #message
```

b. Add an optional completion routine address for interception of <XON/XOFF> control.

4.6 .SCCA

Add an optional completion routine address so that a single <ctrl-C> causes a completion entry even when .TTYIN has not been called.

4.7 NEW AND IMPROVED EMTS

a. .GLINR to get a line, but if no characters available return with C-bit set.

b. .GLINC: Completion routine version of .GTLIN.

c. Provide EMT calls to access the monitors internal conversion algorithms (ASCII to decimal, ASCII to binary, etc.).

d. Provide a .MEMPRO EMT that allows a memory area to be protected for chaining and a .CHAIN EMT that allows external programs to be called as subroutines. These would be used as follows:
   1. Load the start address and return address in protected memory.
   2. Pass common variables by using the FORTRAN convention of pointing R5 at an argument block, or by passing the address of an argument block through the chain area.
   3. Chain with a return switch that causes the monitor to preserve the job and the state of the registers and causes the monitor to restore the calling program when the called program exits.

   This feature would allow any program to call utilities such as PIP, DUP, or DIR as subroutine services.

e. Provide an EMT that returns the address of a channel's I/O Channel Block.

f. Provide an EMT that returns pointers into a job's impure area so that offsets that may change from release to release can be found.

g. Provide a single character version of .GTLIN so that a character could be obtained from the terminal OR the command file.

h. Provide a handler macro (.DRFWD) for forwarding a queue element to another handler.

i. Provide a mechanism for handlers to create and allocate queue elements, queue them to other handlers, and regain control when I/O completes.

j. Provide an EMT to return the physical name of a logical device without having to open a channel to the device.

k. Provide EMTs to mount and dismount logical disks.

l. Add equivalent of the RSX I/O status block to READx, .WRITx, and .SPFUN requests. This would allow handlers to return two words of information specific to the I/O request rather than the 2 bits that are now returned in the channel status word.

m. Add the equivalent of the RSX directive that gets a selected I/O packet from a handler's I/O queue. This would eliminate the need for the awkward internal queueing.

n. Provide programmed request to convert a virtual address to a physical address for any job's virtual address.

o. Provide a programmed request that runs handler SET code as a background job and that does not require overlayed SET code.

5 SYSLIB FUNCTIONS

5.1 NEW FUNCTIONS

a. Provide a set of routines for EIS, FIS/FPP, and CIS simulation on processors that do not have the appropriate hardware.

b. Provide a DATE functions that convert between the internal date word, multiple integers, and ASCII strings.

5.2 IMPROVEMENTS

a. Indentify those SYSLIB functions that are not reentrant.

b. Provide macro calls in the SYSMAC library that allow MACRO-11 programs to call easily the FORTRAN compatible subroutines in SYSLIB.

6 HANDLERS

6.1 DU

Support more than 256 Mb on MSCP devices.

6.2 LD

a. Provide a switch to supress automatic mounting on a boot.

b. Allow multiple dismounting, such as

DISMOUNT LD(0,1,4): or DISMOUNT LD:

c. Provide a way of installing very secure PASSWORD protection in blocks 0 through 6 of an LD device so that protected files are not so easily deleted.

d. Allow LD logical disks to be booted.

6.3 SD

Move as much of SDX and/or SDSX to high memory as possible.

6.4 SL

a. Allow SL to remain ON across a reboot.

b. Allow SL to be installed regardless of SYSGEN compatability.

c. Let <ctrl-H> to swap the two characters immediately BEFORE the cursor.

d. Let escape sequences and control characters that do not perform editing functions to be passed.

6.5 SP

Provide support for more than one spooled device (e.g. SP0 -> LP, SP1 -> LS).

6.6 TT

a. Remove TT: internal support from the FB and XM monitors so that the console handler can be modified without modifying the monitor.

b. Allow LET to work with the console handler in the same way as SL.

c. Support **SET TT DVORAK**.

d. Allow a .READC to TT with a word count of zero so that a completion routine is entered when terminal input is available. This would allow clean input from a .GTLIN to follow.

6.7 XC/XL

a. Provide hooks that can be used by terminal emulators.

b. Provide optional 8-bit support (SET [NO]BIT8).

c. Provide SET option to set ring buffer size.

6.8 MISCELLANEOUS

a. Provide a true multiterminal handler.

b. Remove unnecessary information from the handler block zero. Alternatively, provide a way for SET commands to go directly to a routine in the handler with no processing, allowing the handler to read in its own overlays and do its own interpretation. This would remove a number of restrictions on SET commands.

c. Set RETRY=n for ALL magtape handlers.

d. Provide handler(s) to support DHV11 and DHQ11.

e. When transfering files to magtape, keep the original date instead of substituting the current date. Provide a SETDATE switch or SET command for overriding the date used.

7 LANGUAGES

Provide a FLUSH routine for Fortran and BASIC that would force a write of any records currently in memory. For programs with long waits, this would improve their resistance to system crashes.

8 MARKETING

Make provision for sellers of RT-11 related products (e.g., TSX, STAR) to be resellers of RT-11 on the same distribution media as their product. This would remove the problem of mismatched version numbers.

9 TSX COMPATABILITY

9.1 CHARACTER ECHOING

Provide a sysgen option to defer character echoing on type-ahead.

9.2 COMMAND FILES

a. Provide a DISPLAY statement that displays the indicated text on the terminal even when SET TT QUIET.

b. Provide a PAUSE statement that suspends execution of a command file until a <cr> is typed.

c. Allow up to six optional parameter strings to be specified when envoking a command file, where the parameter strings are substituted in the command file where ever ^n, where n = 1 to 6, appears.

9.3 COMMAND LINES

a. Pass raw command lines in the chain area to user programs.

b. Do not automatically assume that a command line that starts with a number is an invalid command.

10 DOCUMENTATION

10.1 PROGRAMMER'S REFERENCE MANUAL

Add references about version number compatibility for system services. E.g.,

```
.LOOKUP     - V1 -
.DRBEG      - V3 -
.DRBOT      - V4 -
.PEEK       - V5 -
```

10.2 SOFTWARE SUPPORT MANUAL

Document how to load foreground, system, and handler code into high memory in the same manner as V5.4 does for its own routines.

10.3 STORAGE FORMATS

Collect in a single document a description of the storage handling formats that are currently in use by all DEC systems.

==========

RT-11 WISH LIST SURVEY

Score the importance of only those wish list items that are significant for you. If you don't care about a particular item, don't rate it. Use the following codes for scoring items:

4 - Very important (you really want it)
3 - Mildly important (you would like to have it)
2 - Mildly undesirable (you would prefer not to have it)
1 - Very undesirable (you really don't want it)

It will help us in tabulating the results if you could send us your ratings in machine-readable form. Use the same format shown here, but delete any items you are not rating, and list all item ratings in a single column.

If you have comments about specific items, put those at the bottom of the form and we will publish them in the Newsletter.

Send all hardcopy results to: RT-11 Wish List Survey
Multiware, Inc.
2121B 2nd Street, Suite 107
Davis, CA, 95616

Submit machine readable responses to any of the systems that maintain RT-11 DECUS tape submissions on line.

UNISIG

Jim Livingston, **Toolkit** Editor

My comments for this issue will be brief, even though brevity isn't one of my more commonly-encountered virtues. I've commented elsewhere (Digital Review, UNIX Views) on the results of the DECUS U.S. Chapter Board of Directors election. Suffice it to say, here, that I think the membership have given us a clear indication of how they feel about the direction the chapter has been taking over the last two to four years.

In this issue, I'm pleased to be able to include some of the handouts from the Nashville symposium. More will appear in subsequent issues of the **Toolkit** , as they are given some processing to reduce a natural excess of white space. This particular selection seemed to me to be in pretty good shape, as I got them.

I hope you find the handouts interesting; the talks that went with them were quite fascinating, and I'll bet you'll wish you had been there to hear them. Maybe next symposium you'll be able to convince your management that DECUS symposia are a good investment. I'm sorry to report that most of the sessions given by Digital, due to a misunderstanding, did not get taped, so those tapes won't be available, either. I'll do my best to get as much of what was said there in the **Toolkit** , and you should look also in the section for GAPSIG, who are publishing the X-windows talks we heard.

I'll be adding more from Nashville next month; don't miss it.

The Power of vi Made Easy

by

Al Delorey

ULTRIX(tm) Engineering Group

decvax!afd

Nomenclature

The "current line" is the line that the cursor is on.

The "current position" is the spot that the cursor is on.

Commands and cursor motion operate relative to the current position (or current line for line oriented commands).

Crash Recovery

To recover the file that you were editing when the system crashed invoke vi with the -r option and the name of the file. If you cannot remember the file name, simply type "vi -r", and vi will display a list of saved file names for you.

Insert Mode

i	- enter insert mode, before cursor
I	- enter insert mode, at beginning of line
a	- append, insert after cursor
A	- append, at end of line
ni<char>ESC	- will put 'n' occurances of 'char' on a line
o	- open a line for insert after curr line
O	- open a line for insert before curr line
ESC	- end insert mode, return to command mode Ends insert, append and open.
^D	- back out one 'shift width' Useful when auto indent is set.
^^D |	- back out to left edge of screen for current line only. Autoindent level is 'remembered' for next line.
0^D	- back out to left edge of screen Autoindent level is forgotten.
^V	- insert following control character
^T	- indent one 'shift width'
^W	- delete prior word
^U	- delete back to beginning of line

Vi Command Table

Objects	d	c	y	!	<	>	repeat
h or l				no	no	no	
j or k							
0 or ^				no	no	no	no
$				no	no	no	
w or e				no	no	no	
b				no	no	no	
f or F				no	no	no	
t or T				no	no	no	
n|				no	no	no	no
H,M,L							no
/pattern							no
?pattern							no
nG							no
'x							no
%							no
) or (							
} or {							
]] or [[							no
	d	c	y	!	<	>	repeat

Action key:

- d - delete
- c - change
- ! - filter
- > - shift right
- y - yank
- < - shift left

Object key:

- h - left char
- l - right char
- j - down line
- k - up line
- 0 - line start
- $ - line end
- ^ - first non-white space
- w - forw word
- b - back word
- e - end of word
- H - home scrn
- M - middle scrn
- L - to last scrn line
- / - srch forw
- ? - srch back
- nG - move to line number 'n'
- (- sent. start
-) - sent. end
- fx - forw find char (incl)
- { - para start
- } - para end
- Fx - back find char (incl)
- [[- sect start
-]] - sect end
- tx - forw to char (non-incl)
- % - match bracket
- Tx - back to char (non-incl)
- n| - column 'n'
- 'x - to mark 'x' (a-z)

A vi command is of the form: <repeat-count><action><object>

Cursor motion is of the form: <repeat-count><object>

The 6 actions above apply to one line when the action key is doubled.

Cursor Motion Examples:

3w	- Move forward 3 words
5j	- Move down 5 lines

Action-Object Examples:

d/yuk	- delete all text up to the pattern "yuk"
cw	- change the current word (end with esc)
c$	- change upto the end of the line
ctq	- change upto the letter 'q'
df)	- delete through the character ')'
<L	- shift all lines between cursor and the end of the screen to the left

RepeatCount-Action-Object Examples:

5cw	- change the next 5 words (end with esc)
d4j	- delete the next 4 lines
3dw	- delete 3 words

Repeat-Count Commands
(Without Objects)

Multi-line

cc	- change line(s)
dd	- delete line(s)
yy	- yank line(s)
<<	- shift line(s) left
>>	- shift line(s) right
!!	- filter line(s)
J	- join next line(s) to current line
S	- substitute line(s)
^E	- scroll down line(s)
^Y	- scroll up line(s)

Multi-char

x	- delete char(s)
s	- substitute char(s)

Repeat-Count Examples:

3dd	- delete 3 lines
6^E	- scroll screen up 6 lines
4<<	- shift 4 lines to the left
5x	- delete 5 characters

Miscellaneous Commands

^L	- redraw screen
^D	- scroll down a half screen
^U	- scroll up a half screen
^F	- scroll down a full screen (repaints)
^B	- scroll up a full screen (repaints)
mx	- mark position with letter 'x' (a to z) Note: the mark is invisible.
`x	- jump to mark 'x' (a to z)
``	- return cursor to previous context (place of last insert,delete,search)
~	- change case of current character
r	- replace current character
.	- repeat last change (substitute, insert, delete, etc.)
u	- undo last change
U	- restore current line (will undo several changes)
"np	- put back text from nth (1 to 9) previous delete
"xy	- yank text to named buffer 'x' (a to z)
"xp	- put text from named buffer 'x' (a to z)
n	- (letter n) find next occurence of last search pattern (/... or ?...)
N	- reverse direction of last search
;	- repeat last f, F, t or T search
,	- inverse of ;
&	- repeat last substitute (:s) command
^G	- show current file name & line number
^]	- find tag
z<CR>	- redraw with current line a top of screen
z-	- redraw with current line a botm of screen
z.	- redraw with current line a cntr of screen
ZZ	- write the buffer to the file and exit

Examples:

"ayL	- yank text from cursor to end of screen into buffer "a"
"ap	- put text from buffer "a" after cursor
"zy/pattern	- yank text, from cursor to given pattern, into buffer "z".
rd	- replace current char with "d"
"5p	- put back 5th previous delete
xp	- transpose two characters

digital

Pattern Matching

^	- beginning of line
$	- end of line
.	- any character
\<	- beginning of word
\>	- end of word
[str]	- any char in str
[^str]	- any char NOT in str
[x-y]	- any char between 'x' and 'y'
*	- any number of preceeding matches

Ex Commands

Ex commands may be used by starting the command line with a ":".

:w	- write changes out to file
:w pathname	- write buffer out to given filename
:w! pathname	- force file to be overwritten
:q!	- quit editing without writing file
:r pathname	- read in contents of the file (after current line)
:e pathname	- edit given file name
:e! pathname	- edit given file name (even if "no write" warning was given)
:e#, CTRL~, CTRL^	- edit prior file It is possible to yank text between files in named buffers.
:n	- edit next file in arg list
:<addr>s/old/new/g	- substitute addr can be n,n or % for entire file
:so pathname	- source the given ex/vi command file

An extended example of yanking text between files:

vi doc.new	- invoke vi to edit a file
.	
.	- do some editing
.	
:w	- write changes back to disk
:e doc.old	- edit alternate file
"ay5w	- yank 5 words into buffer "a"
^~	- edit alternate file (doc.new) Cursor is positioned at "current line"
"ap	- put text from buffer "a" after cursor

Upper Case Counter-Parts to Lower Case Commands

Upper case versions of commands typically do a similar function as the lower case command:

To end of line:	d : D	c : C	r : R	a : A
Opposite direction:	x : X	p : P	o : O	
To whole line:	s : S	y : Y		

Change, Replace, Substitute

To Change, Replace or Substitute... That is the question!

Change works with objects.
It puts you into insert mode.
Text will be right or left shifted to fit.
End with escape.

Replace works on a single char or to end of line.
It puts you into overstrike mode.
Text is not shifted.
End with escape.

Substitute works on a character or line; with repeat counts.
It puts you into insert mode.
Text will be right or left shifted to fit.
End with escape.

Ctags

To produce a tags file use the ctags(1) command: ctags file(s).

This will produce a file named "tags" where each line contains:

- An object (function name, macro definition, or typedef)
- The filename where the object is defined
- A search pattern that will find the object definition

With the cursor positioned at a function call, "^]" will:

- Lookup the function name in the tags file
- Write out the buffer (if autowrite is set)
- Edit the file name obtained from the tags file
- Position the cursor at the object definition

The command "vi -t tag" will edit the file where the tag is defined and position the cursor at the object definition.

Command Filters

A Command filter takes its input from the buffer, modifies it, and inserts its output back into the buffer in place of the input text.

!xcommand	- The text of the object (x) is passed as the standard input to the given UNIX command. The output from the command REPLACES the object text.
!!command	- Performs the command on the current line. The output from the command replaces the current line.
4!!command	- Performs the command on the next 4 lines. The output from the command replaces the 4 lines.
:r !command	- The output of the command will be read in after the current line.
:w !command	- Send the contents of the buffer to a command. Note that the contents of the buffer are not effected.

Examples:

!Gsort	- sort the all text to eob and replace those lines with the sorted output
:r !spell %	- run spell on the file and read the output of spell into the buffer
:r !date	- Add the date after the current line.
:w !wc	- count the contents of the buffer
:w !lpr	- print the contents of the buffer
:1,25w !lpr	- print lines 1 through 25

Defining Your Own Commands

Use the operator :map to define your own commands. The general format is:

```
:map x command
```

Where 'x' is a letter (or control character) and 'command' is any vi/ex command or combination of vi/ex commands.

'x' can also be a 2 character sequence from "#0" through "#9" which corresponds to the function keys F0 through F9; or 'x' can be an escape sequence that is generated by a keyboard key (the entire sequence must arrive within 1 second).

Examples:

```
:map g G            - g will act like G

:map v ~~~~         - v will change the case of the next
                      4 letters

:map s eas<esc>     - s will append the letter "s" to the
                      current word

:map E :e#<CR>      - edit alternate file

:map ^V^A n.        - ^A will find next & repeat change

:map #2 :!cat /usr/lib/vihelp<CR>
                    - PF2 key will act as a help key
```


Input mode Mappings

Use the operator :map! to define input mode commands. The general format is:

```
:map! x command
```

Where 'x' is a letter (or control character) and 'command' is any vi/ex command or combination of vi/ex commands.

'x' can also be a 2 character sequence from "#0" through "#9" which corresponds to the function keys F0 through F9; or 'x' can be an escape sequence that is generated by a keyboard key (the entire sequence must arrive within 1 second).

Examples:

```
:map! #1 *****      - F1 will insert "*****"
```

To emulate a modeless environment during insert:

```
:map! escOA escka   - ANSI (vt100) cursor up while inserting
:map! escOB escja   -              cursor down during insert
:map! escOC escla   -              cursor right during insert
:map! escOD escha   -              cursow  left during insert
```

Command Buffers

Vi allows you to copy (yank or delete) a vi command into a named buffer and execute that command. The command can then be invoked with @x (x is any letter a-z, where the command is located).

Example:

```
eas<esc>        - insert this into the vi buffer
"sy$            - yank the line into buffer "s"
@s              - execute the command in buffer "s"
```


Startup Commands

The environment variable EXINIT can be set in the .login file, in your home directory, to create your editing environment whenever ex or vi is invoked.

Map commands and named command buffers can also be set up in the EXINIT variable.

```
setenv EXINIT 'set ai aw ic sw=4 redraw terse|
        map g G|map v ~~~~|map E :e#<CR>|map s eas<esc>|
        map K G:r \!spell %<CR>|map S Gi/\<escA\>esc0"ad$dd@a<CR>'
```

ai	- auto indent
aw	- auto write
ic	- ignore case on searches
sw	- shift width = 4 spaces (<, >, ^D, ^T)
redraw	- redraw screen after deletes
terse	- terse error messages
wm	- warp margin (spaces from right edge of screen). Automatic line splitting.
ws	- wrap scan around end of buffer on searches
map	- define your own commands

Unused keys available for vi command mapping:

K V g q v * =

^I ^T ^V ^W

Unused keys available for vi command or insert mode mapping:

^A ^K ^O ^X F0 F1 F2 F3 F4 F5 F6 F7 F8 F9

You can also redefine any built-in vi command keys, as shown in the EXINIT example above (E, s and S).

digital

Source Command Files

Editor options and key mappings can also be specified in a file that you explicitely "source" after invoking vi. Options set in this manner only effect the current invocation of vi.

For example the file .dialexrc could contain the following commands for use when using a dial-in terminal.

```
set noredraw slow
```

Also the file .textexrc could contain the following commands for use when editing english text.

```
set ic wm=10
ab U ULTRIX-32(tm) Operating System
ab UEG ULTRIX-32(tm) Engineering Group
```

This command file would be invoked from within vi as follows:

```
:so ~/.textexrc
```


Example vihelp file

```
INSERT MODE
    i   - enter insert mode, before cursor;    a - append, insert after curso
    ESC - end insert/append mode, return to command mode
COMMAND MODE
  Actions -->          d       c       y       !       <       >     repeat
             ---------+-------+-------+-------+-------+-------+-------+-------+
  0      h, l         |       |       |       |   no  |   no  |   no  |       |
  b      j, k         |       |       |       |       |       |       |       |
  j      0, $         |       |       |       |   no  |   no  |   no  |   no  |
  c      w,e, b       |       |       |       |   no  |   no  |   no  |       |
  t      nG, /pattern |       |       |       |       |       |       |   no  |
           ----------+-------+-------+-------+-------+-------+-------+-------+
  Action key:
        d - delete      ! - filter        y - yank
        c - change      > - shift right   < - shift left
        The 6 actions above apply to one line when the action key is doubled.
  Object key:
        h - left char     l - right char     j - down line   k - up line
        0 - line start    $ - line end
        w - forw word     e - end of word    b - back word
        / - srch forw     nG - move to line number 'n'
  Additional Commands:
        x - del char      :w - write file    ^D,^U - scroll down,up
```

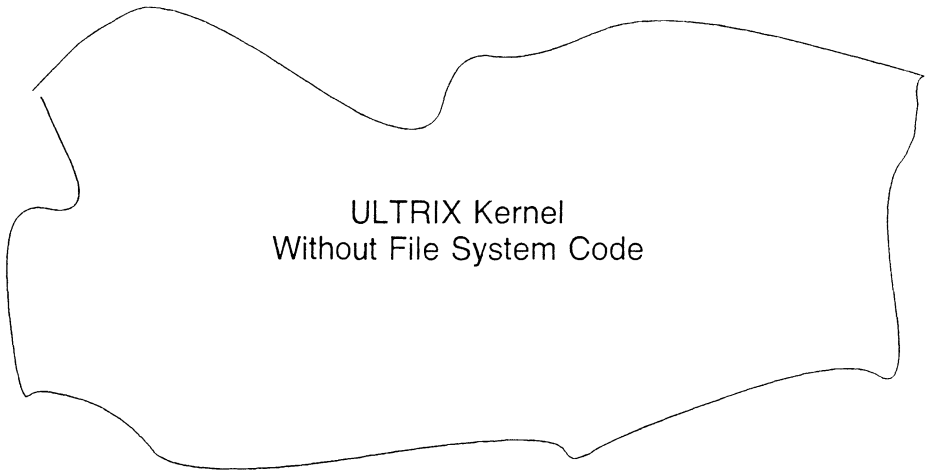
A Generic File System Interface for Unix

R. Rodriguez,
M. Koehler
Ultrix Engineering Group
Digital Equipment Corporation

GFS is the interface between the kernel and an arbitrary number of file system implementations. It has resulted in the separation of generic file system operations from their underlying implementation.

The GFS interface between the kernel and any file systems is strictly procedural. Kernel routines call GFS interface routines which in turn call the routines for a specific file system implementation.

Generic File System Architecture



GOALS

Support multiple hierarchical file systems without regard to implementation. This includes locale and state.

Provide equal or better performance to Ultrix V1.2 and 4.3BSD.

Provide equal or better file system reliability.

Have GFS control all common file system resources and force SFS's to use those resources.

Provide sufficient tools so that customers may add their own SFS's without source code.

ARCHITECTURE

Within the kernel, segregate FS specific functionality from generic functionality.

Coerce the kernel to use the generic operations.

SFS peers may only communicate through GFS.

Don't try to hide Unix semantics.

IMPLEMENTATION

The buffer cache supports logical and virtual blocks.

The mount table is the focus of all file system operations. The functions mapped through the mount structure define the specific file system implementation.

A generic structure is attached to the mount table providing data to the kernel and to the user.

The gnode is the focus of each file. It is divided into three parts: the generic piece, the common piece, and the specific file system piece.

THE MOUNT TABLE

```
struct mount {
        struct fs_data                  *m_fs_data;
        struct gnode                    *m_gnodep;
        struct gnode                    *m_rootgp;
        struct mount_ops {
                int                     (*go_umount)();
                int                     (*go_sbupdate)();
                struct gnode *          (*go_gget)();
                struct gnode *          (*go_namei)();
                int                     (*go_link)();
                int                     (*go_unlink)();
                struct gnode *          (*go_mkdir)();
                int                     (*go_rmdir)();
                struct gnode *          (*go_maknode)();
                int                     (*go_rename)();
                int                     (*go_getdirents)();
                int                     (*go_rele)();
                int                     (*go_syncgp)();
                int                     (*go_trunc)();
                int                     (*go_getval)();
                int                     (*go_rwgp)();
                struct filock *         (*go_rlock)();
                int                     (*go_seek)();
                int                     (*go_stat)();
```

```
        int                     (*go_lock)();
        int                     (*go_unlock)();
        int                     (*go_gupdat)();
        int                     (*go_open)();
        int                     (*go_close)();
        int                     (*go_select)();
        int                     (*go_symlink)();
        int                     (*go_readlink)();
        struct fs_data *        (*go_getfsdata)();
        int                     (*go_fcntl)();
        int                     (*go_bmap)();
        int                     (*go_freegn)();
    } *m_ops;
}
```


FS DATA STRUCTURE

The fs_data structure is maintained by each of the SFS. It is returned back to the user when a GETMNT system call is executed.

It contains the following data:

- Maximum, optimum, and physical block sizes
- Total and free on-disk inodes
- Total, free, and user consumable blocks
- Minimum size of executable before paging image
- UID of mount "owner"
- Mount point and "device" name

A GNODE

```
struct gnode {
        struct  gnode_req {
                struct      gnode *gr_chain[2];
                u_long      gr_flag;
                .
                .
                .
                struct      mount *gr_mp;
                .
                .
                .
                union {
                        struct mount *gm_mp;
                        struct text *gm_txp;
                } gr_un;
        } g_req;
        union {
                char pad[PADLEN];
                struct gnode_common gn;
                struct {
                        struct gnode_common _x;
                        char *free;
                } _freespace;
        } g_in;
};
```


THE COMMON PART

```
struct gnode_common {
        u_int               gc_mode;
        short               gc_nlink;
        short               gc_uid;
        short               gc_gid;
        quad                gc_size;
        struct timeval      gc_atime;
        struct timeval      gc_mtime;
        struct timeval      gc_ctime;
}
```

TIDBITS

The GETMNT() system call provides *accurate* information about mounted file systems. Several tools now take advantage of this information.

The GETDIRENTRIES() system call provides directory information in a generic format. Each SFS is expected to map their directory format into the generic format.

During shutdown, more integrity is insured by synchronously flushing gnodes and disk blocks and by attempting to unmount file systems. /etc/halt cleanly shuts the system down. FSCK can identify clean unmounted file systems and avoid checking them.

The ODS-2 filesystem is supported in read-only form in a prototype system.

Non super-users may mount file systems (with restrictions).

Synchronous file systems have been implemented.

SYNCHRONOUS FILE SYSTEMS

Can be selected on a per file system basis.

Op	Async		Sync	
	wall	sys	wall	sys
read	116K	64.43	117K	65.70
write	113K	44.32	83K	62.83
access	.00362	.00352	.00353	.00345
chmod	.00435	.00417	.00451	.00409
mkdir	.37019	.03439	.50793	.03447
trunc	.00480	.00463	.05221	.00528

Measurements taken on a MicroVax-II and an RQDX2 controller.

PERFORMANCE

A microvax-II with an RQDX3 disk controller.

A Vax 11/780 with a UDA50A disk controller.

A Vax 11/780 with a UDA50A disk controller compared to a VAX 8550 with a BDA50 controller.

Expectations indicate that a penalty is imposed by extra CALLS. This penalty has been eradicated through code reorganization and better structure linkages.

THE FUTURE OF GFS

There is interest in new SFS including AT&T RFS, Brunhoff's RFS, ODS-II, and write-once media (introduction of a much faster file system is being negotiated).

Work is ongoing for all paging and swapping to use GFS.

A publication of the GFS interface and its specification is in the works.

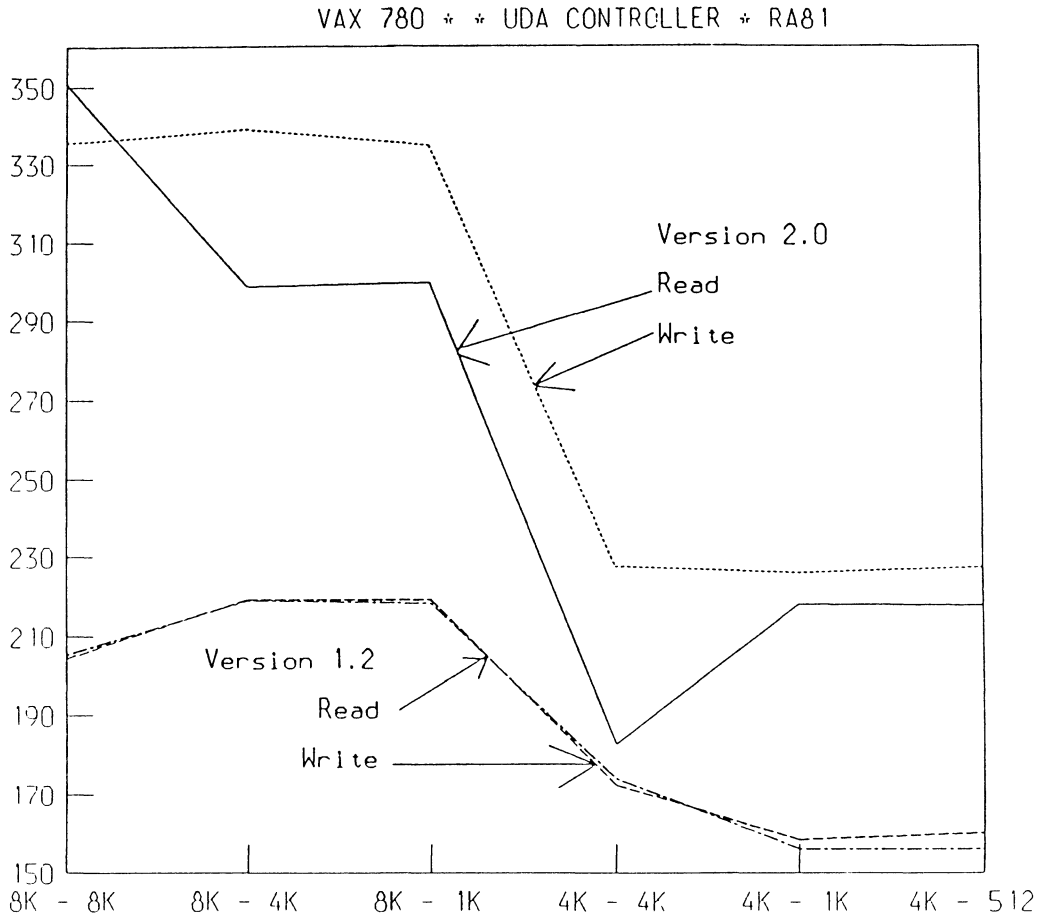


FIGURE 2

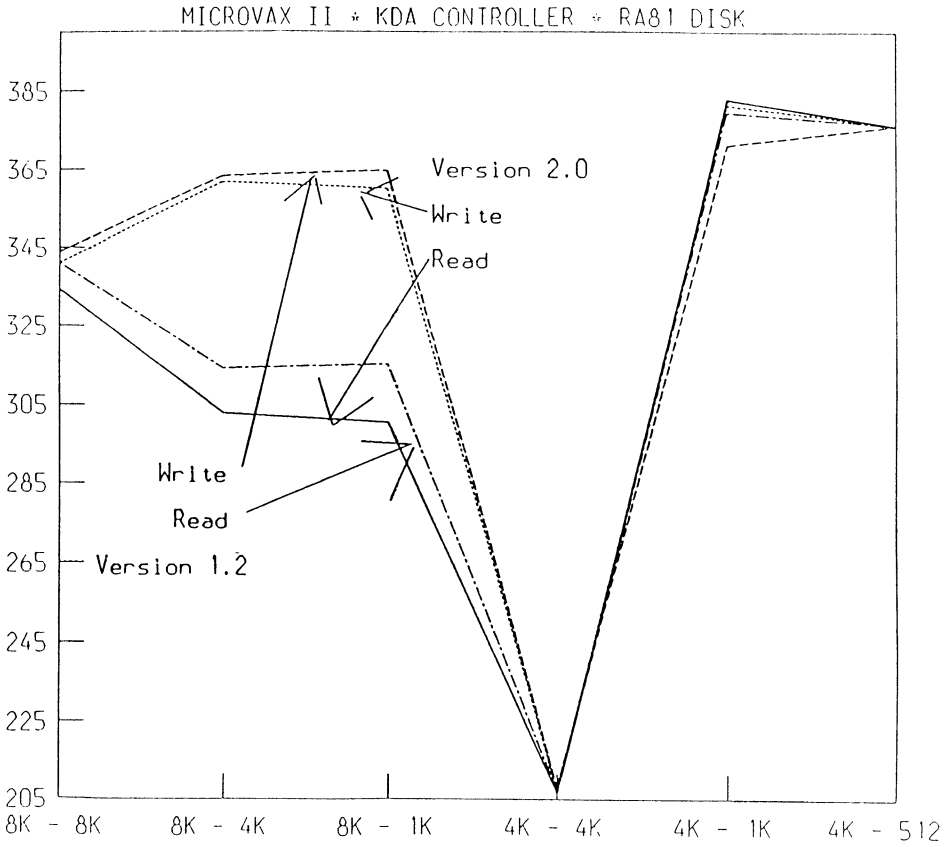


FIGURE 3

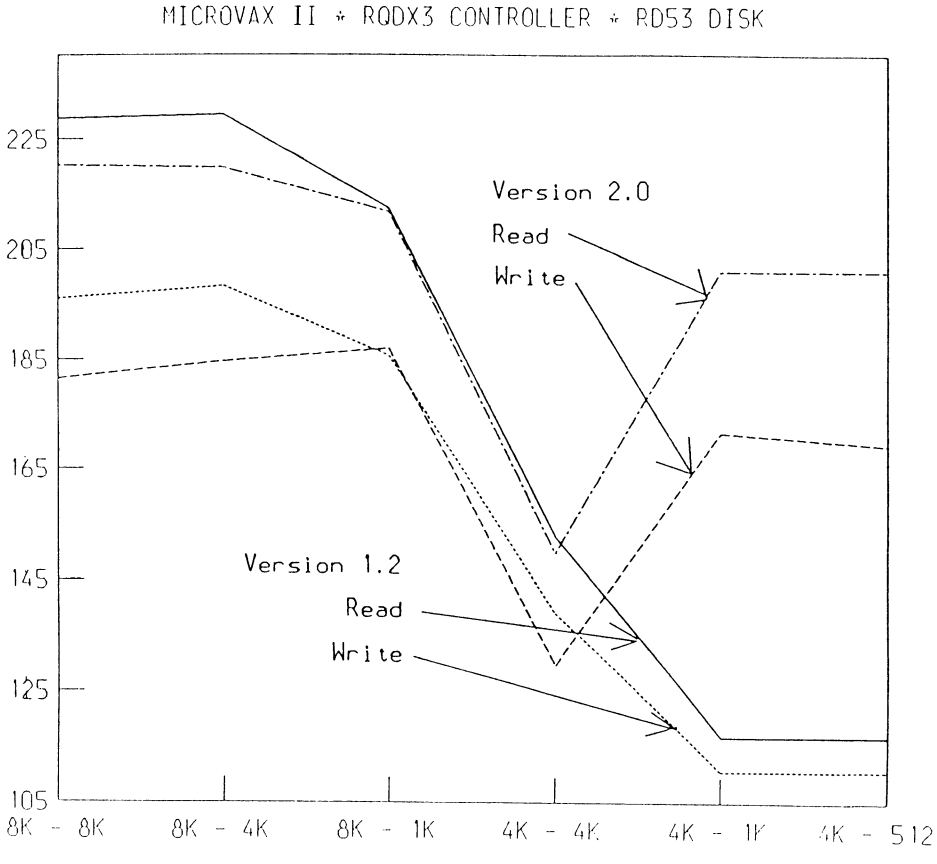


FIGURE 4

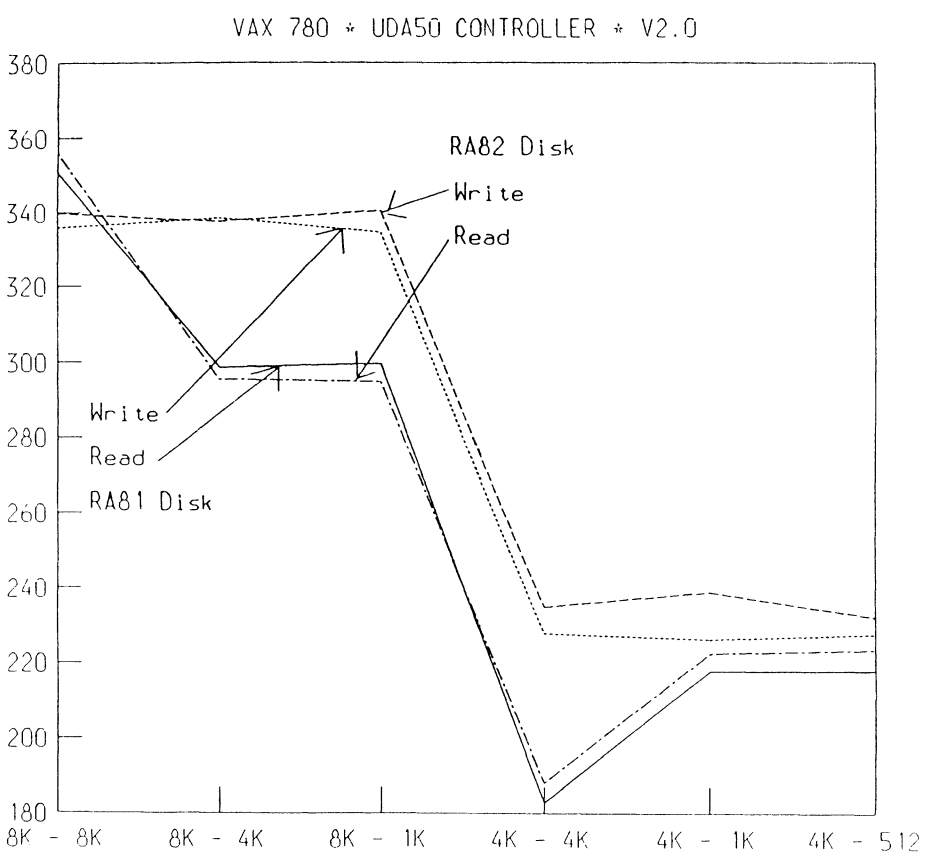


FIGURE 5

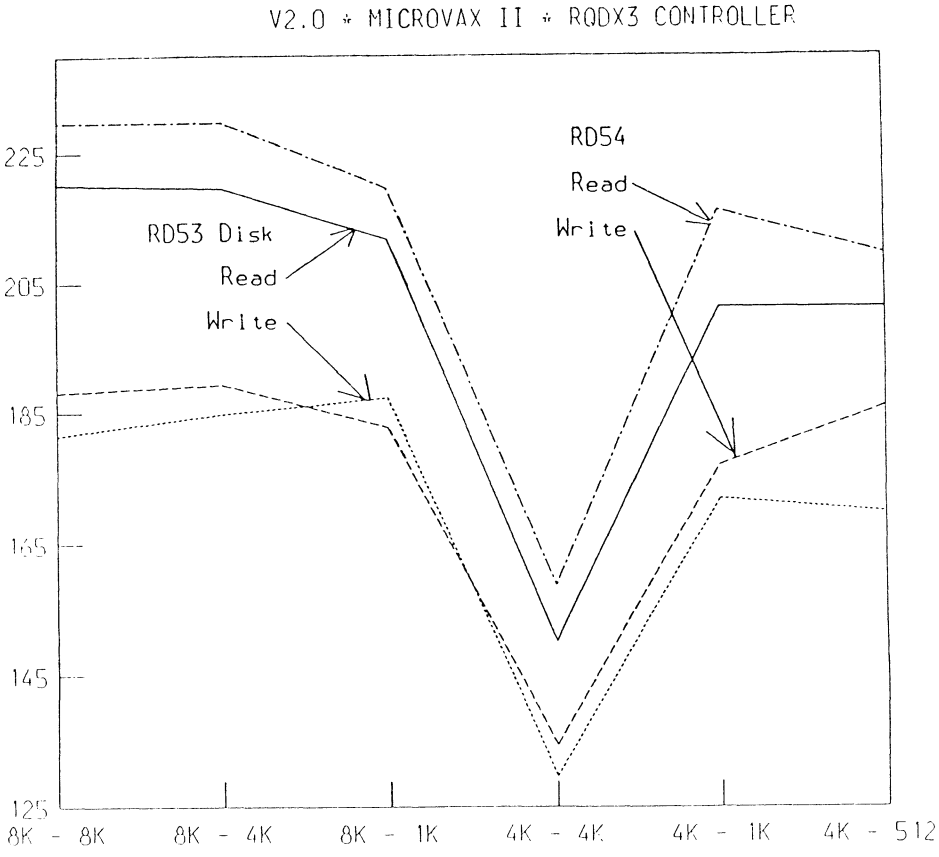


FIGURE 6

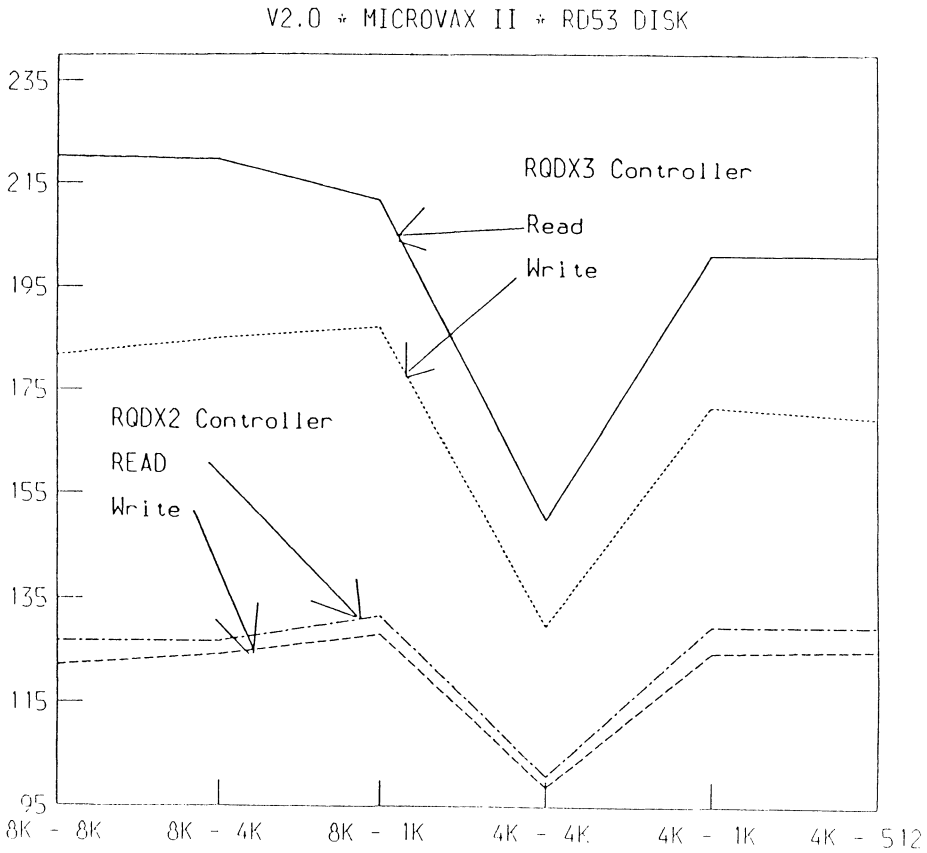


FIGURE 7

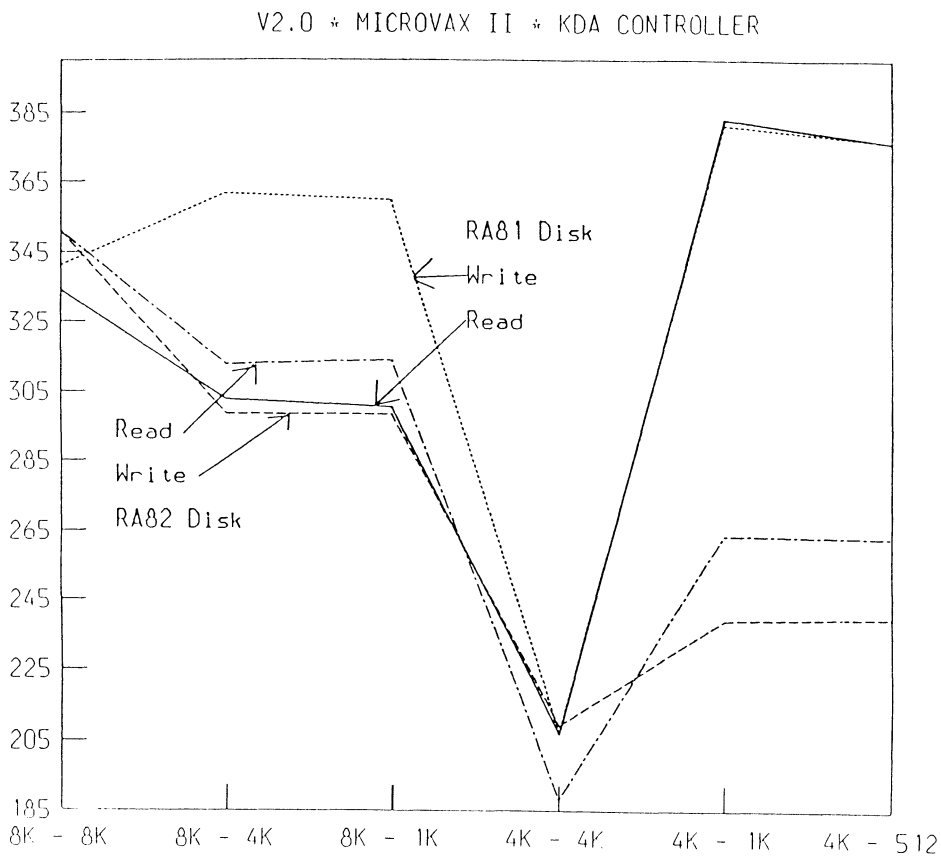


FIGURE 8

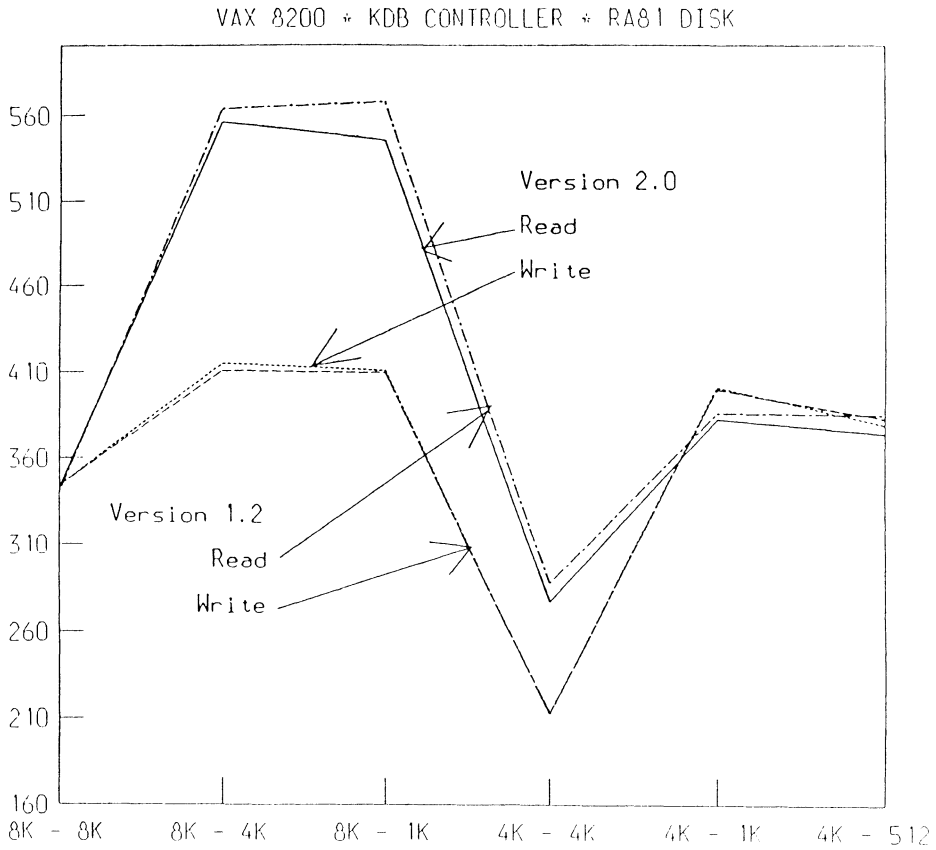


FIGURE 1

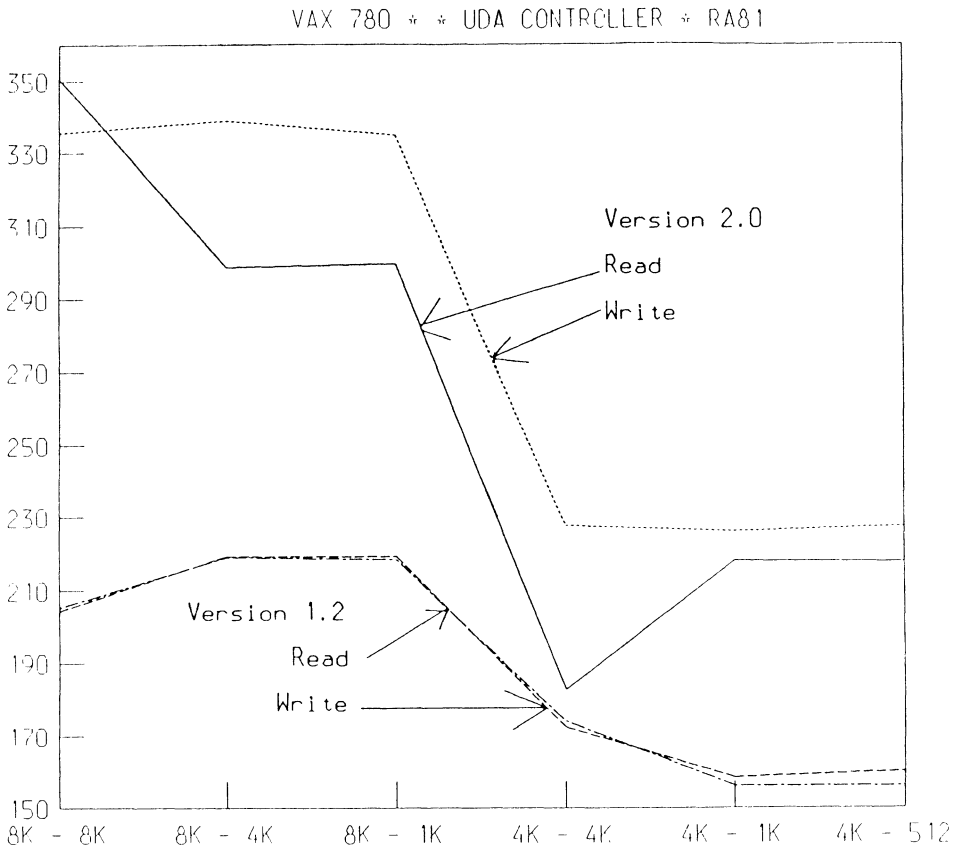


FIGURE 2

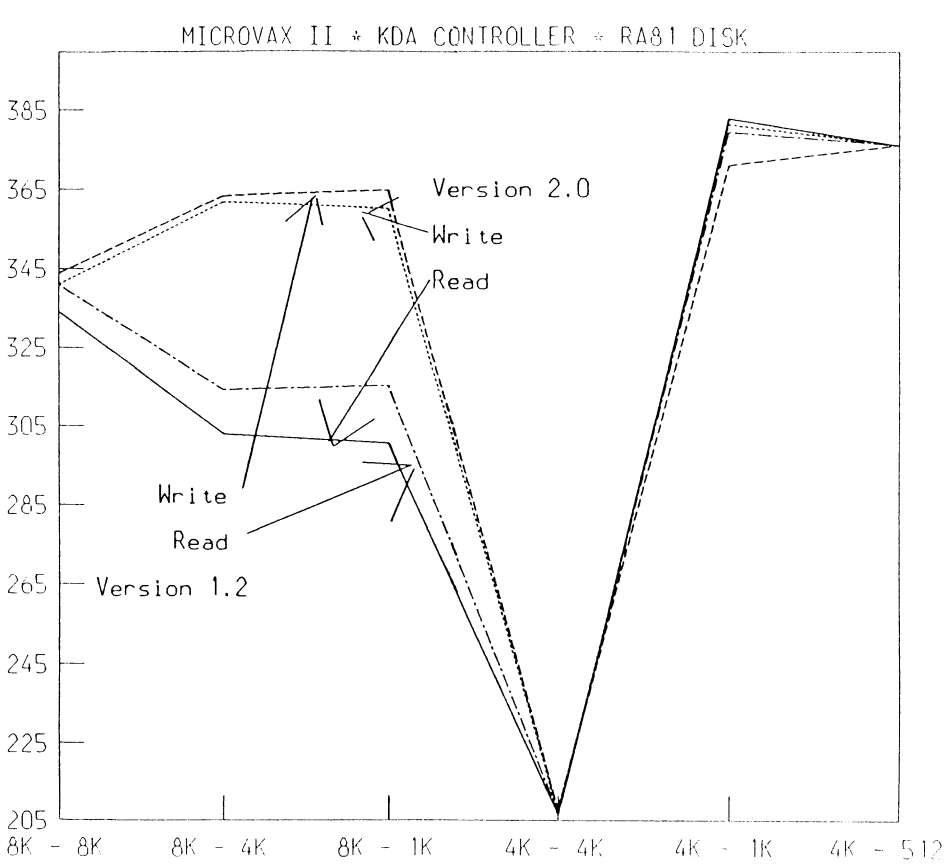


FIGURE 3

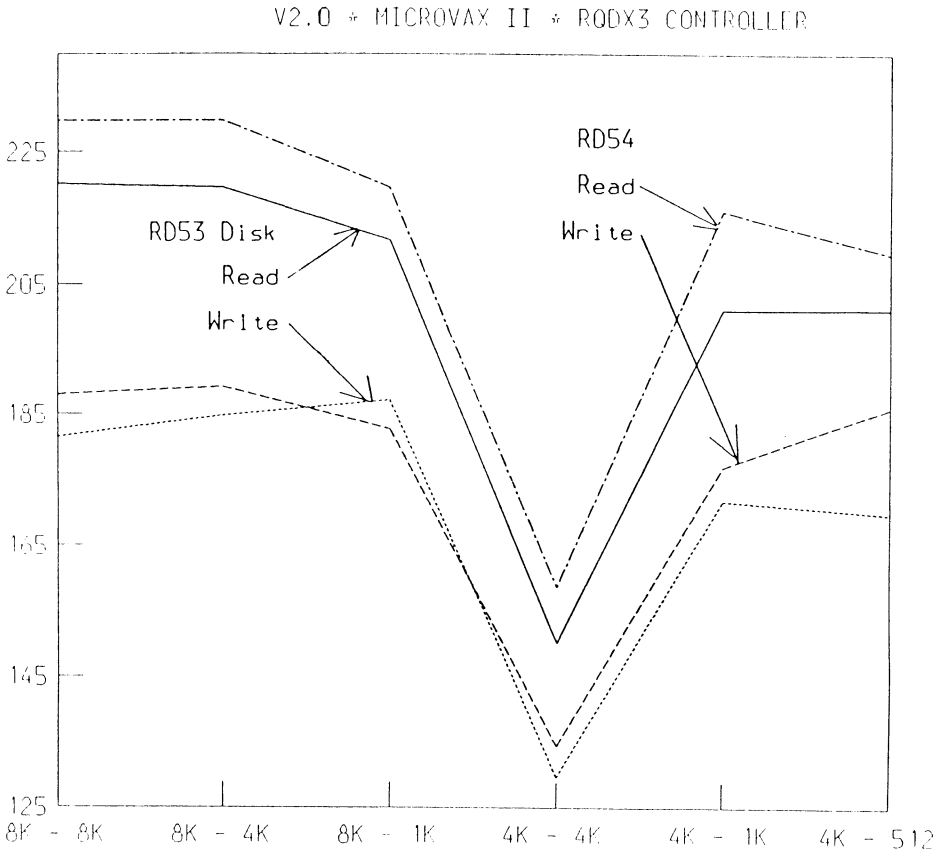


FIGURE 6

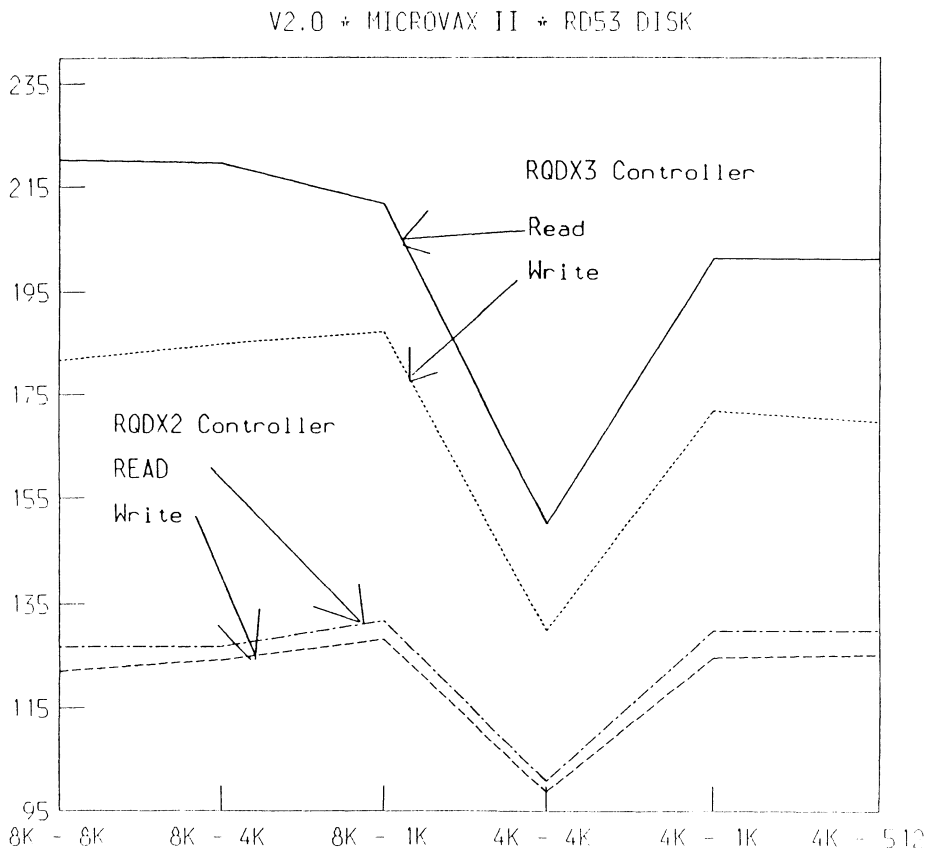


FIGURE 7

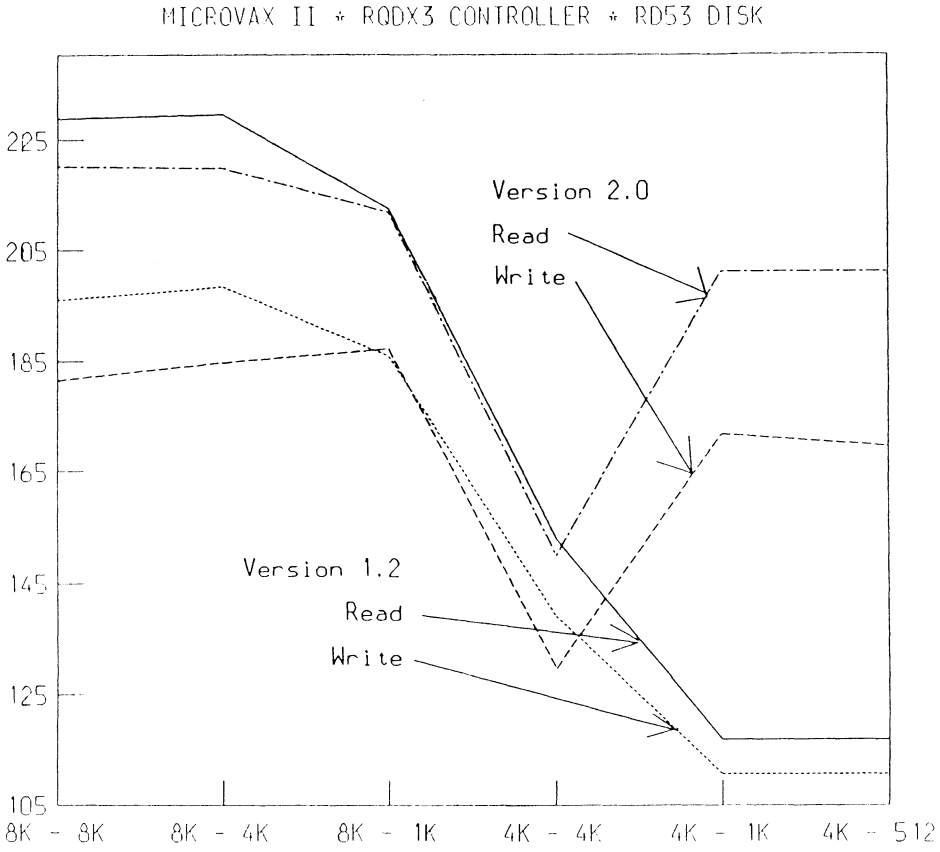


FIGURE 4

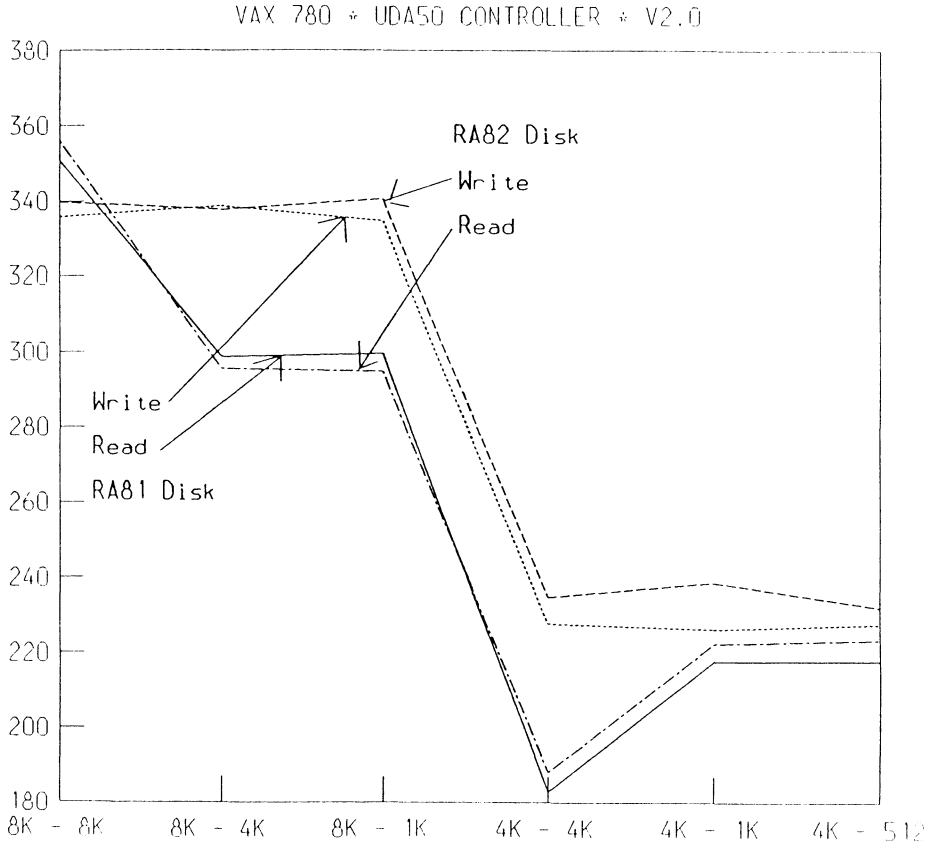


FIGURE 5

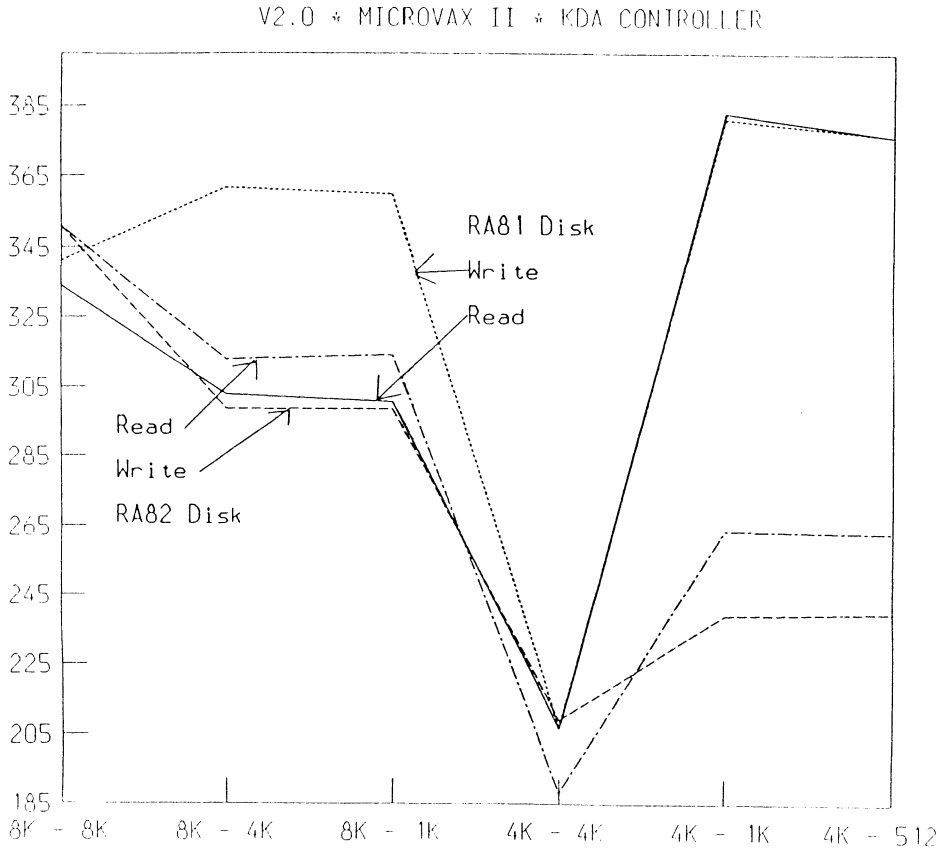


FIGURE 8

ODS II Under ULTRIX/GFS

Robert Rodriguez

Ultrix Engineering

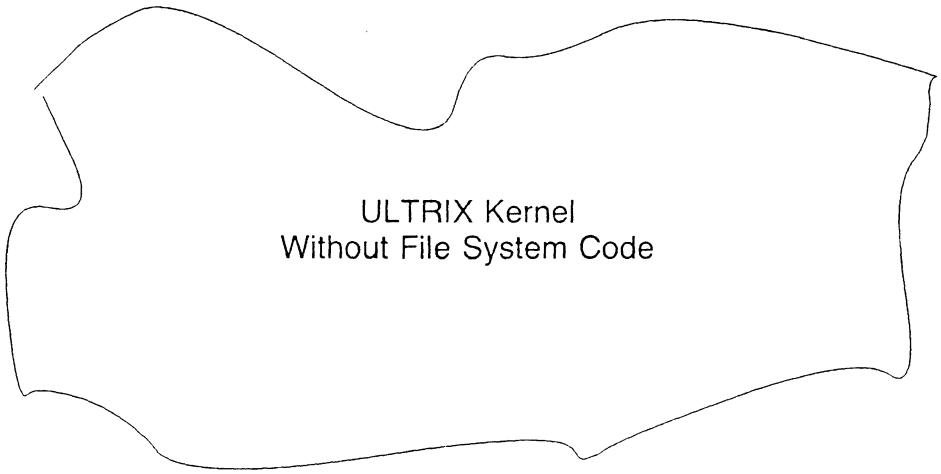
Digital Equipment Corporation

Merrimack, NH 03054

ULTRIX Version 2.0 and GFS

- Generic File System
- We claimed it could support other filesystems.
- Prove it!
- DEC/VMS has its own file system, ODS II.
- Can we put ODS II under GFS?
- What can we gain by doing this?

GFS Architecture



Generic File System Interface				
Native ULTRIX File System	Sun NFS	DEC FILES 11 ODS II (Proto)	MS-DOS File System (Proto)	System V File System (Proto)

ODS II Information

ODS II is constructed from Volumes. Each Volume is either an entire disk, many loosely coupled disks, or many tightly coupled disks. When the Volume is made up of loosely coupled disks each disk may become a Volume on its own. Tightly coupled disks appear as a concatenation of each of the media.

The disk allocation size (called the cluster factor) ranges from 512 bytes to 64 Kbytes. The name space is limited to alphanumerics(0-9A-Za-z), underline(_), minus(-) and dollar sign($). File names can be up to 86 characters (76 characters of name, a dot(.), 3 characters of type, a semicolon(;), and 5 characters of version number). The 'dot' and the 'semicolon' are always present and the name and type can be null.

ODS LAYOUT

All blocks in a ODS Filesystem are visible in the file system. UNIX hides pieces of the disk from users. Some of the special files that ODS defines are:

- INDEXF.SYS – Master index file
- BITMAP.SYS – The cluster allocation map
- BADBLOCK.SYS – Known bad block list
- 000000.DIR – Root directory
- CORIMG.SYS – System dump file
- VOLSET.SYS – Tightly coupled volume set definition
- CONTIN.SYS – Loosely coupled volume set definition
- BACKUP.SYS – Backup History
- BADLOG.SYS – Suspected bad block list

ODS INDEXF.SYS

The index file is the most important file on the disk. It contains the system boot block, the home block, and all of the file headers (both free and used ones). The index file layout looks like:

	Cluster
Boot Block -	1
Home Block -	1
More Home Blocks -	1
More Home Blocks -	2
Backup Home Blocks -	3
More Home Blocks -	3
Backup Index File Header -	4
Index File Bitmap -	5
16 Special File Headers -	5
Lots More File Headers	

ODS HomeBlock

The Home Block contains information about the locations of the reserved files, the device, the file system cluster factor, and volume identification and protection information. The HomeBlock looks like:

```
struct HomeBlock {              /* Home Block Structure */
     int       H_HBLB;          /* Home Block LBN */
     int       H_AHLB;          /* Alternate Home Block LBN */
     int       H_IHLB;          /* Backup Index File Header LBN */
     short     H_VLEV;          /* Structure Level and Version */
     short     H_SBCL;          /* Storage Bitmap Cluster Factor */
     short     H_HBVB;          /* Home Block VBN */
     short     H_AHVB;          /* Backup Home Block VBN */
     short     H_IHVB;          /* Backup Index File Header VBN */
     short     H_IBVB;          /* Index File Bitmap VBN */
     int       H_IBLB;          /* Index File Bitmap LBN */
     int       H_FMAX;          /* Maximum Number of Files */
     short     H_IBSZ;          /* Index File Bitmap Size */
     short     H_RSVF;          /* Number of Reserved Files */
     short     H_DVTY;          /* Disk Device Type */
     short     H_RVN;           /* Relative Volume Number */
     short     H_NVOL;          /* Number of Volumes */
     short     H_VCHA;          /* Volume Characteristics */
     int       H_VOWN;          /* Volume Owner UIC */
     char      H_unused0[4];
     short     H_VPRO;          /* Volume Protection Code */
     short     H_DFPR;          /* Default File Protection */
     char      H_unused1[2];
     short     H_CHK1;          /* First Checksum */
     short     H_VDAT[4];       /* Volume Creation Date */
     char      H_WSIZ;          /* Default Window Size */
     char      H_LRUC;          /* Directory Pre-access Limit */
     short     H_FIEX;          /* Default File Extend */
     short     H_RMINDAT[4];    /* Min. File Retention Period */
     short     H_RMAXDAT[4];    /* Max. File Retention Period */
     short     H_REVDAT[4];     /* Volume Revision Date */
     char      H_MINSEC[20];    /* Min. Security Class */
     char      H_MAXSEC[20];    /* Max. Security Class */
     char      H_unused2[320];
     int       H_SERIALNUM;     /* Media Serial Number */
     char      H_SNAM[12];      /* Structure Name */
     char      H_INDN[12];      /* Volume Name */
     char      H_INDO[12];      /* Volume Owner */
     char      H_INDF[12];      /* Format Type */
     char      H_unused3[2];
     short     H_CHK2;          /* Second Checksum */
};
```

File IDs are a unique 48 bit field describing each file. The File ID is as follows:

```
struct File_ID {
    u_short file_num; /* low 16 bits of file number */
    u_short file_seq; /* file number use id */
    u_char  file_ext; /* high 8 bits of file number */
    u_char  file_rvn; /* relative volume number */
}
```

Files have a unique triple (x,y,z) where x is ((file_ext<<16)|file_num), y is file_seq, and z is file_rvn.

ODS directories are contiguous files containing variable length directory entries. A directory entry contains the size of the entry, the maximum number of versions for this file, a flag field, the file name and its length, and a mapping of version numbers to File IDs. Version numbers need not be continuous. Directories only contain their children.

Directories have the following structure:

```
struct  odsdir {
    short   dlength;    /* length */
    short   verlim;     /* version limit */
    char    flags;      /* flags */
    char    nlen;       /* actual name length */
    char    fname[86];  /* usually less */
    struct {            /* each versions info */
        u_short version;
        struct  File_ID file_id;
    } ver[];            /* variable number */
};
```


All files are described by a fileheader. The fileheader contains all the information about the type of file, where it is located on the disk, the record attributes, the dates and times, the protection information, and a pointer to its parent directory. FileHeaders can be multiple blocks instead of just 512 bytes for large maps and/or large access control lists. The fileheader looks like:

```
struct fileheader {
    struct header header;
    struct ident  ident;
    struct map    map;    /* variable size */
    struct acl    acl;    /* optional & variable */
    u_short       chk;
}
```

ODS FileHeader

```
struct Header {          /* Standard File header */
    u_char  H_IDOF;      /* Ident Area Offset */
    u_char  H_MPOF;      /* Map Area Offset */
    u_char  H_ACOF;      /* Access Control List Offset */
    u_char  H_RSOF;      /* Reserved Area Offset */
    u_short H_FSEG;      /* Extension Segment Number */
    u_short H_FLEV;      /* Structure Level and Version */
    u_short H_FNUM;      /* Low Order File Number */
    u_short H_FSEQ;      /* File Sequence Number */
    u_char  H_FRVN;      /* Relative Volume Number */
    u_char  H_FNMX;      /* High Order File Number */
    u_short H_EFNU;      /* Low Order Ext. File Number */
    u_short H_EFSQ;      /* Ext. File Sequence Number */
    u_char  H_ERVN;      /* Ext. Relative Volume Number */
    u_char  H_ENMX;      /* High Order Ext. File Number */
    struct  UFAT H_UFAT;/* User File Attributes */
    u_long  H_FCHA;      /* File Characteristics */
    u_short H_RECPROT;   /* Record Protection */
    u_char  H_USE;       /* Map Words in Use */
    u_char  H_ACC_MOD;   /* Accessor Privilege Level */
    u_short H_FOWN[2];   /* File Owner UIC */
    u_short H_FPRO;      /* File Protection Code */
    u_short H_BKNUM;     /* Low Order BackLink File Num */
    u_short H_BKSEQ;     /* Back Link File Sequence Num */
    u_char  H_BKRVN;     /* Back Link File Rel. Vol Num */
    u_char  H_BKNMX;     /* High Order BackLink File Num*/
    u_short H_JOURNAL;   /* Journal Control Flags */
    u_short H_unused0;
    u_long  H_HWMARK;    /* File High Water Mark */
};
```


ODS FileHeader (cont.)

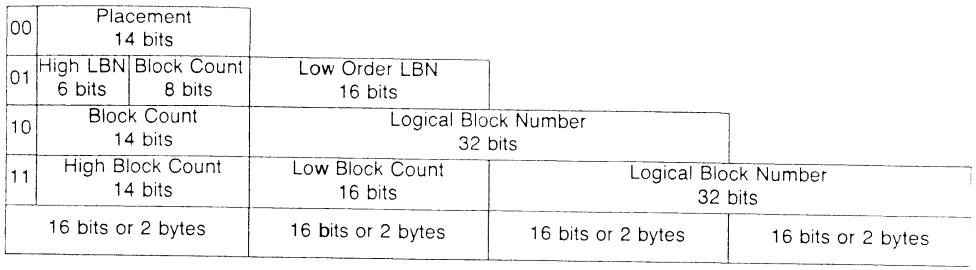
```
struct UFAT {            /* User File Attributes */
    u_char  RTYPE;       /* Record Type */
    u_char  RATTRIB;     /* Record Attributes */
    u_short RSIZE;       /* Record Size */
    u_long  HIBLK;       /* Number of VBN's Allocated */
    u_long  EOFBLK;      /* VBN containing End Of File */
    u_short FFBYTE;      /* First Free Byte */
    u_char  BKTSIZE;     /* File Bucket Size */
    u_char  VFCSIZE;     /* Fixed Control Area Size */
    u_short MAXREC;      /* Maximum Record Size */
    u_short DEFEXT;      /* Default Extend Size */
    u_short GLOBCNT;     /* Global Buffer Count */
    u_short FAT_JUNK[4];
    u_short VERSIONS;    /* Directory Version Limit */
}
```

ODS FileHeader (cont.)

```
struct Ident {  /* Ident area for a file - 120 bytes */
    char    I_FNAM[20];     /* File Name */
    u_short I_RVNO;         /* Revision Number */
    u_short I_CRDT[4];      /* Creation Date and Time */
    u_short I_RVDT[4];      /* Revision Date and Time */
    u_short I_EXDT[4];      /* Expiration Date and Time *
    u_short I_BKDT[4];      /* Backup Date and Time */
    char    I_FNAMEXT[66];  /* File Name Extension */
};

/* For an extended file header */
/* 510 bytes + 2 for checksum */
struct Map {
    u_short H_MAP[255];     /* Map Area */
    u_short H_CHKSUM;
};
```


ODS FileHeader (cont.)

Map Pointers



- Format 00 = 2 bytes, EXACT, ONCYL, LBN, RVN
- Format 01 = 4 bytes, 256 blocks, 2**22 disk LBNs
- Format 10 = 6 bytes, 16384 blocks, 2**32 disk LBNs
- Format 11 = 8 bytes, 2**30 blocks, 2**32 disk LBNs

ODS Under GFS

READ ONLY PROTOTYPE

```
struct mount_ops Ods_ops = {
    ods_umount,
    0,                  /* sbupdat */
    ods_gget,
    ods_namei,
    0,                  /* link */
    0,                  /* unlink */
    0,                  /* mkdir */
    0,                  /* rmdir */
    0,                  /* maknode */
    0,                  /* rename */
    ods_getdirents,
    ods_grele,
    0,                  /* syncgp */
    0,                  /* gtrunc */
    0,                  /* getval */
    ods_rwgp,
    0,                  /* rlock */
    ods_seek,
    ods_stat,
    ods_glock,
    ods_gunlock,
    0,                  /* gupdat */
    ods_open,
    ods_close,
    0,                  /* select */
    0,                  /* readlink */
    0,                  /* symlink */
    ods_getfsdata,
    0,                  /* fcntl */
    ods_freegn,
    ods_bmap
};
```


ODS From Scratch

- MOUNT RELATED – ods_mount, ods_umount, ods_getfsdata, ods_gget, ods_glock, ods_gunlock, ods_grele
- PATHNAME/GNODE RELATED – ods_namei, ods_freegn, ods_open, ods_stat, ods_close
- DIRECTORY RELATED – ods_getdirents
- FILE RELATED – ods_rwgp, ods_seek, ods_bmap

ODS MOUNT

ods_mount, ods_umount, ods_getfsdata, ods_gget, ods_glock, ods_gunlock, ods_grele

The basic procedure was for mount to read in all the ods specific information. Mount reads in the HOMEBLOCK, INDEX FILE FILEHEADER, the STORAGE BITMAP FILEHEADER, and the STORAGE CONTROL BLOCK which is the first block of the storage bitmap file. The storage control block contains the following information:

```
struct ods_scb {
    u_short W_STRUCLEV; /* storage map structure level */
    u_short W_CLUSTER;  /* storage map cluster size */
    u_long  L_VOLSIZE;  /* volume size */
    u_long  L_BLKSIZE;  /* blocking factor */
    u_long  L_SECTORS;  /* sectors per track */
    u_long  L_TRACKS;   /* tracks per cylinders */
    u_long  L_CYLINDER; /* number of cylinders */
    u_long  L_STATUS;   /* status word (long) */
    u_long  L_STATUS2;  /* secondary status */
    u_short W_WRITECNT; /* writer count */
    char    T_VOLOCKNAME[12];/* volume lock name */
    u_long  Q_MOUNTTIME;/* time of last mount */
    char    trash[456];
    u_short W_CHECKSUM; /* block checksum */
};
```


ODS MOUNT (cont.)

ods_mount, ods_umount, ods_getfsdata, ods_gget, ods_glock, ods_gunlock, ods_grele

The basis for mount is a routine called "readfh" that reads in fileheaders for the special files with file id's of 1(index) and 2(bitmap). These files are at positions that can be calculated from the HomeBlock which is also in a know place. Thus we had no real problem bootstraping the mount. We did run into a few bugs though. In order to get the storage control block we had to read the first block of the storage bitmap. That required the routine VBN_TO_LBN which converts virtual blocks of a file to logical disk blocks. In other words, it reads the map structure in the fileheader and figures out the block on the disk to read. That in itself is tricky to write let alone make efficient. We have one that works. The problem is that unlike UNIX, ODS doesn't have a table to index into. You must start at the beginning of the map pointers until you find the one that has the block you are interested in. This is fine for sequential reads (most are) and if you manage to cache the last map pointer. Otherwise, this is O(n**2).

Finally, we had to write ods_getfsdata and ods_gget. Ods_getfsdata required us to get the number of free gnodes and free blocks. The free gnodes is easy but the free blocks requires counting up the bits in a LARGE bitmap that takes some time. Many people played with many different bit counting algorithms. (Given an array B[N], count the number of bits that are on.)

ODS NAMEI

ods_namei, ods_freegn, ods_open, ods_stat, ods_close

In the namei area we had to do several things. First, ods_open and ods_close do nothing since ODS does not support special devices in the filesystem. Second ods_stat seemed easy at first if ods_namei could give us a gnode. However, UNIX and VMS store times differently. UNIX uses time since January 1, 1970 00:00:00 Local Time and VMS uses Nov 17, 1858 00:00:00. Some special assembly does the conversion between VMS and UNIX.

Finally, ods_namei is the first place we have to parse directory structures (we will do it in getdirents too). We must also read the fileheader in, store it in the gnode, and fill in the gnode common area. The directory reading code is fairly simple and localized to a small section of ods_namei. Once we walked through the directory (looking only at the latest version since UNIX can't ask for anything else), we then had to get a gnode. For ODS we cannot store a 512 byte fileheader in an 88 byte area so we have to allocate storage and keep a pointer in the gnode. This points out the reason for the ods_freegn routine. A cache of LRU gnodes is kept in UNIX. If a gnode is reused, then at that time the ods_freegn routine must be called to clean out the gnode of file system specific information (like allocated data pointers).

ODS NAMEI

ods_namei, ods_freegn, ods_open, ods_stat, ods_close

To finish off ods_namei we wrote a special routine to convert the fileheader information into an ULTRIX gnode. Instead of 48 bits for a file id we could use only the 32 bits of file number (H_FNUM). There is no clean way we know of to handle the 16 bits of ODS protection in 9 bits of UNIX protection modes. We know of no way to handle ACLs so we don't. ODS doesn't handle links right so we ignore them. The uid and gid come from the H_FOWN field. UNIX wants the last read and write time and the last change to the inode time and all ODS has is creation, revision, expiration, and backup times so we fake it. Last access is always the current time and last write and change to inode is just the revision time. We also had to handle lookups of '.' and '..' which refer to the current and parent dierctories. '.' was easy and '..' just used the back link file id in the fileheader.

ODS Directories

ods_getdirents

While we knew how to parse directories because ods_namei worked, we still had to write a lot of code to pack ODS directories into the UNIX structure. Most of this was bookkeepping since we got a buffer from the user to fill with variable sized entries from a variable ODS directory structure. Here we needed to be efficient since we were processing characters in small quantities (a few bytes at a time). The only sticking point was that directory blocks are not always full and contain the −1 signifying end of THIS block but there may be more in the next block. The UNIX structure follows:

```
struct gen_dir {
    u_long  gd_ino;          /* unique identifier*/
    u_short gd_reclen;       /* this entry's size */
    u_short gd_namelen;      /* the name's size */
    char    gd_name[MAXNAMLEN + 1]; /* the name */
};
```


ODS Read

ods_rwgp, ods_seek, ods_bmap

The last stumbling block was reading the data that the file contained. The required the ods_bmap function and ods_rwgp (without write at this time). The first problem we had was to remember that ods_bmap had to return the logical block of the first block of the cluster. So for a disk with a cluster factor of 16 (8K chunks) if you want to read the second virtual block ods_bmap had to return the logical block for the 32nd block in the file.

ods_rwgp basically follows the standard UNIX routine to read files. (Most routines we prototyped from the working UNIX routines). Once ods_bmap worked we only had a few things left to do. First, was making sure we caught the end of file correctly. ODS has the first free byte and we have to calculate the exact number of bytes in the file since it is not kept. Also, there may be more blocks allocated than used because of clustering.

ODS Performance

All measurements were taken on a Microvax II running ULTRIX V2.0 with 7 Megabytes of main memory and an Rd54 disk.

Read Performance in KiloBytes/second

	Read Size			
FileSystem	512	1K	8K	64K
UFS	118	154	206	196
ODS	15	31	204	388
RAW	25	42	184	357

ODS Comments and Afterthoughts

The performance of ODS II is largely due to the structure allowing 64K contiguous reads through the filesystem. All of the extra baggage that ODS carries for file attributes has been largely ignored. UNIX has no real way to access that information. There are 3 ways to consider what we have done here.

- Mounting READ-ONLY
- Mounting VMS-COMPATIBLE READ/WRITE
- Mounting UNIX-COMPATIBLE READ/WRITE

Extensions would have to be made to ODS II to allow UNIX special files and to do links and symbolic links correctly. The fact that each fileheader has a hard pointer to its parent causes much concern for doing links in the UNIX fashion. The directories '.' and '..' cause some problem since the name '.' is a legal name for a file in VMS/ODS-land. Finally, the UNIX file system has a concept of fragments which ODS doesn't. On a 128 cluster size ODS filesystem writing one byte in the file allocates 64K on the disk. There may be certain applications where this is unimportant (i.e., strictly large files on the disk).

We see no real problems implementing the ODS write functions. It simply matters which of the 3 options one chooses above. The first case is done and finishing off a UNIX version of ODS II is pretty easy at this point. The hard piece is VMS-COMPATIBILITY and to what extent that is attainable. It certainly is not 100% attainable within the present ULTRIX kernel.

GFS Revisited
or
How I Lived with Four Different Local File Systems

Matt Koehler
mjk@DECVAX.DEC.COM

ULTRIX[1] Engineering Group
Digital Equipment Corporation
Merrimack, New Hampshire 03054

1. INTRODUCTION

The Generic File System (GFS) interface was designed and implemented in the spring of 1986. A more complete reference of GFS can be found in (Rodri1986a). Once the interface was implemented, we were interested in determining its completeness. To achieve this, several widely used file systems were prototyped using the interface. These file systems were: UFS (4.3BSD's Fast File System), System V file system, the MS-DOS[2] file system, and ODS-II (native VAX/VMS file system).

The on-disk structures of each of the file systems were studied so that the maximum number of UNIX[3] (read 4.3BSD) semantics could be implemented.

Each architecture was further examined to understand performance implications of disk and file system layout and how that layout affects the file system implementation. Finally, the needs of NFS[4] were contrasted with the services provided by each file system architecture.

For the remainder of this paper, UNIX and UNIX file system semantics will refer to 4.3BSD and its file system functionality.

2. SYSTEM ENVIRONMENT

The system used for development was a MicroVax-II with seven megabytes of physical memory. The disk subsystem consisted of an RQDX3 controller and an RD54 (160 megabyte) disk drive.

The operating system was ULTRIX Version 2.0 configured with 10% (700K) of memory devoted to the buffer cache. The base operating system differs from V2.0 in that four prototype file systems were added. These additions required only the changing of a data and configuration file. Each of these prototypes took advantage of 4.3BSD's namei caching, LRU gnode caching, and buffer caching. The prototypes implemented the full set of 4.3BSD file system related system calls when ever possible. For example, the System V prototype included code to perform a mkdir system call.

3. THE ODS-II PROTOTYPE

ODS-II (also known as Files-11) is the file system that the RSX and VAX/VMS operating systems use. It is a prevalent file system in the mini-computer marketplace. The ODS-II prototype does not include code for writing or creating files, or modifying directories.

The ODS-II file system is constructed from volumes. Each volume may be an entire disk, many loosely coupled disks, or many tightly coupled disks. When the volume is constructed of loosely coupled disks, each disk may become a volume on its own. Therefore, not all disks in a loosely coupled volume need to be present for the volume to be mounted and used. The tightly coupled disk appears as a concatenation of all the disk media into a single disk address space.

The architecture supports file system block sizes ranging from 512 bytes to 64 kilobytes in 512 byte increments. The format also supports file names of up to 86 characters in length (not including revision numbers). The name space includes all alpha-numeric characters, '_', and '$'. While the prototype includes most of the ASCII character set, creating files with some characters may create difficulty when transporting the ODS-II file system to other operating systems.

3.1. ODS-II Architecture

All disk blocks within an ODS volume appear within the file system.. The boot block and file system control files appear in files with well known identifiers and locations. There are sixteen reserved file identifiers for use by the operating system. The ODS-II specification defines nine of the identifiers.

The first reserved file is **INDEXF.SYS**. The index file resides in the front of the disk and contains a system boot block, a home block, file headers (which are described later), and a file header free map. The home block contains information about the location of other reserved files, the device type, the file system block size, and volume identification and protection information.

The second reserved file, **BITMAP.SYS**, is the storage bitmap file. This file contains a map of all blocks on the system. Each bit within the map indicates whether the corresponding file system block is free. The third file, **BADBLOCK.SYS**, holds all known bad blocks on the device. The fourth file, **000000.DIR**, is the master file or root directory. The fifth file, **CORIMG.SYS**, is the kernel crash image (for example, vmcore.0).

The sixth file is **VOLSET.SYS**. This file defines the relationship between disks in a tightly coupled multi-disk set. The seventh file is **CONTIN.SYS**. This file describes loosely coupled multidisk volumes.

The eighth file is **BACKUP.SYS**. It is used to log and control backups. The ninth file is **BADLOG.SYS**. Disk blocks that the operating system suspects are becoming bad are listed (but not held) here.

3.1.1. ODS-II Directory Description

ODS-II directories are contiguous files constructed of variable length directory entries. A directory entry contains the size of the entry, the maximum number of versions allowed for this name, a flag for describing the directory entry, the file name and its length, and a mapping of version numbers to file identifiers. It should be noted that version numbers need not be continuous. Most implementations of ODS-II allow individual versions of a file to be removed.

The structures held within the directory file contain only the children of the directory. Neither a self-referential file ('.') nor a file referencing the parent ('..') exists.

3.1.2. ODS-II File Structure Description

The attributes of a file are described by the **FILE HEADER**. This header is somewhat equivalent to UFS's on-disk inode and is a common point for providing information about the file. There are four structures to a file header. They are:

3.1.2.1. ODS-II File Header

The first structure is a header containing information to check the validity and accessibility of the file. The major components of this structure are:

---

[1] ULTRIX, MicroVAX, VAX, DEC, and VAX/VMS are trademarks of DIGITAL.

[2] MS-DOS is a trademark of the Microsoft Corporation

[3] UNIX is a registered trademark of A.T.&T.

[4] NFS is a registered trademark of Sun Microsystems Inc.

- Indices to the remaining three substructures
- File record attributes
- Protection attributes
- File characteristics
- Ownership
- Backlinks
- Length

The file record attributes field contains information which describes the structuring of records within the file. There are several defined record formats. Records may be of fixed or variable length; they may be stream oriented with record terminators being carriage returns, line feeds, form feeds or combinations thereof. There is also the ability to have no record information at all. This is analogous to stream oriented files under the UNIX file system.

Protection modes are available on four operations: read, write, execute, and delete. In turn, this protection scheme applies for a user, a group, the world, and the system (root). The protection may also include restrictions to a particular VAX processor mode (user, supervisor, exec or kernel).

There are a myriad of file characteristics. Files may disable backups of themselves. They may change the write strategy to write back or write through. Files may request that reads are followed by a read/compare. They may likewise request that write operations are followed by a read/compare. Files may ask that through best effort, blocks are allocated contiguously. Also files may request that when a block is deleted, the deleted disk block will be overwritten.

The ownership of a file is described by a user identification code (UIC). The UIC is similar to uid and gid used by UNIX systems. The length field is the size of the file in bytes. Finally, the backlink is a pointer to the parent directory.

3.1.2.2. ODS-II File Identification Header

The file identification header contains identity and accounting information about the file. The major components of the file identification header are:

- File name
- Revision number
- Create, revision, expiration, and backup dates

3.1.2.3. ODS-II Map Header

The map header contains map structures allowing file blocks (VBNs) to be transformed into disk blocks (LBNs). The map structures describe the number of contiguously allocated blocks from a starting disk block. Three map types exist allowing up to 2^{30} contiguous blocks to be mapped from a starting block represented in up to 2^{32} bits. Different map types may be intermixed within the same map header. The mapping strategy allows files to have holes or unallocated spaces.

3.1.2.4. ODS-II Access Control Lists

The access control lists (ACLs) are an optional part of the file header. The ACLs hold data permitting exceptions to the permissions described in the ODS-II file header. It is interesting to note the ACLs may be used to permit or deny access to a file for a group or individual.

3.2. ODS-II Prototype Design and Implementation

Since the ODS-II file header is a crucial element for describing a file, it is necessary to associate the header with a gnode. There is space reserved in a gnode for file system implementations to store file information. The space is 88 bytes long; file headers contain 512 or more bytes. Therefore, each *ods_gget* must allocate additional space for the header and attach that space to the gnode. To allow LRU gnode caching, the gnode must preserve the reference to the file header. It becomes necessary to recover the header space when the gnode is reused for another file. Adding a mechanism for header space recovery to the GFS interface allows the LRU gnode cache to work.

3.2.1. ODS-II VBN to LBN Mapping

Another functionality requiring thought is *ods_bmap*, or the mapping of virtual block numbers (VBNs) to logical block numbers (LBNs). With the exception of ODS-II, all file system prototypes have an array of file block indices permitting one to one VBN to LBN transformations.[5] A map structure in ODS-II maps a range of VBNs. This causes *ods_bmap* to iterate through each map structure until the VBN is found.

3.2.2. ODS-II Pathname Translation

The lack of '.' and '..' causes *ods_namei* to special case these names and return valid information. Likewise, *ods_getdirent* needs to return '.' and '..' when returning directory information. Further, the file **000000.DIR** is the root directory for the file system. Since **000000.DIR** and '.' refer to the same directory, **000000.DIR** cannot appear in the name space without causing looping in file system traversals.

Finally, providing code for VBN allocation (that is, extending the file) is complicated by strategies for contiguous files and by multi-disk volumes.

3.3. ODS-II Performance

Since the ODS-II prototype does not support write, it was necessary to use an existing file to measure read performance. **INDEXF.SYS** is the largest file on the disk containing just more than 2 megabytes. This file was read repeatedly asking for a varying number of bytes. The following table compares ODS-II performance to reading blocks from the character device.

Read Performance (in kB/sec)							
	512b	1K	4K	8K	16K	32K	64K
ODS-II	15	31	173	204	280	350	388
Device	25	42	132	184	257	320	357

To measure ODS-II performance with read requests greater than 8K, it was necessary to rebuild the kernel to change the maximum buffer size to 64K. It was also necessary to change *ods_rwgp* to override the clustering factor.

ODS-II attains better file system throughput by increasing the file system block size. Binaries and other large files may be well hosted on an ODS-II volume.

3.4. ODS-II Supporting NFS

There are no architectural limitations restricting the use of ODS-II as a file system served by NFS. After fixing a few implementation problems, NFS worked perfectly.

[5] Calling an MS-DOS FAT chain an array is stretching the analogy, see section 4.1.2 for a detailed explanation

3.5. ODS-II Strengths

The ODS-II architecture is rich with functionality.

There are several file attributes that are not present within any of the other file system architectures. Also, the ability of file systems to span disks is attractive. Finally, the file system performance is good. Allowing 64K blocks reduces the number of transactions handled by a disk controller.

3.6. ODS-II Limitations

The inclusion of delete and an additional grouping by system causes ODS-II file permissions to be not completely representable using UNIX file system semantics.

Only two types of files (regular files and directories) are defined in the ODS-II architecture. Block and character special files, soft links, and named FIFOs cannot be created or accessed. Since each file header has a pointer back to its parent directory, hard links cannot be created.

The lack of '.' and '..' in directories causes problems. While the functionality associated with these files can be emulated, their names are not reserved in the ODS-II file name space.

Finally, ODS-II allows for file system block sizes to be very large. Since there is no concept of fragmentation, a one byte file on a 64K block file system consumes 64K.

4. THE MS-DOS PROTOTYPE

The popularity of the IBM[6] personal computer causes the MS-DOS file system to be one of the most prevalent file system architectures in existence.

The MS-DOS architecture supports file system block sizes of 512 bytes, 1K, or 2K. The 2K file system block size has been added for support of 20M hard drives. File names can have a length of 11 characters which is broken into an 8 character base and a 3 character extension. The base and extension characters are separated by a '.' (which is not stored in the file name). Even though only the names '.', '..' and names beginning with '\0345' or containing '\0' are reserved, most implementations only allow upper case letters, numbers, and a few symbols in the name space.

4.1. MS-DOS Architecture

An MS-DOS file system contains four components: the boot record, the FAT, or file allocation table, the root directory, and file system data blocks. The FAT is usually replicated once for reliability.

4.1.1. MS-DOS Boot Record

The beginning of the boot block contains an index to the boot record. The boot record holds data concerning the format and size of the disk. Included in this record is the size of a data cluster (the file system block size), the number of root directory entries, and the number of sides and heads on the media. An operating system ID, the size and number of copies of the FAT, and the number of system reserved sectors are also included in the boot record.

4.1.2. MS-DOS File Allocation Table

The FAT contains the mappings of VBNs to LBNs. There is a FAT entry for each data block on the file system. FAT entries appear in two formats: 12 bit (1.5 bytes) and 16 bit. The 12 bit format is the most common. 16 bit FATs were introduced with support for 20M hard disks.

Allocated FAT entries are members of a data block chain. Since each entry identifies a data block, determining the value held in the current entry yields the next entry in the FAT chain (and hence next data block). See section 4.1.4 for further information.

[6] IBM is a registered trademark of International Business Machines

FAT entries zero and one are reserved. Entry zero contains a media identifier. Entry one is unused. All other entries contain either an index to the next entry in the FAT chain, a flag indicating the end of the chain (entries with the high 9 or 13 bits set), a bad block flag (0xFF7 for 12 bits, 0xFFF7 for 16 bits), or unallocated block flag (0).

4.1.3. MS-DOS Directory Description

MS-DOS directories are files constructed from not necessarily contiguous blocks containing fixed length directory entries. Each directory entry is 32 bytes long and is defined as follows:

```
struct msdos_dir {
        u_char      md_name[8];
        char        md_ext[3];
        u_char      md_attr;
        char        md_fill[10];
        u_char      md_tod[4];
        u_char      md_cluster[2];
        u_char      md_size[4];
};
```

Each of these fields will be described in section 4.1.4.

As expected, there is one directory entry for every file in the directory. With the exception of the root directory, the first two entries in each directory are for files '.' and '..'. If the first character of the file name is '\0345', then the slot is unused. If the first character is '\0', then the end of the directory has been reached.

The root directory is handled differently from its sub-directories. First, the files '.' and '..' do not exist. This is reflected by the fact that the root directory resides in a reserved area on disk. Further, the root directory is a static set of contiguous blocks causing there to be a fixed number of directory slots.

The directory entry contains data concerning file attributes. Since directories (other than the root of the file system) have two entries (one naming the directory, the other '.'), there are two sets of attributes for each directory. As will be discussed in the implementation section, this causes problems for updating directory attributes.

4.1.4. MS-DOS File Description

The attributes of a file are described in the file's directory structure. Since only 8 bits are used for protection, security is minimal. The protection allowed is:

- Read only
- Hidden — analogous to file names beginning with '.' in UNIX
- System — a remnant of CP/M
- Volume label — refers to the name of the volume not to a file
- Subdirectory
- Archive — indicates if the file has been backed up but not modified

The time stored in the directory entry is the creation or last modification time. It is a binary coded 32 bit value. Its structure is as follows:

```
struct msdos_tod {
        unsigned    sec         : 5;
        unsigned    min         : 6;
        unsigned    hour        : 5;
        unsigned    day         : 5;
        unsigned    month       : 4;
        unsigned    year        : 7;
};
```

Since only 32 discrete values can be stored in the second field, the granularity of file creation time is in units of 2 seconds. The year field holds the number of years since 1980.

The starting cluster field in the directory holds the head of the FAT chain. If the cluster was 2 and we use the FAT chain shown in figure 1, the data blocks 2, 3, and 5 belong to the file.

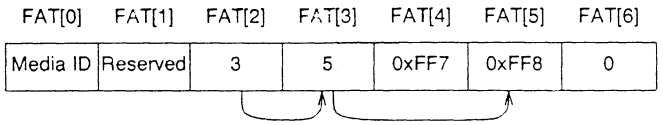


Figure 1

The file size is a 32 bit quantity. This may not reflect the true size of the file. First, MS-DOS directories do not store their size. Second, in an MS-DOS environment, the character '^Z' marks the end of text file. Since these text files contain an end of file marker, zero-length files do not exist.

4.2. MS-DOS Prototype Design and Implementation

While the structures associated with the MS-DOS file system are simple to understand, their layout causes problems. Since the initial target architecture for the MS-DOS file system was an Intel[7] 8088 microprocessor, there was no concern for quantities crossing four byte boundaries. For example, the create time in the directory structure is a 32 bit quantity and its address is 22 bytes into the directory. Likewise, the starting cluster and size cross long word boundaries. This caused every directory encode and decode routine to access all fields as characters and reconstruct them (fortunately, everything is stored in VAX order!).

Because the directory entry is only 32 bytes long, each directory is held within the file system specific part of a file's gnode. This provides a big performance gain when the file can be found in the gnode cache.

4.2.1. MS-DOS File Identification

Another piece of the file system architecture affecting the prototype design was the lack of a file identifier number. Because many UNIX tools require a somewhat unique "on-disk inode number", an algorithm was needed to create file IDs. Since an LRU gnode cache was being used, file IDs must be unique. Also these file IDs must be consistent for every instance of a file. Remember that a directory has two names and attribute structures.

The following algorithm was created for identifying files. If the file in question is not a directory, then the file ID is the starting cluster of the file's parent directory shifted left 10 bits or'd with the directory slot number. For example, if the starting cluster for the parent directory is 2 and the file was found in slot 33, the file ID would be 2081 (2 << 10 + 33).

This scheme works until it is necessary to update directory attributes. The file ID for a directory is simply its starting cluster shifted left 10 bits. To allow *msdos_gupdat* to update both incarnations of a directory, the old file ID (that is, the ID found by or'ing the directory slot number with the shifted block number of the parent) is stored in a file system specific area in the gnode. This modification of the algorithm allows the pathname `x` and `x/.` to produce the same file ID. The root directory causes still one more special case. Since '.' and '..' are not present in the root directory, the root file ID is 2. Note that the root file ID is unique because MS-DOS reserves the first disk block for the boot record,

4.2.2. MS-DOS Pathname Translation

Because the root directory contains no self referential files, it is necessary to special case *msdos_namei* to trap the names '.' and '..'. Also, when a directory is found, *msdos_namei* must store the old file ID and obtain a new set of attributes. This creates some ugliness in the *msdos_namei* code.

4.2.3. MS-DOS VBN to LBN Mapping

A file's blocks are mapped in a space allocation chain through FAT entries. To find the LBN associated with VBN 2, the LBN for VBN 1 needs to be found. Therefore, determining a logical block from a virtual block becomes linear. Each time the FAT is consulted, a 12 bit FAT entry is converted to a 32 bit quantity. This mapping consumes 12 VAX instructions (not including register setups). Likewise, mapping a 32 bit quantity to a 12 bit FAT entry uses 19 VAX instructions. Decoding a FAT chain is more expensive than translating an MS-DOS file name.

There are several methods for reducing the time needed to decode a FAT chain. Since decoding VBN 2 requires knowing the LBN associated with VBN 1, storing information about the last decoded VBN/LBN pair can reduce the FAT search time. This presupposes that files are read sequentially. When a file is being truncated, blocks are removed from the end of the file. Therefore, storing the last VBN/LBN pair can return to linear search time.

The entire FAT chain can be decoded when the directory entry is retrieved from disk. Since the directory entry is cached while inactive, decoding the FAT chain at name resolution may be an acceptable solution.

At the time this paper was written, the MS-DOS prototype stored the most recently used VBN/LBN pairs. Because only read and write performance was measured, this algorithm was sufficient for measuring expected file system performance.

4.2.4. MS-DOS Block Allocation

I was unable to locate a description of how MS-DOS systems allocate blocks to files. Examining files on an MS-DOS system indicated that there is no allocation policy. Blocks within a file were scattered throughout the device. Since many MS-DOS systems depend heavily on diskette drives, this scattering can cause dismal file system performance.

4.3. MS-DOS Performance

To assess read and write performance, a 4 megabyte file was created on the MS-DOS file system. This file was read and written using different block sizes. Measured performance follows:

I/O Performance (in kB/sec)			
	512b	1K	8K
Write	8.3	7.4	7.5
Read	26	26	26

[7] Intel is a registered trademark of Intel Corporation

Even though the *read* and *write* system calls could request I/O in units larger than the file system block size, *msdos_rwgp* breaks the I/O request down into file system block size pieces. Therefore, while the number of system calls was reduced by a factor of 16, performance is not expected to improve greatly since the kernel still posts the same number requests to the disk.

Write performance was poor because *msdos_bmap* must decode a 12 bit FAT entry when locating a free block in the FAT and when traversing the file's FAT chain. Better management of file block lists and free block lists should increase performance dramatically.

4.4. MS-DOS Support for NFS

The lack of an adequate file identifier caused the MS-DOS file system to be initially unusable with NFS. After solving the file ID problem, NFS coexists with MS-DOS.

A significant loss in performance was measured over NFS. Small block sizes may cause more packets to be transmitted between NFS server and client.

MS-DOS files cannot have holes. Currently, it is possible to create a UNIX file, seek well past the end of the file and write a block. This causes the file to have unallocated VBNs. Since MS-DOS file blocks are constructed from a FAT chain, no holes can exist within a file. Finally, NFS uses a file generation number to indicate whether a file has changed identity (by being removed). MS-DOS provides no method of storing this generation number.

4.5. MS-DOS Strengths

Since MS-DOS systems are widely used, the MS-DOS file system can be a good file transport between machines. Allowing such file systems to be read and written under UNIX systems permits MS-DOS machines to have indirect access to many of UNIX's facilities.

The MS-DOS file system also consumes little disk space for file system overhead. On most MS-DOS disks, 93% of the formatted media is available for data blocks.

4.6. MS-DOS Limitations

MS-DOS is primarily a file system intended for use on a single-user microcomputer. There are many significant limitations to the MS-DOS file system in a multiuser or networked environment.

The root directory is a fixed size. Even though a generous number of root entries exist, once the entries are consumed, no files can be created in the root directory. Also, disks no larger than 128 megabytes can be supported.

There are many serious limitations to MS-DOS's file attributes. Files may not have holes. Files may not be of zero length and can have an end of file character. There are no file identification numbers, nor is there any ownership information. File permissions are minimal. No hard or soft links or any special devices can be created or accessed in an MS-DOS file system. Finally, there is no method of storing a file generation number within the directory entry.

5. THE SYSTEM V PROTOTYPE

In one version or another, the System V file system is used by the V6, V7, System III, System V, XENIX[8], 4BSD, and 4.1BSD operating systems. More operating systems provide access to System V file systems than for any of the other file system prototypes.

System V's file system supports file system block sizes in units of 512 bytes, 1K, and 2K (only for A. T. & T.'s 3B5 line of computers). File names can contain up to 14 characters. All ASCII characters except '\0' and '/' may be used within file names. Also the file names '.' and '..' are reserved. Most implementations of the System V file system do not permit the high order bit set within each character.

[8] XENIX is a trademark of Microsoft Corporation

5.1. System V Architecture

A System V file system contains four components: a 512 byte boot block, a 512 byte super block, the on-disk inode table, and file system data blocks.

5.1.1. System V Super Block

The System V super block contains data describing the file system. The principal components are: the size of the on-disk inode table, the size of the file system, a count of free data blocks on the disk including a list of 50 free blocks, a count of free on-disk inodes including a list of 100 free inodes, and identifiers to determine the validity and block size of a file system.

5.1.2. System V Directory Description

System V directories are constructed from not necessarily contiguous blocks containing fixed length directory entries. Each directory entry is 16 bytes long and is defined as follows:

```
struct sysv_dir {
        u_short     sysv_ino;
        char        sysv_name[14];
};
```

The `sysv_ino` field contains a unique file identifier number (the inode number). Having this inode number allows for easy retrieval of the file attributes structure (the on-disk inode).

5.1.3. System V On-Disk Inodes

With the exception of the file name and the inode number, the on-disk inode contains all the data about the file. Each of these structures is 64 bytes long and is defined as follows:

```
struct sysv_inode {
        u_short     sysv_mode;
        short       sysv_nlink;
        u_short     sysv_uid;
        u_short     sysv_gid;
        int         sysv_size;
        u_char      sysv_addr[40];
        u_int       sysv_atime;
        u_int       sysv_mtime;
        u_int       sysv_ctime;
};
```

The `sysv_mode` field contains both protection information and a description of the file type. The protection stored is typical for a UNIX system, three types of protection (permit read, write, or execution) for three levels (owner, group member, or world). The remaining fields in `sysv_mode` describe the file type (for example, block or character special device), and execution attributes (for example, set user ID on execution).

The `sysv_nlink` field allows a number of file names to resolve to the same file. The `sysv_uid` and `sysv_gid` fields describe the owner. The `sysv_size` field gives the byte size of the file. `Sysv_atime`, `sysv_mtime`, and `sysv_ctime` hold the last file access time, the last file modification time, and the last inode change time.

The `sysv_addr` field provides a VBN to LBN mapping for the file. The 40 bytes hold thirteen three-byte disk block numbers. The first ten disk blocks numbers are the direct blocks of a file. The eleventh entry is the first indirect block. This indirect block contains a list of up to 128 blocks that are attached to the file. The twelfth entry is the second indirect block. The second indirect block contains a list of up to 128 blocks that are themselves indirect blocks. The last

entry is the third indirect block. It adds another level of indirect blocks.

5.2. System V Prototype Design and Implementation

The System V file system prototype was based loosely on the System V Version 2 Release 2 (V.2.2) source tape. During implementation, much code was taken and reused from the existing UFS code. The only source that was taken from V.2.2 was the block and inode allocation and deallocation code. Most of the path name translation (*sysv_namei*) and file I/O (for example, *sysv_rwgp*) code was taken directly from the UFS prototype.

5.2.1. System V Block and Inode Allocation

Much like the block map within the sysv_inode, free blocks and inodes are stored in an array. The last entry in the free block (and inode) array points to a block containing 50 more indices to free blocks. Likewise, the last entry in this block points to a block containing 50 more free block indices.

The System V mkfs command understands disk geometries and constructs the free list using an optimal block layout.

Unfortunately, allocating and deallocating blocks occurs frequently. This causes the list to lose its optimal ordering. Since the allocation code simply removes the next block off the list, a file's data blocks can become scattered over the file system.

5.2.2. System V Prototype Functionality

The System V operating system does not support all of the file system related system calls found in 4.3BSD. The GFS System V prototype however, provides code for making and removing directories, renaming a file, truncating a file to a specific, potentially non-zero length, and insuring all cached data blocks are flushed back to disk. With the exception of symbolic links, all file system functionality found in 4.3BSD was implemented in the System V prototype.[9]

5.3. System V Performance

Read and write performance tests were done to a newly created System V file system. This file system was created with file system block size of 1K. Since the free block array was optimally ordered, disk head movement was minimal. This provided for "best case" I/O measurements. Measured performance follows:

I/O Performance (in kB/sec)			
	512b	1K	8K
Write	47	47	48
Read	35	37	38

As with the MS-DOS file system, the file system bandwidth is limited by the small transfer size.

5.4. System V Support for NFS

The on-disk inode for the System V file system contains no space to hold a file generation number. Therefore removing a file and reusing its on-disk inode can cause inconsistencies in NFS. Symbolic links are not supported. Attempting this functionality fails in an NFS environment. Finally, the small file system block size causes a degradation in NFS I/O performance.

[9] There is space in the `sysv_mode` field of the on-disk inode to identify a symbolic link. This creates a file system that is not transportable to a System V operating system.

5.5. System V Strengths

The System V file system is easily understood. Users not familiar with file system implementations can fix corruption with a high probability of success. The file system also supports most of the 4.3BSD file system semantics.

5.6. System V Limitations

The System V file system uses a small file system block size. The small block size increases the number of transactions a device must handle and reduces its effective throughput. Experience has shown that larger machines demand considerable I/O bandwidth. I believe that the current System V file system will be inadequate on a large machine. If the file system block size was to increase (as in the file system for the 3B5), disk space would be wasted on partially filled data blocks.

The functionality for symbolic links cannot be supported without causing problems when moving the file system to a System V machine. NFS encounters difficulties since file generation numbers cannot be stored. Finally, the `sysv_size` field in the System V inode limits the file size to 2^{31} bytes.

6. THE UFS PROTOTYPE

UFS, or the fast file system from 4.3BSD (McKus1983a), evolved from the System V file system. Major improvements were made to increase reliability and performance.

UFS supports file system block sizes of 4K and 8K. These blocks may optionally be broken in 2, 4, or 8 pieces (fragments). File names can contain up to 255 characters. UFS permits the same character set within names as does the System V file system (everything but the names '.' and '..' and the characters '\0' and '/' are allowed).

6.1. UFS Architecture

A UFS disk contains two components: an 8K boot block, and many cylinder groups. These cylinder groups permit blocks to be allocated while attempting to minimize disk head movement.

6.1.1. UFS Cylinder Groups

A cylinder group consists of five components: a super block (or copy thereof), a cylinder group structure, some on-disk inodes, a cylinder group summary structure, and some data blocks. These cylinder groups are spread over the entire file system. This differs from the other prototype file systems in that UFS scatters file system data structures across the surface in an attempt to minimize disk head movement.

The UFS super block contains a description of the file system and the media. Data stored there includes the geometry of the disk, the file system block and fragment size, the size of a cylinder group, and file system configuration parameters.

A cylinder group structure contains the size of the cylinder group, the location of last used on-disk inodes and blocks, and a used on-disk inode map. Also included in this structure is a free disk block map for the cylinder group. The cylinder group summary structure holds the summary of available resources within the group.

6.1.2. UFS Directory Description

UFS directories are constructed from not necessarily contiguous blocks containing variable length directory entries. Each directory entry is potentially 264 bytes long and is defined as follows:

```
struct ufs_dir {
        u_long      ufs_ino;
        u_short     ufs_reclen;
        u_short     ufs_namelen;
        char        ufs_name[256];
};
```

`Ufs_ino` uniquely identifies a file as does `sysv_ino` in the System V on-disk inode. The `ufs_reclen` field describes the length of the UFS directory entry. The `ufs_namelen` field contains the length of the file name. The name length is always in multiples of 4 preventing memory addressing problems. Each of these entries are wholly contained within a disk block and are stored as compactly as possible within a directory.

6.1.3. UFS On-Disk Inodes

As with the System V file system, all information other than the file name and file identifier (on-disk inode number) is held in the inode. The UFS inode structure is as follows:

```
struct ufs_inode {
        u_short     ufs_mode;
        short       ufs_nlink;
        short       ufs_uid;
        short       ufs_gid;
        quad        ufs_size;
        timeval     ufs_atime;
        timeval     ufs_mtime;
        timeval     ufs_ctime;
        long        ufs_db[12];
        long        ufs_ib[3];
        long        ufs_blocks;
        long        ufs_gennum;
};
```

Differing from the System V file system is `ufs_size`, `ufs_atime`, `ufs_mtime`, and `ufs_ctime` which are 64 bits long. These times are in microsecond resolution (depending on the resolution of the machine's clock). Also `ufs_blocks`, the number of blocks allocated to the file, and `ufs_gennum`, or file generation number, have been added to the basic System V on-disk inode.

6.2. UFS Prototype Design and Implementation

The UFS prototype is based strictly on ULTRIX 2.0 UFS code. In fact, it has been the basis for much of the MS-DOS, ODS-II, and System V prototype file system code.

6.2.1. UFS Block and Inode Allocation

The UFS disk block allocation policy attempts to allocate all data blocks for a file in the same cylinder group. Further, the policy attempts to position each block in a rotationally optimal position relative to the position of the previous block in the file.

If the optimal block has been previously allocated, the allocation code first attempts to allocate a block within the same cylinder group. If there are no free blocks in the cylinder group, the code quadratically searches other cylinder groups. If an available block still has not been located, a brute force search is conducted.

For a file, the inode allocation strategy attempts to place the file within the same cylinder group as the parent. When allocating an inode for a directory, the allocation is done from the cylinder group that has the fewest allocated inodes.

6.3. UFS Performance

As expected, the increase in file system block size and the block allocation policy allows for an increased I/O bandwidth. Read and write performance for an 8K block size, 1K fragment size follows:

I/O Performance (in kB/sec)			
	512b	1K	8K
Write	85	148	176
Read	118	154	206

6.4. UFS Support for NFS

As distributed by UCB, the 4.3 UFS on-disk inode does not hold a file generation number. Since unallocated space exists within each on-disk inode, a generation number is held. This is the only functionality that was needed for support of NFS.

6.5. UFS Strengths

Having a large file system block size reduces the number of transactions disks must service. Since UFS obtains data about disk geometry, the placement of newly allocated blocks and inodes is close to optimal. These improvements markedly increased file system throughput.

By allowing large disk blocks broken into fragments, disk space is also put to good use. Finally, by definition, the file system does an excellent job of supporting UNIX file system semantics.

6.6. UFS Limitations

The complicated allocation scheme can become a burden for slower CPUs. Also, corruption cannot be corrected as easily as in the System V or MS-DOS file systems.

7. LIMITATIONS OF THE GFS INTERFACE AND THE UNIX SYSTEM

After prototyping the ODS-II, MS-DOS, System V, and UFS file systems, the limitations of GFS and the UNIX system are better understood.

The file system buffers in GFS can be no larger than 8K without restructing the buffer allocation strategy. Unfortunately, the change is not simply increasing this 8K limitation. System page table sizes should be adjusted, and the buffer allocation code should be changed so as to not limit the buffer size.

The UNIX system call interface provides no method to instruct file system implementations to alter their block allocation strategy. ODS-II permits contiguous files, but the system call interface provides no mechanism for instructing the implementation to do so.

UNIX file protections are an issue. At one end of the spectrum, MS-DOS has limited protection and the file system becomes unusable on a non-friendly machine. At the other end of the spectrum, ODS-II has four permission modes that can function on four different levels. While most of these permissions can be handled, ODS-II's ACLs still cannot be addressed.

Finally, many parameters to file system related system calls assume a 32 bit quantity but files in both UFS and ODS-II can be larger than 2^{32}. The system calls *lseek*, *read*, *write*, and *truncate* all take a 32 bit parameter specifying a length or offset.

8. CONCLUSION

We have learned many lessons from prototyping the ODS-II, MS-DOS, System V, and UFS file systems. The GFS interface is better understood; any needed functionality has been added. We know how NFS functions when serving each of the file systems and have measured basic file system performance using several different file system architectures.

Much work still needs to be done to UNIX and GFS. The UNIX system calls and GFS interfaces using a 32 bit offset or size must be changed to support very large files. Structures holding block indices must be changed to support very large media. Support must also be added for write once media. Finally, the buffer caching code needs to be restructured.

While the work done for this paper is strictly research, the information presented should help file system performance in the future.

9. ACKNOWLEDGEMENTS

Bob Rodriguez deserves special mention. He was the author of the ODS-II prototype that was the basis for the ODS-II/GFS file system.

My management, Steve Reilly, Fred Glover, Dave Cardos, Kent Ferson, Glenn Johnson, Bill Heffner, Jack Smith, and Ken Olsen have graciously allowed me enough time to design and implement the file systems.

Discussions on the GFS interface with Mike Karels and Kirk McKusick from the University of California at Berkeley were timely. Jim McGinness, Dave Roberts, Ken Reilly, Ricky and Larry Palmer, Chet Juszczak, and Jeff Chase answered questions and offered much needed insight.

Finally, I thank the cast of people that critiqued this document.

10. REFERENCES

Readers will find a more complete reference to specifics on file systems and their implementations in the following works:

References

Rodri1986a.
R. Rodriguez, M. Koehler, R. Hyde, "The Generic File System," *USENIX Conference Proceedings*, pp. 260-269, Summer, 1986.

McKus1983a.
M. K. McKusick, W. Joy, S. Leffler, R. Fabry, "A Fast File System for UNIX," *CSRG Technical Report*, vol. 83, no. 147, 1983.

Golds1983a.
A. Goldstein, Files-11 On-Disk Structure Specification, 1 Oct, 1983.

Norto1985a.
P. Norton, *The Peter Norton Guide to Programming the IBM PC*, pp. 99-125, Microsoft Press, 1985.

Ritch1978a.
D. Ritchie, K. Thompson, "The UNIX Time-Sharing System," *The Bell System Technical Journal*, vol. 57, no. 6, pp. 1905-1929, July - August, 1978.

Sandb1985a.
R. Sandberg, D. Goldberg, S. Kleiman, D. Walsh, B. Lyon, "Design and Implementation of the Sun Network Filesystem," *USENIX Conference Proceedings*, pp. 119-130, Summer, 1985.

Thomp1978a.
K. Thompson, "UNIX Implementation," *The Bell System Technical Journal*, vol. 57, no. 6, pp. 1931-1946, July - August, 1978.

Analyzing System Crashes

by

Al Delorey

Ultrix Engineering Group

decvax!afd

smurf::delorey

Some Common Crash Types

Hardware Trap

The system will push onto the interrupt stack the:
PSL, PC, Code, Trap type

Depending on the Trap Type, The code is often the last virtual address that was accessed (which caused the trap). See /sys/vax/trap.h for an explanation of the trap types.

The Ultrix trap routine (/sys/vax/trap.c) is called through the SCB. Trap in turn calls panic.

An example of a trap, is a process that accesses an address that is outside of the process' address space, which causes a trap type 8, segmentation fault.

Hardware Machine Check

The system will push onto the interrupt stack a machine check frame (processor dependant).

The Ultrix machine check routine (/sys/vax/machdep.c, ka???.c) is called through the SCB. If unrecoverable, machine check in turn calls panic.

An example of a machine check, is a parity memory error.

Software Panic

Kernel software detects an internal inconsistency.
The system would be running on the kernel stack.
The kernel routine that detects the inconsistency calls the panic routine.

See the VAX Architecture Handbook or Reference manual for more info.

Using nm

For a system crash that gives a PC on the console, you can use nm(1) to determine what routine was executing.

```
nm -n /vmunix
```

This command will display the name list (symbol table), in numerical order, of the vmunix image. Find the address that is closest to (but less than) the given PC from the crash. That address is the starting address of the routine that was executing.

You can then subtract the start address of the routine from the faulting PC, to determine the offset from the beginning of the routine where the error occured. Then using ADB the offending instruction can be found.

Getting a Crash Dump

The Interrupt Stack is in Kernel Address Space.
It has a fixed size of 3 pages of memory (3x512 bytes).
Its starts just below the address of the dump routine ("doadump"), and grows downward in memory (600 hex locations).

If the Interrupt stack is invalid the crash dump routine in the kernel will not be called.

The crash dump routine can be started manually.
The second long word of the RPB (restart parameter block, see /sys/vax/rpb.h) contains the address of the crash dump routine.
This address can be obtained from console mode by typing "E/P/L 4".
The crash dump code can then be run by typing "S address"

Forcing a Crash

If the system is "hung" you can force a crash dump to be written.
Halt the processor and enter console mode.
Find the address of the dump routine and start the routine manually.

```
>>> E/P/L 4                     - Get address of dump
  P 00000004  00000C00          - Console shows the address
>>> E/G F                       - Exam gen reg F (PC)
  G 0000000F  80001EAD
>>> E PSL                       - Exam the PSL
  M 00000000  04C10004

>>> D PSL 041F0000              - Set to Interrupt Stack
                                  and Interrupt Priority 31

>>> S 80000C00                  - Start execution of dump
```

Another way to force a dump is to halt the processor, then from
console mode set the PC to a -1 and continue. This will force
a segmentation fault and a dump will then be made.
Before continuing, examine the PC and stack pointers since
these will be modified by your actions.

Advantage: Disks are "sync"ed.
Disdvantage: Some machine state is changed.

```
>>> E/G F                       - Exam gen reg F (PC)
  G 0000000F  80001EAD
>>> E PSL                       - Exam the PSL
  M 00000000  04C10004
>>> E/I 0                       - Exam Internal Reg 0 (KSP)
  I 00000000  7FFFFDAC
>>> E/I 3                       - Exam Internal Reg 3 (USP)
  I 00000003  7FFFE2F4
>>> E/I 4                       - Exam Internal Reg 4 (ISP)
  I 00000004  80000C00
>>> D/G F FFFFFFFF              - Deposit -1 in PC
>>> D PSL 001F0000              - Set IPL at 31 to block
                                  interrupts
>>> C                           - Continue the processor
```

If neither of the prior methods work, you may still be able to get a dump.

Initialize the processor before starting the dump routine.
This sets the processor to a known state, which includes setting
the PSL to run on the interrupt stack and sets the IPL to 31,
also memory mapping is disabled.

Depending on the processor, the initialize may corrupt the
ISP, KSP, P0BR, P0LR, P1BR, P1LR.

Advantage: Should work
Disdvantage: More machine state is changed.

```
>>> E/P/L 4                     - Get address of dump
  P 00000004  00000C00          - Console shows the address
>>> I                           - Init the processor
>>> S C00                       - Start execution of dump
```


Some useful ADB commands

```
adb -k vmunix.1 vmcore.1    - invoke adb on crash image
adb -k -w /vmunix /dev/mem  - invoke adb on running system w/ write

?[n]                        - values in disk image are printed.
/[n]                        - values in core file (or mem) are printed.
=                           - virtual address of a symbol is printed.

Formats:                    - Follow the above 3 actions (hex dflt)
    x                         - print short word in hex fmt
    X                         - print long word in hex fmt
    o                         - print short word in octal fmt
    O                         - print long word in octal fmt
    d                         - print short word in decimal fmt
    D                         - print long word in decimal fmt
    s                         - print string which starts at
                                given address

*(scb-4)$c                  - gives stack trace of whichever stack was
                              currently active (interrupt or kernel).
                              Format:
                                funct_3 (args) from addr_3    <= newest
                                funct_2 (args) from addr_2
                                funct_1 (args) from addr_1    <= oldest

                                funct_1 calls funct_2 from addr_1 in
                                funct_1 therefore, the stack frame w/
                                saved pc of addr_1 (ret addr), is the
                                stack frame of funct_2

<addr>$c                    - gives stack trace from stack addr given.

<routine name>+2[/?]i       - print assembly instruct's starting at
                              the beginning of the given routine.
                              "+2" is to skip over the reg save mask.

<addr>[/?]i                 - print assembly instruct's starting at
                              the given address.

<ret>                       - look at next location to the last one

^                           - look at previous location to the last one

[/?]w <value>               - modify last addressed location

$R                          - show register contents

<range>$S                   - extend range of symbolic names (1000$S)
```

ADB SCRIPTS

The directory /usr/lib/adb/ contains adb scripts which format Kernel Data structures.

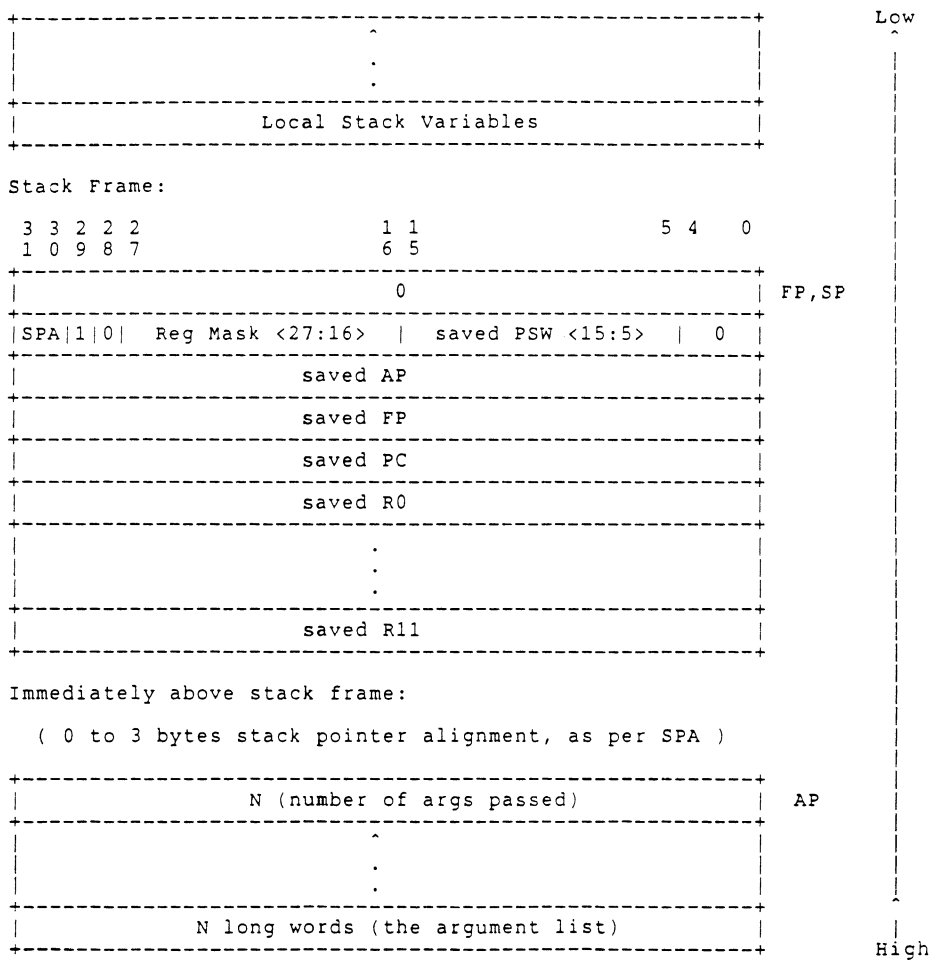
addr$<script-name	- apply the named script at the given addr
u$<u	- apply the user structure script at symbolic addr "u" (the current u block, ie. user structure of current process)
addr$<proc	- apply the proc script at the address obtained from the user structure.
addr$<pcb	- apply the pcb script at the given addr

Examining stack frames with ADB

Usefull for seeing values of local variables

(scb-4)/X	- scb-4 contains the address of the current "in-use" stack - 800nnnnn: system was using interrupt stack 7ffnnnnn: system was using kernel stack
intstack/20X	- starting at addr of intstack print 20 long words in hex format.
u$<u	- first item in the user structure is the Kernel Stack Ptr (KSP).
<ksp>/20X	- starting at addr of kernel stack (KSP) print 20 long words in hex format.

To find a stack frame (call frame for a proc call) look for a 0 long word (condition handler) followed by a long word with bit 29 set, which indicates a "calls" (eg: 2e000000).



The "calls" instruction pushes the argument count onto the stack then aligns the stack and creates the stack frame (call frame) which is the saved register through the condition handler.

(For more information, see the VAX-11 Architecture Ref Man pages 4-71 to 4-75)

Getting Stack Trace of any Process

```
ps -axlk vmunix.n vmcore.n  - Flags (see ps(1))
                              -a All processes (not just your own)
                              -x Even processes w/ no tty
                              -l Long format (more info given)
                              -k Kernel files given
                            - get <pid> of process that was running
                              or any process that you want to look at.

pstat -pak  vmunix.n vmcore.n
                            - Flags (see pstat(8))
                                -p Prints process table for active
                                   processes
                                -a Describe all process slots
                                -k Kernel files given
                            - Look for pid you want, and get the
                              location (that's the memory location
                              of the proc struct).  Its the 1st
                              field and should start like 800nnnnn.
                            - Check the process' state and flag codes.
                              (2nd & 3rd fields, labeled S & F).
                              Definitions of fields are in pstat(8)

adb -k vmunix.n vmcore.n
<proc-loc>/X                - Contents of first field in proc struct
<RET>
<RET>
<RET>
<RET>                       - Contents of 5th field in proc struct
                              (<addr of pte> that maps the u area.
                              See proc.h: proc struct, p_addr field).

<addr of pte>$p             - Set process context for adb
$c                          - Stack Trace of process in question
```


References For Further Info

"A Tutorial Introduction to ADB"
by J.F. Maranzano & S.R. Bourne
See ULTRIX-32 Doc set, Supplementary Documents Volume II: Programmer
Page 3-51

"Using ADB to Debug the UNIX Kernel"
by Sam Leffler & Bill Joy
Not in ULTRIX doc set (unfortunately)
Can be found online in /usr/doc/kdebug

Header files:
/sys/h/
proc.h
user.h

/sys/vax/
pcb.h
trap.h

Crash(1M)
Sys V "crash" program that is only partially converted to
understand the ULTRIX kernel data structures.

NEWSLETTER OF THE VAX SYSTEMS SIG

Pageswapper

Our Mascot

VAX SYSTEMS SIG
DECUS U.S. CHAPTER

TABLE OF CONTENTS

To register for on-line submission to the Pageswapper dial:

(617) 262-6830

(in the United States) using a 1200 baud modem and log in with the username PAGESWAPPER.

Articles for publication in the Pageswapper can be sent (US mail only -- no "express" services please) to:

Larry Kilgallen, PAGESWAPPER Editor
Box 81, MIT Station
Cambridge, MA 02139-0901
USA

Preference is given to material submitted as machine-readable text (best is Runoff source). Line length should not exceed 64 characters and the number of text lines per page should not exceed 48 (these limits are particularly important for sample commands, etc. where simple text justification will not produce a meaningful result).

Please do not submit program source, as that is better distributed on the VAX SIG tape.

Please do not submit "slides" from DECUS Symposia presentations (or other meetings) as they are generally a very incomplete treatment for those readers of the Pageswapper who are not so fortunate as to be able to travel to Symposia. Please DO write articles based on such slides to get the content across to a wider audience than is able to attend.

Change of address, reports of non-receipt, and other circulation correspondence should be sent to:

DECUS U.S. Chapter
Attention: Publications Department
249 Northboro Road (BP02)
Marlborough, MA 01752
USA

Only if discrepancies of the mailing system are reported can they be analyzed and corrected.

Editor's Workfile

(Possible) Future Features in VMS

The article entitled "VMS Futures" in this issue is taken from slides provided by Trevor Kempsell of DEC which he used in his presentation by that title at the 1987 US Spring Symposium in Nashville. There is the by now familiar sort of disclaimer at the beginning indicating that what you see very well may not be what you get.

This brings to mind a particularly insightful comment made by a user at the DEC SIR response session in Nashville. After some others had indicated they really needed the ability to checkpoint jobs, and complained that it had been "promised" a long time ago, this thoughtful user reminded us all that there was no "promise" at all. When DEC breaks its normal silence to tell us what they are working on, experimenting with, or considering, it is only with the understanding that nothing is guaranteed. If users try later to claim that a particular feature was "promised" for the future, it will only make it even less frequent that DEC will share with us what they are considering for future software releases!

SIR Voting Again

This issue of the Pageswapper contains the Fall 1987 SIR Ballot. Thanks to Mark Oakley for pulling all the new choices together, so close on the heels of tabulating the previous list. The Ballot form will be repeated in the back of the newsletter for the next couple of months, but the text of the SIR's will only be printed in THIS issue. Likewise, those with accounts to do on-line Pageswapper submissions will again be able to cast their votes electronically, but they will need a copy of this issue in hand since the text is not presented on line (holding time on calls would strain our resources if voters were still contemplating while logged in).

VMS Notes from Nashville

Coming next month, to a Pageswapper near you. Even beyond the goal of limiting this month's page count to maintain cordial relationships with DECUS staff and my fellow editors, there is the matter of information overload. With VMS Futures, SIR Responses and a new SIR Ballot this month, I felt I better delay something for the August issue.

Internals and Data Structures Manual

Ray Kaplan
PIVOTAL, Inc.

May 3, 1987

I have recently had the pleasure of working with Mike Mehan of Digital Press to get the part of the VAX/VMS Internals and Data Structures Manual that was done out to folks. He brought 1500 copies of what was done to the Symposia in Nashville. Most of them were sold, and you can be sure that the ones that went back with them will be quickly snapped up by internal DEC folks.

So, where to from here? I suggest that you sign the petition provided in the questionnaire section at the back and mail this to me so I can send it on to Mike to help him tell the people at Digital Press that we want to see these publications as soon after the release of the operating system as possible.

There is a school of thought that suggests that if you don't do it, it just won't get done. Copy the petition. Pass it around. Who knows, we just might have a V5 IDSM when we need it!

Do it today. Tomorrow might be too late.

See you!

Ray Kaplan
PIVOTAL, Inc.

VMS Futures

Disclaimer

The purpose of this presentation is to let customers know some features that VMS and Language engineering groups are examining for future releases.

This is NOT an announcement or commitment by Digital that any of the items discussed will appear in any Digital product.

DECnet-VAX

- Enhanced support for network proxies
- Improvements to the routing layer
- Performance improvements in MOM
- Wildcard and multiple command recall support in NCP

VAX/VMS kit changes

- MicroVMS and VMS become one kit
  - Tailoring classifications
  - Kit has three save sets similar to VAX/VMS
  - All kits have useful MicroVMS files
  - All VAX/VMS files will be in the kit
- New tailoring mechanism

- Design center is tailor-off
- Tailor-on is supported, but not optimal

VAX/VMS File System Project

o Files-11 ODS-2 file system
  - Re-worked file highwater marking
  - Alias file entry to include primary and secondary entries

BACKUP

o Prevention of accidental initialization of input disk
o Lock-down of working set for standalone BACKUP
o BACKUP performs its own $MOUNT requests
o New qualifier /IGNORE=(TAPE_LABELS) requires VOLPRO
o Perform more than one operation per standalone boot

ANALYZE/DISK_STRUCTURE (Formerly VERIFY)

o Choice of destination for lost files
o Correct handling of file aliases and directory backlinks

MOUNT

o New qualifier /MULTI_VOLUME

Terminal Fallback Facility (TFF)

o What is TFF
  - TFF translates characters that are written into something the terminal can display.
  - TFF will translate input characters into their equivalent DEC Multi National Character set character
  - for the international market place

o Includes a management utility
  - defines which table is to be used.
  - enable or disable TFF for a terminal or terminals.
  - shows information about what tables are available, and what is being used

CLUSTER

o Improve failover on disks and tapes
  - support failover on DSA disks (UDA, KDA, BDA) to provide more flexibility in high-availability
  - support dynamic failover and mount verification on HSC tapes

o volume shadowing
  - V4.6 support for more members per shadow set

- o LAVc support integrated into VMS
  - as of V4.6, no more separate version for LAVc
  - LAVc key still required and separately licensed
- o Extensions to LAVc
  - support CI and Ethernet nodes in the same cluster, boot nodes can connect to HSC disks
  - support more satellites
  - volume shadowing is only available on HSC disks
  - HSC failover, if disks are dual-pathed
  - server failover, if multiple boot nodes exist
  - Current and future workstation software will be supported

RMS

- o RMS Journaling Merged
  - Key to enable
- o Collated Key Support
  - User-defined collating sequences (multinational keys)
  - Support through $XABKEY extensions
- o Indexed File Sequential $GET speedup
- o RMS Execution Monitoring

  - SET FILE /STATISTICS to enable
  - MONITOR RMS to watch
- o Block-or-Record Access Emulation for RMS-11 Partners
  - COPY to RMS-11 systems now uses block mode
- o $XABITM

  General item-list $XAB Supports:
  - RMS statistics monitoring
  - DAP link parameters
  - DAP extended protection fields

AUTOGEN

- o AUTOGEN feedback
  - Allows user workload to size the system
  - Affects most memory and page/swap-file-related parameters
  - VMS continuously maintains usage information and AUTOGEN grabs snapshot
  - Valid only after "typical workload" has executed

DEBUG

- o Dynamically reformatting REG display.

- o EXAMINE/OPERANDS
- o MACRO support enhancements
- o New predefined windows for screen mode
- o SET MODULE/CALLS
- o SET MODE SEPARATE
- o SET PROMPT/[NO]POP
- o Spawn /INPUT and /OUTPUT
- o INCOMPATIBLE CHANGES
  - EVALUATE command now displays value not address for Macro
  - Screen mode line wrapping in OUT display now at 255 characters
  - DEBUG separate window control (SET MODE [NO]SEPARATE; SET PROMPT/[NO]POP)
- o PROBLEMS CORRECTED/RESTRICTIONS REMOVED
  - SET IMAGE image,image,...

    now sets all the images
  - SET SCOPE Command will automatically set modules for you

BATCH/PRINT FACILITY

- o Significantly improve performance through selective restructuring of system queue file to reduce I/O
  - Replace master pending job list with queue-specific lists

  - Create job entry vector
  - Improve print job scheduling algorithm
- o Support access control lists (ACLs) on queues
- o Add features and flexibility to DCL interface
  - For SHOW QUEUE provide better selection criteria
  - New SHOW ENTRY command
  - New F$GETQUI lexical function
- o Enhance corresponding programming interface
  - Add $SNDJBC item codes
  - Add $GETQUI item/function codes
  - New LIB$GETQUI library routine

New RTL DATE/TIME features, LIB$ facility.

- o Specify output formats other than the standard VMS format
- o Specify input formats other than the standard VMS format
- o Specify input and output formats using languages other than English
- o Selection of language and formats through logical names or through routine calls
- o Multiply delta times

- Convert a VMS internal time to either an integer or a floating point value, based upon a selected unit of time (seconds, minutes, etc.)
- Add and subtract VMS internal times
- Convert an integer or floating point value to a VMS internal time, based upon a selected unit of time (seconds, minutes, etc.)
- Convert a $NUMTIM 7-word array to a VMS internal time

Screen Management

- SMG$ Viewports
  - Portion of a virtual display visible on pasteboard
  - Provides portal onto virtual display that may be moved or resized.
- SMG$ Subprocess Support
  - Method to control subprocesses and I/O via virtual displays
  - A way to have several concurrent subprocesses, each with own display
- SMG$ Menus
  - General support for many usage modes

    Block menus (VT2xx SETUP)

    Vertical menus (aka pull-down)

    Horizontal menus (aka strip menus)

  - Renditions

    Selected items, Unselected items

  - Options

    Default items, One-use-only items

- SMG$ User Renditions
- ANSI has many renditions not on DEC terminals: color text, double underline reduced intensity, rapid blink
- Method provided in SMG for up to eight programmer defined renditions
- Access via rendition-set parameter to output routines

LAT

- Support for access to two separate Ethernet LANs
- $GETDVI returns terminal server and port name for LTAn:
- Support for TT$M_BREAK, generates break at terminal server port (DECserver 200)

LAT/VMS V4.6 - Possibilities

- Full function LTDRIVER, LATCP, and LATSYM now distributed with VMS
- LTDRIVER port QIOs
  - Solicit Connection to Application Device
  - Get server and port name for LTAn:

- Multi-threaded print symbiont (LATSYM)
  - Up to 32 print streams per process

DCL/EDITORS/UTILITIES

- DCL
  - IF-THEN-ELSE construct
  - RECALL/ERASE (erases the recall buffer)
- TPU
  - avoid section file rebuilds
  - WPS keypad
  - /START_POSITION qualifier
  - improved word wrapping
- Utilities
  - Callable mail

Rolling Cluster Upgrade

- Mixed version cluster
  - only between adjacent releases
  - not all new features available
  - not optimum performance
  - not a permanent state

- Continuous operation
  - possible with sufficient redundant resources

VMS SYSTEM MANAGEMENT

- The breadth of VAX/VMS configurations upward and downward has strained the capabilities of the original VMS system management design.
- VMS Development is defining a new internal architecture for system management. This architecture will provide a modular base for all system management efforts over the next several releases of the product.
- The initial implementation is intended to provide a cluster-wide view for managing VAXclusters. This implementation provides a standard DCL-command line interface.

Digital Responds to the Spring 1987 SIR Vote

Mark Oakley
SIR Coordinator
and
Nigel Turner
VMS Engineering

At the Nashville Symposium, Digital responded to the VAX SIG's most recent Software Improvement Request ballot. The ballot was originally published in the January issue of the Pageswapper and the results can be found in the May issue. 364 ballots were returned, a decrease in participation from the previous ballot. Digital remains very committed to the SIR program. Below is a summary of the top 10 items followed by Digital's response, as presented by Nigel Turner, Jake Vannoy, Keith Walls, Jim Krycka, and Bill Robinson.

SIR: S87-26 Position: 1 Points: 215

Abstract: Allow command line editing to work on the entire command.

Description: Currently, command line editing is limited to the last line displayed on the scope. Frequently, errors are found in the command prior to the last displayable line, especially on long commands. This can be worked around in some cases by setting the line length to 132 or deleting characters until the error is reached. Both of these work-arounds are cumbersome and inelegant. Some change is needed to the command line editor, to allow the full command to be edited. One possibility would be to provide a mode to return to the beginning of the command and step forward a line at a time through the command.

Response: This is the most frequently requested terminal driver enhancement and is a very desirable feature. We have considered adding this capability. However, in order to avoid significant degradation of apparent response time, a redesign of the terminal driver is required. There are no current VMS plans for including this feature.

PAGESWAPPER - July 1987 - Volume 8 Number 12
Digital Responds to the Spring 1987 SIR Vote

SIR: S87-28 Position: 2 Points: 158

Abstract: Provide a mechanism to require a $SET PASSWORD to be automatically invoked on expiration of a user's password.

Description: Sites that enforce password expiration and have users locked into captive command procedures to invoke applications such as ALL-IN-1 have a problem when the screen is cleared automatically and the user misses the critical message that the last login permitted has just been made! This results in very frustrated users and system managers. A simple fix to this and many other instances would be a /DEMAND_NEW switch in the authorize file. This switch if set yes would force a $SET PASSWORD to take place during login in lieu of the warning message that the last login had been used.

Response: We believe this is one member of a class of problems concerned with account management. We are currently designing ways in which these problems can be solved with a single style of solution. We therefore do not intend to provide this particular "point" solution but, rather, we intend to solve it by better designing the system management interface.

SIR: S87-27 Position: 3 Points: 156

Abstract: Provide remote printing of a LOCAL file.

Description: Given the power of DECNET it should be possible to provide the queue manager on a remote node with a print command that contained a node name in the file spec. The print symbiont would then utilize the File Access Listener on the source node to read the file in and send it across the network. For example, a user on node IBIS:: would issue the following command:

```
$PRINT/NODE=HAWK/QUEUE=CP01 IBIS::COLOR_TEST.LIS
```

This would result in a print entry placed in the queue on node HAWK that pointed to COLOR_TEST.LIS on node IBIS. Once the linkage is provided for the queue manager to communicate over DECNET many other possibilities quickly become easy. For example:

```
$PRINT/NODE=HAWK/QUEUE=CP01 JAY::CARTOON.TEST
```

Response: We agree that the ability to submit and control remote print jobs through an extension to the current DCL interface would be very convenient and useful. PRINT/REMOTE is clearly a very limited capability that does not copy files, nor does it allow other job-related qualifiers to be passed to the remote system. It is implemented by opening and closing the selected remote file through RMS calls using the "spool on close" option to enter a remote print job. VMS Engineering has seriously considered enhancing the DCL interface as you suggest. However, the current design of the Batch/Print subsystem does not easily lend itself to solving many of the problems faced in implementing a transparent remote printing capability. Remote printing entails much more than a simple extension to the VMS distributed printing capability found in clusters today where there is a shared file system, distributed lock manager, and a single security domain to rely upon. Digital recognizes the need to continue to develop both the distributed and remote printing capabilities of VMS (as well as batch). To this end, we plan to focus on evolving the current print and batch architecture to address current and future requirements in this area. Thus, while future directions of the VMS Batch/Print facility are being assessed, we have no immediate plans to extend the current DCL interface to provide a limited solution to this remote printing problem.

SIR: S87-20 Position: 4 Points: 145

Abstract: Provide a separate file "date last read" from "expiration date".

Description: VMS provides the ability to maintain a pseudo "date of last access" for files by using a volume-wide file retention period to update an expiration date. It would be desirable to have the ability to maintain the date that a file was last read, as well as maintain an explicit expiration date for a file. Knowing for certain the date and time a file was last read can be an important security tool. The date the file was last read should be separate from the date the file was last created and the date the file was last modified.

Response: We have considered this request many times and rejected it on the basis of the certain performance implications. However, we are now seriously considering how we might do this without adversely impacting performance. Most likely, last access date recording will not be the default but will be enabled at some level (volume, directory, file).

SIR: S87-77 Position: 5 Points: 133

Abstract: Enhance AUTHORIZE to report on individual fields in UAF records.

Description: Currently, AUTHORIZE provides only "brief" or "full" reports. It would be quite useful if fields could be included and/or omitted. For example:

```
UAF> SHOW USER/PRIVILEGE
```

would display just the privileges for the account "USER".

Response: In version 4.4 of VMS, we introduced the SYS$GETUAI and the SYS$SETUAI system services. This is the foundation upon which future system management tools will be built in this area. While we have made no decisions about the nature of the user interface, we are giving every consideration to the ease with which VAX/VMS systems can be managed in the future.

SIR: S87-71 Position: 6 Points: 130

Abstract: Support BACKUP over DECnet.

Description: It would be very useful if files could be backed up to a tape drive that was on another node in the network. The implementation should be concise, and not require the creation of DCL procedures, network objects, etc.

Response: We recognize the need for BACKUP to have network capabilities beyond being able to create and read savesets. We plan to provide BACKUP with full access to remote devices, files, and savesets in a future release of VMS. One of the requirements we recognize is that network BACKUP be robust to DECnet or node failure. Neither solution will be engineered in isolation.

SIR: S87-1 Position: 7 Points: 119

Abstract: Provide additional VAXcluster management tools.

Description: Digital stresses that a VAXcluster should be managed as a single domain. To do this, more cluster-wide system management tools need to be implemented. For example, SHOW, SET, ACCOUNTING, etc. should be enhanced to work across the entire cluster (as in MONITOR CLUSTER). Perhaps a DCL TELL command (similar to the DECnet NCP TELL command) should be implemented.

Response: Digital recognizes the "systemness" exhibited by VAXclusters to end users has to date not been seen by the system manager as much as we would like. Although at a low enough level, the system manager will always have to deal with individual nodes for installation and configuration purposes, Digital intends to supply utilities to allow most day-to-day system management operations to be effected cluster-wide without the need to login to every node. This capability is being developed for a future release of VMS.

SIR: S87-51 Position: 8 Points: 108

Abstract: Ability to stop a running process and store all data to disk so that it could be restarted after a system boot.

Description: The following capabilities need to be in VMS:

1. A VMS command to suspend and outswap a process and store any related info (such as open files)

2. A VMS command to restart the process, reopening files etc.

3. A startup procedure to restart outstanding processes on reboot.

Response: Digital recognizes the need to provide some mechanism to protect customers' computing investment in long running jobs. The functionality described in this SIR is similar to that proposed some time ago for the Checkpoint/Restart project, one major difference being that the proposed solution would have required some amount of application programming. Unfortunately, that project was not completed. Checkpointing a running job is very hard to do.

It is especially hard to do in the general case. The entire process context is much more than memory and register contents. As recognized in the SIR, open files, accounting information, etc. have to be saved, and subsequently restored. It's the "et cetera" that is hard to do, maybe close to impossible in a cluster and network environment. We now feel that with the advent of VAXclusters, our computing model has changed to such an extent that this is no longer the right technical solution to the problem for the majority of our customers in the new environment. Today, we believe that the most pressing need that our customers have is for the ability to build applications in a VAXcluster environment that can survive individual CPU and mass storage subsystem failures. Our policy is therefore to consider the introduction of system features that will allow our customers to design and build highly distributed applications. The needs of customers with very long running jobs will be considered as part of this effort, but any solution will look very different from our previous designs and will require application programming.

SIR: S87-57 Position: 9 Points: 104

Abstract: Support ACL's on print and batch queues.

Description: ACL's are needed to control the usage of print devices, print queues, and batch queues. The UIC-based protections are now available on queues, but ACL's are not, so the system manager does not have sufficient granularity in granting access to the system queues and print devices. There are some system managers who would like to set up batch or print queues that could only be accessed by users holding a particular identifier. ACL's can be placed on physical devices, but they control only the ability of users to allocate the devices and do not control their ability to use shared devices such as printers.

Response: Development is currently under way to support access control lists (ACLs) on batch and print queues. We expect to provide this capability in a future major release of VMS.

SIR: S87-56 Position: 10 Points: 99

Abstract: TPU should display all special characters.

Description: Most special characters are not appropriately displayed by TPU. It is useful, and sometimes necessary, to know what characters are actually in a file. TPU should be modified to display special characters.

Response: TPU deliberately does not provide for the display of characters that aren't available in the terminal. Displaying them as "fat characters", as EDT does, would be a severe performance hit. It would also complicate the cursor movement routines of editors based on TPU, since characters would no longer be a single cell wide. The only exception to this rule is the TAB character. TPU will automatically fill a tab with spaces if the user types "within" a TAB. This would not be possible with a multiple character representation of a special character.

Fall 1987 VAX System Improvement Request Ballot

Mark Oakley

HOLD IT! DON'T PUT THIS OFF! THE DEADLINE IS OCTOBER 30. You have an opinion about what is right or wrong with the VAX. Here is your chance to influence the directions of future DEC development. The VAX Systems SIG System Improvement Request (SIR) program is an important method for the VAX user community to provide input to Digital. Your opinion is important, and every ballot adds to the influence of the SIR program. Participation on the Spring 1987 ballot dropped to 365 voters. Pageswapper circulation exceeds 7000 issues, and each issue is read by several users. Please take the time to vote. I really want to hear from you!

On the following pages, you will find the current collection of System Improvement Requests. Please take the time to review these SIR's and assess their effect on your use of VAXes. Then indicate your preferences as described below. THE SIR BALLOT FORM APPEARS IN THE "QUESTIONNAIRE" SECTION OF THIS NEWSLETTER. Also, please fill out the questionnaire portion of the ballot. This information is important to DEC, as it points out which requests are important to a particular segment of the VAX community.

Occasionally there is some confusion about the ballot. You can only vote for the SIR's that are listed below. Please provide your six-digit DECUS membership number. (If you subscribe to the DECUS U.S. CHAPTER SIG NEWSLETTER, then your membership number is the first six digits of the twelve-digit number on the mailing address.) If you are a non-US DECUS member, please provide your full membership number.

The returns from this ballot will be totalled, and Digital will provide a formal response to the 10 items which receive the most votes. The results and DEC's responses will be given at the VAX SYSTEM SIG SYSTEM IMPROVEMENT REQUESTS session of the Fall 1987 DECUS Symposium in Anaheim.

Instructions For Voting

The ballot form contains two sections, a "support" section and an "oppose" section. To indicate your support for an SIR, enter

its number in the "support" section of the ballot. You may list from zero to fifteen SIR's in this section. To indicate your opposition to an SIR you consider detrimental, enter its number in the "oppose" section. You may list from zero to five SIR's in this section.

Please return your ballot IMMEDIATELY. To allow time for DEC to respond, BALLOTS RECEIVED AFTER OCTOBER 30 CANNOT BE COUNTED.

Any ballot not specifying a DECUS membership number will not be counted. Only one ballot per member will be accepted.

Clusters

SIR: F87-1

Abstract: Improve support for VAXclusters with homogenous system disks but heterogeneous user groups.

Description: For VAXes booted in a homogenous VAXcluster increased capability and standardization is needed to prevent users from logging in to all VAXes. Because individual VAXes in a cluster may be "owned" by different groups/departments/people, a method is needed to restrict which VAX(s) a user can access. This would dispense with the management problems of heterogenous System Authorization Files and ad-hoc schemes to lockout users from individual VAXcluster nodes. This problem is particularly applicable to Local Area VAXclusters where the cost of ownership is low and the "owner" factor is high.

SIR: F87-2

Abstract: Implement cluster-accessible tape drives.

Description: Tape drives connected to individual VAXes in a VAXcluster should be usable by all nodes of the cluster. For VAXcluster configurations, the ability to use a tape drive local to another VAX would be very beneficial. Because Local Area VAXclusters cannot have tapes on HSC's, a tape drive has to be configured on every VAXcluster node or the process accessing the tape drive must be on the VAX with the tape drive. Other configurations with only local tape drives would also benefit.

SIR: F87-3

Abstract: Eliminate the requirement for VAX 11/750s to boot from a TU-58.

Description: VAX 11/750s in a VAXcluster should be able to boot from a local system disk without using the TU-58 console tape. VAX 11/750s in a VAXcluster must currently boot from the TU-58 console if an HSC disk is used for the system. This is because the CI750.BIN file must be loaded before accessing the CI. This should not be a requirement for VAX 11/750s that are in a VAXcluster and booting from a local system disk.

SIR: F87-4

Abstract: Provide high-speed communication services on a VAXcluster using SCS, not DECnet.

Description: Communication services between VAXcluster nodes is currently limited to DECnet or file sharing schemes. Digital should implement a communication interface (device driver) that uses the System Communication Services (SCS) to provide high speed data transfer between VAXcluster nodes. This would assist individual sites implementing cluster-shareable devices.

Commercial

SIR: F87-5

Abstract: Allow Datatrieve to accept abbreviated commands.

Description: Datatrieve is a powerful tool which can be difficult to use, because it can not process abbreviated commands, i.e. "SHOW DICTIONARY", instead of "SH DICT". The synonym feature is inadequate because it requires pre-cognition or considerable work to build the synonyms.

SIR: F87-6

Abstract: Increase the completeness, accuracy, and level of detail in VMS accounting records.

Description: Examples include the following:

1. Stock name for print jobs
2. Physical device name(s) for virtual and LAT terminal sessions
3. QIO counts, by device, for each ALLOCATED device and each SHARED device.
4. Report the terminal maximum speed (larger of TRANSMIT and RECEIVE)
5. Record the number of logical and physical mounts for each tape and disk drive (if any)
6. Record the name of the queue a job was submitted to as well as the device or specific queue it executed on.
7. Record queue name for subprocesses created by batch jobs.

SIR: F87-7

Abstract: Provide support for spooled output to terminal auxiliary printer ports (so that printer output can be interleaved with screen and keyboard I/O).

Description: This facility must provide configurability of:

1. Device-independent I/O at driver characteristic level
2. Same level of support as a "normal" printer
3. Terminal driver should know how to "turn on" the printer, and should be able to interleave I/O

4. The sphere of influence of this capability is "process-local" as opposed to system-wide

SIR: F87-8

Abstract: VMS Mount services should include "For READ" or "For WRITE" or "WITH RING" or "WITHOUT RING" in messages to the operator when requesting a tape be mounted.

Description: MOUNT/NOWRITE should generate a request for "WITHOUT RING" or "For READ ONLY" and MOUNT/WRITE the opposite so that operators can quickly determine what is needed.

SIR: F87-9

Abstract: Provide a fast file scan.

Description: The CONVERT utility can scan through an RMS ISAM file at lightning speed. It would be helpful in many instances to have a utility with a callable interface that would provide a high speed scan capability. This capability would be satisfactory with a random record order, but it must execute quickly.

To avoid locking at the record level, the scan could be granted exclusive access to the whole file. This scan capability would be useful for building sub-datafiles from large ones (for Datatrieve) and for standard financial batch jobs such as bill/report generation.

SIR: F87-10

Abstract: Provide a callable interface to the operator messaging services that permits query calls to OPCOM or its replacement.

Description: The current OPCOM interfaces are inadequate for a commercial environment with lots of tape mounts and other requests coming up on cluster consoles. In order to improve on what exists, it would be helpful to have a mechanism to ask OPCOM for outstanding requests of a particular or a subset of operator classes. This would lend to the building of an interactive request management tool which would run on a video or hardcopy terminal. Pending forms should generate requests to PRINTER operators so that this mechanism will cover 99% of the requests an operator needs to handle. Any functionality that crosses the boundaries of SYS$SNDJBC and SYS$SNDOPR should be consistent between them. If this is not possible due to compatability problems, invent a new call that will complement and/or eventually replace these.

SIR: F87-11

Abstract: Enhance the ALLOCATE services to allow requests to be queued.

Description: Enhance the ALLOCATE services to enable a user optionally to queue the allocation request when all qualifying devices are busy. Device allocation should be handled by a queue manager similar to the VMS V4.0 print queue manager, and the allocation request queues should be made cluster-wide to support cluster-visible devices.

User functions should include the ability to specify characteristics required of a generic device, the automatic notification of allocation, the ability to delete an allocation request, the ability to examine the allocation request queue, and the ability to do other interactive processing while waiting for an allocation request to be granted.

Operator functions should include the ability to mark failing devices as unavailable and the ability to force a deallocate. Manager functions should include the ability to define device characteristics and specify physical devices as possessing those characteristics.

Device allocation and deallocation should place records in the accounting file so that charge back accounting can be done for allocated devices.

A mechanism for avoiding deadlocks when multiple devices are allocated should be provided.

Examples:

```
$ ALLOCATE/QUEUED/WAIT TAPE$CLASS:-
    /CHARACTERISTICS=(DENSITY:6250) LOGICAL_TAPE
```

(Queue an allocation request for a tape drive with 6250 bpi capability and wait until the allocation has completed.)

```
$ ALLOCATE/QUEUED/NOWAIT/NOTIFY DISK$CLASS:-
    /CHARACTERISTICS=(RA60) MY_DISK_PACK
```

(Queue an allocation request for an RA60 disk drive and return control to my terminal. Notify me when the allocation has completed.)

```
$ ALLOCATE/NOQUEUED TERMINAL$CLASS:-
    /CHAR=(AUTODIAL,BAUD:1200) DIAL_OUT_MODEM
```

(Allocate a terminal device with a 1200 baud autodial modem but don't queue the request. Give an error if all such devices are allocated.) The queueing capability might be implemented via a symbiont. The queueing capability should also be provided for the MOUNT services.

SIR: F87-12

Abstract: VMS should implement tape automatic volume recognition and provide the security normally associated with volume labeling.

Description: VMS should provide a complete implementation of automatic volume recognition (AVR) for tapes, that may be enabled/disabled by the operator on a per drive basis. This means that (with AVR enabled), when a tape is mounted, the system checks possible labels and honors mount requests without operator intervention, if possible. If a job needs 4 tapes, the operator can mount them all if enough drives are available and then forget about them until somebody else needs the tape drives. It should also be possible for a user to request a tape mount based solely on the tape's label and density. The user should not be required to know what physical devices implement a particular tape density

on a particular system. VMS should also support a "visual id" or "slot number" which is displayed in all operator messages related to the mount.

It should be possible to operate a VMS system in a mode where all tapes are under system/operator control. This means that they are pre-initialized and users are not allowed to change the labeling on the tape without special privilege. The BACKUP utility must also conform to such labeling restrictions, thereby insuring that the BACKUP data is written onto the proper reels. VMS should require explicit operator intervention for unlabeled tapes. It is not acceptable that an unlabeled tape which happens to be on a drive be automatically assigned.

SIR: F87-13

Abstract: All utilities should use a standard format for printable output.

Description: Printable output generated by VAX utilities and compilers comes in a great variety of record formats and carriage control conventions. A particularly awkward convention is the use of embedded ASCII control characters to generate multiple print lines from a single record. There appears to be no standard for this or any other mechanism. As a result it is very difficult to print "printable" output on non-DEC printers or transmit it through heterogeneous networks. Digital should document a standard record format and carriage control convention and modify all facilities to conform to this convention. As an alternative, Digital should provide a utility which converts all currently used formats into a standard format. It seems that this functionality currently exists, distributed between the print symbiont, device driver, and "DEC standard" printers.

SIR: F87-14

Abstract: Provide support for simple project accounting.

Description: The Spring 1985 VAX SIR Ballot contained a request for project accounting in VMS. Digital's response was "We also feel that project accounting is very important...We feel that this is a reasonably complex area and, as such, some of the enhancements that we intend to make in this area will appear over time."

Project accounting is something that is desperately needed at large sites. In its simplest form, project accounting should provide a SET PROJECT command that would write a process accounting record, and start recording a new record with a new account string specified by the user. The account string should be verified before these actions take place. The system manager should be allowed to set up a file which specifies which UIC's are permitted to use individual account strings.

Many sites have immediate government or internal security requirements for "one username per user" level of accountability. DEC should provide this form of project accounting until their full-blown system is available.

SIR: F87-15

Abstract: Enhance BACKUP to provide first and last file names logged for each volume of storage media and an incremental restore capability for a directory structure.

Description: BACKUP should log the first and last file on each volume to assist in choosing tapes for restoration.

Directories or entire directory trees sometimes become unusable. To aid in recovery, BACKUP should support the following procedure:

1. Delete the structure(s) affected.

2. Restore that structure from the last image mode backup.

3. Restore the selected structure(s) in incremental mode.

DCL and Utilities

SIR: F87-16

Abstract: DCL WRITE command needs a method for terminating a write operation without generating a CR/LF sequence.

Description: When using the DCL write statement, there currently is no method to terminate the operation and prevent the CR/LF sequence. This would be useful when positioning the cursor on the display to a particular location, such as a default response indicator or fixed response location. Any subsequent read operation performed from the terminal would need to process any type-ahead text properly as well as normal response characters not typed ahead.

SIR: F87-17

Abstract: More capabilities for VAX-11 RSX BRU.

Description: VAX-11 BRU would be more convenient to use for interchanging files between RSX systems and VMS systems if the VAX-11 BRU were enhanced to know enough of ODS-2 structures to allow access to rooted directories. This feature would permit reading or writing of the rooted portion of the directory tree as if it were the [0,0] directory of the device as BRU sees it.

SIR: F87-18

Abstract: Enhanced command line RECALL capabilities.

Description: The functionality of the command line RECALL facility would be greatly increased if users were able to tailor some features to their specific needs. It would be desirable if these (YES FOLKS we are asking for still more SYSGEN parameters) features could be set on a user-by-user basis, but site by site at boot time would be ok. The expansion tailoring would allow sites to set:

1. The size in bytes of the command line recall buffer.

2. The maximum number of commands to be recalled.

3. The size of the DCL command line expansion area.

4. The size of the DCL command input area. This would allow larger commands to be passed to user-written programs by the foreign command interface processor.

SIR: F87-19

Abstract: Add more capabilities for manipulating DCLTABLES

Description: Many users desire or need to modify DCLTABLES to restrict access to certain commands, command options or add their own or third party software as a DCL command. Some form of support to facilitate this is needed, even an extra cost layered product. A minimal form of support would be a listing program that would produce readable output to allow the user to:

1. Check conflicts in names.

2. Verify options.

3. Determine if the command was added, when it was added, and if it is from VMS.

SIR: F87-20

Abstract: DCL status return enhancements.

Description: Programs that are called by DCL should implement some form of expanded status reporting that is testable at the DCL command level. For example if DIFFERENCES were invoked, some indication in $STATUS if the files were the same or differences were found, would permit users to act accordingly. For example:

```
$DIFFERENCE F1.TXT F2.TXT
$IF $STATUS .EQ. DCL$DIF_NONE THEN $DELETE F2.TXT
```

Some form of documentation would be needed to allow users to write appropriate tests. The return values could be defined either by numeric returns or reserved symbols known to DCL.

SIR: F87-21

Abstract: Enhance SET HOST error reporting.

Description: The DCL trapping of CONTROL_Y within the SET HOST command and the current exit processing of a yes response to the question:

```
Are you repeating CONTROL_Y to abort the remote session...
```

fails to indicate that the SET HOST connection was aborted. Some indication of failure to log off normally would aid in processing errors or performing any needed cleanup.

SIR: F87-22

Abstract: Add the ability to run a detached process for a specified user name.

Description: The ability to run a detached process under a specified user name for a suitably privileged user would provide the ability to do this directly. A technique of putting the run command in a command procedure and doing a SUBMIT/USER= works but may require additional work to get

the job to the start of a batch queue or even require the creation of a batch queue.

SIR: F87-23

Abstract: Make the DCL /LOG qualifier more consistent.

Description: Some commands (Backup, Copy, etc.) accept /LOG, others such as (Print, Submit) use /IDENTIFY to produce documentary output. These commands should all support the /LOG or some new qualifier, /DOCUMENT for example, that would produce documentary output. This new qualifier would be consistent across all commands and ignored on commands that can produce documentary output such as SHOW TIME.

SIR: F87-24

Abstract: Add the capability to capture an interactive session exactly as it happens.

Description: In many cases VMS users need to produce a disk file with the transcript of a terminal session. The need for this is to produce documentation for manuals, turn in as homework for class, or to submit an SPR. The SET HOST/LOG does not completely emulate the terminal output, especially when CR/LF output is suppressed to allow the user to respond to a question on the same line. Also if the SET HOST command has been removed for a user this feature becomes nonexistent. Some command such as SET LOGGING TO <filespec> is needed to provide this feature. The UNIX script utility would provide a good model for this. Obviously if the captured file contained graphics escape sequences or other non-printable characters it would be the user's responsibility to handle them. The ability to record escape sequences into a file might also be a useful debugging tool for some users.

SIR: F87-25

Abstract: Enhance sysgen parameter readability.

Description: SYSGEN would be more useful if it were modified to provide a better organization of parameters, e.g., memory, terminal, timing, security, VMS mystery, etc. Sorting the output into alphabetic order would also make finding the parameter value in the listings easier.

SIR: F87-26

Abstract: Enhance VMSMail.

Description: VMSMail should be enhanced in the following ways:

1. Allow a user to retract a sent mail message. This could be limited to the last message sent. This would be very useful to retract that nasty undelivered mail message sent to the SYSTEM MANAGER before it is read and you end up with mandatory 32 character one time passwords!

2. Provide a facility to append comments to a received mail message and redistribute it.

3. Provide some form of return receipt when the recipient has read your mail message.

4. Provide a facility to allow users to configure the default printer orientation for printed mail messages. Most mail messages are oriented to portrait mode, and not the default landscape mode found on most programmers' printers.

5. DEFINE/KEY in mail should support /ERASE in the same way that the DCL DEFINE/KEY does.

SIR: F87-27

Abstract: Enhance SET HOST/DTE for more modems including those made by Digital.

Description: The Digital DF224 modem is not compatible with SET HOST/DTE/ DIAL=NUMBER. Support is needed for all modern DEC modems. Support for popular third party modems such as HAYES, RACAL-VADIC, etc. would be desired.

SIR: F87-28

Abstract: Enhance SHOW PROCESS command to display open files and information about subprocesses.

Description: Some form of identification is needed for the SHOW PROCESS command to make tracing subprocess trees easier, possibly of the form SHOW PROCESS/SUBPROCESS/ID=<pid>. If a user has two processes running in a batch queue or even from two terminals, and each process has a subprocess it is very difficult to determine which subprocess is owned by which parent. The ability to show the files that a specified process has open is needed. SHOW DEVICE/FILES on one-drive systems with many installed images provides too much output. If this feature could also show the current location within each file, then estimating what portion of a file had been processed by a program would be significantly easier.

SIR: F87-29

Abstract: Enhance MOUNT/FOREIGN for uninitialized tapes.

Description: The MOUNT/FOREIGN command will time out and not complete properly on a VIRGIN BLANK tape. Some fix to avoid failing on a blank tape is needed.

SIR: F87-30

Abstract: Enhanced DEFINE/KEY capabilities.

Description: DEFINE/KEY should support control characters and escape sequences, and allow multiple input lines to be defined with the extra lines being placed into the type-ahead buffer. For example:

```
$!                                  !Customize keyboard
$DEFINE/KEY KP2 "^E"                !EDT go to end of line
$DEFINE/KEY comma "<DEL>"           !EDT delete char at cursor
$!                                  !MULTIPLE INPUTS
$DEFINE/KEY PF4 "^B^H"              !Recall and edit command
$DEFINE/KEY KP3 "MAIL<CR>DIRECTORY NEW MAIL"
```

SIR: F87-31

Abstract: Add a /BELL qualifier to certain DCL commands.

Description: The addition of a /BELL=n qualifier command to DCL to cause the terminal bell to ring N times with a discernable pause would be very useful to draw attention to the terminal when a long running command completes in any fashion.

SIR: F87-32

Abstract: Restore CONTROL_U behavior to pre-V4 status.

Description: The CONTROL_U sequence in V4 fails to provide feedback when the terminal is set /LOCALECHO. This is inconsistent with the other control sequences (^B,^C,^O,^Q,^R,^S,^T,^Y,^Z) all of which provide user feedback. Prior to V4 the ^U sequence both cleared the terminal input buffer and generated a new line/prompt sequence to the terminal. In V4 only the input buffer is cleared which is the expected behavior if the terminal is set /NOECHO/NOLOCAL_ECHO. Given that users of V4 and up may now want this behavior, an additional switch such as /OLD_LOCAL_ECHO would be a way to allow a choice in how the terminal should behave. This request is specifically a request to restore the behavior in /LOCAL_ECHO to what it

was prior to V4 and make no changes to the /ECHO and /NOECHO/NOLOCAL_ECHO behavior.

SIR: F87-33

Abstract: Enhance the DCL DEFINE command.

Description: The DEFINE command should be enhanced in the following area:

1. All features of the ASSIGN command should be provided in the DEFINE command.

2. Because there is a DEASSIGN command to negate the effect of the ASSIGN command, there should be an UNDEFINE command to negate the effect of the DEFINE command.

3. In the DCL documentation, related topics should be listed for each command. The DEASSIGN command should be mentioned in the documentation about DEFINE.

SIR: F87-34

Abstract: The functionality of the DEFINE and ASSIGN commands should be in a single command.

Description: It is confusing to have two DCL commands that perform similar functions. One of the commands should be eliminated and lost functionality transferred to the command that is retained. It would be helpful if some procedure were supplied that would aid with this conversion, as many command procedures would require modification.

Internals

SIR: F87-35

Abstract: Implement a system service equivalent to the lexical function F$FILE_ATTRIBUTES.

Description: Although information delivered by F$FILE_ATTRIBUTES can be obtained via System Services, this is tedious since various RMS data structures have to be established. A direct System Service would be valuable and should not be hard to implement since the code is already there.

SIR: F87-36

Abstract: Images that are linked on VMS Version "4+n" systems, need to execute on Version 4 systems.

Description: Sites that export software to customers are not able to upgrade quickly to new releases of VMS because not all customers have upgraded. There needs to be a way for software that is linked on a "4+n" version of VMS to execute on a version 4.

SIR: F87-37

Abstract: Provide "wildcard" support for the SYS$TRNLNM system service.

Description: From DCL it is easy to display all of the logical names in one or more logical name tables. This capability does not exist at the programming level. The SYS$TRNLNM system service should be enhanced to provide "wildcard" support to provide this functionality.

SIR: F87-38

Abstract: Provide synonym node names for ACP QIO (non-transparent) DECnet operations.

Description: DECnet-RSX provides the ability to define synonyms for actual node names. When such a name is specified in a network connect, that name is translated and the resultant name used as the target of the connection. A synonym node name capability under VMS offering the same functionality as DECnet-RSX is needed.

SIR: F87-39

Abstract: Provide explicit control over all caches when a disk is mounted.

Description: Disk-cache control is needed to support mounting dual ported disks read/write on one system and read only on another. It would also allow better tailoring of cache sizes to activity on the individual disks.

SIR: F87-40

Abstract: "New Mail" notification on login should work for all layered electronic mail systems.

Description: VMS should provide a mechanism that allows layered electronic mail systems to notify a user of new mail at login time. Currently, a user has to activate the layered electronic mail system to determine if there is new mail.

SIR: F87-41

Abstract: Implement higher speed terminal baud rates (e.g. 38400 and 14400)

Description: Many graphics applications produce complex images. These application would execute more quickly if higher baud rates were available. Various hardware vendors support higher baud rates, and Digital's DHU-11 supports 38400, but the DCL command SET TERMINAL/SPEED does not. It would be also useful if baud rates between 38400 and 14400 were available.

SIR: F87-42

Abstract: SYSGEN should provide a DISCONNECT command to cancel the effect of a CONNECT command.

Description: It is very inconvenient to have to reboot the system to get rid of a device. The capability to remove a device should be added to the SYSGEN utility.

Languages, Tools, and Editors

SIR: F87-43

Abstract: The capability to specify "footers" should be added to RUNOFF.

Description: Currently, RUNOFF has the capability to put one or two lines of text at the top of every page, via the .TITLE and .SUBTITLE directives. However, there is no capability to put text at the bottom of every page. It would be useful if such a capability existed.

SIR: F87-44

Abstract: Add a sorting capability to VMS text editors.

Description: It would be useful if a sorting capability was available in the various VMS text editors. Currently, it is necessary to exit the editor to perform a sort. It would be desirable if the syntax of an editor sort command could be similar to the DCL SORT command.

SIR: F87-45

Abstract: Provide a routine to return message text corresponding to the Fortran I/O status specifier "IOSTAT".

Description: It would be very useful to have a routine that would return the message text that corresponds to the Fortran I/O status specifier "IOSTAT".

SIR: F87-46

Abstract: Standardize data-type support for VAX/VMS high-level languages.

Description: VAX/VMS high-level languages do not all support the same data types. As a result, access to data is difficult across different languages. For example, a Cobol variable with 10 digits and 2 decimal places can not be manipulated in a BASIC subroutine. A similar problem exists for some record definitions in the Common Data Dictionary.

SIR: F87-47

Abstract: Make the "TRUE" value of a Fortran logical variable consistent between Fortran and the debugger.

Description: Executing the following Fortran statement causes a value of -1 to be stored in the logical variable:

```
LV = .TRUE.
```

Executing the following debugger statement causes a value of +1 to be stored in the logical variable:


```
DEPOSIT LV = .TRUE.
```

The Fortran expression:

```
LV .EQ. .TRUE.
```

will evaluate to false if the debugger is used to deposit a "TRUE" value. Either Fortran or the debugger should be altered to provide consistent results.

SIR: F87-48

Abstract: Enhance CMS to support wildcard element-expressions for elements in a group.

Description: Currently, CMS supports wildcard element-expressions in the CMS SHOW statement only if the element-expression is not a group name. It would be convenient if wildcards could be used when listing elements of a group. The syntax might look something like:

```
CMS SHOW ELEMENT group:*.typ
```

SIR: F87-49

Abstract: Enhance the alphabet of LIB$TPARSE to recognize high-level-language constants.

Description: It would be convenient if high-level-language constants (e.g. floating point, complex, logical, etc.) were recognized by LIB$TPARSE. It would also be desirable if LIB$TPARSE supported user-defined entries in the alphabet.

SIR: F87-50

Abstract: Enhance the symbolic debugger to provide traceback information on COBOL PERFORM statements.

Description: Currently, the symbolic debugger provides traceback information about CALL statements, but not PERFORM statements, in COBOL programs. Traceback information about PERFORM statements would be very useful for debugging programs. This traceback information should also be provided when a COBOL program aborts.

SIR: F87-51

Abstract: Provide line number support for VAX Macro, that can be used by the symbolic debugger, PCA, LSE, and other utilities.

Description: The usefulness of various utilities is limited because VAX Macro does not provide line number support. PCA (Performance Coverage Analyzer) is severely limited because program locations must be specified as virtual addresses. Thus, coverage and other code-related measurements are not useful.

Debugging would be made easier if line numbers could be specified, instead of cryptic-looking addresses.

SIR: F87-52

Abstract: TPU should display all special characters.

Description: Most special characters are not appropriately displayed by TPU. It is useful, and sometimes necessary, to know what characters are actually in a file. TPU should be modified to display special characters. Digital is reluctant to implement this feature, claiming that performance and speed would be sacrificed. If so, then this feature could be an option that could be selected via a switch on TPU or SET directive in TPU, or both.

SIR: F87-53

Abstract: Add a /NOWAIT switch for TPU.

Description: If TPU had a /NOWAIT switch that could be set when "DCL" commands were issued to be run by TPU, the user could continue with work in TPU while the "DCL" command continued to run in the sub-process. When the sub-process terminated TPU would inform the user of this in some fashion. No more than one "DCL" command running in this mode would be an acceptable restriction as well as requiring a terminal that could have a reserved scrolling region for the message from the "DCL" command to show up in.

Miscellaneous

SIR: F87-54

Abstract: There should be more Digital developers at DECUS Symposia.

Description: With the enormous increase in VAX software provided by Digital over the years, there is an accompanying complexity. The limited number of software developers who are sent to Symposia try valiantly to "cover" for software which is not their own, but it is difficult. Since DECUS symposia are the only way for the user community to engage in a dialog with Digital software developers, there should be a considerable increase in the number of developers who are sent. This could be done by reducing the number of marketeers who are not able to provide technical details. The Exhibit Hall could also be reduced to exclude "seen-before" hardware.

SIR: F87-55

Abstract: Digital should provide Sales Updates for customers.

Description: Digital should provide (even at a price) subscription service to the new product offerings in the "Sales Update". Other vehicles, including waiting for the "Consultants Reference Guide" and the DEC field offices to relay technical information, are unsatisfactory.

Security

SIR: F87-56

Abstract: Better control over DECnet remote file access.

Description: The RMS file protection defines WORLD access to include all those outside the owner's group. It would be useful to define several classes of users as follows:

1. All WORLD users on the local node.

2. All users local to this VAXcluster.

3. All users on nodes within this DECnet area.

LOGINOUT currently gives a process the Identifier "NETWORK" if that process is being created in response to a network request. It would be useful to have greater granularity of access control for network processes by having additional identifiers created based on the node, cluster, and area from which the access is being attempted. This capability might possibly be achieved by having the File Access Listener, LOGINOUT, or some other privileged image set up the additional process Identifiers.

SIR: F87-57

Abstract: Support secondary passwords in DECnet access control strings.

Description: VMS V4 allows user accounts to have two passwords. However, DECnet does not allow the secondary passwords to be used in access control strings. The only way such accounts can be used through DECnet is via proxy logins.

SIR: F87-58

Abstract: Enhance COPY to copy ACL's.

Description: The COPY utility does not currently handle ACL's. It should be enhanced to propagate any ACL's from the source file to the destination file. However, there may be many times when a user is copying another user's file in order to modify it for his/her own purpose. It is likely that in such cases the user would not want to propagate ACL's from the original file. Therefore, this capability should be available via an additional qualifier to COPY, e.g., /PROPAGATE.

SIR: F87-59

Abstract: A mechanism is needed that allows one user to grant other users access to a file only via a user-defined image.

Description: Non-privileged users sometimes need to give other non-privileged users controlled access to data files through a program. Through this facility any user would be able to control who could access his data files and what kind of access they may have. In the current system, in order to allow another user to add a record to a file, that user must be given WRITE access to that file, which means he could alter existing data or delete records from the file.

Presently this requires the system manager to install the program with privilege, which is both an administrative nuisance and a security problem, as the privileged image would also have access to other system data files as well

as the intended files. This mechanism should be under user control, i.e., the user should be able to determine which images could access a file. For example, the UIC of the image and data file could be required to match before access would be permitted. This could be accomplished by an option on the compiler or linker when the image was being created. It could also be implemented by allowing the system manager to install an image with a particular identifier. Then the user could set up the access control list for that file to permit access by that Identifier. This would be less flexible but would permit a user to allow access from images other than his own, e.g., a data base manager. This capability could also be provided by file passwords, since the passwords could be imbedded in the programs that were intended to have access to the files; however, file passwords would be difficult to administer and prone to disclosure.

SIR: F87-60

Abstract: Implement mandatory security controls in VMS.

Description: Many VAX systems are being operated by government agencies or contractors and either are processing or need to process classified data. Mandatory security controls are needed in VMS to support such classified processing. An operating system that could be evaluated by the National Computer Security Center at the B1 level or higher would encourage more sites to use VAXes for classified processing and make system management much easier for those already doing classified processing on existing systems. The system manager should be able to specify a security level for each user account using the AUTHORIZE utility. When a file is created, it should be given the security level of the creating process, and any subsequent access to the file should be controlled in accordance with the mandatory security policies. If a file is edited or copied, it should retain its classification. A utility should be provided to allow a person with a special (SECURITY) privilege to change the classification of a file.

SIR: F87-61

Abstract: End-to-end encryption of logical connections within DECnet-VAX.

Description: The assumption made by DECnet that all nodes and communications paths are trustworthy is not viable in many environments. End-to-end encryption of the data portion of network packets is required in these environments to assure that eavesdropping is fruitless, both in Local Area Networks (broadcast) and Wide Area Networks (multi-hop). This encryption should be implemented so as to be transparent to the application programmer and user. I.e., the mechanism should be located in the NSP (or OSI session) layer. New encryption keys should be generated for each logical connection between cooperating, encryption-capable processors. (Some nodes will not be capable of encryption and should be allowed to participate in the network without performing encryption.) Intermediate nodes should not be required to participate in, or be knowledgeable of, the key distribution/management or the encryption process. The DES algorithm should be utilized in the near term but should be readily replaced as NBS standards change. Provisions should be made for encryption hardware to boost performance where necessary.

SIR: F87-62

Abstract: Security alarm messages to a file.

Description: Add an option to the Access Control Entries (ACE's) that specifies a file into which security alarms for that file/directory are written. This would allow a user to review security alarms for his own files, rather than depending on the system manager to perform the auditing. Of course, security alarms requested by the system manager via the SET AUDIT command should be written to the system-wide security log.

SIR: F87-63

Abstract: Support DECnet proxy access for SET HOST command

Description: When a user logs into a remote host via the SET HOST command and a DECnet proxy exists in the NETUAF on that host, the user should have the option of being automatically logged into that proxy account. This would be extremely helpful to less-advanced users who switch frequently between systems. It would also reduce the chances of disclosing user passwords, since they would not be transmitted across the network if the proxy were used. A /PROXY qualifier could be added to the SET HOST command to allow the user to request proxy access.

SIR: F87-64

Abstract: Provide lexical function for getting RIGHTSLIST information

Description: An F$RIGHTS lexical function should be provided to return the list of identifiers held by a user (similar to SYS$FIND_HELD). Also, an F$ACCESS should be provided to return a boolean logical value indicating whether access to an object is allowed given an input rights list.

SIR: F87-65

Abstract: Allow a general identifier to be the owner of a process.

Description: It should be possible to make a general identifier the owner of a process (in place of a UIC), so that:

1. Owner access will be granted via the protection mask to objects owned by that identifier

2. RMS scratch files will be owned by that identifier and charged against its disk quota.

System Management

SIR: F87-66

Abstract: Stand-alone BACKUP should be supported on a greater variety of devices.

Description: It would be convenient if Stand-alone BACKUP could be booted from RX02 and tape drive units. RX02 drives are faster than RX01 drives, can hold two floppies, and can operate at double density. Stand-alone BACKUP would boot much faster from the RX02.

Currently, Stand-alone BACKUP can be booted from the TK50 and TU58. It seems that this capability could be extended to other non-random access devices (TU78, TU81, etc.). Such a capability might be an attractive alternative to the TU58 and other slow devices.

SIR: F87-67

Abstract: Enhance AUTHORIZE to work on a selection or subset of all users according to selection criteria. Subsequent commands should apply to these criteria.

Description: It would be useful to construct a subset of all users and have AUTHORIZE operate only on this subset. This feature might look like:

```
UAF> SELECT * /ACCOUNT = PROJECT1
UAF> MODIFY * /SELECTION /WSQUOTA = 1000
UAF> LIST * /SELECTION
UAF> .
UAF> .
UAF> .
UAF> SELECT *  /((ENQLM  <  20)  .AND.  (ACCOUNT  =
DATABASE))
UAF> MODIFY * /SELECTION /ENQLM = 50
UAF> LIST * /SELECTION
```

SIR: F87-68

Abstract: There should be a tape-to-tape copy utility on the HSC50 and HSC70.

Description: It would be very useful if a tape-to-tape copy utility was available for making copies of tapes.

SIR: F87-69

Abstract: Provide the ability to limit the number of disks that are displayed, when using the MONITOR utility.

Description: The display that is produced by "MONITOR DISK" includes all of the disks that can be "seen" by the local cpu. This display can be quite long if the local cpu is a member of a cluster with a large disk farm. Frequently, only a few disks need to be monitored. The MONITOR utility should be modified so that particular disks can be included and/or omitted.

SIR: F87-70

Abstract: Enhance BACKUP to provide additional attributes for output files.

Description: BACKUP should provide the same qualifiers that are available on the SET FILE command (e.g. /PROTECTION, /ACL, etc.). Such qualifiers would facilitate the restoration of files.

SIR: F87-71

Abstract: Provide accounting information about terminal servers.

Description: Currently, accounting information does not capture the port number and terminal server name for interactive session that login via LAT's. Port number and terminal server name information is extremely useful for trouble-shooting and determining terminal usage.

VAX/VMS Security

Number next in a series - May, 1987

by
Ray Kaplan
PIVOTAL, Inc.
6892 E. Dorado Ct.
Tucson, Arizona 85715

Hurrumph!

I had hoped that this column would become an active interchange of VAX/VMS security information. Since my mailbox has lain fallow for some months, the column has become infrequent. Normally, I put DECUS things at a good relative interrupt level in my life, but not even the highest interrupt priority will help if an actual interrupt never comes in! As it is, this next article just dribbled out of the queue on its own.

Buy This Book

As I wander around teaching VAX/VMS security and getting involved with people's VAX/VMS security problems and with the larger world of computer security in general, I note that there seems to be a lack of practical literature on the subject. Besides writing my own VAX/VMS security book, I am always on the look out for books and articles on the subject of computer security.

While I have had limited success in my search for VAX/VMS specific literature, I have discovered a lot of good general computer security literature recently.

Back in October, I attended the Computer Security Institute's security conference in Atlanta, where I found a session on "Hacking and emanations demonstrations". This, in itself, was an eye opener, especially since the NSA prevented the emanations demonstration from being given. More on that interesting topic in a future issue! The chair of the session was one Jerome Lobel, who is the director of computer security for Honeywell, Inc. in Phoenix, Arizona. He has a relatively new book entitled "Foiling the System Breakers, Computer Security and Access Control." Since Jerome is a recognized authority on computer security and an active participant in the international

computer security community, finding his book to be well written and articulate came as no surprise.

In the first chapter of the Guide to VAX/VMS security that comes with the full VAX/VMS manual set, there is a great, but tiny, taste of some of the security issues that need to be considered outside of the VAX/VMS specific "nuts and bolts" which the VAX/VMS manual rightly concentrates on. What is missing is a complete introduction to understanding the problem in the first place.

By the way, if you have a MicroVAX, you did not get this manual with your software. You have to buy it separately, and I would recommend that you buy the complete manual set. In the event that you want to buy only the security manual, its number is AA-Y510B-TE.

In my experience, there is too little discussion of the "front end" of this whole security business. Threat analysis, security policy and the like. While that is what Lobel's book does best, I liked his presentation of practical examples. For instance, in the section on trade-secret theft and software privacy there is a good discussion of exactly how large the problem really is.

He has structured the book around what he calls a "computer security awareness program", and presents the material in a set of planning phases which are geared to provide you with a template to use for your own security plan implementation. This, in itself, makes the book a worthwhile purchase.

In part one of the book, entitled "Understanding the Need for Computer Access Controls", he starts out with a discussion of how to analyze your system's security needs by looking not only at system vulnerabilities but at trends in computer fraud and industrial espionage as well. His discussion of the history of computer fraud is complete with examples of alleged computer crimes and even includes some ideas of what computer crimes will be committed in the future. The discussion of personal privacy and data access is complete with a summary of both the domestic and the international major legal issues.

In the "Establishing a System Security Policy" section, he presents information on how to go about classifying and protecting information, risk analysis and technical vulnerability evaluation. If you have ever tried to do this sort of thing at your site, then you know how hard it is to get going without some guidelines. Jerome provides good, practical

guidelines. Since most VAX sites that I see do not have a system security plan, this section is perhaps of greatest interest to VAXers, as it provides a step-by-step guide on how to set up a computer security program.

His discussion of how to select access control tools and technology includes an overview of operating system security and even though it leaves out VAX/VMS, it will widen your perspective on the problems at hand. Included in this section is a brief introduction to DOD trusted computer systems, database security, network security and data encryption.

The discussion of how to complete a secure system design includes introductions to office automation, personal computer protection, and home computer security. He finishes the book with a section on implementing and monitoring access controls and a section on coping with change.

What the book lacks in the way of DEC-specific information is more than made up for by its rich collection of introductory material and practical "how-to-do-its" for computer security implementation.

If you can't seem to get the time to play with your VAXes' security features, I suggest that you buy this book for your manager. The net result should be a new awareness of the potential security problems around your site, and with a little push from you, you might end up with some more time to spend making sure that your VAX is secure.

Priced as a professional book, "Foiling the System Breakers" is not cheap. However, a SIGNIFICANT discount is available on volume purchases. If you decide to buy the book based on my recommendation, drop me a line. I will offer to coordinate a "group buy" if you would like.

Foiling the System Breakers
Jerome Lobel
ISBN 0-07-038357-X
1986, McGraw-Hill

McGraw-Hill Book Company
1221 Avenue of the Americas
New York, New York 10020

till next time, Happy VAXing!

Ray

SDA "FORMAT" Problem Solved

Jamie Hanrahan
Simpact Associates
9210 Sky Park Court
San Diego, CA 92123

In the version of SDA that shipped with VMS prior to VMS V4.0, device driver writers could add their own symbols (for UCB extensions and the like) to those used by SDA's FORMAT command. In V4.0, this feature appeared to go away. Though it's still possible to read an object file to define one's own UCB-prefixed symbols, SDA apparently refuses to use them.

Seth Stern of Reliance Electric recently called me with the solution. It seems that the V4.x SDA's FORMAT command is a lot smaller than that in old versions. If the TYPE field of the data structure being formatted indicates that it's a UCB, FORMAT looks in the device type and class fields of the UCB, and uses internal tables to determine which UCB symbols to use!

This makes sense for DEC standard devices. DEC has defined many, many UCB extensions. With this behavior, when you format (for instance) a terminal UCB, the fields in the extension are interpreted with the symbols relevant to terminals, and those for disks, mag tapes, etc., are suppressed. But it doesn't help those of us who are trying to debug foreign device drivers.

The ideal fix would be for DEC to change this behavior so that if the class and type fields indicate that the UCB is a non-DEC device, FORMAT will use none of the device-specific UCB symbols, but will use any other UCB symbols it knows about--including those defined by foreign driver writers.

In the meantime, Seth offers the following workaround: Define symbols for your UCB extension that start with something other than UCB--for instance, XYUCB for XYDRIVER. Then use the TYPE qualifier on the FORMAT command:

```
SDA> FORMAT/TYPE=XYUCB address
```


Of course, now you don't get the standard UCB symbols (unless you expand $UCBDEF and change all the UCB prefixes to your own). But it beats squinting at EXAMINE output.

INPUT/OUTPUT

A SIG Information Interchange

A form for INPUT/OUTPUT submissions is available at the back of the issue.

To register for on-line submission to the Pageswapper dial:

(617) 262-6830

(in the United States) using a 1200 baud modem and log in with the username PAGESWAPPER.

```
================================================================
Note 416.2       Journal files created using Backup       2 of 2
"Michael R. Pizolato"                7 lines  29-APR-1987 10:09
              -< To all those who inquired... >-
----------------------------------------------------------------
```

I received a lot of requests for copies of my BACKUP command procedure. But, my company won't let me send any without going through hell for permission. So, I'm going to submit it to DECUS, and you'll be able to get it on tape from them. It's easier for me to get publication permission that way than permission to send tapes the other way. I hope to have it out soon. Sorry about that.

Michael R. Pizolato
AT&T Technology Systems
Dept. 323610
555 Union Blvd.
Allentown, PA 18103
215/439-5500

```
================================================================
Note 548.4            Disc Corruption Problem             4 of 5
"Ray Kaplan"                          22 lines  15-MAY-1987 02:55
                  -< A still later reply >-
----------------------------------------------------------------
```

On the cluster: A while back (early March) I heard of a serious problem with shared disks. Seems that the XQP on one processor doesn't know how to let other cluster members know that INDEXF.SYS itself has fragmented. This is rumored to trash the disk. The fix that was suggested was (as you might guess) to make INDEXF.SYS bigger that you will ever need it to be. This was reported under V4.5 with HSCs.

On the the stand alone disk, I haven't heard anything specific.

At the Nashville Symposium, I ran into a fellow that is seeing the contents of random disk blocks come out when his SYS$ANNOUNCE and SYS$WELCOME files are printed out. Might be the same problem. If you want his name (and are still interested in this after all of this time) - please call.

Ray Kaplan
PIVOTAL, Inc.
6892 E. Dorado Ct.
Tucson, Arizona 85715

```
================================================================
Note 548.5            Disc Corruption Problem             5 of 5
"Kevin Angley"                        14 lines  26-MAY-1987 11:23
       -< Index file corruption in clusters does exist >-
----------------------------------------------------------------
```

There is indeed a problem with the index file on disks shared in a cluster. Telephone support will send you a patch. Naturally, the patch has bugs. It reintroduces the problem with append giving you an error message whenever it uses group privilege to write to a file. Because of this problem, we retreated to the original 4.5 and will take our chances. These problems were reported to TSC, with appropriate expectations for results.

It is also interesting to note how worthless DSIN is ... neither the index file or the problem we reported with the patch that TSC is sending out is mentioned in this "online database of up to the minute news flashes of problem reports". DSIN is worthless.

Kevin Angley
3301 Terminal Drive
Raleigh, NC 27604
(919) 890-1416

```
================================================================
Note 567.8          BACKUP Hints and Questions            8 of 9
"Bruce Bowler"                         6 lines  27-APR-1987 13:35
                    -< 45' cable length >-
----------------------------------------------------------------
```

Re 567.6. I just talked to my service engineer this morning and he mentioned that he had talked to A DECie from Colorado who confirmed that when dealing with K.STI and TA78 there IS a cable length restriction of 45' (not meters, feet) on the SDI cable. This coupled with the lengths of cables that DEC sells effectively limits you to 12' or 25' cables.

Bruce Bowler
General Electric
1 River Road
Bldg 2 Room 609
Schenectady, NY 12345

```
================================================================
Note 567.9          BACKUP Hints and Questions            9 of 9
"Ken A L Coar"                         6 lines  26-MAY-1987 13:45
              -< A DECie [?] says USE /CRC >-
----------------------------------------------------------------
```

Just a note from Nashville, with no interpretation at all: I overheard a VMS developer tell users NOT TO USE /NOCRC, even if their tape drives do a CRC is hardware. I was passing by, and couldn't stick around to find out more - but there it is.

#k

Ken A L Coar
General Dynamics
Office Systems
12101 Woodcrest Executive Drive
Creve Coeur, MO 63141
(314) 851.4003 (CST)

Note 585.5 Anyone use defrag programs? 5 of 7
"Gus Altobello" 33 lines 12-MAY-1987 22:37
-< Is more information available? >-

I've taken note of all the preceding warnings concerning file id's, etc, and held off on procuring a disk defragmenter. But the arrival of our new system and SA482 disk stack have added to my feeling that I must do something about my disk defragmentation problems. Having no one to hang around at odd hours to backup/restore (and also since there's no good time to pull the disks offline on our system) I'm moving in the direction of evaluating one of these products.

The best product for our needs appears to be Diskeeper by Executive Software, both because of functionality and minimal size of the manual. I intend to run full offline backups of all disks to be used in the evaluation. But still I fear the entry of a future note here, detailing my demise.

Has anyone had any experience with this product, good or bad? How about any others? If the discussions here must remain vague concerning specific companies, could anyone with horror stories contact me directly?

I feel a defragmenter would be an asset to our system's performance, but am willing to be warned off if they all seem too dangerous...

Gus Altobello
PO Box 11274
Hauppauge, NY 11788
516/435-7036

Note 590.4 VAX 11/750 Word Processor 4 of 4
"Offline Submission" 14 lines 22-APR-1987 06:44
-< VAX 11/750 Word Processor >-

We are using Mass-11 on our 8 meg 750. Performance is good up to about 15-17 simultaneous users (with 15-20 doing other things). We use it for letters, program documentation, notes, everything. It has an EDT editor so you may have a shorter learning curve.

Bradley Sheppard
Coast to Coast
1000 16th Street
Washington DC 20036

Telephone: 202-293-8000

Date: April 15, 1987

Note 593.8 Memory disk driver going away??? 8 of 8
"George Walrod" 4 lines 4-MAY-1987 08:06
-< Don't Believe Everything You Read >-

Lets clear the rumor up, I talked to VMS development About the PDDRIVER going away. Their reply was an (I quote) "Are you kidding me, If any thing it will be improved".

George Walrod
4260-b chain bridge rd
fairfax, va 22030

```
================================================================
Note 599.6            Any PSI users out there?            6 of 6
"RICHARD HARTUNG"                    11 lines  26-MAY-1987 15:03
                      -< A PSI user >-
----------------------------------------------------------------
```

I use PSI for the classic use - connecting to a public network (TYMNET). We have found it to work very well. We have about 40 sites with up to 8 terminals on a concentrator and many more sites with one terminal and a modem to call a local TYMNET number. The only difference is on character echo, but the users get used to the application and just type away and look later.

I am going to try to use this for a DecNet link to some new sites we are putting in and would like to know of the pitfalls of this approach. Thanks.

RICHARD HARTUNG
AIRCO
575 MOUNTAIN AVE
MURRAY HILL NJ 07974
201-771-1246

```
================================================================
Note 603.11           Using TK50 for BACKUP            11 of 11
"ROBERT G. SIMPSON"                  14 lines  30-APR-1987 12:31
                -< TK50's red button/light >-
----------------------------------------------------------------
```

RE: .8 ... MOUNT gives error message ... "volume is not software enabled"

I have seen this problem several times and it was solved by cycling the red button out and back in a few times. I can only remember doing this when mounting the first tape, though, so I don't know if it works in situations as described.

RE: .5, etc.

It is very important to instruct operators not to touch the handle of the TK50 drive unless the red light is off. Lifting the handle while the tape is rewinding is fatal to the tape cartridge and the drive.

ROBERT G. SIMPSON

DRAVO CORPORATION, 300-12
ONE OLIVER PLAZA
PITTSBURGH, PA 15222

```
================================================================
Note 607.8    DECserver 200 application ports - how?     8 of 10
"Linwood Ferguson"                   27 lines  13-MAY-1987 17:22
          -< Accessing Server Ports - We do it >-
----------------------------------------------------------------
```

We also have Decserver 200 ports working as both incoming and outgoing dialup ports. They are somewhat flakey, however (occasionally hanging), and are not as robust at clearing modem problems as a DHU port (which cycles DTR periodically which for our Microcom modems tends to clear some weird states they occasionally get into).

Some notes: It also works on a Decserver 100, though you have to have a modem that can ignore all modem control lines, and you have to be prepared for all the security risks of having it not know when things are hung up. I mention it only to in case you want to test whatever you're doing on a different and simpler box.

Do not try async decnet over either a 200 or 100 port. It works, for all of about 2 minutes, then crashes the system. Dec says "no support". Too bad; we want to use Decservers instead of DHU/DMF/DHV's.

Other note: we have connected lots of foreign devices (scanners, device control systems) to Decserver ports, both 100 and 200's, and accessed it both with SET HOST/DTE, and directly from QIO's in programs- they all seem to work identically to DHU/DHV ports. No problems. Compliments to Dec on transparency. Now just fix it so I can tell what physical port I'm logged into!!!

Linwood Ferguson
MJ Systems, Inc.
P.O. Box 5223
2564 Ivy Road (22901)
Charlottesville, Va. 22905-0223
804/977-2732

Note 607.9 DECserver 200 application ports - how? 9 of 10
"Gus Altobello" 8 lines 14-MAY-1987 01:28
-< Re: 607.8 -- Async DECnet on a server >-

Linwood, get info from your DEC representative about the DECrouter 200. It's supposed to be specifically for server support of asynchronous DECnet.

We've got a couple on order, if there's interest (and it continues past the delivery date) I'll let y'all know how well they do their job.

Gus Altobello
PO Box 11274
Hauppauge, NY 11788
516/435-7036

Note 609.6 Utility to read/write IBM tapes 6 of 6
"Reece Pollack" 5 lines 18-MAY-1987 17:50
-< More VMS <=> MVS tape exchange >-

DEC published a set of VMS DCL and MVS JCL to do magtape transfer in the DSIN a while back; their JCL is real close, but still has a bug or two in it. One warning though, my MVS Systems Programmer tells me that MVS doesn't like to work with ANSI tapes with a blocksize greater than 2048, which is the ANSI spec maximum.

Reece Pollack
American Satellite Company
MS 34/MIS 1801 Research Blvd
Rockville, MD 20850

Note 611.15 LAVC Performance 15 of 17
"Linwood Ferguson" 98 lines 13-MAY-1987 15:56
-< LAVC Performance - Experiences >-

We've had some experiences, mostly negative, with LAVC's and performance. We still use them, but we've learned some things not to do.

We tried using a 750 as a boot node, with a MicroVAX added as a method of getting more performance out of an expensive 750 (i.e. with all the hardware it had on it). It failed miserably, but in a way that many applications will not experience. Our application uses lots of shared files, and we were unable to completely separate users and their files onto separate machines (though we managed it about 80%). The problem is that the lock manager in a cluster assigns the master of the lock to the first use to open a file. Lets say a particular file is used 5% by node A, and 95% by node B. If B opens the file first, everything is good. If A opens it first, ALL lock requests initiated on B must go over the link to A and be serviced there.

The behaviour we experienced was that the MicroVAX would run fine, while the 750 would spend anywhere from 10% to 95% (literally) of its time on the interrupt stack servicing the MicroVAX. The MicroVAX, due to its faster speed (including what I understand is a design "feature" of the 750 that when servicing requests over the LAVC link makes it run slower than the .65 MIP machine suppose to be) has the same problem, but to a far less degree. This resulted in overall significantly improved throughput, but completely unpredictable response, especially on the 750. Attempts to "direct" the locks to the right CPU through startup jobs that opened and held the locks worked to a large extent, but pose huge management problems (like what happens during a cluster transition when all locks are randomly reassigned, and what happens when you want exclusive access to a file, e.g. backup). We pulled it out of production.

The good news was that the actual transfer of data, when lock activity was at a minimum, was more than acceptable. We expected a drastic difference in performance between jobs accessing local disks and disks on the remote node, and the difference, while significant, was quite acceptable in an interactive environment.

Environments with little file sharing (like our development environment here) work quite well. Here the problem seems to be more actual servicing requests for data, disk files, etc., particularly of the operating system. While Dec may fix this when they allow LAVC's to boot from a separate system disk and join an existing cluster (or if? anyone know when?), you can do something to help now. For people using a few images or libraries a lot, this can make a significant difference:

SYS$SYSROOT is already a search list like:

```
MJS$DUA0:[SYS2.],SYS$COMMON:
```

Add to it, at the front (and remember the /SYSTEM/TRANSLATION=(CONCEALED,TERM) qualifiers) a directory on a local disk:

```
MV3$DUA0:[LOCAL.],MJS$DUA0:[SYS2.],SYS$COMMON:
```

Now put frequently used images, libraries, whatever in directories off of [LOCAL.], e.g. [LOCAL.SYSLIB], [LOCAL.SYSEXE], etc. If they are installed before, simply REPLACE them with INSTALL. This makes the access of frequently used images much faster, and more importantly puts a lower load on the boot node. Remember NOT to do this with data files needed to be updated simultaneously on all nodes, like SYSUAF or VMSMAIL.

Other performance notes: We changed LOCKDIRWT to try directing SCS directory functions to faster nodes. It probably worked, but we certainly couldn't tell any difference. We cannot find a SYSGEN way to do anything about the locking problem above. We did get better statistics, but no visible improvement by tuning the MSCP parameters for the satellite node, especially increasing the buffers on big MicroVAXes doing lots of work (they seem to be set more for VAXstation type sizes).

We'd appreciate any feedback anyone else has on performance Dec's documentation is a bit weak in this area.

Footnote: when the interrupt stack time is high, most of the time is in the drivers for the DEQNA/DELUA, not the lock manager code itself. Yet the SCS statistics show little data exchange, and the interrupt stack correlates perfectly with incoming and outgoing Dlock requests. This is definitely not a hardware error (unless its a design problem) as we've observed it at 3 sites (Dec was insistent that it had to be); no one has given a

good explanation of what it is actually doing, unless it is an error in the SPM reporting itself.

Linwood Ferguson
MJ Systems, Inc.
P.O. Box 5223
2564 Ivy Road (22901)
Charlottesville, Va. 22905-0223
804/977-2732

```
================================================================
Note 611.16                LAVC Performance                16 of 17
"M. Erik Husby"                          16 lines  14-MAY-1987 08:59
                        -< LAVC and DECType >-
----------------------------------------------------------------
```

Our experience with LAVC is minimal, essentially because we can't get DECType to run on it!! As the boot node has a large number of DECType users, this problem is hampering the production use of the LAVC. DEC's telephone support was more of a hinderance than a help. After 4 weeks of trying their suggestions (we could only run the LAVC night), none of which worked, we finally talked with the DECType Product Manager. Who told us that "DECType is not certified to run under a LAVC and we have no plans to do so". We got him to have some of his programmers to dial-in to our machine which they did the other night. They are now setting up a LAVC at their office to try and figure out what DECType could be doing wrong.

The problem seems to be with DECType's handling of print queues.

I will post the resolution of this problem as soon as I get it.

M. Erik Husby
Project Software & Development
14 Story St.
Cambridge, MA. 02138
(617)-661-1666

```
================================================================
Note 611.17                LAVC Performance                17 of 17
"Frank J. Nagy"                          6 lines  21-MAY-1987 07:17
                -< LAVC locking and VMS V4.6 >-
----------------------------------------------------------------
```

A Nashville attendee told me that one of the changes in VMS V4.6 is to change the way the lock dictionary is distributed in a cluster to prevent slow nodes from dragging everyone down by essentially weighing the algorithm such that locks will be mastered on faster nodes. This should help (significantly?) the LAVC locking problems when using 750s as reported in 611.15.

Frank J. Nagy
Fermilab
PO Box 500 MS/220
Batavia, IL 60510
(312)840-4935

```
================================================================
Note 613.11                ACL Problems                    11 of 11
"Ken A L Coar"                           21 lines  26-MAY-1987 14:12
        -< My experiments with VAXnotes protection >-
----------------------------------------------------------------
```

Back to the problem with VAXnotes conferences: After about an hour of experimenting, I concluded that VAXnotes creates a conference by getting the current default protection, removing DELETE access from all categories, and stuffing the result into a $XABPRO control block. Since this is stating an explicit protection, the DEFAULT_PROTECTION ACE is not applicable - it is only used when no protection code has been specified.

If my conclusion is correct (I experimented by changing the server login file to change its default protection, as well as changing my own and creating conferences directly), I think the VAXnotes development people should address this. As it is, the whole sequence of inherited protection features given us in VMS V4 is totally ignored, and the behaviour seen is identical with normal file operations under VMS V3.

As a side note: CREATE /DIRECTORY propagates the protection correctly, removing DELETE access. Maybe VAXnotes development should examine that code for pointers..?

#k

Ken A L Coar
General Dynamics
Office Systems
12101 Woodcrest Executive Drive
Creve Coeur, MO 63141
(314) 851.4003 (CST)

```
================================================================
Note 616.2         Experience w/RDB,RALLY,TEAMDATA?         2 of 4
"ROBERT G. SIMPSON"                      8 lines  30-APR-1987 12:16
                -< Rdb Runtime/TEAMDATA/Xway >-
----------------------------------------------------------------
```

We recently installed Rdb Runtime, TEAMDATA, and Xway on our 9MB MicroVAX II. For Rdb, there are 24 pages of known problems in the release notes manual and 27 pages in the online version, most of which did not apply to our runtime version. One user executing TEAMDATA slowed down the system until we adjusted our SYSGEN parameters. Xway provides conversion between various storage formats once you have the files on the VAX. Transfer to and from personal computers is not provided. Rdb/VMS V2.2 requires at least VMS V4.4.

ROBERT G. SIMPSON
DRAVO CORPORATION, 300-12
ONE OLIVER PLAZA
PITTSBURGH, PA 15222

```
================================================================
Note 616.3        Experience w/RDB,RALLY,TEAMDATA?        3 of 4
"Larry Kilgallen"                        2 lines   3-MAY-1987 07:04
                -< RE: 616.2 - What is Xway? >-
----------------------------------------------------------------
```

What is the full name, and if not DEC, who is the vendor? Inquiring minds want to know.

Larry Kilgallen
Box 81, MIT Station
Cambridge, MA 02139-0901

```
================================================================
Note 616.4        Experience w/RDB,RALLY,TEAMDATA?        4 of 4
"Jack Patteeuw"                          7 lines  19-MAY-1987 10:29
                    -< A DEC Product ! >-
----------------------------------------------------------------
```

The proper name is "VAX Xway". The part number is Q*729 and the SPD is 27.36.00.

I have never used the product or read the documentation but if recall the sale blurb it is a package to allow the transfer of Lotus 1-2-3 spread sheet (on your PC) to and from DECalc and DECalc+.

Jack Patteeuw
Ford Motor Co.
Electrical and Electronics Division
31630 Wyoming
Livonia, MI 48150
313-323-8643


```
================================================================
Note 619.4                unwanted formfeed               4 of 10
"Richard Herdell"                        24 lines   6-MAY-1987 14:27
                  -< LN03 Extra Page Fix >-
----------------------------------------------------------------
```

Our LN03 plagued us with a formfeed after every print job was finished until the device control module specified during queue initialization was modified.

This module, in our case, PORTRAIT.TXT, had an <ESC>c in it. I believe this causes a hardware reset of the LN03. This module was modified to <FF><ESC>c, replaced into SYS$LIBRARY:SYSDEVCTL.TLB and the extra page went away. I'm not sure why this works but it does, at least in this case.

The queue set up is as follows:

```
INITIALIZE/QUEUE/DEFAULT=(NOFLAG,NOFEED)/FORM=20 -
 /SEPARATE=(RESET=PORTRAIT)/START TXXX
```

The form setup is as follows:

```
DEFINE/FORM/NOTRUNCATE/NOWRAP/WIDTH=80 -
 /STOCK=DEFAULT/LENGTH=66/SETUP=PORTRAIT -
 /MARGIN=(TOP=0,BOT=0)  PORTRAIT 20
```

Richard Herdell
7000 Hollister
Houston, TX 77040

```
================================================================
Note 619.5                unwanted formfeed               5 of 10
"Reece Pollack"                          10 lines   6-MAY-1987 19:10
              -< Print Symbiont Documentation >-
----------------------------------------------------------------
```

I really hate to scream RTFM, but try reading the VAX/VMS Utility Routines Reference Manual, Chapter 9 -- Print Symbiont Modification (PSM) Routines. It explains in detail why and when the print symbiont issues form feeds. I'm not claiming that this chapter is easy to read, but this stuff is documented.

Basically, the symbiont will issue form feeds anytime it thinks it isn't at the top of a page when it wants to be, and this includes after page and form setup modules which contain what it thinks is printable text.

Reece Pollack
American Satellite Company
MS 34/MIS 1801 Research Blvd
Rockville, MD 20850

```
================================================================
Note 619.6                unwanted formfeed                6 of 10
"Bob Hassinger"                          25 lines   7-MAY-1987 09:44
            -< Obscure documentation locations... >-
----------------------------------------------------------------
```

> I really hate to scream RTFM, but try reading the VAX/VMS Utility Routines Reference Manual, Chapter 9 -- Print Symbiont Modification (PSM) Routines. It explains in detail why and when the print symbiont issues form feeds. I'm not claiming that this chapter is easy to read, but this stuff is documented.

Readability aside, this comment points out a problem that keeps coming up for me in the VMS documentation. Why would the average system manager who is just interested in setting up normal print queues ever think to plow through all the PSM stuff in volume 8A looking for clues. All the device control library references in the master index refer to chapter 9 of the System Manager's Reference Manual which is currently in volume 5A. I would suspect that would be where the average manager discovers there is such a thing as a device control library. I doubt the average person is likely to go off exploring all through more or less obscure material in remote locations on the chance it might turn up something.

In the past I have had the same kind of problem with RMS. They used to expect you to muck around in all the macro level FAB and RAB stuff to discover basic things about the file system that have nothing to do with macro level programming.

Bob Hassinger
Liberty Mutual Research Center
71 Frankland Road
Hopkinton, MA 01748
617-435-9061


```
================================================================
Note 619.7                unwanted formfeed                7 of 10
"CHUCK MCMICHAEL"                         7 lines  12-MAY-1987 15:55
            -< meanwhile, back at the VAX... >-
----------------------------------------------------------------
```

RE: 619.5 I have read Chapter 9, as a matter of fact. However, as far as I know, I'm not using any page or form setup modules. I'm simply issuing a START/QUEUE command and letting the defaults fall where they may. DEC was unable to come up with anything helpful when I asked around at the DECUS symposium last week. I'm still open to suggestions.

CHUCK MCMICHAEL
PITTSBURGH CORNING CORP
800 PRESQUE ISLE DR
PITTSBURGH PA 15239
412-327-6100

```
================================================================
Note 619.8                unwanted formfeed                8 of 10
"Reece Pollack"                          17 lines  18-MAY-1987 18:15
                -< FormFeed? What FormFeed? >-
----------------------------------------------------------------
```

If I remember right, we were trying to eliminate the form-feed issued when the queue is first started. This FF is issued so that the flag and/or burst page(s) will be aligned right, since they assume they are already at the top of a page. The correct way to eliminate this FF is to write a customized print symbiont which supplies a modified JOB_SETUP routine which does all of the functions of the DEC-supplied routine, minus the initial form-feed.

For a real get-down-and-dirty quick fix, you could make a copy of the standard print symbiont sharable image and patch it, but this means getting a fiche license in order to keep up with the updates.

By the way, I agree that a lot of the good tidbits are spread all over the documentation set, mostly buried pretty deeply, but I can't really suggest a better organization. The manuals for our IBM MVS system fill a small room (rather than a small bookcase) and expect even more from the reader than the VAX manuals...

Reece Pollack
American Satellite Company
MS 34/MIS 1801 Research Blvd
Rockville, MD 20850

Note 619.9 unwanted formfeed 9 of 10
"Jamie Hanrahan" 16 lines 18-MAY-1987 20:22
-< before writing a symbiont, try this >-

Here is an easy way to get rid of the unwanted form feed caused by non-DEC-standard escape sequences (or anything else the print symbiont thinks will mess up the top-of-form position) in setup modules:

Put an <ESC><DCS> in front of the offending escape sequence(s), and an <ST> at the end. <DCS> is 90 hex; <ST> is 9C hex. This effectively encloses your stuff in quotation marks; the print symbiont will pass it through intact (with the <ESC><DCS> and <ST>, I believe), and otherwise ignore it.

This advice is from one of the DEC folks at the VMS Advanced Q&A in Nashville. (See what you miss by flying out Friday afternoon?) I haven't tried it myself. I don't know what to do if your device does something "interesting" in response to <ESC><DCS> .

Jamie Hanrahan
Simpact Associates
9210 Sky Park Court
San Diego, CA 92123
619-565-1865

Note 624.1 warning on Monitor Ethernet command 1 of 2
"Keith Roberts" 13 lines 13-MAY-1987 14:09
-< No problem on our 750 or MVII >-

> I tried the patching suggested in the April 1987 Pageswapper article "Undocumented Monitor Display Classes" and managed to crash two of my VAXes.

I Did the patch to monitor under 4.5 on both my 750 and MicroVAX II...No Problem. Also, I patched it on the MicroVAX boot member at the Digital exhibit hall [Nashville] (running with 5 2000's). I ran it for at least a 45 minutes. It was interesting, the boot member saw a max of 191k bytes/second...average was about 60k bytes.

We have ordered hardware/software for a LAVC which should be in in about a month, It was interesting to see how busy the ethernet was.

Keith Roberts
SASC Technologies Inc
109 Mass Avenue
Lexington, Ma 02173
(617) 377-5959

Note 624.2 warning on Monitor Ethernet command 2 of 2
"Kevin Angley" 3 lines 26-MAY-1987 10:56
-< Another vote of confidence >-

Regarding MON ETHERNET patch, I have had no problem with it on a 8300. It is NOT particularly useful in my opinion, but fun to watch.

Kevin Angley
3301 Terminal Drive
Raleigh, NC 27604
(919) 890-1416

Note 625.4 Are there TU81+'s that work? 4 of 6
"Stuart Renes" 5 lines 28-APR-1987 18:08
-< proper MOUNT req'd >-

I presume that you ENABLED the tape data cache with the MOUNT command (you can see this with SHO DEVICE/FULL MUA0:)

.a simple, but easily overlooked issue....

Stuart Renes
AT&T Technologies, Inc.
Mail Stop: 2793
3000 Skyline Dr.
Mesquite, TX 75149
(214) 288-2286

Note 625.5 Are there TU81+'s that work? 5 of 6
"Linwood Ferguson" 14 lines 13-MAY-1987 16:16
-< Fast-yes, Work-Not often >-

We have TU81+'s installed on 750's, 785's, and an 8500. There is an incredible difference in performance on the 8500, apparently due to CPU speed (though I cannot swear it is not the BI, we have no reason to think our Unibus is overloaded, particular on the 785's).

As to the original question "TU81+'s that work", we've had lots of trouble with then, apparently hardware trouble, but I'm beginning to wonder. Out of about 19 of them, over half have had long (2-7 day) downtime, most repeatedly. While Dec usually "finds the problem", it almost always seem to come back. Usually it is in the form of unexplained "fatal controller errors". Anyone have similar experiences?

Linwood Ferguson
MJ Systems, Inc.
P.O. Box 5223
2564 Ivy Road (22901)
Charlottesville, Va. 22905-0223
804/977-2732

PAGESWAPPER - July 1987 - Volume 8 Number 12
INPUT/OUTPUT

Note 625.6 Are there TU81+'s that work? 6 of 6
"Gus Altobello" 23 lines 14-MAY-1987 01:20
-< TU81+: Perhaps not hopeless... >-

Our TU81+ was generating about 300-500 errors per day, most of the vague variety. DEC comes in, pokes around, and often changes the head (yes, we've head the head changed twice, if I recall correctly).

Vaxsim complains mightily about this device, and the DEC field tech once suggested we just remove Vaxsim. Other sites with TU81+'s have had the same problem. DEC tells the field guys to install Vaxsim, and when they do, the customer calls constantly because of the high reported error rate.

One thing that seemed to help was an adjustment our DEC Tech tried, which he'd never tried before. Seems these drives can be equalized for particular tape types. You put the controller into a special mode, mount your favorite flavor of tape, and it runs for awhile. When done, the drive is equalized to work best with that tape.

Since the last head change and equalization (and whatever other pokings the tech did) our errors are down to about 30-50 a day (quite an improvement).

Gus Altobello
PO Box 11274
Hauppauge, NY 11788
516/435-7036

```
================================================================
Note 626.5                Backups and Allin1                5 of 6
"Mark Hyde"                                 4 lines  23-APR-1987 08:13
         -< Crucial to ALL-IN-1? I'd start with... >-
----------------------------------------------------------------
```

All of OA$DATA. All of a user'sU ALL-IN-1 subdirectory structure [.A1...]

mark

Mark Hyde
Digital Equipment Corp
360 Interstate North Parkway
Suite 600 IPO1-6/C2
Atlanta, Ga 30339

```
================================================================
Note 626.6                Backups and Allin1                6 of 6
"ROBERT G. SIMPSON"                         7 lines  30-APR-1987 12:45
                  -< Re: Crucial to All-in-1 >-
----------------------------------------------------------------
```

Re: Crucial to All-In-1

In addition to OA$DATA, all of the OA$SHAREn directories. E-mail messages are moved to these directories after they have been sent. We have successfully restored OA$DATA, OA$SHAREn, and all user subdirectory structures [.OA...] from backups taken when All-In-1 was down.

ROBERT G. SIMPSON
DRAVO CORPORATION, 300-12
ONE OLIVER PLAZA
PITTSBURGH, PA 15222


```
================================================================
Note 634.1      NETSERVERS: a question and a warning      1 of 2
"Jack Patteeuw"                             10 lines  28-APR-1987 16:32
                    -< Another warning >-
----------------------------------------------------------------
```

A lot of my users have recently discovered the $ SET PROCESS/NAME command and now love to put cute saying out there from their LOGIN.COM file. This however causes an interesting problem.

If the user is logged in (BATCH or INTERACTIVE) and a NETWORK session is started, it will abort because the network session will try to set the process name to the same thing which is a "no-no".

The solution I used was to put a $ON WARNING THEN CONTINUE at the beginning of the users LOGIN.COM.

Jack Patteeuw
Ford Motor Co.
Electrical and Electronics Division
31630 Wyoming
Livonia, MI 48150
313-323-8643

```
================================================================
Note 634.2      NETSERVERS: a question and a warning      2 of 2
"Michael R. Pizolato"                       14 lines  29-APR-1987 09:57
                    -< A way around it. >-
----------------------------------------------------------------
```

I like using "cute" names for my processes. I got around the duplicate name problem using two things:

1. A command procedure that selects, "randomly," one of ten process names, trying again if it picks one that was already picked, and giving up after 10 unsuccessful tries.

2. Using IF F$MODE() .EQS. "INTERACTIVE" THEN @PROC_NAME to only allow the command procedure to run for interactive logins.

This works fine for me. I did spend a lot of time coming up with good names, though :-).

Michael R. Pizolato
AT&T Technology Systems
Dept. 323610
555 Union Blvd.
Allentown, PA 18103
215/439-5500

Note 635.1 CALLABLE EDITORS IN MAIL 1 of 1
"Lorin M. Ricker" 18 lines 21-APR-1987 00:28
-< Callable Editors from almost Anywhere >-

You're right about that trick with MAIL, and it works great and is a big saver. Analogous/similar tricks are now working with other DEC tools; either a logical name (like RDB$EDIT or RDO$EDIT (excuse my flakey memory)) or a command (there's one in the Symbolic Debugger) turn on the CALLABLE_TPU or _EDT interfaces, and you can take advantage of your favorite editor.

TPU-hackers, note: by using the TPUSECINI logical name, you can get your own personalized TPU section file fired up, be it EVE, quasi-EDT, or a home-brew. Further note for TPU-hackers: in the case of a home-brew, be aware that, after the first entry into your TPU-section, the TPU$INITIALIZE procedure is NOT called again (on subsequent re-entries from the "parent" program (Debugger, MAIL, RDO...)) -- you must be careful about assumptions pertaining to editor state, global TPU variable values, etc. upon re-entry... usually, you'll need to look at exit-procedures with an eye to TPU's next (potential) call.

Lorin M. Ricker
EVANS & RICKER, Inc.
7062 N. Cambridge Ave.
Portland, OR 97203
(503)289-3709 / (503)682-0179

Note 638.1 TK50 Transfer Problem 1 of 1
"Lorin M. Ricker" 11 lines 21-APR-1987 00:18
-< TK50 Transfer Problem >-

---

hmmm... seems like it should work: have you tried this? Initialize the tape on the 11/23 under RT-11 (creating an RT directory structure), then write files to it (COPY). You should then be able to mount the tape on the MicroVAX (MOUNT/FOREIGN), and then EXCHANGE should be able to get at the files (MOUNT/VIR MUA0:) as logical disks. This has worked for us without problems on console media (CSA1:, with floppies on 780's and TU58's (slowly!) on 750's), but I've not had the opportunity to try it with TK50's. No reason why it won't work tho'... My memory of FILEX isn't very good, so I'll leave it to someone else to suggest a "symmetric" VAX-->RT-11 approach.

Lorin M. Ricker
EVANS & RICKER, Inc.
7062 N. Cambridge Ave.
Portland, OR 97203
(503)289-3709 / (503)682-0179

Note 639.1 TSV05 at 100 ips. 1 of 2
"Mark Hartman" 7 lines 1-MAY-1987 19:52
-< Shoot it... >-

The DEC controller is, as you have noted, simply too slow (as is the Emulex TC02).

While I will not advocate (here) any other manufacturer, our experience with the DEC version of the Cipher F880 (that is, a TSV05) leads us to consider it unacceptable.

Mark Hartman
Jadtec Computer Group
546 W Katella Ave
Orange CA 92667

Note 639.2 TSV05 at 100 ips. 2 of 2
"Linwood Ferguson" 8 lines 13-MAY-1987 17:02
-< What TSV05 Version? >-

> TSV05 doesn't run at 100ips

Have you tried version 1.1 of the TSV05 driver? It had comments that made it sound like it improved performance. Unfortunately, we have not had access to one yet to try it (they are all on client machines).

Linwood Ferguson
MJ Systems, Inc.
P.O. Box 5223
2564 Ivy Road (22901)
Charlottesville, Va. 22905-0223
804/977-2732

Note 641.0 Anyone Using Cortex Appl. Factory No replies
"MICHAEL GRATTAN" 7 lines 21-APR-1987 14:08

We are converting from an old Datapoint network to Dec. We are going to use the Application Factory from Cortex to rewrite our applications. I am curious to know if there is anyone else who is running the Factory. I would like to know what kind of cpu you are using, how many terminals/users you have and any particular performance problems (and hopefully fixes) you have had. Thank you.

MICHAEL GRATTAN
FAIRHAVEN CORP.
358 BELLEVILLE AVE.
NEW BEDFORD, MA. 02742
617-993-9981 EXT 106

PAGESWAPPER - July 1987 - Volume 8 Number 12
INPUT/OUTPUT

Note 642.0 print&batch complaints 3 replies
"John Osudar" 28 lines 22-APR-1987 20:18

As long as we're complaining about the queue manager/job controller/symbionts (e.g. note 630)... some thoughts, complaints, etc. about how VMS V4 print (and batch) job handling works: Has anyone else found that "page setup modules" (used in defining forms) don't quite work right? I tried defining a form using a page setup module, and found that it (a) caused some of the job setup modules (/SETUP in the form definition) to be skipped, and (b) repeated some page setup modules multiple times per page. Also: does it bother anyone else that you can define "job setup modules" and "page setup modules" ONLY in a form definition, and you can specify "file setup modules" ONLY in a PRINT command? We have a laser printer for which we spent real $$$ to get our letterhead digitized. A typical document would like the letterhead called up on the first page of a file. This can be done just file with PRINT/SETUP, but you can't define a form to do it -- putting the stuff in the form's /SETUP list causes the job flag page to come out on letterhead. Wouldn't it make sense to have the same setup features available both in a form definition and on the PRINT command? While we're talking about forms definitions: why does DEFINE/FORM have defaults such as /MARGIN=(BOTTOM=6) and /TRUNCATE, while the SHOW QUEUE/FORM/FULL command "neglects" to list the MARGIN qualifier if BOTTOM=0, and the TRUNCATE qualifier if it's set /NOtruncate? Seems a little inconsistent...

Finally: would anyone else find benefit from enhanced batch job scheduling (we finally got print job scheduling by size in VMS V4, but there isn't much available for batch jobs) -- or, my preference, a "batch symbiont" that would allow similar flexibility in processing batch queues as that presently available with "output" queues?

John Osudar
Argonne National Laboratory
9700 S. Cass Ave.
Bldg. 205 A-051
Argonne, IL 60439-4837
(312) 972-7505

```
================================================================
Note 642.1              print&batch complaints            1 of 3
"Bruce Bowler"                          21 lines  27-APR-1987 13:51
                         -< I agree!! >-
----------------------------------------------------------------
```

Regarding improved batch handling, you bet we would like to see improved batch handling (even if it only involved a better queue selection scheme based on cpu availability, rather than queue slot availability). There currently is a CPU rating algorithm used by the terminals server products (not all that robust an algorithm but ...).

It would seem to me that DEC should develop a uniform and consistent rating system for CPU availability that could/would be used by products that have a reasonable need to know CPU load for scheduling purposes, these products would include terminal servers, batch queues, and I am sure others that will come down the pipe in the future. One of my big complaints about the algorithm currently used by the terminal server is that a large compute-bound job running at lower than normal interactive priority (say 0 or 1 vs. 4) can drive a CPU's rating to 1 very fast, when in reality it should not do so. There for some method for determining what is chewing up the CPU time needs to be added to the algorithm so that it can behave in an intelligent manner.

There was a program on a DECUS tape a while ago called ZEUS that attempted to do this (reasonably well I might add) that DEC should scrutinize for ideas on how to proceed.

Bruce Bowler
General Electric
1 River Road
Bldg 2 Room 609
Schenectady, NY 12345


```
================================================================
Note 642.2              print&batch complaints            2 of 3
"Larry Kilgallen"                        4 lines  2-MAY-1987 21:51
          -< CPU statistics are insufficient >-
----------------------------------------------------------------
```

At present, VMS does not keep statistics on CPU utilization according to the priority at which the utilizing process was running. Changing that collection would logically not come before the next major release of VMS.

Larry Kilgallen
Box 81, MIT Station
Cambridge, MA 02139-0901

```
================================================================
Note 642.3              print&batch complaints            3 of 3
"Bruce Bowler"                          14 lines  4-MAY-1987 07:03
                -< Not (necessarily) true >-
----------------------------------------------------------------
```

Larry, I am aware that VMS does not keep CPU statistics according to priority, and while I must admit I am not a complete guru at internals it would seem to me that one solution would be to have some process (JOBCTL comes to mind) do something at DEFPRI that takes a known number of clock cycles, timing itself while it does this. Then taking some measure of expected clock cycles versus wall time multiplied by some CPU rating factor, a reasonable measure of CPU availability results. It is not my expectation that this would be a constantly running process, but rather an event driven or clock driven action with a user selectable period.

I still see that this would not/could not come before the next release but I think that it is a reasonable thing to ask DEC to do for the user.

Bruce Bowler
General Electric
1 River Road
Bldg 2 Room 609
Schenectady, NY 12345

Note 643.0 Virtual Disks in DecNet-DOS= 1 reply
"DEREK FIELDS" 14 lines 24-APR-1987 07:11

Is anyone doing extensive work with DecNet-DOS. We are installing an Ethernet network using IBM PCs as terminal emulators. We are also looking in the Virtual Disk Utility. It seems that this utility could solve a major corporate headache:

We have 100 users (for example) and 20 copies of Lotus 1-2-3. We want to allow all 100 users access, but only 20 at a time to stay within the limits of our license agreement. The solution that occurs to us is to set up either 1 Virtual Disk with 1-2-3 on it and allow 20 (but not 21) users to access it or to set up 20 Virtual Disks and allow only one user at a time to use it. The problem with either solution is how to monitor access to the Virtual Disk, since no task is created on the host when the Virtual Disk is attached to the local PC. Any ideas?

DEREK FIELDS
NEW JERSEY BOARD OF PUBLIC UTILITIES
1100 RAYMOND BLVD
NEWARK, NJ 07102
201-648-2417

Note 643.1 Virtual Disks in DecNet-DOS= 1 of 1
"M. Erik Husby" 8 lines 27-APR-1987 11:41
-< DECnet DOS virtual disks and FAL >-

You will get a task on the host when you access a virtual disk.

That task will be running FAL. So that is where you can control your access. You should only need one virtual disk. Study the SYS$SYSTEM:NETSERVER.COM for ideas on how to keep track of how many are accessing the virtual disk. You should be able to do something similar to NETSERVER.COM's permanent servers.

M. Erik Husby
Project Software & Development
14 Story St.
Cambridge, MA. 02138
(617)-661-1666

Note 644.0 Wanted: MAIL improvements 9 replies
"John Osudar" 43 lines 28-APR-1987 17:47

Just about every VMS user uses VMS MAIL. VMS MAIL isn't bad; DEC has put some work into "improving" it, especially between V3 and V4, and it works reasonably well as an electronic mail system. (Not bad for an "exercise to learn VAX assembler coding"!) So what am I complaining about? Well, for starters, here's a small list:

(1) Return receipts!!! Most reasonable mail systems (and even some unreasonable ones) provide for return receipts. DEC claims "too many users have complained that this would be an invasion of their privacy", so they won't do it. First of all, I'd like to meet those users -- everyone I've ever asked about this feature has liked it. Second, since DEC likes making everything optional, it would be practical, sensible, useful, and entirely acceptable to allow a user to say "SET NORECEIPTS" or something like that, setting a flag in the user's MAIL profile so that the sender gets told "User XXX does not produce return receipts". (Then I can go talk to XXX's supervisor and find out why XXX doesn't want people to know if he read their mail...) My personal opinion is that DEC doesn't want to add this (or too many other useful features) to a product that we get free with VMS, which would make it competitive with (better than?) similar products that cost real bucks.

(2) Add other features that "reasonable" mail systems have -- CC: lists, for example.

(3) Document and support the foreign protocol interface. This feature is in common use by a number of (free and commercial) products, and some of us could use it to replace our old kludgy code (which makes use of the "Mail-11" network protocol) that talks to non-DECnet networks and various foreign mail systems.

(4) Fix deficiencies in the "improvements" that came along with VMS V4. For example, provide a way to delete an entire folder, or even all the folders in an entire mail file, along with all the associated MAIL$*.MAI files, without wasting time moving things into wastebaskets, or otherwise shuffling around things that you're really trying to get rid of. I've spent many long hours waiting for MAIL to finish deleting all the messages in each of the folders of a mail file, when all I wanted was to delete the entire file and all associated MAIL$*.MAI files.

I'm sure other users have other complaints/requests; I hope those will be added to this note. It seems to me that these are reasonable requests. (Some other things I've heard requested, such as having VMS MAIL do store-and-forward, are more reasonably offered in an extra-cost product. Of course, one of us users might come up with a free version one of these days...)

John Osudar
Argonne National Laboratory
9700 S. Cass Ave.
Bldg. 205 A-051
Argonne, IL 60439-4837
(312) 972-7505

```
================================================================
Note 644.1          Wanted: MAIL improvements           1 of 9
"MICHAEL GRATTAN"                    11 lines  29-APR-1987 06:18
                -< While we're complaining >-
----------------------------------------------------------------
```

We're converting from a system that doesn't even have anything like MAIL, so we're just thrilled to pieces that we finally a useful tool. However, I agree with you whole heartedly re:return receipts. I know that some of our users will ask why that isn't there when we introduce them to this system.

One thing that I would like to see is an 'address list' of valid accounts that you could send mail to. The system manager (or some such person) could flag accounts that shouldn't get mail. In this way you wouldn't have to know that Fred's accountNY is called FOOBAR or whatever bie name it has.

MICHAEL GRATTAN

FAIRHAVEN CORP.
358 BELLEVILLE AVE.
NEW BEDFORD, MA. 02742
617-993-9981 EXT 106

```
================================================================
Note 644.2          Wanted: MAIL improvements           2 of 9
"David Shepperd"                     15 lines  29-APR-1987 20:36
                   -< More on MAIL >-
----------------------------------------------------------------
```

I agree that MAIL could be "fixed". At our facility it is the most visible VMS product on the cluster, yet it doesn't have the friendliness that such a program should have.

Re the CC: lists, you can do that...well sort of. MAIL reads each token on the "To:" line and tries to translate it as a logical name. If that fails, it uses it as it is. If it succeeds, then the text of the translation replaces the text and the process continues. So, if you $DEFINE CC " " for example (note the space), then you can place a CC::STAFF on your "To:" line and MAIL will replace the CC with a space. Likewise, STAFF could be a logical name defined as a list of people ($DEFINE STAFF "TOM,DICK,NODE::HARRY") or as a pointer to a distribution list ($DEFINE STAFF "@SYS_MAIL:STAFF"). This can also bite you if you try to MAIL messages to people who have names that match any of your local or system logicals.

David Shepperd
Atari Games Inc
675 Sycamore
PO BOX 361110
Milpitas, Ca 95035-1110
(408) 434-1711

Note 644.3 Wanted: MAIL improvements 3 of 9
"John Osudar" 12 lines 30-APR-1987 15:49
-< ...but I don't want to fix MAIL myself! >-

I agree that there are ways to get "kind-of sort-of" CC-lists. (For that matter, products that sit on top of VMS MAIL, such as Microsystems' Mass-11 MAIL, implement return receipts via a kludge using the subject field!) That, however, is not the issue I wanted to raise. DEC's MAIL-11 protocol includes a flag indicating "whether the system accepts CC-lines" -- so obviously someone thought about including them, and didn't do it. A GOOD mail facility will distinguish between the recipients and the CC-list, so that you can tell if the message was addressed to you, or if you were included only on the CC-list. (For things like meeting notices, this can be important!) The whole point is, DEC could do it, DEC should do it, and we should tell DEC that we WANT them to do it.

John Osudar
Argonne National Laboratory
9700 S. Cass Ave.
Bldg. 205 A-051
Argonne, IL 60439-4837
(312) 972-7505

Note 644.4 Wanted: MAIL improvements 4 of 9
"Larry Kilgallen" 10 lines 2-MAY-1987 21:58
-< I disagree >-

I would prefer to keep the more complex mail features like return receipts in the layered products, so that money spent on VMS development will go toward things not presently offered at all, and so DEC will not have to raise the price of base VMS software maintenance so much. I don't want to play silly bureaucratic games with a recipient over whether they have read their mail (given privilege I could always check their unread mail count anyway), what matters to me is whether they have ACTED on it. I think fancy features should be paid for by those who want them, not by all of us.

Larry Kilgallen

Box 81, MIT Station
Cambridge, MA 02139-0901

Note 644.5 Wanted: MAIL improvements 5 of 9
"John Osudar" 11 lines 2-MAY-1987 23:05
-< I agree and disagree >-

Larry, I agree with you -- up to a point. If it were truly the case that doing without some features would keep prices down, give DEC more time to do REALLY useful things, etc. I would shut up. The only problem is that we still seem to pay for "fancy features" -- only they are ones that DEC cooks up, or a small special interest within the user community demands, and not those that a fairly large number of users really want. An alternative would be to document things well enough so that an enterprising user could implement enhancements (and possibly give them to the Sigtape?) But that doesn't seem to happen too often either. Anyway, that's enough ranting and raving (from me, at least...)

John Osudar
Argonne National Laboratory
9700 S. Cass Ave.
Bldg. 205 A-051
Argonne, IL 60439-4837
(312) 972-7505

Note 644.6 Wanted: MAIL improvements 6 of 9
"Larry Kilgallen" 16 lines 3-MAY-1987 07:12
-< Callable mail would help people "roll their own" >-

Note that Monday's "VMS Futures" presentation at the Spring 1987 (Nashville) Symposium mentioned "callable mail" as a possible feature of a future major release of VMS whose version number could not be mentioned by DEC employees at the symposium.

(For those fairly new to reading between the lines, consider that the possibility of having such a feature in a version of VMS which DEC folk are forbidden to name is a lot better than the possibility of having it in a version of VMS so far out that DEC folks couldn't even guess what the name would be.) The clear

implication was that there would be more information at the Fall 1987 Symposium in Anaheim, so start working on your boss now to get your trip planned (not that the DECUS Symposium Committee really need more people attending).

Larry Kilgallen
Box 81, MIT Station
Cambridge, MA 02139-0901

Note 644.7 Wanted: MAIL improvements 7 of 9
"Bob Hassinger" 27 lines 3-MAY-1987 11:18
-< Too much for the wrong features... >-

Larry, I do not think return receipts and maybe automatic copies to the sender of forwarded (and annotated) messages are all that "fancy" for a tool so widely used. In any case, the problem here is that every layered product for VMS tends to cost a minimum of what something like the FORTRAN compiler costs on the same system! A few minimal features added to MAIL simply will not justify such a cost here. Also, the examples I have seen of DEC's ideas in this area are not very good. We end up spending a lot of money for a product that goes far beyond what we need, at a cost much greater than we can pay and in the process we loose the basic flavor of VMS MAIL which is what makes it so popular and usable across a very wide user base.

I would have gladly traded many of the "improvements" in V4 MAIL for a couple of these features that have been requested for many years.

Judging from everything I have heard on the subject from DEC people and others at Symposia and elsewhere it seems clear this IS a case of DEC protecting layered products at some level. If the layered products where good and affordable for my needs that might be OK but they have yet to get it right as far as I am concerned. I think it is this perception that keeps the requests coming for putting these features in VAX MAIL.

Bob H

Bob Hassinger
Liberty Mutual Research Center
71 Frankland Road
Hopkinton, MA 01748

617-435-9061

Note 644.8 Wanted: MAIL improvements 8 of 9
"Larry Kilgallen" 37 lines 4-MAY-1987 06:26
-< Let DEC work on NEW features >-

The trade-off of a finite budget to make only certain changes is exactly what is supposed to be modeled by the SIR process. The last Mail-related SIR I recall to make the big time (top ten) was a laundry list of everything which could be imagined, such that everyone with any feelings about mail would vote in favor of at least one of them. In total, however, DEC would end up re-implementing the message router in VMS, which just HAS to cost money to DEC, which means eventually to us.

If VMSmail were to contain a store-and-forward feature, I think it should be the ability to work in better harmony with the Message Router. The message should go directly (read efficiently) ala VMSmail if the target node is up, and go via store-and-forward if the target node is down or if store-and-forward is specifically requested (for a quick return to DCL). This adaptive transmission is not available at all today, the message goes either one way or the other. Since there IS a way (albeit with cost) to get store-and-forward for VMSmail from DEC today, let the VMS developers concentrate on those aspects of mail which

a) Only VMS can provide

and

b) Are not available AT ALL today

I think callable mail is a perfect example. Let us presume that I do not like the present user interface (I don't have a complaint right now, but under pressure I could certainly come up with one). I think providing me (and you, and you) with the ability to roll my own is highly preferable to providing full-blown alternate interfaces. Why should DEC work on an ugly FMS-based interface for you when they could be working on a beautiful TDMS-based interface for me (or the other way around).

Larry Kilgallen
Box 81, MIT Station

Cambridge, MA 02139-0901

```
================================================================
Note 644.9          Wanted: MAIL improvements            9 of 9
"John Osudar"                        38 lines  5-MAY-1987 17:53
          -< Let DEC work on USEFUL new features >-
----------------------------------------------------------------
```

First: callable MAIL sounds like an excellent idea. I look forward seeing it when my VMS V<future release>.0 kit comes...

Second: I hate repeating myself, but as I said earlier...

- new features whose complexity is at the "message router" level (e.g. store-and-forward) OUGHT to be extra-cost products

- certain features that are useful, should not require a great deal of effort to implement, and have been requested by many users (e.g. return receipts, CC-lists) SHOULD be implemented in VMS MAIL. (I have yet to hear anyone from DEC say "return receipts will take too much of our time and effort to implement"; instead they say, in effect, "you don't REALLY want return receipts, so we won't do them". Guess what -- I didn't REALLY want mail folders, but they did them anyway...)

- Making the product user-tailorable, and documenting the "back doors" that can be used by user-developed software (e.g. foreign protocol interface) is an acceptable way to give us some of what we need. (Callable MAIL fits in nicely here.)

In fact, I am strongly in favor of having user-callable interfaces for as many VMS components as is practical. However, I think we need to be very, VERY careful not to allow DEC to give us software that has too little standard functionality with the excuse that "the user can roll his own" via the callable interface. My implementation of return receipts via callable MAIL may look nothing like, and be totally incompatible with, your implementation -- and if our systems are linked by a common network, that can be a serious problem. There IS room for DEC to enhance existing software, without expending an excessive amount of effort (or development money); but we, the users, must

agree on the features we need most, and communicate that information to DEC.

John Osudar
Argonne National Laboratory
9700 S. Cass Ave.
Bldg. 205 A-051
Argonne, IL 60439-4837
(312) 972-7505

```
================================================================
Note 645.0                Errors on RD53                 1 reply
"MICHAEL GRATTAN"                      7 lines  29-APR-1987 08:03
----------------------------------------------------------------
```

We have a uVAX II running VMS. Every once in a while I'll get an error on the RD53 disk. When I look in the error log it says something about "SEQUENCE NUMBER RESET - OPERATION SUCCESSFUL". I have asked field service and they are vague although they say not to worry. But still... Does anyone know what this error means? Since this is my system disk, I am concern about having to replace it. Would greatly appreciate ant information.

MICHAEL GRATTAN
FAIRHAVEN CORP.
358 BELLEVILLE AVE.
NEW BEDFORD, MA. 02742
617-993-9981 EXT 106

```
================================================================
Note 645.1                Errors on RD53                 1 of 1
"Mark Hartman"                         14 lines  1-MAY-1987 19:44
                    -< Packet hiccups... >-
----------------------------------------------------------------
```

Sometimes the MSCP protocol gets confused due to noise, imperfections, or cosmic ray hits. Since MSCP is a "packet" protocol, it uses some of the same error-correction algorithms as X.25 or packet radio - including packet sequence numbers. If the drive and controller start to get hopelessly confused, there's generally some kind of "hold everything and start over" signal that either master (controller) or slave (drive) can issue. This probably has the side effect of initializing the packet sequence number, which means that your error is very probably informational only.

MSCP tends to put out LOTS of "informational" "errors"... I for one wish that they could be excluded from error listings (much as the RSTS /NOTAPE_ERRORS qualifier)...

Mark Hartman
Jadtec Computer Group
546 W Katella Ave
Orange CA 92667

Note 646.0 Multiple Queues on one Printer 5 replies
"Pat Murphy" 19 lines 30-APR-1987 11:39

We have an application for connecting two generic queues to a single printer, with each of the queues having different default forms. We I had thought that by using generic queues sending their output to a single output queue, this would be be possible by using a device control library on the generic queues, each with a different /SEPARATE=RESET=xxx module. When I found out that one cannot attach a device control library to generic queues, I thought "ok, let's attach it to the output queue and just put the /SEPARATE modules on the generic queues" but still no joy.

I've currently got a kludge going that involves command procedures on remote nodes but I wonder if anyone else has run into this problem (and no, we can't afford to get another printer!)

Pat Murphy
National Radio Astronomy Observatory
P.O. Box "O"
Socorro, NM 87801-0387
(505) 772-4337

Note 646.1 Multiple Queues on one Printer 1 of 5
"John Osudar" 13 lines 30-APR-1987 16:15
-< one kludge or another... >-

There are ways to accomplish this, some of which are bigger kludges than others, and none of which (as far as I know) involve just using the standard DEC software. I set up a similar scenario using server queues and my EXECSYMB server symbiont: all of the generic queues feed into a single server queue. The queue processor for that server queue looks at the original (generic) queue name, and based upon that, requeues the job to the real queue with the proper qualifiers (/FORM, /SETUP, or whatever). Costs you one or two process slots (for EXECSYMB and a queue processor) but it works... If you are interested in more details, I can give you examples of what I did. The software itself will be on the Nashville VAX Sigtape (or I can send along an advance copy). [See also note 530.13]

John Osudar
Argonne National Laboratory
9700 S. Cass Ave.
Bldg. 205 A-051
Argonne, IL 60439-4837
(312) 972-7505

Note 646.2 Multiple Queues on one Printer 2 of 5
"Linwood Ferguson" 21 lines 13-MAY-1987 16:39
-< Two queues - One Printer >-

We had the same problem (needing to have two queues serving the same printer), and solved it a way that probably shouldn't work, but does.

We simply defined two terminal queues (its an LN03) that reference the same device. We found that you needed to put a WAIT in the startup commands; apparently the first initialize does something that prevents the second from running, but if you wait about 10-30 seconds it seems to work. Once you get them started, we've had no problems. Of course if you want to stop/change forms/anything else it is a real pain (we don't). It is NOT a good solution, but we were limited by the

application software (which let you specify different queue names, but not form types, and we wanted to print both landscape and portrait on the LN03).

Try it (preferably sometime when you can shut down if it all gets hung up). No guarantees, it's probably not suppose to work that way.

Linwood Ferguson
MJ Systems, Inc.
P.O. Box 5223
2564 Ivy Road (22901)
Charlottesville, Va. 22905-0223
804/977-2732

Note 646.3 Multiple Queues on one Printer 3 of 5
"Reece Pollack" 15 lines 18-MAY-1987 19:03
-< Printer sharing w/ DECservers >-

If you have an Ethernet and a DECserver 200, there is an easy way to do this. The LATSYM print symbiont is designed to share a printer attached to a terminal server with other processors (the server keeps a queue of pending print requests), and I doubt if the server cares whether the competing processor is itself or not. This doesn't require a cluster, but does require the server hardware, etc.

RE .2: You are right, this should not work. The print symbiont allocates the printer to prevent other processes from accessing it, but I guess it doesn't object to allocating it twice. The only problem is that I'll bet that if you queue two jobs to be printed on this printer, one through each queue, you end up with mixed output. Also, a print symbiont can handle 16 queues, and if you ever ended up with two print symbiont processes with one process handling one queue and the other handling the other, you'll have problems.

Reece Pollack
American Satellite Company
MS 34/MIS 1801 Research Blvd
Rockville, MD 20850

Note 646.4 Multiple Queues on one Printer 4 of 5
"Frank J. Nagy" 8 lines 21-MAY-1987 07:05
-< Multiple nodes w/DECServer printers >-

We have a couple of Printronix printers attached to DECServer-200s with several nodes having print queues for the printers. In one case there is one queue on a large CI-cluster and two queues on an LAVC for one printer. On the LAVC, if you PRINT two jobs via the generic queue, job #1 goes into execution queue A and starts printing. Job #2 goes into queue B and waits until job #1 is completed! Surprise, no "interleaved" output! Another neat job from Digital!

Frank J. Nagy
Fermilab
PO Box 500 MS/220
Batavia, IL 60510
(312)840-4935

Note 646.5 Multiple Queues on one Printer 5 of 5
"Tony Carter" 17 lines 21-MAY-1987 10:43
-< Symbionts and SHARE privilege >-

Yes, multiple queues on one machine to the same port (or service) on a Decserver 200 will work.

As far as why VMS doesn't care if you create two queues to the same printer, it is probably the infamous SHARE privilege. It allows you to do I/O to a device owned by someone else. The description of this privilege in the manuals says that it is normally only granted to things such as print symbionts. We have a dial-out modem connected to our terminal server. When a fully privileged user connected up to it's VAX port (LTAn:) that had already been allocated by another user, we saw the interleaved input and output that you're talking about. Very nasty. Needless to say we do not have SHARE on for anyone but SYSTEM now.

Tony Carter
P.O. Box 846
Middleton, MA 01949

(617) 245-6600

```
================================================================
Note 647.0              Mail for RSX-11M+              5 replies
"John P. McGrath"                    3 lines  30-APR-1987 18:52
----------------------------------------------------------------
```

Does anybody know of a MAIL program that runs under RSX-11M+ and is compatible with VMS MAIL? Unfortunately I do not have a lot of money to spend on this.

John P. McGrath
Software Consulting Services
3162 Bath Pike
Nazareth, PA 18064
(215) 837-8484

```
================================================================
Note 647.1              Mail for RSX-11M+                1 of 5
"John Osudar"                        12 lines  30-APR-1987 19:42
                     -< how about free? >-
----------------------------------------------------------------
```

A long time ago ("in a galaxy far, far away") someone "accidentally" put a set of VMS MAIL-compatible programs on the RSX Sigtape. Copies of that software have been floating around ever since, and as far as I know, they still work. (They're installed on our RSX systems here, but lately VMS has been keeping me too busy to spend much time on RSX systems, so I haven't tried them under the latest versions of M and M+).

I'll find out which tape the stuff was on when our resident RSX guru returns from the DECUS Symposium. If you don't have that particular tape, I'm sure someone can copy what you want onto a tape and send it to you.

John Osudar
Argonne National Laboratory
9700 S. Cass Ave.
Bldg. 205 A-051
Argonne, IL 60439-4837
(312) 972-7505


```
================================================================
Note 647.2              Mail for RSX-11M+              2 of 5
"Mark Hartman"                       2 lines   1-MAY-1987 19:39
                     -< what language? >-
----------------------------------------------------------------
```

John, do you know what language those programs for 11M were written in? I have some uses for such a MAIL system.

Mark Hartman
Jadtec Computer Group
546 W Katella Ave
Orange CA 92667

```
================================================================
Note 647.3              Mail for RSX-11M+              3 of 5
"John Osudar"                        5 lines   2-MAY-1987 22:58
                -< you won't like the answer >-
----------------------------------------------------------------
```

I believe that what was on the RSX Sigtape included ONLY the object modules. (As I recall, someone at DEC wrote it and gave it away to the RSX Sig, then caught hell from his bosses for doing so. But some copies got out anyway. At least that's the legend that's gone around with the software...)

John Osudar
Argonne National Laboratory
9700 S. Cass Ave.
Bldg. 205 A-051
Argonne, IL 60439-4837
(312) 972-7505

Note 647.4 Mail for RSX-11M+ 4 of 5
"Bruce R. Mitchell" 8 lines 8-MAY-1987 17:14
-< Source for MAIL package >-

The MAIL subsystem for RSX-11M/M+ is written in three languages:

Fortran, Basic-Plus-2, and MACRO. It was on the Fall 1981 and Fall 1982 RSX SIG tapes. There were two versions; RSX's mail and DECNET group mail. (The RSX group version is slightly better).

Bruce R. Mitchell
Machine Intelligence and Ind. Magic
PO Box 816
Byron, MN 55920
(507) 284-9202

Note 647.5 Mail for RSX-11M+ 5 of 5
"Mark Hartman" 1 line 8-MAY-1987 20:50
-< Really? >-

Bruce - just confirming that this is the VMS MAIL-compliant package?

Mark Hartman
Jadtec Computer Group
546 W Katella Ave
Orange CA 92667

Note 648.0 VAX TDMS vs. FMS 5 replies
"Richard Herdell" 24 lines 5-MAY-1987 11:04

After reviewing the SPDs for VAX FMS & VAX TDMS, these conclusions present themselves:

FMS is an early generation, forms control package

where

TDMS is a more recent 4gl forms control package which is integrated into the VAX Information Architecture.

I have been told, rightfully or wrongfully, FMS is still around because it is heavily entrenched in the user community but TDMS is meant to displace it. Any ideas?

My application interest is to control screen level data input and I'm interested in any war stories concerning these two sets of routines.

Richard Herdell
7000 Hollister
Houston, TX 77040

Note 648.1 VAX TDMS vs. FMS 1 of 5
"Larry Kilgallen" 28 lines 5-MAY-1987 11:49
-< TDMS does not replace FMS >-

Some may have seen TDMS as a replacement for FMS, but it never happened. Besides the use of FMS by pre-existing programs, there is the major factor that TDMS does not allow field-level manipulations to the degree that FMS does. In FMS one can easily write a validation routine, for example, to establish some arbitrary standard (e.g., numbers entered must be evenly divisible by 3), and that cannot be done in TDMS. On the other hand, TDMS does provide asynchronous functions, allowing a program to continue processing until input is provided. That capability is not in FMS.

At the Spring 1987 DECUS US Symposium in Nashville last week, there was also discussion of the possibility of DEC implementing another forms interface, compliant with the forthcoming ANSI (?) standard. Since DEC has played a major part in the standards committee work (having a variety of experience implementing the beasts), it would seem likely that DEC would implement a third choice beyond FMS and TDMS. The nature of the forms interface described in the standard is sufficiently different from either FMS or TDMS is such that an implementation would NOT be an extension of either existing package. While the standards committee has dealt with generic issues such as the separation of form from function, users at the Nashville symposium gave comments to DEC on what they would expect in the way of VMS-specific features for an implementation of the standard (AST routines, etc.).

Larry Kilgallen
Box 81, MIT Station
Cambridge, MA 02139-0901

```
================================================================
Note 648.2                VAX TDMS vs. FMS                2 of 5
"Bob Hassinger"                    20 lines   5-MAY-1987 12:30
          -< FMS will not go away anytime soon... >-
----------------------------------------------------------------
```

One of the interesting pre-existing applications that uses FMS is All-In-1. It is my impression that All-In-1 uses enough of the unique features of FMS that are not available in TDMS so that conversion to TDMS would be "a major pain" if it was possible at all. In my own work I find I always seem to need the lower, field level, functions to make things work the way I want so there is no choice but to keep using FMS.

Yes, many at DEC made it quite clear they intended for TDMS to replace FMS but it never happened and, in fact, I wonder if FMS is not the stronger of the two products today. At first we were told FMS would go away or not be developed and the FMS contingent in DECUS made a it clear they did not like the idea, then we were told DEC was going to work to bring the two products together and then we saw new releases of FMS but with no signs of convergence. At the Symposia I still sense more interest and involvement in FMS.

The story of how we got to this sorry state will have to wait for another day...

Bob Hassinger
Liberty Mutual Research Center
71 Frankland Road
Hopkinton, MA 01748
617-435-9061

```
================================================================
Note 648.3                VAX TDMS vs. FMS                3 of 5
"Mark Hartman"                     4 lines   8-MAY-1987 20:49
                -< But you never know... >-
----------------------------------------------------------------
```

Another interesting note is that DEC is continuing development on FMS from both the PDP-11 and VMS sides. This is not normally done unless the VMS side will be around for awhile.

Mark Hartman
Jadtec Computer Group
546 W Katella Ave
Orange CA 92667

```
================================================================
Note 648.4                VAX TDMS vs. FMS                4 of 5
"Linwood Ferguson"                31 lines  13-MAY-1987 16:32
             -< FMS - Hated, but best going >-
----------------------------------------------------------------
```

We evaluated FMS and TDMS about 2+ years ago, after already doing several months of development in FMS. Conclusion (remember, TDMS may have changed a lot by now):

1) We absolutely HATE FMS

2) We use FMS instead of TDMS

The reason is primarily those in a previous reply: FMS lets you handle fields as they are typed, rather than later. We do lots of this (for instance, most fields that refer to other files we link to a LOOKUP function through the FIND key; FMS lets you interrupt that screen entry and go off in a UAR (User Action Routine), inquire against that file, and finally return a value

to the first screen all relatively (as much as anything is) transparent to the programmer that is writing the particular screen in use). It is my understanding that TDMS still does not have these kinds of field validation routines.

FMS is a performance hog, poorly integrated, and a pain to program in. But if you take the time, you can do more for the user than you can with TDMS. Now, when we occasionally are lazy and use DISPLAY/ACCEPT or DCL to put something on a screen, spoiled users complain.

If we had it to do over, however, we would spend the first few months doing our own version of FMS!

Linwood Ferguson
MJ Systems, Inc.
P.O. Box 5223
2564 Ivy Road (22901)
Charlottesville, Va. 22905-0223
804/977-2732

```
================================================================
Note 648.5                VAX TDMS vs. FMS                5 of 5
"Steve Duff"                          22 lines  14-MAY-1987 06:08
                         -< FMS/TDMS >-
----------------------------------------------------------------
```

In addition to the field functionality limitations on TDMS, I can attest to the fact that the development cycle with TDMS is a major burden because of the involvement with CDD, building libraries, etc, etc... All this soaks up cycles and man hours during programming, test and Q/A, and your requirements in this regard, I believe, should be considered carefully.

On the other hand, TDMS provides essentially complete data mapping (though with limited control), and is easier to work with in a multi-thread transaction environment. TDMS programs are often remarkably short as a result.

Having spent considerable time with both products, I tend to prefer FMS. We usually write a data-mapping layer in the code which isn't hard to do, and substitutes for that aspect of TDMS without the involvement of the lugubrious CDD business. At run-time, I haven't seen much difference between the two products (screen management isn't where most programs spend time anyway.)

FMS and TDMS are really quite different products, about the only thing they share in common is the style of the forms editor.

Steve Duff
Software Factory
2401 E. 17th St.
Suite 190
Santa Ana, Ca. 92701
(714) 752 2972

```
================================================================
Note 649.0          Minor problem with NCP HELP\          3 replies
"Jamie Hanrahan"                      18 lines  6-MAY-1987 00:16
----------------------------------------------------------------
```

In NCP, how does one get HELP on a subtopic that consists of multiple words? For instance..

```
NCP> HELP SET CIRCUIT
```

works fine, but

```
NCP> HELP SET CIRCUIT MAXIMUM BUFFERS
```

does not, even though the first command shows that 'MAXIMUM BUFFERS' is a valid option for 'SET BUFFERS' (as, indeed, it is). You can get the information by saying

```
NCP> HELP SET CIRCUIT MAXIMUM
```

but then you get information for *every* option starting with 'MAXIMUM' (and there are several of them). Is there a way to make the second command, or something like it, work?

Jamie Hanrahan
Simpact Associates
9210 Sky Park Court
San Diego, CA 92123
619-565-1865

```
================================================================
Note 649.1          Minor problem with NCP HELP\          1 of 3
"George Walrod"                        6 lines   6-MAY-1987 10:03
                    -< Ambiguous Grammar >-
----------------------------------------------------------------
```

Sorry Jamie I have the same problem.

The problem seem to be with ambiguous grammar in subtopic, submit an SPR, but I don't see a solution as far as help. But a note added to help documentation saying beware of ambiguous grammar when making help topics.

George Walrod
4260-b chain bridge rd
fairfax, va 22030

```
================================================================
Note 649.2          Minor problem with NCP HELP\          2 of 3
"Reece Pollack"                        5 lines   6-MAY-1987 19:55
               -< HELP with Multi-word Topics >-
----------------------------------------------------------------
```

If I'm not mistaken, if you ask for help on the topic just above the multi-word topic, you can then enter the multi-word topic at the "Help topic?" prompt. Give it a try -- I seem to remember this to be the only way to get to subtopics below a multi-word topic as well.

Reece Pollack
American Satellite Company
MS 34/MIS 1801 Research Blvd
Rockville, MD 20850


```
================================================================
Note 649.3          Minor problem with NCP HELP\          3 of 3
"M. Erik Husby"                        2 lines   8-MAY-1987 08:47
                    -< SPR on NCP HELP >-
----------------------------------------------------------------
```

I submitted an SPR on this problem quite a while ago and never received a response. I have no work arounds either.

M. Erik Husby
Project Software & Development
14 Story St.
Cambridge, MA. 02138
(617)-661-1666

```
================================================================
Note 650.0              Heavyweight update!              6 replies
"Michael R. Pizolato"                  8 lines   8-MAY-1987 10:49
----------------------------------------------------------------
```

After a long and arduous struggle, I got my update services started again and just received 436 pounds of software from DEC (I have a lot of layered products). I got VMS up to V4.5, and lots of other goodies, too. Does anyone have any stories about installing some of the more recent versions of VMS? I am now running 4.2, and I want to go all the way to 4.5 in one shot. What about impact on layered products or applications developed using layered products or languages? Anything at all, really.

Michael R. Pizolato
AT&T Technology Systems
Dept. 323610
555 Union Blvd.
Allentown, PA 18103
215/439-5500

```
================================================================
Note 650.1              Heavyweight update!              1 of 6
"Joel Gallun"                          6 lines   8-MAY-1987 15:45
                       -< Update Help >-
----------------------------------------------------------------
```

I belive you can go directly to 4.4 and then upgrade to 4.5. 4.4 is a complete distribution while 4.3 and 4.5 and all the other odd tenths of a version are just patch kits. (that's what we used to call them in my PDP-11 days, anyway)

Joel

Joel Gallun
oao corp
7500 greenway ctr
greenbelt, md 20770

```
================================================================
Note 650.2              Heavyweight update!              2 of 6
"Linwood Ferguson"                     9 lines  13-MAY-1987 16:23
            -< 4.2 --> 4.5 Problems - Runoff >-
----------------------------------------------------------------
```

> Problems between 4.2 and 4.5?

We are mostly on 4.4 and 4.5; both are very solid (we didn't even bother taking most nodes from 4.4 to 4.5).

One problem we had around 4.3 was RUNOFF stopped processing lots of .REQUIRE's correctly. All of our documentation was in Runoff, and it all stopped working. We are still using the 4.3 (I think) version of Runoff.

Linwood Ferguson
MJ Systems, Inc.
P.O. Box 5223
2564 Ivy Road (22901)
Charlottesville, Va. 22905-0223
804/977-2732


```
================================================================
Note 650.3              Heavyweight update!              3 of 6
"Bruce Bowler"                         4 lines  18-MAY-1987 13:07
                          -< RUNOFF >-
----------------------------------------------------------------
```

We use .REQUIRE's all the time in our documents and have had NO problems with them on 4.3, .4 or .5. What sort of problems are you experiencing, and what has DEC's response been??

Bruce Bowler
General Electric
1 River Road
Bldg 2 Room 609
Schenectady, NY 12345

```
================================================================
Note 650.4              Heavyweight update!              4 of 6
"Linwood Ferguson"                    30 lines  18-MAY-1987 13:58
                -< Runoff Error Still Around >-
----------------------------------------------------------------
```

> We use .REQUIRE's all the time in our documents and have had NO
> problems with them on 4.3, .4 or .5. What sort of problems are
> you experiencing, and what has DEC's response been??

When doing a file with LOTS of requires, we get:

%RUNOFF-W-COR, Can't open required file xxxxxxx

"unexpected operating system error"

This happens after (pardon my memory) about one or two hundred requires. Once it starts, it never quits. I just did this again with VMS4.5A. We're running a version with a date (in the image header) of 23-Jun-85; the current version is 22-Mar-86. I think that working version was 4.3 or 4.2, though I can't swear to it.

Dec said "yes, it does, we'll SPR it". That was so long ago I lost the sequence number, and have never bothered calling again. I guess I should. I never got a followup call from them. I should have sent in the SPR myself, but get tired of them being swallowed by the black hole somewhere between here and New

England.

One other item of note (in regard to big jumps in versions): somewhere along the way, and I don't remember exactly when, runoff changed the way it handled .REQUIRE default/relative file specs. It use to be that it defaulted to the current directory, then it defaulted to the last filespec required. I have not checked to see if they ever changed back (after I put in LOTS of directory specs, I didn't bother to take them out again).

Linwood Ferguson
MJ Systems, Inc.
P.O. Box 5223
2564 Ivy Road (22901)
Charlottesville, Va. 22905-0223
804/977-2732

```
================================================================
Note 650.5              Heavyweight update!              5 of 6
"Reece Pollack"                        17 lines  18-MAY-1987 18:38
                -< OK to skip V4.3 update... >-
----------------------------------------------------------------
```

I jumped from V4.2 to V4.4 and then did the V4.5 update, and this works well, but you can't go directly to V4.5. Good luck with the layered products though, I know a few things broke around V4.4/5. COBOL has (had?) a bug introduced to the ACCEPT WITH CONVERT statements, but this may have been fixed in V4.5. A number of our old BASIC applications written under V1.X of BASIC as a result of RTL and/or RMS changes, but I really wasn't surprised by that. The PRINT/PAGES qualifier broke with V4.5 (it's now ignored), but you can use the V4.4 print symbiont if you need this qualifier more than the bug fixes. A good "gotcha" is the introduction of the SUBROUTINE statement -- these are great except that SUB ceases to be an acceptable abbreviation for SUBMIT, but DEC has always said that commands would be unique to 4 chars, not 3...

I probably have forgotten a few things (I'm doing this from memory), but the system didn't fall apart. Good luck and try not to get a hernia from lifting all that documentation.

Reece Pollack
American Satellite Company
MS 34/MIS 1801 Research Blvd
Rockville, MD 20850


```
================================================================
Note 650.6              Heavyweight update!              6 of 6
"Edward Chan"                          1 line  19-MAY-1987 16:38
        -< Be sure to upgrade LAT PLUS after vms 4.5 >-
----------------------------------------------------------------
```

Edward Chan
1501 Harbor Bay Parkway
Alameda, CA 94501

```
================================================================
Note 651.0        Need VMS TM11 device driver         No replies
"Offline Submission"                  14 lines   9-MAY-1987 12:07
----------------------------------------------------------------
```

We are looking for a VMS device driver that supports the old TM11 controller/drives. We must have sources since the hardware is not quite TM11-compatible (one is Unibus, the other is 22-bit Q-bus.) We prefer public domain, but can pay some dollars to avoid doing it ourselves.

Lawrence M. Baker
US Geological Survey - OEVE
345 Middlefield Road, M/S 977
Menlo Park, CA 94025

Telephone: (415) 329-5608

Date: May 5, 1987

```
================================================================
Note 652.0   Termtable definitions for SMG routines.   2 replies
"Michael R. Pizolato"                 17 lines  11-MAY-1987 13:28
----------------------------------------------------------------
```

Does anyone know of any efforts to collect and distribute TERMTABLE definitions for the SMG Run-Time Library routines? Right now I'm looking for definitions for Tektronix terminals in particular, but I feel that a comprehensive collection would benefit everyone and should be put together.

If no one is doing it yet, I'm willing to do the collecting. I guess the best way to do the distribution would be through DECUS. But, before I get deluged with TERMTABLE definitions, I would like some suggestions on how to collect the stuff without killing myself (for example, I certainly wouldn't want anything on paper). Then I could publish the rules of the game here, so that everyone will know what to do.

Any suggestions?

:-) :-) :-)

Michael R. Pizolato
AT&T Technology Systems
Dept. 323610
555 Union Blvd.
Allentown, PA 18103
215/439-5500

Note 653.0 HSC-50 runs only with 3 phase power?? 4 replies
"MIKE SCHOEPKE" 10 lines 13-MAY-1987 11:36

Is there any way to reconfigure the HSC-50 to work on single phase power? Our computer room is located in an office complex that does not three phase available. Any information would be appreciated. Mike Schoepke Paddock Publications PO Box 280 Arlington Heights, Il 60006 (312) 870-2667

MIKE SCHOEPKE
PADDOCK PUBLICATIONS
PO BOX 280
ARLINGTON HEIGHTS, IL.
60006
(312) 870-2667

Note 653.1 HSC-50 runs only with 3 phase power?? 1 of 4
"MICHAEL GRATTAN" 5 lines 14-MAY-1987 06:23
-< Power equipment >-

Just a thought, while you may not be able to touch the HSC-50, perhaps you could look at some power conditioning/supply equipment. I keep getting all this literature about all the wonderful things they can do with your electrical power. Perhaps there's some kind of conversion equipment to go from single phase to three phase.

MICHAEL GRATTAN
FAIRHAVEN CORP.
358 BELLEVILLE AVE.
NEW BEDFORD, MA. 02742
617-993-9981 EXT 106

Note 653.2 HSC-50 runs only with 3 phase power?? 2 of 4
"Bob Hassinger" 23 lines 14-MAY-1987 08:08
-< Conversion to three phase possible, but... >-

Are you sure you can not get a version for single phase? I can not understand this for a device like the HSC-50 which is mostly electronics, not large motors and so on. Can you get the disk drives configured for single phase?

If you really had to, I think you could make three phase from your single phase with a rotary converter - basically a single phase motor turning a three phase generator. I understand they are available although a lot of people seem to avoid them due to cost and so on.

Also, I know of people who use a tricky device, a kind of a transformer I think, that takes 220 three wire single phase and gives "three phase" power that will run things like three phase motors (i.e. motors on larger woodworking machines and the like). This has always seemed to me as though it was being done with mirrors so I am not too confident about it, particularly where you are not dealing with rotating power devices such as motors.

Based on what I have seen I would say to try hard to get real three phase power or else try VERY hard to eliminate the need (like by getting DEC to sell you a single phase model).

Bob Hassinger
Liberty Mutual Research Center
71 Frankland Road
Hopkinton, MA 01748
617-435-9061

```
================================================================
Note 653.3     HSC-50 runs only with 3 phase power??      3 of 4
"Bruce Bowler"                        12 lines  18-MAY-1987 13:15
                      -< Check again >-
----------------------------------------------------------------
```

Your in an office complex that doesn't support 3Phase power? sure sounds to me like someone is handing you a line. Most large A/C units like those found in office complexes use 3 phase, most (if not all) of the power into the complex will be three phase.

In short, I can't believe that 3 phase is not available. I can believe that the building manager doesn't know it's available.

Bruce Bowler
General Electric
1 River Road
Bldg 2 Room 609
Schenectady, NY 12345

```
================================================================
Note 653.4     HSC-50 runs only with 3 phase power??      4 of 4
"Jack Patteeuw"                       12 lines  19-MAY-1987 10:53
              -< Too much for just one phase >-
----------------------------------------------------------------
```

The HSC50 "probably" requires 3 phases because it would draw too much over just one phase ! This is the case of the 4-high RA81 cabinet requiring three phases where the three high only required one phase.

re: .3

Bruce is correct. Almost ALL commercial building of even modest size have three phase brought into them because the cost of running the HVAC blowers is much less. As a matter of fact, it is impossible in our building to get 220v 1-phase!!

Jack Patteeuw
Ford Motor Co.
Electrical and Electronics Division
31630 Wyoming
Livonia, MI 48150
313-323-8643

```
================================================================
Note 654.0   Mail, Networks, Slaves, and Dist Lists   No replies
"Linwood Ferguson"                    63 lines  13-MAY-1987 16:56
----------------------------------------------------------------
```

We have an unusual mail problem that Colorado Springs hasn't been much help with. We have a medium size network (20 nodes), slow (lots of busy multipoint nodes), and send lots of mail, most to distribution lists involving at least 2-3 nodes each.

This takes relatively forever (sometimes 3-5 minutes) between the TO: and the SUBJ:.

Originally we defined distribution lists as logicals which in turn were files:

```
        GROUPA  is "@SYS$MANAGER:GROUPA.DIS"

                    contains: A::FRED
                              B::HARVY
                              C::JUDY
```

Works. To keep updating to a minimum (these change a lot), we put the list on one node, so GROUPA might be only on A::, and on D:: its definition is

```
        GROUPA is "A::@SYS$MANGER:GROUPA.DIS"
```

Works. Real slow as it creates a FAL process to read the file, then a MAIL process on each destination node. Next we tried this:

```
GROUPA (on D::) is "A::GROUPA"
GROUPA (on A::) is "@SYS$MANAGER:GROUPA.DIS"
```

Rationale was that D:: would ship the "address" to A:: who would recognize it as a distribution list, and the slave process would read the .DIS file and send the mail. Guess what: it worked. Much faster.

Except. Once in a while, the mail didn't go through. The NETSERVER.LOG process shows a Link Abort (in the example it would be on A:: when sending to GROUP from D::). No explanation, no further error. No identifiable Decnet problems (everything else works). It is repeatable, but intermittent. It seems a little related to load, but not in any firm way.

DEC's response: Mail group says "You're using Decnet, don't talk to us". Decnet group says "lets check your parameters, ..."; no problem found, but also no knowledge of how the mail master/slave process really work.

Anyone doing this (passing "addresses" that translate to address lists on the salve)? Is it working? Any ideas?

My pet idea is that the master treats it as one "user", but the slave treats it as lots. The slave sends an acknowledgement to the master, who drops the link. The slave then aborts. Since all this takes time, and is not synchronous, sometime the slave finishes and sometime it doesn't. Dec listens politely to this and then asks me to check INCOMING or OUTGOING again for the 10th time.

Help? Anything appreciated.

Linwood Ferguson
MJ Systems, Inc.
P.O. Box 5223
2564 Ivy Road (22901)
Charlottesville, Va. 22905-0223
804/977-2732


```
================================================================
Note 656.0              Replacement of BNT              No replies
"George Walrod"                          2 lines  18-MAY-1987 11:05
----------------------------------------------------------------
```

Has anyone heard of a replacement for BNT called the BNA on the 8xxx BI CPUs?

George Walrod
4260-b chain bridge rd
fairfax, va 22030

```
================================================================
Note 657.0    Memory disk driver efficiency questions    1 reply
"John Osudar"                           30 lines  19-MAY-1987 14:18
----------------------------------------------------------------
```

There's been some discussion in previous Pageswappers about the VMS "memory disk" driver (written by Jay Olson) that comes in your VMS kit as PDDRIVER, and is used to enable stand-alone backup to boot from TK50's. A "RAM disk" can be useful in a number of applications where a fast-access "disk" would improve performance (frequently-accessed files, temporary "work" files, etc.) Comments have been made (at DECUS and in other forums) that PDDRIVER is bad for your system, because it does its data transfers with a MOVB instruction at IPL 8.

If you read the fiche, you'll find that this is true; PDDRIVER uses IOC$MOVTOUSER and IOC$MOVFRUSER to do its data transfers. These routines contain code that looks like: BITW to see if at end of page; BNEQ to skip end-of-page code; MOVB one byte of data; SOBGTR on the byte count to the top of the loop.

Since I used to be an RSX driver hacker, I recall a thing called BLXIO that RSX used to speed up memory-to-memory data transfers. This was a sequence of MOV instructions; you computed an offset into the list and branched there, and it did the transfers at one instruction per word.

My question is: has anybody who uses PDDRIVER thought about (or actually implemented) a similar scheme to improved PDDRIVER's performance? I've looked at the code, and it appears to be relatively simple to replace the IOC$MOVxxUSER calls with equivalent code that could, for example, do a short table of byte moves and a long table of quadword-aligned quadword moves.

Before I'd attempt to implement such a thing myself, I'd like to know (a) if someone has tried it and found it isn't worth doing; (b) if there's some reason why it shouldn't be done; or, preferably (c) that someone has done it, is using it, and is willing to give it away for free. (I'm always willing to run someone else's debugged, useful, and free software!)

John Osudar
Argonne National Laboratory
9700 S. Cass Ave.
Bldg. 205 A-051
Argonne, IL 60439-4837
(312) 972-7505

Note 657.1 Memory disk driver efficiency questions 1 of 1
"Jamie Hanrahan" 19 lines 19-MAY-1987 17:54
-< don't use MOVWs, use MOVC3 instead >-

I have used the BLXIO technique in a driver that accessed a shared memory device on a MicroVAX I. It was necessary because the shared memory had to be mapped in "non-cached mode" (obviously), and MOVC won't work to/from non-cached locations. An early version of the driver allocated a number of SPTEs (not just one) and mapped the entire user buffer into system space (at start I/O time; naturally the buffer was probed and locked at FDT time). The final version just did buffered I/O; the code was cleaner and the speed difference for buffers under 4Kbytes was negligible. Anyway, we did a BLXIO-type move between the user's data and the desired part of the shared memory.

We tried doing it that way on the uVAX II, but MOVC3 was considerably faster. If anyone is planning to improve PDDRIVER I would suggest opting for dynamically mapping the entire user buffer (or at least more than one page; four or eight pages would probably handle most transfers) and using MOVC3.

Jamie Hanrahan
Simpact Associates
9210 Sky Park Court
San Diego, CA 92123
619-565-1865

Note 658.0 Berkeley UNIX on uVAXen No replies
"Tony Carter" 8 lines 22-MAY-1987 15:10

Does anyone out there run standard Berkeley Unix on a MicroVAX or a Vaxstation? We are looking to do just this, but the main problem is the bootstrap. No one seems to have it for the uVAXen. Ultrix is not really a solution that we want to go to.

Tony Carter
P.O. Box 846
Middleton, MA 01949
(617) 245-6600

Note 659.0 Logicraft on VAX? No replies
"MICHAEL GRATTAN" 4 lines 26-MAY-1987 12:02

I am wondering if anyone has had any experience using Logicraft products that run MS-DOS software on a VAX. I am interested in any comments you might have. Thanks.

MICHAEL GRATTAN
FAIRHAVEN CORP.
358 BELLEVILLE AVE.
NEW BEDFORD, MA. 02742
617-993-9981 EXT 106

*

*

RT-11 DUCM/DYC GRAF11 DEL DIR **PLOT-10** IMAGE LIBED FSTATS MS/DOS TIC-TAC-TOE QIX VAX-LIB-**DATMAN/VAX** EDTPlus SPICE2 **TREEDUPL** LISPEX MCLS TYPE_PLUS MINC DISK_USE FRAG EDTEX PORT LOCATOR TECO CHPLOT NANNY DIR11-W WATCHDOG INACTIVE ACCOUNTS IMGSPL ICE TEX EDITOR **VAX-LIB-4** GRAPHIC UTILITIES SETAUX.ARC STATPK FIGure KERMIT Distribution TENBACKU **JUICER** VTEDIT 2022 VAX-LIB-3 VISTA EDITOR MTU TDE RSTSOPEN DRAWTREE WATCHDOG PRM-1 SMARTMAILER TEN SPELL **DECPoint of Sale** JUICER **PARALLEL Library V2** RTMULTI and Addo SMARTMAILER for RSTS/E CU FILTRA Spring 86 RT-11 SIG CP/M KERMIT S Invasion for PRO Bonner La SPLICE RUNOFF VAX-LIB-3 VAX-LIB-2 IMAGE SPELL **TURBOCOM** FNDFIL PC-8088 Collection #10 VT20 TOOLKIT PLATOOLS **SMARTMAILER** DEPROC LaTex KERMIT-11 **FANCY FONTS** XMIT CU ReGis to HPG CED International RUNITOFF JP5-JP6 FODT PASCAL-OS/8 ANISMT WPSIM **PARALLEL LIBRARY** DECSYSTEM-20 SIG Spring 85 CAMERA DELPHIN HACK BIBENTRY APFELN DIGITIZING Acid Docume Generator VAX-LIB-2 **AMAR-10 AMAR-20 DATMAN/VAX** IMAGE RT-11 DUCM/DYC GRAF11 DEL DIR PLOT-1 IMAGE LIBED FSTATS MS/DOS TIC-TAC-TOE QIX VAX-LIB-5 DATMAN/VAX EDTPlus SPICE2 TREEDUP LISPEX MCLS TYPE_PLUS AMAR-20 **DISK_USE** FRAG EDTEXT PORT LOCATOR TECO CHPLOT NANN DIR11-W WATCHDOG INACTIVE ACCOUNTS IMGSPL **ICE TEXT EDITOR VAX-LIB-4** GRAPHIC UTILITIE SETAUX.ARC STATPK FIGure KERMIT Distribution TENBACKUP JUICER VTEDIT **2022** VAX-LIB-3 VIST EDITOR MTU TDE RSTSOPEN DRAWTREE WATCHDOG PRM-11 **SMARTMAILER** TEN SPELL DECPoint of Sa JUICER **PARALLEL Library V2** RTMULTI and Addons SMARTMAILER for RSTS/E CU FILTRA Spring 86 RT-1 SIG CP/M KERMIT S Invasion for PRO Bonner Labs APFELN RUNOFF **VAX-LIB-3** VAX-LIB-2 IMAGE **SPEL** TURBOCOM FNDFIL **PC-8088 Collection #10** VT200 TOOLKIT PLATOOLS SMARTMAILER DEPROC **LaTe** KERMIT-11 FANCY FONTS XMIT CU ReGis to HPGL CED International RUNITOFF JP5-JP6 FODT PASCAL-OS/ ANISMT TECO WPSIM **DECSYSTEM-20 SIG** Spring 85 CAMERA DELPHIN HACK BIBENTRY APFELN KERMI S DIGITIZING Acid Document Generator VAX-LIB-2 AMAR-10 AMAR-20 DATMAN/VAX IMAGE VT200 TOOLKI COMPRO EVENTS **PC8088** Collection #9 TECO Cherry Tree Workstation Bookings System EXPORT Data Inputt Generator CMSBROWSE PERSONNEL INVENTORY MS/DOS COMMS Selection Electronic Grade Book CP/ KERMIT LaTex JUICER SPELL PORTACALC DPRINT DUNGEON MINC **BUDGET** BUG CALC C Langua System **DPROC "DEP"** DECENC **DECmate II OS/278** DIAL DTC GAMMA-11 GDADL **LISP** for RSX-11 **MEM** KERMIT S **VAX-LIB-6** SPICE 3A6 **VT200 TOOLKIT** RUNNOFF SPLICE SPY:RSX TCOPY **SPELL** VT-200 COMPR EVENTS CMSBROWSE UNDELETE DIAL BLOCKER SCAN CODER BITMAP DTC/PC ADDRESS BOO LaserWriter **PORTACALC** SPICE 3A6 PRO/Smart Mailer CBASIC2 Accts JP5-JP6 Payable/Receivable McGraw-Hi Payroll SEDT: EDT/WPS Screen CLNDRS:A **Calendar** Program INDEX AKCOUNT CORPHONE **E-Systems** Grab Ba RGT RDG PLTXSMB ICON DEVICS DATATRIEVE Library Collection CMSBROWSE EXPERT FPaint IMAG DBMS/Spreadsheet for MS/DOS AMAR-1 AMAR-2 RDIR/SQMAP PC-8088 Collection #1 UP TIME REPORTE RT-11 DUCM/DYC GRAF11 DEL DIR PLOT-10 IMAGE LIBED FSTATS MS/DOS TIC-TAC-TOE QIX VAX-LIB **DATMAN/VAX** EDTPlus SPICE2 TREEDUPL LISPEX MCLS TYPE_PLUS EXPORT DISK_USE FRAG EDTEX PORT LOCATOR TECO CHPLOT NANNY DIR11-W WATCHDOG INACTIVE ACCOUNTS IMGSPL ICE TEX EDITOR **VAX-LIB-4** GRAPHIC UTILITIES SETAUX.ARC STATPK FIGure KERMIT Distribution TENBACKU **JUICER** VTEDIT 2022 VAX-LIB-3 VISTA EDITOR MTU TDE RSTSOPEN **DRAWTREE** WATCHDOG PRM-1 **SMARTMAILER TEN SPELL** DECPoint of Sale JUICER **PARALLEL Library V2** RTMULTI and Addo SMARTMAILER for RSTS/E CU GRAPHKIT FILTRA Spring 86 RT-11 SIG CP/M KERMIT S Invasion for PR Bonner Labs RUNOFF **VAX-LIB-3 VAX-LIB-2** IMAGE SPELL TURBOCOM FNDFIL PC-8088 Collection #10 **VT2** **TOOLKIT** PLATOOLS SMARTMAILER DEPROC **LaTex** KERMIT-11 **FANCY FONTS** XMIT MEMO ReGis to HPG CED International RUNITOFF JP5-JP6 FODT PASCAL-OS/8 ANISMT CODER WPSIM DECSYSTEM-20 SIG Sprin 85 CAMERA DELPHIN HACK BIBENTRY APFELN REPORTER DIGITIZING Acid Document Generator VAX-LIB-AMAR-10 AMAR-20 **DATMAN/VAX** IMAGE **VT200 TOOLKIT** COMPRO EVENTS **PC8088 Collection #9** TECO Cher Tree Workstation Bookings System EXPORT Data Inputter Generator CMSBROWSE PERSONNEL INVENTOR MS/DOS COMMS Selection Electronic Grade Book CP/M KERMIT **LaTex** JUICER SPELL PORTACALC DPRIN DUNGEON MINC **BUDGET** BUG CALC C Language System DPROC "DEP" DECENC DECmate II OS/278 DIA DTC GAMMA-11 GDADL LISP for RSX-11 MEMO PORTACALC VAX-LIB-6 **SPICE 3A6** VT200 TOOLKI RUNNOFF **SPLICE** SPY:RSX TCOPY **SPELL** VT-200 COMPRO EVENTS CMSBROWSE UNDELETE DIA BLOCKER SCAN CODER BITMAP DTC/PC ADDRESS BOOK LaserWriter PORTACALC SPICE 3A6 PRO/Sma Mailer CBASIC2 Accts TREEDUPL Payable/Receivable McGraw-Hi Payroll SEDT: EDT/WPS Screen CLNDRS: Calendar Program INDEX AKCOUNT CORPHONE E-Systems Grab Ba RGT RDG PLTXSMB ICON DEVIC DATATRIEVE Library Collection CMSBROWSE EXPERT FPaint IMAGE DBMS/Spreadsheet for MS/DOS AMAR-1 AMAR-20 RDIR/SQMAP PC-8088 Collection #11 UP TIME REPORTER RT-11 DUCM/DYC GRAF11 DEL DIR PLO 10 IMAGE LIBED FSTATS MS/DOS TIC-TAC-TOE QIX VAX-LIB-5 DATMAN/VAX SPICE2 RT-11 DUCM/DYC G

DECUS PROGRAM LIBRARY

NEW LIBRARY PROGRAMS AVAILABLE FOR THE VAX/VMS FAMILY OF COMPUTERS

DECUS NO: VAX-247 **Title:** LEFTWILD.COM **Version:** March 1987

Submitted by: Allan J. Mui, Manufacturers Hanover Trust Company, New York, NY **Operating System:** VAX/VMS V4.3 **Source Language:** DCL **Keywords:** DCL

Abstract: In VAX/VMS DCL, the use of partial wild cards in output file specifications is not supported, e.g., you cannot issue the command

"$" rename *1.dat *2.dat". In certain cases such usage of wild cards would be ambiguous. In the case given above, however, there is no ambiguity. This command file was written to allow the above and similar commands to be issued by passing the verb and its arguments as parameters to the command file. Other types of wild card constructions could be similarly allowed with similar command files. In this way, the syntax of DCL can be extended.

Notes: Only the asterisk wild card is permitted in file names passed to this command file.

Media (Service Charge Code): 600' Magnetic Tape (MA) **Format:** VMS/BACKUP

DECUS NO: VAX-248 **Title:** SIM: A Simulator for the M68010 **Version:** February 1987

Submitted by: Walter H. Burkhardt, Univ. Stuttgart/Inst. fur Informatik, D-7000 Stuttgart 1, West Germany **Operating System:** VAX/VMS V3.7 **Source Language:** PASCAL **Memory Required:** 1MB **Software Required:** PASCAL in case of modifications. **Keywords:** Motorola

Abstract: This system simulates the Motorola M68010 microprocessor. The program to be simulated has to appear in the S100 format, as given in the system UCAMS (a universal microprocessor cross-assembler; the needed portions are included here).

The execution of the simulation can be controlled and the contents of the memory cells and the registers can be manipulated interactively or by a command-file.

The programs are written in PASCAL and the complete documentation is given in German. There are several explained examples in the documentation.

The chapter "BENUTZERANLEITUNG" in the documentation gives a guide to the usage of the system.

Notes: Complete documentation is given in German.

Media (Service Charge Code): 600' Magnetic Tape (MA) **Format:** VMS/BACKUP

DECUS NO: VAX-249 **Title:** MACS: The MACRO Searcher **Version:** February 1987

Submitted by: Walter H. Burkhardt, Univ. Stuttgart/Inst. fur Informatik, D-7000 Stuttgart 1, West Germany **Operating System:** VAX/VMS **Source Language:** PASCAL **Memory Required:** 500KB **Software Required:** PASCAL System for modifications. **Keywords:** MACRO

Abstract: This system searches through an object file for the detection of repetition of code sequences. The number of occurences and the code sequences will be found.

The object file can be for any given system. For this reason, this system needs some information about the object code in which the object file is given:

. Description of the address-coding
. Decription of the adressing modes
. Description of the machine codes

This information is given in a code-list.

The output from this system is in "outfile" with the output information. The macros that are found are sorted according to their length and listed in disassembled form. Also given are the addresses at which the macros are found.

The complete documentation for the system is in German. MACRO.DOC gives the description of the system, MACRO.PAS contains the listing for the programs, MACRO.EXE is the runtime image for the system.

The documentation explains the working of the system with a given example for the 6502.

Notes: Complete documentation is in German.

Media (Service Charge Code): 600' Magnetic Tape (MA) **Format:** VMS/BACKUP

DECUS NO: VAX-250 **Title:** UCAMS: Universal Cross-Assembler for Microprocessors **Version:** February 1987

Author: J. M. Weis

Submitted by: W. H. Burkhardt, Univ. Stuttgart/Inst. fur Informatik, D-7000 Stuttgart 1, West Germany **Operating System:** VAX/VMS V3.7 **Source Language:** PASCAL **Memory Required:** 1MB **Software Required:** PASCAL for modifications **Keywords:** Motorola

Abstract: This system serves as a universal cross-assembler especially for microprocessors.

This cross-assembler is created by the command file UCAMS.COM. The source programs and guidelines for the construction of the system can be found in the documentation.

DEFASSEMB.EXE;1	Programs for translating the description of the set of codes in internal form and storage in a file.
ASSEMBLER.EXE;1	Cross-Assembler.
NEWASSEMB.EXE;1	2-Pass version of the Cross-Assembler (faster than ASSEMBLER, but without optimization and restrictions with forward references).

Instruction set descriptions for several processors; (are available to the Assembler in the library).

I8008.;3	Intel 8008
I8080.;3	Intel 8080
IM6100.;3	Intersil 6100
M6800.;3	Motorola 6800
M68000.;4	Motorola 68000
PIP2650.;3	Signetics
PIP 2650 SCMP.;3	National SC/MP
Z8000.;3	Zilog 8000

Media (Service Charge Code): 600' Magnetic Tape (MA) **Format:** VMS/BACKUP

DECUS NO: VAX-251 **Title:** FRAGMENT **Version:** February 1987

Submitted by: Susan Gorham, Atlas Specialty Steels, Welland, Ontario, Canada L3B 5R7 **Operating System:** VAX/VMS V4.3 **Source Language:** DCL **Keywords:** File Management

Abstract: This utility is a very handy tool to aid in analyzing the effectiveness of your RMS file characteristics. A batch control file is included to automate the procedure by resubmitting itself at monthly intervals.

An entire disk is scanned for all files over 1000 blocks (excluding .exe's) and the headers of these files are examined. Adjustments to this selection criteria can be easily made.

A report is produced showing by file, the current file allocation, size of first file extent (which will usually indicate size at last compression for permanent files), the files organization (seq, idx), CBT (if contiguous_best_try is set), the files extension quantity and the number of headers and extents currently in use.

After comparing monthly reports, you can track the files growth and effectiveness and base file tuning on this data.

Notes: Installation instructions included.

Documentation not available.

Media (Service Charge Code): 600' Magnetic Tape (MA) **Format:** VMS/BACKUP

DECUS NO: VAX-252 **Title:** KEYPADS **Version:** April 1987

Submitted by: Ron Burke, Westinghouse, Defense & Electronics Ctr., Baltimore, MD **Operating System:** VAX/VMS V4.X **Source Language:** DCL **Keywords:** Tools - Applications Development

Abstract: The program KEYPADS graphically displays the contents of a keypad. The keypad state name refers to which keypad state you wish to output the keypad settings. If omitted or given no value, then the current keypad state is assumed. If you use an * in this field, then the legend keypad (which outputs the name of every key in the keypad) will be output instead.

The keypad portion symbol refers to which portions of your keypad are to be displayed. If omitted or given no value, then the entire keypad is assumed. If you use a {or} (or the default) in this field, then either the left and/or right halves of the keypad are output to you. The left part of the keypad has the arrow keys, the E keys, and the F keys. The right part of the keypad is the traditional VT100 series keypad (the PF keys, the KP keys, etc.).

Media (Service Charge Code): 600' Magnetic Tape (MA) **Format:** VMS/BACKUP

DECUS NO: VAX-253 **Title:** DISK_MANAGER **Version:** April 1987

Submitted by: Bob Reardon, Schlumberger Well Services, Houston, TX **Operating System:** MicroVMS V4.4, VAX/VMS V4.4 **Source Language:** MACRO-32, VAX FORTRAN **Memory Required:** 2MB **Keywords:** Utilities - Disk - VMS

Abstract: DISK_MANAGER gathers useful disk statistics quickly and easily and presents them in a convenient format. It enables the system manager to answer such questions as:

. Which directories use the most blocks?
. Of the blocks in use, how many have not been accessed in a given number of days?
. How many blocks are being used by certain types of files, such as .TMP, .MAI etc.?
. How many files have extended headers?
. How many blocks could be made available by archiving all files that haven't been used in forty (or any other number) of days?
. How many blocks could be saved by allowing only a certain number of versions of any file?

An optional output file can be produced that is convenient for post-processing by a user-written program. Such a program is included as an example: it produces summary statistics for all accessible disks.

Media (Service Charge Code): 600' Magnetic Tape (MA) **Format:** VMS/BACKUP

DECUS NO: VAX-254 **Title:** Super EDT Emulator **Version:** V4.3, April 1987

Author: Roger Fraser

Submitted by: Gerald Marsh, Plessey Defence Systems, Christchurch, Dorset, England BH23 4JE **Operating System:** VAX/VMS V4.4 **Source Language:** TPU **Hardware Required:** VT type terminal **Keywords:** Editors

Abstract: This submission consists of TPU source for a super duper EDT emulator. It was written for use by the Technical Support Group, but soon found it's way around the user community.

It was written to give the EDT emulator some of EVE's clever bits, so that we would not have to learn a new editor at our late stage in life! There are a few other goodies like on-the-fly justification and pagination. This is useful when RUNOFF seems too involved for simple memos.

To obtain the TPU section from the source, follow the instructions at the top of the source file.

To find out the additional features type {PF1} (GOLD), then "H" after invoking. PDSTPU_ contains the TPU source which contains instructions on customizing.

Notes: Operating system VMS V4.2 or higher is required.

Restrictions: Should be 8192 to spawn subprocesses.

Media (Service Charge Code): 600' Magnetic Tape (MA) **Format:** VMS/BACKUP

DECUS NO: VAX-256 **Title:** DM/SD/WPE/COLORS **Version:** May 1987

Submitted by: Dale E. Coy, Los Alamos National Laboratory, Los Alamos, NM **Operating System:** VAX/VMS V4.5 **Source Language:** DCL, MACRO-32, VAX FORTRAN, VAX TPU **Hardware Required:** For DM$SD,

VT52 or ANSI-compliant or Digital Equipment Corporation terminal. For WPE, VT1XX or VT2XX compliant terminal. For COLORS, ReGIS compliantcolor terminal (VT241). **Keywords:** Editors, Terminal Handler

Abstract: This submission contains three sub-directories:

. DM$SD (Directory Manager and Set Default)
. WPE (Word-Processing-Like Editor)
. COLORS (VT241 Colors Management)

DM (Directory Manager V7.1A) is a utility which allows you to more easily manage, clean up, and otherwise work with your files and directory structure. DM is particularly useful if you have large numbers of files or sub directories and is helpful in encouraging users to clean up their directories (by making it easy to do so). It is invaluable for sorting through the DECUS SIG tapes after they have been loaded. Your favorite editor may be used from DM.

SD (Set Default V4.2A) is a utility which shortens the commands for SET DEFAULT and SHOW DEFAULT and expands the capabilities of the SET DEFAULT command.

WPE (Word-Processing-Like Editor V2.3) WPE is a full implementation of WPS-PLUS (TM) for editing ASCII files. WPE is an extremely powerful text editor.

Features include:

. All of WPS-PLUS that is reasonable (full function editing).
. Two-window editing.
. Multiple files.
. Bookmarks.
. Insert and examine special characters.
. Print files with special characters.
. Fix up files by removing CR/LF.
. Automatic tailoring for .COM, .HLP, .FOR, and .TPU files.
. Read-only interface (called MORE).

COLORS (Colors Management V3.1) is a suite of programs for managing and setting "default" colors for ReGIS color terminals. Having a VT241 (or other color ReGISterminal) is much more fun if you use color combinations other than red, blue, green. These programs make it easy for the user to control his/her terminal colors. A side effect is the provision of a "system default" set of pleasant colors.

Notes: Operating system VMS V4.4 or later is required. Full documentation is provided for all of the programs in .TXT, .WPL (for WPS-PLUS) and .LN03 (very fancy) forms. Two memory cartridges are required to print the .LN03 files.

Media (Service Charge Code): 600' Magnetic Tape (MA)
Format: VMS/BACKUP

NEW LIBRARY PROGRAMS AVAILABLE FOR THE PROFESSIONAL-300 SERIES OF COMPUTERS

DECUS NO: PRO-163 **Title:** PROPLOT **Version:** V3.0A August 1986

Submitted by: Ronald Getts, BF Goodrich R&D, Brecksville, OH **Operating System:** P/OS V2.0A **Source Language:** FORTRAN 77 **Memory Required:** Standard **Hardware Required:** LA50 or LVP16 (HP7475 or HP7470 if hard copy desired. EBO and color monitor helpful, but not required. **Keywords:** Graphics, Plotting

Abstract: This diskette contains software for the Pro-350 developed at FGoodrich R&D in Brecksville, OH.

PROPLOT does least squares curve fitting to polynomial equations, graphs the resulting curves on the monitor, and has provisions for hard copy to an LA50, LA100 or DEC (HP) two or six pen plotter.

This new version automatically supports color monitor and/or HP7475, HP7470 or DEC LVP16 plotters, if present. This provides color graphics support.

Data can be input from the keyboard or from a data file. The program asks the user questions regarding parameters and allows creation of data files for later recall. Scaling is automatic or controlled by the user.

Notes: Operating system P/OS V2.0 or higher required.

Media (Service Charge Code): One RX50 Diskette (JA) Format: FILES-11

DECUS NO: PRO-167 **Title:** FUNCTIONS **Version:** V1.0, March 1987

Submitted by: Michael Levin, Swampscott, MA **Operating System:** P/OS V2.0A **Source Language:** BASIC-PLUS-2 **Memory Required:** 512K **Software Required:** PRO/Tool Kit V2.0 or later and BASIC-PLUS-2. **Hardware Required:** Graphics expansion board (EBO) **Keywords:** Graphics

Abstract: This program is based on graphics which can be produced by trigonometric functions. It is in two parts. The first part allows the user to experiment with making his own designs by providing values for parameters to eight distinct functions. The other part is a very impressive graphics demo (lasting about 15 minutes) which displays some interesting effects of functions (3-D containers, etc.). The program is fully menu-driven and is ready to run from the PRO/Tool Kit (the BASIC-PLUS-2 libraries and CGL must be installed).

Notes: This program is menu-driven; the only needed documentation is obtained by pressing the HELP key. Can be used with either black and white or color monitor.

Restrictions: A ready to run task image is included. BASIC-PLUS-2 source code is not available.

Sources not included.

Media (Service Charge Code): One RX50 Diskette (JA) **Format:** FILES-11

DECUS NO: PRO-168 **Title:** Dollar Value LIFO Calculator **Version:** V2.0, March 1987

Submitted by: James & Chris Jannes, Northport, NY **Operating System:** PRO/VENIX **Source Language:** PASCAL **Memory Required:** 128KB **Software Required:** PRO/VENIX V2.0, **Hardware Required:** Pro-350 with 512KB and hard disk. **Keywords:** Accounting

Abstract: Program LIFO (last in, first out) calculates Dollar Value LIFO. It is a method of inventory calculation that uses total dollar value, not the physical quantity of goods, when calculating the value of inventory pools.

Dollar Value LIFO is the most widely used method of inventory valuation used by companies that have adopted a LIFO system. A LIFO system is advantageous because it presents a lower net income for tax purposes. In addition, the Tax Reform Act of 1986 permits the use of published indices for small businesses (revenue ¦ $5,000,000), a convenience that was not permitted in the past.

This makes the Dollar Value LIFO approach even more attractive and easier to use.

Notes: Operating System PRO/VENIX V2.0 required. The source code is not available; the executable code is provided.

Sources not included.

Media (Service Charge Code): One RX50 Diskette (JA) **Format:** VENIX

NEW LIBRARY PROGRAMS AVAILABLE FOR THE DECSYSTEM-20 FAMILY OF COMPUTERS

DECUS NO: 20-190 **Title:** KERMIT **Version:** January 1986

Author: Frank da Cruz, et al., Columbia University, New York, NY

Submitted by: Steve Attaya, Wiener Enterprises, Harahan, LA **Operating System:** CP/M V2.2, 3.0, MS/DOS V2.1, 3.1, TOPS-10 release 7.1, TOPS-20 release 6.1, VAX/VMS V4.2 **Source Language:** BASIC-PLUS2, BLISS-32, C, FORTRAN 77, FORTRAN IV, MACRO-10, MACRO-11, MACRO-32, VAX-11 PL/1. Various **Memory Required:** System Dependent **Hardware Required:** RS-232 Port **Keywords:** KERMIT

Abstract: KERMIT is a protocol for transferring sequential files between computers of all sizes over ordinary asynchronous telecommunication lines using packets, checksums and retransmission to promote data integrity. KERMIT is non-proprietary, thoroughly documented, and in wide use. The protocol and the original implementations were developed at Columbia University and have been shared with many other institutions, many of which have made significant contributions of their own. KERMIT is presently available for nearly 200 different machines and operating systems, and additional versions are always under development.

Restrictions: Not all versions implement all features.

Media (Service Charge Code): User's Manual (ED), 2400' Magnetic Tapes (PB)

NEW LIBRARY PROGRAMS AVAILABLE FOR THE DECSYSTEM-20 FAMILY OF COMPUTERS

DECUS NO: 20-191 **Title:** SNIFF **Version:** V3(2), May 1986

Submitted by: David Fordyce, Texas Instruments Incorporated, Dallas, TX **Operating System:** TOPS-20 release 5.1 **Source Language:** Rutgers' PASCAL **Software Required:** Rutgers' PASCAL-20 **Keywords:** System Management - VMS

Abstract: SNIFF identifies for the user any other detached/interactive jobs logged in under his/her user number on a DECSystem-20, and gives the user an interactive means of selectively disposing of them.

If the user does have detached jobs, for each job, SNIFF lists the number of the job, the program that the job is running, whether the job is the current job (the last job that the user logged in) or a detached job. SNIFF thenenables the user to ATTACH to a detached job, to LOGOUT a detached job, or to leave the detached job in its present state and continue with his job that is currently logged in. Several command options are supported if the user doesn't want to bother with each individual job, but wants to just purge his other jobs from the system.

Media (Service Charge Code): 600' Magnetic Tape (MA)

DECUS NO: 20-192 **Title:** MLIST **Version:** V7(25), May 1986

Submitted by: David Fordyce, Texas Instruments Incorporated, Dallas, TX **Operating System:** TOS-20 release 5.1 **Source Language:** MACRO-20, PASCAL **Software Required:** Rutgers' PASCAL-20 to rebuild part of MLIST Package. Part of Columbia University's MACRO-20 MACRO package included. **Keywords:** Mail

Abstract: MLIST provides a means of maintaining a system-wide

"database" of mailing lists (suitable in format for use in TOPS-20 electronic mail system such as MM, MS, BABYL, etc.) WITHOUT using an editor.

Media (Service Charge Code): 600' Magnetic Tape (MA)

NEW LIBRARY PROGRAMS AVAILABLE FOR THE PDP-11 COMPUTER FAMILY

DECUS NO: 11-868 **Title:** TAPUTL **Version:** V2.03, March 1987

Submitted by: Stephen Bart, Brookhaven National Laboratory, Upton, NY **Operating System:** RT-11 V5.0X **Source Language:** FORTRAN 77 **Memory Required:** 28KW **Keywords:** Utilities - Tape

Abstract: The TAPUTL utility will copy data from tape to tape, tape to disk, or disk to tape. It can also space, write end of file marks, dump, and rewind tapes. The utility assumes no particular file structure on the tape and can be used with tapes of essentially any format (including tapes with a variable record length within a file) and with records of any size up to a specified maximum (4096 words in standard version). The maximum record size can be modified easily by editing and recompiling the source code. The utility treats tapes as non RT-11 file structured media (a file structured tape can be considered non file structured) with a file being defined as the data between two end of file marks (BOT and EOT count as end of file marks).

The program will accept commands like any other RT-11 utility, either by first running the program and issuing a Command String Interpreter (CSI) comma and or by installing the program on the SY: device and using Concise Command Language (CCL) commands. The latter feature makes it extremely easy for the user to define his/her own commands with the UCL/UCF interface.

TAPUTL was written and tested most extensively with the TM11 driver (MT:) and to a lesser degree the TS11 driver (MS:), but should work with any tape driver which accepts the standard .SPFUN commands.

Restrictions: FORTRAN 77 is required to recompile the source code.

Media (Service Charge Code): One RX01 Diskette (KA) **Format:** RT-11, 600' Magnetic Tape (MA) **Format:** RT-11

DECUS NO: 11-869 **Title:** PLOT: Scientific Graphs on DEC LVP16 or HP Plotters **Version:** January 1987

Submitted by: Brian Coulter, Agricultural Institute, Wexford, Ireland **Operating System:** RSX-11M **Source Language:** FORTRAN 77 **Hardware Required:** Digital Equipment Corporation LVP16 or any Hewlett Packard Plotter. **Keywords:** Graphics, Hewlett Packard, Plotting

Abstract: Plot is an interactive data driven program for drawing graphs and maps from simple X - Y data. Headings, legends, axis names, scaling, regression lines, maps, multi color lines, dashed lines etc. may be chosen. The program gets its instructions in three ways. When the program is run, the user specifies the file name of the raw data to be plotted. They may also include switches or options to specify the size of paper, that joined points are required etc. The program asks a series of questions about limits, titles and captions etc. and then reads the data file which contains sets of X,Y,P values; i.e. the coordinates of each point with the pen or plot type to be used. Additional captions or legends may be positioned on the graph by X,Y,P, title points.

Simple plots are very easy to specify, only when the full features of the program are required will the process become a little more complex.

Media (Service Charge Code): User's Manual (EA), One RX01 Diskette (KA) **Format:** FILES-11, 600' Magnetic Tape (MA) **Format:** FILES-11

REVISIONS TO LIBRARY PROGRAMS

DECUS NO: 11-490 **Title:** TSXLIB: A FORTRAN Callable Library Implementation of EMTs for TSX-PLUS **Version:** V6.2, March 1987

Submitted by: N. A. Bourgeois, Jr., NAB Software Services, Inc., Albuquerque, NM **Operating System:** RT-11, TSX-PLUS **Source Language:** MACRO-11 **Software Required:** FORTRAN compiler **Hardware Required:** MMU to support TSX-PLUS **Keywords:** FORTRAN, Libraries - RT-11, TSX

Abstract: TSXLIB is a library of FORTRAN callable routines that implement the TSX-PLUS system services which are unique to TSX-PLUS. The library has been updated to include all TSX-PLUS unique services through TSX-PLUS V6.2.

Like RT-11, TSX-PLUS offers the MACRO-11 programmer a number of system services. These services are implemented via both the RT-11 programmed requests (for those services common to both RT-11 and TSX-PLUS) and raw EMT instructions (for those unique to TSX-PLUS). RT-11 makes its system services available to the FORTRAN programmer through the system subroutine library, SYSLIB. TSX-PLUS also honors the bulk of the service requests in the SYSLIB routines. TSXLIB, however, makes the TSX-PLUS unique EMTs available to the FORTRAN programmer.

These TSX-PLUS library routines provide facilities to support communication lines, detached jobs, device allocating and deallocating, file structured device mounting and dismounting, communication between running programs, job privileges control, job status monitoring, program performance analysis, real time program execution, shared runtime systems, shared files, special files information, spooler control, subprocess control, system status information, communication between running programs and a terminal, program control of the terminal, ODT activation mode, user name control, windowing, and several miscellaneous EMTs.

The TSXLIB distribution kit includes the MACRO-11 source modules for all the routines, a user's manual in machine readable form, an indirect command file to build the library, and the implemented library.

Changes and Improvements: Updated for TSX-PLUS V6.2.

Media (Service Charge Code): One RX02 Diskette (LA) **Format:** RT-11, 600' Magnetic Tape (MA) **Format:** RT-11

DECUS NO: VAX-208 **Title:** IMAGE **Version:** V04-05C, March 1987

Submitted by: C. J. Chapman, Philips Defence Systems, Crawley, Sussex, England RH10 2PZ **Operating System:** MicroVMS V4.4, VAX/VMS V4.4 **Source Language:** DCL, FORTRAN 77, MACRO-32 **Memory Required:** 14.8KB virtual allocation **Keywords:** System Management - VMS, Utilities - VMS

Abstract: The IMAGE utility is a system management tool that enables the Systems Manager to obtain information on system processes or user processes. IMAGE is very useful for taking a snapshot look at your system to establish what images are currently executing. IMAGE executes on both hardcopy (Digital Equipment Corporation's LA series) and video terminals (Digital Equipment Corporation's VT series ansi escape mode) continuously displaying the following data:

. User name, process id, uic, process state and type.
. Base priority, current priority, CPU min/sec (day/hr).
. Disc i/o, page faults, system/user image executing.
. Balance set, node, date, time.

Additional functions include:

. System image monitoring.
. User image monitoring using batch and detached processes with data recording and replay capability.

Release Notes are included with this utility together with the necessary files to relink between minor releases of VAX/VMS. Future releases will follow.

Notes: Operating system VMS V4.0 or later required.

Changes and Improvements: Documented in Release Notes.

Sources not included.

Media (Service Charge Code): 600' Magnetic Tape (MA) **Format:** VMS/BACKUP, or order VAX-LIB-6

DECUS NO: 11-665 **Title:** PB: Device Handler for Data I/O System 19 Prom Programmer **Version:** April 1987

Submitted by: Michael M. Iloff, Moses Electronic, D-7000 Stuttgart 1, West Germany **Operating System:** RT-11 V5.4 **Source Language:** MACRO-11 **Memory Required:** 582 words **Hardware Required:** Data I/O System 19 Universal Programmer 990-1900 **Keywords:** Device Handlers, PROM

Abstract: This handler was derived from Digital Equipment Corporation's PC11 high speed paper tape reader in order to allow for device-independent execution of file and command transfer via PIP.SAV to and from the DATA I/O SYSTEM 19 UNIVERSAL PROGRAMMER 990-1900 via a DLV11-J line at address 176520 and vector 320. It needs a running line time clock under a system generated monitor with device timeout feature for reading from the programmer device.

This version uses programmed issuing of handshake commands rather than interrupt structure in order to achieve fast response of prom programmer upon "user buffer full" recognition. This handler exploits the timeout feature substantially for block handling in "PIP".

Notes: The system is generated with a device-timeout feature.

Changes and Improvements: XM bug fixed, address set code added. See PB.MAC header. Adapted to operating system RT-11 version 5.4.

Restrictions: Running line time clock. RT-11 version 5.4 is required due to new device handler macros.

Documentation not available.

Media (Service Charge Code): One RX01 Diskette (KA) **Format:** RT-11, 600' Magnetic Tape (MA) **Format:** RT-11

DECUS NO: VAX-129 **Title:** FORTRAN Programming Tools **Version:** VIII.0, April 1987

Submitted by: A. Ragosta & L. Jurgeleit, US Army ARTA, MS 207-5, Moffett Field, CA **Operating System:** VAX/VMS V4.4 **Source Language:** DCL, FORTRAN 77, MACRO-32 **Memory Required:** Varies **Keywords:** Debugging, System Management - VMS, Tools - Software Development

Abstract: The FORTRAN Programming Tools are a series of tools used to support the development and maintenance of FORTRAN source codes. Included are a debugging aid, source code maintenance aids, print utilities, a CPU time monitoring program, a NAMELIST-like package, a general purpose filter and a library of useful, well-documented routines. These tools assist in reducing development time and encouraging high quality programs. Although intended for FORTRAN users, some of the tools can be used on data files or other programming languages.

Notes: Uses VMS Version 4.0 or later BRKTHRU System Service.

Changes and Improvements: Bug fixes, new source code clean-up program, new ASCII Filter program, efficiency improvements.

Media (Service Charge Code): 600' Magnetic Tape (MA) **Format:** VAX/ANSI, or order VAX-LIB-4

DECUS NO: VAX-146 **Title:** WATCHDOG **Version:** V4.2-10, MAY 1987

Submitted by: George Walrod III **Operating System:** VAX/VMS V3.X-4.0-4.5, **Source Language:** MACRO-11 **Keywords:** Security, System Management - VMS, Utilities - VMS

Abstract: WATCHDOG is a program which monitors an interactive process for inactivity. A process is logged out after a defined interval. An inactive process is indicated by no change in CPU time and no buffered I/O count within a defined interval. Messages will be sent to the inactive process at a defined interval until the maximum inactive time limit is reached. A final message is sent to the user and an optional message is sent to the central operator making note that a user has been stopped. Another option includes ignoring a group of users. Many options exist and are documented. You should enjoy the comments made by the developer.

Changes and Improvements: Totally rewritten in VAX/VMS MACRO. Support virtual terminal driver's disconnect function. Special exceptions on the basis of username, account, identifier, UIC or terminal (wildcards may be used). Exceptions allow special users to have their own start/stop values as well as options.

Media (Service Charge Code): 600' Magnetic Tape (MA) **Format:** VMS/BACKUP, or order VAX-LIB-4

SUBMITTING ARTICLES TO THE HMS SIG NEWSLETTER

The purpose of the HMS SIG newsletter is to serve as a forum to share information related to DEC hardware with the members of the SIG. As such, the existence of the newsletter is entirely dependent on your contributions. If you have an HHK item, a better or safer way to do something, product news, a tutorial article of general interest, etc., we are interested in publishing it in the newsletter. It is intended that the HMS newsletter be published at least four times a year.

You can submit material to either the editor, Bill Walker, or the assistant editor, Carmen Wiseman. We can accept submissions in a wide variety of formats:

- Items can be sent to the assistant editor on VMS format RX50s or IBM PC format 5 1/4" floppies.
- The editor can handle just about any reasonable media, but prefers RT-11 format diskettes.
- Hard copy, like cash, is always acceptable. If it is camera-ready it will save us a lot of typing, but we don't insist on it. You can also use the "Hardware Submission Form," which you will find in the "Questionnaire" section of the combined newsletters.
- Those of you that have access to DCS can send things to WALKER or WISEMAN. DCS is usually checked on a daily basis.
- You can reach the editor on CompuServe as "Bill Walker 71066,24" or via EasyLink mailbox 62752448. You can reach the assistant editor via EasyLink mailbox 62960090 (be sure to say ATTN: or TO: Carmen Wiseman somewhere in the message).

In any event, if you have anything to submit, send it! If it is a mess, but we can read it, we will get it in the newsletter somehow. Finally, if you have any question about submitting material, call one of us. The telephone numbers are listed below.

Contributions can be sent to:

William K. Walker
Monsanto Research Corp.
P.O. Box 32 A-152
Miamisburg, OH 45342
(513) 865-3557 (work)
(513) 426-7094/0344 (home)

OR

==

Carmen D. Wiseman
Digital Review
Prudential Tower, Suite 1390
800 Boylston Street
Boston, MA 02199
(617) 375-4361

DECUS U.S. CHAPTER
SUBSCRIPTION SERVICE SIGS NEWSLETTERS ORDER FORM
(U.S. Members Only)

As a member of DECUS U.S. Chapter, you are entitled to contribute and subscribe to the DECUS monthly publication, **SIGs Newsletters**. You also have the opportunity to subscribe to the Symposia Proceedings which are a compilation of the reports from various speakers at the U.S. National DECUS Symposia.

- **No Purchase Orders will be accepted.**
- The order form below must be used as an invoice.
- All checks must be made payable to DECUS.
- All orders MUST be paid in full.
- Minimum of $25.00 for orders placed via a credit card.
- No refunds will be made.
- The address provided below will be used for all DECUS mailings; i.e. Membership, Subscription Service and Symposia.
- SIGs Newsletters Price is for a one-year subscription beginning the month following receipt of payment.

Name _____ DECUS Member # _____

Company _____

Address _____

_____

City _____ State _____ Zip _____

Telephone # ( ) _____

Subscription Service Offering	Unit Price	Qty	Total
SIGs Newsletters	$35.00	_____	_____
Spring '86 Proceedings (SP6)	15.00	_____	_____
Fall '86 Proceedings (FA6)	15.00	_____	_____
Spring '87 Proceedings (SP7)	15.00	_____	_____
Fall '87 Proceedings (FA7)	15.00	_____	_____
		Total Amt	$ _____

☐ MASTERCARD ☐ VISA ☐ DINERS CLUB/CARTE BLANCHE

Credit Card # _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ Expiration Date _____

I understand that there will be no refunds even if I decide to cancel my subscription.

Signature _____

For Digital Employees Only

Badge # _____ Cost Center _____

Cost Center Mgr. Name _____ Cost Center Mgr. Signature _____

MAIL TO: Subscription Service, DECUS (BP02), 219 Boston Post Road, Marlboro, MA 01752-1850, (617)480-3418.

FOR DECUS OFFICE ONLY

Check Number _____ Bank Number _____

Amount $ _____

DECUS U.S. CHAPTER APPLICATION FOR MEMBERSHIP

☐ New Membership ☐ Update to current membership profile Current DECUS Member.# ___ ___ ___ ___ ___ ___

Please provide a complete mailing address, include zip code in accordance with postal regulations for your locality.

Are you an employee of Digital Equipment Corporation? ☐ YES ☐ NO

NOTE: Please print clearly or type!

Name: _____
(First) (Middle Initial) (Last/Family Name)

Company: _____

Address: _____

_____

_____

City/Town/State/Zip: _____

_____

Telephone: Home ( ) _____ Work ( ) _____

How Did You Learn About DECUS? Please Check Applicable Item.

1 ☐ ANOTHER DECUS MEMBER	4 ☐ DIGITAL SALES	13 ☐ LOCAL USERS GROUP
2 ☐ SYMPOSIA	5 ☐ HARDWARE PACKAGE	14 ☐ SPECIAL INTEREST GROUP
8 ☐ DECUS CHAPTER OFFICE	6 ☐ SOFTWARE PACKAGE	7 ☐ SOFTWARE DISPATCH (Digital Newsletter)
10 ☐ DIGITAL STORE	12 ☐ ADVERTISING	

_____

Do you wish to be included in mailings conducted by Digital (for marketing purposes etc.?) ☐Permission ☐Refusal

Type Of Digital Hardware Used: Please Check Those Applicable To You.

_____

20 ☐ DECMATE	52 ☐ LSI-11	21 ☐ PROFESSIONAL	5 ☐ WPS-8
82 ☐ DECSYSTEM-10	3 ☐ PDP-8 FAMILY	22 ☐ RAINBOW	51 ☐ WPS-11
83 ☐ DECSYSTEM-20	50 ☐ PDP-11 FAMILY	54 ☐ VAX FAMILY	

Major Operating Systems? Languages Used: Please Check Those Applicable To You.

1 ☐ ADA	26 ☐ CORAL-66	47 ☐ FOCAL	67 ☐ OS/8	109 ☐ RT-11
2 ☐ ALGOL	28 ☐ COS	48 ☐ FORTRAN	68 ☐ PASCAL	97 ☐ TECO
5 ☐ APL	34 ☐ DATATRIEVE	51 ☐ GAMMA	72 ☐ PL-11	70 ☐ TOPS-10
7 ☐ BASIC	35 ☐ DBMS	110 ☐ IAS	92 ☐ RPG	71 ☐ TOPS-20
17 ☐ BLISS	38 ☐ DECNET	53 ☐ IQL	81 ☐ RSTS/E	111 ☐ ULTRIX/UNIX
19 ☐ C	43 ☐ DIBOL	58 ☐ MACRO	83 ☐ RSX	104 ☐ VMS
22 ☐ COBOL	45 ☐ DOS-11	65 ☐ MUMPS	91 ☐ RMS	107 ☐ WPS-8

Type Of Business (Environment)/Computer Applications
Please Check That Which Best Describes Your Business/Application.

21 ☐ ACCOUNTANCY
7 ☐ BANK
64 ☐ BUSINESS/COMMERCIAL
74 ☐ BUSINESS/INFORMATION SYSTEMS
57 ☐ CHEMISTRY
54 ☐ CLINICAL LABORATORY
63 ☐ COMPUTATION
11 ☐ CONSUMER ELECTRONICS
18 ☐ CONSULTANT
72 ☐ DATA ACQUISITION
52 ☐ DATA COMMUNICATIONS
13 ☐ DATA PROCESSING SERVICES
71 ☐ DATA REDUCTION
17 ☐ DIGITAL EMPLOYEE-ENGINEERING
15 ☐ DIGITAL EMPLOYEE-MARKETING
16 ☐ DIGITAL EMPLOYEE-SERVICE GROUP
60 ☐ EDUCATIONAL ADMINISTRATION
1 ☐ EDUCATION/PRIMARY
2 ☐ EDUCATION/SECONDARY
61 ☐ EDUCATION-TECHNOLOGY
3 ☐ EDUCATION/UNIVERSITY
67 ☐ ENGINEERING
65 ☐ FINANCE/ACCOUNTING
77 ☐ GOVERNMENT
75 ☐ GRAPHICS
4 ☐ HOSPITAL
62 ☐ INDUSTRIAL
55 ☐ LABORATORY/SCIENTIFIC
14 ☐ LIBRARY
58 ☐ LIFE SCIENCES
70 ☐ MANUFACTURING
79 ☐ MARKETING
59 ☐ MEDICAL RESEARCH
6 ☐ MILITARY INSTALLATION
23 ☐ NUMERICAL CONTROL
68 ☐ OEM-COMMERCIAL
78 ☐ OEM-TECHNICAL
56 ☐ PHYSICAL SCIENCES
20 ☐ RESEARCH/DEVELOPMENT
10 ☐ RETAIL
73 ☐ SOFTWARE DEVELOPMENT
53 ☐ TELECOMMUNICATIONS
19 ☐ TELEPHONE/UTILITIES
51 ☐ TIMESHARING
80 ☐ TRAINING/INSTRUCTION
66 ☐ TYPESETTING/PUBLICATION

Special Interest Groups (SIGs) Enrollment
I Wish To Participate In The Following DECUS U. S. Chapter Special Interest Groups.

3 ☐ ARTIFICIAL INTELLIGENCE
7 ☐ BUSINESS APPLICATIONS
2 ☐ COMMERCIAL LANGUAGES
6 ☐ DATA MGMT. SYSTEMS
31 ☐ DAARC (LABS)
5 ☐ DATATRIEVE/4 GL
8 ☐ EDUSIG
10 ☐ GRAPHICS APPLICATIONS
11 ☐ HARDWARE AND MICRO
35 ☐ IAS
27 ☐ LARGE SYSTEMS
16 ☐ L & T
14 ☐ MUMPS
15 ☐ NETWORKS
34 ☐ OFFICE AUTOMATION
36 ☐ PERSONAL COMPUTER
18 ☐ RSTS/E
17 ☐ RSX
19 ☐ RT-11
32 ☐ SITE MGMT. & TRNG
21 ☐ UNISIG
26 ☐ VAX

Job Title/Position - Please Check:

1 ☐ CORPORATE STAFF
2 ☐ DIVISION OR DEPARTMENT STAFF
3 ☐ SYSTEMS ANALYSIS
4 ☐ APPLICATIONS PROGRAMMING
5 ☐ SYSTEMS ANALYSIS/PROGRAMMING
6 ☐ OPERATING SYSTEM PROGRAMMING
7 ☐ DATABASE ADMINISTRATION
8 ☐ DATA COMMUNICATIONS/TELECOMMUNICATIONS
9 ☐ COMPUTER OPERATIONS
10 ☐ PRODUCTION CONTROL
101 ☐ CORPORATE DIRECTOR OF DP/MIS
102 ☐ ADMINISTRATIVE ASSISTANT
103 ☐ TECHNICAL ASSISTANT
104 ☐ SERVICES COORDINATOR
105 ☐ MANAGER
106 ☐ ANALYST
107 ☐ PROGRAMMER
108 ☐ DATABASE MANAGER
109 ☐ DATABASE ADMINISTRATOR
110 ☐ MANAGER OF DP OPERATIONS

Citizen of The United States of America? ☐ YES ☐ NO Country:_____

Signature:_____ Date:_____

Forward To: DECUS U. S. Chapter
Digital Equipment Computer Users Society
Membership Processing Group
219 Boston Post Road, BP02
Marlboro, MA 01752-1850
Phone: (617)480-3418
DTN: 8-296-3418

STEERING COMMITTEE LISTS

ARTIFICIAL INTELLIGENCE SIG

CHAIR
Cheryl Jalbert
JCC
128 West Broadway
Granville, OH 43023
(614) 587-0157

VICE-CHAIR
OPS5 WORKING GROUP CHAIR
Don Rosenthal
Space Telescope Science Inst.
Homewood Campus
Baltimore, MD 21218
(301) 338-4844

NEWSLETTER TASK FORCE CHAIR
ADMINISTRATIVE ASSISTANCE
Becky Wise
Amdalh CSD
2200 North Greenville Ave.
Richardson, TX 75081
(214) 699-9500 x 272

NEWSLETTER EDITOR
Terry Shannon
Digital Review
Prudential Tower
800 Boylston St. Suite 1390
Boston, MA 02199
(617) 375-4321

SYMPOSIA COORDINATOR
Pam Vavra
Hughes Aircraft EDSG
P.O. Box 902 E52/D220
El Segundo, CA 90245-0902
(213) 616-7071

MEMBERSHIP COORDINATOR
SUITE COORDINATOR
Chris Goddard
Simpact Associates
9210 Skypark Court
San Diego, CA 92123
(619) 565-1865

SESSION NOTE EDITOR
George Humfeld
Naval Sea Systems Command
PMS 350 ED Dept of the Navy
Washington, DC 20362-5101
(202) 692-0137

ASS'T SESSION NOTES EDITOR
David Frydenlund

STORE REPRESENTATIVE
Sally Townsend
Inst. Defense Analysis
1801 N. Beauregard St.
Alexandria, VA 22311
(703) 845-2122

PSS REPRESENTATIVE
Tom Viana

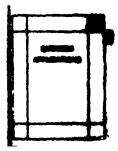
PUBLIC DOMAIN SOFTWARE TF CHAIR
Jim Sims

SITE COORDINATOR, NASHVILLE
Dennis Clark

REPORTER TO THE UPDATE, DAILY
Bill Lennon

DEC COUNTERPART
Art Beane
Hudson, MA

MEMBERS-AT-LARGE
David Slater
George Winkler
Jeff Fox
John Williamson
Wayne Graves
Matt Mathews



BUSINESS APPLICATIONS SIG

CHAIRMAN
George Dyer
Gallaudet University
800 Florida Ave. NE-EMG Bldg
Washington, DC 20002
(202) 651-5300

COMMUNICATIONS REPRESENTATIVE
SESSION NOTE EDITOR
STORE REPRESENTATIVE
NEWSLETTER EDITOR
Steve Lacativa
Price Waterhouse
153 East 53rd Street
New York, NY 10022
(212) 371-2000 x 3107

SYMPOSIA COORDINATOR
Steve Simek
IRT Corporation
3030 Callan Road
San Diego, CA 92121
(619) 450-4343

LRP AND MARKETING COORDINATOR
Arnold L. Epstein
D-M Computer Consultants
Rolling Meadows, IL 60008
(312) 394-8889

LIBRARY REPRESENTATIVE
David Hittner
Projects Unlimited
3680 Wyse Road
Dayton, OH 45414
(513) 890-1800

CL SIG LIAISON
Becky Burkes-Ham

DMS SIG LIAISON Joe Sciuto MEMBERS-AT-LARGE
Robert D. Lazenby
Dixie Beer Dist., Inc.
Louisville, KY

Robert Kayne
Gallaudet College
Washington, DC

Ray Evanson
Paragon Data Systems
Winona, MN

DEC COUNTERPARTS
Sue Yarger
Merrimack, NH

Ray Arsenault
Merrimack, NH

COMMERCIAL LANGUAGES SIG

CHAIR
Dena Shelton
Cullinet Software Inc.
2860 Zanker Rd., Suite 206
San Jose, CA 95134
(408) 434-6636

SYMPOSIA COORDINATOR
Ray Strackbein
Palm Desert, CA

LIBRARY COORDINATOR
Philip Hunt
System Industries
Milpitas, CA

COMMUNICATIONS REPRESENTATIVE
NEWSLETTER EDITOR
Ted Bear
Ramtek
2211 Lawson Lane
Santa Clara, CA 95950
(408) 988-2211

SESSION NOTE EDITOR
Bob Van Keuren
Userware International, Inc.
2235 Meyers Avenue
Escondido, CA 92025
(619) 745-6006

ASS'T NEWSLETTER EDITORS
Beverly Welborne
Diocese of Gary
LaPorte, IN

Kevin Cullen
VITA-Mix Corp.
Holmstead Falls, OH

Daniel Cook
Userware International, Inc.
Escondido, CA

BASIC Working Group Members
Mark Hartman
Jadtec Computer Group
Orange, CA

Rocky Hayden
Userware International, Inc.
Escondido, CA

Bill Tabor
Computer Products
Pompano Beach, FL

Ted Bear
Ramtek
2211 Lawson Lane
Santa Clara, CA 95950
(408) 988-2211

COBOL WORKING GROUP MEMBERS
Keith Batzel
Crowe, Chizek & Co.
South Bend, IN

MaryAnne Feerick
RDBS Inc.
Kernersville, NC

Bill Leroy
The Software House, Inc.
Atlanta, GA

Herbert J. Matthews IV
ManTech international Cor.
Alexandria, VA

Jim Welborne
Crowe, Chizek & Co.
South Bend, IN

Jim Wilson
Pfizer Inc. QC Div.
Terre Haute, IN

DIBOL WORKING GROUP MEMBERS
Neil Baldridge
CompuShare
Lubbock, TX

Becky Burkes-Ham

Colin Chambers
Software Ireland Rep. Inc.
Portola Valley, CA

Mark Derrick
WAAY-TY
Huntsville, AL

Gary A.P. Kohls
Milwaukee, WI

Ken Lidster
Disc
Sacramento, CA

Kenneth M. Schilling
MCBA
Montrose, CA

Marty Schultz
Omtool Inc.
Tewksbury, MA

Marty Zergiebel
The Software Gallery
Brookfield, CT

RPG WORKING GROUP MEMBERS
Keith Batzel
Crowe, Chizek & Co.
South Bend, IN

DEC COUNTERPARTS
Tom Harris
Nashua, NH

Jim Totten
Nashua, NH

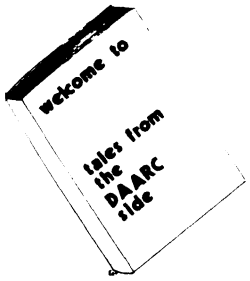
Joe Mulvey
Nashua, NH

Shirley Ann Stern
Nashua, NH

STANDARDS REPRESENTATIVES
BASIC
Dan Esbensen
Touch Technologies, Inc.
Escondido, CA

COBOL
Bruce Gaarder
Macalester College
St. Paul, MN

DIBOL
Eli Szklanka
TEC
Newton, MA



DAARC SIG

CHAIRMAN
James Deck
Inland Steel Research Lab.
3001 East Columbus Drive
East Chicago, IL 46312
(219) 392-5613

SYMPOSIA COORDINATOR
Mack Overton
FDA
Chicago, IL

COMMUNICATIONS REPRESENTATIVE
NEWSLETTER EDITOR
Ellen Reilly
William H. Rorer
500 Virginia Drive
Ft. Washington, PA 19034
(215) 628-6547

DEC COUNTERPART
Bill Forbes
Marlboro, MA

HARDWARE & INTERFACING
Peter Clout
Los Alamos National Lab
Los Alamos, NM

MATH STATISTICS & ANALYSIS
Herbert J. Gould
C.C.F.A. Univ. of Ill. Medical Ctr.
Chicago, IL

PROCESS CONTROL-INDUSTRIAL AUTOMATION
Bill Tippie
Kinetic Systems Corp.
Lockport, IL

RS-1
George Winkler
CPC International
Argo, IL

DATA MANAGEMENT SYSTEMS SIG

CHAIRMAN
Joseph F. Sciuto
Army Research Institute
5001 Eisenhower Ave
Alexandria, VA 22333
(202) 274-8221

COMPTROLLER
Alan Schultz
Land Bank National DP Center
7300 Woolworth Ave
Omaha, NE 68124
(402) 397-5040

SYMPOSIA COORDINATOR
Keith Hare
JCC
P.O. Box 463
Granville, OH 43023
(614) 587-0157

SYMPOSIA COORDINATOR
Barbara Mann
TRW
Redondo Beach, CA
(213) 532-2211

COMMUNICATIONS REPRESENTATIVE
NEWSLETTER EDITOR
Mark S. Crego
Mantech Int'l Corp.
2121 Eisenhower Ave.
Alexandria, VA 22314
(703) 838-5677

SESSION NOTE EDITOR
Mark Morgan
Farm Credit Banks
P.O. Box 141
Springfield, MA 01102
(413) 732-9721

MEMBERSHIP COORDINATOR
Vacant

PRODUCT DIRECTION COMMITTEE
PAST SIG CHAIRMAN
Steve Pacheco
Ship Analytics
North Stonington, CT 06359
(203) 535-3092

WORKING GROUP COORDINATOR/
DATABASE WORKING GROUP
Jim Perkins
PSC, Inc.
20 Kimball Ave., Suite 305
Shelburne, VT 05401
(802) 863-8825

FORMS WORKING GROUP
ANSI STANDARDS COORDINATOR
Paul W. Plum, Jr.
Lukens Steel Company
Coatesville, PA
(215) 383-2024

NON-DIGITAL WORKING GROUP
Doug Dickey
GTE Government Systems
1700 Research Blvd
Rockville, MD 20850
(301) 294-8400

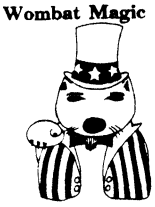
RMS WORKING GROUP COORDINATOR
Allen Jay Bennett
Lear Siegler Rapistan
555 Plymouth N.E.
Grand Rapids, MI 49505
(616) 451-6429

PRE-SYMPOSIUM SEMINAR COORDINATOR
Rocky Hayden
Userware International
Escondido, CA
(619) 745-6006

AI SIG LIAISON
David Slater
Institute for Defense Analysis
Alexandria, VA
(703) 845-2200

DATATRIEVE SIG LIAISON
William L. Tabor
W.L. Tabor, Inc.
Coral Springs, FL
(305) 755-7895

DEC COUNTERPART
Wendy Herman
Nashua, NH
(603) 881-2494



DATATRIEVE/4GL SIG

CHAIRMAN
Joe H. Gallagher
Research Medical Center
2316 East Meyer Blvd
Kansas City, MO 64132
(816) 276-4235

PAST SIG CHAIRMAN
Larry Jasmann
U.S. Coast Guard
10067 Marshall Pond Rd.
Burke, VA 22015
(202)267-2626

SYMPOSIA COORDINATOR
T.C. Wooi
E.I. duPont DeNemours & Co.
Engineering Dept.
P.O. BOX 6090.
Newark, DE 19714-6090

ASS'T SYMPOSIA REPRESENTATIVE
Lisa M. Pratt
Vitro Corporation
Nuwes Code 3144
Keyport, WA 98345
(206) 396-2501

COMMUNICATIONS REPRESENTATIVE
NEWSLETTER EDITOR
LRRP
Donald E. Stern, Jr.
Warner Lambert Company
10 Webster Road
Milford, CT 06460
(203) 783-0238

ASSOCIATE EDITOR
Steve Cordiviola
Kentucky Geological Survey
311 Breckinridge Hall
Lexington, KY 40506
(606) 257-5863

Pasquale (Pat) F. Scopelliti
Corning Glass Works
Mail Stop MP-RO-01-1
Corning, New York 14831
(607) 974-4496

Lorey B. Kimmel
Thomson-CGR Medical Corp.
10150 Old Columbia Road
Columbia, Maryland 21046
(301) 290-8757

ASST. VOLUNTEER COORDINATOR
Susan Krentz
NKF Engineering
12200 Sunrise Valey Dr.
Reston, VA 22091
(703) 620-0900

PRE-SYMPOSIA SEMINARS
Dana Schwartz
15719 Millbrook Lane
Laurel, MD 20707
(301) 859-6277

SESSION NOTES EDITOR
Wanda Anderson
SRI International MS: pN341
333 Ravenswood Avenue
Menlo Park, CA 94025
(415) 859-2577

CAMPGROUND
Bert Roseberry
Commandant (G-APA-1)
2100 2nd Street, S.W.
Washington, DC 20593-0001
(202) 267-2629

WW EDITOR
PIR COORDINATOR
LRRP
Philip A. Naecker
Consulting Engineer
3011 N. Mount Curve Ave.
Altadena, CA 91001
(818) 791-0945

DEC COUNTERPARTS
DATATRIEVE
Mary Ann Fitzhugh
110 Spit Brook Road
ZK2-2/M28
Nashua, NH 03060
(603) 881-2329

ARTIST & LIBRARY REP.
Bart Z. Lederman
I.T.T. World Communications
67 Broad Street (28th Floor)
New Yor, NY 10004
(212) 607-2657

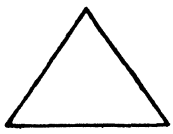
POWERHOUSE W/G CHAIR
Randall R. Barth
Searle Fred Vietri
TSO Financial Corp.
Five TSO Financial Center
Three Hundred Welsh Road
Horsham, PA 19044-2009
(215) 784-0520

MEMBER LRPP COMMITTEE
Michael G. Graham
Sanders Associates, Inc.
NAM 3-1, C.S. 2044
Nashua, NH 03061-2004
(603) 885-5206

DMS & CL SIG LIAISON
William Tabor
Computer Products
Pompano Beach, FL

SMARTSTAR WORKING GROUP CHAIR
Leslie Byars
Corning Electronics
3900 Electronics Drive
Raleigh, NC 27604
(919) 878-6301

ACCENT-R USER GROUP LIAISON
Winston Tellis
Fairfield University
North Benson Road
Fairfield, CT 06430
(203) 255-5411



EDUSIG

CHAIRMAN
Robert A Shive, Jr.
Millsaps College
Jackson, MS 39210
(601) 354-5201

SYMPOSIA COORDINATOR
Mary Jac Reed
Off Comp Based Instruction
University of Delaware
305 Willard Hall
Newark, DE 19716
(302) 451-8161

COMMUNICATIONS REPRESENTATIVE
Robert W. McCarley
Millsaps College
Jackson, MS 39210
(601) 354-5201

NEWSLETTER EDITOR
Fred Bell
Taft College
29 Emmons Park Drive
P.O. Box 1437
Taft, CA 93267
(805) 763-4282

PSS COORDINATOR
VAX SYSTEMS SIG LIAISON
Donald C. Fuhr
Tuskegee Institute
Tuskegee Institute, AL 36088
(205) 727-8242

ADMINSTRATIVE APPLICATIONS COORD.
Dave Cothrun
Taft College
29 Emmons Pk Drive
P.O. Box 1437
Taft, CA 93268
(805) 763-4282

SESSION NOTE EDITOR
Paula Barnes
Guilford College
5800 West Friendly Avenue
Greensboro, NC 17410
(919) 292-5511

DEC COUNTERPART
Gary Finerty
Marlboro, MA

DECUS Graphics Applications SIG

GRAPHICS APPLICATIONS SIG

CHAIRMAN
William Kramer
NAS Systems Engineering Branch
NASA Ames Research Center
Moffett Field, CA 94035
(415) 694-5189

SYMPOSIA COORDINATOR
Bijoy Misra
Smithsonian Institution
60 Gordon St., MS39
Cambridge, MA 02138
(617) 495-7392

COMMUNICATIONS REPRESENTATIVE
NEWSLETTER EDITOR
Michael Anton
Schlumberger
P.O. Box 591293
Houston, TX 77259-1293
(713) 928-4838

ASSOCIATE NEWSLETTER EDITOR
Charles D. Carter
Huntington Alloys, Inc.
Technology Dept.
P.O. Box 1958
Huntington, WV 25720
(304) 526-5721

WORKSTATION WORKING GROUP COORD.
Bob McCormick
Video Communications, Inc.
1325 Springfield Street
Feeding Hills, MA 01030
(413) 786-7955

ENGINEERING GRAPHICS WORKING GROUP COORD.
Eric Rehm
Gonzaga University
SPOCAD
E 502 Boone
Spokane, WA 99258
(509) 484-6814

SESSION NOTE EDITOR
Carol Schwob
Florida Altantic University
Academic Computing
500 N.W. 20th Street
Boca Raton, FL 33431
(305) 393-2640

LIBRARY COORDINATOR
Mike McPherson
Michigan University
269 Engineering Bldg.
East Lansing, MI 48824
(517) 353-9769

STANDARDS COORDINATOR
Jim Flatten
Ames Lab
2581 Metals Dev.
Ames, IA 50011
(515) 294-7908

VOLUNTEER COORDINATOR
Dick McCurdy
University of Florida
Room 216, Larsen Hall
Gainsville, FL 32611
(904) 392-4915

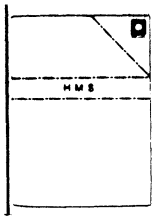
LIBRARY COMMITTEE
James M. Turner
Saber Technology
San Jose, CA

DEC COUNTERPART
Rick Berzle
Spit Brook, NH

INFORMATION OFFICER
Mike York
Boeing Computer Services
P.O. Box 24346 M/S 03-73
Seattle, WA 98124
(206) 342-1442

HUMAN INTERFACE WORKING GROUP COORD.
Dottie Elliott
Northrop Services Inc.
P.O. Box 12313
Research Triangle PK, NC 27709
(919) 541-1300

DATA DISPLAY WORKING GROUP COORD.
Joy Williams
Eaton Corp.
P.O. Box 766
Southfield, MI 48037



HARDWARE MICRO SIG

CHAIRMAN
VAX SIG LIAISON
Thomas J. Provost
MIT/LNS Bates Linac Facility
Middletown, MA

PRODUCT PLANNING COORDINATOR
George Hamma
Synergistic Technology
Cupertino, CA

SYMPOSIA COORDINATOR
PRE-SYMPOSIUM SEMINAR COORDINATOR
Mike Allen
Lawrence Livermore Nat'l Lab.
Livermore, CA

COMMUNICATIONS COORDINATOR
John G. Hayes
Information Systems
South Central Bell
Birmingham, AL

NEWSLETTER EDITOR
William K. Walker
Monsanto Research Corp.
Miamisburg, OH

ASSISTANT EDITOR
Carmen D. Wiseman
Digital Review
Boston, MA

SESSION NOTE EDITOR
DAARC SIG LIAISON
Bill Tippie
Kinetic Systems Corp.
Lockport, IL

STANDARDS COORDINATOR
CAMAC WORKING GROUP COORDINATOR
Peter Clout
Los Alamos National Lab
los Alamos, NM

LUG COORDINATOR
Gregg Giesler
Los Alamos Science Lab
Los Alamos, NM

TOEM (CHIPS & BOARDS)
Jack J. Peterson
Horizon Data Systems
Richmond, VA

HHK (HARDWARE HINTS & KINKS)
Wayne Kesling
Monsanto Research Cor.
Miamisburg, OH

UNIBUS HARDWARE
Ron Bogue
LIV Aerospace & Defense Co.
Dallas, TX

PERFORMANCE MEASUREMENT COORD.
William Wallace
600 W. Washington Street
Peoria, IL

CSS COORDINATOR
Pratap Gohel
E.I. duPont
Ingleside, TX

NETWORKS SIG LIAISON
Sandra Traylor
Target Systems
Yorba Linda, CA

VAX SIG LIAISON
Dave Schmidt
5100 Centre Avenue
Pittsburgh, PA

UNISIG LIAISON
Jim Livingston
1 Results Way
Cupertino, CA

SITE SIG LIAISON
Emily Kitchen
A.H. Robins Co.
Richmond, VA

RT-11 SIG LIAISON
Gary Sallee
Sallee Software Consulting
yorba Linda, CA

RSX SIG LIAISON
Hans Jung
Associated Press
New York, NY

MEMBERS-AT-LARGE
Mike Rembis
American Dade
Costa Mesa, CA

Hans Dahlke
Richland, WA

Jim Cutler
EDS Tower
16533 Evergreen
Southfield, MI

DEC COUNTERPARTS TERMINALS
Nina Abramson
Maynard, MA

TOEM (Chips & Boards)
Art Bigler
Marlboro, MA

DIAGNOSTIC
George D. Cooke
Maynard, MA

STORAGE
Marilyn Fedele
Maynard, MA

MSD (Micro Systems Develp.)
Roy Rodgers
Maynard, MA

PRINTER PRODUCTS
Frank Orlando
Maynard, MA

DECUS EUROPE LIAISON
Hans Zoller

IAS SIG

CHAIRMAN
Mike Robitaille
Grumman - CTEC, Inc.
6862 Elm Street
McLean, VA 22101
(703) 556-7400 x 431

NEWSLETTER EDITOR
Frank R. Borger
Radiation Therapy
Michael Reese Medical Center
Lake Shore Drive @ 31st St.
Chicago, IL 60616
(312) 791-2515

WHIMS COORDINATOR
Kathleen Anderson
Eaton Information Management
System Division
Hampton, VA
(804) 326-1941

RSX LIAISON
Ray French
Boeing Computer Services
Seattle, WA
(206) 655-6228

MEMBER-AT-LARGE
Doug Reno
Abbot Laboratories
North Chicago, IL
(312) 937-7504

CHAIRMAN EMERITUS
Bob Curley
Division of Medical Physics
University of Pennsylvania
Philadelphia, PA
(215) 662-3083

SYMPOSIA COORDINATOR
Lynda L. Roenicke
Special Systems Branch
Naval Ocean Systems Center
271 Cataline Blvd Code 742
San Diego, CA
(619) 225-7569

LIBRARY COORDINATOR
Bob Schuldt
INCO Inc.
McLean, VA

MEMBER-AT-LARGE
Kerry Wyckoff
Salt Lake City, UT

DEC COUNTERPART
Nancyfaye Autenzio
Stow, MA
(617) 496-9606

LANGUAGES AND TOOLS SIG

CHAIRMAN
Sam Whidden
American Mathematical Society
201 Charles St.
P.O. Box 6248
Providence, RI 02940
(401) 272-9500

VICE CHAIR
SYMPOSIA COORDINATOR
Earl Cory
Eaton Corporation
31717 Latienda Dr.
Westlake Village, CA 91359
(818) 706-5385

COMMUNICATIONS REPRESENTATIVE
STORE REPRESENTATIVE
CHAIR, TECH. PROD. OF DOC. WORKING GROUP
Howard Holcombe
RCA
Front & Cooper Sts.
Camden, NJ 08055
(609) 338-4946

NEWSLETTER EDITOR
Al Folsom, Jr.
Fischer & Porter Co.
E. County Line Rd.
Warminster, PA 18974
(215) 674-7154

SESSION NOTES EDITOR
Mark Katz
GTE Government Systems
100 First Avenue
Waltham, MA 02154
(617) 466-3437

AUSTRALIAN L&T INTERFACE
Gordon Brimble
Bldg. 180 Labs Area
Defence Research Centre
Box 2151 GPO
Adelaide, S.A. Australia 5001
(61)(8)259-6119

GAPSIG INTERFACE
Jim Flatten
Ames Lab
304 Metallurgy
Ames, IA 50001
(515) 294-4823

INTERSIG COORDINATOR
Dorothy Geiger
Wollongong Logistics Group
49 Showers Drive #451
Mountain View, CA 94040
(415) 948-1003

EUROPEAN METHODS
L&T INTERFACE
Bernd Gliss
Max-Planck-Institute
Heisenbergstra Be 1
7000 Stuttgart 80, W. Germany
(711) 686-0251

DMS & DTR LIAISON
Keith Hare
JCC
P.O. Box 381
128 West Broadway
Granville, OH 43023
(614) 587-0157

DEC COUNTERPART
Celeste LaRock
110 Spit Brook Road
ZK02-3/Q08
Nashua, NH 03062

PAST CHAIR
PRODUCTIVITY TOOLS COORDINATOR
Kathy Hornbach
Digital Equipment Corporation
110 Spit Brook Rd., ZK02-2/R55
Nashua, NH 03062
(603) 881-2505

CHAIR TECO WORKING GROUP
Mark J. Hyde
Advanced Computing Services
209 Ardsley Drive
DeWitt, NY 13214
(315) 446-7223

CHAIR, BASIC WORKING GROUP
Stephen C. Jackson
SCJ Consulting, Inc.
7260 University Avenue N.E.
Suite 105
Minneapolis, MN 55432
(612) 571-8430

CHAIR, CONFIG. MGMT. WORKING GROUP
Mark Alan Kidwell
Texas Instruments Inc.
P.O. Box 869305 M/S 8435
Plano, TX 75086
(214) 575-3547

DEC COUNTERPART PDP-11 SOFTWARE
Joe Mulvey
Digital Equipment Corp. ,ZK01-3/J10
110 Spit Brook Road
Nashua, NH 03062-2642
(603) 881-1218

LISP/AI COORDINATOR
Don Rosenthal
Space Telescope Science Institute
Homewood Campus
Baltimore, MD 21218
(301) 338-4844

LIBRARY REPRESENTATIVE
SIG TAPE LIBRARIAN
PUBLIC DOMAN SOFTWARE W/G CHAIR
Tony Scandora
Argonne National Laboratory
CMT 205
Argonne, IL 60439
(312) 972-7541

CHAIR, C WORKING GROUP
James Maves
Eaton Corp, IMSD
31717 Latienda Drive
Box 5009
Westlake Village, CA 91359
(818) 706-5375

STANDARDS COORDINATOR
TOOLS INTEGRATION W/G CHAIR
Jay Wiley
Bechtel Power Corp
12400 East Imperial Highway
Norwalk, CA 90650
(213) 807-4016

ASS'T TO CHAIR
TEX/LATEX COORDINATOR
JR Westmoreland
Custom Software Products
UTAH Power & Light
1407 West North Temple
Annex 6/208
Salt Lake City, UT 84116
(801) 535-4784

RECORDING SECRETARY
Melodee Westmoreland
Custom Software Products
1456 E. Hilda Drive
Fruit Heights, UT 84037
(801) 533-2350

SESSION CHAIRS COORDINATOR
Gary C. Lelvis
IMSL
2500 Park West Tower One
2500 City West Blvd.
Houston, TX 77042-3020
(713) 782-6060

CHAIR, FORTRAN WORKING GROUP
Scott Krusemark
8473 Daisywood Ave N.W.
North Canton, OH 44720
(261) 499-6251

VOLUNTEERS COORDINATOR
Louise Wholey
Measurex Corp
1 Results Way
Cupertino, CA 95014
(408) 255-1500 x 5571

CHAIR, MACRO WORKING GROUP
CHAIR, BLISS WORKING GROUP
Gerald Lester
Computerized Processes Unlim.
2901 Houma Blvd. Suite 5
Metairie, LA 70006
(504) 889-2784

HUMAN INTERFACE COORDINATOR
CHAIR, COBOL GENERATOR WORKING GROUP
CHAIR, TRAINING & ED SERV. WORKING GROUP
Jim Wilson
Pfizer
QC Division
P.O. Box 88
Terre Haute, IN 47808
(812) 299-2121 x 271

VICE CHAIR
SYMPOSIUM LOGISTICS COORDINATOR
PRE-SYMPOSIUM SEMINAR COORDINATOR
Terry Medlin
Survey Sampling, Inc.
1 Post Road
Westport, CT 06432
(203) 255-4200

CHAIR, DIBOL WORKING GROUP
Bruce Mebust
Vicom
9713 Valley View Road
Eden Prairie, MN 55344
(612) 944-7135

MASTERS COORDINATOR
CHAIR, PROJECT MGMT. WORKING GROUP
Dena Shelton
Cullinet Software Inc.
2860 Zanker Rd. Suite 206
San Jose, CA 95134
(408) 434-6636

APL WORKING GROUP CHAIR
Bob Van Keuren
UserWare International, Inc.
2235 Meyers Ave.
Escondido, CA 92025
(619) 745-6006

WISHLIST COORDINATOR
Shava Nerad
MIT
77 Mass Avenue W91-219A
Cambridge, MA 02139
(617) 253-7438

WORKING GROUPS COORD.
ASST. SYMPOSIUM LOGISTICS COORD.
Joseph Pollizzi, III
Space Telescope Science Institute
3700 San Martin Drive
Homewood Campus
Baltimore, MD 21218
(301) 338-4901

CHAIR, MODULA II WORKING GROUP
Jack Davis
Phillips Home Interactive Systems
111 North Shore Drive
Knoxville, TN 37919
(615) 558-5206

CHAIR, SCAN WORKING GROUP
CHAIR, PL/1 WORKING GROUP
David Ream
Lexi-Comp
26173 Tallwood Drive
N. Almstead, OH 44070
(216) 777-0098
(216) 468-0744

CHAIR, PASCAL WORKING GROUP
Wayne Sewell
E-Systems, Garland Div.
Box 660023, MS 53730
Dallas, TX 75266-0023
(214) 272-0515 x3553

CHAIR, PDP-11 LAYERED PROD. W/G
William I. Tabor
W.I. Tabor, Inc.
11652 N.W. 30th St.
Coral Springs, FL 33065
(305) 755-7895

CHAIR, SOFTWARE METRICS W/G
Michael S. Terragas
LDS Church
50 E. North Temple, 27th Floor
Salt Lake City, UT 84150
(801) 531-3246

CHAIR, RPG WORKING GROUP
Charles Williamson
Hargray Telephone Co.
P.O. Box 5519
Hilton Head Is., SC 29938
(803) 686-1204

CHAIR COBOL WORKING GROUP
Edward W. Woodward
SAIC
10210 Campus Point Drive
M/S # 24
San Diego, CA 92121
(619) 546-3758

STEERING COMMITTEE
CHAIR, LARGE SYS. MAIN. W/G
George Scott
Computer Sciences Corp.
304 West Route #38, P.O. Box N
Moorestown, NJ 08057
(609) 234-1100

STEERING COMMITTEE MEMBERS

Ted Bear
Ramtek
2211 Lawson Lane
Santa Clara, CA 95950
(408) 988-2211

Ray Strackbein
P.O. Box 7487
Mission Hills, CA 91346
(805) 254-8811

Barry C. Breen
Sundstrand Data Control, Inc.
1500, N.E. 36th St.
P.O. Box 97001
Redmond, WA 98073-9701
(206) 885-8436

LARGE SYSTEMS

CHAIR
Leslie Maltz
Stevens Institute of Technology
Computer Center
Hoboken, NJ 07030
(201) 420-5478
BITNET: LMALTZ@SITVXB;
ARPANET: SIT.MALTZ@CU20B.COLUMBIA.EDU

ASST. SYMPOSIUM COORDINATOR
Robert C.McQueen
Stevens Institute of Technology
Computer Center
Hoboken, NJ 07030
(201) 420-5454
BITNET: RMCQUEEN@SITVXB;
ARPANET: SIT.MCQUEEN@CU20B.COLUMBIA.EDU

COMMUNICATIONS REPRESENTATIVE
NEWSLETTER EDITOR
Clyde Poole
The University of Texas at Austin
Department of Computer Science
Taylor Hall 2.124
Austin, TX 78712-1188
(512) 471-9551
ARPANET: etp@sally.utexas.edu

MENU COORDINATOR
OPEN

HARDWARE COORDINATOR
Clive Dawson
Microelectronics & Computer Technology Corp.
9430 Research Blvd.
Echelon Bldg. #1, Suite 200
Austin, TX 78759
(512) 343-0860
ARPANET/CSNET: CLIVE @ MCC. COM

LANGUAGES COORDINATOR
OPEN

SYMPOSIUM COORDINATOR
Betsy Ramsey
American Mathematical Society
P.O. Box 6248
Providence, RI 02940
(410) 272-9500 x 295
ARPANET: EWR@XX.LCS.MIT.EDU

SYSTEM SOFTWARE COORDINATOR
OPEN

SPECIAL PROJECTS COORDINATOR
E.F. Berkley Shands
Washington University
Department of Computer Science
P.O.Box 1045
St. Louis, MO 63136
(314) 889-6636
UUCP: BERKLEY@WUCS.UUCP

NETWORKS COORDINATOR
Don Kassebaum
Computation Center
University of Texas at Austin
Austin, TX 78712
(512) 471-3241
ARPANET:CC.KASSEBAUM@A20.CC.UTEXAS.EDU

SITE SIG LIAISON
Gary C. Bremer
Emerson Electric Co.
Electronics and Space Division
8100 W. Florissant
St. Louis, MO 63136
(314) 553-4448/4447

SPECIAL PROJECTS
INFORMATION CENTERS COORDINATOR
Ralph M. Bradshaw
Johnson & Johnson
Research & Scientific Services
Management Information Center
Raritan, NJ 08869-1489
(201) 685-3434

COMMERCIAL SYSTEMS COORD.
Robert L. Boyd
General Electric Co.
Microelectronics Center
P.O. Box 13049
Research Triangle Park, NC 27709-3049
(919) 549-3627
ARPANET:RLB@RTPARK.GE.COM

DEC COUNTERPARTS
Dave Braithwaite
Marlboro, MA

Rich Whitman
Marlboro, MA

Reed Powell
Marlboro, MA

MUMPS SIG

CHAIRMAN
Mark Berryman
Plessey Peripheral Systems
Irvine, CA

SYMPOSIA COORDINATOR
Chris Richardson
Computer Sciences Corp.
Ridgecrest, CA

NEWSLETTER EDITOR
Mark J. Hyde
Advanced Computing Services
DeWitt, NY

VAX LIAISON
Coyett A.J. Dese
VA DM&S Verification & Dev. Ctr.
San Francisco, CA

DEC COUNTERPARTS
Beatrice Walther
Marlboro, MA

Diane Brown
Marlboro, MA

NETWORKS SIG

CHAIRMAN
Bill Brindley
Naval Security Group Command
(202) 282-0527

COMMUNICATIONS REPRESENTATIVE
Bob Gustafson
Northeast Utilities
(203) 665-5082

NEWSLETTER EDITOR
Vickie Hess
2510 Limestone Lane
Garland, TX 75040
(214) 495-7353

SYMPOSIA COORDINATOR
Sandy Traylor
Target Systems, Inc.
(714) 921-0112

TECHNOLOGY COORDINATOR
Bill Hancock
(817) 261-2283

STANDARDS COORDINATOR
Jim Ebright
Software Results Corp.
(614) 267-2203

DEC COUNTERPART
Carole Greenfield

OFFICE AUTOMATION SIG

CHAIR
Katherine "Kit" Trimm
Pivotal, Inc
Tucson, AZ
(602) 886-5563

VICE CHAIRMAN
Ralph Bradshaw
Johnson and Johnson
Raritan, NJ
(201) 685-3434

COMMUNICATIONS REPRESENTATIVE
E. Catherine Ditamore
ARA Services
Philadelphia, PA
(215) 238-3638

SYMPOSIA COORDINATOR
Mitch Brown
GenRad, Ind.
Waltham, MA
(617) 369-4400 x3052

NEW MEMBER COORDINATOR
Tricia Cross
American Mathematical Society
P.O. Box 6248
Providence, RI 02940
(401) 272-9500

BOF COORDINATOR
Ray Kaplan
PIVOTAL, Inc.
Tucson, AZ
(602) 886-5563

NEWSLETTER EDITOR
Therese LeBlanc
T.M. LeBlanc & Assoc.
Wheeling, IL
(312) 459-1784

LIBRARY
Bob Hassinger
Liberty Mutual Research Center
Hopkington, MA
(617) 435-9061

OA TAPE COORDINATOR
Mary Jane Boiling
Foreign Mission Board
3806 Monument Avenue
Richmond, VA 23230
(804) 353-0151

SYMPOSIA ASSISTANT
Sal Gianni
Northeast Utilities
Hartford, CT
(203) 665-5652

STORE COORDINATOR
Mike Jackson
Air Force Operational
Test and Evaluation Center
Kirtland AFB, NM
(505) 846-5641

PERSONAL COMPUTER SIG LIAISON
Cheryl Johnson
Grinnell College
Grinnell, IA
(515) 236-2570

OA LUG COORDINATOR
Tom Orlowski
American Council on Education
1 DuPont Circle (Suite 110)
Washington, DC
(202)939-9371

OA SIG COORDINATOR
Joe Whatley
Neilson Media Research
375 Patricia Avenue
Dunedin, FL 33528
(813)734-5473

PERSONAL COMPUTER SIG

CHAIR
Barbara Maaskant
UT Health Science Center
7703 Floyd Curl Drive
San Antonio, TX 78284
(512) 691-7351

PRO WORKING GROUP CHAIRMAN
Thomas R. Hintz
Univ. of Florida
IFAS Computer Network
Bldg. 120
Gainesville, FL 32611
(904) 392-5180

DECMATE WORKING GROUP CHAIRMAN
Cheryl Johnson
Grinnell College
P.O. Box 805
Grinnell, IA 50112-0813
(515) 236-2570

RAINBOW WORKING GROUP CHAIRMAN
Lynn Jarrett
Union Tribune Publishing Co.
Computer Systems
350 Camino De LaReina
San Diego, CA 92108
(619) 293-1130

VAXMATE WORKING GROUP CHAIRMAN
Fritz Howard
Eastman Kodak Company
901 Elmgrove Road, 1E45-LP
Rochester, NY 14650
(716) 724-5331

LIBRARY REPRESENTATIVE
LIBRARIAN
Ron S. Hafner
Hafner and Associates
P.O. Box 2924
2499 Wellingham Dr.
Livermore, CA 94550
(415) 422-2149

COMMUNICATIONS REPRESENTATIVE
NEWSLETTER EDITOR
Kenneth LeFebvre
Sytek Inc.
19 Church St.
P.O. Box 128
Berea, OH 44017
(216) 243-1613

NEWSLETTER CONTRIBUTING EDITOR (PRO)
Gary Rice
McDonnell Douglas
5555 Garden Grove Blvd.
MS: K20 77/200
Westminster, CA 92683
(714) 952-6582

PRE-SYMPOSIA SEMINAR COORDINATOR
Vince Perriello
Crosfield Composition Systems
570 Taxter Road
Elmsford, NY 10523
(914) 592-3600

SYMPOSIA COORDINATOR
Rick Eliopoulis
5258 Vickie Drive
San Diego, CA 92109
(619) 225-7867

SESSION NOTE EDITOR
Alan Bruns
Allied Electronics
401 E. 8th Street
Fort Worth, TX 76102
(817) 336-5401

CAMPGROUND COORDINATOR
Jim Wilson
Ntl Tech Inst for the Deaf
Rochester Inst. of Tech
P.O. Box 9887
Rochester, NY14623
(716) 475-6241

MEMBERS-AT-LARGE
Michael Bowers
Univ. of California
Animal Science Department
Davis, CA 95616
(916) 752-6136

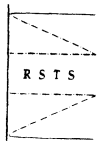
Theodore Needleman
Hardcopy Magazine
Seldin Publishing, Inc.
1061 S. Melrose, Suite D
Placentia, CA 92670

Russ Wertenberg
Sandia National Labs
Division 8352
Livermore, CA 94550
(415) 422-2663

DEC COUNTERPARTS
PRO
Lin Olsen
Maynard, MA

DECmate
Louise Brandwein
Merrimack, NH

Rainbow & VAXMATE
Katrina Holman
Littleton, MA



RSTS SIG

CHAIRMAN
Charles Mustain
Stark County School system
Louisville, OH

SYMPOSIA COORDINATOR
Scott W. Pandorf
Kittle's Home Furnishings
Indianapolis, IN

ASS'T SYMPOSIA COORDINATOR
Wef Fleischman
Software Techniques
Cypress, CA

NEWSLETTER EDITOR
Open

LIBRARY REPRESENTATIVE
Susan Abercrombie
Ventrex Laboratories Inc.
Portland, ME

PRE-SYMPOSIA SEMINAR COORDINATOR
Bruce Gaarder
Macalester College
St. Paul, MN

WISH LISTS COORDINATOR
Neal E. Goldsmith
Software Techniques, Inc.
Cypress, CA

VICE CHAIRMAN
WISH LISTS & TAPE COPY COORDINATOR
Philip Hunt
System Industries
Milpitas, CA

EDUSIG LIAISON
George Wyncott
Purdue University Computer Center
W. Lafayette, IN

RSTS PRODUCT PLANNING COORDINATOR
Errol E. Ethier
Information Design and Management, Inc.
Shrewsbury, MA

MEMBERS-AT-LARGE
Ed Beadel
Instructional Computer Center
Oswego, Ny

Scott Daily
Great Lakes Chemical Corp.
W. Lafayette, IN

Mark Gilmore
Cal State University
Long Beach, CA

Mark Hartman
Jadtec Computer Group
Orange, CA

Jeff Killeen
Information Design & Management
Hopedale, MA

Newton J. Munson
Rochester Institute of Technology
Rochester, NY

RSX SIG

CHAIRMAN
Dan Eisner
Perkin-Elmer Corp.
Garden Grove, CA

SYMPOSIA COORDINATOR
Rick Sharpe
Toledo Edison
Toledo, OH

PRE-SYMPOSIUM SEMINAR COORDINATOR
Hans Jung
Associated Press
New York, NY

COMMUNICATIONS REPRESENTATIVE
Jay Allen Bennett
Lear Siegler Rapistan
Grand Rapids, MI

NEWSLETTER EDITOR
MULTI-PROCESSORS WORKING GROUP COORDINATOR
Bruce Mitchell
Machine Intelligence & Industry Magin
Byron, MIN

STORE COORDINATOR
Jim Hopp
Carlton Financial Computation
South Bend, IN

SESSION NOTE EDITOR
Burt Janz
Northern Telecom Inc.
Concord, NH

LIBRARIAN
Glenn Everhart
Mt. Holly, NJ

CAMPGROUND COORDINATOR
Jerry Ethington
Prolifix Inc.
Frankfort, KY

DEC COUNTERPARTS
Lin Olsen
Nashua, NH

Dick Day
Nashua, NH

WORKING GROUP COORDINATOR
Sharon Johnson
Epidemiology
Minneapolis, MN

WORKING GROUP CHAIR
Evan Kudlajev
Philadelphia Electric Co.
Philadelphia, PA

RSX GROUP CHAIR SOFTWARE CLINIC COORD.
Roy S. Maull
U.S. Air Force
Offutt AFB, NE

SOFTWARE CLINIC COORDINATOR
Bruce Zielinski
RCS
Moorestown, NJ

VOLUNTEER COORDINATOR
Gary Maxwell
U.S. Geological Survey
Menlo Park, CA

SRD WORKING GROUP COORDINATOR
Bob Turkelson
Goddard Space Flight Center
Greenbelt, MD

ACCOUNTING & PERFORMANCE WORKING GROUP COORD.
Denny Walthers
American McGaw
Irvine, CA

MENU COORDINATOR
Ed Cetron
Center for Biomedical Design
Salt Lake City, UT

MEMBERS-AT-LARGE
Jim McGlinchey
Warrenton, PA

Jim Neeland
Hughes Research Labs.
Malibu, CA

Anthony E. Scandora, Jr.
Argonne National Laboratory
Argonne, IL

Ralph Stamerjohn
Creve Coeur, MO

RT-11 SIG

CHAIRMAN
John T. Rasted
JTR Associates
58 Rasted Lane
Meriden, CT 06450
(203) 634-1632

COM. COM VOTING REP.
COBOL CONTACT
Bill Leroy
The Software House, Inc.
P.O. Box 52661
Atlanta, GA 30355-0661
(404) 231-1484

Standards Coordinator
Robert Roddy
Naval Ship Research Ctr.
Bethesda, MD 20084
(301) 227-1724

MACRO CONTACT
Nick Bourgeois
NAB Software Services Inc.
P.O. Box 20009
Albuquerque, NM 87154
(505) 298-2346

NEWSLETTER EDITOR
TECO CONTACT
PRODUCT PLANNING CONTACT
John M. Crowell
Multiware, Inc.
2121-B Second St. Suite 107
Davis, CA 95616
(916) 756-3291

NETWORKING CONTACE
Jim Crapuchettes
Omnex Corp.
2483 Old Middlefield Way
Mountain View, CA 94043
(415) 966-8400

WISH LIST CONTACT
UNIX/ULTRIX CONTACT
Bradford Lubell
L.A. Heart Lab, UCLA
10833 Le Conte Avenue
Los Angeles, CA 90024-1760
(213) 206-6119

TSX & C CONTACT
Jack Peterson
Horizon Data Systems
P.O. Box 29028
Richmond, VA 23229
(804) 740-9244

RUNOFF CONTACT
John Davis
Naval Ship Research Center
Code 2950
Bethesda, MD 20084
(203) 265-2226

LUG CONTACT
Ned Rhodes
Software Systems Group
2001 North Kenilworth St.
Arlington, VA 22205
(703) 534-2297

PERSONAL COMPUTERS
Dennis V. Jensen
AMES Labs. ISU/USDOE
310 Metallurgy
Ames, Iowa 50011

SYMPOSIA COORDINATOR
Milton Campbell

TAPE COPY GENERATION
TAPE COPY DISTRIBUTION
RT DECUS LIBRARY CONTACT
Tom Shinal
Syntropic Technology
P.O. Box 198
Waterford, VA 22190
(703) 882-3000

PRE-SYMPOSIUM SEMINAR
RT-11 SUITE MANAGER
Bruce Sidlinger
4335 N.W. Loop 410, #209
San Antonio, TX 78229
(512) DIG-ITAL

BASIC CONTACT
Ralston Barnard
Div 7523
Sandia Labs
Alburquerque, NM 87185
(505) 844-5115

PRO RT-11 & HARDWARE
Bill Walker
Monsanto Research Corp.
P.O. Box 32, A-152
Miamisburg, OH 45342
(513) 865-3557

FORTRAN CONTACT
Robert Walraven
Multiware, Inc.
2121-B 2nd St. Suite 107
Davis, CA 95616
(515) 294-4823

OTHER LANGUAGES
Gary Sallee
19912 Fernglen Drive
Yorba Linda, CA 92686
(714) 970-2864

SITE SIG

CHAIRMAN
DMS SIG Liason
Larry W. Hicks
Relational Database services
P.O. Box 644
121 S. Main St.
Kernersville, NC 27285-0644
(919) 996-4882

SYMPOSIA COORDINATOR
Sue Abercrombie
48 Malilly Rd.
Portland, ME 04103
(207) 772-2837

SESSION NOTE EDITOR
LARGE SYSTEMS SIG LIAISON
Gary Bremer
Emerson Electric Co.
8100 W. Florisant
St. Louis, MO. 63136
(314) 553-4448

NEWSLETTER EDITOR
NETWORKS SIG LIAISON
OA SIG LIAISON
Gregory N. Brooks
Washington University
Behavior Research Labs
1420 Grattan St.
St. Louis, MO. 63104
(314) 241-7600 ext. 257

LIBRARY COORDINATOR
RSTS SIG LIAISON
Timothy Frazer
Specialized Bicycle Components
15130 Concord Circle #77
Morgan Hill, CA. 95037
(408) 779-6229

HARDWARE COORDINATOR
HMS SIG Liason
Emily Kitchen
A.H. Robins Co.
1211 Sherwood Ave. RT-2
Richmond, VA. 23220
(804) 257-2925

COMMUNICATIONS COMMITTEE REPRESENTATIVE
AI SIG Liason
Terry C. Shannon
Digital Review
160 State St.
6th Floor
Boston, MA. 02109
(617) 367-7190

PRE-SYMPOSIA SEMINAR COORDINATOR
Phillip Ventura

STAFF MANAGEMENT
Adam Zavitski
Simmonds Precision ICD
3100 Highland Blvd.
Raleigh, NC. 27625
(919) 872-9500

MEMBERS-AT-LARGE
Ann Goergen
Texas Instruments
13510 N. Central
M/S 437
Dallas, TX. 75266
(214) 995-4629

HMS SIG Liason
RT SIG Liason
David Hunt
Lawrence Livermore National Lal
MS L-230
P.O. Box 808
Livermore CA. 94550
(802) 656-3190

Gary Siftar
Digital Equipment Corporation
Tulsa, OK.

DEC COUNTERPARTS
Joe Allen
Stow MA.

Lil Holloway
Bedford MA.

Susan Porada
Marlboro, MA.

UNISIG

CHAIRMAN
Kurt Reisler
Hadron Incorporated
9990 Lee Highway
Fairfax, VA 22030
(703) 359-6100
decvax!seismo!hadron!klr

SYMPOSIA COORDINATOR
Stephen M. Lazarus
Ford Aerospace, MS X-20
3939 Fabian Way
Paulo Alto, CA 94304
(415) 852-4203
ihnp4!fortune!wdll!sml

SESSION NOTE EDITOR
Sam Kimery
716 Second Street NW
Rochester, MN 55901
(507) 281-1505

COMMUNICATIONS REPRESENTATIVE
NEWSLETTER EDITOR
James W. Livingston
Measurex Corporation
1 Results Way
Cupertino, CA 95014
(408) 255-1500 x 5556
ihnp4!decwrl!jwl

ADMINISTRATIVE DAEMON
Dorothy Geiger
The Wollongong Group
49 Showers Drive, 451
Mountain View, CA 94040
(415) 948-1003
ihnp4!decwrl!dgeiger

TAPE LIBRARIAN
Carl Lowenstein
Marine Physical Laboratory
Scripps Institute of Oc'graphy, P-004
LaJolla, CA 92093
(619) 294-2678
(ihnp4 decvax akgua dcdwest ucbvax)
!sdcsvax!nmlvax!cdl

USENET LIAISON
Joe Kelsey
FlexComm Corporation
711 Powell Avenue, SW
Renton, WA 98055
allegra!fluke!joe

STANDARDS COORDINATOR
Jeff Gilliam
National Semiconductor
2900 Semiconductor Drive, MS C2303
Santa Clara, CA 95051
(408) 721-3801
ihnp4!nsc!voder!jeff

MINISTER WITHOUT PORTFOLIO
Norman Wilson
Bell Laboratories, 2C-529
600 Mountain Avenue
Murray Hill, NJ 07974
(201) 582-2842
(decvax ihnp4)!research!norman

DEC COUNTERPART
Roseann Maclean
Merrimack, NH
(603) 884-5702
decvax!maclean

VAX SYSTEMS SIG

COMMUNICATIONS REPRESENTATIVE (CORE)
Don Golden
Shell Oil Company
Westhollow Research Center
P.O. Box 1380, Room D2158
Houston, TX 77251-1380

SYMPOSIUM COORDINATOR (CORE)
Jack Cundiff
Horry-Georgetwon
P.O. Box 1966
Conway, SC 29526

ASS'T SYMPOSIUM COORD. (CORE)
David Cossey
Computer Center
Union College
Schenectady, NY 12308

HANDOUT COORDINATOR
Ken Johnson
Meridian Technology Corp.
P.O. Box 2006
St. Louis, MO 63011

NEWSLETTER EDITOR
Lawrence J. Kilgallen
Box 81, MIT Station
Cambridge, MA 02139-0901

VICE CHAIR
WORKING GROUP COORDINATOR (CORE)
Ross W. Miller
Online Data Processing, Inc.
N 637 Hamilton
Spokane, WA 99202

LIBRARIAN (CORE)
Joe L. Bingham
Mantech International
2320 Mill Road
Alexandria, VA 22314

VAXcluster WORKING GROUP
Thomas Linscomb
Computation Center
University of Texas
Austin, TX 78712

NETWORK WORKING GROUP
Bill Hancock
Dimension Data Systems, Inc.
P.O. Box 13557
Arlington, TX 76094-0557

MicroVAX WORKING GROUP
Ray Kaplan
Pivotal, Inc.
P.O. Box 32647
Tucson, AZ 85715-32647
(601) 886-5563

SYSTEM IMPROVEMENT REQUEST (CORE)
Mark D. Oakley
Battelle Columbus Labs
Room 11-6-008
505 King Avenue
Columbus, OH 43201-2669

MULTIPROCESSOR WORKING GROUP
Eugene Pal
U.S. Army
CAORA (ATOR-CAT-C)
Fort Leavenworth, KA

PRE-SYMPOSIUM SEMINAR COORDINATOR (CORE)
Susan Rehse
Lockheed Missiles
Bldg 101, Org. 19-50
Sunnyvale, CA 94088-3504

VAXeln WORKING GROUP
Bob Robbins
Array Computer Consultants
539 Timberridge Dr.
Longwood, FL 32779-2646

REAL TIME/PROCESS CONTROL WORKING GP
Larry Robertson
Bear Computer Systems Inc.
5651 Case Avenue
North Hollywood, CA

LUG COORDINATOR
STORE REPRESENTATIVE (CORE)
David Schmidt
Management Sciences Associates
5100 Centre Avenue
Pittsburgh, PA 15232

FIELD SERVICE WORKING GROUP
D. Slater
Institute for Defense Analysis
1801 North Beauregard Street
Alexandria, VA 22314

LARGE SYSTEMS INTEGRATION WORKING GP
Leslie Maltz
Stevens Institute of Tech.
Computer Center
Hoboken, NJ 07030

VOLUNTEER COORDINATOR
Elizabeth Bailey
222 CEB
Tennessee Valley Authority
Muscle Shoals, AL 35660

COMMERCIAL WORKING GROUP
Bob Boyd
GE Microelectronics Center
MS 2P-04
Post Office Box 13409
Research Triangle Park, NC 27709

SECURITY
C. Douglas Brown
Sandia Labs
Division 2644
P.O. Box 5800
Albuquerque, NM 87185

MicroVAX WORKING GROUP
Barbara Dow-Pleines
Magic One
1971 Mount Pleasant Road
San Jose, CA 95148
(408) 238-0861

MIGRATION AND HOST DEVELOPMENT
VAXintosh WORKING GROUP
Jim Downward
KMS Fusion Incorporated
3941 Research Park Drive
Ann Arbor, MI 48106

REAL TIME/PROCESS CONTROL WORKING GP
Dennis Frayne
McDonnell Douglas
5301 Dolsa Avenue
Huntington Beach, CA 92646

INTERNALS WORKING GROUP
Carl E. Friedberg
In House Systems
165 William Street
New York, NY 10038

Ask the WOMBAT WIZARD
Submission Form

To submit a problem to the WIZARD, please fill out the form below and send it to:

WW Editor, Philip A. Naecker
Consulting Software Engineer
3011 North Mount Curve Avenue
Altadena, CA 91001
USA

Name:_____ DECUS Membership No. _____

Affiliation:_____

Address:_____

_____

_____

Telephone Number:_____

Statement of Problem:_____

_____

_____

_____

_____

_____

Please following the following guidelines when submitting support material:

1. If you are trying to demonstrate a method or a concept, please simplify the procedures, records, and other information to the shortest form possible.

2. Annotate your attachments. Simple comments or hand-written notes ("Everything worked until I added this statement.") go a long way toward identifying the problem.

3. Keep an exact copy of what you send. And number the pages on both copies. But send everything that is related to your question, even remotely.

4. If you would like a direct response or would like your materials returned, please don't forget to include a stamped, self-addressed envelope large enough to hold the materials you send.

DATATRIEVE/4GL SIG
Product Improvement Request Submission Form

Submittor: DECUS Membership Number:
Address: Firm:

Phone: Product or Products:

How to write a PIR

A PIR should be directed at a specific product or group of products. Be sure to give the full name of the product(s) and version numbers if applicable. Describe the functionality you would like to see in as complete terms as possible. Don't assume that the PIR editors or software developers know how it is done in some other software product - state specifically how you want the software to function. Provide justification of your request and give an example of its use. If you can, suggest a possible implementation of your request.

Abstract: (Please limit to one or two short sentences.)

Description and Examples: (Use additional pages as necessary.)

[Put my name and address on reverse side, thus:]

PIR Editor, Philip A. Naecker
Consulting Software Engineer
3011 North Mount Curve Avenue
Altadena, CA 91001
USA

DTR/4GL SIG Spring 1987 PIR Ballot

DECUS Membership Number: __ __ __ __ __ __

CPU Types (Check all that apply):

VAXes___ PDP-11's___ DECsystems___ Other (Specify)____________

Application Types at your site (Check all that apply):

___ Business EDP/MIS
___ Education
___ Office Automation
___ Other (Specify)____________________
___ Software Development
___ Engineering/Scientific
___ Service Bureau

Number of years using computers:____ **Number of years using 4GL's:** ____

Products Used (Check all that apply):

___ DTR-11 ___ VAX-DTR ___ CDD ___ TDMS ___ DBMS(any)
___ FMS ___ RS1 ___ Oracle ___ Ingress ___ Rdb
___ Others (Specify)____

____

PIR Number	Points	PIR Number	Points
____	____	____	____
____	____	____	____
____	____	____	____
____	____	____	____
____	____	____	____
____	____	____	____
____	____	____	____
____	____	____	____
____	____	____	____
____	____	____	____
____	____	____	____
____	____	____	____
____	____	____	____
____	____	____	____
____	____	____	____
____	____	____	____
____	____	____	____
____	____	____	____
____	____	____	____
____	____	____	____
____	____	____	____

Be sure to return your ballot by July 1, 1987

Return to:

Philip A. Naecker
3011 N. Mount Curve Ave.
Altadena, CA 91001

H M S S I G

HARDWARE SUBMISSION FORM -- A SIG INFORMATION INTERCHANGE

Message _____

_____

_____

_____

_____

_____

_____

_____

_____

_____

_____

_____

Contact
Name _____

Address _____

_____

_____

Telephone _____

Type of equipment _____

SUBMIT ANY TYPE OF HARDWARE PROBLEMS AND/OR FIXES.

SEND TO:

William K. Walker
Monsanto Research Corp.
P.O. Box 32 A-152
Miamisburg, OH 45342

OR
==

Carmen D. Wiseman
Digital Review
Prudential Tower, Suite 1390
800 Boylston Street
Boston, MA 02199

IAS WHIMS

WHAT: (Describe your WHIM) (Please print or type)

WHY: (Describe the reason for the WHIM)

HOW: (Make any suggestions for a possible implementation

Name: _____

Company: _____

_____

Address: _____

_____

_____

Phone: _____

Please mail to:

Kathleen M. Anderson
EATON Information Management
Systems Division
2017 Cunningham Drive
Suite 208
Hampton, Virginia 23666

Phone: (804) 326-1941

IAS SIG MEMBERSHIP SURVEY

Name: _____

Address: _____

_____

_____

_____

Telephone: _____

Current Hardware: (Include number and type of processors, mass storage devices, communication devices, etc.)

IAS Release: (Indicate release of IAS under which these systems are running)

Software: (Indicate software running on these systems, i.e., DECNET, Decus C, etc.)

Application: (Indicate the type of application running on the system.)

Contacts: Would you be willing to be placed on a list of contacts? If so, what areas?

Features: Do you have any features which you would like IAS to include?

Any further comments?

fold

Frank R. Borger
Michael Reese Medical Center
Dept of Radiation Therapy
Lake Shore Drive at 31st Street
Chicago Il 60616

Languages & Tools SIG

MASTERS APPLICATION

Name: _____ **Title** _____

Company: _____

Address: _____

_____

_____ **Phone:** ( ) _____

Network Address: _____ **Date:** _____

The Languages & Tools SIG has established the designation "LANGUAGES AND TOOLS MASTER", to be applied to selected, qualified people willing to share their expertise in various subjects with others. Masters are people who are knowledgeable enough in one or more languages or tools to be comfortable answering questions about them. The qualifications of an L&T Master are: expertise in a specific area, a willingness to have his/her name published as a Master, and a willingness to volunteer services in different ways. Each product may have several Masters, and there is an overall Masters Coordinator who is a member of the L&T Steering Committee.

Masters are asked to serve other users (and, under some circumstances, DEC), as a resource on products within their competence. In addition to being listed in the L&T Masters Directory (published in the newsletter) as available for occasional telephone consultation, Masters may act as 'Doctors' at Symposium Clinics, present Symposium sessions on the products of interest to them, field test products, interact with DEC product managers when appropriate, or act as a reference for a product for Digital salespeople. Especially on mature products, the SIG is anxious for knowledgeable users to offer product tutorial sessions at Symposia, and Masters can be of great help here. At Symposia, Masters will wear an identifying button bearing the legend "Ask Me About....." and the name of the language or tool in which he/she specializes.

If you'd like to serve as an L&T Master, please mark the products on which you are willing to answer questions with an **"M"** (for Master). Please mark any other products running at your site with an **"A"** (for "also running") to provide users with a broader picture of your facilities. (Although not an L&T product, Mumps is included here at the request of the Mumps SIG as a service to Mumps users). You may request removal of your name from the Masters Directory at any time, although you may continue to be listed for a month or two, because of publication lead times.

I am qualified to act as an L&T Master for the following products: ☐ Mumps

☐ Debug	☐ Bliss	☐ CMS	☐ TPU	☐ C	☐ Test Manager
☐ Pascal	☐ Basic	☐ MMS	☐ EVE	☐ Ada[1]	☐ Runoff & DSR
☐ Fortran	☐ Cobol	☐ LSE	☐ EDT	☐ APL	☐ TEX & LATEX
☐ Document	☐ Dibol	☐ SCA	☐ TECO	☐ RPG	☐ Cobol Generator
☐ VAX Notes	☐ Emacs	☐ PCA	☐ PL/I	☐ Scan	☐ Software Project Mgr

Briefly describe your experience with those you checked. _____

_____

_____

_____

How long have you held your present position? _____

Are you able to attend at least one symposium each year? _____

Users are encouraged to seek assistance with products by calling appropriate Masters listed in the Directory. As a Master, your name and telephone number will be published in the Masters Directory, and users will call on you for limited help from time to time. Please check, below, any additional activities you might do:

☐ Field-test new versions of your product at your work site.

☐ Provide feedback on the product when needed by its DEC product manager.

☐ Act as a reference for the product at the request of Digital Sales or Marketing people.

Mail to: **Dena Shelton, L&T SIG Masters Coordinator, Cullinet Software, Inc., 2860 Zanker Road, Suite 206, San Jose, CA 95134.**

[1] Ada is a trademark of the DoD

Languages & Tools SIG

WISHLIST QUESTIONNAIRE

Name: _____ **Title** _____

Company: _____

Address: _____

_____

_____ **Phone:** ( ) _____

Network Address: _____ **Date:** _____

The Languages & Tools SIG is principally concerned with the DEC and public domain software products listed below. If your request directly involves one of these products, please check which one (if you have more than one request, please use a separate form for each):

☐ Debug	☐ Bliss	☐ CMS	☐ TPU	☐ C	☐ Test Manager
☐ Pascal	☐ Basic	☐ MMS	☐ EVE	☐ Ada[1]	☐ Runoff & DSR
☐ Fortran	☐ Cobol	☐ LSE	☐ EDT	☐ APL	☐ TEX & LATEX
☐ Document	☐ Dibol	☐ SCA	☐ TECO	☐ RPG	☐ Cobol Generator
☐ VAX Notes	☐ Emacs	☐ PCA	☐ PL/I	☐ Scan	☐ Software Project Mgr

If your request or suggestion doesn't relate to one of the products listed above, check which one of the following Language & Tools SIG topics it concerns:

☐ Newsletter	☐ Symposium Sessions	☐ Pre-Symposium Seminars
☐ Masters Program	☐ Working Group Activities	☐ Session Notes
☐ Information Folder	☐ SIG Tape	☐ DECUS Store Item
☐ Other L&T SIG topic: _____		

Wish List Request—brief description: _____

_____

_____

_____

Complete description—please explain your request thoroughly; don't assume we know details of other products or services; give examples. _____

_____

_____

_____

_____

_____

_____

_____

_____

_____

_____

_____

_____

_____

Mail to: **Shava Nerad, L&T Wishlist Coordinator, MIT, 77 Mass Ave. W91-219A, Cambridge, MA 02139**; (617)253-7438

[1] Ada is a trademark of the DoD

DATAGRAM

DATAGRAMs are short messages, comments, requests, or answers that are published in NETwords. Please fill in the sections below and send the DATAGRAM to:

Vickie Hess
NETWords Editor
2510 Limestone Ln.
Garland, Tx. 75040

Title: _____

Message: _____
_____
_____
_____
_____
_____
_____
_____
_____
_____
_____
_____
_____

Your Name: _____
Address: _____
_____
Telephone: _____

If this is a reply to a previous DATAGRAM, what #? __

Signature: _____ **Date:** _____

Place
Stamp
Here

Vickie Hess
NETWords Editor
2510 Limestone Ln.
Garland, Tx. 75040

Fold Here

OFFICE AUTOMATION SIG

SYSTEM IMPROVEMENT REQUEST SUBMISSION FORM

Name _____ **Address** _____

Firm _____ _____

Telephone _____ _____

INSTRUCTIONS: System Improvement Requests (SIR) can be either hardware of software; please check the category addressed by this SIR. Under ABSTRACT, give a brief definition of the capability you would like. In the DESCRIPTION section, give a detailed description and examples of what you want. Be specific; don't assume that we know how other products function. Justify the usefulness of the capability and give an example of its use.

HARDWARE IMPROVEMENT

DECmate _____

PRO-Series _____

Rainbow _____

Other _____

SOFTWARE IMPROVEMENT

ALL-IN-1 _____

CP/M (DECmate) _____

CP/M (Rainbow) _____

Other _____

WPS _____

P/OS _____

MS-DOS _____

ABSTRACT _____

_____

_____

DESCRIPTION _____

_____

_____

_____

_____

_____

_____

_____

_____

E. Catherine Ditamore
ARA Services
Corp MIS
The ARA Tower
1101 Market Street
Philadelphia, Pa. 19107

Professional Wish List Ballot

Use this ballot to show which items on the Professional Wish List are most important to you. Put the number of the most important item on the list in space 1, the next most in space 2, etc.

1._____	10._____	19._____	28._____	37._____
2._____	11._____	20._____	29._____	38._____
3._____	12._____	21._____	30._____	39._____
4._____	13._____	22._____	31._____	40._____
5._____	14._____	23._____	32._____	41._____
6._____	15._____	24._____	33._____	42._____
7._____	16._____	25._____	34._____	43._____
8._____	17._____	26._____	35._____	44._____
9._____	18._____	27._____	36._____	45._____

Please add the following to the wish list:

_____

_____

_____

_____

Comments:

Name:_____

Company:_____

Address:_____

_____

_____

Work Phone:_____

Home Phone:_____

Return Ballot to:

Thomas Hintz
DECUS Professional Working Group
University of Florida
IFAS Computer Network
1022 McCarty Hall
Gainesville, FL 32611

Rainbow Wish List Ballot

Use this ballot to show which items on the Rainbow Wish List are most important to you. Put the number of the most important item on the list in space 1, the next most in space 2, etc.

1._____	10._____	19._____	28._____	37._____
2._____	11._____	20._____	29._____	38._____
3._____	12._____	21._____	30._____	39._____
4._____	13._____	22._____	31._____	40._____
5._____	14._____	23._____	32._____	41._____
6._____	15._____	24._____	33._____	42._____
7._____	16._____	25._____	34._____	43._____
8._____	17._____	26._____	35._____	44._____
9._____	18._____	27._____	36._____	45._____

Please add the following to the wish list:

_____

_____

_____

_____

Comments:

Name:_____

Company:_____

Address:_____

_____

_____

Work Phone:_____

Home Phone:_____

Return Ballot to:

Lynn Jarrett
DECUS PC Sig Rainbow Working Group Chairman
Union Tribune Publishing
P.O. Box 191
San Diego, CA 92108

PC POSTSCRIPT

PC Postscripts are short requests, comments and responses to be published in the *Postscript Section* of the PC SIG Newsletter. Please respond to the following:

__ Y/N This is a reply to a previous Postscript. __ __ Issue Mo. _____ No.

Title:_____

Message:_____

_____

_____

_____

_____

_____

_____

_____

_____

_____

_____

_____

_____

Name: _____

Address: _____

_____

Phone: (_____) _____-_____

Signature: _____ **Date**_____

DECUS PERSONAL COMPUTER SIG QUESTIONNAIRE

General:
I would like information on _____
I would like to see an article
in the newsletter on _____
I would like to see a symposium
session on _____

I am willing to write an article(s) on: _____
I am willing to be contacted by PC SIG members by telephone to give
assistance/advice on: _____
Phone number to call: Area Code (___) # _____ Times _____

I attend DECUS Symposiums : ____always ____Sometimes ____never
I expect to attend these symposiums ____Fall 85 ____Spring 86 ____Fall 86

I use/own: ___Rainbow(s) ___PRO(s) ___DECmate(s) ___Robin ___Other_____
I use the machine(s) checked above: ___at work ___at home ___both
If a work, total number of DEC PC's at your site: _____
I also use: ___VAX ___IBM or other mainframe ___IBM/other PC
Type of use: ___business ___educational ___government ___other_____
Primary Operating System: ___MS-DOS ___CP/M ___both equally
___P/OS ___UNIX ___other_____

I belong to a local DEC PC Group: ___yes ___no
There is a user group in my geographic area: ___yes ___no
I would like information on starting a user group: ___yes
I use a modem: ___often ___reluctantly ___never
___for work ___for pleasure ___both

Here is information on he DEC PC User Group I belong to or know of:

Name of Group _____
Name of Contact Person _____
Address _____
_____
Telephone (___)_____
Supports ___Rainbow ___PRO ___DECmate ___Robin ___LUG ___Gold Key

Here is a DEC oriented bulletin board not on your list, or new information on a
listed board:

Name of Board _____
Full name of Sysop _____
Address if known _____
City and State _____
Telephone Number _____
Other Info: _____
Supports ___Rainbow ___PRO ___DECmate ___Robin

The subjects of greatest ineres to me are:

___word processing	___project management
___spreadsheets	___specialized vertical software
___database	___(type)_____
___graphics	___Other:_____
___communications	___Rainbow
___programming	___PRO
___software reviews	___DECmate
___technical articles	___Robin

___DEC Gossip and News ___Other:_____

If I had it to do over again, I:
___would buy another DEC Rainbow/PRO/DECmate (circle one)
___might buy another Rainbow/PRO/DECmate if it was a bargain (circle one_
___would not buy another Rainbow/PRO/DECmate (circle one)

Will you continue to subscribe at the new price of $35/year? ___yes ___no
Feel free to enclose another page(s) with comments!
Do you feel that leaving the prices out of the newsletter
___is appropriate
___is very annoying
___makes the articles less useful

Do you feel that Decus should revise its (anti)commercialism policy?
___yes
___no

Name_____
Company_____
Address_____
City/ST/ZIP_____
Work Phone (___)_____
Home Phone (___)_____

Return to:
Barbara Maaskant
Computing Resources
The University of Texas Health
Science Center at San Antonio
7703 Floyd Curl Drive
San Antonio, Texas 78284

fold here, flap under----------

stamp

Barbara Maaskant
Computing Resources
The University of Texas Health
Science Center at San Antonio
7703 Floyd Curl Drive
San Antonio, Texas 78284

Information Resource Sign Up Sheet
Personal Computing Special Interest Group - PC SIG

Are you willing to be an information resource for other PC SIG members? Placing your name on the *Contact List* means you are willing to answer questions within the span of a brief telephone conversation. A Contact is not expected to be a consultant. *Please Register below.* Your name and phone number (including restrictions) will be posted in the PC SIG Newsletter.

First Name:_____ **Last Name:**_____

Address:_____

_____

City:_____ **State:**_____ **ZIP:**_____

Phone:(_____) _____-_____

Areas of Expertise:_____

_____

_____

_____

_____

Suggestions for Additional Services the SIG can Provide:

_____

_____

_____

Barbara A. Maaskant
UTHSCSA Computing Resources
7703 Floyd Curl Drive
San Antonio, Texas 78216

PERSONAL COMPUTING SPECIAL INTEREST GROUP
VOLUNTEER FORM

Name_____
Company_____
Address_____
City_____ State_____ Zip Code_____
Telephone_____

What special talents do you have?_____

When do you attend symposia?

- ☐ Always
- ☐ East Coast Only
- ☐ West Coast Only
- ☐ Occasional Attendance
- ☐ Other (please specify)
  _____
  _____

Please check if you are interested in helping with any of the following activities:

Symposia Related Activities:

- ☐ Session Chairs _____
- ☐ Campground Volunteer_____
- ☐ Suite Volunteer_____
- ☐ DECUS Store _____
- ☐ Software Clinic_____
- ☐ Panels_____ (indicate topics)_____
- ☐ Technical Sessions_____ (indicate topics)_____
- ☐ Articles for Update.Daily
- ☐ Write letters of appreciation
- ☐ Equipment Setup

Ongoing SIG Activities:

- ☐ Working Groups_____ (indicate which groups)_____
- ☐ Newsletter_____
- ☐ Public Domain Software Project _____
- ☐ Write Software for Special SIG Needs_____

Other SIG Activities: (please specify) _____

Do you wish to see the PCSIG undertake any activities which it is not currently doing? Please specify.

_____

Would you be willing to coordinate the activity you have listed above? ☐ Yes ☐ No

Thank you

RT-11 WISH LIST SURVEY

Name (optional) _____

Address (optional) _____

_____

_____

DECUS Number (optional) ----------

1.1	___	3.1	___	3.7u	___	3.13a	___	5.1b	___
1.2	___	3.2a	___	3.7v	___	3.13b	___	5.2a	___
1.3	___	3.2b	___	3.7w	___	3.13c	___	5.2b	___
1.4	___	3.2c	___	3.7x	___	3.13d	___	6.1	___
1.5	___	3.2d	___	3.7y	___	3.14	___	6.2a	___
1.6	___	3.2e	___	3.7z	___	3.15	___	6.2b	___
1.7a	___	3.3a	___	3.7aa	___	3.16	___	6.2c	___
1.7b	___	3.3b	___	3.7bb	___	3.17a	___	6.2d	___
1.8	___	3.3c	___	3.7cc	___	3.17b	___	6.3	___
1.9a	___	3.3d	___	3.7dd	___	3.17c	___	6.4a	___
1.9b	___	3.4a	___	3.7ee	___	3.17d	___	6.4b	___
1.9c	___	3.4b	___	3.8a	___	3.17e	___	6.4c	___
1.9d	___	3.4c	___	3.8b	___	3.17f	___	6.4d	___
1.10	___	3.5a	___	3.8c	___	3.18	___	6.5	___
1.11	___	3.5b	___	3.9a	___	3.19a	___	6.6a	___
1.12	___	3.6a	___	3.9b	___	3.19b	___	6.6b	___
1.13	___	3.6b	___	3.9c	___	3.19c	___	6.6c	___
1.14	___	3.6c	___	3.9d	___	4.1	___	6.6d	___
2.1	___	3.6d	___	3.9e	___	4.2a	___	6.7	___
2.2	___	3.6e	___	3.9f	___	4.2b	___	6.8a	___
2.3	___	3.6f	___	3.9g	___	4.3	___	6.8b	___
2.4	___	3.6g	___	3.9h	___	4.4a	___	6.8c	___
2.5	___	3.7a	___	3.9i	___	4.4b	___	6.8d	___
2.6	___	3.7b	___	3.9j	___	4.5a	___	6.8e	___
2.7	___	3.7c	___	3.9k	___	4.5b	___	7.	___
2.8	___	3.7d	___	3.10a	___	4.6	___	8.	___
2.9	___	3.7e	___	3.10b	___	4.7a	___	9.1	___
2.10	___	3.7f	___	3.10c	___	4.7b	___	9.2a	___
2.11	___	3.7g	___	3.10d	___	4.7c	___	9.2b	___
2.12	___	3.7h	___	3.10e	___	4.7d	___	9.3a	___
2.13	___	3.7i	___	3.10f	___	4.7e	___	9.3b	___
2.14	___	3.7j	___	3.10g	___	4.7f	___	10.1	___
2.15	___	3.7k	___	3.10h	___	4.7g	___	10.2	___
2.16	___	3.7l	___	3.10i	___	4.7h	___	10.3	___
2.17	___	3.7m	___	3.10j	___	4.7i	___		
2.18	___	3.7n	___	3.10k	___	4.7j	___		
2.19	___	3.7o	___	3.10l	___	4.7k	___		
2.20	___	3.7p	___	3.10m	___	4.7l	___		
2.21	___	3.7q	___	3.10n	___	4.7m	___		
2.22	___	3.7r	___	3.11a	___	4.7n	___		
2.23	___	3.7s	___	3.11b	___	4.7o	___		
2.24	___	3.7t	___	3.12	___	5.1a	___		

Send Responses to: RT-11 Wish List Survey
Multiware, Inc.
2121-B Second St. Suite 107
Davis, CA 95616

INPUT/OUTPUT Submission Form

A SIG Information Interchange

Please reprint in the next issue of the Pageswapper

If this is a reply to a previous I/O, which number? _____

Caption: _____

Message: _____

_____

_____

_____

_____

_____

Contact:

Name _____

Address _____

_____

_____

_____

Telephone _____

Signature _____ Date _____

Mail this form to: Larry Kilgallen, PAGESWAPPER Editor
Box 81, MIT Station, Cambridge, MA 02139-0901, USA

To register for on-line submission, dial (in the United States):
(617) 262-6830 and log in with the username PAGESWAPPER.

PAGESWAPPER - July 1987 - Volume 8 Number 12
INPUT/OUTPUT Submission Form

Tear out or photocopy reverse to submit an I/O item

Larry Kilgallen, PAGESWAPPER Editor
Box 81, MIT Station
Cambridge, MA 02139-0901
USA

Internals and Data Structures Manual Petition

Dear Digital Press,

Thank you for bringing the portion of the VAX/VMS Internals and Data Structures manual that was done with you to the Nashville Symposium. As a VAX/VMS practitioner I would ask you to speed the production of the completed version so that we can buy it from you. The timely production of a detailed, up-to-date internals manual is of major importance to me in my VAX/VMS related work.

IN ADDITION, we would like to see a Version 5 VAX/VMS Internals and Data Structures manual published as close to the release date of the operating system as possible.

Thank you for your consideration in this matter.

---------- Tape shut, please don't staple ----------

Name	Organization	Signature

---------- Tape shut, please don't staple ----------

---------- Fold Here ----------

Please
Don't
Forget
Postage

From: Ray Kaplan
PIVOTAL, Inc.
6892 E. Dorado Ct.
Tucson, AZ 85715

To: Ray Kaplan
PIVOTAL, Inc.
6892 E. Dorado Ct.
Tucson, AZ 85715

---------- Fold Here ----------

System Improvement Request Submission Form

Page 1 of _____

Submittor: Firm:

Address: Phone:

How to write an SIR:
Describe the capability you would like to see available on VAX systems. Be as specific as possible. Please don't assume we know how it's done on the XYZ system. Justify why the capability would be useful and give an example of its use. If you wish, suggest a possible implementation of your request.

Abstract (Please limit to four lines):

Description and examples (use additional pages if required)

PAGESWAPPER - July 1987 - Volume 8 Number 12
System Improvement Request Submission Form

Tear out or photocopy reverse to submit an SIR

Mark D. Oakley
Battelle Columbus Division
Room 11-6-008
505 King Avenue
Columbus, Ohio 43201-2369
USA

VAX System SIG Fall 1987 SIR Ballot

DECUS membership number _____ (six digits)

Our site uses the following VAX cpus (check all that apply)

8700/8800 ____ 8600/8650 _____ 8500/8550 _____ 8300/8200 _____
11/780,11/782,11/785 ____ 11/750 ____ 11/730,11/725 ____
MicroVAX I,II _____ MicroVAX 2000, VAXstation 2000 ____

We use VAXes in the following applications(Check all that apply)

Business EDP _____ Software Development _____
Education _____ Computer Science Research _____
Data Acquisition/Control_____ CAD/CAM _____
Service Bureau _____ Hardware Development _____
Scientific/Engineering _____ Office Automation _____
Telecommunications _____ Other _____

I support the following as the most important System Improvement Requests. (List from zero to fifteen SIR's):

-------- -------- -------- -------- --------

-------- -------- -------- -------- --------

-------- -------- -------- -------- --------

I oppose the following SIR's as detrimental. (List from zero to five SIR's):

-------- -------- -------- -------- --------

Mail to:

Mark D. Oakley
Battelle Columbus Division
Room 11-6008
505 King Avenue
Columbus, OH 43201-2693
USA

To be counted, your ballot must be received by October 30.

Tear out or photocopy reverse to vote on SIRs

Mark D. Oakley
Battelle Columbus Division
Room 11-6008
505 King Avenue
Columbus, Ohio 43201-2693
USA

Printed in the U.S.A.

Production Staff:

Mark Grundler: Communications Committee Chair
Palmer College

Don Golden: SIG Publications Chair
Shell Oil

Judy Arsenault: Publications Manager
DECUS

Judy Tessier: Phototypesetter/Graphics Designer
DECUS

Circulation:

DECUS SUBSCRIPTION SERVICE
DIGITAL EQUIPMENT COMPUTER SOCIETY
219 BOSTON POST ROAD, (BP02)
MARLBORO, MA 01752-1850

label
is not
it old
include
llation,
versity,

STATUS CHANGE

Please notify us immediately to guarantee continuing receipt of DECUS literature. Allow up to six weeks for change to take effect.

( ) Change of Address
( ) Please Delete My Membership Recor
(I Do Not Wish To Remain A Membe

DECUS Membership No: _____

Name: _____

Company: _____

Address: _____

_____

State/Country: _____

Zip/Postal Code: _____

_____

Mail to: DECUS - Attn: Subscription Servic
219 Boston Post Road, BP02
Marlboro, Massachusetts 01752-1
U