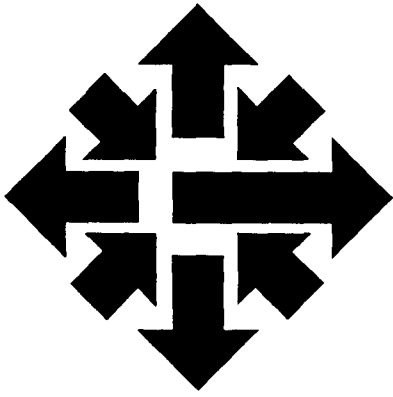


MW  
80729



The ACSS

# Newsletter

University of Minnesota  
Twin Cities  
March 1988

## Communications

### Electronic Messaging with ALL-IN-1

Jill McAllister and Marisa Riviere  
BITNET: JLM@UMNACVX, MARISA@UMNACVX

ALL-IN-1 is an application package that runs on DECNET networks as well as on single mainframes. ALL-IN-1 has its own Electronic Messaging subsystem that allows you to send mail to other ALL-IN-1 users. It is also possible to use Electronic Messaging to transfer VMS mail to ALL-IN-1 and to send ALL-IN-1 mail to VMS users. This article explains some of the basic uses of this subsystem.

ALL-IN-1 can be run on any terminal type by specifying that you do not want to use screen mode. For the purposes of this article, however, we assume that you are running on a terminal (or micro) that allows you to emulate a DEC VT-type terminal, and thus are in screen mode. When you log on, you will be told if you have new mail waiting. Notices of new ALL-IN-1 mail appear on your terminal screen, even if the mail arrives when you are not logged into ALL-IN-1.

#### Starting Electronic Messaging

To use Electronic Messaging from this workstation, first log into ALL-IN-1, and then type in **EM**. There is a specialized command for reading new mail, as you will see on the the package's menu of commands. Simply type **RN** and your first new message will appear. A carriage return will put you back at the EM menu. If you have additional new messages, you can type in additional **RN** commands.

*continued on page 55*

#### In This Issue . . .

Now you can listen to your All-in-1 electronic mail over the phone. See Alan Kaufman's article on DEC-Talk.

Thomas Kovarik describes the useful 'make' utility on the UX.

Every quarter ACSS offers free short courses on mainframe computing and inexpensive microcomputer courses. Our spring quarter short course appears in this issue.

We will phase out parts of our dial-in RJE services in 1988. Roger Gulbranson provides details in his article. Elsewhere in this issue, Dave Bianchi describes the NOS/VE file system, and Lawrence Liddiard reviews some of ACSS's accomplishments in 1987.

**Forums on  
Instructional  
Software**  
(See page 72)

# Help Page

## HELP-LINES

### Mainframe (CA,NV,UX,VX):

626-5592 8 am to 5 pm weekdays

### Artificial Intelligence:

625-8332 3 to 4 pm weekdays

### Data Base:

626-1887 10 to 11 am weekdays

### Graphics:

626-5592 8 am to 5 pm weekdays

### Microcomputer:

626-4276 9 am to 4 pm, weekdays  
(Closed 12:00 to 1:30 pm,  
Monday, Tuesday, and Friday.)

### Statistics:

626-1887 1 to 3 pm weekdays

### Text Analysis:

625-8332 3 to 4 pm weekdays

### Text Processing:

625-1391 10 am to noon, Tuesday,  
Wednesday, Thursday

## CONSULTING

### Walk-In

#### East Bank:

128C Lind Hall; 10 am to 4 pm, weekdays;  
7 to 9 pm Wednesday

#### West Bank:

140 Blegen Hall; 11:15 am to 1:45 pm, Tuesday;  
12:45 to 1:45 pm Wednesday; 11:00 am to 2:00  
pm Thursday

#### Microcomputer:

125 Shepherd Lab; 9 am to 4 pm, weekdays  
(Closed from 12:00 to 1:30, Monday, Tuesday,  
and Friday.)

---

### Electronic Mail Consulting

Consulting is now available via the mail facility on all ACSS systems (the CA, NV, VX, and UX). Send mail to user name CONSULT for questions after hours and for low-priority questions that are not critical to your immediate computing work. Replies will be sent to your account through the mail facility on your system.

---

### Instructional Computing Consultant

Department instructors may call 626-0200 for assistance in choosing ACSS systems (CYBER/NOS, CYBER/VE, ENCORE/UNIX, VAX/VMS), software, and for answers to any other inquiries on using computers for instructional computing.

---

## COMPUTING INFORMATION CENTER

128A Lind Hall, 625-7397, YZE6075@UMNACCA or MAD@UMNACVX

**Computing account and grant applications** available for CYBER, ENCORE, and VAX computers.

**Short course enrollment.** Short course schedules and class descriptions available.

**Assistance in ordering vendor documentation.** Vendor documentation is not always available in the University bookstores and may be ordered directly from the company.

**Complete documentation collection.** Reference copies of vendor and all other documentation for ACSS software.

**Free ACSS documentation.** General information, and mainframe and microcomputer information available.

**Computing Newsletters.** Subscribe to the *ACSS Newsletter* and the *ACSS Microcomputer Newsletter*. Newsletters from other computing centers are also available for reference.

## Sending a Message

Once you have typed in **EM** and are in the Electronic Messaging subsystem, you can create a mail message by entering the **C** option, as listed on the menu. You are then prompted for the receiver(s). There are several lines beginning

**TO:**

A carriage return moves your cursor from one **TO:** line to the next. A tab moves your cursor out of that section and down to the next section where the lines begin with

**CC:**

Again, a carriage return moves you from line to line, while a tab moves you out of that section.

The next line is labeled

**Subject :**

You are *required* to enter something on this line. After this, a carriage return will take you out of the header information, skipping over the last three fields, and into the buffer where you can start typing your message.

If you use a tab, instead of a carriage return, you will not leave the header information after entering the subject line. Using a tab, you can move your cursor through three more fields one at a time. The first is **PRIORITY**. You should leave that one at **FIRST CLASS**. The second field is **DELIVERY RECEIPT**. It is **NO** by default, but you can change this to **YES** if you wish to be notified upon delivery of your message. The last field is **READ RECEIPT**. This, too, is **NO** by default and may be changed. As above, a carriage return takes you out of the header and puts you into the buffer for typing in your message.

## Creating Your Message

There are two possible editors that you can use from within **ALL-IN-1**. When you are validated to use **ALL-IN-1**, **ACSS** gives you the editor **EDT** as your default editor. You may change this to the **WPS** editor, the other possible editor, from your User Profile screen. (There will be more information on the User Profile in a later article.)

Since **EDT** is the default, we will assume that you are entering your message with this editor. (If you need help learning **EDT**, you may use our on-line training software. Leave **ALL-IN-1** by typing the word **EXIT**, and, at the **\$** prompt, enter **RUN EDT2CAI**.)

Once you have finished your message, **CONTROL-Z** gives you an asterisk prompt (i.e., puts you into **EDT**'s line mode), and typing in **ex** at the asterisk prompt, followed by a carriage return, ends your mail message. You are now back at the **EM** menu. There will be a message at the bottom of your screen reminding you that you must explicitly mail this message. Typing in an **s** for send will send it.

## Using Folders

**ALL-IN-1** Electronic Messaging creates four folders for you. These are stored in your **ALL-IN-1** File Cabinet. Their names are **Inbox**, **Read**, **Created**, and **Outbox**.

continued on page 56

Mail that you have created but not sent is in your Created folder. Once you send it with the **s** command, a copy is moved to your Outbox folder, where it remains until you delete it.

All new mail goes to your Inbox folder, where it stays until you read it. Once read, it goes to your Read folder, although you may choose to place it in any other folder that you have created. (A later article will discuss folders and File Cabinet management.) If you delete a message, ALL-IN-1 places it in your Wastebasket, which is also a folder. You may explicitly empty your Wastebasket, or you may leave the messages there. Note that the Wastebasket is not automatically emptied when you finish your ALL-IN-1 session. You may end up with a very large Wastebasket of messages, and you will be charged for this disk space. To empty the Wastebasket, use the command **EW**. (This command does not appear on the Electronic Messaging menu. **EW** does appear on the menu for File Cabinet Management, but it can also be used from within Electronic Messaging.)

Currently, ACSS is not routinely emptying ALL-IN-1 Wastebaskets. This is an option we may choose to implement at some future date, so it is best not to leave messages in that folder if you think you may want them at some future time.

### Other Options

There are many other options which you can use from the Electronic Messaging menu. You can forward mail you have received to another user. You can create a mailing list, or distribution list. This allows you to send one mail message to the list, and then every person on the list gets a copy of your message. You can attach another document to your mail message. You can defer the sending of mail until a specific date and time. And, as mentioned above, you can import your VMS mail into the ALL-IN-1 workstation.

Finally, you can exit from ALL-IN-1 at any menu by typing the word **EXIT**.

### Training Software

There is very good on-line training available from within ALL-IN-1. This includes training from the Main Menu which provides an overview of the ALL-IN-1 subsystems, and there is also more specific training available from the subsystem menus themselves which include the Electronic Messaging subsystem. To use this training, you must have a terminal which can emulate a VT100 (or higher number model), and you must have entered **SET TERM /VT100** at some point prior to entering the ALL-IN-1 workstation software. Each menu that provides training has the option **TR**. Simply enter **TR**, and the instructional software will provide you with all directions necessary.

### Validations

If you would like to be validated to use ALL-IN-1, or would like to add the ability to use DECTalk (see the article in this issue) to your validations, please call Jill McAllister at 626-1878. Also call if you have any questions or are experiencing any problems with ALL-IN-1.

# The VAX Speaks!

## DECTalk Reads your Electronic Mail over the Phone

*Alan Kaufman and Marisa Riviere*

BITNET: ALAN@UMNACVX, MARISA@UMNACVX

We have installed an experimental feature on VX — DECTalk, from Digital Equipment Corporation — that allows you to “read” your All-in-1 mail (that is, you can hear it read) over any touch-tone telephone. To use DECTalk you have to be an All-in-1 user, but we may change this requirement, depending on the demand and interest, to also handle general VMS MAIL messages.

Please refer to the article on All-in-1 in the February issue of this *Newsletter* for information on requesting validation for All-in-1. Once we have validated you to use All-in-1, we will give you a six-digit identification number of your choice for DECTalk access. As we explain later in this article, you will use this identification number to access your mail file (and eventually other documents) by phone.

The number to call for DECTalk access is 626-0017. (At the moment, there is no rotary device for this number, so don't be surprised if the line is busy. Just try again later.) When DECTalk answers, it will ask you to key in your identification number on the touch-tone phone. DECTalk will repeat each one of the numbers that you press, then it will tell you to wait while it checks your identification number.

DECTalk can handle your All-in-1 mail folders, file messages, delete old messages, and forward prewritten answers. To take full advantage of DECTalk, you will need to acquire a certain degree of familiarity with All-in-1 mail. For further information, see the preceding article in this issue, which tells you how to use All-in-1 mail.

DECTalk can also transfer your regular VMS Mail to the All-in-1 mail folders so you can listen to your VMS MAIL messages over the phone. Although this option may seem attractive, there are several reasons why you may not want to select it. Your choice depends on the type of mail you receive and what you want to do with it. All this is explained in detail in the VX on-line document ACSS\$Writeup:DECTalk. You can display this document at your terminal with the command

```
$ type ACSS$Writeup:dectalk
```

or make a copy on the VX printer with

```
$ print/char=pstd ACSS$Writeup:dectalk
```

The rest of this article explains some fundamental DECTalk commands.

### DECTalk Basics

To read and handle your mail messages, DECTalk works by performing commands that you enter by pressing the keys on a touch-tone phone. In general, you will issue a command (press a key) after you hear DECTalk say the word **ready**, but for some commands you can press the required key at any time. The most commonly used commands are shown on the

*continued on page 58*

sample session later in this article. A more complete list of phone key commands is included on the DECTalk writeup.

Keep in mind that DECTalk reads your messages, but it does not understand them. DECTalk searches available dictionaries for punctuation and recognizable verbs in order to give some human expression into the spoken text. When DECTalk can not pronounce a word, it spells it. If you are sending messages to someone who uses DECTalk, you can help them understand your message over the phone by keeping the message short and by using short sentences.

### Sample Session

First dial 626-0017.

DECTalk answers your call with a message telling you that you have reached the ACSS VX system.

DECTalk then asks you to enter your identification code number. You have 10 seconds to key in your first number. DECTalk will repeat each number as you key it. DECTalk then checks your identification number for validation.

If the validation is correct, DECTalk tells you if you have any new (unread) messages. After that, it says *ready* and waits for your instructions.

Press the **read** key (Key 1 on your phone's key pad) to listen to your new messages. Press **next** (Key 2) to listen to the next message. (Messages are read in reverse order, with the last message received read first, etc.)

Press the **cancel** key (Key 9) to interrupt the current message and log out. DECTalk will ask you to confirm your request. Press the **confirm** key (Key #) and DECTalk will tell you that your session has ended. You can then hang up the phone.

### Other Useful Commands

To make DECTalk repeat a paragraph, press the repeat key (Key 7) while the paragraph is being read. DECTalk will continue reading to the end of the paragraph, repeat the entire paragraph, and then continue with the rest of the text.

When reading very long messages, DECTalk may want to check to see if you're still on the line; if you're not, DECTalk will hang up to free the phone line. To check, DECTalk will interrupt the text of a long message each three minutes to issue a time-out notice and request you to press any key to continue. Press any key within 20 seconds to continue listening to your messages.

If you need help, press the **help** key (the \* key).

### For More Information

Additional information on DECTalk Mail is available in the *DECTalk Mail Access User's Guide*, Digital Publication Number AA-EA42A-TE.

# TELNET Now Available on CA and NV

*Paul Tranby*

BITNET: PDT@UMNACCA, PDT@UMNACVX

Internet: PDT@ux.acss.umn.edu

The latest release of CDCNET from Control Data Corporation now provides support for some of the applications in the Internet (TCP/IP) protocol suite. Currently supported are TELNET connections on NOS and NOS/VE and FTP connections to NOS/VE. This software has been installed at ACSS and is available for experimentation.

Any host, workstation, terminal server, or personal computer that has access to the campus Internet and supports TELNET and FTP should be able to create interactive connections to CA and NV via TELNET and be able to transfer files to and from the NV machine via FTP. FTP for NOS (CA) will be available in the future.

The internet names and numbers for CA and NV are:

ca.acss.umn.edu 128.101.63.6  
nv.acss.umn.edu 128.101.63.7

This early release of TCP/IP for CDCNET does not yet support the ethernet address resolution protocol (ARP). This means that before any host can communicate with CDCNET via TCP/IP, CDCNET must have the remote hosts 48 bit hardware ethernet address statically configured. Many of the hosts around campus have already been configured. If you cannot connect to CA or NV, however, send your host's internet name and number and hardware ethernet address to Paul Tranby at one of the mail addresses above or call at 626-0815.

## Dial-in RJE Services to be Phased Out

*R. L. Gulbranson*

We have recently reevaluated our dial-in RJE (Remote Job Entry) services. We looked at both the connect-time revenue and the total income produced as a result of a dial-in RJE session. What we found is that costs to provide these services are much greater than the total revenue produced.

Our current RJE services are determined by two parameters: the connection protocol and the connection speed. We support two protocols — HASP and MODE4A (historically called UT200) — and two speeds — 2400 bps (bits per second) and 4800 bps. This results in four types of sessions.

Our major cost for RJE services is the maintenance of the CDC2551, which is the now-obsolete front-end that provides these services. If we can eliminate this cost, we believe that the service will again be cost-effective. We will move some of the RJE service to our new CDCNET front-ends. However, we do not see this as other than a prolonging of a service that must eventually be eliminated.

We believe that the number of users utilizing this service is quite small. We are prepared to provide individualized service to migrate these users to other forms of communication. If you

*continued on page 60*

*continued from page 59*

currently utilize these services, and have not been contacted, please call Paul Tranby (626-0815) to discuss how we can best accommodate your communication needs.

The timetable below indicates when we plan to eliminate or move these services.

### **Timetable**

- by May 1:** eliminate MODE4A service, eliminate 2400 bps service, and move present 4800 bps HASP service to CDCNET. (This may require a phone number change.)
- by July 1:** evaluate costs to continue to provide 4800 bps HASP service, and determine future timetable for continued service.

*Data Base Update*

## **Software to be Removed from CA**

*Peter Oberg*  
BITNET: PJO@UMNACVX

On July 1, 1988, ACSS will remove the following software packages from the CYBER CA. We will no longer offer these packages because the vendors no longer support them or because the packages have been used very little, if at all, in the last two years.

- FMS      File Management System for Quick Query package.
- KWIC      Keyword in Context by CDC.
- TRIAL      A card-oriented bibliographic retrieval system by Northwestern U.

If you are interested in continuing to use any of these packages, please contact me at the above electronic mail address or phone 626-0200.

## Make Progress with make

Thomas Kovarik

**Question:** I am a programmer on the UX, and I need to distribute a program, consisting of many routines, to other UNIX sites. What UNIX utility will help me?

**Answer:** You should learn **make**.

Wouldn't it be nice to simply type the word **make** and have your computer get all the files together needed to compile a main program and all the necessary subprograms? Suppose it then compiled these for you using exactly the correct parameters, then linked in all necessary modules and libraries to create the executable file? Well, that is what a frequently used utility on UNIX systems does.

The **make** utility takes all instructions it needs from a file called **Makefile**. **make** is so intelligent it even checks the date and time of the last modification of source or header files and the last compilation producing relocatables, so it does the minimum work needed to produce or reproduce your executable file.

For example, suppose Mr. Brown wants to give his new agricultural engineering program to Dr. Green. He places all source and header files for all his routines and subroutines, no matter what languages they were written in, in one subdirectory and permits that subdirectory (or sends it across BITNET or copies it to a floppy disk if Dr. Green is on another machine). One of those files is the **Makefile**, where easy to understand instructions to **make** are placed. Dr. Green types:

```
cd newdirectory
make
```

and **make** does all the compilation and linkage necessary (the minimum possible) to produce the executable file. Should Dr. Green then want to change a few lines of source or header files, he simply types:

```
vi sourcefile1 sourcefile2
make
```

and **make** realizes that some updates have occurred and does the minimum work necessary to reproduce the executable. What a marvelous way to give software to a colleague! In fact, this is how software is distributed worldwide for both large and small packages on UNIX systems. In fact, *UNIX itself* is distributed in this way.

But where do you get this magic **Makefile**? One way is to copy an existing **Makefile** from the HELP-Line or from a colleague and modify it slightly for your needs. To find out more about **make**, type:

```
man make
```

*continued on page 62*

or to send a hard copy to, say, bin 443 at room 128B Lind Hall, type:

```
man -t make | lpr -t -Plind128b -Cea.443
```

Or, as is typical of the convenience of UNIX, you can simply type:

```
mkmf
```

to make a **Makefile**. (The people who designed UNIX truly thought of everything). To find out more about **mkmf** and to learn about the syntax of the **Makefile**, type:

```
man mkmf
```

With **make** you can use any or all of the following types of source programs or subprograms (among others) to easily produce an executable file:

```
C
Efl
Fortran
Header Files (Include files)
Lex
Lisp
Object files
Pascal
Ratfor
Assembler
Yacc
```

If you use **mkmf** to create your makefile, it will offer the following options for more power and flexibility:

<b>all</b>	compiles and loads a program or library.
<b>clean</b>	removes all unnecessary files.
<b>depend</b>	edits the <b>Makefile</b> and regenerate the dependency information.
<b>extract</b>	extracts all the object files from the library and places them in the same directory as the source code files. The library is not altered.
<b>index</b>	prints an index of functions.
<b>install</b>	compiles and loads the program or library and moves it to its destination directory.
<b>library</b>	compiles and loads a library.
<b>print</b>	prints source code.
<b>tags</b>	creates a tag file for the editor, for C, Pascal, and Fortran source code files.
<b>program</b>	compiles and links a program.
<b>update</b>	recompiles only if there are source code files that are newer than the program or library.

In fact, **Makefile** is *the* UNIX standard for software distribution and exchange worldwide. When a UNIX site generates a new C compiler or a new version of the operating system or any software product, the installing person simply goes to the new subdirectory and types **make**.

For more information about **make**, see ACSS's *UNIX Programmer's Guide*, a compilation from the *Berkeley UNIX Reference Set*.

## Notes on Another Year

Lawrence Liddiard

BITNET: LIDDIARD@UMNACVX

UX: laI

At ACSS we do our yearly civil service performance reviews for our staff in mid-January and about the same time the University provides the financial reports for December. At that time, we also begin reviewing the past calendar year and planning for the next fiscal year. In this column I'd like to use the results of this work to review some of our achievements (and frustrations) in 1987 and summarize some of our plans for Fiscal 1989.

Over the past couple of years a simple quarter formula has proven to be accurate within 3 to 4 percent for the total fiscal year. This formula implies that in our quarter-based University schedule scholastic computing remains at a stable level in the fall, winter, and spring, and it has a highly variable summer usage. The actual budget for Fiscal 1988 was put together in April of 1987, so you can imagine our satisfaction this January when the formula predicted that we had fairly accurately estimated income eight months ago.

Just when you have a nice little estimator, other factors may threaten its simplicity. In the February 1988 "Academic Priorities" section of *Commitment to Focus*, the administration recommends adoption of a semester rather than a quarter system starting September, 1990.

### An Overview of Systems, 1988

Currently Academic Computing Services and Systems is divided into two sections. As the Deputy Director of Systems, I am responsible for central systems, networking, and operations groups. These groups are further divided into these specific areas: NOS and NOS/VE, UNIX, and VMS systems; general programming language and environment; measurement; data communications; electronic messaging; and computer operations and services.

**NOS:** Our old stable system required updates in March, June, and December to go from 2.5.1 to 2.5.3 CDC releases. Why change? The latest release had the initial TCP/IP communication and File Transfer (FTP) protocols software needed to connect the CA CYBER with ACSS's three other systems and prepare it for the standards proposed for the University's backbone network.

The major glitch in these changes to NOS was the disappearance of the SRU multiplier in the March change, leading us to believe users had forsaken the CA system. Extensive testing is done before a release update; but snafus still occur.

**NOS/VE:** We installed CDC's latest operating system on the IT Engineering Center Dual CYBER 830 on September 3, changing it from a NOS system named MD to a NOS/VE system named NV. Besides the latest ICEM software, we added CYBIL, Pascal, and C languages in a December update.

In computing circles, it's often said that after six months of preparation, a new system will be heavily used. I think that a more realistic estimate is a year, and this turned out to be the case

*continued on page 64*

for NV. For Fall Quarter the use of NV was only slightly above the previous year, but during several days in early February 1988 we saw its CPU and memory saturated.

**Figures 1 and 2** display the changing levels of CPU usage on our CYBER systems for the first six months of Fiscal 1987 and 1988, with averages for the periods represented by two horizontal bars.

**UNIX:** The greatest change to this modular system occurred in September when we installed new processor, memory, and I/O cards that made it twice as fast with four times the memory and I/O throughput. Its model number went from a Multimax 120 to 320. The changeover required several days to reformat and reload disks.

The biweekly system software changes required earlier in the year to cure instability problems have decreased to monthly updates as it now runs weeks between software crashes. Occasionally an infinite process initiator program can still cause accounting difficulties.

**VMS:** CPU and memory saturation during long stretches of November and December have required us to make software and hardware changes and upgrades. In its second year of stable operation, we added additional VMS services like conferencing (NOTES); consulted with departments that needed to tune working sets of large applications; and expanded account capabilities. ACSS also used its full VMS license to provide reduced cost Right-To-Copy licenses for University departmental VAXes.

**Figures 3 and 4** display the changing levels of CPU usage on our UNIX and VMS systems for the first six months of Fiscal 1987 and 1988, with averages for the periods represented by two horizontal bars.

**Programming Languages and Programming Environments:** We added more educational software (ToyOS, Scheme, Modula, Ada/Ed, C++, Smalltalk, TeX, Prolog, Parlog, APL, and an iPSC-hypercube simulator) to the UNIX system. In continuing our long standing admiration of the Pascal language, a person in this group is also a member of the ANS and ISO standard committees. This has resulted in a new Extended Pascal standard and improvements to the Pascal environments on all our systems. We worked with DEC to provide a reasonably priced software library for departmental MicroVaxes using our Computing Information Center facilities.

**Measurement:** 1987 was the last year of a successful effort to link all ACSS accounting records to the ACCSTAT data base to ensure accurate resource measurement and billing.

**Data Communications:** Although problems (particularly electrical power problems) continued in several data component parts of the telecommunications system, the LANmark Ethernet service introduced in March has been very satisfactory. ACSS has used LANmark and a Kinetic Fastpath Appletalk/Ethernet gateway to link up Lauderdale's Appletalk network, consisting of Apple and Phonenet transceivers, to a larger campus-wide Appletalk internet sponsored by our micro group.

By April we had given up trying to make an interim DG MV/4000 serve as a TCP/IP gateway to CA and waited for the CDCNET solution. In mid-year Postscript printing service was provided from our central systems to various campus laser printers.

**Networking:** ACSS's associate director for networking currently attends meetings of the All-University Committee for Network and Communications Planning, the MRnet Technical, CICnet operations, and the state-wide educational network, besides heading the Information Systems Technical Group. As a result of this activity ACSS plans to participate in the growth of networking within the University, state, and region. Planning has begun on replacing the ACSS Tellab equipment with packet-based TCP/IP terminal servers.

**Electronic Messaging:** To focus our efforts in electronic mail within ACSS, we selected a staff person to co-ordinate our services, ranging from BITNET to our version of the Electronic Postmaster. One major improvement we made in 1987 was the uniform mail syntax allowed by the new mail router for Internet and BITNET.

We soon realized that entering names in the original Electronic Postmaster was next to impossible, and changed it to make it easier to use.

**Computer Operations and Services:** After a retrenchment in staff, ACSS's operations group successfully kept four different computer systems running around the clock. They also provided consistent services on the Xerox 8700 laser printer, color and electrostatic plotters, and our magnetic tape library throughout the year. Operations staff worked with the State Joint Purchasing group to make economical volume purchases of good quality computer supplies such as magnetic tape, computer paper, and printer ribbons.

### **What Will 1988 Bring?**

In 1988 we plan to:

- provide quality magnetic tape service on the UX and further stabilize that system;
- expand networking and electronic mail services;
- provide more CPU power and memory on the VX;
- add new instructional languages and packages;
- work with departments to make our systems better meet their needs;
- continue to lower rates when possible; and to
- provide conversion guidelines and help for those moving from NOS to our other systems.

Finally, the University has indicated that ACSS will probably be moving from Lauderdale to a new location in 1988. If we move, there will be a short period when our systems will be not available. We'll provide our users with information on this as soon as we know the University's plans. We hope that such a move will allow the scattered staff of ACSS to recombine into more efficient and convenient service groups.

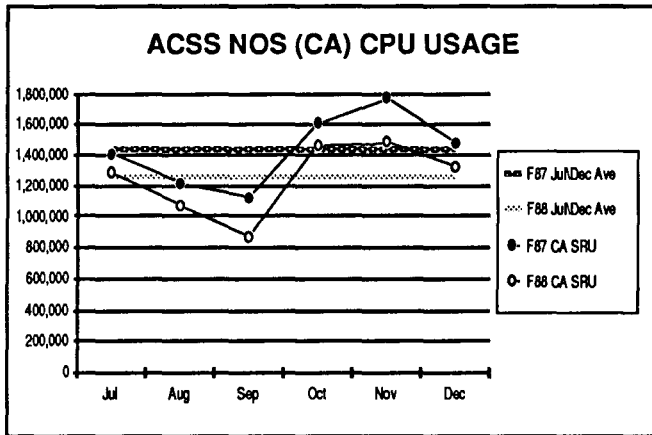


Figure 1: CPU use of the CYBER CA for the first half of fiscal years 1987 and 1988, with averages.

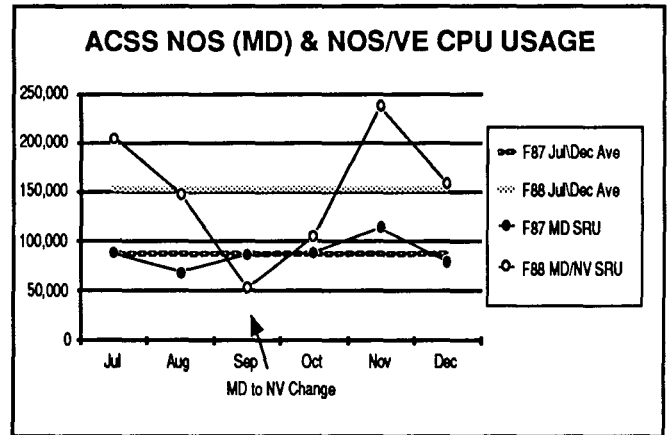


Figure 2: CPU use of the CYBER MD and NV for the first half of fiscal years 1987 and 1988, with averages.

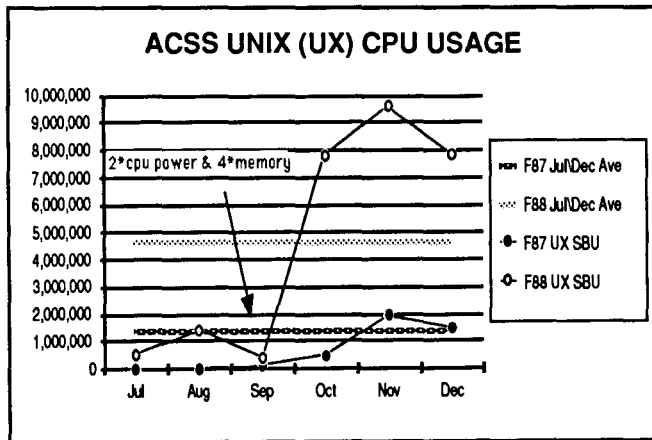


Figure 3: CPU use of the UNIX UX for the first half of fiscal years 1987 and 1988, with averages.

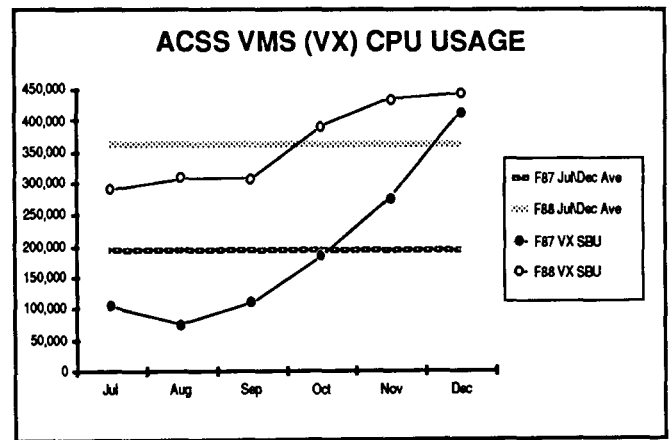


Figure 4: CPU use of the VAX VX for the first half of fiscal years 1987 and 1988, with averages.

# The NOS/VE File System

Dave Bianchi

BITNET: DJB@UMNACCA

*This article is the third in a series describing the NV NOS/VE system.*

In this article I discuss the NOS/VE file system. For more documentation on NOS/VE, please contact the ACSS Computing Information Center in 128A Lind Hall.

## NOS/VE Catalogs

NOS/VE uses a **hierarchical** file system grouping files into **catalogs**. You always have access to two catalogs: **\$LOCAL** and **\$USER**. (A NOS/VE catalog is similar in concept to a directory on VMS or UNIX.)

**\$LOCAL** Files in the \$LOCAL catalog are *temporary*. They can be accessed only by you and they are automatically deleted when you log out. \$LOCAL contains temporary files that you create and system-defined local files like \$OUTPUT, \$INPUT, OUTPUT, and INPUT. \$LOCAL cannot have subcatalogs.

**\$USER** Files in the \$USER catalog are *permanent*. They exist until you explicitly remove them. You can permit other users to access your permanent files.

## The Working Catalog

When you log in, NOS/VE assigns your default (or working) catalog to \$LOCAL. To make sure that any files that you create exist after you log out, you can set your working catalog to \$USER in your user prolog file with the command:

```
SET_WORKING_CATALOG $USER
```

## Permanent Files

Permanent files are stored in catalogs organized in a hierarchy. The top level of the hierarchy consists of the *family catalogs*. Under the family catalogs, the system creates a *master catalog* for each user. By convention, the name of the master catalog matches the user name. On the NV system, there are two families: \$SYSTEM and CADVE. The \$SYSTEM family contains all system files. All user master catalogs are created under the CADVE family.

You reference a file by its complete *path name*, including any catalogs that contain it. The general form of a complete path name is:

```
:family.username.catalog(s).filename.cycle.position
```

*continued on page 68*

You do not always have to specify the complete path name. Often you can just specify the file name, as in:

**PROLOG**

In this case, the login family name and the value of the working catalog are used to locate the file. For example, if your working catalog was \$USER and your family and user name were CADVE and ABC1234, then the complete path name for the above file would be:

**:CADVE.ABC1234.PROLOG**

You must specify a more complete path name when you want to refer to files in another master catalog or under another family (such as \$SYSTEM).

**File Cycles and File Position**

The last two fields in the full path name format are cycle and position.

A permanent file can have up to 999 different versions. Each version is called a *cycle*. A file reference can include a cycle number and file position. Use any of three system-defined keywords to access particular cycles of a file:

<b>\$HIGH</b>	The highest existing cycle of the file
<b>\$LOW</b>	The lowest existing cycle of the file
<b>\$NEXT</b>	The next new cycle of the file (equal to \$HIGH+1).

Most commands work with the highest cycle of a file if the cycle is not specified. One important exception: **DELETE FILE** deletes the \$LOW cycle of a file by default. For example, to create a new version of a file for editing, use:

**COPY\_FILE DATA.\$HIGH DATA.\$NEXT**

You can include a file position in a file reference to indicate how a file is to be positioned prior to use. Specify one of the following positions:

<b>\$ASIS</b>	Do not change the position of the file.
<b>\$BOI</b>	Position the file to the beginning-of-information.
<b>\$EOI</b>	Position the file to the end-of-information.

A file position can be used, for example, to append one file to the end of another:

**COPY\_FILE DATA1 DATA2.\$EOI**

**For More Information**

For more information on files and catalogs, see these CDC manuals: the *NOS/VE Tutorial* and *SCL for NOS/VE System Interface*.

# ACSS

## Spring Quarter Short Courses

### 1988

#### SHORT COURSE POLICY

Each quarter ACSS offers a large number of free computer short courses for the University community. Although these are usually introductory topics, the instructors spend a lot of time preparing for them. We ask people to register for these classes by phone, mail, or in person, so instructors know how many people to expect if materials are to be distributed. When registration for each class reaches classroom capacity we keep a waiting list of those wanting to attend. We always have waiting lists. If you do not show up for class, or call to cancel, you are depriving those who wish to attend.

ACSS avoids overbooking classes so that instructors and class participants are not inconvenienced. However, if the large number of "no-shows" continues, we will be forced to begin charging for these free classes and we will terminate telephone registrations.

Please be considerate. Think of those staff who organize and teach ACSS short courses. And, in particular, think of those users who are on the waiting list. If you aren't able to attend a class, a simple phone call (625-7397) will enable us to contact someone who can.

Registration begins Monday, March 14.

#### MAINFRAME COURSES

Mainframe courses are free. To register call 625-7397. Some classes have size limitations: Register early.

#### Introductory

Introduction to Computers	SKB	April 4-15	MWF	2:15-4 pm
Introduction to UNIX Operating System	JLM/TEK	April 5-21	TTh	2:15-4 pm
Introduction to VMS Operating System	JHS	April 18-27	MWF	2:15-4 pm
Introduction to NOS/VE Operating System	TEK	April 26-May 3	TTh	2:15-4 pm
Introduction to NOS 2 Operating System	CPL	May 2-13	MWF	2:15-4 pm

#### Advanced

Using SAS (Statistics)	DLF	May 2-6	MWF	3:30-5 pm
Beginning C Programming	JLM/TEK	May 10-26	TTh	2:15-4 pm
Using SPSS* (Statistics)	DLF	May 13-20	MWF	3:30-5 pm
Electronic Mail (VAX, BITNET)	JHS	May 16-18	MW	2:15-4 pm
INGRES (Data Base Management)	PJO	May 23-27	MWF	2:15-4 pm

**MICROCOMPUTER COURSES**

FEES: 1) U of MN Students, 2) U of MN Staff/Faculty, 3) Non-University persons.

General

Strategies for Networking Microcomputers & Workstations	May 3	T	1:30-4 pm	FREE
Micro/Mainframe Communications	May 9	M	2:30-4:00	FREE

IBM and IBM-compatible Courses

Limited to 10 people.

				(1) (2) (3)
Introduction to Micros: MS-DOS				
Section 1	March 29-31	TWTh	1:30-3:30 pm	\$40,\$50,\$80
Section 2	April 11-13	MTW	1:30-3:30 pm	\$40,\$50,\$80
Section 3	May 3-5	TWTh	9:30-11:30 am	\$40,\$50,\$80
Introduction to WordPerfect*				
Section 1	April 14-15	ThF	9:30-noon	\$25,\$35,\$60
Section 2	May 17-18	TW	9:30-noon	\$25,\$35,\$60
*A WP Workbook is required: \$12 at Williamson Hall Bookstore Electronics Desk.				
Introduction to Microsoft Word for IBM	May 25-26	WTh	1:30-4 pm	\$35,\$45,\$85
Managing Your Hard Disk				
Section 1	April 26-27	TW	1:30-4 pm	\$40,\$50,\$80
Section 2	May 10-11	TW	1:30-4 pm	\$40,\$50,\$80
Beginning Lotus 1-2-3				
Section 1	April 21	Th	9:30-noon	\$25,\$35,\$60
Section 2	June 2	Th	9:30-noon	\$25,\$35,\$60
Intermediate Lotus 1-2-3	May 26	Th	10:00-noon	\$25,\$35,\$60
Introduction to dBase III				
Section 1	April 4-8	MWF	1:30-3:30 pm	\$40,\$50,\$80
Section 2	May 10-12	TWTh	10am-noon	\$40,\$50,\$80
Programming in dBase III	May 31-June 1	TW	9:30-noon	\$40,\$50,\$80

### Macintosh Courses

Hands-on. Limited to 10.

#### Macintosh Fundamentals

Section 1	March 30	W	12:15-1:00 pm	FREE
Section 2	April 12	T	12:15-1:00 pm	FREE
Section 3	April 26	T	12:15-1:00 pm	FREE
Section 4	May 6	W	12:15-1:00 pm	FREE
Section 5	May 19	Th	12:15-1:00 pm	FREE

(1) (2) (3)

#### Introduction to Microsoft Word for Macintosh

Section 1	April 5-7	TTh	1:30-4 pm	\$35,\$45,\$85
Section 2	April 27-28	WTh	9:30-noon	\$35,\$45,\$85
Section 3	May 12-13	ThF	1:30-4 pm	\$35,\$45,\$85
Section 4	May 24-25	TW	9:30-noon	\$35,\$45,\$85

#### Intermediate Microsoft Word for Macintosh

section 1	April 14-15	ThF	1:30-4 pm	\$35,\$45,\$85
section 2	May 19-20	ThF	1:30-4 pm	\$35,\$45,\$85

#### Beginning Excel

Section 1	April 19-21	TTh	1:30-4 pm	\$35,\$45,\$85
Section 2	June 2-3	ThF	1:30-4 pm	\$35,\$45,\$85

#### Intermediate Excel

May 4-5	WTh	1:30-4 pm	\$25,\$35,\$85
---------	-----	-----------	----------------

#### Introduction to Desktop Publishing with PageMaker

May 27	F	1:30-4 pm	\$35,\$45,\$85
--------	---	-----------	----------------

#### Developing Applications for the Macintosh Using Pascal

April 18-22	MWF	1:30-3:30 pm	\$25,\$35,\$60
-------------	-----	--------------	----------------

#### Using Hyper Card

Section 1	April 28-29	ThF	1:30-3:30 pm	\$25,\$35,\$60
Section 2	May 16-18	MW	1:30-3:30 pm	\$25,\$35,\$60
Section 3	May 31-June 1	TW	10:00-noon	\$25,\$35,\$60

#### Using HyperTalk (HyperCard Programming Environment)

May 23-24	MT	1:30-3:30 pm	\$25,\$35,\$60
-----------	----	--------------	----------------

#### MacApp Object Pascal

May 9	M	1:30-4 pm	\$25,\$35,\$60
-------	---	-----------	----------------

The following Macintosh class is demonstration only. No hands-on.

#### Preparing A Dissertation on a Macintosh

May 13	F	9:30-noon	\$15,\$25,na
--------	---	-----------	--------------

## ACSS SHORT COURSE INFORMATION

**Prerequisites:** Please note any prerequisites for the class you are interested in. Instructors will not be able to review any prerequisite requirements. If you need more information on short courses, call the Computing Information Center at 626-7397.

**Limits:** All microcomputer classes and many mainframe classes have limits to class size. Please try to register early to be sure of getting a place. If you decide to withdraw from a class, please do so as soon as possible, so that we may make the space available to others.

**Registration:** Registration is located at the ACSS Computing Information Center, 128A Lind Hall. (Hours: 8:00 am to 4:30 p.m., Monday through Friday.) Fees must accompany registration. Mail registrations will be accepted. Deadline for registering is 4:00 p.m. on the last working day before the class begins. Please call and give us your name if you plan to attend a free class; also call to cancel if you decide later not to attend, so we know how many to expect. We reserve the right to cancel a course if registration is insufficient. For registration information call 625-7397.

**Fees:** Fees following microcomputer courses above are for 1) University students, 2) University staff and faculty, and 3) non-University persons. Course fees may be paid by cash or check, or *include* a signed University journal voucher with registration information. Fees must be paid at the time of registration.

*No refunds* will be made after the class has begun. A penalty—reduction of the refund—will be assessed if you withdraw from a class within three days of the beginning of the class.

**Note:** Monday, May 30 is a University holiday. No classes will be held.

**Registration begins Monday, March 14.**

*Instructional Computing*

## Forums on Instructional Software

*Steven Brehe*

Information Systems, in cooperation with Projects Assist, Minnemac, and Woksape, and other departments, is sponsoring a series of weekly presentations, *Forums on Instructional Software*, on instructional computing. All lectures are held on Thursday afternoons, from 1:30 to 3 pm.

The spring schedule will be announced soon. Call 626-1090 for further information and to add your name to the Forum's mailing list.

## **SPSS<sup>x</sup> Trends on the VX**

*Bruce Center*

BITNET: BAC@UMNACVX

We have just received a copy of **SPSS<sup>x</sup> Trends** for a three-month trial on VX. **SPSS<sup>x</sup> Trends** contains:

- A state of the art ARIMA modeling procedure (ARIMA).
- Seasonal adjustments using the Census Bureau X11 procedures (X11ARIMA).
- Exponential smoothing (EXSMOOTH).
- Enhanced regression facilities including weighted least squares (WLS), two-stage least squares (2SLS) and regression with first order auto-correlated errors (AREG).
- Spectral analysis (SPECTRA).
- Fit a selected model to the time series and produce forecasts (CURVEFIT).
- A host of utilities to simplify the task of analyzing and forecasting time series data. These include the plotting of autocorrelations, partial autocorrelations, cross-correlations, goodness of fit to specified models, etc.

SPSS<sup>x</sup> Trends is implemented as an integral part of SPSS<sup>x</sup> 3.0 rather than as a separate package, so that it can be used along with any other commands or procedures in SPSS<sup>x</sup>. Thus, to invoke, SPSS<sup>x</sup> Trends, just type:

```
$ SPSSX3/OUTPUT=outfile.ext inputfile.ext
```

or

```
$ SPSSX3
```

for interactive use as before.

There is a surprisingly comprehensible manual (time series and English are usually incompatible tongues) available at the Computing Information Center in 128A Lind Hall: *SPSS<sup>x</sup> Trends*, SPSS Inc., Chicago Ill, 1988.

Since this is only a three-month trial, it's essential that users show their interest by using the package, or it will be removed from the VX after the trial period.

For information or to express your opinions about this package, please contact me at 625-2538 or on BITNET at BAC@UMNACVX.BITNET.

# Symbolic Algebra System: REDUCE on VX

Michael J. Frisch  
BITNET: MJFRISCH@UMNACVX  
VX, UX, CA: MJFRISCH

We have installed the REDUCE Version 3.2 system for symbolic algebraic manipulation on our VAX VMS system. It is a complete language for symbolic calculations and the program itself is written in that language, one based on LISP.

REDUCE is a system for carrying out algebraic operations accurately, no matter how complicated the expressions become. REDUCE can:

- expand and order polynomials and rational functions,
- differentiate and integrate symbolically,
- perform substitutions and pattern matches,
- simplify and factor expressions automatically and under user control,
- calculate with symbolic matrices,
- calculate using arbitrary precision integer and real arithmetic,
- do Dirac matrix calculations of interest to high energy physicists, and
- write long symbolic expressions into a file directly in Fortran format for solving problems both symbolically and numerically.

REDUCE is run interactively with the VMS command:

```
$ reduce
```

and it can also be run in batch using the **SUBMIT** command.

## Documentation and HELP File

A 12-page user's guide is available when you enter:

```
$ TYPE ACSS$WRITEUP:REDUCE
```

or

```
$ EDIT /READ_ONLY ACSS$WRITEUP:REDUCE.LIS
```

To obtain a printed copy on the VAX, enter:

```
$ PRINT /CHAR=PSTD /NAME=xx.nnn ACSS$WRITEUP:REDUCE
```

where *xx* is the site code and *nnn* is the bin number where the output should be delivered. For example, EA.425 is a bin number in Lind Hall. (A laser-printer copy of the user's guide is available for the price of photocopying in the Computing Information Center, 128A Lind Hall.) The user's guide explains how to get the 135-page *REDUCE User's Manual*. It also tells how to access seven self-paced instructional "lessons" that you can run in interactive mode.

The information in this article is reproduced as a HELP file in:

**\$ MOREHELP APPLICATIONS REDUCE**

### An Example Using REDUCE

The following simple example computes the derivative of the function

$$x^{x^x}$$

The mathematical definition of this function requires that we express it in REDUCE as  $x^{(x^x)}$  where ^ means exponentiation. For this example, the user inputs are in lower case and the REDUCE outputs are in upper case. To check the results, the derivative is integrated to show that the original function is computed.

```
$ reduce
REDUCE 3.2, 15-Apr-85 ...

1: df (x^(x^x), x);

      X
      (X + X)      2
      (X          * (LOG (X) *X + LOG (X) *X + 1)) /X

2: int ((x^(x^x+x) * (log (x) ^2*x+log (x) *x+1)) /x, x);

      X
      (X )
      X

3: bye;
```

### Comparison With the CYBER Version

Note that version 2 of REDUCE is available on the CYBER CA system under the name REDUCE2 with documentation in WRITEUP,REDUCE2. Version 2 doesn't have all the capabilities of version 3.2 that runs on the VAX (for example, symbolic integration isn't available), only batch use is allowed on the CYBER, and the memory area for solving problems is smaller on the CYBER. However, because the CYBER is about twice as fast as the VAX, less computer time is needed to solve equivalent problems. Because of our prices for computer time, the costs are likely to be about the same.

# Unused Packages to be Removed from CA

Michael J. Frisch  
BITNET: MJFRISCH@UMNACVX  
VX, UX, CA: MJFRISCH

The following packages will be removed from the CYBER CA on June 12. They are not being used as far as we can tell. If you are thinking of using these packages on CA after June 12, please let us know so we can revise our plans as necessary.

- |         |                                                                                                                                                                                                   |
|---------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| ALMAP   | An algebraic manipulator. To our knowledge, the REDUCE2 package can do all that ALMAP can.                                                                                                        |
| AUGMP   | AUGMENT Multiple Precision package.                                                                                                                                                               |
| FEMPAC  | Instructional finite element package. The ANSYS package on the VAX VX can do all that FEMPAC can.                                                                                                 |
| FORPREP | Fortran preprocessor used with ALMAP.                                                                                                                                                             |
| FUNPACK | Special functions library. The IMSL library contains these functions.                                                                                                                             |
| GPM     | Nonlinear optimization library. The GRG2 library can do all that GPM can and more.                                                                                                                |
| MP      | Multiple Precision library used with AUGMP.                                                                                                                                                       |
| NONSAP  | Nonlinear structural analysis package. The ANSYS package on the VAX VX can do all that NONSAP can.                                                                                                |
| PERTCST | Program Evaluation and Review Technique, cost analysis package (PERT/COST).                                                                                                                       |
| PERT78  | Program Evaluation and Review Technique package (PERT/TIME). Micro-computer packages (such as MacProject on the Macintosh) can do much more than PERT78 can do for small to medium size problems. |
| SAP4    | Linear structural analysis package. The ANSYS package on the VAX VX can do all that SAP4 can.                                                                                                     |
| SIMPLX  | Linear programming library. The IMSL library can solve these problems, too.                                                                                                                       |

## **Future M77 Becomes Current**

*Janet Eberhart*  
BITNET: JME@UMNACCA

The future version of M77 will become current on Sunday, March 20th. The new version corrects two problems: incorrect results for double-precision exponentiation (using \*\*) of very small negative numbers, and a blank page that occurs before the source listing when routed to the printer.

## **ENCORE GNU Emacs Editor Upgraded**

*Janet Eberhart*

We upgraded the display editor GNU Emacs to 18.49 on February 12. GNU Emacs has an extensive help facility within the editor. The Help Tutorial (accessed by typing `CTRL-h t`) is an interactive tutorial that can teach beginners the fundamentals of Emacs in a few minutes. Other Help facilities include Help Apropos (`CTRL-h a`) which helps you find a command given its functionality. Type `man gnuemacs` for more information.

## **Proposed Fortran 8x Standard Available**

*Janet Eberhart*

A copy of the draft proposed revised American National Standard for the Programming Language Fortran is available at the Computing Information Center, 128A Lind Hall. A summary of the standard is also available.

## **Dartmouth Basic on UX**

*Sue Arneson*  
BITNET: SLA@UMNACVX  
UX: sla

Standard Dartmouth Basic Verion 1.8 is now available on the UX machine. Refer to the on-line documents `help languages` and `man basic` for more information.

## Conventions

Throughout this and other ACSS publications, we have adopted these conventions:

- Messages and prompts from the ACSS computers appear in *plain type*, like *this*.
- Words that the computer systems replace with a specific name, value, or other information appear in *italic type*, like *this*.
- Commands you type at your terminal keyboard appear in **bold face type**, like **this**.
- Words that must be replaced by a specific name, value, or command that you type in appear in **bold italic type**, like ***this***.
- Comments to interactive sessions and program files are enclosed in { curly braces, like this }.

Here's an example:

***save, filename***

is a command you type in. You type ***save*** and replace ***filename*** with the name of your file. The system may respond with the message

*filename* ALREADY PERMANENT { An example of a system message. }

where *filename* will be replaced by the name of the file you attempted to save.

- The symbol <CR> refers to the carriage return (or RETURN) key on the terminal. The <CR> serves as a terminator for commands you type at your terminal. In most cases we do not show <CR>; we assume you know to type it after every command.

# Phones/Hours/Labs

## ACSS PHONE NUMBERS

Administrative Office: 626-1600

HELP-Line 626-5592

Access:	
CYBER (CA)	626-1620
CYBER (NV)	626-1622
ENCORE (UX)	626-1681
VAX (VX)	626-1641
RJE (2400 baud)	626-1656
RJE (4800 baud)	626-1663
Accounts:	
CYBERs, ENCORE, VAX	625-1511
Computer Hours (recorded message)	626-1819
Computing Information Center	625-7397
Contract Services	625-2303
East Bank I/O, 128B Lind Hall	625-5082
Engineering Services	627-4357
Equipment Maintenance/Repair	627-4357
Graphics Software	626-5592
Information, Lauderdale	626-1600
Lauderdale Computer Room	626-0550
Lauderdale Services	626-1838
Magnetic Tape Librarian	626-1838
Math and Statistics Packages	625-5830
Microcomputer HelpLine	626-4276
Newsletter Subscription	625-7397
Permanent File Restoration	626-0595
Project Assist	626-1090
Public Labs (Managed by ACSS)	
170 Anderson Hall	624-6526
140 Blegen Hall	624-5278
14 Folwell Hall	625-4896
306B Lind Hall	625-9032
32 OMWL	624-6526
130 Physics	625-6820
9 Walter Library	626-1899
Lab Manager (14 Folwell Hall)	625-7850
Publications Information	626-1093
Short Course Registration	625-7397
Shuttle Bus Service	625-9525
System Status (recorded message)	626-1819

For the phone numbers of consulting services see the Help Page.

## PUBLIC LABS TWIN CITIES CAMPUS

Location	Batch	Interactive	Micro
<i>East Bank</i>			
ApH 204			X
Arch 148			X
GenH		X	
ComH		X	
DieH 207, 270		X	X
EddyH Annex 54			X
ElkH 121, 124	P	X	
FolH 14, 14a	P	X	X
FronH		X	
LindH 26		X	
LindH 128B	P	X	
LindH 306B			X
MasCanCtr M39		X	
MechE 308		X	
MoosT 8-425			X
Nich 109			X
Phys 130		X	X
PioH		X	
SanH		X	
TerrH		X	
VinH 4		X	
VinH 203			X
WaLib 9		X	X
<i>West Bank</i>			
AndH 170	P		X
BlegH 140	P	X	
MdbH		X	
OMWL B2	P	X	X
<i>St. Paul</i>			
BaH		X	
CentLib B50			X
ClaOff 125E	P	X	
ClaOff B22			X
McNH 69			X
Vet 43B			X

P means Printer only. For more information see WRITEUP,LABS.

## SYSTEM OPERATING HOURS

	CYBER (CA), ENCORE (UX), VAX (VX)	CYBER (NV)	Low Rate
M-F	7 am - 4 am	7 am - 1 am	8 pm - 4 am, 7 am - 8 am
Sat	4 am - 9 pm	7 am - 10 pm	all operating hours
Sun	6 pm - 4 am	6 pm - 1 am	all operating hours

# Contents

Help Page .....	54	ACSS Spring Quarter Short Courses .....	69
<b>Communications</b>		<b>Instructional Computing</b>	
Electronic Messaging with ALL-IN-1 .....	53	Forums on Instructional Software .....	72
The VAX Speaks! .....	57	<b>Statistics Packages</b>	
TELNET Now Available on CA and NV .....	59	SPSS: Trends on the VX .....	73
Dial-in RJE Services to be Phased Out .....	59	<b>Math and Engineering Packages</b>	
<b>Data Base Update</b>		Symbolic Algebra System: REDUCE on VX .....	74
Software to be Removed from CA .....	60	Unused Packages to be Removed from CA .....	76
<b>Consulting Corner</b>		<b>Programming Languages</b>	
Make Progress with make .....	61	Future M77 Becomes Current .....	77
<b>Computing Reflections</b>		ENCORE GNU Emacs Editor Upgraded .....	77
Notes on Another Year .....	63	Proposed Fortran 8x Standard Available .....	77
<b>NOS/VE News</b>		Dartmouth Basic on UX .....	77
The NOS/VE File System .....	67	<b>Conventions</b> .....	78
<b>Short Courses</b>		<b>Phones/Hours/Labs</b> .....	79
Short Course Policy .....	69		

---

**The ACSS Newsletter**  
March 1988  
Volume 22, Number 3

**Acting Director of ACSS:** Michael M. Skow  
**Editors:** Steven Brehe, Paula Goblirsch

The *ACSS Newsletter* is published monthly by Academic Computing Services and Systems of the University of Minnesota, Twin Cities. Deadline for articles is the 10th of the month preceding publication; deadline for short announcements is the 15th. The *Newsletter* is produced with an Apple Macintosh Plus running Microsoft Word, FullPaint, MacDraw, Adobe Illustrator, and Aldus Pagemaker software, with camera-ready copy produced on the Apple LaserWriter Plus.

Direct comments, suggestions, articles, and announcements to the editors at the address below, or call (612) 626-1828 or 626-1093. For a free subscription call (612) 625-7397, or send your name and address to the Computing Information Center, 128A Lind Hall. Electronic Mail: YZE6075@UMNACCA or MAD@UMNACVX. On-campus address changes *must* include your department's name and your *departmental* address.

The University of Minnesota is committed to the policy that all persons shall have equal access to its programs, facilities, and employment without regard to race, religion, color, sex, national origin, handicap, age, veteran status, or sexual orientation.

Copyright 1988 University of Minnesota. Permission to copy is hereby granted, provided that proper acknowledgement is given.

---

**The ACSS  
Newsletter**

**Academic  
Computing  
Services and  
Systems**

Technical Publications  
100 Lauderdale CF  
University of Minnesota  
2520 Broadway Drive  
Lauderdale, Minnesota 55113

Nonprofit Org.  
U.S. Postage  
PAID  
Minneapolis, Mn.  
Permit No. 155

UNIVERSITY ARCHIVES  
10 WaLib

Deliver to current occupant.

March 1988