

UNIVERSITY COLLEGE CORK

Computer Bureau Newsletter

October 1986

Editorial

Welcome to the beginning of the new academic year! A lot has been going on while you've been away sunning yourselves, and the main details are inside this issue.

Contacting the Bureau

One of the most important changes concerns the way in which you contact the Bureau. Up to now, this has been mainly by phone, and because we are short of staff, this has led to a lot of frustration with phone calls going unanswered or staff not being available. From October 1986, the normal method for users to contact Bureau staff will be by electronic mail through the VAX or IBMs. There two main advantages to this for the user: no more unanswered phone calls; and a better quality of response. because staff are able to give a considered reply without continually interrupting work (which might be your work) to take a call (from somebody else).

If you have not used the mail system before, see the article on electronic mail in this issue, which explains what it does. There is also a leaflet available from the Advisory Service explaining it in more detail. Naturally this method does not apply to people who do not have access to a terminal or who have no UserID. We will still take phone calls, of course, but calls from people who do have UserIDs will take a lower priority to those from people without.

Academic Advisory Service

During the coming year we hope to expand our services to academics. A number of new facilities are mentioned in this issue, but it is on the quality and timeliness of the advice we give that we shall also be judged. The Bureau currently has only a limited Advisory Service because of staffing problems, but it is hoped to resolve these during the course of 1987.

The present Advisory Office will continue to handle matters relating to existing and future research projects, and to provide advice on hardware and software selection. However, because of the dispersed nature of the information and skills at our disposal, *initial enquiries* should normally be directed to the new Help Desk for onward routing to the relevant member of staff.

The Help Desk is on the 4th floor of the Science Building, and will be open from 10.00am to 12.30pm and 2.30pm to 4.30pm each working day. Queries and requests accompanied by written details which specify clearly the particular problem will be dealt with more quickly than those given only verbally. Email can be sent to UserID HELPDESK (on either IBM or VAX), and the phone number for enquirers without UserIDs or terminal access is 2609. All articles in this issue carry the UserID of the author, so you can contact them direct if you wish. There is a complete list of Bureau staff on the last page.

New Installations

A certain amount of new equipment is being installed this year. In addition, this issue carries revised prices for the most common PC configurations and software. The major new acquisitions are:

- The VAX is to get a long-overdue memory upgrade to 8Mb in the very near future. This should reduce the amount of paging, and thus improve performance.
- The 16 VAX terminals in the basement of the Dairy Science building will be supported from a MICROVAX. This room is used mainly for undergraduate teaching, and the dedication of a machine means freeing the 11/780 from the problem experienced last year of a noticeable deterioration in response speed when a class logged in.
- A MICROVAX is also being installed in the Department of Electrical Engineering, to complement the one installed last year in Civil Engineering. This now means the engineers are independent of the 11/780 for their on-line work.
- A new Graphics Laboratory is currently being fitted out in the basement of the Science building. While this is primarily for Computer Science teaching purposes, it will be available by arrangement to other departments. Each workstation comprises AT-compatible pc, 640k, 20Mb hard disk, 5.25in floppy, EGA colour graphics, mouse and printer, with colour plotting available.
- The Bureau now has a facility for scanning typescript and turning it into an ASCII text file. It can cope with both Pica (10-pitch) and Elite (12-pitch) typescript as well as some dot-matrix output, provided the original is reasonably clean and clear. This means that reports and other documents which were originally typed can now be transferred to a computer for editing, reprinting or transmission.
- There is also a QMS Kiss laser printer available, printing 8 sheets of A4 per minute from a pc or the VAX. It is compatible with DIABLO 630, EPSON FX/DX and QUME SPRINT codes. There is, however, a sheet charge for this, as we have to pay the sheet-charged maintenance costs of the machine. Currently we are charging 2p per sheet as an introductory offer until Christmas, when we will review usage. EPSON graphics will also print on this machine.

Peter Flynn (CBTS8001)

Operations

The Operations section of the Bureau will be assuming responsibility for the Micro-Centre in the basement of the Boole Library from 1st October. Opening hours will be from 8.45 am to 9.30 pm. The co operation of all staff and students is requested in allowing us to shut promptly at 9.30 pm.

The Computer Room in the Science Building will be manned from 8.30 am to 9.30 pm. Outside normal working hours all Operations queries should be addressed here. Last printout distribution will be at 9.00 pm.

The 8.30am start by Operations staff means that system maintenance downtime should not affect users to any great extent. However, despite best efforts to have the system available at 9.15 am, we frequently encounter a 10–15 minute overrun. We are trying to reduce this as much as possible, but would ask for your patience and understanding in the interim.

User registration for the mainframe computers will be run from the Operations area of the MicroCentre. UserIDs will not be dealt with (nor passwords changed) in the Bureau's offices in the Science Building.

Diskettes will now only be on sale to students in the Operations area of the MicroCentre. They will initially retail for £1.50 each.

Users at remote sites should be aware that the provision of links to the mainframes is a costly exercise which has to be borne by the individual departments. Modems are currently retailing at approximately £1000 per pair, and 4-channel multiplexers at £1500 per pair (both exclusive of VAT).

Inspection of all terminal rooms during the summer recess showed that many of the terminals were in an extremely sorry state. Wear and tear from heavy use is one thing, but vandalism is quite another, and necessitates the expenditure of our limited resources on extra repair work. Since this money could otherwise be spent on improving equipment, the user community is the ultimate loser. Please treat equipment as if it were your own, and report faults as they occur to the Computer Bureau. **Users caught vandalising equipment will have their UserIDs removed from the system.**

Denis O'Sullivan (DENIS)

Electronic Mail facilities on VM/SP and VAX/VMS

With the exception of some of the brighter undergraduates, I'm afraid very few users avail of the excellent electronic mail facilities on our mainframe computers. As regards 'internal mail' within the College, I suppose this is to be expected — after all, people are more used to picking up the telephone than to sending notes (paper or electronic). (Note that the normal method of communicating with the Bureau is now by electronic mail — Ed.)

Contact with the outside (academic) world is quite another matter. Cross-fertilization of ideas is the academic's life-blood. It seems almost axiomatic that any facility that can provide fast, convenient contact between researchers engaged in

similar areas of work ought to be heavily utilized. Yet, with the notable exception of the NMRC and the Applied Psychology, Biochemistry and Geography Departments, this has been slow to happen in UCC, although the pace is now quickening.

Why this should be I can't really say — is it because of inertia or lack of curiosity or even just laziness or what? Surely not, in a modern go-ahead academic environment! Maybe there genuinely isn't any real demand for such facilities — maybe the good old parcel post serves the UCC academic community perfectly adequately. Maybe even, research at UCC is so original that there's nobody out there worth contacting. As I said, I don't know — perhaps some kind soul would care to enlighten me!

Currently, BITNET/EARN usage is completely free and will be until the end of 1987. After that we'll have to pay for the privilege, though we're not yet clear what the costs will be. Likewise, the costs of HEANET are only minuscule while it is still a pilot project. But, inevitably, it will not come cheap in the long run. If BITNET/EARN usage at UCC remains at the current level, we may have to consider dropping out and instead use a HEANET 'gateway' — possibly via UCD. This would be a definite step backwards, though.

By way of reminder, here's a 'bare-bones' outline of what we have available on our VAX and IBM systems to help you communicate with the outside world via the BITNET/EARN and HEANET networks.

VAX/VMS Commands

- **MAIL** — Provides direct mail to any other VAX/VMS user. Use the subcommand **SEND** to send mail. You will be prompted for the relevant details. For mail to sites on BITNET/EARN, specify the destination in the form `jnet%"userid@node"`. Use the subcommand **READ** to read incoming mails.
- **SEND** — This is part of the 'jnet' package. Use **SEND** to send mail (which you have prepared in a file), messages or data files to local users or to remote users on BITNET/EARN.
- **GMAIL** — Versatile utility but it takes a while to master it. Can be used to send mail to any user on BITNET, HEANET, ARPANET, CS-NET, USENET, UUCP and a host of other networks. To use, enter **SETUP GMAIL** and then type **GMAIL**.
- **POST** — To mail to a user on HEANET or JANET (the UK's internal network) etc. To use, enter **SETUP HEANET** and then type **POST**.
- **SENDGATE** — Predecessor of **GMAIL**, now out of date, really. To use, enter `JAN_SYS::SENDGATE` and follow the prompts.

Supplementary to the above commands, Peter Flynn has developed a 'gateway' between BITNET/EARN and HEANET which has been in operation since last January. While this is not of direct relevance to UCC users, who have access to *both* networks anyway, users at TCD, UCG, Maynooth, NIHE Limerick and NIHE Dublin are

not on BITNET/EARN, and they use the gateway to send and receive mail between HEANET and BITNET/EARN. If you have colleagues at these sites who wish to use BITNET/EARN, all they must do is to prepare the note to be sent by inserting a single record specifying `FOREARN userid@nodename` at the top of the file. Then, all that is required is to send the note to the UserID HEAGATE on our VAX and the rest is handled transparently. (In fact, the GMAIL system actually uses HEAGATE for notes to HEANET sites).

Within the MAIL system, if you have a lot of mail and you want to keep some (if not all) but want to delete or ‘hide away’ the rest, there is, of course, the RTFM solution — ‘read the ██████ manual’, but everything you need is there in the HELP command within MAIL. However, I guess it isn’t the best for seeing the wood for the trees. Here are a few pointers.

When you type the command MAIL, you get either a menu of the latest mail you have received and have not read yet, or the list of accumulated mail items that have been read but not deleted by you yet. Say you get a new mail message from somebody. Type MAIL and then DIRECTORY: there’s only one entry in the list. That’s because of the ‘folder’ concept. Mail is kept in folders. New, unread mail items go in a folder called NEWMAIL. Once you’ve looked at new mail (and not deleted it) it gets automatically shifted to folder MAIL. If you haven’t any new mail and you type MAIL and DIRECTORY then you’ll only be shown the contents of the MAIL folder.

Now it often happens that you’d like to keep some mail messages but have them categorised in some form. Simple — just READ *nn* (the number of the message, displayed in the DIRECTORY listing within MAIL) to read the item you want, then type FILE foldername (eg FILE RESEARCH). If the folder does not already exist, you will be asked if you really want to create it. Answer YES or NO accordingly.

To see the contents of a folder type DIR foldername. To see a list of all your folder names, type DIR/FOLD. To set yourself into a specific folder without getting a directory listing of it, type SET FOLDER foldername.

Look carefully at the SET command within MAIL. You can make some permanent settings, like, for instance, putting your own real name as well as UserID on the heading of any mail items you create. You can also specify that any mail you send or reply to be sent out and that a local copy be mailed to yourself, which you can then file away in the folder.

IBM VM/SP Commands

First, the standard ‘mail-related’ commands:

- **NAMES** — Used to define a reference nickname for local or remote users. Helps avoid having to remember oddball UserID and nodenames.
- **SENDFILE** — (or SF for short): used to send a file (in NETDATA format using IBM’s RSCS) to another user either locally or at a remote node.
- **NOTE** — Enables you to a) enter an edit environment to compose a note or message; b) send the note to a local or remote user; c) append the note to a

NOTEBOOK file (for future reference). Optionally, you can also specify that you require an automatic acknowledgement when the recipient actually reads the file.

- **RDRLIST** – (or **RL** for short): mail files always arrive in your virtual reader. **RDRLIST** enables you to a) examine (**PEEK**) individual reader files; b) discard (**PURGE**) unwanted items; c) read in (**RECEIVE**) individual items (and optionally, append them to a **NOTEBOOK**)
- **RECEIVE** — Reads in a file from your virtual reader. If it's a mail-type file it can optionally be appended to a cumulative **NOTEBOOK** file. **RECEIVE** can return an 'acknowledgement-of-receipt' to the sender if they specified that as a requirement.

These are the "basic building block" commands for mail-related processing on the IBM. We also have some public-domain utilities that greatly enhance the capabilities of these commands and even almost provide the capabilities of relatively expensive packages like IBM's **PROFS** etc.

These 'exotica' normally utilize the **MAILER** service machine on the academic IBM. This run a package known as the Columbia University Crosswell Mailer, which is a 'mail transport server' that provides a multitude of functions beyond the scope of this article. Basically, though, **MAILER** can transparently ship your mail (via other mailers and gateways) to one or more users at nodes on **BITNET**, **ARPANET**, **CS-NET**, **UUCP** and many other networks.

The routines that utilize **MAILER** (plus a few others) are all kept in a separate minidisk — to use them you must first enter the command **SETUP MAILER**. Here is a summary of them:

MAIL — Extremely versatile but greedy on memory — you'd probably require about 960k minimum to use it. Here's just some of what the **MAIL** command can do:

Read in all outstanding mail files into file **UNREAD NOTEBOOK**, then produce a **FILELIST**-like display with a summary of all the mail items (who from, subject etc). Under **PF**-key and cursor control you can then

- delete entries
- move entries to specific notebook files
- reply to specific entries
- forward entries to other local or remote users
- print entries
- and several other functions as well

Using the command **MAIL user AT node** you can carry out the same functions as the standard **NOTE** command — the difference is that **node** doesn't have to be just a **BITNET/EARN** node, it can be on any of a whole range of networks (for example **MAIL WIZARD AT OZ.ARPA**).

MAILBOOK — Use this command to read mail notebooks. Mail items in the notebooks can be printed, forwarded to another user, displayed, replied to, or deleted. In addition, mail items can be copied or moved to another notebook. When a notebook is selected for processing, it is read, information is extracted from each mail item and displayed for you to select individual mail items for processing. You can read, delete, reply to, print, or forward an item, copy or move an item to another file, and sort the menu of mail items. **MAILBOOK** is really a subset of **MAIL**.

NMENU — Very similar to **MAILBOOK** — this is a routine to aid the viewing and maintaining of a notebook produced by the **NOTE** and **RECEIVE** commands. Notebooks or mail logs are processed by this routine. These are normally files whose filetype is **NOTEBOOK** or **LOG**. **NMENU** provides commands to view, reply (using the **NOTE** or **BITNOTE** commands), forward (to another user or users), print a hardcopy copy of a note or delete individual notes contained in a notebook. This process is broken into two main parts; a) a menu screen, produced to summarize the notes contained in a notebook and b) a view screen which displays the contents of a particular note from a notebook. Commands can be entered from the menu screen or the view screen to perform various actions on a note (eg, reply to a note or delete a note from the notebook).

BITNOTE — Very similar to the standard **NOTE** command except that the destination node isn't restricted to just **BITNET/EARN** (for example **BITNOTE WIZARD AT OZ.ARPA**)

John Murphy (CBSY8001)

Floating-Point Systems, Inc

FPS are manufacturers of superminicomputers in Swords, Co Dublin. Their machines are fast processor engines which are frontended by a VAX and run FORTRAN code at around 38Mflops (our VAX does around 1Mip).

FPS have kindly offered to make one of their machines available to the Universities and NIHEs while it is on their testbed in the factory, using the HEANET connection. This means that until April/May 1987, this high-speed facility is available to UCC researchers who have heavy core-intensive numeric, especially matrix, programs to run, who are finding difficulty in running them on our VAX and IBM processors because of the time taken.

The resident software currently available (apart from FORTRAN) consists of the two public domain packages **GAUSSIAN-82** and **SPICE** (vectorised).

Proposals for taking up this offer should be directed to the Advisory Service. We are required by FPS to provide only one nominated contact per site (the Bureau), to avoid the crossing of communications paths.

The method of operation will be for user FORTRAN code to be compiled here on our VAX using the compiler they have supplied. Data files and executables are then FTP'd to the Swords VAX, and the batch job submitted remotely. Termination will cause notification by mail, and the results can then be FTP'd back from here. As

this will incur EIRPAC charges, authorisation may be required from January 1987 from the intending user's Head of Department (January is when we may have to start billing user departments for HEANET usage).

Peter Flynn (CBTS8001)

FORTRAN Corner

Starting with this edition, future editions of the newsletter will contain a section for the language FORTRAN, its uses, abuses, optimization techniques and so forth. We will start in this edition with a simple example of one the things you should be aware of and careful about when programming in FORTRAN. Many of you will already have learned this lesson painfully. This article may save others.

Let us look at a simple program in FORTRAN:

```
REAL*4 X,Y,Z
X = 1.0
Y = 2.0
Z = 0.0
IF (X .GT. 0.0) THEN Z = X + Y
WRITE(6.*) X,Y,Z
STOP
END
```

Looks OK, it compiles without errors, links and in fact runs. But the results are not as expected if one follows the logic of the program. The output from this program will be:

```
1.0  2.0  0.0
```

for X, Y and Z respectively.

The problem is with the IF statement. Compile your program with the /LISTING option specified (FORT/LIST on VAX/VMS, FORTVS progname (LIST SOURCE) for VM/CMS) and then look at the listing file generated. After the section where your source statements are listed will be a list of the variables within the program and how much space (usually in bytes) has been allocated to each of them. For the program above there will be in fact not just the three variables X, Y and Z, but also another variable called THENZ.

How is this? you may ask. The answer is that the IF...THEN construction must terminate with an END IF. Because FORTRAN allows you to embed blank spaces within the name of a variable, the THEN Z was taken as a variable name. An advantage of the acceptance of embedded spaces is that you can have more descriptive names for your variables. The disadvantage of course can be seen from the above example.

The correct way to specify the IF statement in the example could be one of the following:

- Using the THEN keyword, you must use END IF as well:

```
IF (X .GT. 0.0) THEN
Z = X + Y
END IF
```

- Without the THEN:

```
IF (X .GT. 0.0) Z = X + Y
```

If you would like to make any contributions or wish to share a FORTRAN oddie please mail ADVISORY.

Michael O'Halloran (CBSY8003)

PC prices

These prices are correct at the time of printing. There is hot competition between suppliers on many items, so please check with the Computer Bureau if you are contemplating a purchase.

Towards cheaper software

There is a considerable amount of Public Domain Software (Freeware) around, some of which is nearly as good as commercial offerings costing ten times as much, and in a few cases, it is even better. There are several outfits who run listings (on disk) of the software on the market. Most of these lists are too vast to go into here, but some of the more respectable ones are available in the Bureau for browsing. If you are trying to cut your software costs, it is well worth checking out the cheapos before committing yourself. In some cases, this kind of software is available via BITNET from a remote server (eg MACSERVE@CUNYVM for Apple Macintosh), and you may have network colleagues elsewhere who can send you software down the line. (Note that the law on copyright and the protection of intellectual property must still be observed.) These items are marked as it under 'price' in this listing. Items marked pd are in the public domain, but a charge is made for the implementation.

There is also a growing amount of Shareware available. This is a relatively new concept in software marketing, and provides a means by which a) the author of a program can get widespread distribution and popularity without the need to appoint expensive agencies, and b) you (the user) can freely and quite legally copy the product to others for this purpose and for them to try it out. Then, if you like the product, you can register either as an unsupported user for a very small sum (a few pounds), or as a fully-supported user for a larger, but still affordable, sum (usually a few tens of pounds). This latter option normally includes a properly printed manual, your own disks of the latest release of the product, telephone hotline support, and a commission paid to you when someone who has copied your disk also goes for full registration. These items are marked with a dual price as sw under 'price' in this listing.

Peter Flynn (CBTS8001)

NOTE

*All prices are inclusive of discounts and VAT
and are approximate, because of the dollar rate*

PC HARDWARE

	MONO	COLOUR (CGA)
IBM PC		
PC/XT 256k, twin floppy	1341	1636
PC/XT 512k, twin floppy	1505	1790
PC/XT 640k, twin floppy	1622	1907
PC/XT 256k, 10Mb hard disk	1757	2042
PC/XT 512k, 10Mb hard disk	1921	2206
PC/XT 640k, 10Mb hard disk	2038	2323
PC/XTS 640k, twin floppy	1911	2196
PC/XTS 640k, 20Mb hard disk	2342	2627
PC/XT286 640k, 20Mb hard disk	2888	3173
PC/AT 512k, 20MB hard disk	3347	3585
PC/AT 640k, 20Mb hard disk	3476	3714
PC/ATE 512k ,30Mb hard disk	4145	4383
PC/ATE 640k, 30Mb hard disk	4263	4501

IBM printers

PROPRINTER	410
PROPRINTER XL	499
QUIETWRITER (inc. sheet feed)	1205
COLOUR ink jet	7679

	MONO	COLOUR (CGA)
Sperry PC		
PC 200 256k, twin floppy	1784	2084
PC 400 256k, 20Mb hard disk	2173	2473
PC 250 256k, twin floppy	2055	2355
PC Micro IT 512k, 20Mb hard disk	3246	3546
PC-IT 512k, 44Mb hard disk, 80287	4234	4534

	MONO	COLOUR (CGA)
Amstrad PC		
512k, single floppy	563	788
512k, twin floppy	709	934
51211, 20Mb hard disk	1210	1350
<i>(come with GEM, GEM PAINT and 2 button mouse)</i>		

	MONO	COLOUR (CGA)
Tandon PC		
PC 256k, twin floppy	1640	1990
PC 256k, 10MB hard disk	1758	2108
PC/AT 640k, 20MB hard disk	2711	3061
PC/AT 512k, 40MB hard disk	3891	4241

PC SOFTWARE

VTERM (VT100 emulator)		215
KERMIT (file transfer & emulations)		fw
PROCOMM (terminal emulator)		fw
DBASE 3+ (database)		720
MULTIPLAN (spreadsheet)		171
LOTUS 1-2-3 (spreadsheet)		750
LEX (wordprocessor)		375
WORDSTAR (wordprocessor)		406
PC-WRITE (editor & wordprocessor)	sw 15/94	
PERSONAL EDITOR		109
XEDIT		pd 5
DISPLAYWRITE 3 (wordprocessor)		468
\TeX (typesetting system)		pd 200
Device drivers for \TeX (each)		pd 100
<i>FONTASY</i> (display lettering)		50
Professional FORTRAN (compiler)		526
IBM FORTRAN (compiler)		307
IBM PASCAL (compiler)		307
IBM MACRO (assembler)		153
XENIX (operating system)		380
XENIX (development system)		440
EXECUVISION (slide presentation)		501
IBM GRAPHING ASSISTANT		117

PC Screen to 35mm Slide Presentation

The Polaroid Palette computer image recording system produces 35mm colour slides of images created with an IBM PC or compatible, and allows you to substitute any of 72 colours for the 16 IBM CGA colours. Images maybe text or graphics generated by an application package such as Lotus 1-2-3, IBM Graphing Assistant or Execuvision. The Bureau's Polaroid Palette is now available for general use. For further information contact the Help Desk or mail me.

Michael Gordon (CB0P8006)

Administrative Systems

In the past year the Student Records system has been integrated with the new Fee system so that the entire Student Records system is now on-line.

Registration captures all the student data, allowing class lists and Local Authority bills to be produced after each registration session.

Examinations were processed for the first time, printing Marksheets and Board-sheets. Marks were read in using specially-produced Optical Mark Reader forms filled in by the departments. This enabled marks to be updated to the system in minutes, and judgments to be assigned automatically. Results letters and lists for the Stone Corridor noticeboards could then be produced automatically.

It is hoped to be able to give each department the data on their students, on a PC disk, in the near future.

Lyn Behan (ADFN8103)

DOBIS/LIBIS Library System

DOBIS/LIBIS is an integrated library management system developed by IBM at the universities of Dortmund and Leuven (Louvain). It provides online facilities for:

- public access to the catalogue
- searching
- cataloging
- circulation control
- acquisitions
- periodicals
- catalogue maintenance

All these functions, except periodicals, are at present operational in the Boole Library. Batch facilities available include:

- report generation
- catalogue production
- accessions bulletin
- management information statistics
- database maintenance

There are currently approximately 130,000 documents on the system catalogue with about 9,000 users.

Public access

A number of terminals are available in the Boole Library at which library users have direct access to the system catalogue. Thus the user can search online, helped by user-friendly screens, for any catalogued document via author, title keyword or subject area. The locations of the various copies of the document can then readily be found. Library staff are always available if the user has any difficulty at the terminal.

Jerry Buckley (ADLB8002)

INFO Service Virtual Machine on the Academic IBM

If you enter the command `CP QUERY NAMES` (or `SEND` it from the VAX), you'll notice an oddball UserID called `INFO`. This is a sort of jack-of-all-trades Virtual Machine which accepts commands sent by users on either of our IBMs or from the VAX (or, indeed, in theory from any BITNET/EARN node, though for the present it's closed off to them). `INFO` tries to carry out these user commands and return the result to the user. The `INFO` server code is written in REXX — see the article on using REXX elsewhere in this newsletter.

To send a command to `INFO` from the Academic IBM just enter `TELL INFO` command. To send a command to `INFO` from the Administrative IBM just enter `TELL INFO AT IRUCCIBM` command. To send a command to `INFO` from the VAX just enter `SEND INFO@IRUCCIBM` command. For details on available commands just send `INFO` a question mark ('?') or the word `COMMANDS`. `INFO` can perform a variety of tasks:

UserID Lookup

If you want to find out who is user `SCCS6099` then `INFO` can oblige — just send it the command `WHOIS SCCS6099`. Conversely, if you want to find out the likely UserID of C J Bloggs just send `INFO` the command `LOCATE Bloggs:` and you'll get a list of all the Bloggs family who've got a UserID.

Finding BITNET/EARN node details

If you want to find where is node `DHDIBM1` or who is the contact person at node `YALEVM` or what machine they use at node `WEIZMANN` then `INFO` can oblige — just send `INFO` the command `BITNET NODE nodename`. Conversely, if you want to get a list of node names at a particular location then enter the command `BITNET LOCATE searchstring`.

Finding HEANET node details

If you want to find out the `HEANET` node name for a particular institute then just send `INFO` the command `HEANET SITES`. If you want to find out full details of a particular node then send `INFO` the command `HEANET NODE nodename`.

UCC Staff Details

This one will be available 'real soon now' — all we need is a copy of the file used to compile the internal telephone directory. This facility will enable you to look up the phone number or room number etc. of any staff member.

Bulletin Board

`INFO` also has a Bulletin Board facility (admittedly rather unsophisticated but it works!). Basically, `INFO` can receive a file from a user and make it 'publicly available' with the option of allowing entries to be appended to it by any other user. `INFO` can send you a list of all Bulletin Board items or of all items that have been added to since a specified date. Possible uses would be:

- Notices for Clubs, Societies etc.
- Notices for individuals (want to sell your bike?)
- Discussion/ Conference Groups (correspondence on the meaning of life?)

The only restriction is that nothing illegal, defamatory or in breach of copyright be included! Send INFO the command BULLETIN for further details.

Message Re-Broadcasting ('chat') Facility

The INFO command SIGNON causes any further messages that you send to INFO (that aren't preceded by a slash [/]) to be re-broadcast to any other user that happens to be signed on. This sounds like a rather trivial facility but it does have long-term potential uses, eg for multi-user conferencing within UCC, rather like the existing BITNET RELAY service.

John Murphy (CBSY8001)

Using REXX on VM/SP

First came the EXEC Command Language Interpreter. Every VM user has had occasion to use EXEC at some stage — though only a very few bothered to do anything exotic with it. Next came EXEC2 — more of the same with a few frills added. EXEC2 appeared in 1979 and certainly seemed a lot more flexible than its predecessor.

An IBM employee named Mike Cowlshaw, working at their Havant plant, didn't think much of EXEC2, however, and over the next 4 years worked up about 6000 hours of his spare time designing and implementing what he felt was a more versatile Command Language. It caught on like wildfire within VNET (IBM's world-wide network) and eventually the powers-that-be within IBM decided that it was A Good Thing. So it went public with VM/SP Release 3 in late 1983 under the official title of System Product Interpreter but more familiarly known as REsource EXecutive or REXX. Since then it has also been implemented on the IBM PC as part of the PC/BOND product.

REXX is an interpreted language (rumours of a compiler abound but nothing is yet officially available) and hence isn't really suitable for heavyweight tasks. It is ideal for

- Jobs that one might perform only intermittently
- Tailoring of standard CMS commands
- XEDIT macros
- Extended-precision simple-arithmetic desk-calculator tasks

(I've been told that REXX can add two one-million-digit numbers in a few seconds but I haven't checked that out!)

REXX has a syntax akin to PL/I or PASCAL. Because of this and because of the versatile debugging facilities it has even been used by Computer Science departments in the US to teach 1st year undergraduates the principles of Structured Programming.

Currently, several CMS commands and XEDIT macros are written in REXX. A full description of REXX syntax, functions etc is beyond the scope of this article. Instead, I've included a demo program to illustrate REXX in use. It's the sort of quick-and-dirty once-off task for which REXX is ideal. I would suggest that you sit yourself in front of an IBM terminal, enter HELP REXX MENU and then peruse the program and check up on any items that aren't immediately obvious.

```

/*REXX Demo
* Enter HELP REXX MENU, study this routine and check up on any
* commands, functions etc. that you want to know more about.
* For clarity. all REXX variables are in lowercase and the
* commands, functions etc are in uppercase.
*
* This routine reads in a list of up to 100 names and addresses.
* sorts them in Surname sequence and then print the sorted list.
* The list is free-format, with a comma separating the name and
* address. The name part is composed of ONE OR MORE forenames
* or initials, separated by at least one blank space.
* See the end of this file for sample data.
*
* We could write modified records to a temporary file,
* sort it via the CMS SORT command and then type the results.
* Just for show, however, let's see how REXX can handle sorting etc.
*/

ADDRESS COMMAND;
DO n = 1 To 100                                /* NOTE- 100 records max */
  "EXECIO 1 DISKR DEMO DATA A"                /* Stack the next record */
  IF RC=0 THEN LEAVE                          /* (unless we meet E.O.F) */
  PARSE FULL name "." address /* Read the data record from the stack */
  /* Note the embedded function calls in the next two statements */
  surname = STRIP(WORD(name,WORDS(name)), "B") /* Extract surname */
  forename = STRIP(DELWORD(name,WORDS(name)), "B") /* "" forename */
  address = STRIP(address, "B") /* "strip" address */
  array.n = LEFT(fullname,25) address /* Put details in "array" */
END
"FINIS DEMO DATA A"
max_names = n - 1                               /* Count of data records read */

IF max_names > 1 THEN
  DO /* a shuttle sort -seems to be faster than bubble sort */
    no_to_be_sorted = max_names - 1
    DO k = 1 TO no_to_be_sorted
      DO j = k TO 1 BY -1
        jplus1 = j + 1
        IF array.j <= array.jplus1 THEN ITERATE K
        temp = array.j /* Do */
        array.j = array.jplus1 /* a */
        array.jplus1 = temp /* swop */
      END /* Do j */
    END /* Do k */
  END /* of the shuttle sort */

```

```

/* Finally, type out the results */
DO n = 1 to max_names; SAY array.n; END
EXIT;

/*
** Sample input data - file DEMO DATA

P.J. Flood, The Banks, Lee Road, Cork.
Toby Joseph Patrick Jugg, The Manse, Clonakilty, Co.Cork.
Mary Catherine Eleanora Heggerty, Upper Lyreacrompane, Listowel, Co.Kerry.
Mary Jane Eleanora Heggerty, Lower Lyreacrompane, Listowel, Co.Kerry.
Mr. B.I.G Spender, 99 Rochestown Road., Cork.

** Sample output produced

Flood, P.J.                The Banks, Lee Road, Cork.
Heggerty, Mary Catherine  Upper Lyreacrompane, Listowel, Co.Kerry.
Heggerty, Mary Jane       Lower Lyreacrompane, Listowel, Co.Kerry.
Jugg, Toby Joseph Patrick The Manse, Clonakilty, Co.Cork.
Spender, Mr. B.I.G.       99 Rochestown Road, Cork.
R: I=0.15/0.19 14:49:53

```

This lot took me about 30 minutes to type, test and debug. I wonder how long it would take using a ‘standard’ compiler?

John Murphy (CBSY8001)

EVE

Once there was EDT, now we have EVE. Much to the relief of many, EVE is not Adam’s partner but the new Extensible Vax Editor and the replacement for EDT. EVE is an interactive editor just like EDT and you can use it to create and edit many types of files — letters, memos or programs.

Why another editor? EDT has been around now for a number of years and since it was first introduced VAX/VMS has under gone many enhancements. So too have devices like terminals, so EVE has been developed to take advantage of these advances.

EVE has many advantages over EDT of which the following are most important: it is EDT plus much more.

- It is faster and demands less system overhead
- Two ways to insert text — Insert or Overstrike mode
- It supports multiple windows — you can view two files on the screen
- VAXTPU procedures: you can automate editing and programming sessions

Like EDT, EVE is written in the VAXTPU programming language. VAXTPU is a high performance programmable text processing utility. You can tailor the EVE editing interface to suit your editing style with VAXTPU.

How do you invoke EVE? Use the command `EDIT/TPU filename.type` (the PF2 key is the HELP key, as with EDT in full screen mode).

Call to the HELPDESK (4th floor, Science Building) and collect a handout which contains a tutorial for EVE if you would like to get started with it.

Michael O'Halloran (CBSY8003)

So Your Program Has Gone Crazy?

*(Or what can you do when the results say **HOT** and the the program says **NOT**)*

There can often be a crisis of faith where computers are concerned. However, the most testing moments are similar to the situation I'm about to describe.

- Your program has compiled (beautifully) — no errors — no warnings — not a hint of trouble
- Linking and loading had gone a treat
- Then you run the program
- **DISASTER** — The program bombs out with a) an 'access violation' or b) a 'shared page altered'.
- Less dramatically, the program produces a) different results to a previous run with the same data and program (!) or b) results that are clearly wrong.
- Self-doubt. You double check the data — perfect! You double check the program — irrefutable logic!
- Crisis of Confidence. If I'm right — who is wrong? No more computers for me!

If you find your faith slipping here are a few points upon which to meditate:

- Have I initialised all my variables? If you haven't you could get 'rubbish' left in memory after someone else's program.
- Are my arrays big enough? When you use variables to index into an array the compiler cannot check in advance whether you are overflowing bounds of the array or not. If you do overflow you may try to access memory which is not assigned to you and hence have 'access violations' etc. Should that memory belong to you it certainly won't be the data that you're expecting to be there (eg on the VAX use the /CHECK=BOUNDS compiler option if necessary)
- Are my COMMON areas really common? Similar problems to the above occur if common areas do not match up. (This can happen especially if a program and subroutine are not both recompiled if a change has been made to the common.)
- Have I been fair to my subroutines? Did pass the parameters the subroutine expected? Note specially whether they are to be variables or arrays. Also ensure that, if you are using library subroutines or functions, you understand the parameters properly.

- Could I be underflowing or overflowing? In calculations the variables must be large enough to hold the results and the intermediate values. This is very important in floating point calculations and double-precision may be required.

I hope this will stem the flight back to the abacus.

Henry O'Keefe (CBSY8002)

SPSS-X

SPSS-X is now available to users of the VAX 11/780. Although SPSS-X resides only on the IBM 4341, a facility has been implemented whereby a VAX user can prepare an SPSS-X job, and then by issuing a few commands have the job sent to the IBM for processing. The output can either be printed directly on the system printer or can be re-directed back to the user on the VAX. There are a number of restrictions however and these are

- the user must have a corresponding account on the IBM (If you are in doubt whether you have or have not check with the Operations Section)
- You cannot use the SPSS-X commands SAVE and GET.
- You are limited to using one and only one data file for each job.

To use this facility first type **SETUP SPSSX**. If you need information as to usage type **SPSSX HELP**. To actually use the facility just type **SPSSX** and you will then be prompted for the name of the SPSS-X control file, followed by the name of the data file. You will be prompted for the destination of the output: selecting the letter P will send the output directly to the printer, while selecting the letter F will cause the output to be returned in a file to you.

A number of manuals pertaining to SPSS-X are available from the Advisory Service. They are:

- SPSS-X Basics
- SPSS-X Users Guide
- SPSS-X Introduction to Statistics Guide
- SPSS-X Advanced Statistics Guide

No department should be without them! For the first time user of SPSS-X there is an Introductory Video Course available in the Computer Bureau. This course is complete with notes and is an essential and excellent introduction to preparing your data and using SPSS-X. Contact the Help Desk for further information.

Michael O'Halloran (CBSY8003)

Handy Hints for VM/SP

Here is a method of avoiding accidental erasure of files (or at least making it more difficult to erase them!)

You may not be aware that mixed case filenames or types are permitted by CMS. In other words, **filename = mY, filetype = fILe** is quite valid. The **filemode**, however, *must* be in uppercase. This has a handy spinoff insofar as the user cannot erase a file with mixed-case name directly from the terminal other than via

- ERASE fn *
- ERASE * ft
- LISTFILE(EXEC) and CMS ERASE
- ACCESS nnn (ERASE) etc

(and you wouldn't do any of those daft things, now would you?)

Because any command typed in from the terminal is converted to uppercase, any files with lowercase names can only be deleted in a roundabout way — hence the 'safety factor' in using mixed or lower case names.

The following general-purpose REXX routine could be used to rename a file to an entirely lower-case filename and filetype — to use it just type LOWERCASE fn ft fm.

```
/* LOWERCASE -routine to rename a file to lower case filename/type. */
ADDRESS COMMAND
PARSE UPPER ARG fn ft fm .
IF fn = "" | ft = "" THEN EXIT 999
IF ft = "" THEN ft = "A"
"STATE" fn ft fm
IF RC ^= 0 THEN EXIT 999
lowercase_file = TRANSLATE(fn ft,'abcdefghijklmnopqrstuvwxy',
'ABCDEFGHIJKLMNOPQRSTUVWXYZ') fm
"RENAME" fn ft fm lowercase_file "( TYPE )"
EXIT:
```

The only convenient way to erase an individual file that has a mixed or lower case filespec is as follows:

```
LISTFILE(EXEC)
XEDIT Cms EXEC A , delete irrelevant entries and FILE
CMS ERASE
```

John Murphy (CBSY8001)

MINITAB

MINITAB version 5.1 has settled down nicely on the VAX 11/780. The transition from version 82.1 should have been totally transparent to all concerned. To see what is new in version 5.1, enter MINITAB and type the command NEWS. Of most interest in the latest release is MINITAB's support for graphics. Commands like HISTOGRAM and PLOT have been enhanced (GHIST and GPLOT), to provide the user with a higher quality of output than was originally available. The device drivers available for MINITAB at the moment are TEKTRONIX 4010, 4105, DEC VT125, VT240, HDS, TAB and HP7470A devices. You will notice that the CALCOMP 1037 and CALCOMP models M81 and M84 are not included in this list. Hopefully these will be available soon.

MINITAB will also be available on the MICROVAXen in the Dairy Science terminal room and the Statistics Department, and a new release will be installed on the IBM 4341 towards Christmas. Keep a look out for an announcement to this effect on the log message.

Michael O'Halloran (CBSY8003)

P-STAT

P-STAT was made available on the VAX a) to provide a data management tool which did not require extensive or expensive training to use, and b) to provide those who are not professional statisticians with a range of statistical analysis routines combined with powerful file management and reporting capabilities.

The package has all the common statistical procedures normally asked for by those doing survey analysis, and the design quality of output is substantially higher than that from SPSS or MINITAB, allowing tables and listings to be copied directly into reports, theses etc, if they are printed out on a quality printer. It also provides sophisticated character-handling, a facility not available in many statistical (and even some data management) packages. It can read and write SPSS and BMDP Save Files, which means you can pass data between the packages in order to use other statistical routines.

P-STAT is completely interactive: to run it, type `SETUP PSTAT` and then just type the command `PSTAT`. On-line Help is available within the package and there is a reference manual and command guides available in the Bureau. The only quirk to remember is that commands can be multiline, each line ending with a Carriage-Return, so the final line of a command must end with a dollar sign (\$).

Peter Flynn (CBTS8001)

The `SETUP` command on IBM and VAX

`SETUP` is a new command common to IBM and VAX which we have introduced to simplify access to commonly-used software facilities.

SETUP on VAX/VMS

There is no point in having myriads of logical assignments, global symbols etc for every installed package, compiler etc for everybody — regardless of whether or not they need to use them. That only wastes memory space. It is much more practical that individual users have an easy-to-use mechanism enabling them to specify what items they really need, whether it be a particular package, applications program, utility, compiler, interpreter, library file, etc hence the `SETUP` command.

Type `SETUP ?` for details.

SETUP on VM/SP

Traditionally, new compilers, application packages, utility routines, libraries etc have all been lumped together on the CMS Y-Disk. The result is that our Y-Disk is a veritable cornucopia of files — a large number of which nobody is quite sure any longer belong with what package!

This is wasteful in many ways — when you `LOGON`, CMS creates an in-core table of *all* accessed files. If lots of those files are inessential and there are a lot of users logged on then we suffer unnecessary memory overheads!

For some time now, all new application packages have been put in their own minidisk area. In future, as new versions of applications currently crammed onto

the Y-Disk are installed, this approach will become the standard. (Also, the Y-Disk will get a long-deserved spring cleanup!)

However, all this involves a few problems. Users need a simple transparent mechanism to link them to the application that they want to use. They need some mechanism for ‘remembering’ which linked mini-disk is used for which package, what mode it should be accessed as etc...hence the SETUP command.

SETUP is quite versatile (OK, I wrote it so maybe I’m biased anyway!). It provides a few useful frills:

- It can decide whether or not it needs to LINK and ACCESS a separate mini-disk for a specified package/application.
- If it has to link to a specific minidisk then it can determine if it should use the next available address or a specific LINK AS minidisk address. Similarly, it can determine if it should access in a specific access mode or use the next available letter.
- Optionally, SETUP can invoke a ‘prologue’ routine for the specified package. The prologue routine might issue FILEDEFs, GLOBAL TXTLIBs, NUCXLOADs etc, and other such tasks to save time for the user.
- Optionally, once the user tells SETUP that they have finished using a specific package, it can invoke an ‘epilogue’ routine. The epilogue routine might clear FILEDEFs, issue NUCXDROPs etc
- SETUP ‘remembers’ which minidisks are linked and/or accessed, which package belongs with which minidisk etc. Should the user run into problems and have to re-IPL CMS then they can have everything re-accessed as before just by typing SETUP

Enter SETUP ? and then HELP SETUP for details. Enter SETUP (LIST) to see the packages etc that SETUP can currently deal with.

John Murphy (CBSY8001)

TROLL — linking to files owned by other TROLL users

When you logon and enter the TROLL environment you have two disk areas allocated to you:

- Disk 560 — your own disk space, to which you have read and write access
- Disk 561 — the TROLL System Disk area, containing commands, macros, help files etc

To list files on your own disk area just enter LISTF DATE SIZE ALL;. To list files on the TROLL System Disk area just enter LISTF SEARCH TROLLSYS DATE SIZE ALL; (that’ll list off *thousands* of files, though!).

You could list all the MACRO files on TROLLSYS via LISTF SEARCH TROLLSYS DATE SIZE MACRO*;. Similarly, you could list all the INFO files on TROLLSYS via LISTF SEARCH TROLLSYS DATE SIZE INFO*;

The `SEARCH` parameter causes the specified area to be searched. Remember that the `TROLLSYS` files are always accessed once you've IPLled `TROLL`.

Currently, it is assumed that no highly confidential `TROLL` data files exist at `UCC`. Consequently, all the `TROLL` disk areas can be linked to by any other user in `READ-ONLY` mode and the files can be read. This makes sharing of files between users fairly straightforward. For instance, data for student exercises could be stored in one specific `UserID`.

Suppose you needed to look at a large data file owned by `STEC8099`. First, tell `TROLL` to access user `STEC8099` via the command `ACCESS STEC8099;`. That command links you to disk 560 owned by user `STEC8099`. Some users have an extra `TROLL` disk area, normally disk 561. This can be linked via `ACCESS STEC8099 ADDRESS 1377;` (1377 is the decimal value of disk address 561 hexadecimal in case you're wondering!).

Next, `TROLL` has to find out what files are on that disk: type `SEARCH STEC8099;`. To list all `DATA` files owned by the `UserID` to which you've linked type `LISTF SEARCH linkid DATE SIZE DATA*;`. To find whom you are connected to enter `LKACCESS;`.

Remember that while you are accessing another user's disk area they may be logged on and be updating files. If `TROLL` should object because of problems accessing a file just enter `REACCESS user-id;`. Be aware also that if you are accessing another user's disk area and they are not logged on but happen to log in in the interim then they have only read-only access to their own disk area! Therefore, make a point of finishing with the other user's data as soon as possible and enter the command `DELACCESS userid;` to let the user get full read-write access to the files.

John Murphy (CBSY8001)

New USERLIBRARY on VM/SP

If you happen to be one of those rare paragons, a user who actually remembers what appears within the august pages of the Bureau Newsletter, then you may recall that several issues ago there was a somewhat euphoric description of a new version of the `USERLIBRARY` system. It was due to be installed 'real soon now' on the `IBM`. Alas, it didn't meet the deadline. I had no intention of incurring an effluent-striking-air-vent effect by introducing a whole new set of commands halfway through term. The next year it had to be postponed yet again because of an avalanche of troublesome new goodies (remember the trauma of `VAX/VMS 4.0` — well, that was just one of many). At any rate, 'real soon now' is now!

Why bother with a new system, you cry! Simple — we all like a bit of peace and quiet so as to just get on with more useful things! Unfortunately, the old system has been something of a *Sword of Damocles* hanging over *every* person using it. Basically, any talented and enterprising hacker could do *anything*, from thievery to destruction of any or all files in the `USERLIBRARY`. The 'expert' hacker could be tolerated — at least they usually knew enough not to rock the boat unduly. The 'dabblers' have been the real danger — *they* have always been likely accidentally

to do a lot of time-wasting damage purely through sheer ignorance. Other problem areas with the old USERLIBRARY have been the amount of regular maintenance required, its inflexibility and lack of any form of disk space quotas etc.

The new system is based on the 'Service Virtual Machine' concept — basically, all files are 'managed' by the server. Security will hopefully (touch wood) be far less of a problem than before. We've tested it out in a 20-user environment and it behaved acceptably — however, only time will tell as regards response rates under heavier loads.

The following 'old' commands ceased to function at the end of September 1986: LOADFILE, KEEFFILE, ERAZFILE, LIBLIST.

HELP details are available on the following new commands: LIBQUERY, LIBGET, LIBKEEP, LIBRENANE, LIBARCH, LIBPROT, LIBERASE.

For an overview of the new system, enter HELP USERLIB and for further details enter HELP `new_command_name`.

John Murphy (CBSY8001)

Bureau Staff

Bureau Staff can be contacted at the following UserIDs. Please use the telephone only if you cannot send email. The nodenames of the mainframes are:

- IBM 4341 (Academic) — IRUCCIBM
- IBM 4341 (Administration) — UCCADMIN
- VAX 11/780 (Academic) — IRUCCVAX

Initial enquiries, new projects and requests for help should be directed to the HelpDesk.

Lyn Behan	<i>K-Accounts & Database</i>	ADFN8103	×2610
Geraldine Buckley	<i>Operations</i>	CBOP8017	×2227
Jerry Buckley	<i>Library System</i>	ADLB8002	×2489
Margaret Buckley	<i>Fees & Student Records</i>	ADRE8102	×2609
David Chandler	<i>Operations</i>	CBOP8008	×2227
Madeleine Cotter	<i>Operations</i>	CBOP8009	×2227
Robert Dumigan	<i>Operations</i>	CBOP8015	×2227
Brian FitzGerald	<i>Admin Database Administrator</i>	ADRE8003	×2610
Peter Flynn	<i>General Management & Policy</i>	CBTS8001	×2215
Michael Gordon	<i>Administrative Projects</i>	CBOP8006	×2370
Paul Keegan	<i>Operations</i>	CBOP8018	×2227
Donal Kingston	<i>Documentation</i>	CBD08001	×2489
Katherine McLoughlin	<i>Appointments & Administration</i>	CBWP8002	×2215
Catherine Malone	<i>Operations</i>	CBOP8019	×2227
John Murphy	<i>Systems Management</i>	CBSY8001	×2226
Seán O'Farrell	<i>Payroll & Exams</i>	ADRE8007	×2610
Michael O'Halloran	<i>Systems Analysis</i>	CBSY8003	×2226
Henry O'Keeffe	<i>Systems Analysis</i>	CBSY8002	×2437
Anne O'Mullane	<i>PCs, terminal bookings</i>	CBWP8014	×2622
Denis O'Sullivan	<i>Operations Management</i>	DENIS	×2502
A N Other	<i>Academic Advisory</i>	ADVISORY	×2611
Nora ní Riabhaigh	<i>Operations, MicroCentre</i>	OPERATNS	×2735
Paul Waters	<i>Help Desk, consultancy</i>	HELPDESK	×2609