EUUG

EUROPEAN UNIX® SYSTEMS USER GROUP NEWSLETTER

Volume 4, No. 3
AUTUMN 1984

EUUG
European UNIX† Systems User Group

Newsletter   Vol 4   No 3

Autumn 1984

Call for Papers
European UN*X Systems User Group Spring Conference
Paris (France), April 1-3, 1985

The EUUG spring conference will be held at le Palais de Congres, Paris on April 1st, 2nd, and 3rd, 1985. People wishing to make a presentation should submit a 250 word abstract to the Techical Sessions Chairperson no later than Dec. 31st, 1984 or via snail mail to Helen Gibbons by Nov. 30th.

Submissions must include:

Presenter's name[1]	Institution	Network address	Mail address
Office phone number	Home phone number	Language of presentation[2]	Audio/Visual needs [3]

[1]   In case of multiple authors, full information should be given for the presenter only. The only additional information required for the co-authors is their institution.

[2]   Simultaneous translation facilities will be available, provided technical translators to both English and French are available. Requests for simultaneous translation of languages other than French and English should be made to Helen by National Group chair people as soon as possible.

[3]   We might require speakers to use 35mm slides (instead of overheads).

Receipt of abstracts will be acknowledged via electronic mail.

Full papers (in troff -ms) will be required by Feb. 15th, 1985 and should not be submitted until acceptance is given.

Apart from the Technical Sessions Chairperson, the following people may be contacted for other information:

Michel Gien (Conference organiser)   Helen Gibbons (EUUG secretary)
CNET, PAA/TIM   Owles Hall
38-40 Rue du General leclerc   Buntingford
ISSY-LES-MOULINEAUX, France   Hartfordshire SG9 9PL, England
Tel: +33 1 638-6287   Tel: +44 763 73039
UUCP: mcvax!vmucnam!mg   UUCP: qtlon!euug!helen (soon)

Network Events Ltd. are organising an exhibition in le Palais de Congres to be open on April 2nd, 3rd, and 4th. Information may be obtained from:

Beryl Jarvis or Graham Shearsmith
Printers Mews, Market Hill
Buckingham MK18 1JX, England
Tel: +44 280 815226
Telex: 83111

Final notes:

Each speaker will be allocated 30 minutes, which means that we will be having about 21 technical presentations. Six speakers have already be selected, leaving only 15 slots. Conference registration will be limited to 700. The tutorial sessions will be conducted en Francais and no translation will be provided.

Conference motto (unofficial):

Having a lovely wine.
Wish you were beer.

David M. Tilbrook (Technical Sessions Chairperson)   17th October, 1984
Tel: +44 1 581-8155, +44 252 540377
UUCP: {mcvax, qtlon}!ist!dt

European UNIX System User Group Meeting
Cambridge University, 19/21 September 1984

Peter Collinson
Secretary

Introduction

I hesitate to start this report with the magic words 'this was another good conference', but it was. There were three components: technical sessions in the Chemistry Lecture theatre; industry sessions held in the University Arms Hotel and organised by /usr/group/UK; and a large exhibition organised by Network Events Ltd., a company specialising in (yes, you guessed it) running exhibitions. I did manage to get to the exhibition, but didn't make the industry sessions, I will endeavour to get someone to write a report of what happened there.

This report is again helped vastly by the abstracts which were supplied by speakers before the event, for which many thanks. We hope that the papers submitted for the conference will be printed soon. It is also hoped that the Proceedings for Paris will be printed before the conference, so speakers can talk about other things and confuse us all.

Day 1 - 19th September

Well, this wasn't Day 1 for me, I had spent most of the day before in committee meetings of various sorts. This meant considerable quantities of local ale the previous night. Nothing daunted, I managed to make the really early breakfast and start the long trek down to the lecture theatre. It didn't rain - I had my lucky umbrella.

9.45am   **Opening of the conference**
          Richard Stibbs, Cambridge University

Richard had mostly domestic matters to talk on, the most important being where to find the best beer in Cambridge. He violently denied the rumour that Greene King had sponsored the conference, and said that 'Abbot Ale' was not a trademark of that well known footnote.

After that David Tilbrook had a few minutes to introduce the people who were chairing sessions. He had brought a cutting from Computer Weekly of 13th September. This was in their gossip column and reads (all the spaces are theirs):

Peculiar

ANYONE wanting to attend one of the scintillating technical sessions at the forthcoming European UNIX systems user group meeting in Cambridge has to apply to (wait for it): (qtlon,ukc)! ucl-cs! nigel qtlon! ist! dt.

Is it any wonder that the rest of the world thinks that Unix people are a bit peculiar. Unix? Yukk! (sic) Geddit?!

The item was signed *Chad*. Recent attempts to send mail to their quoted net address: Yukk! (sic) Geddit?! has resulted in a profound silence.

9.50am UNIX Europe
Vanni Papi, UNIX Europe Ltd.

This was really an opportunity for Mr. Papi to announce AT&T's hospitality suite. More seriously, he briefly described the state of the art in UNIX Europe. It is still quite a small organisation of only 12 people. They are responsible for all UNIX licensing in Europe and will also be running training sessions.

9.55am Communication Product Announcement
Joanne Miller, AT&T Technologies

Abstract

This talk will describe the current set of COMMKIT‡ Networking Software products for UNIX System V. A directional statement of the future of UNIX networking will be covered that relates to the network architecture, the operating system and the COMMKIT line.

Comment

The major thing of interest to me was the announcement of the availability of the new uucp 'Honey Danber' which Brian Redman talked about at the conference in Nijmegen. It seems that it will never be on a System V tape — it will always be a separate product.

The talk boiled down to a statement of intent which said: AT&T will use agreed international and de-facto standards, will document the protocols and make this documentation available so that other systems can converse with AT&T products. They will also supply technical support for customers.

10.28am Development of the MicroVAX 1
Robert Short, DECWEST Engineering

Abstract

Last year Digital announced the first of a new family of computers called MicroVAX. MicroVAX is a proper subset of the VAX architecture intended for VLSI implementation. The first member of this family, the MicroVAX I, was developed at DEC's engineering site in Bellevue, Washington, making it the first major DEC hardware product designed and developed outside of New England. The MicroVAX I processor includes a custom MOS VLSI datapath chip, a cache and translation buffer. This talk describes the MicroVAX I hardware development, including:

- The advantages of subsetting the full VAX architecture to allow a small machine to be designed.

- The goals of the hardware design team.

- The microcode/hardware tradeoffs necessary to provide good performance while still meeting the cost/size/power goal.

- A fairly detailed overview of the hardware architecture of the machine.

Comment

The aim of the exercise was to make a VAX on a chip with the same cost as the M68000. The time scale for the project was 18 months, which is a short development period for a project of this sort. The time scale meant that the machine was based on the Q-bus which so that a totally new bus was not required. It was also decided to simplify matters by removing some of the 'fancy' but little used instructions from the VAX architecture. The machine can run non-privileged VAX native mode

---

‡ COMMKIT is a trademark of AT&T Technologies.

programs which are able to run under VAX/VMS on other machines in the VAX range.

The removal of PDP-11 compatibility mode, the decimal instruction set and some of the string instructions meant that there was considerably less micro-code than a full VAX. Instruction decode is easier because addressing is simplified by the removal of the PDP-11 instructions, the decode looks at a single byte at a time. All virtual addresses are 32 bits.

The following decisions were made in order to get a 32-bit processor on a 16-bit bus. The code block size is a longword and block mode on the Q-bus allows the processor to read 2 words at a time. The CPU microcode does not know about the 16-bit bus and does not wait for writes to complete down the bus. There is an instruction pre-fetch FIFO to speed things up a bit.

Well, DEC had one of these in the exhibition running ULTRIX. It apparently is faster than the VAX-11/730 but slower than the VAX-11/750.

☞ Coffee ☜

11.20am **Large Systems experience - System 370**
        J. Carl Hsu, AT&T Bell Laboratories

Abstract

This talk will discuss will discuss the problems encountered and the experience gained during the development of the UNIX system for the IBM System 370.

Comment

Indian Hill is the largest computer centre in Bell Labs. Disc storage is 10^{12} bytes! The organisation is a technology leader in computing and networking applications, providing standard UNIX systems to other AT&T sites and also generating add-on packages to the standard system.

With the IBM machines, the objectives were to put UNIX on the IBM 370 in order to provide the full power of large systems and provide a working system for the computing centre environment. The implementation was a joint AT&T/IBM effort and uses an IBM supervisor for low level functions. The software is standard UNIX and is mostly in C. The system will support 180 users.

P and V semaphores were used instead of sleep/wakeup because the IBM architecture allows processes to be arbitrarily blocked in the kernel.

11.48am **The portability dream**
        Neil Urquhart, Sphinx Ltd.

Abstract

The range of microcomputers offering UNIX has grown and with it the requirement to transport existing software between operating systems. The difficulties in porting software is seen able to be classified as four categories: UNIX development; machine implementation; programmer techniques; and software dependency.

The development of the UNIX operating system, from Version 6 through to System V, is overviewed, with special note being made of their idiosyncracies. Discussion of the contribution of the Berkeley implementation is included together with its influence on the new release of operating particular features which influence the transportability of application software or programs.

Examples of idiosyncratic code are presented to illustrate differences between machines, UNIX implementations and programming styles. C is claimed to be a portable language; it is discussed, with examples as to how much it encourages programmers to deviate from the 'straight and portable'. Some of the UNIX tools which are known to avoid portability difficulties are illustrated together with their limitations. Software dependency is discussed and the difficulties it can cause are

illustrated.

12.07pm Multi-processing on UNIX
Steven J. Buroff, AT&T Bell Laboratories

Abstract

This talk will describe the development of a UNIX operating system that runs on multiprocessor configurations. The system currently runs on AT&T 3B20 computers, but the ensuing discussion applies equally well to other machine architectures that support a multiprocessor environment. Existing user level C programs can run without recompilation on uniprocessor and multiprocessor models of the 3B20 computer, unless the program contains system dependent code (e.g., ps(1)).

The talk will cover the critical region solution that was used (semaphores), dead-locks and resource starvations, scheduling, the protection scheme used for device drivers.

Comment

The abstract says it all. The current system is a dual 3B20 system and runs 1.6-1.9 times a single processor system. The speed improvement is typically 1.7 with a mixed workload.

☞ Lunch ☜

2.31pm Interactive three dimensional molecular graphics under UNIX
C.Huang, Computer Graphics Laboratory, University of California

Abstract

The Computer Graphics Laboratory at UCSF was established in 1976 for research on the structures and interactions of proteins, DNA, drugs and other molecules of importance in biomedicine.

MIDAS (Molecular Interactive Display and Simulation) is a large interactive molecular modelling graphics package developed at UCSF under UNIX, originally on a PDP-11/70 with an Evans and Sutherland Picture System 2 in black and white. It now runs on a VAX-11/750, and provides a flexible tool for the study of small and large molecules and their interactions, taking full advantage of available interactive three dimensional color display capabilities on both the Evans and Sutherland Picture System 2 and Multi Picture System and eventually on a PS300. Bond rotation, interactive monitoring of several distances and 'docking' with real time representations of molecular surfaces is well supported.

Among its more innovative features is an unusually coherent hierarchical database for storage of macromolecules which minimizes both storage space requirements and access time. The 'tool building' philosophy encouraged by UNIX has resulted in a well organized and maintainable program that is well suited to reimplementation on UNIX-based graphical workstations such as the Silicon Graphics IRIS. The lecture will be illustrated with color slides and a movie. Supported by US National Institutes of Health research grant RR1081.

Comment

I was particularly impressed by the real-time nature of the pictures which could be generated.

3.08pm Does Darth Vader still program in Fortran? (or what do they really do at Lucasfilm)
     Sam Leffler, Lucasfilm

Abstract

The original question posed by Bill Reeves at the USENIX meeting held in Boston, Massachusetts was 'Does Darth Vader program in C?'; as one can see by the title the answer was clearly 'No, he programs in Fortran.' This presentation will attempt to update the audience on this noteworthy subject, as well as items of similar importance. High quality audio-visual material will be prominent in the presentation in an attempt to distract the audience from noticing that this talk is almost completely content-free.

For those that don't believe the previous paragraph, the presentation will concentrate on describing what Lucasfilm does, how they use computer technology, and, most importantly, how UNIX is an integral part of everyday work, from the mundane to the exotic. Slides and 16mm film will be shown of the most recent work from the computer graphics project.

Comment

Sam's answer to the question:

Does Darth Vader still program in Fortran?

was

No, the Ewoks do it for him.

Lucasfilm is split into several sections: Administration; Industrial Light and Magic, which is responsible for special effects; Skywalker Development Co., responsible for building a 'ranch' which will be used for filming; Sprocket Systems which does post production work on film; and the Computer Research and Development, where Sam works.

Sam talked about the computer based system which is used to fuse together several images in a process called 'blue-screen matting'. This is where separate films of different objects are taken on a blue background and the images are joined together eliminating the blue. Previous systems used a special machine where the original images are projected and a new negative made. Lucasfilm now have a system where the images are joined using a computer system.

Another area of interest at Lucasfilm is the use of computers to synthesise images. Sam showed a number of astounding slides, the best of which (or at least the one which sticks in my mind) was a picture called '1984'. This showed four pool balls on a table being struck by a cue ball. The pool balls had the obvious numbers. The four balls had all moved and the picture showed their motion blur. Sam pointed out that the high-lights in the balls (pin-points of light) contained the reflections of the room around the pool table. The picture was really great, and very hard to describe.

Sam talked about 'Andre and Wally Bee', a 2-minute animated film generated totally by computer. He was going to show the film, but the projector broke down, so we had an early tea. The film was shown when the projector bulb had been replaced, it has everything: three dimensionality, motion blur, forests generated automatically, to name just a few features.† Great stuff.

☞ Tea followed by the film ☜

At this point, I decided to visit the exhibition and skip the remaining talks for the afternoon. Everyone at the conference had to make the decision about what talks to miss so they could get to the exhibition. The program organisers had thought that two hours would be sufficient time for most people to go in the lunch break. However, they hadn't reckoned on the eons which it took to have lunch in St. Catherines.

---

† And you didn't need to worry about —v, either.

So, I left and made my way along the road to the exhibition hall. On my way past the RAC sign saying UNIX SYSTEMS 84, I began to feel that EUUG had come a long way since those 20-odd people met in Scotland those several years ago. Still.....

The exhibition was large, at least by previous standards. Most exhibitors were in blue booths, obviously supplied by the organisers, Network Events. It felt very professional, and that was no bad thing. Certainly, the exhibitors who I talked to felt that it was OK, and since there were 2500 visitors, it must be adjudged a success.

One of the things which caught my eye was the SUN system with the wide screen. This was fun to play with and is the way I want to talk to computers.

At the low end of the market, Torch's Unicorn sitting behind a BBC computer is still the cheapest UNIX system you can buy. However, it really is VERY SLOW. The salesman told me that Torch is bringing out a system with a M68000 running at twice the speed of the current version. So, if you're thinking of buying a Unicorn, wait. Of course, the other problem is that the system is UNIX System III 'with the usual Berkeley enhancements'. The visible bell in **vi** was fun on the colour screen, giving bands of different colours.

Possibly, the smallest machine, in physical terms, was the Spirit from UNIQIX Ltd. This looked an interesting product.

Anyway, I can't do real justice to all the exhibitors, so I'll stop. Let's hope that most of them come to Paris, so that perhaps there will be more time to look at them.

Meanwhile, back in the Chemistry lecture theatre:

4.10pm UNIX user interfaces for applications
Stephen Travis Pope, BST - Basic Software Technology Dept.

Abstract

All computer applications need some man-machine interface method. UNIX is especially well suited as a base for applications because programmers have access to system utilities and can substitute their own front-ends for or on top of the 'shell' program. Developing new user interfaces that are tailored to particular applications requires, however, that the designer repose several basic questions about the desired application, its users and computer I/O in general.

Computer configurations involving bitmap terminals and mice can also be used for very-high-level interfaces and programs can be built to take full advantage of these features. So-called 'window systems' as seen on the Xerox Alto, Lisp machines and the Apple Lisa computers can also be implemented in multi-user unix environments with excellent results.

Several current projects being undertaken at the BST dept. of PCS GmbH will serve to demonstrate special, window and/or menu based user interfaces for applications in program development, databank management systems, networking and computer music. Topics of this part of the discussion also include the hardware for the current PCS multi-processor implementation of the **wsh** window shell.

4.35pm Winnie: a new multiple window screen editor
Patrick Amar, United Software Artists Inc.

Abstract

We shall describe a new full screen editor called Winnie. Winnie belongs to the Emacs family. It has two major enhancements over Emacs: a more flexible windowing discipline and a multi-language extension facility. As opposed to Emacs, we have strived to build a small and efficient implementation. Winnie uses no more than 35 Kbytes of storage for code (compared to 121 Kbytes for Emacs on VAX) and loads the computer much less than Emacs.

Winnie can handle arbitrarily many windows of any size and placed anywhere on the screen. Windows can hide each other as sheets of paper on a table, or as screens in the Xerox Star station or the Apple Lisa machine. For the present time we handle only alphanumeric screens, but a graphic screen version is forthcoming.

5.05pm   Questions and answers
           Vanni Papi, UNIX Europe

As I had left, I asked Sean Leviseur from UKC to write some notes. So, this bit is from him.

Q.   When will virtual memory support be available for System V?

A.   Virtual memory will be available in the last quarter of 84 for System V.2.

Q.   What will be in System V.3?

A.   Chiefly file locking and support for virtual memory and paging.

Q.   Will Edition 8 be generally released?

A.   No.

Q.   How will the presence of UNIX Europe affect pricing?

A.   European prices will be the same as in the States.

Q.   What price will the upgrade to System V.3 be?

A.   This has not yet been decided.

Q.   Will Edition 8 be released to educational institutions?

A.   It will only be released under strict supervision to six American universities.

Q.   When will the UNIX manuals be stabilised?

A.   With the next release. They are currently being restructured, only one set is currently available. It would be useful if people could comment on the current System V manuals.

Q.   Will System V be unbundled?

A.   It is intended to unbundle more of UNIX, but not until next year.

Q.   Will the information flow from the States improve with the existence of UNIX Europe?

A.   We will have more people from the States, so hopefully we should get information quicker.

Q.   What about European availability of the Blit and its software?

A.   The Blit is sold by Olivetti in Europe, although it is still available from AT&T's old distributors in Europe. The software is available from UNIX Europe, as is the source code.

☞ End of Day 1 ☜

And some of us went onto AT&T's hospitality suite to obtain the odd alcoholic beverage; and then onto the Reception for all conference attendees in the University Combination room, where even more of the you-know-what was consumed.

Day 2 - 20th September

The lucky umbrella didn't work this morning, the heavens opened and tried to wash Cambridge into the River Cam.

9.33am MMUs & the UNIX kernel
Robert Jung, Root Computer Ltd.

Abstract

UNIX system performance depends on many factors, one of which is the memory management unit (or MMU). Porting the UNIX kernel to Motorola 68000 and 68010 based systems has given Root a unique insight into the implementation and performance of MMU designs. This talk describes the kernel's interaction with the MMU, observations and recommendations of MMU design and implementation, and a brief discussion of the memory-management schemes Root has come across.

Comment

Root has done many ports of the UNISOFT UNIX system. They have also used a number of MMU's but the talk centred on comparing the Stanford MMU with the Motorola 68451 MMU. The Stanford MMU is also called the SUN MMU, and has nothing to do with the company of the same name.

The Standford MMU is better for memory allocation, switching processes and accessing the user area of the current process. But it is worse at accessing other processes, which is used for swapping. The Stanford MMU is cheaper and runs faster than the M68451.

10.01am Paging in the UNIX system
Steven J. Buroff, AT&T Bell Laboratories

Abstract

Two research derivatives of the UNIX system have supported paging for several years: Reiser 32V, and BSD. Work is under way at AT&T Bell Labs to bring together the features of both of the systems [and others] to form a demand paged kernel for UNIX System V. This talk will discuss three areas of this work: requirements, architecture, and implementation.

Comment

The requirements are: there should be no user program changes for either binary or source; the system must not hurt users who don't require paging, this means that if you want to use a paging system because it is faster, then you can do so; and the system should provide the capability of large address spaces if that is wanted.

The idea was to generate a general model of memory management in the UNIX kernel and to abstract all the code which deals with it into one generalised set of routines. Different memory allocation methods can then be used because a clean internal interface has been designed. The main primitive in the design is a *Region*, which is an area of memory. It can be shared or private and is manipulated by a set of well defined operations. These operations are: create, delete, attach, detach, grow, load and copy. The system allows copy on write by adroit use of the page descriptors.

The current implementation is for a paging kernel which appears to work. There didn't seem to be a swapping implementation as yet.

10.28am **Productizing (zic) UNIX**
         Armando Stettner, Digital Equipment Corp.

Abstract

DEC is now offering a UNIX product. This talk will discuss some of the problems that were encountered when creating that product.

Comment

At the time of the decision to supply and support UNIX, it was decided to base the system on 4.1BSD. The system was fast and flexible and had support for many peripherals, so there would be much less work for DEC to do in preparing it to be a product. Also, at that time more VAX's ran 4.1BSD than any other system.† The intention was to switch to 4.2BSD when it came out.

The goals of the product were that it should be as least as reliable and predictable as 4.1/4.2BSD. Also, it should be compatible with those systems. DEC didn't want to introduce yet another flavour of UNIX. DEC have not altered anything in the program execution environment and have only altered one user program (**tar** now follows symbolic links).

There were several questions:

Q. Can you install user written device drivers?

A. Yes, the system is supplied in a configurable binary with important files being supplied in 'un-compiled' form.

Q. Does ULTRIX have the device drivers which control non-DEC devices.

A. Yes, anything on the 4.2 distribution is also on ULTRIX. But, it will obviously be harder for DEC software support to provide advice and bug fixes on the 'foreign' peripherals.

Q. What does the management think about that?

A. ULTRIX must fulfill the expectation of what UNIX does. So, the foreign device drivers must be supplied because they are part of UNIX.

Q. Can ULTRIX support the new DEC device clusters?

A. Not at present.

Q. Does DEC intend to move to System V?

A. No.

☞ Coffee ☜

11.17am **A secure high-speed transaction protocol**
         Sape J. Mullender, Centrum voor Wiskunde & Informatica, Amsterdam

Abstract

Most computer networks use a byte stream protocol for communication between processes, which suffer from two important drawbacks: the addressing mechanisms provided often are process-dependent or location-dependent, and communication is slow. While carrying out research into distributed operating systems at the Vrije Universiteit and the Centre for Mathematics and Computer Science, we have developed a transaction-oriented transport protocol, aimed for high-speed, with an addressing mechanism that is not only more general, but provides a protection mechanism as well. The basic mechanism for communicating between processes is the transaction: a client process sends a request to a server process, which carries out the request and returns a reply. Protection is

---

† At some point in the conference, I was told by a DEC person that 25% of VAXes in the UK are running UNIX.

provided by using ports, chosen from a sparse address space, for addressing services. These ports serve as a 'capability' for communicating with the service. Through its simplicity, the transaction protocol can achieve high transmission rates (more than 200 Kbytes/sec process-to-process, eventually).

The protection mechanism will be described, and the mechanisms for realising high transmission speeds.

11.48am Connecting UNIX systems using a token ring
        Robbert van Renesse, Vrije Universiteit, Amsterdam

Abstract

As part of the research on distributed operating systems being done at the Vrije Universiteit, we have implemented a set of network-oriented programs for use on several UNIX machines connected by a high-speed token ring. With these tools it is possible to transfer files between machines, log in to remote machines, and implement multimachine shell scripts. The transaction protocols discussed in another paper at this EUUG meeting are used to implement two basic services: a 'shell server' and a data transfer service. Other services are easily implemented as shell scripts that use these services. A file transfer program, for instance, executes the command **to < file1** on one machine, and **from > file2** on the other machine. More examples of these facilities and their implementation and performance are discussed in the paper.

Comment

Some real process-to-process throughput figures were mentioned: VAX/VAX - 25Kbytes/second and PDP-11/PDP-11 - 10Kbytes/second.

12.08am A project development environment for UNIX
        Malcolm Crowe, STRG Paisley College of Technology

Abstract

A software environment for project development should include tools for all phases of the development process. Many such tools already exist, or are under development, in the UNIX system.

However, apart from archiving systems such as SCCS, little support exists in UNIX for quality and project management. A key activity of quality management is concerned with the identification and control of all items produced during software development.

Until recently configuration management has been an essentially manual activity, possibly performed by a member of the project team. In this paper, we describe facilities which allow project management to implement a configuration management plan on a per-project basis.

The basis of the system is an enhancement to the UNIX file system (not affecting the kernel), which does not alter the user interface for naive users. The resulting environment retains all the UNIX tools, while allowing for their use to be restricted in various ways on 'controlled' objects.

The system is available to other participants in the Alvey Programme.

Comment

This was an interesting idea which uses the C library to alter the nature of the file system. Files can now have: long file names (even on V7); a project defined set of attributes; multiple versions; and controlled access. The file system also supports the notion of projects and project hierarchies. The file attributes are defined by name and act rather like shell environment variables.

Jim McKie who was chairing this session got some biographies, official and unofficial, which he used to introduce the next three speakers.

Mike Karels

Official: University of Notre Dame (Indiana), B.S Micro-biology 1978; worked in the Molecular Biology Department of the University of California, Berkeley 1978-1983, doing bacterial genetics and kernel hacking in UNIX V7 - 2.9BSD; in August 1983, joined the Computer Systems Research Group.

Unofficial: In 1982, the Paris EUUG conference was blessed with the attendence of Bill Joy. His performance at that conference gave us the phrase 'Paris mode'. Sam Leffler was brought over to the Bonn conference to apologise, and did such a good job that he also attended the Dublin conference. Due to his sterling performance in Eire, we required two UCB people, Eric Allman and Kirk McKusick to compensate at the next conference. This conference is please to have Mike Karels and we have yet to decide who will apologise for him.

Tom Killian

Official and unofficial: Tom Killian began his career as a high-energy experimental physicist but was unable to convince his collegues of the value of Computer Science. Since 1983, he has been with the Computer Science Research Center at Bell Laboratories, where he has successfully dealt with painful childhood memories of MVS and SCOPE.

Greg Chesson

Jazz drummer: CC Riders 1967-9, Woody Herman Orchestra 1969-70; B.S Math, Union College, New York 1972; M.S. Computer Science, University of Illinois 1975; Ph.D. Computer Science, University of Illinois 1977; Member of the technical staff at Bell Labs, 1977-1983; Chief scientist at Silicon Graphics 1983 onwards. At Bell Labs, Greg was responsible for V7 multiplexed files; line disciplines, character drivers and boot programs in V7; packet drivers; circuit simulation, PLA, board layout software in UCPS; design and simulation of Datakit protocols; and the design and implementation of the B-machine 10mips processor for Bell. At Silicon Graphics, Greg has done the XNS network software for 4.2BSD, System V and VMS.

Unofficial: Greg Chesson, known as Bambi to his friends, has made contributions to the UNIX world, none of which should be mentioned prior to his presentation to ensure a reasonable reception. He is well known for his ability to find tone-deaf band leaders who let him exercise his other talents which are normally executed with the tact and diplomacy for which he is well known. His hardware skills are second to few. It is said that there is yet to be a machine which Steve Johnson couldn't overload or Greg couldn't break. Greg enjoys sports that don't involve standing; and the only skill about which he shows any modesty is his extraordinary imitation of a synchronised swimmer using Datakit.

2.36pm **Life after 4.2: measuring and improving the performance of 4.2BSD**
Mike Karels, CSRG, CSD, EECS, University of California, Berkeley

Abstract

The 4.2 Berkeley Software Distribution of UNIX for the VAX includes a number of new features and facilities that substantially increase its utility. However, it has several problems that can severely affect the overall performance of the system. These problems were identified with kernel profiling and system tracing during day to day use. Once potential problem areas had been identified benchmark programs were devised to highlight the bottlenecks. These benchmarks verified that the

problems existed and provided a metric against which to validate proposed solutions. This paper examines the performance problems encountered and describes modifications that have been made to the system since the initial distribution. It also describes other work underway or planned at Berkeley.

Comment

Mike started with some general observations on 4.2. First of all, it seemed slower than 4.1, and the system throughput was down by about 20%. The many new servers added considerable system overhead. The new fast file system altered the workload characteristics, so that processes which were previously disc bound were now processor bound.

The system was measured in order to get some idea of what was happening. The results showed that the micro operations were about the same speed as 4.1, although **pipe** and **exec** were a bit slower. The measurements also showed that name translation was about 40% of the system call overhead. Symbolic links add a measurable overhead.

There is some mileage in improving some user level programs. For instance, programs accessing the password file can be made to go faster. The standard I/O library has been altered to use optimal buffering.

In the kernel, the name lookup routine has been made 60% faster by use of caching. This results in an 8% improvement in the total system time. The dz and dh drivers use the silo for slow transfers, which loads the system. The code has been altered to use a single interrupt for the intermittent transfers which constitute most terminal input. The clock interrupt routine has been made to work faster. The arguments to the **exec** system call used to be read into the kernel a character at a time, the new system uses the much faster block copy code to read in strings. The context switch code has been made to run faster. Pipe performance has been improved by supplying more buffering. There have been several other minor alterations.

The system will be available to the public before December 1995.

New things which are being worked on a UCB include: a network file system which connects systems using a single tree using a remote mount system call; there is work being out into protocol layers; and a reliable remote procedure call mechanism.

At the end of his talk, Mike made a presentation of a game of 'Battlecars' to Armando Stettner. Apparently, Armando is famed for his accident prone driving and has received many car (or perhaps I should say automobile) related presents at US USENIX conferences.

3.00pm **Processes as files**
     Tom Killian, AT&T Bell Laboratories

Abstract

We describe a new file system, /proc, each member of which, /proc/nnnnn, corresponds to the address space of the running process whose pid is nnnnn. Access to these files is restricted, via the normal file protection mechanism, to the process owner. Lseek(2), read(2), and write(2), allow inspection and modification of the process' image. Other services are available via ioctl(2), including stop/go on demand, selective interception of signals, and the ability to obtain an open file descriptor for the process' text file. The technical problems related to the implementation of /proc on a VAX under the 8th Edition of the Unix operating system have mostly to do with the paging system. Security issues are also considered.

The window-based interactive debugger **pi**, developed by T. A. Cargill, is the first major user of /proc. It can control multiple processes dynamically and asynchronously. Thanks to the network file system, /n, these processes may be running on several different machines. We also

describe an efficient, almost portable **ps**(1).

Comment

This is such a reasonable addition to the file system name space, it took a genius to think of it. **Pi** is based on the Blit terminal.

3.31pm   Multicast ring protocols for real time games and other useful pursuits
Greg Chesson, Silicon Graphics Inc

Abstract

Many great advances in computer science have been motivated by things that some (though not all) would deem as unimportant. This talk is about the solution to the less than pressing problem of being able to 'fly' multiple flight simulators in formation. Only the future will tell if this solution is a great advance, but it is brought to you by the implementor of multiplexed files and this conference's shoo-in for the hairy knees contest.

A demonstration (on video tape) will be shown.

Comment

This is fairly verbatim from Greg's initial slides:

WHAT   do we want to do?
             To use a Ethernet as a token ring.

WHY   do we want to do it?
             To synchronise real time processes.

MOTIVATION to transform a real-time flight simulator on UNIX into a dogfight program for several machines running several programs controlled by several 'pilots'.

Features of the flight simulator include the ability to choose one of several aircraft - Cessna 180, 747, F15, F18 and F16. Weapons on the aircraft include Sidewinder, rocket and cannon. Pilots of the Cessna spent most of their time circling the airfield and picking off other people as they take off. The output is in 3 colours with a 1K resolution screen, input is by mouse - no joystick yet.

The problem is how to get one program running on one machine update all the other programs on the position of their aircraft. The mechanism used is to broadcast datagrams rather than having a 'star' network of virtual circuits.

This talk was certainly one of the highlights of the conference. The video tape spend most of the conference winging its way in a non-simulated aircraft across the Atlantic. It reached Cambridge by Friday evening, and was worth waiting for.

☞ Tea ☜

4.31pm   UNIX IPC: where it's been, why it left, where it might be going
Mike O'Dell, Group L Corporation

Abstract

Few topics in the Unix community have provoked as much discussion and as many implementations as the issue of providing 'good' interprocess communication (IPC) for UNIX. The problem, of course, is not with pipes, for everyone agrees they are wonderful. The problem of interest is a scheme whereby unrelated processes can rendezvous and communicate. Indeed, it is a safe guess that any IPC scheme which has appeared in print has been implemented at least once in some UNIX system somewhere.

If there have been many implementations of IPC for UNIX, why is it that none of them ever seem to catch hold and be adopted widely as a general mechanism? Why indeed!

This talk will attempt to provide some historical background by reviewing some of the more influential IPC implementations, and discuss the author's view of the question posed above. Finally, the author will propose a model for unifying two important stream IPC facilities, and make some speculations as to how this unification might point the way toward a 'natural embedding' of IPC in UNIX.

Comment

This was one of the best talks of the conference. I feel that I cannot do any justice to it from my notes - you will all have to wait until the proceedings are published. Suffice it to say that the talk gave a really comprehensive review of the many different schemes for IPC with an idea of the problems and advantages of eac⁻.

☞ End of Day 2 ☜

Well, via AT&T's hospitality room to the conference dinner, which was a pretty splendid affair. After the dinner, The Instruction Set did a side splitting selection of sketches. The EUUG committee were asked to sit in the front, we were bearing up for custard pies or something of that ilk, but nothing really nasty happened - we just laughed a lot. I also met Mr, Mrs and Miss Tis who were very nice.

Day 3 - 21th September

9.33am **Implementation of OSI protocols under UNIX in the EIES network**
J. Loveluck, Bull

Abstract

The Esprit Information Exchange System (EIES) is an infrastructure to support collaborative R and D projects in information technology within the European Study Program in Information Technology (ESPRIT) launched in 1984 by the European Economic Commission. The work is being carried out by a consortium of 6 industrial partners.

The EIES will provide the electronic mail, teleconferencing, document handling and transfer, file transfer, remote login services which are necessary for cooperative R and D work.

The project aims at a maximum connectivity of potential users through the use of Open Systems Interconnection ISO services and protocols, and starts with an implementation under UNIX.

After a short description of the objectives of the project, the paper describes the detailed architectural choices made for the interconnection of local area networks and wide area networks; the addressing scheme is discussed, and some considerations given to the management aspects of the network. Finally the main choices made for the implementation under UNIX are described.

10.12am **The Instructional workbench: a CAI system and more**
Thomas B. Reddington, AT&T Bell Laboratories

Abstract

Any computer-assisted instruction (CAI) system must allow an author to easily create courses for producing on-line dialogue with a student. Instructional Workbench (IWB), the CAI system implemented for the UNIX operating system, is particularly efficient for building a friendly human-machine interface and, therefore is a strong alternative to other CAI systems and programming languages.

Two important features of IWB that will be discussed are the design of the authoring language and the authoring system that enables novices to 'mass-produce' on-line dialogue programs (courses) without requiring a detailed knowledge of the authoring language.

A production system served as the model for the authoring language of IWB, called TOPIC language. This design has proved to be particularly useful for defining the logic inherent in complex human-machine interactions and for easing the work required to modify that logic. Features which are 'built-in' to the TOPIC language are:

- I/O from files

- keystroke verification of user input using regular expressions

- terminal independent screen management capability.

The design of the TOPIC language has been so successful that parts of IWB that interact directly with a user are written in the TOPIC language.

The design of the IWB authoring system was driven by the needs of the intended authors: subject matter experts rather than programmers. The authoring system is template-based and allows authors to produce on-line dialogue programs quickly by filling in the 'holes' of the templates. The templates, which are data driven programs coded in the TOPIC language, allow an author to build dialogues by specifying the data particular to an interaction. In CAI common types of templates are true/false questions, multiple choice questions, and tests. In addition the authoring system can be extended by the addition of templates written by a programmer with some familiarity with the IWB TOPIC language. This robustness has enabled the authoring system to be useful in areas outside of CAI such as on-line help systems.

The presentation will also discuss other features, such as computer-managed instruction, that make IWB more than just a CAI system.

Comment

This is brand new and is not yet a product. I think that it looks very good.

10.34am What is happening at Pyramid
Robert Reglan-Kelly, Pyramid

I am afraid I missed this talk.

☞ Coffee ☜

11.22am UNIX in Australia, 1984
John Lions, University of New South Wales

John started by talking about the history of UNIX in Australia. He then spoke of some of the work which Tim Long has done in making the file system faster. This involves keeping the inode list of free disc blocks in ascending order; varying the look-ahead from between 1 and 32 blocks depending on experience; and finally coalescing requests for I/O into one large request. John then spoke about the scheduling system which Sydney University uses to get 100 students working on a VAX. The mechanism is based on the user having a share of the resources in the machine; the system is called MUSH and is where UCB got their disc quota system from.

The Australian alternative to the UUCP network is called ACSNET. This was originally called the Sydney Unix Network, or SUN; but unfortunately this name has been stolen by another company in the UNIX world. The name is still retained for the software. ACSNET is a message passing service involving multistage transfers but with no explicit routing. We are promised more on the system when Piers Lauder comes to Paris.

John noted that there was a general movement from DEC hardware in Australia; he mentioned that Melbourne was very happy with the Pyramid.

John finished which a few slides: one was of a wallaby reading the Kernighan/Pike Book - and very fetching it was too.

11.55am UKUUCP and other EUUG software
Lee McLoughlin, Westfield College, University of London

I have the benefit of Lee's overhead projector slides for this - Ta.

UKUUCP is a single UUCP combining the best features of the dozen or so UUCP versions found in the UK. It includes work done by Mike Bayliss, Tony Luck, Jeff Smith, and Lee himself. UKUUCP includes hooks for dial-in/call-out on the same line, on V7 and 4.2BSD and has loads of bug fixes. Its performance is enhanced by 40-fold by improving the scanning of queues.

The code should port easily to other systems. It is currently running on VAXes, PDP 11s, LSI 11/23, various 68000s and the High Level Hardware Orion. The code is running in the following UNIX systems: V7, 4.1/4.2BSD, System III and System V.2. The code has failed on GEC 63's and Perqs.

The main claim to fame is that it can transfer over very simple connections, such as PADs, GECs, PRIMEs, and other systems which provide only tty facilities for networks.

UKUUCP needs York Box support (this is underway); X.25 support as in System III/V; and many other things.

The code is available on the EUUG tape. You require a V7 license (or better).

12.11 EUUG business meeting
Emrys Jones, EUUG

The main business was to report that there had been one written submission about the proposed constitution with syntactic alterations. The constitution will be sent to all members for a postal ballot in the near future.

Actually, there is also one matter of report which Emrys did not mention but which I thought I would slip in here. At the Committee meeting in Nijmegen, it was decided to award an honorary membership to Alan Mason in recognition of his work in laying the foundations of the EUUG.

2.30 Speech input and UNIX
R.M. Johnstone, University of Glasgow

Abstract

This work investigates the performance of speech input in certain application areas.

The preliminary work involved selecting a task which seems particularly well-suited to use of speech input, and building a speech system around the task (using a UNIX host), following principles derived from previous work in the area). Low-level recogniser functions are controlled by a set of simple C programs. Vocabularies are stored as UNIX text files, for re-use in the next speech session. The facilities of **emacs/mlisp** were of considerable use in implementing and maintaining dialogues for particular tasks.

Initial experiments involved the experimental manipulation of task variables (e.g. operator experience, system training procedure, vocabulary composition), and certain recogniser parameters. This type of work aims to produce guidelines for applying speech recognition, which will enable a system to be optimally tuned.

Our current project looks at how well speech recognition can replace keyboard input. Once again, our answer will depend on a large number of variables, including user cooperativeness and experience, and characteristics of the particular task. Candidate tasks include Pascal programming and **nroff** command insertion.

2.57pm Measuring disc I/O on the VAX
Nick Nei, University of Glasgow

Abstract

This paper describes a project under way at Glasgow University to gather statistics about disk performance on a VAX running Berkeley UNIX 4.1. These results will be used to construct a stochastic model for the behaviour of the disk subsystem. We hope that by modifying the parameters on the model and studying the results we can discover new ways of improving the disk and file system performance.

Comment

Nick was rushed when he talked about this at Nijmegen and was given more time to present the results again. The main idea was to measure disc performance in order to find useful measures of spread and central tendency of disc traffic, i.e. the mean arrival of requests for disc transfers and the pattern of requests. Armed with the mathematical model, it should be possible to predict future trends and generate reconfiguration forecasts.

The measurements taken were at the start of the *strategy* routine, which is the point where UNIX says 'OK here's something to do on the disc' to the relevant disc driver.

The measurements showed a generally exponential distribution but there are two interesting peaks: one at a few micro seconds and a one at 20ms. Why these peaks exist is still being investigated.

3.20pm   sndawk - a signal processing language
Dan Timis, Institut de Recherche et Coordination Acoustique/Musique, Paris

Abstract

Signal processing and sound synthesis often use pipelines of programs performing specific treatments as filtering, changing the sampling rate or the gain etc. on binary floating point samples. Users who want to make their own algorithm will spend sometimes, for a very simple thing, time and energy to write, to compile and to test a program in C or Fortran.

Sndawk (sound awk) provides an interpretative signal processing programming language simple to learn and to use. Inspired from the well known pattern scanning and processing language **awk**, it respects much of its syntax as it respects much of the syntax of C.

This talk will discuss the structure and use of **sndawk**.

☞ Tea ☜

4.30pm   Pontifications, Accusations, Prognostications & Mystifications
Chaired by David Tilbrook, Imperial Software Technology

Abstract

This session will be an open forum for discussion of UNIX and its future. Each panelist will make a 3 minute presentation on their views and prejudices, after which the floor will be open for questions, comments and discussion.

Comment

This session is impossible to take notes on and luckily Richard Stibbs had organised a tape recorder to record all the important noises with. I had not thought of that and will certainly do it again, — thanks for the idea, Richard.

The session started with a couple of small presentations. The first was introduced by David Tilbrook.

"I don't want to embarrass people by asking: who has an illegal copy of John Lions' book?† But when I put out the advanced notice saying *The much-xeroxed John Lions*, I got an enthusiastic response. It was a remarkable book, we owe him a lot, and we would like to give him a small token of our appreciation in lieu of royalties." The present was a pint beer mug bearing the arms of John's college, engraved with *To John Lions, in appreciation from the EUUG, Cambridge, September 1984*.

David was then presented with a large pair of big red inflatable lips by Jim McKie, who said: "There is someone here who has done a lot of work for this conference and it's often been said that he has the biggest mouth in the UNIX world - so here's the biggest lips to go with it."

John Lions produced a book, or rather two books, describing the workings of UNIX V6. They were aimed at teaching undergraduates about the internals of the operating system but ended up training nearly all the UNIX hackers who existed at that time. AT&T sat on the book, only allowing one copy per site. To prevent any repetition of the incident, licencees of V7 were prevented from teaching the operation of the kernel.

The proceedings then really started with each member of the panel making a three minute presentation.

Armando Stettner - DEC.

"From its beginning, UNIX could never be considered a 'standard' operating system, or rather it has always been considered to be evolving. Once UNIX allocated its first inode it started to evolve. Its evolution was in the hands of a very few people; beginning with a small group in Bell Labs, Murray Hill and moving onto small groups in Universities around the world. There were other groups inside Bell Labs who were interested in turning UNIX into a tool for research and development of applications. Berkeley got into it, and did the right kind of things with 3BSD and 4BSD. Or at least in the early days between 3BSD and 4.1, they did the kind of things which were needed.

Now we're into new era, UNIX in a larger sense will no longer be driven, or certainly no longer controlled, by purely technical and research people. Thankfully, I believe, 4BSD will continue to evolve along the path that makes sense.

Perhaps unfortunately, UNIX evolution is now in the hands of the corporations, IBM, Perkin-Elmer, AT&T, (*DEC? from the audience*), DEC, yes. These people will be driven by their perceptions of their customer's perceptions of what they think they need. I can only hope that these new features and capabilities, and the functionality that the companies implement will fit into the frame-work and architecture of UNIX. I feel that this is part of the job of every UNIX guru and wizard who work for these companies that sells or distributes UNIX. Hopefully, the UNIX architecture will also evolve to facilitate its use on new technologies and new ways of building systems.

With the new kids on the block, the corporations, I don't know where UNIX family is going. I can only hope that it will evolve and those of us who are involved in its evolution will keep an open mind for new things and new values."

John Lions, University of Sidney

John started his talk with a visual demonstration of C.P. Snow's law: *every culture is composed of two subcultures*. This doesn't translate easily onto the printed page. He wanted to talk about the small group of people

"..... who can carry forward the true UNIX tradition. Which is not only to take two steps forward all the time. But occasionally to take one step backwards; and not only to add a few new features to the system; but also to remove some redundant, over-grown, over-ripe features from time to time. I think that if we all took a vow to declare 1985 the Year of the Pruning Saw and spend a lot of time cutting back, removing things which aren't really needed; then UNIX may thrive. Without this, it will grow and nobody can do what Ken Thompson and Dennis Richie did for so long, namely, cut things back. If the unimpeded growth is allowed to continue, then the UNIX tree will die in the foreseeable future."

Tom Killian, AT&T

"It says on my badge that I am from AT&T but I would like to issue a disclaimer: *The opinions expressed are those of the speaker and do not necessarily reflect those of AT&T, its lawyers, cupholders, inquisitors, or anybody else for that matter.*

Some of the questions brought up recently at this conference have been to do with the question 'What is UNIX?', and I propose a number of possible answers to this.

● A trademark of AT&T Bell Laboratories.

● It ought to be a trademark of Brian Kernighan, since he came up with the name.

● Is it a kernel?

● Is it a set of standard utilities?

● People watching this conference from the outside are probably convinced that it is a cult sub-culture.

- It's almost certainly a state of mind.
- In some sense, it's also what runs on Dennis Ritchie's machine.

I think the UNIX philosophy is summed up in something which was written a long time ago by William of Occam, I am permitted to quote it in Latin, since this is Cambridge (*he then quoted it in English*†): 'Objects should not be multiplied unnecessarily'. This is something which has guided UNIX from the beginning. There were a very large number of decisions which were made early on by Ken Thompson and other people in his office writing things on the blackboard. By and large, the decisions were right and it's very dangerous if you go in and mess with them. Such things as the file system and the simple kind of scheduling were very crucial to making UNIX as successful as it was on the machines which were available at that time. Obviously, some these things are going to have to evolve for UNIX to remain viable. But I think that the philosophy that you have a minimal spanning set of necessary operations is very important. If we lose sight of this UNIX will be cooked."

Mike Karels, UCB

"First of all, I will try to describe what I see of where the system is going and how we are trying to get there. Then I shall pick up on what the previous speakers have said.

The model of the computing environment that we were getting looking at in Berkeley 4.1 and which formed the major driving factor in designing 4.2, was that the single machine timesharing system was probably not going to be the way of the future. We are looking at workstations connected by a network, most probably ARPANET, and hopefully to the rest of the world on long-haul nets. So there are a lot of communications facilities and networking in 4.2. To parallel that, there are interprocess communication facilities. (*Tape garbled*) The new directions in 4.2 come from that. Although there are communications facilities, there aren't any really distributed facilities in existence. I don't think UNIX ever will be a distributed operating system, this is tempting nature. On the other hand, UNIX can provide facilities of a distributed nature. This is something which is just getting going.

Another strong feeling that I have about UNIX is that 4.2 has gotten to the point of being a very large complicated beast, not only in terms of user facilities but also in terms of the way things are done inside the kernel. One of my long goals has been to redesign the kernel unifying things. Most of this would be invisible from the outside, but its one of the things I'de love to spend a few years on."

Teus Hagen, CWI

"Some time ago, we had languages, and we started to say that we must standardise those languages so that we could port programs between systems. So, Fortran was a big effort and after a while the Americans discovered that structured languages were a good idea, and so we had C. C has become a little standardised now.

UNIX helped a lot in standardising operations on the machine. I don't think that a lot of people involved in 'standardising operations' can make something which is portable.

4.2 was a lot of work, mainly in the area of networking and I think that's one of the first problems. The networking implementation is very young, it's one of the first so it has to be changed. The resulting system will not be UNIX any more. We have to wait until somebody pops up doing real networking. So, I think that in a little bit, UNIX can withdraw; and in the end we will have a real networked system. But I don't know what that will be."

Steve Bourne, DEC

"At Silicon Graphics, I was considering how to make our business decisions safe against the vagaries of all the different technologies and groups. There are lots of different versions of UNIX out there, and if you were a small company like we were, which version do you chose? Well, of course, what you have to look at, is who the big guys are; or at least, who is likely to be leading the charge.

---

† He put it on the overhead projector in Latin and whipped it away before I had a chance to copy it.

What you have to look at is AT&T, IBM, DEC and (Berkeley). I've put Berkeley in parentheses because they are not a large corporation, or at least not yet, but they are clearly a major force in making technological advances in UNIX.

I am not sure what you should conclude from it. I've spent some time at this conference nattering about how AT&T and IBM are facing off; and how AT&T are yet to have a well oiled marketing machine. Well, anyway so that was one question which I was interested in.

The answer, of course, isn't very interesting. We must hedge our bets. What I mean by this is that we try to chose the subset of UNIX which we use so that programs would port. There isn't much help for porting in terms of tools to look at libraries, and to look at porting difficulties; telling you whether your program is going to port from one place to another.

The question is should we accept divergence? UNIX never legislated anything, the group which made it never legislated anything. There are two more questions: can we as a group continue to make the right decisions and secondly, if we can, can the large group continue to make progress?"

General discussion

David then opened the discussion up to the floor. Sorry folks, you aren't going to get a verbatim transcription of that for two reasons: first, I don't have the time. Secondly, the tape isn't exactly hi-fi quality and some of the floor contributions are totally lost in crackles and hiss. Much like conversation on the UK telephone system these days.

Mike O'Dell kicked off the discussion with a longish statement (most of which is lost). His main thesis was that UNIX makes a number of powerful statements about what one might reasonably expect in an environment where programs are developed. It managed to do this by not writing a paper about it but by implementing it. He then moved on to say that people seem to think that putting a C compiler onto a system imparts some of the magic of UNIX. It doesn't, a bad system with a C compiler is just a bad system on which you might be able to program in C.

The discussion then moved onto the marketing and standards in UNIX. There was general assent to the idea that restraining developments by the imposition of 'standards' was a bad thing. UNIX will always alter to fit the environment in which the system is to be run, whether this is done by companies or universities. The main hope is that whatever is done to the system is done well. One voice from the audience did plead for a single standard, even if it is a bad one, he saw that the problem was that there were too many dialects. A number of voices were raised against this view, the argument seems to be between those who think UNIX is an evolutionary process and believe that 'official' standards will inhibit this; and those who want and need a stable base for the development of applications software.

The question was raised as to whether you should run System V or 4.2 on a VAX? Most people in the audience said 4.2 - perhaps that says something about the audience. David then asked whether we should all be trying to run Edition 8 rather than 4.2. John Lions said that if Rob Pike was to be believed: the file system throughput under Edition 8 is considerably greater than under 4.2; the kernel is more compact; and line disciplines are a reasonable alternative to sockets. Cries of 'not true' from Mike Karels.

John Lions then moved the conversation onto the availability of bit-mapped displays. There was a general feeling that the cost would reduce to a level where they were affordable in the same way as a standard VDU is today.

Discussion moved onto how network file systems should be implemented. Tom Killian said that the strength of method where remote files were joined to the local file system as part of the tree was that all the current binaries just work, the *namei* routine in the kernel just works harder. Mike Karels endorsed this.

There was then a lot of discussion, with not many new points. David wrapped up the conference and we finally got to see the video tape of Greg Chesson's flight simulator system.

Endpiece

We did, as usual, have space for writing silly comments on a blackboard. The idea was for people to write down the error message which they found of least use excluding the '?' and 'TMP' messages from **ed**. We got the following list, I leave it as an exercise for the reader to can check whether they are true or not.

local symbol botch — V7 **ld**.

ERK! - (V6 **passwd** & V7), we had a later spelling correction to URK!

oops — **termlib**.

Bailing out at line ... — **awk**, especially when the ... is replaced by 1.

eh? — **chess**.

Values of B will give rise to Dom — V6 **mv**. This is my favourite.

Invalid keyword "else" — C compiler.

Very funny — V6 **pr**.

argc=2 — V6 **comm**.

Clock may be set wrong — SCCS **get**.

mo40 — C compiler.

Intruder alert — 4BSD **whoami**.

Jackpot — V6 **diff**.

Room must be Cory or Evans — 4BSD **chfn**.

Termination code 132 — 4BSD **f77**.

ERROR
Illegal character 104 (octal) — **pcc**.

Lints little mind is fried — V7 at least. I checked this on 4.2, and the correct coding for that system is 'Lints little mind is blown'.

No Toy clock — V6 **date**.

Modify failed — **csh**.

Syntax error near line 1 — **awk** for a 1-line program.

Mere mortals mustn't use that mantra — **sendmail**.

What? — **quiz**.

Well, that's it. This file is now about 70000 bytes, which is too long. Thanks to all who made the conference work.

Comments from the Survey taken at Cambridge

David Tilbrook*

The surveys handed in at the Cambridge conference have been been entered (into a TIPs data base naturally).

The following were the additional comments, with my comments (I just couldn't resist in some cases) in [...].

General comments:

- Preparations!! Programme should show parallel events side by side.

- Programme should go out with booking form so that you can decide which sessions (technical/industry) you wish to book for.

- Why do you have to pay for a catalogue?

  [you didn't sucker]

- How about a late licence in the bars?‡

- B2B + separate usr/group commercial meeting.

  [I have no idea what B2B is]

- Don't run parallel sessions because some people would like to attend both and therefore miss out.

- Improve organisation : 3 queues @ 15 min. each to register!!

- It is too expensive and is in Paris too often.

- Needs to be cheaper for we poor academics.

- Eat more 'possum.

- Follow the prepared programme.

- Too many last minute changes of the programme.

  [One speaker didn't come. One arrived after no word for three months, and reversal of talks in graphic session was at request of the speakers ... would you want to follow Sam?]

- Simultaneous translation might help and should be organised.

  [Is planned for Paris, but we will need to be warned in advance that presentations are to be given in languages other than English or French]

Comments regarding accommodation, venue and meals:

- I'd like a lower price for boarding.

- St. Catherines rooms were cold.

- Instruct Pembroke Dining Hall staff to be polite!

- Conference dinner should be included in cost of conf. (but not accom.).

- 4 course lunch was nice but took too long.....!!

- Buildings too far apart.
- Sessions not held in one building.
- Put things closer together.
- Exhibition should be in same building as technical sessions.
- Lunch too far from tech. session (rain!)
- Exhibition too far from tech. session (rain!)

  [ER II (reign!)]
- Change of session for special lectures impossible due to 10 minute walk
- If sessions closer together more possible to mix which sessions are attended.
- Having things in different locations means it is not possible to make many short visits to the exhibition (due to travel-time latency), so one is almost obliged to miss sessions to visit the exhibition at all.
- Please, please, please try and keep the exhibition, tutorials/industrial sessions and technical sessions in ONE BUILDING like Nijmegen or Leeds.

[I think I get the message]

Comments regarding presentations, view graphs, and A/V kit:
- Photo-copies of all slides & a summary of each talk.
- Keep viewgraphs uncluttered.
- Don't ask to have the lights changed.
- Learn to use the microphone.
- Keep view graphs VERY simple.
- Speak more slowly please and give the copies of the slides.
- Better prepared slides in some cases, and less superficial in some cases.
- Speakers should not try to make time by speaking very quickly even if their talk is long and interesting. Allow more time in the schedule.
- More details, less sales!
- I think more talks should be given by academics rather than sales directors or any other salesmen. There should be two sessions: one for system work and other for application area.
- A short course in how to use the audio visual equipment and the light dimmer switches.
- Most presentations were very high standard, but a few of the speakers should go on courses to learn how to give a presentation. Why not produce a sheet to send to all speakers, giving basic principles about slide presentations - eg. don't put too many words on the slide! -- Andrew Dunn, Rutherford Appleton Lab.

  [Okay ... You prepare it Andrew, and I'll hit them with it!]
- Better preparation and control of audio-visual equipment
- Better Audio/Visual support is needed.
- Sound for speakers of tech sessions was BAD.
- Should be instructed in use of visual aids and microphones before hand.
- More documentation from the speakers and it would be good if slides were more professional This is for the technical sessions. It is the big difference between technical and industry sessions.

  [what about the density of blue suits]

- Proceedings should be sent to attendees.
- Lets have speakers who do not spice their talks with in-jokes about EUUG committee.
- Speakers should be able to give demonstrations of their software, maybe during special sessions.
- Better control of AV.
- Distributed notes would avoid the frustration of not being able to write fast enough.
- More detail and technical content.
- Only problem is that some speakers assume we can hear questions from the audience - this is not always true. Warnings didn't have much affect.

Comments regarding scheduling:
- Avoid overlap of tutorials with main sessions (eg. opening).
- Needs more free time.
- It was difficult to find time to get to the exhibition (Nijmegen was better).
- Not enough time to attend technical sessions + visit exhibition. Had to choose between them. Split locations didn't help.
- With late dinner, exhibition could stay open for at least an hour after end of technical presentations.
- The opportunity to visit the exposition, wander through different sessions, was very poor compared with Nijmegen.
- (No) overlapping industry & technical sessions.
- Try to avoid parallelism of technical and industrial sessions with similar topic. It is difficult to choose.
- Exhibition should stay open later in the evening.
- Better co-ordination breakfast - beginning of sessions (Fri industry session)
- Keep to published items!
- Longer presentations (technical).
- Crowding, too many speakers in one session.

Comments regarding large percentage of foreigners:
- I'm an American speaker.

  [then you are probably more than just disturbed]
- But wasn't the quality good!

  [gave 3 as to how much he or she was disturbed]
- Less Americans though still have them.
- [disturbed but ...] Bell & Berkeley are important.
- I was disturbed by the large percentage of English speaking speakers.

Suggestions for future speakers:
- Speakers who will: discuss National language support; speak clearly without mumbling; and acquaint themselves with the audio-visual aids before the sessions.
- Dennis Ritchie

  [who?]

- Lauren Weinstein, Peter Langston, Steven Bourne

  [psl?? ... discussing empire no doubt]
- Dennis Ritchie & Brian Kernighan
- Lets hear from Dennis Ritchie, Ken Thompson, Brian Kernighan, Steve Johnson, Bill Joy, John (?) Horton, (UUCP< USENET< MAIL etc)

  [John Horton maybe, but Mark ... live!!! not unless he responds to the 'n' command ... and Bill in Paris again? only if he pays us to let him give another Sun marketing pitch]
- Prof. David Turner, UKC - writing an OS in an applicative language.

  [got it peter]
- kremvax!chernenko

[Ken and Brian were invited last April at the Australian boon-doggle and the invitation to Brian was repeated by pc@ukc today]

Suggested topics and comments on presentations:

- Needs to be more "hard" UNIX -- Lucasfilm was fun, but nothing to do with UNIX & there was too much of that work I think
- Industry sessions are important to ensure purity of technical sessions.
- Need more coverage on what the state of Unix is in the different European countries.
- York ADA/X25

  [A paper on X25 has already been submitted and accepted]
- Could we have more speakers on film making using UNIX please.
- More about people's own private projects that might be interesting. Thank you.

  [you're welcome]
- This conference should try to give some opportunities for European Unix to be presented, with no compromise on the high technical level which should be maintained and even improved.
- UNIX future developments (hardw. and soft) applications.
- Refereeing of Papers with preference for CURRENT (ie incomplete) work rather than presentations of results. This should not be an "academic" conference.
- Re importance of industry sessions - keep it for those who like it
- Very interesting. Not particularly relevant to me.

  [hmmm]
- Some talks should have been made during workshops. No real discussions.

[F.Y.I. We already have a paper from Germany (X25) and two from Sweden (ber@enea responded to threats)]

Comments regarding the Tutorials:

- Tutorials on day 1 started 9:00 when latest schedule stated 10:00. This was most annoying.
- Serge Gorray never showed up for his talk, causing great waste of time.
- More tutorials. Especially YACC which was cancelled!
- Bad organisation led to many people missing the first session and the wrong choice of a 9:00 start was taken. The YACC tutorial was never rescheduled either.

Results from the Survey taken at Cambridge

Number of questionaires returned: 148

Question:	Yes	No	Num
Was this your first UNIX conference:	62	86	148
Was this your first EUUG conference:	85	61	147
Will you attend any future EUUG conference:	142	1	143
Did you attend and /usr/group/uk sessions:	44	101	145
Did you attend any of the tutorials:	25	123	148
Was your attendance at the conference useful:	146	1	147
Did you visit the exhibition:	144	3	147

Questions requiring 0 to 4 response (0 means not at all, 4 means very)	Total	Num	Average
How important was the exhibition:	345	147	2.34
How do you compare this exhibition to others:	242	93	2.60
How important is a single location:	508	147	3.45
How important is full board:	294	146	2.01
How important are the technical sessions:	545	148	3.68
How important are the industry sessions:	279	142	1.96
How important are the other events:	409	145	2.82
Were the technical sessions useful:	446	147	3.03
Were you disturbed by the large percentage of non-european speakers:	125	145	0.86

Thanks to:

Comments on the Survey taken at Cambridge

David M. Tilbrook

The survey was prepared by Jim McKie and myself. Jim is the newsletter editor and I seem to have become the permanent conference technical sessions chairperson. Our objective was to obtain feedback on what people wanted at these conferences and how important the various aspects were, relative to each other.

As always, with any survey, the results must be evaluated in context. It was prepared in a hurry, and those people who chose to attend the /usr/group/uk and/or the exhibition Friday, wouldn't have had the opportunity of filling in the survey.

But, no matter how unscientific the preparation and the sampling, we received 148 returns which is approximately 40% of the registrants, and I think I can use the survey results as being relatively representative of the conference, at least of those conferees who attended the technical sessions.

I have included the table again (sorted from most important to least) dropping the exhibition comparison and usefulness of tech. sessions questions. I am also interpreting the 'Non-european speakers' question as being 'How important is it to have less ...'

Question:	NA*	0	1	2	3	4	Average
technical sessions:	0	1‡	2	8	21	116	3.68
single location:	1	1	4	14	36	92	3.45
other events:	3	14	9	29	30	63	2.82
exhibition:	1	7	28	46	39	27	2.34
full board:	2	33	15	43	27	28	2.01
industry sessions:	6	24	26	43	29	20	1.96
less foreign speakers:	3	90	19	14	10	12	0.86

The only surprise for me that full board didn't come higher. This is because I am frequently told that it is important for academics to have a single bill to pay as it makes it easier to get refunded. On the other hand, many conferees have reasonable expense systems so for them it wouldn't matter. I guess this question should have been split into different parts such as 'How important is inclusion of: 1) breakfast; 2) lunch; 3) dinner in single conference charge?' ... comments please.

Anyway, given the above I think Jim, myself and others who felt that the importance of having the conference in one location was greater than that of the exhibition were correct.

Unfortunately, the Paris conference has already been set. We won't have any industry sessions, but although we initially were to have a single hotel for all events, this was changed to accommodate a larger exhibition, even though the survey shows that one location is more important to people than an exhibition.

As for the rest of the survey, and in response to the comments, I propose that the schedule remain the same, except there will be fewer speakers with an average of 30 minutes per speaker and 5 minutes in each session for announcements and playing with the light switches and microphones.

Instruction sheets on operation of all the A/V equipment will be prepared for the session chairs.

Something must be done about the use and appearance of slides. The SLC USENIX conference required the speakers to deliver (before the conference) a set of 35mm slides. I would appreciate comments on the viability of this restriction. (To Andrew Dunn ... are you willing to prepare suggestions sheet please?).

---

* no response

‡ I looked this one up to determine why he/she came at all. The respondent said he/she would come again, but nothing was deemed as important except the central location. (I guess it was for the beer or seeing Jimmy's knees). Furthermore, the respondant was disturbed by the large %age of non-europeans (3), but commented "But wasn't the quality good!"

I won't feel guilty about inviting foreigners, but I will continue to actively encourage local speakers. Simultaneous translation will be provided given notice. With the reduction in the number of speakers, there will be no room for sales talks. If any company couples providing interesting speakers with slots for sales pitches, we'll just have to do without those interesting speakers (but not quietly).

Comments please.

I will be putting out a call for papers when I return from Paris Monday. We will be attempting to prepare a proceedings prior to the conference, so potential speakers be forewarned.

EUUG NATIONAL GROUPS

UKUUG
Chairman — Sunil Das,
Computer Science Dept.
The City University,
Northampton Square,
London, EC1.
England
Tel: 01-253-4399

NLUUG
Chairman — Teus Hagen,
ACE,
Associated Computer
Experts Bv,
nz voorburgwal 314,
1012 RV Amsterdam,
The Netherlands.
Tel: 020 24 54 44
mcvax!ace!teus

Secretary — Dr. M. van Gelderen,
NIKHEF-K,
Kruislaan 411,
Postbus 4395,
1009 AJ Amsterdam,
The Netherlands.

EUUG-S
Chairman — Bjorn Eriksen,
ENEA Svenska AB,
Box 232,
S-183 83 Taby,
Sweden.
Tel: 46 8 756 72 20
mcvax!enea!ber

Secretary — Hans Johansson,
Logica Svenska AB,
Norra Stationsg. 79-81,
S-113 33 Stockholm,
Sweden.
Tel: 46 8 34 91 10

AFUU
Chairman — Mr. Bernard,
c/o 152 bis,
avenue Marx Dormoy,
92120 Montrouge,
France.
Tel: Montrouge 655 45 50

Secretary — Ms. J-M Langlade,
c/o 152 bis,
avenue Marx Dormoy,
92120 Montrouge,
France.
Tel: Montrouge 655 45 50

DKUUG
Chairman — Keld-Jorn Simonsen,
Indre By-Terminalen,
University of Copenhagen,
Studiestraede 6 o.g.
DK-1455 Copenhagen K,
Denmark.
Tel: 45 1 1201 15
mcvax!diku!keld

FUUG
Chairman — Heikki Arppe,
Technical Research Centre
of Finland,
VTT/ATK,
Lehtisaarentie 2A,
2F-00340 Helsinki,
Finland.
Tel: 358 0 4561

Vice-
Chairman — Johan Helsingius,
Tontunmaentie 32 AS,
02200 Espoo 20,
Finland.
Tel: 358 0 460 033
mcvax!enea!tayes!cjn

GUUG
Chairman — Johannes Koehler,
GUUG,
Hauptstra 60,
8520 Erlangen,
Germany.

Vice-
Chairman — Daniel Karrenberg,
Universitaet
Dortmund,
Informatik IRB,
Postfach 500500,
D-4600 Dortmund 50,
Germany.
Tel: 231 755 2444
mcvax!unido!dfk

EUUG-I
Secretary — Giusi Molinari,
and Systems & Management spa,
Financial Piazza Solferino 7,
Manager 10121 Torino,
Italy.
Tel: (39) 11 538246

President — Alessandro Camici
Vice-
President — Gregorio Lerma

PUBLICATIONS AVAILABLE FROM EUUG

Title	Cost
1) UNIX Micros Catalogue Edition 1 (detailed description of all UNIX computers available on the market from almost 100 manufacturing sources)	£5.00 (Members) SOLD OUT
2) Report on Bonn Meeting 11-13 April 1983	£2.00
3) EUUG Conference Proceedings: Autumn 1983 Dublin	£2.00
Spring 1984 Nijmegen	£5.00
Autumn 1984 Cambridge (not yet in print)	£5.00
4) Language C — Standard Proposal	£2.00
5) USENIX Newsletter Vol 9 No. 1 February 84	Free of Charge

NOTE: The EUUG is NOT able to provide a subscription to the USENIX Newsletter — applications should be made to Deborah Scherrer, Usenix Association, PO Box 7, El Cerrito, CA 94530, U.S.A.

Publications Soon to be Available

Proceedings from the Spring EUUG Conference in Cambridge
UNIX Standard Proposal
UNIX Products Catalogue
UNIX Micros Catalogue Edition 2
* Current EUUG Newsletters are available at £2.00 per copy *

If you wish to order any publications please use the form below and return it with your remittance to the EUUG Secretariat.

ORDER FORM
PAYMENT MUST BE ENCLOSED

To: EUUG Secretariat, Owles Hall, Buntingford, Herts. SG9 9PL, ENGLAND

NAME ....PLEASE PRINT
ADDRESS ....
....
Please send me ....no. copy/ies of the following:-

....
I enclose my remittance in the sum of £.................... and understand a receipt will be sent to me.
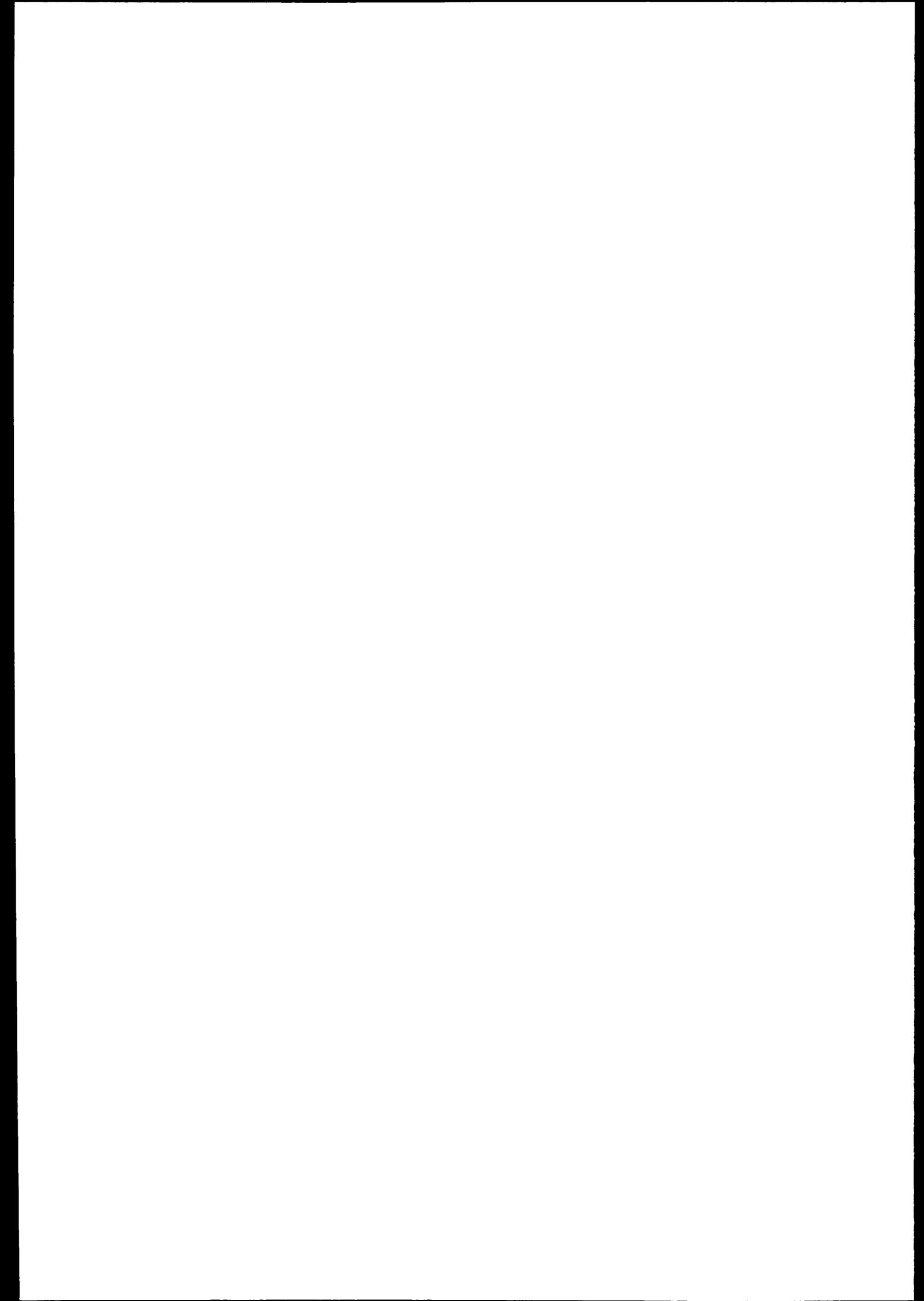
EUUG TAPE DISTRIBUTIONS — Cost £30—40 (UKL) each tape

NOTE — these are ONLY available to EUUG Members

NOTE —EUUG D1, D2, D3 have licensed material so at least a UNIX Version 7 (or System III, System V) source licence is needed for these tapes.

EUUG D1 R6   The EUUG D1 Release 6 distribution is a UNIX V7 system specially made for small DEC PDPs (11/23, 11/40 etc.) The kernel supports the UK terminal driver. A paper describing how to boot from tape is supplied. A copy of the source license agreement for at least UNIX version 7 should be enclosed with the request for distribution. The distribution tape is 2400 feet tape (1600 BPI).

EUUG D2   This distribution tape has the early Pascal compiler of the Free University of Amsterdam. A copy of the source license agreement for at least UNIX Version 7 is needed.

EUUG D3 R4   This distribution has the UNIX Networking software, news release 2.10.2, fully updated with bugs found up to date of shipment, and some other auxiliary programs. A second tape contains also the news received on the Continent for the last year. If a copy of the source license agreement for at least UNIX version 7 is included with the request, a fairly debugged version of UUCP with some accounting enhancements and X.25 support will be added to the distribution tape. The distribution is 600 feet tape (800 or 1600 BPI) — and a 2400 feet tape (1600 BPI).

EUUG D4   This distribution tape has the main UNIX software tools. The tools are sampled by the STUG (Software Tools Users Group). Most of the software is written in Ratfor, for which a Fortran support tool is included. If you want to have UNIX tools available on your non-UNIX system this tape is a must.
    The tape is available in different formats: DEC RSX, DEC VMS, UNIVAC, IBM VMS, UNIX tar, MIT (Machine Independent Tape) line fee format, MIT card format (80 columns). The distribution is a 2400 feet tape (800 or 1600 BPI). Please mark what kind of tape format is needed (MIT line feed format is default). The STUG group receives a $10 profit per tape from the EUUG automatically.

EUUG D5   This 600 feet tape contains a collection of benchmark programs. The software is in the public domain. No license agreement is needed to receive this distribution.

USENIX TAPES AVAILABLE FROM EUUG — 83.1 @ $75 and 83.2 @ £75
Usenix tapes contain contributions of software from various UNIX system Group Members (USENIX and EUUG). No list of contents is available.

NOTE: TAPE DISTRIBUTIONS ARE ONLY AVAILABLE TO EUUG MEMBERS
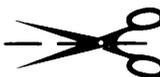
TO ORDER EUUG TAPE DISTRIBUTIONS

If you wish to receive an EUUG Distribution the following is needed:

— Your address (and institutional order number)
— the tape number (and format identification for EUUG D4)
— a copy of your source license (EUUG D1, D2, D3)

Please do NOT send an exchange tape to us. This causes delays and postage costs

Please do NOT send any money. A bill for the tape and postage costs will be sent separately

Orders should be sent to:- EUUG Distributions,
                Centrum voor Wiskunde en Informatica
                Kruislaan 413,
                1098 SJ Amsterdam,
                The Netherlands

- ✂ - - - - - -

ORDER FORM
EUUG TAPE DISTRIBUTIONS

To: EUUG Distributions, c/o Frits Thomas, Centrum voor Wiskunde en Informatica, Kruislaan 413, 1098 SJ Amsterdam, The Netherlands.

PLEASE PRINT
NAME ....
ADDRESS ....
....
I would like to order the following:-
....
....
....
I understand a bill will be sent to me separately.

A.F.U.U.

The A.F.U.U. (Association Française des Utilisateurs de systèms UNIX), the French Unix Systems Users Group, was founded in 1982.

After two years of existence, the French Association has over 120 members from all levels of UNIX usage: Hardware and Software companies, academics, and end-users.

Once a year, the A.F.U.U. organises a general meeting followed by technical conferences and an exhibition of UNIX hardware and software. For 1984, this meeting will be held at the Hotel PLM St-Jacques, Paris, on October 30 and 31.

Besides, the French Group has set up two technical committees, headed by high-level computer scientists. The first one, led by Michel GIEN (of the Centre National d'Etudes de Télécommunications) deals with the standardization and portability of UNIX; while the second, headed by Humberto LUCAS (from GOULD Informatique France) is working on the French uucp network (FNET) and its connections with other foreign networks.

The A.F.U.U. is also publishing, since June 1984, a Newsletter, which serves as an open tribune to its members, for adds, articles and news.

In France, while UNIX came later than in the other European countries, the use of the system is spreading out in the big companies (such as BULL and THOMSON) as well as in the smaller ones.

The address of the French group is: ASSOCIATION FRANCAISE DES UTILISATEURS DE SYSTÈMES UNIX, 152bis, avenue Marx Dormoy 92120 MONTROUGE France phone 33(1) 655 45 50, telex 205291, Chairman: Jean-Louis BERNARD, Secretary: Jean-Claude DOMONT, Treasurer: Didier TRASSAERT. Please contact Jeanne-Marie (Ms) LANGLADE for all enquiries.
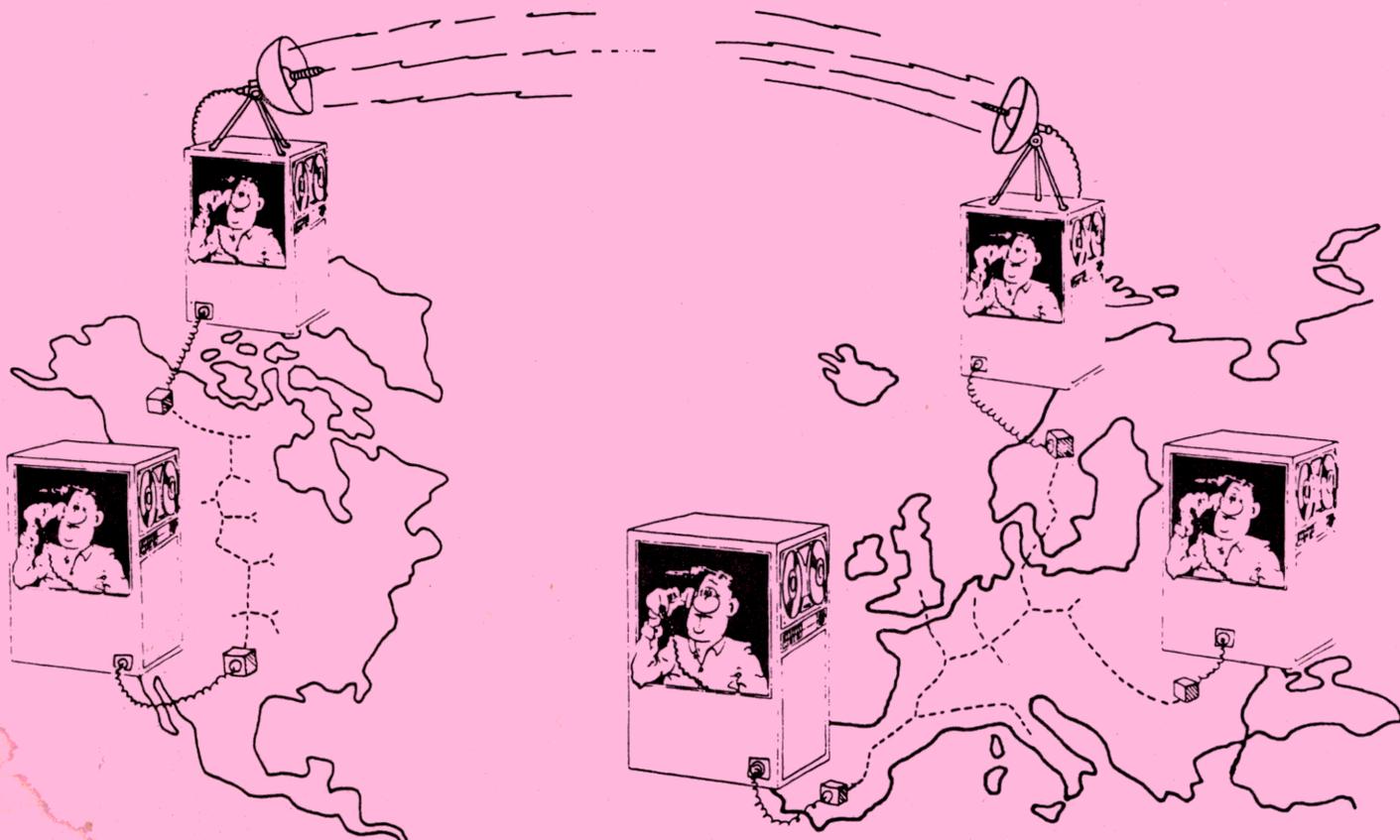
The Secretary
European Unix® Systems User Group
Owles Hall
Buntingford, Herts.
SG9 9PL.
Tel: Royston (0763) 73039.