NEWSLETTER

EUUG

European UNIX® systems User Group

Volume 10, No. 1
Spring 1990

CONTENTS

EUROPEAN UNIX® SYSTEMS USER GROUP NEWSLETTER

Volume 10, Number 1
Spring 1990

ISSN 1011-4211

Editorial Team:

Philip Peake
Publications Executive

CETIA
9 Avenue de Laponie
ZAC de Courtaboeuf
91940 Les Ulis
France
philip@cetia.fr

Alain Williams
Editor

Parliament Hill Computers Ltd.
7 Prospect Street
Caversham
Berkshire RG4 8JB
U.K.
addw@phcomp.co.uk
Tel: +44 734 461232
Fax: +44 734 474194

Typesetting:

Gina Baikenycz

The Instruction Set Ltd.
City House
190 City Road
London EC1V 2QH
U.K.
gina@inset.co.uk

Printed by:

Rank Xerox (Copy Bureaux) Ltd.
68-74 Rochester Place
London NW1 9JX
U.K.

Published by:

The EUUG
Owles Hall
Buntingford
Hertfordshire SG9 9PL
U.K.
euug@EU.net
Tel: +44 763 73039
Fax: +44 763 73255

Editorial

Alain D. D. Williams
addw@phcomp.co.uk

Parliament Hill Computers Ltd

Unix is Spreading

In the last issue we read about Unix in Czechoslovakia and thought it unexpected enough.

In this issue Dr Vladas Leonas writes from Moscow and tells us how they have had Unix for several years, maybe KREMVAX did have a grain of truth in it after all.

Anke Goos has been to East Germany and has news from there. She has taken over the EUnet column and this forms part of her first report. Please welcome and encourage her in this, let her know what **you** want to hear about.

Read about this on pages 34 and 48.

We will certainly hear a lot more from this part of the world as the Iron Curtain rusts away.

Conferences

Conferences form an important part of Unix User Group activities. Meeting and talking to others in your field is as worth while as listening to the papers and tutorials.

There is a lot of choice, check out the announcements on pages 9, 31, 37, 43 as well as the Unix Calendar on page 69. See what Marco Franzen though of the EUUG conference in Vienna on page 19.

Remember – even if you do not manage to get to a particular conference, the proceedings are usually available though your User Group.

There is more to a conference than just going to listen to fine work, people are needed to put forward papers. So why not do presentation on what you are going now? It will be an excellent way of letting others know of you achievements and of getting feedback on things which have not (yet) gone right.

A guide to authors can be found on page 80.

Future EUUGN Dates

The number of copies of the EUUGN that are printed has now increased to 5,000. This lead to a complaint from Joe, our printer, he wanted some more time.

To keep the publication dates the same the copy dates for future issues has been brought forward a week, the new dates are printed below.

Remember these important dates:

	Copy Date	Publication Date
1990	16 April	1 June
	16 July	1 September
	15 October	1 December
1991	14 January	1 March

Want some free advertising ?

There is often some space left at the end of an article on a page. I will fill this with short announcements as space permits, so send me brief outlines of your products or services.

If you want to advertise in a more formal way you can place full or half page advertisments. The rate for EUUG members £300 per full page. Please contact Owles Hall for full details.

Contributors Wanted

As ever I am always pleased to receive contributions from readers, be this a paper, a short announcement or some information for the UNIX calendar.

Do not worry if your English is not perfect, I can help you.

Personal Bit

Congratulations are due to Helen Gibbons who runs the EUUG secretariat at Owles Hall. She is engaged to be married on 23rd June – no he has got nothing to do with computers.

We wish them all the best and a happy life together.

From Unix To Open Systems

Cédric Thomas

Pierre Audoin Conseil
65 rue Desnouttes
75015 Paris
France

Cédric Thomas has zero years of Unix programming experience, he is not even a Unix user and not even a Data Processing specialist. That probably explains why, 10 years ago, he chose to become a "Strategy" consultant for the computer industry to avoid completing his PhD in Economics. As a director with PAC in Paris, Cédric has developed since 1984, amongst other things, an annual survey on Unix trends both in his country and worldwide.

Summary

Unix standardisation strategies are generating the market for open systems.

In this paper, we shall propose a general model for open systems including, apart from the application, three functional areas which we shall call environments: the application environment, the operating environment and the hardware environment.

This model makes it possible to place the importance of new technologies in relation to the market: standards and norms, client-server architectures, Distributed Network Computing (DNC), user interfaces, etc.

Introduction

Standardisation in the Unix industry has not stopped at the operating system. As in the rest of the data processing industry it has also concerned languages, and the task of POSIX working groups is gradually becoming more oriented towards the whole software environment. In the meantime, X/Open is probably the driving force behind the structuring of the market, with its proposal for a Common Application Environment (CAE).

The fact that Unix is the operating system which serves as the basis for work on norms is not sufficient to turn Unix itself into a norm. The norm which is emerging at present (POSIX) is not an operating system but an interface between an operating system and application software. There is nothing to prevent the development and marketing (i.e. implementation on machines) of operating systems other than Unix but which nevertheless conform to the norm.

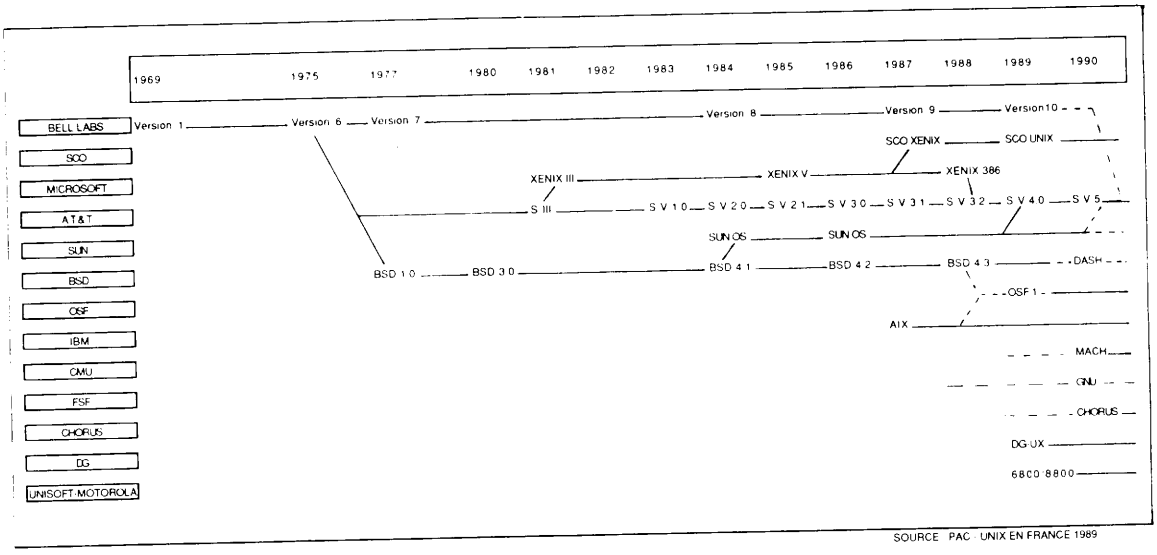
AT&T's entire Unix standardisation strategy has led to the industry becoming structured in such a way as to encourage the appearance of products in competition with Unix. The structuring of the industry thus leads to diversification. Unix was simply the first tree in a whole forest of open systems.

The new families being developed, such as the OSF and MACH families (one could also mention the AIX family, or even the GNU family) will take their share of the markets. The product which goes under the trademark of Unix will have contributed to the creation of a larger market, the market for "open systems"; it will subsequently be identified exclusively with the System V families and will be in competition with the other

families. Or, to use more familiar terms, the Unix operating system will be in competition with other "open" operating systems (see chart).

It is already becoming apparent that the open systems market of the mid-90s will be intensely competitive. In technological terms, standards will make it possible to substitute one vendor for another. Any loss of competitiveness will inevitably mean the withdrawal of one producer to the advantage of its immediate rival.

As with Hewlett Packard's takeover of Apollo, mergers will be made a great deal easier by standardisation, which will allow the fusion not only of technologies and product catalogues but also customer bases. Standardisation leads directly to the concentration of the industry.



THE HISTORICAL DEVELOPMENT OF UNIX FAMILIES

From the traditional model...

The surge in technology which is a feature of the development of the Unix market has been paralleled by innovative concepts whose importance is not immediately clear in the context of conventional ideas about data processing. Renewal is necessary and, if a picture speaks a thousand words, a graphic description of this process will no doubt be more vivid than a painstaking demonstration. In such a way, what follows describes a rather complex concept through a simple model.

Of course, the model should not be taken for the reality. The point of the model is to give a general outline and to situate the importance of the innovations and the links between them. The model expresses the general rule. However, in an industry where the main forms of competition are the race for technological progress, the segmentation of markets and the differentiation of products, it should be borne in mind that all the players are jockeying for the most advantageous positions.

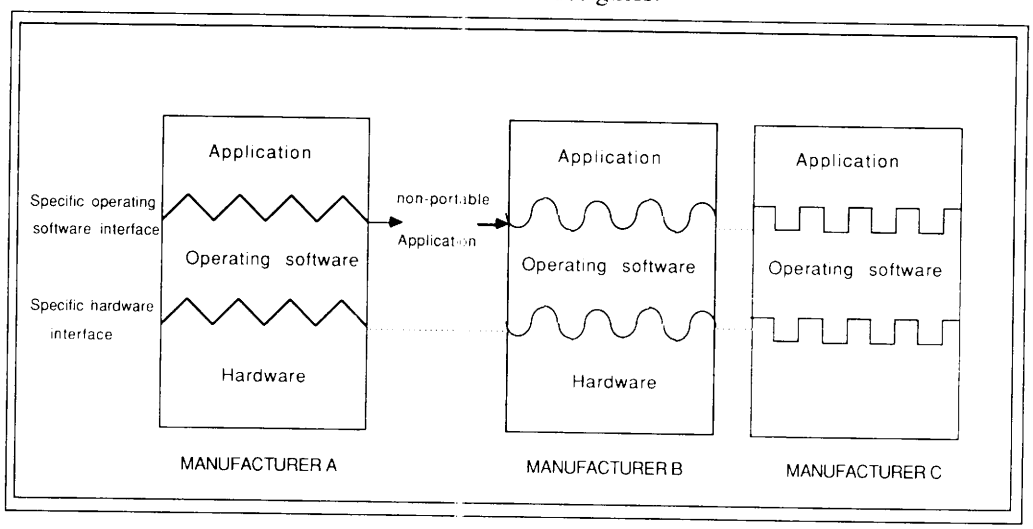
We have attempted to show in diagram form what a data processing configuration might look like and to highlight the significant innovations of the Unix market. Our model is directed towards this end and is based on layer diagrams (like those of the OSI reference model). These have the advantage of clarity as they have a single dimension, a single thrust and thus expose the essential connections between foundation and superstructure.

The traditional model represents a data processing configuration in three parts: hardware, operating software, application (see diagram). The operating software includes the operating system and the various utilities that go with it and that are sold by the computer manufacturer or by strategically attached specialised vendors. In the traditional model the emphasis is on the link between hardware and operating software because, in the development process, computer architecture comes first. As a result, the operating software is geared to the architecture and its specific features. The application which is developed on this basic setup in turn incorporates those specific features

and hence cannot be ported from one system to another. That, briefly, is the inherent condition of an industry of proprietary systems.

.... to the open system model

The fundamental change wrought by Unix was to turn the traditional process for developing operating systems on its head. The architectural restrictions which for manufacturers normally determined the development of an operating system did not apply to Unix. As Unix was not a "product" and hence was not bound by the limits of an industrial strategy, the only specifications it was required to meet were self-imposed by its designers.



SOURCE : PAC - UNIX EN FRANCE 1989

A VIEW OF NON-PORTABILITY OF APPLICATIONS

Traditional operating software is bulky, as it takes into account the computers on which it runs and has to be capable from the outset of meeting all users' likely expectations. Unix, on the other hand, is less complicated and does not seek to meet all needs in advance. This approach is manifest in the system design which retains only essential functions in the kernel. The remainder is transferred to the periphery. The investments made by software developers targeting the Unix market tend to be in this periphery: development tools, file management tools, text processing and editing tools, communication, windowing, configuration management tools, etc. In the periphery, a distinction can be made between products which are systems-oriented (shell utilities, system commands, real-time or fault tolerant extensions, etc.) and those which are application-oriented (user interface, data management, programming tools, communication).

Above the operating system proper there is no longer that broad mass of programs known as the application. It has given way to something more structured in which it is possible to distinguish a "solution" part of the application (the application proper) and a "tools" part, built up on a "system" part.

Using this distinction, we can now consider the application as topping off a three-level structure made up of two levels of software and one level of hardware which we shall call environments (see diagram):

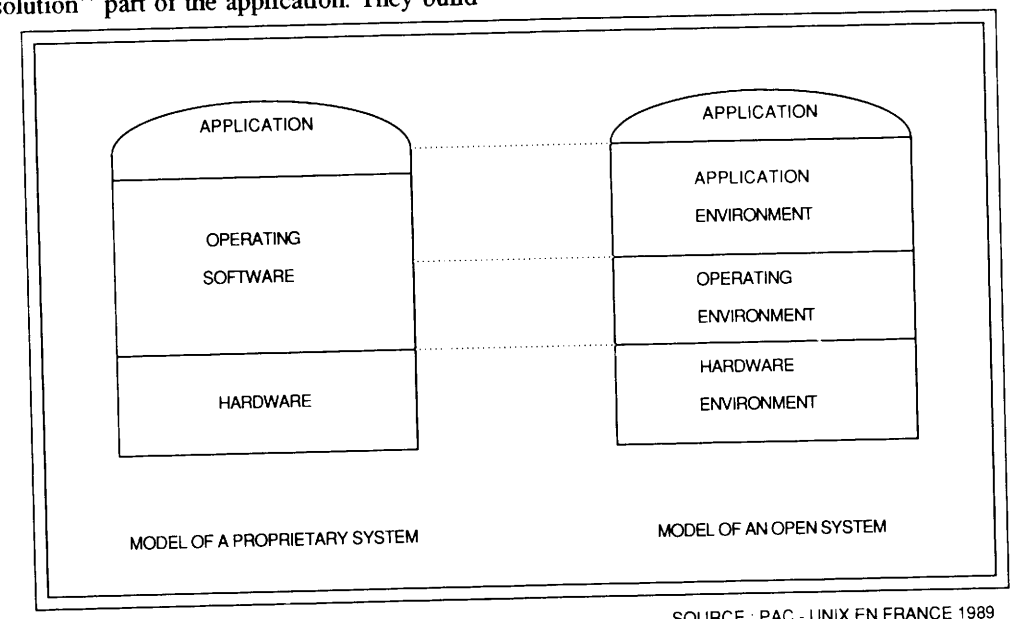
— application environment

— operating environment

— hardware environment

From an industrial point of view, the application environment and operating environment layers are made up of products offered on the market by manufacturers or increasingly (and this is something new) by specialised producers. These products mean that application developers, either at users or in software houses, now work only on the "solution" part of the application. They build this part "above" the application environment using the tools available at that level.

By paying close attention to the industrial and technical reality it is possible to determine a structure for each of these layers and hence to envisage a general model of an open system (see diagram).



STANDARD 1 SOURCE : PAC - UNIX EN FRANCE 1989

TWO BASIC MODELS

The layer structure of the hardware environment appears if the complete configuration is considered as a set of concentric rings: 1) processor, 2) bus structure (one or more buses), 3) peripherals.

Following the same pattern the next level, the operating environment, includes closely linked software such as 1) the system kernel, 2) the shell and its utilities, 3) the system's integrated extensions.

The application environment has the most complex structure. In terms of the approach used in our model it has four elements: 1) programming languages and tools, 2) the user interface, 3) data and file management, 4) communication.

The complete model (which can be refined) proposes a five-layer structure for the application environment, in relation to which each of its parts can be constructed. There are thus several specific models: one for programming tools, a second for the user interface, a third for data management and a fourth for communication.

The upper level of the general model is the application. It includes only the "solution" part of the traditional application layer, of which all of the "tools" part is now incorporated into the application environment.

Developing a model such as this one is not gratuitous. It makes it possible to situate the importance of the technical innovations and strategic behaviour which are structuring the open systems market.

Standardisation, the work being undertaken by consortia, user interface strategies, Distributed Network Computing (DNC) are all determined in relation to different sets of technologies within the operating environment and the application environment.

Open systems

The evolution of the concrete forms of Unix is a continuous historical and technical phenomenon. The upsurge in new product announcements indicates that new ways of writing and implementing Unix are transforming the market.

At a purely technical level, the most significant innovations toady are perhaps:

— rewriting the kernel in object-oriented language

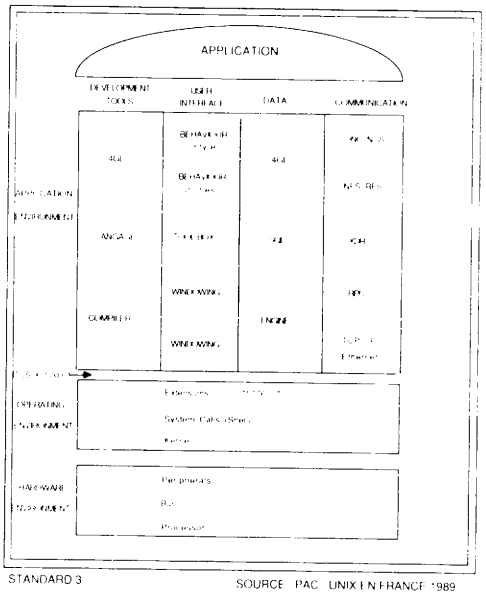
— implementing the system on multiprocessor architectures

— extending virtual memory capacity.

At a more structural or industrial level, systems-independent means of certification or validation are being seen as increasingly important. This independence guarantees a possible harmonisation on the basis of which differentiated but nevertheless compatible products can be developed.

What is the definition of an open system? It is a system which conforms to POSIX and to the rules on openness laid down by one of the standards authorities recognised by the market. As there are several of these authorities there can be several conceptions of open systems without the idea of openness being brought into question.

Rules on openness are themselves of strategic importance. This explains on the one hand why associations and consortia are competing to impose their "legitimate" conception of openness and on the other hand why producers are joining forces. Operating systems themselves become an area of competition, forming a specific market with more or less specialised producers, and products with different functions corresponding to different needs.



COMPREHENSIVE MODEL OF AN OPEN SYSTEM

Norms and standards

In the Unix industry it is important to distinguish between norms (i.e. the result of official procedure) and standardisation (i.e. de facto standards), as the two approaches do not apply to the same object. This distinction sheds light on how the market might develop in the long term and on producers' strategies.

The major distinction to remember is that the object to which a norm applies is an interface, i.e. a level of services rendered by one technological domain to another (by the operating environment to the application environment, for example). Standardisation, however, gives only an implicit description of the interface, being based on products which are characteristic of the technological domain concerned and which offer the required level of service.

It should be borne in mind that no official body is at present working on norms for the product Unix. Unless there is a sensational, but unlikely, turn of events, Unix is not and never will be a norm. However what is becoming an international norm (POSIX) is an interface between an operating

environment, whatever the products of which it is made up, and the application environment which it supports.

Besides norm procedures, strategic manoeuvres are taking place around standardisation. That is the reason for the competition between Unix International and OSF.

The position of X/Open is more complex but also more interesting. The members of this group of manufacturers recognise each other on the basis of POSIX but they are not bound to adopt exactly the same operating system. The group's strategy consists in defining and adopting a Common Application Environment. In this way the specific features of each architecture are respected and the individual advantages of each vendor are preserved.

Unlike POSIX working groups, which look at specific, precise subjects, X/Open looks at the whole application environment. What characteristics must a data processing system have in order to be standard? The definition of CAE attempts to answer this question.

CAE is not a standard itself: it is a super-set identified with the application environment layer of the open system model. It includes norms once they have been defined and products (de facto standards) if norms do not yet exist.

User interface

The graphic user interface carries out extremely important functions in present day data processing. Advanced interfaces are genuine graphic application environments which act as a framework for several functions:

— man-machine communication

— communication between applications

— communication between systems

The technical and economic importance inherent in the choice of such an interface becomes apparent. The model used here, with three main layers, is based on the X-Window model: 1) windowing, 2) tool box, 3) behaviour. Choosing similar technologies in one layer must not prevent dissimilarities in another layer and vice-versa. Given this basis it is possible to imagine multi-vendor, multi-architecture environments. The foundation of the interface is the window manager, or the windowing system. Identifying a layer with the windowing system supposes that the latter has a certain degree of autonomy. That means that there can be a windowing system independent of the graphic style.

In the next layer up, the tool box offers developers higher level commands than the primary functions of the windowing system proper. These commands are the building blocks of a graphic application. Also referred to as "widgets" or graphic objects, they include buttons, menus, command bars etc. All these building blocks depend on a basic layer of the tool box called "intrinsics" whose role is to ensure the connection between the widgets and the windowing system.

Lastly comes the behaviour layer. This is what the user sees, and it too includes two distinct sets of elements: utilities, which ensure that applications can work together, and style, the particular look and feel of an interface.

The emergence of user interfaces and of a reference model offers precise opportunities to developers of packages. The most opportunities are to be found in the upper layers of the model. In the tool box layer there are widely varying needs for widgets and although widget libraries are available in the public domain there will always be room for new tools such as "wysiwyg" widgets for text applications, graphic widgets for diagrams, or specific widgets for telecommunications, development applications, process control, etc.

Distributed Network Computing (DNC)

Distributed Network Computing is the shape which data processing seems to be taking at present. The concept of Distributed Network Computing is based on the division of applications between "client" workstations, for everything that concerns interactivity with the user, and "servers", for processing applications wherever they may be located. Whether it is a marketing concept or a technological reality, Distributed Network Computing is an element which structures the open systems market. DNC is the result of identifiable technological advances in both software and hardware and, an unavoidable condition, is based on standards.

In software terms, the basis of the Distributed Network Computing concept is the client-server

arrangement already integrated into advanced DBMS. The main foundations for the emergence of DNC are the spread of the X-Window windowing system, the maturing of major distributed environments such as ONC or NCS, and the development of targeted "client environments" for office automation workstations.

The client-server arrangement is also to be found in hardware. Here, micro-computers or workstations are the typical clients. Client systems are tending to be RISC machines in scientific and technical fields, while business and OA machines tend to be 32-bit workstations (this must be the market for 386 platforms) or the X-terminal, a new category of hardware based on X-Window technology.

Servers are genuine processing "engines" specialising in the various types of tasks which such machines are required to carry out (calculation, archiving, database, communication, etc.). DNC encourages the entry of new producers and the appearance of differentiated systems intended as servers. Lastly, and above all, DNC is based on standards: communication standards such as OSI, TCP/IP or, for LANs, Ethernet, Netware, Token Ring, Starlan etc.; file sharing or integrated environment standards such as NCS, NFS, RFS, ONC, etc.; and user interfaces, themselves based on X-Window technology, the two standards being Open Look and Motif. It should also be pointed out that the development of DNC, although in theory independent of machines and operating systems, in the end involves only machine bases sufficiently widespread to be considered standards, such as MVS-370, VMS VAX and MS/DOS-INTEL architectures, and Unix machines. Apart from IBM which, with SAA, is implementing its own conception of DNC, Unix appears to be the driving force in this field today.

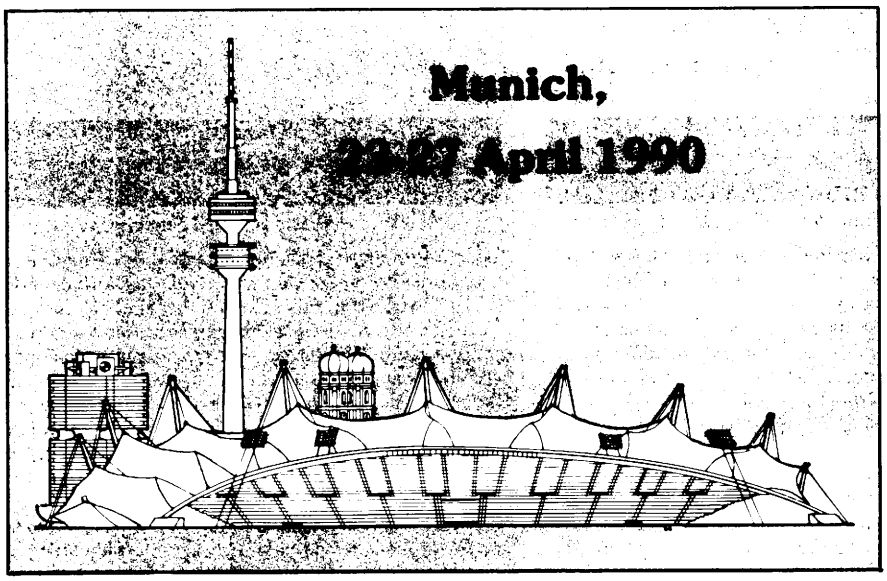
The concept of DNC sheds new light on the importance of standards. DNC can only develop on the basis of standards and it is not possible to imagine assemblies of dissimilar configurations with systems and software from different vendors unless they speak a common language. The investment is so great as to be out of reach for a single producer unless it has a monopoly or a quasi-monopoly position.

By imposing a certain degree of technical harmonisation, standards protect users and manufacturers against innovations which destabilize the market. In helping to reduce technological uncertainty, norms and standards are shaping the data processing of the 90s. The client-server arrangement is also to be found in hardware. The client-server arrangement is also to be found in hardware.

European UNIX® systems User Group

EUUG Spring '90 Conference and Exhibition

Sheraton Hotel Munich



PROGRAMME

Technical Tutorials on UNIX and closely related subjects will be held on Monday 23rd and Tuesday 24th April. The three day conference will follow starting with an opening session on Wednesday 24th at 09.30 and will continue until approximately 16:00 on Friday 27th April. The Conference will address the subject of UNIX as an example of a maturing technology. The papers will cover generic issues and standards such as the implementation of generic virtual memory systems or implementations of standardised protocols, and will highlight the fact that UNIX is becoming the platform for significant technological advances like object oriented programming and multi-threaded processes.

About thirty real experts will present their papers in two major sessions on:

- Object Oriented Applications
- Operating Systems
- Network Services
- User Interfaces
- Network File System
- Applications
- Security

The Exhibition

The Exhibition will take place at the Sheraton on 24-25-26th April. Companies wishing to book exhibition space should contact the exhibition organisers:

MSM GmbH
Messe-Service-Merkhoffer
Karl-Schmid-Str. 6
Postfach 82 04 40
D-8000 Munich 83 Tel: + 49 89 429343
West Germany Fax: + 49 89 421046

EUUG

European UNIX*systems User Group

PROGRAMME OF EVENTS

TUTORIALS

Only EUUG National Group or Direct Members are permitted to attend Tutorials.

Each Tutorial lasts for one whole day and will start at 09.30.

Monday 23rd April

Registration from 08.00 in the Ballroom Foyer

Tutorial M1 – Programming with OSF/Motif™

Tutor: Hans-Joachim Brede, BREDEX GmbH

The X Window System™ has become an industry standard software package. Because it is a very low level interface it requires higher level libraries for building applications with consistent and easy to use user interfaces. These libraries consist of two parts. One describes the architecture and contains basic routines (Intrinsics). The other contains the user interface elements (Widgets). OSF/Motif is such a toolkit implementation which becomes very widely accepted.

This tutorial gives an introduction into the OSF/Motif components. It explains the Intrinsics architecture and introduces the Motif widgets. The delegate will learn how to use and how to customize those widgets at application startup time.

Extra focus will be given to UIL (User Interface Language) which is a new language that allows programmers to separate form and function of the user interface in his code. UIL is also a powerful tool for fast prototyping.

This tutorial is targeted to people who want to start programming with OSF/Motif or other toolkits. It requires basic knowledge of the X Window System artchitecture and knowledge of programming in C.

Hans Joachim Brede is managing director of BREDEX GmbH, which has focused on user interface related topics for a couple of years, and has given tutorials and training on those topics.

Tutorial M2 – Advanced Network Programming

Tutor: Richard Stevens

Little information exists on the actual development and programming of software to run in a network environment. In the UNIX world this means understanding sockets or TLI, and also the basis primitives provided for concurrent programming (fork, exec, signals). Some knowledge of communication protocols is also needed, although at a different level from that described in most current networking texts. The goal of the tutorial is to provide the programmer with the basics required to write network programs and to develop and examine actual examples. The orientation is towards sockets and TCP/IP, since these are the most widely used today. Also, most System V network packages provide a socket interface today.

The tutorial is divided into three parts:

- Introduction (15%) A programmer's view of the TCP/IP protocols, overview of UNIX process handling, coding a daemon process.
- Berkeley sockets (50%) All the socket system calls, TCP and UDP client server examples, reserved ports, stream pipes, passing file descriptors, asynchronous I/O, I/O multiplexing, out of band data, broadcasting, inetd superserver, constructing Internet addresses.
- Examples (35%) Security in 4.3 BSD, remote command execution, Internet ping program (raw sockets), pseudo terminals and remote login.

Intended Audience: UNIX C programmers interested in learning how to write programs that communicate across a network. A basic understanding of networking concepts is assumed.

Richard Stevens received his Ph.D in systems engineering from the University of Arizona in 1982. He is the author of the forthcoming book "UNIX Network Programming". Since 1982 he has been with Health Systems International in New Haven, Connecticut, where he is Vice President of Research and Development.

European UNIX®systems User Group

EUUG

PROGRAMME OF EVENTS

TUTORIALS

Monday 23rd April

Registration from 08.00

Tutorial M3 – Chorus

Tutor: Mark Rozier and Francois Armand

The CHORUS architecture has been designed for building "new generations" of open, distributed, scalable Operating Systems. CHORUS has the following main characteristics:

- a communication-based technology, relying on a minimal Nucleus integrating distributed processing and communication at the lowest level and providing generic services used by a set of subsystem servers to provide extended standard operating system interfaces.
- real-time services provided by the real-time Nucleus, and accessible by "system programmers" at the different system levels.
- a modular architecture providing scalability, and allowing in particular dynamic (re)configuration of the system and its applications over a wide range of hardware and network configurations, including parallel and multiprocessor systems.

A UNIX system (called CHORUS/MIX) has been developed along this architecture: others, such as object oriented systems, are currently being studied.

The tutorial will give a "detailed overview" of CHORUS/MIX, by describing the services it provides as well as its internals architecture. It is particularly aimed at people willing to get familiar with distributed system technology. The knowledge of the UNIX interface and of some of its internals will be helpful to fully benefit from the second part of the tutorial.

The first part will present the CHORUS architecture principles, the services provided by the CHORUS nucleus, and its internal structure. Particular attention will be paid to IPC, threads and virtual memory management. Comparisons with other existing distributed systems approaches will be given.

The afternoon will concentrate on the UNIX sub-system which runs on top of the CHORUS nucleus. The extension of the UNIX kernel services to distribution as well as new services (IPC, real-time, multi-threaded processes) provided at the CHORUS/MIX interface will be discussed. The modular architecute of CHORUS/MIX will also be detailed providing a clear understanding of "how things work". Examples will illustrate the presentation.

Outline of the day:

1. Introduction (Morning)
   - CHORUS architecture, subsystems, modularity
2. The CHORUS Nucleus
   - Executive Nucleus (Fundamental abstractions: Actors, Threads, ipc, Regions)
   - Supervisor (Access to hardware, real-time features)
   - Virtual Memory Manager (Interface, Architecture, External mappers)
   - Network Access (Interface, architecture)
3. CHORUS/MIX (Afternoon)
   - Introduction, goals
   - Services provided (Extension of UNIX services, ipc, real-time, multi-thread)
   - Implementation architecutre (modularity, servers, protocols, configurability) Examples of system calls operation
4. Current Status and Future Directions

Marc Rozier is the head of the CHORUS Operating System development team within Chorus Systemes. He graduated from ENSIMAG (Grenoble) before earning a doctor's degree in Computer Science from INPG (Grenoble).

In 1981-82 he was involved in the CESAR project at IMAG (Grenoble), working on the Validation of Distributed Systems. He gained experience in programming languages for distributed applications and distributed systems.

He joined INRIA in 1982 as a researcher in the CHORUS distributed operating system project. He worked on both the design and implementation of two version of CHORUS. In 1987, he became one of the founders of Chorus systemes.

Francois Armand is reponsible for the distributed UNIX aspects of the CHORUS Operating system. He graduated from ENSEEIHT (Toulouse) in 1977, and got his first job at STERIA, a software company. In 1980 he joined the SOL Project at INRIA, where he ported a UNIX/V7 system on a BULL DPS6 mini-computer. He then designed and implemented the SOL UNIX compatible system on this machine, and wrote the specifications for its port on an Intel 80186 based computer. He then designed a UNIX emulation on top of the BULL DPS7/GCOS7 system. In 1984 he was seconded to CNET to work on the interconnection of UNIX with Transpac.

He joined the Chorus Project at INRIA in 1985, where he contributed to the design of the UNIX Interface in CHORUS and ported and extended the UNIX File system and associated tools. At the beginning of 1987 he joined Chorus systemes, when the Company was formed.

EUUG

European UNIX® systems User Group

PROGRAMME OF EVENTS

TUTORIALS

Tuesday 24th April

Registration from 08.00 in the Ballroom Foyer

Tutorial T4 – Project Athena

Tutor: Dan Geer

A quantum change in the way in which technical computing is best done is well under way. Hardware progress (such as cheap cpu cycles), software progress (such as network transparent window systems), and political progress (especially the appreciation of open systems) represent substantive differences between the world of even two years ago and the world now possible. Project Athena has spent the last six years contructing one of the best examples of computing environments. At M.I.T. ten thousand users are sharing a pool of eleven hundred workstations and one hundred various servers; all figures are growing in real time. This tutorial will discuss the components of a network services model of computation subject to the real life constraints of the academic setting. Particular emphasis will be placed on wide-area system management and the maintenance of a coherent computing environment on a large scale, heterogeneous computing infrastructure.

A. Introduction to the Athena Technical Plan
B. Objectives and requirements of the Athena system
   1. Athena objectives
   2. List of Athena requirements
C. Project Athena's model of computation
   1. Introduction
   2. The nature of academic computing
   3. Hardware
   4. Coherence
   5. Using the system
   6. The Athena File Storage Model
D. Evolution to the Athena model
   1. Phase I -- Off the shelf
   2. Phase II -- Public Workstations
   3. Phase III -- Private Workstations
E. Athena Developments in Network Services
   1. Service Management System
   2. Authentication, Name, List and Accounting Services
      1. Kerberos Authentication and Authorisation System
      2. Kerberos Applications
      3. Hesiod Name Service
      4. List Service
      5. Accounting Service
   3. Storage Services
      1. Remote Virtual Disk System
      2. Remote file Service
   4. Other Services
      1. Zephyr Notification Service
F. Athena Developments – Workstations
   1. User Interface support
      1. Display management interface
      2. Windowgram facility
   2. Workstation infrastructure
      1. Workstation system
      2. Toehold system
   3. Videodisk system Extensions
G. Athena Developments -- Other
   1. Operating System
      1. Athena changes to Berkeley 4.3 UNIX
      2. Athena changes to Digital 2.0 Ultrix
      3. Athena changes to IBM ACIS 4.2A UNIX
   2. Laboratory support system
H. Miscellaneous Plans
   1. Protection of third party software
   2. Generating System Software
J. Athena Standards and Specifications
   1. Project Athena standards
   2. Related standards and publications
   3. Athena standard third-party software
K. Standard System Configurations and UNIX configuration files
   1. Common configuration base
   2. Workstation configuration
   3. Server configuration
   4. Service configurations

Daniel E. Geer, Jr., is the Manager of Systems Development, Project Athena, where he is responsible for all technical development, including X, Kerberos, Hesiod Zephyr, Moira and all other aspects of the Project Athena Network Services System.

Previously, he was Systems Manager for the Health Sciences Computing Facility, School of Public Health, Harvard University.

He holds a Bachelor of Science in Electrical Engineering and Computer Science from MIT, and a Doctor of Science in Biostatistics and Epidemiology from Harvard University. Active in USENIX, he was most recently on the Program Committee for the Washington, DC Conference.

European UNIX®systems User Group

PROGRAMME OF EVENTS

TUTORIALS

Tuesday 24th April

Registration from 08.00

Tutorial T5 – An Introduction to 4.3 BSD Internals

Tutor: Tom Doeppner

This tutorial is an introduction to 4.3 BSD internals. It is geared to the system administrator or C programmer who has had little or no experience with the internals of UNIX. It will cover the following topics:

★ the UNIX process
- internal representation
- system calls
- synchronization
- signals
- scheduling

★ the file system
- incore data structures
- directories
- buffer cache
- disk layout
- terminals and pseudo terminals
- device driver interface

★ virtual memory
- address spaces
- paging mechanisms
- swapping

★ interprocess communicaiton
- sockets
- networking interface

Thomas W. Doeppner Jr. received his Ph.D in computer Science from Princeton University in 1977 and has been on the faculty at Brown University since 1976, where his research interests are in operating systems and parallel programming. He has lectured extensively on UNIX internals over the past six years for the Institute for Advanced Professional Studies.

Tutorial T6 – Systems V – Kernel Structure and Flow

Tutor: Bill Rieken

This tutorial introduces participants to UNIX internals and device driver interfaces without detailed study of source code. Key control blocks, major routines and basic synchronization primitives are described in high-level structure and flow terminology. The course is intended for applications programmers who need to know how UNIX works inside without a detailed code-level knowledge. Systems programmers who need an overview of the kernel before studying its code in depth should also benefit.

The course explores files and processes: file subsystem (file table, inode table, mount table, super block); buffer cache (buffer headers, sleep/wakeup, want/lock flags); process subsystem (process image, open files fork, exec, signal, kill); and I/O subsystem (device switch tables, terminal I/O, buffer (disk) I/O, interrupt handlers, critical sections).

William "Bill" Rieken Jr. M.S. is an independent software consultant specializing in UNIX training and consulting. He has over 19 years of experience with more than 30 organisations in business, industry and government, covering a broad range of business data processing, scientific computing and systems software projects. He is currently working on system programming for a distributed AIX system with X windows.

He is past President and Education Chairman of the San Francisco Golden Gate Chapter of the ACM, a member of the IEEE Computer Society and the ASM and former Marketing Referral Chairman of the Independent Computer Consultants Association (ICCA).

Wednesday 25th April
Registration from 08.00

Preliminary Technical Programme

KEYNOTE	
	Richard F Rachid UNIX, Mach and OSF
coffee-break	
OPERATING SYSTEMS I	
	Francois Armand, Frederic Herrmann, Jim Lipkis, Marc Rozier Multithreaded Processes in CHORUS/MIX
	Herman Moons, Ulf Hollberg Distributed Computing in Heterogeneous Environments
	Iain Elliot The AIX Version 3 Kernel on IBM's New Risc Architecture
lunch	
OBJECT ORIENTED APPLICATIONS	
	Sanjiv Gossain RApp: A Generic Routing Application in C++
	Peter Polkinghorne PatMat - a C++ Pattern Matching Class
coffee break	
NETWORK SERVICES	
	Gabriele Cressmann-Hirl Network Management Gateways
	Nils Meulemans A Simple MultiPurpose Internet Directory
	Toshiharay Harada, Takehiko Nishiyama, Hidekazu Enjo PCSERVE: an attempt to integrate PC users into the UNIX community

Electronic Mail at the Conference

Delegates can be reached during the Conference by EUnet mail. Messages will be printed and posted on the message board.

To reach people at the Conference use the following address:

firstname-lastname-organisation@euug-con.EU.net

or

firstname-lastname-organisation@euug-conf.uucp

Delegates should limit the amount of messages forwarded to this address.
Distribution lists should not be forwarded.
Delegates will also be able to send mail.

Thursday 26th April

Registration from 08.00

Preliminary Technical Programme

OBJECT ORIENTED PROGRAMMING

C Vieville, A Derycke, P Vilers
Architecture of a Collaborative System using Smalltalk and UNIX

Marion Wittstock, Jurgen Schiewe
Object-orientation: The way to cooperative software development

N Giambiasi, C Oussalah, L Torres
Clean Semantics of Multiple Inheritance

coffee break

OPERATING SYSTEMS II

Zsolt Hernath, Makar Szokolov
BUD – Backtrackable UNIX Data Control

Bala
Yeast Yet another Event Action Specification Tool

Dieter Konnerth
Dynamic Driver Loading in UNIX

lunch

USER INTERFACE

Nayeem Islam
The XView user interface toolkit: An approach to Object Oriented Programming in C

Mikael Wedlin
Interface between Unix and Hypercard

D L Jenkins
Supporting Intensive Processing and Interaction in a CAD Application under UNIX

L Carrico, N Guimarcaes, P Antunes
INGRID: A Graphical Tool for User Interface Construction

coffee break

Bernard Wobker, Klaus Gewald, Georg Faerber
UNIX Development in the Federal Republic of Germany

Tourist Programme

A tourist program has been arranged for those accompanying delegates, but who do not find the technical aspects as exciting as we do.

The programme includes visits to various parts of Munich, some local Castles and even a boat ride.

These can be booked, and paid for, on arrival at your hotel.

Friday 27th April
Registration from 08.00

Preliminary Technical Programme

NETWORK FILE SYSTEM	
	Ian Chapman The Design of a PC-based NFS Client from a DOS programmers perspective
	Helmut Kalb, Snoopy Schmitz NFS Compatible Archiving of Documents using WORM Discs
	Joseph Moran, Russel Sandberg, Don Coleman, Jonathon Kepecs, Bob Lyon Breaking Through the NFS Performance Barrier
coffee break	
APPLICATIONS	
	Thomas Merz HIT-Multicode
	George M Taylor An X-windows Teletext Service for UNIX
	R Koster Porting Banking Software from VMS to UNIX
lunch	
SECURITY	
	Pascal Beyls Is UNIX resistant to computer virus?
	Norbert Ondra The Administerability of UNIX Systems
	Ingo Hoffmann, Stanislaus Gefroerer Authentication using a Chipcard

Student Grants

Grants are being offered to assist students to attend the Conference. An application must be made well in advance of the Conference. A decision will be made before the event whether an application qualifies for a grant. Payment will not be made until after the Conference but the applicant will be able to proceed in the knowledge that the grant will be forthcoming.

Priority will be given to:

1. Students giving a talk at the Conference.
2. Students doing work for the EUUG or a National Group.
3. Students.
4. Other deserving cases like research students.

How to Book

A Conference Registration Booklet giving full details of the programme and events, cost, hotel booking, sightseeing programme, plus booking forms has been printed and was distributed to all EUUG Members early in January, 1990. If you would like a copy of the booklet sent you please complete the form at the foot of this page.

Costs will be as follows:

Tutorials	
Tutorial per person	Members Only £280
* Tutorial per person if Registration and payment on the door	Members only £380

Conference	
3 Day Conference if booked in advance	Members £280 Non-Members £360
*3 Day Conference if Registration and payment on the door	Members £380 Non-Members £460

*These can only be accepted if space allows.

Students will be able to obtain a 50% reduction for the Conference. If there is enough space available students can apply for a 50% reduction for attending a Tutorial.

Conference, Tutorial and Hotel Enquiries

Conference and Tutorial
Mrs Helen Gibbons
The Secretariat,
EUUG,
Owles Hall,
BUNTINGFORD,
Herts. SG9 9PL
U.K.

euug@EU.net
Tel: +44 763 73039
Fax: +44 763 73255

Conference Hotel
"Front Office Group Coordinator"
The Munich Sheraton
Arabellastrasse 6,
D-8000 Munich 81
W Germany
Tel: +49 89 92643414
Fax: +49 89 916877

Conference Registration Booklet Request Form

Please send a copy of the above booklet to:

Name .....

Company .....

.....

..... Date .....

Please complete and send to Mrs Helen Gibbons at the address above.

Advance Booking Form for Conference and Tutorials

Please complete this form and send it, with cheque or evidence of payment, to **EUUG Secretariat, Owles Hall, Buntingford, HERTS SG9 9PL, U.K** (Block Capitals please). Please note that forms sent without cheque or evidence of payment will be returned to you unregistered.

Surname ..... Usual First Name .....

Company/Organisation .....

Address .....

Country ..... Post/Zip Code .....

Telephone/Fax/Telex/Email .....

EUUG member? Yes ☐ No ☐ Student? Yes ☐ No ☐

Please read the sections on "COSTS" and remember that pre-booking saves money.

All payments must be made in pounds sterling (£)

CONFERENCE

Please reserve me a 3-day place for the Technical Sessions £ _____

TUTORIALS (members only)

Please reserve me a place for Tutorial No _____ on Monday 23rd April £ _____

Please reserve me a place for Tutorial No _____ on Tuesday 24th April £ _____

Please reserve an extra ticket at the social event £35 ☐ £ _____

Do you require vegetarian meals? Yes ☐ No ☐

EUUG

Please send me as an institutional member of EUUG via the appropriate national group Yes ☐ No ☐

TAPE

Please reserve me a copy of the Conference Tape ☐ 1600 BPI ½" £ _____

All tapes £35 each. ☐ QIC-24 ¼" £ _____

PAYMENT METHOD Total £ _____

☐ UK Cheque, Banker's Draft or Eurocheque. The cheque must be enclosed.

☐ Direct Payment. The bank advice note showing details and date of payment must be enclosed. All bank charges must be borne by you and not the EUUG—please tell the bank this. EUUG must receive the actual amount due.

☐ by VISA

☐ by ACCESS/EUROCARD/MASTER CARD

Name as it appears on the card (block capitals) .....

Address of card holder .....

.....

.....

Card Account No. ..... Date of Expiry .....

Signed ..... Date .....

The Vienna Conference report

Marco Franzen
franzen@tubsibr.uucp

Techn. Univ. Braunschweig
D-3300 Braunschweig

Marco Franzen is a student of computer science at the Technical University of Brunswick (TU Braunschweig). To support himself he started working at the Institute of Operating Systems and Computer Networks. His job ? Porting applications and system software between different flavours of Unix. When he realized several years later that he now had the means to his end, he quit this job (at least officially :-) and is now trying to finish his studies.

As I don't like hectic activity I decided to travel on Sunday. I would then have enough time to organise my accommodation and transport within Vienna before the opening cocktail on Monday evening. So it happened that I sat during the daylight of Sunday in the train. (Though travelling eastward the daylight is shorter...)

Shortly before Vienna my neighbour asked me whether I was also going to attend *The Conference*. I affirmed, but as it turned out he was a mathematician and meant another conference at another university in Vienna.

When I arrived at Vienna it was a mild summery night. Actually it was unusually warm the whole time that I was there. Since the hotel where I had reserved a room was on a nearby street and the appropriate underground railway line should be opened only a month later, I walked.

Alas, I had not realised just how long the street with my hotel was. Of course it was very long, and the hotel was at the other end. (The Austrian language is not quite the same as the German. The Austrian word for even the widest and longest street sounds to a North German as that for a narrow alley.)

So carrying my luggage became harder and harder with every step, but eventually I arrived at my... Surprise! It was more of a youth hostel than a hotel! I should have realised that due to the very low price. Though I got a single bedroom as ordered (they had also multi-bed rooms), it had no washing facility and the toilet and shower/bath rooms were down a long corridor.

But I was tired enough to be happy with what I had.

Monday

The ho(s)tel was somewhat stingy with breakfast. You had to beg for each roll separately. But at least it was cheap.

Then I walked to the main ticket-office of the public transport company. This time, without luggage, I had to walk only a short distance due to the fact that my hostel was at this end of the long street. A quick chat assertained that a weekly season-ticket was best for me, and I was amazed how cheap it was, even without any student discounts and the such.

When I thought about it, the underground railway is as old or as new as UNIX and also still under development. Public transport in Vienna is excellent. They have an underground railway,

some buses and a lot of trams. The tram net is so dense that one runs down every second street (or alley:-) in the city, and the tram lines extend out to the surrounding villages.

I really don't understand why so many people still drive cars through Vienna. The air in the town is really bad in the evening. To avoid to be impolite I must admit at this point that the drinking water in Vienna is very tasty. It seems to be spring water. (When you order a coffee in a coffee-house you get almost always a glass of water with it.)

I used my newly acquired and wonderful ticket to do my own small sightseeing tour. You can read in any travel guide-book what I saw, but I am about to tell you only what you can't find there;-). As the most noted sight lies in the pedestrian precinct (which is also reachable from the underground), I used this fact to purchase some towels there so that I could take full advantage of the shower bath of the hostel.

It was now time to go to the opening cocktail in the City Hall at the invitation of the mayor of Vienna. One good thing about EUUG events is that you don't have to worry about being wrongly dressed, so I went directly to the City Hall. (If I had registered first at the EUUG desk in the University I would have known that full dress was desired, but it didn't seem to matter anyway.)

The doorman at the entrance told me in friendly but firm tones that I could not enter without an invitation card. Not having registered I had none. But he pointed at an EUUG helper and said that there was someone distributing these cards...

The opening cocktail itself took place in a grand room with big lustres. There was a buffet area, a microphone for speeches at one end, and an organ for the solo entertainer at the other. In the speech of the day we learned that our conference was important, that the City of Vienna was a UNIX user and UUGA member, and that its City Hall was an open system, too. Teus Hagen then thanked them for the invitation and the buffet was opened. That meant that the entertainer began to play and that more and more people went into the entrance-hall. The cocktail was the first opportunity that I had to talk with the other delegates.

Tuesday

The University of Economy has a very modern building even with (a rather strong) air conditioning. There were EUUG signposts everywhere from the tram stop to the registration desk, so that it was really easy to find.

Immediately before the registration desk there were some terminals and a blackboard, grouped, constituting the conference mailing system[1]. This new service turned out very successful and was used heavily (on average one message sent and received per delegate every day). Its usage seemed to be bounded only by the number of terminals available.

At the registry you got not only your nameplate and copy of the proceedings, but also a floor plan and a daily conference newsletter. At the beginning there were also Paris t-shirts available.

I went to the tutorial that I had booked:

'Beyond 4.3BSD', given by Kirk McKusick and Michael Karels from Berkeley.

It is always pleasant to listen to people who know what they are talking about. Though I did not meet the prerequisite of being an experienced kernel hacker (I've never seen any line of UNIX source code as far as I remember – except for that fragment that shows up on some t-shirts) Amazingly I found that I could follow to some extent.

They explained the modifications done in the tahoe release of BSD4.3 and those being or to be done for BSD4.4, which is tentatively scheduled for 1990 (according to John S. Quarterman). I think there are very important modifications, not only the well known ones.

One of the most widely expected extensions is the full ISO-OSI support. In this area UCB cooperates with several other organisations, including two European (UK) universities for the X.400 services. For these the socket interface will be changed.

The other big change is for POSIX conformity. This means the termio interface from System V (with Berkeley extensions to access the variables of the old 'new tty driver'), but also new concepts of job control, group sets and signals.

1. See page 10 EUUGN Winter 1989.

But I find the restructurings of the Virtual Memory design also very important: a copy-on-write scheme shall be employed for fork(2) and mmap(2), and light-weighted processes (threads) will be implemented. The Virtual Memory and Filesystem caches will also be unified, and the (w:-)hole system shall be more secure.

There will be compatibility functions in the library that emulate the old system calls during one release, so that applications have time to migrate to the new interfaces.

At the end those present were offered a discount card for the tutors' book about 4.3BSD internals. I have meanwhile ordered a copy.

Wednesday

The first day of the conference proper. It started with a database lecture as keynote. Professor Ahmed Elmagarmid talked about heterogeneous distributed database systems, where the problem lies in the autonomy of the individual database systems.

The first session proper began with Dominic Dunlop speaking about standards. He compared the evolution of converging and diverging standards with the biological evolution with some species dying out and others popping up to fill a niche. By means of the video media he pointed out that commercial success, a formal standard, and technical quality can be completely different things. But the UNIX user can hope for a situation similar to that of the gramaphone media, since a disk record player works with records of several standards (for size and speed). A very relaxed and funny performance: I didn't count, but he held up at least a dozen of audio, video, and data media.

Lunch was in the mensa (university cafeteria), where everybody could compile her/his meal her/himself. This way one could live, for example, on salad and pastries only.

I think that the most interesting session of that day was the one about security. Two talks summed up some known possible holes that one should be aware of, the second of was especially concerned with IP networks.

Ernst Piller proposed a new login procedure where the user has to compute and enter a one-time password from a randomly chosen matrix of letters displayed by the computer and from the password proper. The computer has to compute this one-time password itself and compare it to the entered one. This way the password proper is out of the login dialogue and can't be observed by an eavesdropper; the observed one-time password will not be valid for the matrix of letters displayed at another login. (The installation and change of the password proper has still to be done without observers, but this occurs less often. Due to enhanced security at login time you may change the password proper even less often.)

There was a very interesting talk about international copyright laws by Alicia Dunbar Gronke[2].

In the evening a BOF session had been scheduled with one of the big UNIX vendor associations. Since I am not a Bird Of *that* Feather, I did not attend.

Thursday

The first talk of the morning presented much statistics about sales numbers and the like. Not my domain.

The next two lectures were on modelling. The first of them described an AI program for the diagnosis of problems with the line printer. The second modelled a set of NFS workstations and a fileserver by stacking the models of the protocols one on top of the other. This way the whole model is easier to write and understand and the submodels can be reused for other purposes.

The so-called coffee-break (which took place on all days, of course) consisted of several kinds of very good cake and either coffee in non-returnable plastic waste-cups or fruit juice in reusable bottles. All this was distributed about the exhibition so that more people took notice of the firms present.

The next session was about RISC. The first performance was by Daniel Klein. He described how he became a RISC fan from a CISC fan.

He presented much data about the code that some compilers generate for the same C program. The data showed the frequency of the generation of the individual instructions, of their membership in the classes that he had partitioned the instruction set into, and of the addressing modes generated.

2. See page 2 EUUGN Winter 1989.

From the fact that for the CISC architectures 70% of the instructions were not generated, whereas for RISC only 30% of the instructions were not generated he concluded that with CISC the most instructions were 'unusable'. The same conclusions for the addressing modes.

Though the lecture(r) was greatly applaused I did not understand why the conclusion was be so far-reaching. Couldn't it be possible that a smarter compiler had combined more of the simple instructions to the complexer ones? And there was only the static instruction distribution considered, but instructions like decrement-and-branch-if-nonzero are typically used in a loop. (Explanations and flames welcome.)

A more theoretical approach to the question was made by Kurt Judmann. His models resulted that without a pipeline the RISCs would be half as fast as the CISCs. But with a four-stage pipeline and a 60%-hit-rate cache the RISC would be four times as fast. The uniform execution time for all instructions makes the pipeline more effective or, to put it in other words, makes a shorter pipeline suffice, while software dependencies bound the benefits for very long pipelines.

The next was only a sales talk for SPARC machines.

After the lunch there was a tools session, the most interesting of which was to me the ODA talk. ODA is an ISO standard for an Office Document Architecture (and I think also a CCITT standard for an Open Document Architecture) that means a standard for representing multi-media documents. To interchange documents between different document processing systems requires converting between formats. With many formats it is easier (and requires a smaller number of translators) to convert to and from one common format. The people at the Carnegie Mellon University have created a toolkit for manipulating ODA objects that helps in writing such translators (or maybe even document processing systems). Now the good news: this CMU ODA Tool Kit shall be distributed with the next MIT X tape.

The network management session contained two interesting talks. Josef Matulka had measured the user-level performance observed by a distributed application using TCP vs. UDP with several packet lengths. Of course UDP was faster, but the impact made by the packet length (not only the magnitude but also the exact number) was amazing. Sending 1022 bytes can be very different from sending 1024, so that it may be worth looking for such anomalies when tuning an application for a given environment. The programs are available from Josef <matulka@awiwuw11.bitnet>.

The other notable talk was by Philippe Blusseau. He told us how to connect TCP/IP LANs with ISDN, that means plugging ISDN under IP instead of Ethernet or X.25 using a new router. The exact policy for channel usage and timeouts, however, should depend on the tariffs of your particular PTT.

The conference dinner was celebrated at two distinct locations far apart at the same time. In addition to the announced place, castle Miller-Aichholz, in the last minute a Heuriger evening was arranged. So you couldn't meet everyone at the dinner.

Friday

The last conference day started with a session about Graphical User Interfaces. The first paper was very Sun-specific.

Then Timothy Erhart reported on X Display Servers, those specialised microcomputers that run only X. The great advantage is the lower price compared to diskless workstations. Instead of file servers you then need compute servers. He argumented that using a diskful node as a compute server would avoid the network load on paging and swapping compared to using a diskless workstation as (compute and) display server. On the other hand compute intensive and graphics intensive applications run better on their (diskless) workstations. But one disadvantage remained unsaid: Using a specialised computer keeps you at the mercy of its manufacturer when you need a firmware update, while the X source is free. Since X is still under development this point is more important than for an ASCII terminal.

Robin Faichney expanded further his results about redirection of graphical I/O. I had gladly read his previous paper about that subject, but, alas, the Brussels proceedings are no longer available. The goal seems to be to transfer the benefits from traditional UNIX I/O redirectioning (like piping) to graphical systems (though I see already problems for curses-like applications).

The next session was about Transaction Processing. The first talk described how to make

the decision when buying hardware for database transactions: model the aimed configuration, develop a benchmark, measure with the computer configurations offered.

The last session was perhaps the most interesting. It started with Donal Daly about the Trinity College Dublin implementation of the Esprit project COMANDOS platform for the development and online management of distributed applications. They are implementing COMANDOS as a native kernel on the VAX and ns32k architectures, called OISIN. On top of it they provide a (BSD) UNIX emulation, called ROISIN.

The opposite direction of bringing UNIX and Comandos together was taken by the GUIDE project, as already published at the Portugal conference: they implement the Comandos platform on top of UNIX (System V flavour). The talk at this conference was about the performance problems with System V's semaphore interface and the work-around they used. This semaphore interface is very general as it allows operations on multiple semaphores at once, but this complexity was unneeded in this case. So they implemented a device driver that did only what they wanted. This raised the performance of the overall system by a factor of two or three, but was still too slow. So they made a pre-test on the user-level that prevents an unnecessary system call in the usual case (no collision). This made the needed breakthrough: a performance factor of several hundreds. (I would have been interested to see what gain the second step without the first would have brought. Wouldn't this suffice? I think that this is the strategy that the next version of BSD will use.)

Then the winners of the conference competition were presented. The theme had been: new commands of REAL benefit for REAL users. Notable answers were:

LEMAhelp(1) interactive preprocessor for the lettermatrix login program.

cw(1) call wife. Sends a message to your wife and tells her that you will be late for dinner.

rism(1) reduced instruction system manager. Generates on behalf of the system manager to requests such as "please increase my disk quota". It always outputs the string "No."

gethostforfree(1) wait for prompt dollars DIAGNOSTICS intended to be self-explanatory

woman(1) does the same job as the man command, but in half the time and without complaining

ssh(1) turn off the workstation fan

open_register_window(2) for 40+ Mips chips, used to cool the chip down

The winners were:

ittm(1) invent time travel machine. Enables the user to catch up with all late projects' (prize: EUUG/Vienna watch with slanted face)

gmb(1) Get Me a Beer. Talks to bd(8), the beer daemon (prize: beer mug)

rmspeak(1) remove the speaker and slides from the podium.
-r Removes the memory of the speaker from the delegates' mind (prize: One of Dominic Dunlops phono media: a sirupy music record)

The last official talk was John S. Quarterman's. He compared social aspects of EUUG and USENIX, their conferences and their networks.

Call for Papers

The Fifteenth JUS UNIX Symposium

Tokyo July 9-11, 1990

Papers are sought in all areas of UNIX-related research and development for the technical program of the 1990 the fifteenth JUS UNIX Symposium. Papers which are accepted for the conference will be published in the conference proceedings and shall be presented during the two days of technical sessions at the conference.

Topics of Interest

Appropriate topics for presentation include, but are not limited to:

- Internationalization, Localization (Japanization), Standardization,
- Network, Distributed processing, Parallel processing,
- Software development environment, Language,
- Business Applications,
- Security, Reliability,
- User Interface,
- Implementation, Porting,
- System management.

Schedule for Contributed Papers

All submissions will be considered - however, papers detailing new and interesting work will be regarded much more favorably than thinly disguised product announcements. The fifteenth JUS UNIX Symposium is requiring that **full papers** written in Japanese or English to be submitted. The full paper should include a 100-300 word abstract, a discussion of how the paper relates to other work, illustrative figures (where appropriate), and citations to relevant literatures. The full papers should contain on the order of 8-12 pages of single spaced typeset materials. All final papers must be submitted in a camera-ready format or electronic format (TEX or troff if possible).

Please submit full paper as soon as possible, and mail one hard-copy or one electronic-copy to the addresses below.

The final deadline for receipt of submissions is **April 10, 1990**; papers received after this deadline will not be considered. Notification of acceptance or rejection will be made by **May 1, 1990**. Final camera-ready papers are due by **May 31, 1990**.

To submit a paper or request additional information, please contact:

Japan Unix Society,
Hayabusa-cho 1-12-505
Chiyoda-ku, Tokyo 102
Japan.

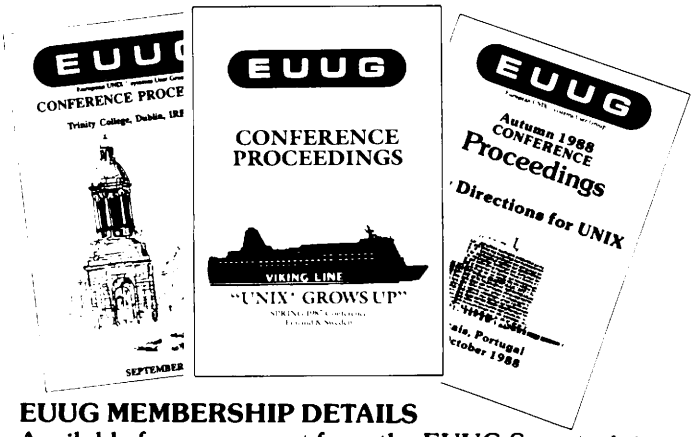
Tel: +81 (3) 234-5058
Fax: +81 (3) 234-9454
E-mail: sympo15th@jus.or.jp

EUUG

European UNIX® systems User Group

ADDITIONAL PUBLICATIONS AVAILABLE FROM EUUG

Title		Cost
EUUG—CONFERENCE—PROCEEDINGS		
Autumn 1983	— Dublin	£2.00
Spring 1984	— Nijmegen	£5.00
Autumn 1984	— Cambridge	£5.00
Spring 1985	— Paris	£5.00
Autumn 1985	— Copenhagen	£10.00
Spring 1987	— Finland/Sweden	£20.00
Autumn 1987	— Dublin	£20.00
Spring 1988	— London	£20.00
Autumn 1989	— Vienna	£20.00
European E-Mail Directory, Available January approx.		£18.00
4.3BSD MANUAL	— User's Manual Set	£25.00
	Programmer's Set	£25.00
	System Manager's Manual	£10.00
Language C—Standard Proposal		£8.00
USENIX Conference Proceedings C++ —Santa Fe, 1987		£30.00
A Directory of Electronic Mail !%@:: Addressing & Networking (Frey & Adams)		£20.00



EUUG MEMBERSHIP DETAILS
Available free on request from the EUUG Secretariat.

UKUUG PROCEEDINGS
Winter 1988 –UKNet Meeting, Canterbury **£10.00**

If you wish to order any of the above publications, or receive details of EUUG Membership, please use the order form below and return it to the EUUG Secretariat, Owles Hall, Owles Lane, Buntingford, Herts. SG9 9PL, U.K. If you are ordering publications then your remittance should be enclosed by cheque, bank draft etc. made payable to EUUG.

ORDER FORM

Name _____

Address _____

_____

_____

_____

Phone _____

Network Address _____

The prices shown include surface postal charges. All payments must be in Sterling by means of a cheque drawn on a UK bank, or a Eurocheque, or instruction to charge your VISA card (quoting the number, the date of expiry, the name of the card holder, and the address which VISA use when corresponding with you).

Please send me _____ no. copy/ies of the following:

_____

_____

I would like to receive membership details of the EUUG YES/NO

User's Manual Set (3 Volumes) at **£25.00/set** = £ _____
Programmer's Manual Set (3 volumes) at **£25.00/set** = £ _____
System Manager's Manual (1 volume) at **£10.00/set** = £ _____
Total £ _____

We accept
Visa, Access, Euro Card, Master Card, UK cheques, and Euro cheques.

Please complete name and address of credit card holder if different from name and address on order form.

NAME _____

ADDRESS _____

_____

_____

Expiry Date: _____

Credit Card No:

I enclose my cheque, bank draft etc., in the sum of £ _____ and understand a receipt will be sent.

Make your cheque payable to 'EUUG' and post it with this order form

To: EUUG Secretariat, Owles Hall, Owles Lane, Buntingford, Herts. SG9 9PL, United Kingdom.
Tel: Royston (0763) 73039 +44 763 73039
Fax: Royston (0763) 73255 +44 763 73255

For BSD manuals please sign Manual Reproduction Authorisation Notice overleaf.

This page may be photocopied for use.

EUUG Executive Report

Helen Gibbons
euug@EU.net

The European UNIX systems User Group
Owles Hall
Buntingford
United Kingdom

Helen Gibbons is the business manager of the EUUG and is contactable at the EUUG secretariat.

The EUUG Spring Conference, which will be held at the Sheraton Hotel in Munich from 23rd to 27th April, 1990, will provide a comprehensive UNIX update with presentations by leading personalities in the UNIX field from Europe, the Far East and the U.S.A.

A truly international event, it will commence with two days of Tutorials (23rd and 24th April) covering Programming with OSF/Motif; Advanced Network Programming, Chorus; Project Athena; 4.3 BSD Internals and System V - Kernel Structure and Flow.

Additional lectures of general interest - e.g. UNIX Development in the Federal Republic of Germany - will also be included as well as such highly topical subjects such as the ability of UNIX to resist a computer virus.

The event will be supported by a comprehensive Technical Exhibition.

Full details have been sent out by post to all members and appear again elsewhere in this newsletter. Early booking is recommended.

The policy continues of running two major conferences each year. The second one in 1990 will be held at the Nice Acropolis, Nice, France from 22nd to 26th October. Suggested subjects are Software Management for large projects, OSI and OSI application on a UNIX platform, System Administration in a Heterogeneous environment, Security and Audit, UNIX in non-English speaking environments and User Interface Management Systems. A Call for Papers has already gone out and the submission deadline is 30th April, 1990.

The EUUG Executive Committee held an administrative meeting on the 3rd December, 1989 in Amsterdam and a further meeting is planned for the 5th February, 1990 in London. The committee, as is its normal practice, carefully reviewed the EUUG accounts, and was pleased to commend the Secretariat for its good financial management. The financial situation of the EUUG remains healthy with adequate reserves and ways are constantly being sought to maximise their investment.

The Secretariat at Owles Hall is currently seeking a UNIX Operator to join the team at Owles Hall in Buntingford, Hertfordshire. The job would be one

of great interest to a member of the UNIX community as it will carry with it the chance to attend EUUG meetings and conferences and to take a really active part in the European Network. As the Secretariat at present has no sophisticated computer equipment, the new operator will have the chance to start from scratch in setting up a suitable system for the EUUG, and will be expected to act in an advisory capacity as well as running the day to day operations, liaising with the network and undertaking complete setting for newsletters and brochures. The office at Owles Hall is small and informal, so someone is being sought who will fit in well with the present cheerful team.

Donnalyn Frey as been engaged to do public relations work for the EUUG in the USA with the objective of increasing the awareness and understanding of the EUUG and EUnet in the States and making them aware of EUUG conferences in particular.

The EUUG has been talking to possible new groups in Czechoslovakia, Turkey, The Soviet Union*, Greece and Switzerland and has also made presentations in Spain. The number of Groups presently within membership is 16 representing about 4,000 members.

A meeting of the backbone managers has recently been held in Nice at which Mr Yves Devillers was confirmed as the Chairman and Peter Houlder was appointed as Treasurer. EUnet had also been represented at an informal meeting of the Groups Résaux IP Européane. Mr Dominic Dunlop will be representing the EUUG at the forthcoming ISO meeting.

Sun Microsystems has donated a Sun - 4/280 server to the EUUG to operate as the international backbone computer. The total throughput of MCSUN is more than 3 gigabytes a month. The transport of information happens though public and private switched networks, leased lines and the public telephone network. Almost all international news bulletin traffic is handled by MCSUN, which amounts to 90,000 bulletins each month in more than 600 active categories. At any given time more than 40,000 bulletins can be stored and the mail service is accessed by well over 8,000 systems.

The Directory of Electronic Mail !%@:: Addressing and Networking by Donnalyn Frey and Rick Adams, as reviewed on page 70 of the 1989 Winter newsletter, is now obtainable from the EUUG Secretariat at a cost of £20 including postage and packing.

Current Executive Committee Members are:

Chairman	Teus Hagen	+31 77 594082	teus@oce.nl
Vice Chairman	Michel Gien	+33 1 30 570022	mg@chorus.fr
Finance	Nigel Martin	+44 1 251 2128	nigel@inset.co.uk
Public Relations	Kim Biel-Nielsen	+45 42 894999	kbn@kontiki.dk
Conferences	Dr Ernst Janich	+49 731 1526 464	janich@nixulm.de
Conferences	Neil Todd	+44 491 671964	neil@gid.co.uk
Networks	Daniel Karrenberg	+31 20 5924112	dfk@cwi.nl
Publications	Philip Peake	+33 1 69 07 82 47	philip@vogon.cetia.fr
	Johan Helsingius	+358 0 427 632	julf@penet.fi
	Norman Hull	+353 503 31745	norman@q2rs.ie
	Frances Brazier	+31 20 5483885	frances@psy.vu.nl
Business Manager	Helen Gibbons	+44 763 73039	euug@EU.net

Elections will be held at the EUUG meetings in Munich when the Governing Board meets on the 21st and 22nd April, 1990.

* Elsewhere in this newsletter is a report on the history of UNIX in the Soviet Union.

UNIX SYSTEM ADMINISTRATOR

The European UNIX systems User Group requires a suitably qualified computer expert to set up and operate a computer system using UNIX at its Secretariat at Buntingford in Hertfordshire, England.

The work will include:

- selection, procurement and installation of the system
- establish the machine as part of the European network EUnet
- daily administration of the system
- provision of an advisory service to the national groups
- training and support of the other Owles Hall staff in UNIX office administration and software packages
- modification and enhancement of the Informix-based conference registration software
- specification, analysis, design and implementation of other productivity tools for the EUUG
- typesetting, by use of troff, newsletters and brochures
- establish and maintain a database of group members with associated label and report software
- recommend and set up software accounting systems.

The successful candidate must be a team player, but also be able to operate with initiative in a largely unsupervised manner. The machine and system will be providing an European service and as such forms a very visible element of the EUUG administration. The person chosen must also be able to fit into a small, busy informal office.

Willingness to travel to visit national groups and to remain on-site during EUUG Conferences is essential.

Salary negotiable.

Applications including full CV should be sent to:

Mrs H. M. W. Gibbons
EUUG Secretariat
Owles Hall
Buntingford
Herts SG9 9PL
England

Tel: +44 763 73039
Fax: +44 763 73255

All applications will be dealt with in strictest confidence.

USENIX Association News for EUUG Members

Donnalyn Frey
donnalyn@frey.com

Frey Communications
Fairfax, VA USA

Donnalyn is the USENIX Association Press Liaison. She provides members of the press, USENIX Association members, and EUUG members with information on the activities of the USENIX Association.

1990 Winter USENIX Association Conference

The 1990 Winter USENIX Conference in Washington, DC drew 1,517 attendees.

Dr. James E. Tomayko, of the Software Engineering Institute, Carnegie Mellon University, delivered an entertaining and informative keynote address on "NASA's Manned Spacecraft Computers." Tracing NASA's sometimes hilarious meanderings through 13-bit and 32-bit words; from Burroughs to IBM; and concluding with the information that NASA had selected PS/2s running AIX and the DACS Ada compiler for the space station project, Dr. Tomayko amused over a thousand USENIX conference attendees. He concluded his talk by remarking that with NASA's decision in favor of AIX, UNIX was now truly "out of this world."

Concurrent Sessions

These new experimental sessions track provided attendees with a venue to exchange information in an informal setting. Talks were given by Andrew Hume on make and regular expressions; Mary Seabrook on getting the most from support; John Quarterman on surviving in networkland; Richard Stevens on NAWK – a new version of AWK; Tom Christiansen on PERL - a system administration language; and Eric Allman, Evi Nemeth and Mike O'Dell on submitting and presenting papers at USENIX.

Ethics Session

The conference also featured a special session on Ethics in the Computer Industry, moderated by Rob Kolstad. The panel included a communications attorney, a Chief Executive Officer, and an ethicist.

Best Student Paper

"Disk Scheduling Revisited" by Margo Seltzer, Peter Chon and John Ousterhout of the University of California at Berkeley received the USENIX Association's Best Student Paper Award.

The Terminal Room at the Conference

The USENIX Association hosted a Terminal Room which had modems for a dialout connection, as well as a T-1 Internet connection, provided by UUNET Communications Services. Conference attendees could log onto their home or work systems to read their mail and contact other UNIX users directly from the conference. All equipment was donated by various sponsors.

Facilities were also available to create cartridge tapes of GNU and public domain software. During the conference, electronic mail was sent to attendees with the address Your_Name@conference.usenix.org. The terminal room was staffed each day of the conference by USENIX Association volunteers and ran almost around the clock.

The 1990 Summer USENIX Association Conference

The 1990 Summer conference will be held on June 11-15, 1990 at the Marriott Hotel in Anaheim, California, home of Disneyland. The conference will emphasise retrospectives, analyses of tradeoffs, and critical thinking, with the theme of: "Beyond Mere Data: Perspective, Insight, Understanding" Papers at the conference will cover subjects such as:

- software release systems
- user interfaces, windowing, and graphics
- compilers, debuggers, tools, and runtime issues
- file systems
- distributed systems
- UNIX kernel approaches
- fault-tolerancy, reliability, and security
- computer architectures that stretch UNIX.

1990 C++ Conference

The 1990 C++ Conference will be held in San Francisco, California on April 9-11. It will be devoted exclusively to C++ and will offer an intensive three day program bringing together in-depth tutorials along with technical sessions covering a broad spectrum of work.

1990 USENIX Workshops

Definite dates have been selected for workshops in UNIX Security and Mach (please refer to the Calls for Papers published in this issue of the EUUGN). Workshops are also planned for Large Installation Systems Administration, Software Development Environments in UNIX, and Standards.

Further Information on Conferences and Workshops

If you need further information on registering for upcoming USENIX Association conferences or workshops, contact the USENIX Conference Office at:

22672 Lambert Street
Suite 613
El Toro
CA 92630
USA

Email to:

judy@usenix.org or
{uunet,ucbvax}!usenix!judy

Tel: +1 714 588 8649
Fax: +1 714 588 9706

USENIX UNIX SECURITY WORKSHOP

Portland, Oregon, USA
August 27-28, 1990

The second USENIX UNIX Security Workshop is to be held in Portland, OR on Monday and Tuesday, August 27th and 28th, 1990. Matt Bishop will again be chairing this workshop. It will bring together researchers in computer security dealing with UNIX and system administrators trying to use UNIX in environments where protection and security are of vital importance. It is intended to provide an environment where researchers can discuss their latest results, where researchers and practitioners can discuss the applicability of those results to practical problems, and where system administrators can share their unique solutions and techniques for dealing with problems. The topics covered by this workshop include both theoretical topics and everyday problems. We expect each participant to present unique attributes of his/her environment and/or research and contribute a short (five minute) discussion (and paper) detailing some result or solution from their environment or work.

Some topics to be considered include: modeling the UNIX operating system theoretically, password security (password file integrity, enforcing choice of a safe password, spotting and handling crackers), network security (problems arising from logins over an unprotected ethernet, containing a break-in to one machine in a networked environment), security in a distributed system or environment, file system security (auditing packages, security in an NFS environment), computer worms, viruses, and other phenomena, new designs to obtain C-level (or better) certification, making existing UNIX systems more secure, and locating and fixing UNIX security problems, or any other problem or contribution that participants make.

WORKSHOP FORMAT

This gathering will follow a "workshop" format rather than a "paper presentation" format. Submit a one or two page summary describing a problem and, if you have one, a solution or if not, a possible approach or approaches which looked promising but failed (or which you have not yet tried). Also, please include with your submission a set of five (or so) topics that you'd like to hear about. It is possible that some participants will not present their papers at this workshop.

The workshop chairman will collate the papers to schedule sessions which have appropriate audiences. It is anticipated that some sessions will include all participants; some may require breaking into smaller groups. Send your submissions to the address below by May 22, 1990.

FOR FURTHER INFORMATION CONTACT:

Matt Bishop
Department of Mathematics and Computer Science,
Bradley Hall
Dartmouth College
Hanover, NH 03755
Phone: (603) 646-3267
UUCP: decvax!dartvax!Matt.Bishop
Internet: Matt.Bishop@dartmouth.edu

FOR REGISTRATION DETAILS CONTACT:

USENIX Conference Office
22672 Lambert St., Suite 613
El Toro, CA 92630
Tel: 714-588-8649
EMAIL: judy@usenix.org

Call for Papers: USENIX Mach Workshop

Burlington, Vermont USA
October 4-5, 1990

The use of Mach in what has traditionally been the Unix community is growing as DARPA and OSF increase their Mach-related activies and more vendors are supporting Mach on a variety of platforms. Because Mach itself is changing rapidly and there has not been any convenient mechanism for communication among developers, the Usenix Association is pleased to sponsor its first Mach workshop, at which researchers, vendors, and users can share results of Mach-related development work and status reports on work-in-progress.

Although tutorials may be offered in conjunction with the workshop, the workshop itself will be oriented primarily towards those who have actually worked with Mach or have done Mach-based applications development, and will not itself be tutorial in nature. The program will consist largely of refereed papers and panel discussions.

Abstracts of 350-700 words should be sent to Melinda Shore at the address below (those submitting hardcopy abstracts should send five copies). The deadline for submissions is June 22, 1990. All submissions will be acknowledged. Authors will be notified by July 20, 1990, and full papers will be required by August 27, 1990.

For further information about the workshop, contact the program chair:

Melinda Shore
mt Xinu
2560 Ninth St., Suite 312
Berkeley, CA 94710
(415)644-0146
shore@mtxinu.com

Program Committee:

Alan Langerman, Encore Computer Corporation
Douglas Orr, Carnegie-Mellon University
Homayoon Tajalli, Trusted Information Systems
Avadis Tevanian, NeXT, Inc.

For information about registration, contact:

USENIX Conference Office
22672 Lambert Street, Suite 613
El Toro, CA 92630
Tel: 714-588-8649
FAX: 714-588-9706
EMAIL: judy@usenix.org

USENIX ASSSOCIATION

ORDER FORM

CONFERENCE & WORKSHOP PROCEEDINGS

Qty	Proceeding			Add'l Unit Price	Total	Foreign Postage	Total
___	Washington DC Conference	Jan.	'90	$25	$_____	$15	$_____
___	Graphics Workshop V	Nov.	'89	18	$_____	10	$_____
___	Distributed and Multiprocessor Workshop	Oct.	'89	30	$_____	20	$_____
___	Large Installation Sys. Admin. III Workshop	Sept.	'89	13	$_____	9	$_____
___	Baltimore Conference	June	'89	20	$_____	15	$_____
___	UNIX Transaction Processing Workshop	May	'89	12	$_____	8	$_____
___	Software Management Workshop	Apr.	'89	20	$_____	15	$_____
___	San Diego Conference	Feb.	'89	30	$_____	20	$_____
___	C++ Conference	Oct.	'88	30	$_____	20	$_____
___	C++ Workshop	Nov.	'87	30	$_____	20	$_____
___	Graphics Workshop IV	Oct.	'87	10	$_____	15	$_____
___	Washington DC Conference	Jan.	'87	10	$_____	20	$_____
___	Graphics Workshop III	Dec.	'86	10	$_____	15	$_____

Total price of Proceedings _____
Calif. residents only add applicable sales tax _____
Total foreign postage _____
Total enclosed _____

* Discounts are available for bulk orders. Please inquire.

PAYMENT OPTIONS

☐ Check enclosed payable to USENIX Association. ☐ Purchase order enclosed.
☐ Please charge my: ☐ Visa ☐ MasterCard

Account # _____ Exp.Date _____

Signature _____

Overseas? Please make your payment in U.S. currency by one of the following:
* Charge (Visa, MasterCard, or foreign equivalent)
* International postal money order
* Check - issued by a local branch of a U.S. Bank

1/90

Shipping Information

Orders to U.S. and Canada are shipped via printed matter. Please allow 2 weeks for delivery. Foreign orders are shipped via air printed matter. Please allow 10-14 days for delivery. Shipment by UPS, first class, or airmail is available upon request.

Ship to: _____

Please mail this order form to: USENIX Association
2560 Ninth Street
Suite 215
Berkeley, CA 94710

Setting up a USSR UUG

Dr Vladas Leonas

Interquadro
4, 2-nd Novopodmoscovny per.,
Moscow, 125130
USSR

V. V Leonas was born on 23rd June 1956 in Moscow. After graduating from school in 1973 he entered the Moscow Aviation Institute, Facility of Applied Mathematics. In 1980 he graduated with a specialisation in systems software and entered a postgraduate course at the Institute of Mathematics and Cybernetics of the Lithuanian AS. He presented a Candidate Thesis in 1984 (a Candidate degree is equivalent to a PhD). From then until 1988 he was Head of the Operating Systems Laboratory at the Program Systems Institute of the AS of the USSR. In 1987 he received the Academic Status of a Senior Researcher.

Since 1988 at the Soviet-French-Italian joint Venture Interquadro, he was at first Portable Operating Systems Department Manager, later Scientific Research Director.

He is the author of approximately 50 papers and articles, and translator (from English into Russian) of several monographs.

He is married with one daughter (14 years old), and has a big Newfoundland dog.

UNIX™ in the USSR: 1980-1990

The first time that the word UNIX was uttered in the USSR was somewhere in 1979-1980. By the end of 1980 this word had become well known to a small group of specialists from different institutions, who formed a semi-formal/semi-informal Portable Operating Systems SIG (or more correctly - Machine Independent Operating Systems SIG) under the aegis of the State Committee of the USSR for Science and Technology. This group included mainly those who used SM-4 computers (a Soviet made mini-computer, fully compatible with the DEC PDP-11/40). This was not some form of alliance between institutions, but rather cooperation for information exchange between programmers.

From rather stochastic meetings and contacts during 1980-1981 this SIG managed, by 1982, to organise regular (twice monthly) seminars at the Advances Training Institute of the Ministry of the Automobile Industry of the USSR. By 1983 two different operating systems, both compatible with UNIX Version 6, had been implemented; this was a result of close cooperation and information exchange in the USSR. One of these two operating systems was called INMOS (a Russian acronym for Instrumental Portable Operating System) and was implemented by a team from the Institute of Electronic Control Computers, which later moved to the newly founded (in 1983) Institute for problems of Informatics of the AS of the USSR. The other version was called NMOS (a Russian acronym for Machine Independent Operating System) and was implemented at the above mentioned Advanced Training Institute for the Ministry of the Automobile Industry of the USSR.

In 1983 the first UNIX training courses in the USSR were started at the Advanced Training

™ UNIX is trademark of AT&T in the USA and other countries.

Institute of the Ministry of the Automobile Industry of the USSR. During the first teaching year there were two streams of students (2 and 3 groups respectively), who received an intensive full day 6 week MNOS course (including a course on C language programming). These training courses are still running, although the students receive a slightly different course, which lasts 8 weeks and is based on another implementation of the operating system. This is called DEMOS (a Russian acronym for Dialogin Common Portable Operating System). DEMOS is compatible with UNIX version 7 and is the result of joint development between the following institutions:

- Institute of Atomic Energy named after I. V. Kurchatov,
- Advanced Training Institute of the Ministry of the Automobile Industry of the USSR
- Scientific-Production Union Tsentrprogrammsystem.

The current version of INMOS is also compatible with UNIX version 7.

Between 1984 and 1987 a lot of work was completed. For example, in the Program Systems Institute of the AS of the USSR alone the following were designed and implemented in this period:

- MicroPROLOG Programming System for the MNOS environment.
- Real-Time version of MNOS.
- Experimental version of the operating system MICROS (for a LSI-11 compatible computer, equipped with two 8" floppy disks and 1Mbyte "electronic" (semiconductor) disk – ie a RAM disk but no hard disk).
- New methods of increasing operating systems portability.

Meanwhile the number of users of UNIX-like operating systems was growing, and first publications on, for example, use of CAD systems in the UNIX-like environment for the design of agricultural machines, or for office automation, or process control began to appear.

As regards Russian language publications connected with the UNIX operating system and the C programming language, it is necessary to admit, that such publications first appeared in 1982-1983 and began to grow in quantity rather rapidly. 1984 was the year when the first western monograph (connected with UNIX and C) was published in Russian:

1984 H Lorin, H M Deitel "Operating Systems" (Addison-Wesley, 1981), 20,000 copies.

1985 M Dahmke "Microcomputer Operating Systems" (McGraw-Hill, 1982), 20,000 copies.

1985 P Calingaert "Operating Systems Elements" (Prentice-Hall, 1982), 25,000 copies.

1985 B W Kernighan, D M Ritchie "The C Programming Language" (Prentice-Hall, 1978) under one cover with A R Feurer "The C Puzzle Book" (Prentice-Hall, 1982), 15,000 copies.

1985 R Fauthier "Using the UNIX System" (Prentice-Hall, 1981), 30,000 copies.

1985 K Christian "The UNIX Operating System" (John Wiley & Sons, 1983), 24,000 copies.

1985 H L Helms "Computer Language Reference Guide" (Sams, 1984), 15,000 copies.

1986 M Banahan, A Rutter "UNIX the Book" (John Wiley & Sons, 1982), 15,000 copies.

1987 R Thomas, J Yates "A User Guide to the UNIX System" (McGraw-Hill, 1982), 11,000 copies.

1986 S H Kaiser "The Design of Operating Systems for Small Computer Systems" (John Wiley & Sons, 1983), 50,000 copies.

1986 S R Bourne "The UNIX System" (Addison-Wesley, 1983), 25,000 copies.

1987 H M Deitel "An Introduction to Operating Systems" (Addison-Wesley, 1984), 30,000 copies.

1987 P J Brown "Starting with UNIX" (Addison-Wesley, 1984), 20,000 copies.

1988 M I Bolsky "The C Programmer's Handbook" (Prentice-Hall, 1985), 140,000 copies.

1988 D W Topham, H V Truong "UNIX and Xenix. A Step-by-Step Guide" (Prentice-Hall, 1985), 40,000 copies.

1988 M Waite, S Prata, D Martin "C Primer Plus" (Sams, 1984), 75,000 copies.

1988 R E Berry, B A E Meekings "A Book on C" (Macmillan, 1984), 12,000 copies.

1989 A R Feurer, N H Gehani "Comparing and Assessing Programming Languages Ada, C, and Pascal" (Prentice-Hall, 1984), 50,000 copies.

1989 M R M Dunsmuir, G J Davies "Programming the UNIX System" (Macmillan, 1985), 30,000 copies.

UNIX Today

It is not a simple task to give today's picture of UNIX in the USSR because of the large number of places where there are experienced UNIX users or even local informal UNIX User Groups. Such places are, for example, Moscow, Leningrad, Kiev, Riga, Novosibirsk, Kalinin, Zhaporozhje, Odessa, Kazan, to name but a few.

In order to give the reader just a small impression of the current picture the last part of this paper is devoted to the description of the work of the main (but not all!) well known teams in the USSR.

At the Interbranch Scientific-Production Union Electronmash (which includes the Institute of Electronic Control Computers) much attention is devoted to the development of portable software. As a result of this all models of the SM line of computers are equipped with the UNIX-like operating system (specially designed and implemented) and the rather wide spectrum of application software. At the present time the problem of creation of a unified portable operating environment for all models of the SM line computers (those already existing, and those designed for future production) is under heavily discussion.

The Scientific-Production Union Tsentrprogrammsystem is involved in the development and porting of UNIX-like operating systems for Soviet microcomputers based on the Intel 8088/86/286/386 family of microprocessors, including development and implementation of device drivers and different application software packages (software tools, database management systems, network software).

The organisation Mobilnost (which means Portability) was founded in 1989. This is the leading organisation in the USSR for the implementation of the Special Purpose State Program for Machine Indepedent Operating Systems and Portable Software, whose aim is to lead and coordinate all the works in the framework of the above mentioned Special Purpose State Program, including development, implementation, training and maintenance of the systems and applications software.

Specialists at the Institute for Problems of Cybernetics of the AS of the USSR are developing a portable operating system CLOS, based on the idea closely connected with the concept of object oriented programming. In order to provide UNIX compatibility most of the UNIX system calls are emulated in CLOS. For the same reason the standard I/O library and the command language interpreter (the shell) are implemented under CLOS. The first version of CLOS was implemented for a 16 bit DEC compatible computer. At the present time the first version of CLOS is being ported on to a MC86020 based workstation.

A lot of interesting work is being done at the Institute for Problems of Informatics of the AS of the USSR. Among this is work on removing device drivers from kernel address space (for 16 bit computers with a small address space), a bi-processor version of the kernel of the UNIX-like operating system (for a bi-processor computer, in which one of the processors works as a specialised file system processor), relation database management system, and many others, such as; building of modern user interface systems with PC like X Windows Terminals instead of normal terminals. At the present time a new POSIX conforming operating system for a Intel 80386 based computer is under development.

The Interquadro joint venture specialises in building turn-key systems for different area, such as office automation, CAD/CAM, process control, etc. As a component for such systems Interquadro uses the UTEC-32 family of MC68010 based computers with the QUIX operating system which was jointly developed by Interquadro and the French company Aniral UTEC Informatique International SA. QUIX is compatible with UNIX version 7, but is much faster. Different applications software packages for QUIX were also developed and implemented at Interquadro.

In concluding this paper it is necessary to admit that the interest in UNIX and UNIX-like operating systems in the USSR is growing rapidly.

Association Française des Utilisateurs d'UNIX® et des Systèmes Ouverts

CONVENTION UNIX 90

26 to 30 March 1990
CNIT Paris-La Défense

The Association Française des Utilisateurs d'Unix et des Systèmes Ouverts (AFUU), in cooperation with the Bureau International des Relations Publiques (BIRP), is organizing the CONVENTION UNIX 90 at CNIT Paris-La Défense.

CNIT, at the very heart of La Défense, has been restructured, and beneath its single-span vault, the largest in Europe, now houses the enlarged European Business Center. It will allow us to host, in splendid style, more than 150 exhibitors and a programme of seminars and conferences of international standing.

Special effort has been made to make all the conference talks accessible to our European neighbours; thanks to a service of quality simultaneous translation from French to English and English to French.

We are especially pleased to announce this year the presence and the active support of the EUUG in this event: Norman Hull and Daniel Karrenberg will present a demonstration of EUnet on a stand which is entirely taken up by the EUUG. This, together with other activities, will allow our visitors to get to know the organisation and activities of our European federation.

Everything is ready so that this meeting, a truely splendid platform of the up to date offerings in open systems and the UNIX environment, responds to the needs to of the D.P. Managers and of all those interested in strategic computing.

General Organisation of Convention UNIX

Exhibition:	of 27 to 30 March 1990 Open: 9:30-18:00 On 30 March: closes at 16:00.	Hall d'exposition of CNIT
Seminars:	26 and 27 March 1990 Times: 9:00-12:00, 14:00-17:00	4 rooms in parallel
Conference:	28, 29, 30 March 1990 Open on 28th: 10:00-12:00, 14h-17h30 29th and 30th: 9:00-12:30, 14:00-17:30	2 rooms in parallel

For further information on registration contact:

B.I.R.P
25, rue d'Astorg
75008 Paris
FRANCE

Mrs Agnès Lambert
Tel: +33 1 47 42 20 21
Telex : 643 982
Fax: +33 1 47 42 75 68

The exhibition is free on production of an admission card.

Prices

Tutorials	Members of AFUU and EUUG	1000 FF
	Non-members	1500 FF
Conference	Members of AFUU and EUUG and universities	750 FF / 1 day 1500 FF / 3 days
	Non members	1250 FF / 1 day 2500 FF / 3 days
	Students	250 FF / 1 day 500 FF / 3 days

The above prices will be doubled on March 16th.

Tutorials On Monday March 26

SENDMAIL – Mr Yves Devillers (Inria)

Sendmail mechanisms. RFC 822/uucp/domains addressing. Models for internetwork mail router. Detailed study of different sendmail.cf integration and debugging. Sendmail versions. The "bind" name server, its integration into sendmail. IDA-sendmail. (This tutorial is intended for system administrators having good knowledge of sendmail).

Sessions 1 and 2, full day tutorial: 9.00-12.00, 14.00-17.00

The Postscript Language – Mr Gilles Dauphin (Télécom Paris)

PostScript is a description page language, result of many research years. It became a standard in the field of laser printing. The tutorial is an introduction to PostScript: syntactic elements, stacks, arithmetic operations, graphic operations, objects, variables, procedures, character fonts, PostScript programming.

Session 3, half-day tutorial: 9.00-12.00

UNIX Device Drivers – Mr Jean-Marc Barreteau (Convex)

This seminar will take place after a presentation of the UNIX operating system. Such notions as special files, system call, inodes, U structure, will not be reviewed. However, they will be used extensively during this half-day seminar. First, general structure and operation will be presented. Then individual functions making up a driver will be introduced, with their sequencing during an I/O operation. Furthermore, integration of a driver into an operating system will be discussed, highlighting those system kernel functions which are to be used by the driver. At the end of the first part of the seminar the system calls (open, read, write, ioctl, close) will be analysed, and a sample pseudo-device will be written. The second part will be devoted to an in-depth analysis of an existing TTY driver and to an implementation of some supplementary functions for this driver. Finally, the concept of a pseudo-driver will be presented.

Session 4, half-day tutorial: 14.00-17.00

OSF/1, The Programming Environment For The Future – Alan Lehotsky and David Kirschen (OSF, Cambridge)

This seminar provides a close look at the components of the OSF/1 operating system. Integrating several advanced operating system technologies, the OSF/1 offering is based on the Mach kernel from Carnegie Mellon University. The program provides an overview of the architecture and features of the OSF/1 operating system, including its compatibility with other Unix-based systems. At the completion, the attendees will know at a high level, how to port existing applications to OSF/1, how to write applications using the special features of OSF/1, and what the overall features of OSF/1 are with regard to architecture, standards, application portability, and compatibility. The OSF/1 seminar is geared toward a wide technical audience. Programming experience and familiarity with the Unix operating system is recommended.

Sessions 5 and 6, full day tutorial: 9.00-12.00, 14.00-17.00

Implementing And Using Windowing With UNIX – Mr Daniel Dardailler (Inria)

This presentation describes different windowing systems under Unix as seen from the user. The presentation is aimed mainly at application developers; problems related to a new windowing system implementation will be dealt with their respective advantages ans drawbacks. Then, we will give detailed description of two windowing systems which are currently the most popular: X11 and News. Finally, the application portability problem will be dealt with.

Sessions 7 and 8, full day tutorial: 9.00-12.00, 14.00-17.00

Tutorials On Tuesday March 27

UNIX, Internal Mechanisms – Mr François Armand (Chorus Systèmes)

This one-day seminar will help you become familiar with the internal mechanisms of the Unix System V Kernel. To take full advantage of the seminar, good knowledge of Unix System (commands and programming) is required. Major system components (file handling, process management) will be presented, as well as their inter-relationship. Presentation of each subject will combine static aspects (concepts and data structures) with dynamic ones (algorithms, actions performed on different system calls to the kernel). Particular attention will be paid to those oustanding Unix features. Also, comparison will be made related UNIX 4.3.BSD mechanisms.

Programme of the day: General concepts, the concept of a process; naming. Objects bound to a process. General structure of the kernel. File handling, input-output; file types. Magnetic media structure. Associated tables. Unix Cache: its role, its management, its advantages and drawbacks. Description and implementation of system calls. Particular case: pipes. Structure of a driver. Terminal management. Process management: process descriptor. Concept of a text. Memory management. Synchronisation, mutual exclusion. Sequencing, priority handling. System calls implementation. Creating and killing a process. Concepts of swapping and memory pagination. Signal handling.

Sessions 9 and 10, full day tutorial: 9.00-12.00, 14.00-17.00

C++ Language – Mr Frédéric Lung (Consultant)

C++ is an object-oriented language developed by Bjarne Stroustrup at Bell Laboratories (AT&T) in the early eighties. C++ is a superset of the C Language; it introduces type verification and abstract data-type declaration features and supports object-oriented programming. The seminar will subsequently present all three above-mentioned language characteristics.

Session 11, half-day tutorial: 9.00-12.00

Graphics Under X-Window – Mrs Michèle Coutant (SLX)

This tutorial presents the diversity of graphic standards: GKS, PHIGS, CGI, CGM, ..., their evolutions: GKS-3D, PHIGS+, PHIGS++, and the problems due to their integration in a distributed windowing system (X-Window, PEX).

The seminar is geared toward a wide technical audience of people who use graphics and go under a windowing system.

Session 12, half-day tutorial: 14.00-17.00

UNIX System V Administration – Mr Michel Wurtz (Institut Géographique National)

This presentation is aimed at those having already sufficient knowledge of the main Unix commands, shell (sh) in particular, and who are, or will be, entrusted with the system and its administration responsabilities. All the aspects of such a task will be dealt with.

Sessions 13 and 14, full day tutorial: 9.00-12.00, 14.00-17.00

An Introduction To The UNIX Operating System – Said Fawzi (AFUU)

This tutorial is devoted to the people who have to work in a Unix environment. After a general overview, we approach the basic commands, the file system, the directory structure and its manipulation. Then we detail the files and the access rights, as well as the editors. The shell will be described under its two features: command interpreter with I/O redirections, process management, and programming language with control structures, variables, built in commands and shell-scripts. We end by some considerations on backup and printing.

Sessions 15 and 16, two half-days tutorials: 9.00-12h and 14.00-17.00

The language for the tutorials will be French, except for the tutorial on OSF/1 (English).

Technical Sessions Wednesday 28 March

Room A 10.00 to 12.00
Opening Session: "Open Systems, Coming Of Age"
Chairman Mr Cédric Thomas (Pierre Audoin Conseil)

Mr Walter de Backer
Director of the DP Management, ECC - European Community Commission

Mr John Totman
Director of European Operations, X/Open Company Ltd

Mr Victor Izquierdo de Loyola, Sub Director General de Coordinacion
Informatica Ministerio para las Administraciones Publicas

Mr Farouk Kamoun
Président Directeur Général, Centre National de l'Informatique de Tunis

Pause 12.00-14.00

Room A 14.00-15.30
User Interface Session
Chairman to be announced

X et Chorus: architecture d'un terminal graphique
Mr José Rogado (Gipsi SA)

Optimizing windows System toolkit in a server based environment
Mr Nayem Islaam (Sun USA)

Room B 14.00-15.00
Real Time Session
Chairman Mr Humberto Lucas (Encore Computer)

Le multiprocesseur D3000 et DUNE-ix.
Mr Gérard Morisset (Dune Technologie)

CX/RT, Unix temps réel sur Night-Hawk
Mr Jean-Marc Brangoulo (Harris Computer)

Real/IX, Unix temps réel de Modcomp
Mr Herve Le Jeune (Modcomp)

Pause 15.30-16.00

Room A 16.00-17.30
Visualisation Tools Session
Chairman Mr Michel Grave (Onera)

L'image synthetique, un outil essentiel en sortie des super-ordinateurs
Mr Jean-Francois Colonna (Ecole Polytechnique, CNET)

Onyx, un générateur interactif d'interfaces graphiques en C, au-dessus de X-Window
Mrs Fabienne Dravers (Cognitech)

Room B 16.00-17.30
Large System Session
Chairman Mr Michel Wurtz (Institut Géographique)

Expérience sur EBASE 2000 appliquée á la sécurité et á la messagerie
Mr Le Floch (France Télécom)

L'expérience Unix à Lissac
Mr Michel Laissus (Lissac)

Unix in an English administration
Mr Mark Rickard (Inland Revenue)

Experience with distributed Unix
José Luis Ortigosa (Repsol Butano)

* There will be simultaneous translation between English and French during the conference.

Technical Sessions Thursday 29 March

Room A	9.00-10.30	**Room B**	9.00-10.00
Tools (I) Session		**PME-PMI Session**	
Chairman	Mr Yves Legrandgérard (Ecole Poly)	**Chairman**	Mr Jean-Michel Cornu (consultant)
	Gestion d'un Agenda Electronique par mail Mr Pierre Rolin (ENS T. A.)		Unix dans le milieu industriel Mr Olivier Ishacian (Robot soft)
	wsmon: A Working-Set Monitor Mr Marty Itzkowicz (Sun)		Unix caché à l'utilisateur final Mr Olivier Desnoux (Sofiac)
	Bridge: un générateur d'interfaces Jacques Durand (LRIM)		Application de Base de données Mr Bruneau (CNTS)
			Applications multimedias Mr Jean-Pierre Lichtin (VT Com)
	Pause 10.30-11.00		
Room A	11.00-12.30	**Room B**	11.00-12.30
Tools (II) Session		**System Products Session**	
Chairman	To Be Announced	**Chairman**	Mr Pascal Beyls (Bull)
	The Shell + /rdb as a 4GL Mr Evan Schaffer (R Schaffer Wright)		Unix System V.4 Mr Jean-Philippe Bardin (Intel)
	Interface entre SGBD et graphiques Mr de Cock (E N des Mines)		Open Desktop Mr Philippe Bichon (SCO)
	L'internationalisation sous Unix Mr Michel Gouget (Newlog)		OSF/1 for an open future Mr Paul Wahl (OSF Munich)
	Pause 12.30-14.00		
Room A	14.00-15.30	**Room B**	14.00-15.30
New Directions For Operating Systems,		**Market & Strategy Session**	
Chairman	Mr Sylvain Langlois (Chorus Systèmes)	**Chairman**	Mr Francis Ducroux (Renault SA)
Panel Session Featuring:	Mr Jacques Fèbvre (OSF Grenoble) Directeur de Centre Recherche		Le marché du terminal X Mr Lionel Vogel (Gipsi SA)
	Mr Michel Gien (Chorus Systèmes) Directeur General, Responsable R & D		Open Systems in the 1990's Mr Don Tapscott (DMR Group)
	Mr Marc Shapiro (Inria) Responsble du projet SOR		Marché Européen Mr Cédric Thomas (PAC)
	Mr Gérard Vandome (Bull) Projet ESPRIT COMANDOS		La portabilité et les standards Mr Jean-Michel Cornu (Consultant)
	Pause 15.30-16.00		
Room A	16.00-17.30	**Room B**	16.00-17.30
New Directions For Operating Systems		**Tools Session**	
Chairman	Mr Sylvain Langlois	**Chairman**	Mr Rémi Saccoman (Framentec)
Panel Session	Discussion		HP Softbench, HP Encapsulator, UIMX Mrs Anne Dewilde (Hewlett-Packard)
			WISET: Wipro Software Evaluation Mr N.S Muralidhara and Mrs Sarala
			System Manager Mr Anureg Narula (Silicon Graphics)

Technical Sessions Friday 30 March

Room A	9.00-10.30	**Room B**	9.00-10.30
Security Session		**Information Processing Session**	
Chairman	TBA	**Chairman**	Mrs Anne Francois (Consultant)
	La Sécurité sous Unix Mr B. Visser (Unisys)		Scope Gestion Mr Antoine Harmand (Thom'6)
	Prototype for intrusion detection in a secure network Mr J.R. Winkler (Planning Research)		Le systéme SAS sous Unix Mr Christophe Delprat (SAS Institute)
			La complémentarité d'Ingres et d'Unix Mr Gérard Dodeman (Ingres)
	Pause 10.30-11.00		
Room A	11.00-12.30	**Room B**	11.00-12.30
Distributed Systems Session		**Migration & Integration Session**	
Chairman	Mr Francis Capy (DEC PRL)	**Chairman**	Mr Frédéric Nikitenko (OR Santé)
	Unix et la répartition Mr Francois Armand (Chorus)		Informatique de gestion sous Unix Mr Jean-Claude Dubourvieux (Philips)
	Encapsulation des communications dans un systeme à objets répartis Mr Yvon Gourhant (Inria)		Unix et postes de travail MS-DOS Mr Charnay (Menuiseries Lapeyre)
			Choix entre OS/9 et Unix Mr Jacques Vela (VS9)
			Acceptation d'Unix par la convivialité Mr Pavot (Ivao)
	Pause 12.30-14.00		
Room A	14.00-15.30	**Room B**	14.00-15.30
Man/Machine Interface Session		**4GL, AGL Session**	
Chairman	Mr Peter Van der Linden (Gipsi SA)	**Chairman**	Mrs Véronique Mansart
	To Be Decided		L'atelier de genié logiciel Pallas Muriel Simon (Steria Developpement)
			Uniface: developpement d'applications Mr Claude Geslin (Infi)
			Open Case: atelier de genié logiciel Mr N.C Boistière (Oriane)
	Pause 15.30-16.00		
Room A	16.00-17.30	**Room B**	16.00-17.00
Man/Machine Interface, Panel Session		**Network Session**	
Chairman	Mr Peter Van der Linden (Gipsi SA)	**Chairman**	Mr Philippe Dax (Télécom Paris)
	Discussion		Systèmes distribués Mr Michel Kochanski (Sun)
			Tootsi Mr Jean-Marc Goetz (Télésystemes)
			Liaison terminal host sur réseau X-25 Mr Robert Becker (Experteam SA)

The language for all the conference will be French and English (simultaneous translations) (14 translators during 3 days).

UKUUG News

Mick Farmer
mick@cs.bbk.ac.uk

Birkbeck College
Malet St
London WC1
England

Mick is a lecturer at Birkbeck College (University of London) and the Secretary of the UKUUG. His interest is in all aspects of Distance Learning and he is the Senior Consultant (Software) for LIVE-NET, an interactive video network connecting London's colleges. He is also a member of the University's VLSI Consortium, mainly because the design tools draw such pretty pictures.

Start Bit

With our large London conference coming up (see below) I'm often asked how to contact various UKUUG personnel, so I'll start this report with a number of distribution lists.

- ukuug@ukc.ac.uk the executive and administrative committee.
- ukuug-exec@ukc.ac.uk the executive committee.
- ukuug-tech@ukc.ac.uk technical enquiries concerning the London conference programme.
- ukuug-conf@ukc.ac.uk general enquiries concerning the London conference.

Membership Figures

A steady increase in the number of new members has taken our membership figures past 400, an 8% improvement in three months. In my last report I said that we were approaching saturation point in respect of academic members. How wrong can you be! Five academic sites have joined in the last three months. The table below shows the breakdown of our membership at the end of January.

Category	Number
Academic	103
Commercial	240
Honorary	5
Individual	52
Student	1
Total	401

Note that we have introduced student membership and are still discussing possible additional categories.

Cyfarfod '89 Technegol y Gaeaf

A very successful technical meeting was held in Cardiff (Wales) on 11-13 December last year, attended by over 150 people*. Jim Craigie's

* The abstracts to papers delivered at this conference are printed at the end of this newsletter.

X.400 tutorial was also well attended, no doubt showing widespread interest in the latest developments in this area. Robert Evans and his team from the strangely named Department of Computing Mathematics, University of Wales College of Cardiff are to be thanked. Our thanks also to the UNIX Book Service for donating a copy of the new 4.3BSD Book as a gift to Robert. Their address is:

The UNIX Book Service
35 Bermuda Terrace
Cambridge CB4 3LD
England

+44 223 313273

System Administration Workshop

By the time you read this our one-day workshop will have taken place (14 February). Hosted by Neil Todd (GiD Ltd.) the presentations include

- Report from the POSIX working group on System Administration
  Martin Kirk, British Telecom Research Laboratories.
- A la Recherche de /tmp Perdu
  Lindsay Marshall, Newcastle University.
- Computer Security Risk Management
  Peter Trueman, TopExpress Ltd.
- Mail between X.400 and RFC 822 environments
  Andrew Macpherson, STC Technology Ltd.
- Overview of Project Athena
  Dave Edmondson, Lee McLoughlin, Stuart McRobert, Imperial College.
- Software Management in a Heterogeneous Environment
  Andrew Findlay, Brunel University.
  Piete Brooks, Cambridge University.

A VHS video of the proceedings of this workshop will be available from April. For details contact:

Birkbeck College Video Services
Department of Computer Science
Birkbeck College
Malet Street
London WC1E 7HX
England

+44 71 631 6351

Note that VHS videos of our previous workshop on UNIX security are also available from the above address, price £50.00 (plus VAT in the UK).

UNIX — The Legend Evolves

This summer we are hosting a major conference at the Royal Lancaster Hotel (London, England) from 9-13 July. Many of the major workers in the UNIX and operating systems community are speaking. Bell Labs are using this conference to make the first public announcement of *Plan 9*, their successor to UNIX .

- Speakers from Bell Labs include† Jon Bentley, Tom Duff, Stuart Feldman, Brian Kernighan, Rob Pike, Dave Presotto, Dennis Ritchie, and Ken Thompson.
- Other speakers include Kirk McKusick and Mike Karels from Berkeley, Andrew Tanenbaum from Vrije University, and Doug Comer from Purdue University.

This meeting follows the traditional EUUG format of tutorials on Monday and Tuesday with the conference from Wednesday until Friday. An exhibition will run concurrently. For further details see the distribution lists at the beginning of this article.

Free Conference Place!

Competitions are a *fun* element at UNIX conferences with fantastic prizes for the winners. Remember the red Ferrari in London April '88! Here are some examples of previous competitions.

- Complete the following sentence: "I go to UNIX conferences because ...".
  Winning entry: "to find out if anyone ever gets my e-mail".
- Invent a new error number.
  Winning entry: E_NO_TOBACCO – read on an empty pipe.
- Invent a new signal number.
  Winning entries:
  SIG_MA – mother board violation;
  SIG_TITANIC – floating-point exception.

† in alphabetical order!

You are invited to *invent a competition* for our London conference (see above). The winning entry will receive a free registration for London. Send your entry to me (preferably by e-mail).

Winter '90 Technical Meeting

This meeting will take place at Queens' College, Cambridge (England) on 17-19 December, 1990. As with our other winter meetings this one will have a strong networking flavour.

Forthcoming Events

We continue to organise our events as far ahead as possible. Our Summer '91 meeting will be held in Liverpool (England) in June or July and the Winter '91 meeting will be held in Edinburgh (Scotland) in December.

London UNIX User Group (LUUG)

This lively group continues to meet on the last Thursday of each month (except December). Andrew Findlay (Andrew.Findlay@brunel.ac.uk) is the organiser and he should be contacted for more details.

FaceSaver Project

This project is now well under way as most of the people who went to our Cardiff meeting will recall. Acorn Computers Ltd. have kindly donated to the UKUUG one of their R140 computers (running UNIX of course :-), a frame grabber, and two monitors. The software is being developed by a team at Imperial College, London. Faces will be available in a number of formats including the USENIX format and a PostScript format. For further details contact me at the address given at the beginning of this article.

Software Distribution Service

This is another service that the UKUUG have introduced for our members. The service is based at Imperial College where a UKUUG-purchased disk has been installed. The software is currently being loaded but already people have been requesting the latest version of X11.

Stop Bit

In May this year London's telephone numbers are being changed. The 01- code is being split into an inner London code, 071-, and an outer London code, 081-. There will be a lengthy changeover period during which all three codes will be usable.

Newsletter Editor

Tired of Fleet Street? Looking for a new challenge?

This may be for you! The UKUUG has a vacancy for its Newsletter editor. The successful candidate will collect information for inclusion, produce camera-ready artwork, and coordinate distribution and printing. Initially there will be six issues annually, the style and contents to be decided by the UKUUG executive and the editor.

Prospective candidates should

- Live in the UK.
- Have a network connection.
- Have experience of producing a Newsletter.
- Have access to the necessary equipment for producing the Newsletter.

The UKUUG will pay all costs and a fee (to be negotiated) upon successful production of each Newsletter. Help and advice is available for setting up the production.

For further information contact:

UKUUG Secretariat
Owles Hall
Buntingford
Herts SG9 9PL

Tel: +44 763 73039
Fax: +44 763 73255

Net: ukuug-exec@ukc.ac.uk

EUUG

European UNIX* systems User Group

PRELIMINARY ANNOUNCEMENT and CALL FOR PAPERS

EUUG Autumn '90 Conference and Exhibition at

Nice Acropolis
22-26 October 1990

Preliminary Announcement

The 1990 European UNIX* systems User Group Technical Conference will be held from 22nd-26th October, 1990 at the Nice Acropolis, Nice, France.

Technical Tutorials on UNIX and closely related subjects will be held on Monday 22nd and Tuesday 23rd October, followed by the three day **Conference** with commercial **Exhibition** finishing on Friday 26th October.

A pre-conference registration pack containing detailed information will be issued in June 1990.

Call for Papers

The EUUG invites papers from those wishing to present their work. Full papers or extended abstracts must be submitted. All submitted papers will be referred to be judged with respect to their quality, originality and relevance.

Suggested subject areas include, but are not limited to:

- ★ *Software Management for large projects*
  - + *Configuration Management*
  - + *Maintenance Management*
  - + *Update and Release control*
- ★ *OSI and OSI application on a Unix platform*
- ★ *System Administration in a Hetrogeneous environment*
- ★ *Security and Audit*
  - + *Secure Unix*
  - + *Securing existing systems*
- ★ *Unix in non English speaking environments*
- ★ *User Interface Management Systems (UIMS)*

Important Dates

Submission deadline	30th April, 1990
Acceptance notification	10th May, 1990
Final paper	30th June, 1990
Closing date for Student Grant Application	1st September, 1990

Method of Submission

Full papers or extended abstracts must be submitted by post to the EUUG Secretariat and, if possible, in electronic form to euug-nice@eu.net. All submissions will be acknowledged by return of post.

Tutorial Solicitation

Tutorials are an important part of the EUUG's biannual events providing detailed coverage of a number of topics. Past tutorials have been taught to be leading experts.

Those interested in offering a tutorial should contact the EUUG Tutorial Executive as soon as possible.

Additional Information

We will be pleased to provide advice to potential speakers. We can be contacted at the addresses below.

If you wish to receive a personal copy of any further information about this, and future EUUG events, please write, or send electronic mail, to the Secretariat.

Useful Addresses

Secretariat
EUUG
Owles Hall
Owles Lane
Buntingford
Herts
SG9 9PL
UK

Tel: +44 763 73039
Fax: +44 763 73255
Email: euug@Eu.net

Conference & Tutorial Executive
Neil Todd
GID Ltd
1 Captains Gorse
Upper Basildon
Reading
Berkshire RG8 8SZ
UK
+44 491 671 964

neil@gid.co.uk.

Programme Chair
Johan Helsingius
OY Penetron Ab
Box 90
02201 ESPOO
Finland

Tel: +358 0 427632
Fax: +358 0 428725
email: julf@fuug.fi

Local Organisation
AFUU
Anne Garnery
11 Rue Carnot
94270 Le Kremlin Bicetre
Paris
France

Tel: +33 1 46709590
Fax: +33 1 46589420
email: anne@afuu.fr

Programme Committee
Pierre Louis Neumann
INRIA
BP 105
78153 Le Chesnay Cedex
France

Tel: +33 39635562
Fax: +33 39635330
email: neumann@inria.fr

Dr. Elod Knuth
MTA SZTAKI
Computer & Automation Inst.
Hungarian Academy of Sciences
H-1502 Budapest
POB 63
Hungary

Tel: +36 1 1665 644
Fax: +36 1 1667503
email:

Daniel Klein
CMU
Software Engineering Institute
Pittsburgh PA 15213
USA

Tel: +1 412 268 7791
Fax: +1 412 268 5758
email: dvk@sei.cmu.edu

For further details, contact:
The Secretariat
European Unix® systems User Group
Owles Hall, Buntingford, Herts SG9 9PL, UK
Phone: (+44) 763 73039 Fax: (+44) 763 73255 Network Address: euug@Eu.net

Quick EUnet update

Anke Goos
Anke@germany.eu.net

Computer Science Department, University of Dortmund
IRB - Anke Goos
4600 Dortmund 50
P.O.Box 500 500, W-Germany

Originally a journalist and student of journalism with a focus on "new media" at the University of Dortmund, she got hooked onto Unix by a job for User Information in the German EUnet backbone. She got a dubious reputation for work on the European E-Mail Directory, the EUnet glossy, German EUnet documentation, editing the Newsletter of the German Unix User Group and annoying her technical colleagues in the EUnet backbone with reminders about "what the users want". Besides, she is writing articles for German (Unix) magazines, if she can't defend it and tries to pursue her studies, currently with a research paper about - "surprise, surprise" - European computer networks.

Hi EUnet folks,

Before taking up the network column I had some ideas about what it should look like: it should not only reflect what EUnet is by its contents, but also by the spirit of the article and by its format. This may mean some sort of mail to EUnet folks, maybe even an article but also News from the EUnet users, that may be printed with some neat programs like the news-pretty printer. The article should be posted to eunet.general in order to inform the users.

But an underlying feature of EUnet is that you should avoid any unnecessary work, take whatever you have at hand to ensure a service at a low price. So, I'll try to give a quick update about current and recent EUnet activities at the backbone.

Cooperation in RIPE

There are great efforts under way for establishing IP (Internet Protocol) as another InterEUnet service. This is being done in cooperation with other European networks like HEPnet and EARN within the coordinating framework of RIPE (Réseau IP Européen). The vision is to establish this as a standard European infrastructure. From EUnet Daniel Karrenberg from CWI is the driving force and IP wizard. Due to these activities even the association of the national X.400 networks, RARE (Réseaux associés pour la Recherche Européenne) is currently moving and could possibly integrate the IP infrastructure as underlying protocols below OSI applications.

Due to these combined efforts for InterEUnet I also see that the scheme for the distributed archives for Public Domain software and information has become somewhat stuck. I may be wrong, but it is obvious that the UK and Scandinavia are better off in the archive scene, while the rest of Europe is missing the tireless work of volunteers, taking up responsibility for a piece of or a whole archive. Probably some coordination would be useful to synthesise the strengths instead of re-inventing the wheel at each archive-site.

EUnet starter kit

The EUnet starter kit is a collection of all the software and documentation, that a EUnet beginner would need for an easy start. Frank Kuiper of CWI is currently combining some of this into an EUUG tape, as far as his work at CWI gives him some free time.

EUnet in Eastern block countries

As far as I know now, at the beginning of 1990, there has been a Yugoslavian backbone for years. As for Hungary, the Austrian backbone was trying to set up a connection with a link to an academic site, but they ran into problems with the Hungarian X.25 network. A link from the Finnish backbone to Unix users in Estonia (a country in

the Western part of the Soviet union) has been under discussion since the middle of 1989. Basically EUUG representatives fear political problems as they don't want to interfere with the movement for autonomy, while a Unix User group in Moscow could show up as an official representative. I guess there are some more activities in other countries that I don't know about.

During 1989 the Unix developers and users group UEAG in East Germany exchanged information by sending tapes every four months to their roughly 500 member organisations. At the beginning of this year several West Berlin EUnet sites tried to set up links to East Berlin. Strangely enough, it seems that the telephone connections between East and West Berlin are very bad, while links to West Germany and other countries are much better. Anyway, there is a great need for a Unix network, as the Unix users are lackng information and documentation of any kind, while paper, printing and copy machines are a scarce commodity in East Germany. We found it depressing to see the far worse situation in those regions far away from Berlin, like Karl-Marx-Stadt in the South. You may find several articles from East German Unix Users about developments under the "cheese cover" - as they call their former state - in the spring issue of the GUUG newsletter.

The European E-Mail Directory

Oh well, let's see. I was surprised to read in the last EUUG newsletter that the EMD would be ready in January, as no one had told me who would do my bit :-). I am still updating the introductory part of the European E-Mail Directory and prototyped a text to spread my knowledge on how to prepare and edit the national parts for the EMD. I am pushing Daniel Karrenberg that he should finish the software part, before he's totally off and away into RIPE.

For next year's issue there is definitely be a need for someone else to look after the software part of the E-Mail Directory. Nevertheless the level of mail from users in my mailbox last year showed, that there is a need for some kind of online e-mail directory for European networks. On the US side of the Atlantic people like John S. Quarterman and Tracy LaQueue are trying to transform network books like "The Matrix" and "The Texas E-Mail Directory" into an online information services as by Matrix Inc. There may be an interest for cooperation, at first for a mail access to the European E-Mail Directory. But once more, there is a need for volunteers in close cooperation with the EUnet backbone to program this service. Some experience and software methods for the E-Mail Directory may be useful. Am I the only EUnet user who would like to see such a service... ?

The EUnet Glossy

Daniel Karrenberg and yours truly finished the information booklet that describes EUnet and how to connect to it in spring 1989. It is written in English, and gives an idea of what the overall idea of the network is, while each national backbone should be able to include a page in the national language. It has been held up before printing since the Vienna conference in September, due to some problems with EUUG representatives. Partially this may be due to different ideas about what an information booklet should look like. I assume that it will be printed by the time of the Munich conference, but by then parts will already be out or date (grrrr).

This and that in EUnet

The volunteers for the project of MOC (Mailing On Conferences) are working on an enhancement to provide an individual login for receiving and sending your mail during the EUUG conference in Munich and - maybe, depending on cooperation of the German PTT - IP access.

In January the Norwegian backbone moved quickly from *ndosl*, which was supported by a company to *nac*, obviously driven by the Norwegian Unix User Group.

A last hint: Maybe you want to ask for a site in a specific European country, but there's no online access to the E-Mail Directory and you don't know for example, what the current hostname of the Norwegian backbone is. Please send your mail to postmaster@norway.eu.net or postmaster@austria.eu.net, @france.eu.net, @iceland.eu.net, ...

Drop me a line, what you would like to see in the next EUnet column, information that you are missing or want to give. I may give you the answer.

Disclaimer: Wonders take a little longer :-)

At the Oxford/Berkeley Summer Engineering Programme, 2-13 July, 1990, in Oxford, England...

Introduction to 4.3BSD UNIX Internals: Data Structures and Algorithms

A five-day course / Monday-Friday, 2-6 July, at the University of Oxford... just before the London UKUUG conference, "UNIX—the Legend Evolves"

This course presents a broad overview of how the UNIX kernel provides its basic services. Individuals involved in **technical and sales support** can learn the capabilities and limitations of the system; **applications developers** can learn how to effectively and efficiently interface to the system; **systems programmers** without direct experience with the BSD (Berkeley) UNIX kernel can learn how to maintain, tune, and interface to such systems.

The course places heavy emphasis on the installation and operation of networks using 4.3BSD UNIX or derivatives. It should be especially useful for those who install or maintain hosts and networks connected to the Internet, and those who modify or port the 4.3 network kernel or utilities to other systems.

Faculty

Marshall Kirk McKusick, Research Computer Scientist, Berkeley Computer Systems Research Group, University of California at Berkeley

Michael J. Karels, Principal Programmer, Berkeley Computer Systems Research Group, University of California at Berkeley

Fee

£825, including course materials, lunch, refreshments each day and the text, Samuel J. Leffler, Marshall Kirk McKusick, Michael J. Karels, and John S. Quarterman, *The Design and Implementation of the 4.3BSD UNIX Operating System* (Addison-Wesley, Reading, Mass., 1989). Registration is limited and advance registration is required. Accommodation is not included.

For a free brochure describing this course and 17 others in the Oxford/Berkeley Summer Engineering Programme, 2-13 July at the University of Oxford, and containing full registration information and forms, write to Continuing Education in Engineering, University Extension, University of California, 2223 Fulton St., Berkeley, CA 94720; telephone +01 (415) 642-4151; fax +01 (415) 643-8683 (Attention: Engineering); Internet: ceeunex@garnet.berkeley.edu

Programme

Monday, 2 July

MORNING

Kernel overview—**McKusick**
- Kernel terminology
- Basic kernel services
- Process structure

AFTERNOON

Kernel resource management—**McKusick**
- Memory management
- Paging and swapping
- Scheduling
- Signals

Tuesday, 3 July

MORNING

Kernel I/O structure—**McKusick**
- Special files
- Line disciplines
- Multiplexing I/O
- Autoconfiguration strategy
- Structure of a disk device driver

AFTERNOON

File system overview—**McKusick**
- File system services
- Block I/O system (buffer cache)
- File system implementation
- Vnodes and NFS

Wednesday, 4 July

MORNING

System tuning—**McKusick**
- Performance measurement
- System tuning
- Crash dump analysis
- Security issues

AFTERNOON

Interprocess communication—**McKusick**
- Concepts and terminology
- Basic IPC services
- Example use of IPC and network facilities
- Subnets, address resolution, and trailers
- Routing policy

Thursday, 5 July

MORNING

Networking structure—**Karels**
- Development of TCP/IP for UNIX
- Internet protocols and addresses
- System layering
- System interface

AFTERNOON

Network management—**Karels**
- Network installation with 4.3
- Network configuration
- Routing issues
- Testing and troubleshooting
- Gateway issues

Friday, 6 July

MORNING

4.3 network internals—**Karels**
- System layers and interfaces
- IP/TCP examples
- Changes in 4.3BSD

AFTERNOON

Other topics—**Karels**
- Network utilities including Telnet, FTP, rlogin
- Performance work in TCP/IP
- Other network protocol families
- Future work at Berkeley

University of Oxford and University of California at Berkeley

Call Doc Strange

Colston Sanger
doc.strange@olibc1.oliv.co.uk

Olivetti International Education Centre

Colston Sanger is a lecturer at the Olivetti International Education Centre, Haslemere, UK and a visiting lecturer in the Faculty of Engineering, Science and Mathematics at Middlesex Polytechnic. He is very sorry that he wasn't able to do a column for the last issue...

FMLI — the Forms and Menu Language Interpreter

FMLI, the Forms and Menu Language Interpreter, is new in UNIX and UNIX/386 System V Release 3.2. FMLI syntax is a bit like shell, but also object-oriented. FACE, the Framed Access Command Environment, that I mentioned in my last column is actually an application built with FMLI.

FMLI provides a framework, a consistent 'look and feel', for applications that use menus and forms. It controls many aspects of screen management for you — with the result that you don't have to concern yourself with the low-level details of creation or placement of menus and forms on the screen, or of providing users with a means of navigating between or within them. Also, FMLI is terminal-independent. It will work on any character terminal: in colour if the terminal supports colour, otherwise in monochrome. In fact, the whole look and feel of FMLI is designed to be compatible with OPEN LOOK, the AT&T/Sun Microsystems graphical user interface for intelligent workstations.

'Object-Oriented'

I said that FMLI is object-oriented. An *object* in FMLI is either a form, menu or text frame and the items those frames contain. When you define a form or menu, you are defining an object. An *object operation* is an action that can be performed on an object. Object operations can be ordinary UNIX commands, which the FMLI interpreter passes to the shell for execution, but are much more likely to be FMLI built-in commands or keywords. For example, here is a fragment of FMLI code:

```
action=OPEN FORM Form.ph.lookup
```

It specifies the action to take when a selection is made from a menu. In the example, `OPEN` is recognised as a keyword, a FMLI command that forces an object operation to occur. The type of object to `OPEN` is a `FORM` whose name is `Form.ph.lookup`. Here I'm using both the `FORM` type-cast and the FMLI naming convention for forms (*i.e.*, `Form.*`), but it's really only necessary to use one or the other.

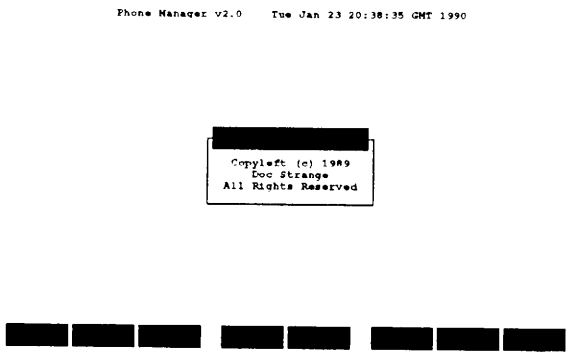
A Sample Application

The best way to introduce you to FMLI is probably to build a sample application. What I've done is built yet another version of the classic *phone_mgr* script for managing a list of names, addresses and telephone numbers. (See the 'UNIX Clinic' column in Vol.8 No 2 — Summer 1988.)

Anyway, here goes.

Typically, the scripts for a FMLI application consist of a set of frame definition files, each defining a single menu, form or text frame. In addition, applications can include three (optional) definition files: an initialisation file, a commands file and an alias file. You can use these three files to define global features of your application. I've only used one of these optional files in the *phone_mgr* application: the initialisation file. Here it is:

```
# Init.ph
#
# phone_mgr introductory object
title="Phone Manager v2.0"
text="\n  Copyleft (c) 1989\n     Doc Strange\n  All Rights Reserved."
rows=4
columns=25

# banner line
banner="Phone Manager v2.0    'date'"
bancol=center

# Colour Attributes
screen=black
banner_text=white
window_text=white
active_border=cyan
inactive_border=red
active_title_text=black
active_title_bar=cyan
inactive_title_text=black
inactive_title_bar=red
highlight_bar=cyan
highlight_bar_text=black
```



If you use an initialisation file, you can supply its name as an argument to the *-i* option to the FMLI interpreter, as in

```
$ fmli -i initialisation_file
```

or as I've done in packaging the *phone_mgr* application as a shellscript:

```
# phone_mgr - Phone Manager v2.0
# (FMLI version)
tput init
exec fmli -i Init.ph Menu.ph
```

The initialisation file above first defines a transient introductory frame (with the application name and a bogus copyright message) that is displayed when the application is invoked and is then cleared and replaced by the initial menu frame. This introductory frame is displayed again briefly when you exit from the application. The `title` descriptor defines the title that appears in the title bar of the introductory frame; the `text` descriptor defines the text that appears in the frame; `columns` defines how wide it is; and `rows` defines how high it is.

Next I define a `banner` line, a line that is displayed at the top of the screen the whole time the application is running. `bancol` says I want the banner centred (in fact, this is the default). I haven't defined the `working` descriptor — a string used to notify users that they must wait until FMLI completes an activity — so the default '`working...`' will be displayed at the righthand edge of the banner line.

Finally there are the colour attribute descriptors that let you define the colours of various elements

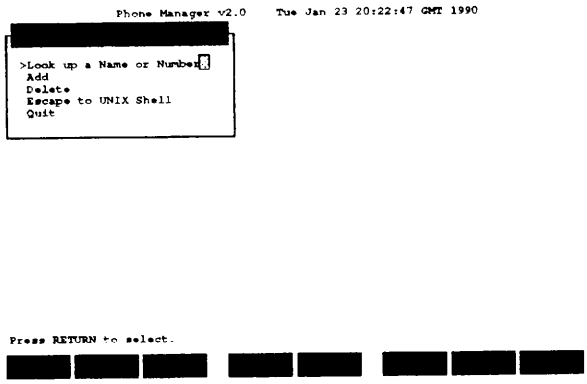
of the FMLI screen. These colour descriptors can only be defined in the initialisation file and, because of the nature of *curses*(3X), they must be set in pairs. In the initialisation file above, I've simply copied in the colour descriptors used in the FACE initialisation file. Obviously, if your terminal doesn't support colour, these descriptors are ignored.

```
# Menu.ph - Phone Manager main menu
#
menu="Menu"

# where to get help if the user
# presses the HELP key
help=OPEN TEXT Text.ph.mhlp

# menu items, with actions
name=Look up a Name or Number
itemmsg="Press RETURN to select."
action=OPEN FORM Form.ph.lookup

name=Add
itemmsg="Press RETURN to select."
action=OPEN FORM Form.ph.add

name=Delete
itemmsg="Press RETURN to select."
action=OPEN FORM Form.ph.del

name=Escape to UNIX Shell
itemmsg="Press RETURN to select."
action=unix

name=Quit
itemmsg="Press RETURN to select."
action=exit
```

There's not an awful lot to say about this. The `menu` descriptor defines a title to appear in the title bar of the menu frame; and `help` defines a help text frame to open if the user presses the HELP function key. The other descriptors are for menu items. `name` is what will appear on the menu; and `itemmsg` defines a message that will appear in the message line (the second line from the bottom of the screen) if you navigate to this menu item. (There are two ways of navigating to a menu item: you can either use the cursor keys or type the first part of the item's name in upper or lower case — it isn't case-sensitive.) The `action` descriptor defines what will happen if you select this menu item.

The Main Menu

Here is the main menu frame for the *phone_mgr* application:



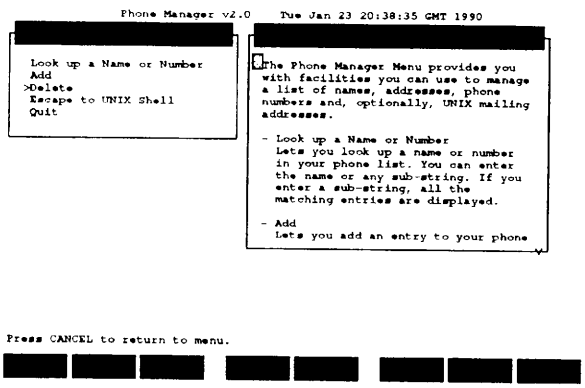
A Help Text Frame

It's always a good idea to provide help, and FMLI has a dedicated HELP function key. The help text frame below, `Text.ph.mhlp`, is invoked if you press HELP while in the main *phone_mgr* menu. It's pretty straightforward, see `Text.ph.mhlp`.

It has a `title`, and `columns` and `rows` like those you've seen before. The `text` descriptor just defines the text to be displayed. Notice the back-quoted `message` command near the end of the file. `message` is a FMLI built-in command used to display messages on the message line. I could have used `message -b` here to ring the terminal bell as well. The `lifetime` descriptor can be either 'shortterm', 'longterm', 'permanent' or 'immortal' (!). 'shortterm', as it is here, means

that the help text frame will be removed from the screen when another frame becomes current.

```
# Text.ph.mhlp - help for main menu
#
title="Help for Phone Manager Menu"

# Width of this screen
columns=40

# Height of this screen
rows=15

text=" The Phone Manager Menu
 provides you with facilities
 you can use to manage a list
 of names, addresses, phone
 numbers and, optionally, UNIX
 mailing addresses.

  - Look up a Name or Number
    Lets you look up a name or
    number in your phone list.
    You can enter the name or any
    sub-string. If you enter a
    sub-string, all the matching
    entries are displayed.

  - Add
    Lets you add an entry to your
    phone list.

  - Delete
    Lets you delete an entry from
    your phone list. You can enter
    the name or any sub-string. If
    the name or sub-string you
    enter matches more than one
    entry, you will be asked to
    choose which one you wish to
    delete.

  - Escape to UNIX Shell
    Lets you temporarily escape to
    the UNIX System shell. To
    return to this menu, press
    CTRL-d.

  - Quit
    To quit from this menu.
"
# do not wrap (re-format) text
wrap=FALSE

# Display message
`message "Press CANCEL to return \
 to menu."`

# frame will disappear when
# no longer current
lifetime=shortterm
```



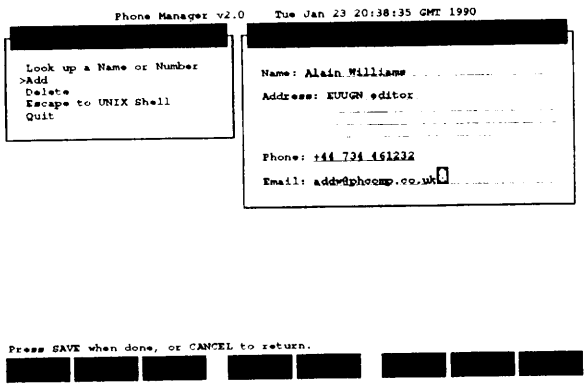
Forms

Forms are pretty easy too, up to a point. For example, here is the definition of the form used to add names and addresses, see `Form.ph.add`.

I've truncated this because the rest is just more field descriptors — essentially more of the same. The `title` and `help` descriptors you know all about by now. Skipping over the `done` descriptor for the moment, the field descriptors consist of the field `name` and its position within the form (`nrow` and `ncol`). These are followed by the input field position (`frow` and `fcol`) and its size (`rows` and `columns`). '`size=TRUE`' means that the field can actually be longer than its defined length and will scroll as required.

Now back to that `done` descriptor. This is where I began to find it difficult to do what I wanted to do with FMLI. `done` defines what happens when you press the SAVE key. `done` is a single-instance descriptor, one that can appear only once in a form, and it is of type *command*. The command here is `update`, which updates (refreshes) the screen. What I want 'done', however, is to have the contents of the input fields appended as a tab-separated line to my list of names, addresses and phone numbers (held in the file `phone.nos`), so I have to use a back-quoted expression that echoes (using the FMLI built-in command `echo`) the five input fields (`$F1` to `$F5`) to standard output.

```
# Form.ph.add
#
# Title
form="Add a Name and Number"
help=OPEN TEXT Text.ph.ahlp

# What to do when the user presses SAVE key
done=`echo "$F1 $F2     $F3     $F4     $F5     $F6     $F7" | putline` update

# Clear message line when frame is closed
close=`message -p ""`

# Form fields
name="Name: "
fieldmsg="`message -p Press SAVE when done, or CANCEL to return.`"
nrow=1
ncol=1
frow=1
fcol=7
rows=1
columns=30
scroll=TRUE

name="Address: "
nrow=3
ncol=1
frow=3
fcol=10
rows=1
columns=30
scroll=TRUE
.
.
.
```



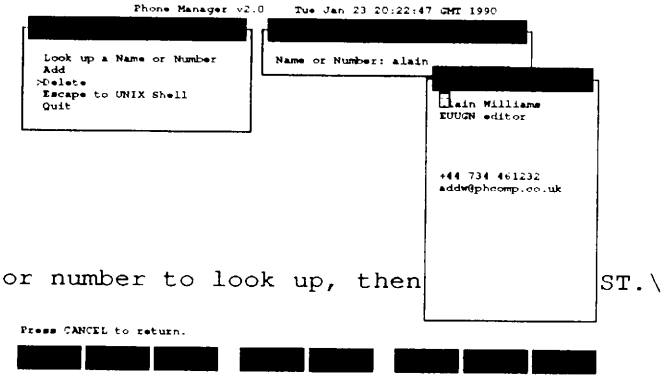
Then what? In fact, another difficulty. The current UNIX System V Release 3.2 version of FMLI has only a limited set of input/output redirection operators: < and >, but not >> or 2>. (It isn't really practicable to run a FMLI application from within a 'here document' (<<) because an explicit FMLI `exit` command is required to exit from an application.) The complete set (<, >, >>, 2> and 2>>) will be available in the UNIX System V Release 4.0 version of FMLI, but for now it's just plain irritating. To make up for this current limitation, the only way I can think of appending the input names and addresses to the `phone.nos` file is by piping standard output on to a stupid little shellscript called *putline*, the contents of which are:

```
# putline - stupid little
# shellscript
#
line >> phone.nos
```

OK, cool it. Let me show you the code for the 'Look up a Name or Number' form, see `Form.ph.lookup`.

A lot of this will be familiar to you — the input field descriptors, for example. By the way, in the `fieldmsg` descriptor, the `-p` option to the `message` command means make the message permanent, which is why I have to blank it out later with the `close` descriptor (`close` is evaluated whenever a frame is closed).

```
# Form.ph.lookup
#
# Title of this form
form="Look up a Name or Number"

# Where to get help
help='message "Enter a name or number to look up, then SAVE or LIST.\
 Press CANCEL to return."'

# Check if name exists and display it, otherwise display a suitable message
done='message "Looking for \"$F1\".";
        grep -i "$F1" phone.nos > /dev/null;
        regex -e -v "$RET"
                '0'     ''set -l COMMAND="OPEN TEXT Text.ph.lookup \"$F1\"";
                        ''
                '1'     ''message -b "Sorry, cannot find \"$F1\".";
                        set -l COMMAND="NOP";
                        ''
'$COMMAND

close='message -p ""'

# Input field
name="Name or Number: "
nrow=1
ncol=1
frow=1
fcol=17
rows=1
columns=20
scroll=true
fieldmsg="'message -p Enter a name or number to look up, then SAVE or LIST.\
 Press CANCEL to return.'"

# Field validation
# Valid input is one or more NOT space or tabs only
valid='regex -v "$F1" '^[       ]*$' false '^[^        ]+$' true'
invalidmsg="'message -b'Must be a name or number."


# Define FKey 8 as LIST
# NB. Fkey 3 (SAVE) does the same thing
name=LIST
button=8
action=done
```



What's new in this form are the field validation descriptors and the definition of an extra function key. I'd also like to discuss the definition of the done descriptor in this form in more detail.

Field Validation

The `valid` descriptor for the input field in the 'Look up' form above is defined to use the FMLI built-in command `regex` to validate the field.

`regex` uses the UNIX System V *regex*(3X) and *regcmp*(3X) functions, whose regular expressions are subtly different from those of *awk*, *ed*, *grep*, *sed*, *etc.* (See the 'UNIX Clinic' column in Vol.9 No.2 — Summer 1989) FMLI's `regex` command takes as input a stream of text and compares each line against one or more patterns.

These patterns represent regular expressions that are provided as arguments to the `regex` command line (so far so good: just like *grep*). However, a template — a string that is written to standard output if the corresponding pattern is matched — must appear after each pattern on the command line.

`regex` also provides ten registers, `$0-9` and `$m0-9`, in the pattern and template respectively. For example, the FMLI back-quoted expression

```
`readfile /etc/passwd | regex "^(fred:.*)\$0\$" '$m0'`
```

is a sort of *grep*. It scans `/etc/passwd` for a line starting with the login name 'fred' and writes the whole line to standard output. (The UNIX System V Release 4.0 version of FMLI will include as built-in commands a `fmlicut`, `fmliexpr` and `fmligrep`.)

As well as writing a template to standard output if a pattern is matched, `regex` also returns the string-value 'true', which is analogous to the way standard UNIX commands return a value of 0 on successful completion. If no pattern is matched, `regex` returns the string-value 'false'. The `valid` descriptor in the 'Look up' form illustrates this:

```
valid=`regex -v "$F1" '^[        ]*$' false '^[^         ]+$' true`
```

as well as the use of the `-v` option to tell `regex` to use the argument that follows (rather than standard input) as input.

Yet More On `regex`

The `done` descriptor in the 'Look up' form uses `regex` with a `-e` option, which tells it to evaluate the corresponding template and write the result to standard output. To remind you:

```
# Check if name exists and display it, otherwise display a suitable message
done=`message "Looking for \"$F1\".";
        grep -i "$F1" phone.nos > /dev/null;
        regex -e -v "$RET"
                '0'     ''set -l COMMAND="OPEN TEXT Text.ph.lookup \"$F1\"";
                        ''
                '1'     ''message -b "Sorry, cannot find \"$F1\".";
                        set -l COMMAND="NOP";
                        ''

`$COMMAND
```

Maybe I should explain what I'm trying to do here. What I'm trying to do is run a *grep*, then test the exit status (in *shell*, `$?`) to decide what to do next.

Now, I should be able to use `regex` instead of the *grep*. After all, I've just illustrated how to use `regex` to search for 'fred' in `/etc/passwd`. In the end I decided to use *grep* with the System V *-i* option (ignore case in searching) because the alternative seemed to be a horrendous regular expression. I then use `regex` in a kind of *shell* `case` statement to test the value of the built-in variable `$RET`, the exit status of the last executable (i.e., not built-in command?) run by the FMLI interpreter. Notice the multitude of quotes here.

Redefining Function Keys

FMLI provides two levels of screen-labelled function keys, known as SLKs (pronounced 'slicks'). Function keys F1-7 are defined by default as CANCEL, CHOICES, SAVE, PREV-FRM, NEXT-FRM, HELP and CMD-MENU. They may be disabled, but cannot be redefined.[1] (If your terminal doesn't have function keys, you can use the alternate keystrokes CTRL-f*n*, where *n* is a number.) F8 is undefined by default, but will be defined as CHG-KEYS if any of the alternate set F9-15 are defined. (F16 will also be defined as CHG-KEYS if this is the case.)

You can define which set of SLKs appear on the screen by setting the single-instance descriptor `altslks`. If `altslks` evaluates to `TRUE`, SLKs 9-16 will be displayed when a frame object is opened. `altslks` can appear in the initialisation file, or in menu, form or text frame definitions. You can also define the SLK layout, with the single-instance descriptor `slk_layout`. Two groupings of SLKs are supported: '3-2-3' and '4-4'. '3-2-3' is the default.

The 'Look up' form provides an example of the definition of a SLK:

```
# Define FKey 8 as LIST
# NB. Fkey 3 (SAVE) does the
# same thing
name=LIST
button=8
action=done
```

`name` is the screen label, `button` is which function key and `action` is what to do when this key is pressed.

Co-processing

A feature of FMLI that I wanted to use in the 'Look up' screen, but didn't in the end is co-processing. It consists of five built-in commands.

The `cocreate` command initialises a process and sets up pipes between it and a co-process. `cosend -n` sends information with 'no wait' down the pipe to the co-process. `cocheck` checks the incoming pipe for information; and `coreceive` does a 'no wait' read on the pipe. `codestroy` and the external UNIX command *vsig* clean up afterwards.

When I read about co-processing in the *FMLI Programmer's Guide* it seemed exactly what I needed to look up a name and then display the address and phone number. It could all be done in one screen I thought. Wrong. Or maybe I'm just not a programmer these days. Apart from more `regex` difficulties (separating the name, address and phone number fields — not insurmountable), there was the multiple match problem, which meant multiple output 'pages'.

Perhaps I'll have another try at it some time.

To Continue

`Text.ph.lookup` isn't all that interesting. It's the kludge I came up with after I gave up on co-processing:

```
# Text.ph.lookup
#
`grep -i "$ARG1" phone.nos |\
awk -F"\t" '{ printf "%s\n%s\n%s\n%s\n%s\n%s\n%s\n\n",\
$1, $2, $3, $4, $5, $6, $7 }' > /usr/tmp/lookup`

# Clear message line
`message -p ""`

title="Entries found:"

# frame will disappear when no longer current
lifetime=shortterm

done=`rm -f /usr/tmp/lookup`true

# Display list of names and addresses found
text="`readfile /usr/tmp/lookup`"
```

```
# Height of this screen
rows=15

# Width of this screen
columns=`longline`

# do not wrap (re-format) text
wrap=FALSE

# Display message
`message "Press CANCEL to return."`
```

It does introduce the FMLI built-in command `readfile`, and also `longline` which I use to set the width (number of `columns`) of the frame.

Moving on, the 'Delete a Name or Number' form is essentially the same as the 'Look up' form so I won't bore you with it here. Its `done` descriptor opens the menu `Menu.ph.del2` so that in the case of multiple matches against the name you type in, you can select which one you really want to delete. The code for `Menu.ph.del2` is:

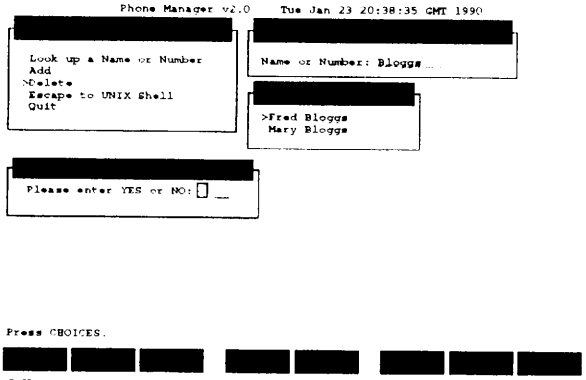
```
# Menu.ph.del2 - found possible entries to delete
#

menu="Matching Entries are:"

# where to get help if the user presses the HELP key
help="`message Press RETURN to select entry, else CANCEL`"

# Create menu items
`grep -i "$ARG1" phone.nos | regex '^([^         ]*)$0.*$' '
        name=$m0
        action=OPEN FORM Form.ph.dsure "$m0"
        itemmsg=Press RETURN to select.'`
```

It does something fancy with `regex` to dynamically create the menu items. I freely admit that I don't understand how this piece of code works, but it is modelled on an example in the *FMLI Programmer's Guide.*

Finally, I open the form `Form.ph.dsure` which asks you to confirm that you want to delete a name and address:

```
# Form.ph.dsure - Are you sure?
`message -p ""`

# Title of this form
form="Are you sure?"

# Where to get help
help=`message "Press CHOICES then DELETE, or CANCEL to return."`
```

```
done='set -l YESNO="$F1" ;
        shell "
                if [ \"$YESNO\" = \"YES\" ]
                then
                        grep -v \"$ARG1\" phone.nos > /usr/tmp/phone.nos
                        mv /usr/tmp/phone.nos phone.nos
                        echo \"Entry deleted. Press CANCEL to return.\"
                else
                        echo \" False alarm: entry has *not* been deleted.\"
                fi
        " | message 'update

# Input field
#
name="Please enter YES or NO: "
nrow=1
ncol=1
frow=1
fcol=25
rows=1
columns=3
fieldmsg="'message -b Press CHOICES.'"
rmenu={ YES NO }
choicemsg="Press SAVE or DELETE to delete this entry. CANCEL to return."
menuonly=TRUE
value=

# Define FKey 8 as DELETE
# NB. Fkey 3 (SAVE) does the same thing
name=DELETE
button=8
action=done
```



There's a couple of interesting things here. First, instead of going through all that `regex` stuff again in the `done` descriptor, I use the built-in `shell` command to invoke the standard UNIX *shell* to do an `if-then-else-fi`. The output of that is then piped into the FMLI built-in `message` command.

Second, I use `rmenu` in the definition of the input field to provide a restricted range of values (in fact, either 'YES' or 'NO') that can only be selected by pressing the CHOICES key.

The conclusion

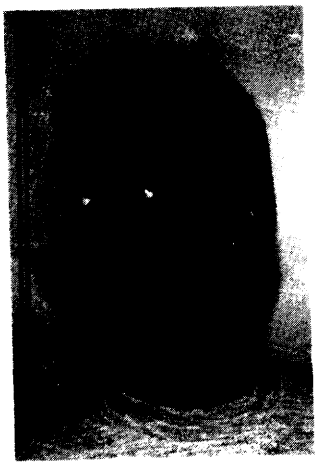
I like FMLI. Sure I found it irritating, even exasperating at times — the currently incomplete set of input-output redirection operators, all those quotes, `regex` and the lack as yet of a `fmligrep`. But a lot of my short-tempered exasperation was the fairly natural consequence of simply not knowing *how* to do things — which is to be expected when you're learning a new language. FMLI does make the job of building form and menu-based applications on character terminals significantly easier. It also channels you implicitly towards a consistent 'look and feel' that is compatible with X.11 and the OPEN LOOK graphical user interface. That has got to be good news for users.

I'll post the code for *phone_mgr* to eunet.sources so if you're interested, look for it there.

Addendum

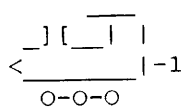
I've just remembered that I also meant to tackle the question of how you integrate home-grown FMLI applications with FACE. Briefly, yes it is possible and it is easy to do.

EUUG Software Distribution



Frank Kuiper
euug-tapes@EU.net

Centre for Mathematics and Computer Science
Amsterdam



The new year is still very young, but a lot of things have happened already: the prices have gone up; two distributions have a new release; and I'm contemplating the use of new distribution media.

Even CWI, my employer who generously allows me to fiddle with the EUUG Distributions, feels the pressure of inflation. But, don't freak out yet, because the price increase is no more than ten guilders per distribution.

Well, after this initial shock, now for the good news, which most of you will now be ready for. There is a new release of the X Windowing software: release 4.
This X11R4 (EUUGD15) is now so big (60 Mb in compressed form!) that I could barely fit it on one 600 ft cartridge, and had to use two 2400 ft 1/2" tapes. This has made the price of this distribution higher, unfortunately.

The other new release is for EUUGD14, aka ISODE. A few days before I wrote this article, I received version 6.0. For those of you who recently received version 5.9 from me, if you want the 6.0 version (which only slightly differs from 5.9), please return the tape, with a small note stating your wish (and don't forget to add a return address!). A few weeks ago I decided to not wait any longer but ship the 5.9 release, as I did not know the exact release date for version 6.0.

And now for something completely different. With the growing popularity of 8mm tapes (e.g. Exabyte, Gigastore, etc), it occurred to me that some of you might be interested in receiving the EUUG distributions on such a tape. My idea is to put almost all the (public domain) EUUG distributions on one tape (this excludes EUUGD1-EUUGD4), thereby giving you some 500 Mb of useful (:-) material, conveniently on one tape. I'm not sure yet what the price of this might be, but it will be in the order of DFl 1000,- (yes, one-thousand guilders). Which is about half the price compared to purchasing the distributions separately.
My question to you is whether you would appreciate the ability of using 8mm tapes and what you would like to have me put on the tape: like I suggested, as much as possible, or selected items like for example the X distribution in uncompressed format. Use my e-mail address (euug-tapes@EU.net), or any other means that you find appropriate and easy.

For the EUUG-Munich conference, April 23-27 this year, we plan to have a conference tape again. A very serious one this time compared to the Vienna "Games" tape. The theme of this tape will be "Graphics". If you have special request items, please contact me and I will try to get a hold on the software. But do this as soon as possible, as the deadline for this distribution (beginning of March) is comming rapidly closer.
If you have other things you would like to see become (part of) a EUUG Software Distribution, please tell me so. I know that some people would like to have the X Test Suite (which is 13 Mb in compressed form, and did not fit on the distribution tape anymore, sorry), the latest version of Kermit, etc. And although it is rather time consuming for me to try and keep updates of every program ever distributed on an EUUG tape, I will try to do so, especially with items like X, GNU, ISODE and ET++.

That's it for now. Below you'll find the list of currently available tapes and how to order them.

As always, anyone is invited to make their own tools, games, etc available for publication on an EUUG tape. Please contact me for more details. Don't hesitate, just put the results of many nights of serious programming and hacking in the public domain, and you might even become famous!

This is a list of all the current (January 1990) EUUG software distributions. It is a short description of the available tapes. Any changes to the contents of the tapes, as well as announcements of new tapes will be placed in the EUUG Newsletter. I am working on a method so you can easily by, e-mail, find out which programm is on which distribution. For the moment you will have to e-mail, call, or write me to find out.

Prices of the tapes are in Dutch guilders (DFl), and do not include VAT-taxes. Prices do include postage cost for surface mail within Europe. Any special shipment costs, like with DHL, will be billed through.

The first price listed with each distribution, is for 1/2", 9-track, reel tapes in tar 1600 bpi format, the second one is for distributions on 1/4" cartridge tapes in (Sun) QIC-24 format. Prices for 800 bpi reel tapes and QIC-11 cartridges may differ from the ones listed.
Note that you have to be an EUUG member (or a member of a local UUG) to obtain tapes at list prices. Non-members will have to pay an extra DFl 300,- per tape.

EUUGD1 R6: UNIX V7 system, specially made for small DEC PDPs (11/23, 11/34, etc). The Kernel supports the UK terminal driver. V7 source licence minimum.
Price: DFl 130.-/190.-

EUUGD2: Early Pascal compiler of the Free University of Amsterdam. V7 source licence minimum.
Price: DFl 130.-/190.-

EUUGD3 R3: Currently not available. We're working an a new tape, dubbed "Starter Kit", containing public domain news and mail programs.

EUUGD4: Software tools, sampled by the Software Tools Users Group. Most of the software is written in Ratfor, for which a Fortran support tool is included. This tape is available in different formats: DEC RSX, DEC VMS, UNIVAC, IBM MVS, UNIX tar, MIT line feed format, and MIT card format (80 columns).
Price: DFl 160,-/190,-

EUUGD5: A collection of benchmark programs made up by EUUG.
Price: DFl 70,-/190,-

EUUGD6: (USENIX 83.1) USENIX tape, containing contributions from various UNIX System Group Members. This is a licence dependent distribution: V7, V32, SIII, V6 or no licence disclosure available.
Price: DFl 250,-/310,-

EUUGD7: UNIX|STAT Version 5.2. A collection of about 25 data manipulation and analysis programs written in C by Gery Perlman.
Price: DFl 70,-/190,-

EUUGD8: A collection of useful software, based on the so called Copenhagen tape (EUUG UNIX conference Autumn 1985).
Price: DFl 130.-/190.-

EUUGD9: A collection of useful software, based on the so called Florence tape (EUUG UNIX conference Spring 1986).
Price: DFl 160,-/190,-

EUUGD10: MMDFIIb. Multichannel Memo Distribution Facility (version IIb). This is a powerful, domain oriented mail system with access control and the ability to communicate over a variety of network systems including TCP/IP, JANET, UUCP, PHONENET, etc. It has been ported to a variety of UNIX's including but not limited to 4.[123]BSD, 2.9BSD, System III/V on a variety of different hardware. You should first obtain a licence agreement by

sending a message to euug-tapes@mcvax. Return the signed licence with your order.
Price: DFl 100,-/190,-

EUUGD11: This is the 'Boat' tape; the Helsinki EUUG 1987 spring conference. It contains about 25 Megabytes of programs, games, etc. Including: jove, less, nag, news, rn , uEmacs, uuencode and larn.
Price: DFl 130.-/190.-

EUUGD12: This is the Dublin EUUG 1987 autumn conference tape. It contains about 26 Megabytes of programs, games, etc. Including: copytape, crc_plot, fastgrep, jove, kermit, notes, uupc, nethack, cron, sendmail, mh, Recipes, brl-gw, isode, pcip, pctelnet.
Price : DFl 130.-/190.-

EUUGD13: The latest conference tape for the London EUUG 1988 spring conference tape. It contains things like: cake, chat, config, copytape, graphedit, kermit, little-st, mcc, mstools, news, pd-diff, pdtar, perl, postscript, psfig, pshalf, shar, rpc, moria4.85, omega, arc, backup, smail, sush, watcher, and much, much more.

Price : DFl 130.-/190.-

EUUGD14: →NOW A NEW VERSION← This is version 6.0 of this non-proprietary implementation of some of the OSI parallel protocols suites as defined by the International Organisation for Standardisation (ISO), the International Telegraph and Telephone Consultative Committee (CCITT), and the European Computer Manufacturer's Association (ECMA).

This release is coded entirely in C, and is known to run under the following operating system without kernel modifications:

BSD
4.2 and 4.3
Ultrix
AT&T
UNIX
SVR2 and SVR3
AIX
HP-UX
ROS
Pyramid OsX

Since a Berkeley UNIX system is the primary development platform for ISODE, the documentation and source are somewhat slanted towards that environment. The tape contains some 12Mb of both tools and documentation in machine readable form. EUUG will send you a tape only.
Price: DFl 130,-/190,-

If you want the complete documentation on paper (some 800 pages!) with the tape, you will have to order this distribution as follows:

Send a cheque or a purchase order for 200 Pounds Sterling to:

Department of Computer Science
Attn: Soren Sorensen
University College
Gower Street
London, WC1E 6BT
United Kingdom
Telephone: +44 1 387 7050, extension: 3680

Specify either 1600 bpi 1/2-inch reel tape, or sun 1/4-inch cartridge tape. The tape will be written with tar format and returned with a documentation set via DHL. Do not send tapes or envelopes. Documentation only is the same price.

EUUGD15: Here it is! The complete X11 Windowing system, as distributed by MIT, release 4: X11R4. Do to the vast growth in user contributed software, this distributions now totals 60 Mb in compressed form. This results in two 1/2", 9-track tapes, and one 600 ft, Qic-24, 1/4" cartridge. This includes the core system, as well as much, very much user contributed software.
Price : DFl 260.-/200.-

EUUGD16: This is the Brussels EUUG 1989 spring conference tape, and consist entirely of software from the GNU project from the Free Software Foundation (not to be confused with OSF :-). On this tape you will find: ispell, g++1.31, awk, gcc-1.33, gdb-3.1, Cscheme, emacs, lisp-manual, libg++1.32, binutils, bison, ghostscript, gas-dist, gawk2.02, gnews2.0, gnuchess, make3.27, oops-2.2, pace, ps-emacs, scheme, sed-1.01, tar-1.04 and torture.
Price : DFl 130.-/190.-

EUUGD17: This tape contains the software for ET++. From the abstract of the "Autumn 1988 EUUG Conference Proceedings":
"ET++ is an object-oriented application framework implemented in C++ for a UNIX environment and conventional window system. The architecture of ET++ is based on MacAPP and integrates a rich collection of user interface building blocks as well as basic data structures to form a homogeneous and extensible system."
It totals some 18Mb of software that the people of the Institut fuer Informatik of the University of Zurich were so kind to let us, mere mortal souls, play with. Have fun.
Price: DFl 130.-/190.-

EUUGD18: This is the "Vienna EUUG 1989 autumn conference tape", and consist entirely of games! There is a SUN specific set, a set for the X Windowing System environment, and a general usable set. All the games supplied are working, and have been tested at CWI by our "Games Keeper <play@cwi.nl>". For many games he added additional features, not found in the originals.
Some of the games included are: for SUN: Asteroids, Mahjongg, Othello, Qix, Sdi, Tetris. For the X environment: Xtrek, Xgo, Xwanderer, Xrobots. General games: Nethack, Adventure, Atc, Empire, Reversi, Yahtzee, Trek73, Backgammon, Corewars, MazewarsV, Vtrek, and lots, lots more.
If this doesn't bring some fun back into using computers, I don't know what else can :-)
Price : DFl 130.-/190.-

EUUG Software Distributions Order Form

If you want to order any tape, please write to:
EUUG Software Distributions
c/o Frank Kuiper
Centrum voor Wiskunde en Informatica
Kruislaan 413
1098 SJ Amsterdam
The Netherlands

For information only:
Tel: +31 20 5924121 (or: +31 20 5929333)
Fax: +31 20 5924199
Telex: 12571 mactr nl
Internet: euug-tapes@EU.net

Please note that for distributions D1, D2 and D4 a copy of your source licence agreement with AT&T for at least UNIX version 7 should be enclosed. Note also that you have to be an EUUG member (or a member of a national UUG) to obtain tapes at list prices. Non-members will have to pay DFl 300,- per tape extra as handling fee. Please enclose a copy of your membership or contribution payment form when ordering. Do not send any money or cheques, you will be invoiced.

All 1/2", 9-track, reel tapes come in tar format, 1600 bpi. 800 bpi is possible on request. Cartridge tapes come in tar format, written with dd, with a blocking of 126b. This is a so-called QIC-24 format, written on a Sun. QIC-11 is available on request.

This page may be photocopied for use.

Name: .....

Address: .....

.....

.....

I would like to order the following:

.....

.....

.....

EUUG (or national
UUG) membership form enclosed? Yes / No
Copy of AT&T source licence enclosed? Yes / No

"I declare to indemnify the European UNIX systems User Group for any liability concerning the rights to this software, and I accept that EUUG takes no responsibilities concerning the contents and proper function of the software."

Signature: .....

Date: .....

AT&T Column

Gill Mogg
gill@uel.uucp

Unix Europe Limited (UEL)
International House
Ealing Broadway
London W5 5DB

Gill Mogg is in Market Communications at Unix Europe Limited.

The guest writer this issue is Susan J picus.

Susan is Head pf Programming Languages Department, AT&T Unix Software Operation, USA.

C++ and Object Oriented Programming

Introduction to Data Abstraction

One of the major problems faced by programmers is that, in the software world, maintenance and modification is a fact of life. It has been estimated to account for as much as eighty percent of the total life cycle. Programmers are faced with changes of all descriptions - from requirements, to data representation, and to hardware. Traditional programming languages that focus on algorithms and building programs as a collection of functions do not always facilitate the implementation of such changes.

Data abstraction, on the other hand, focuses on the formation of data objects and the operations that operate on those objects. Object Oriented Programming is based on the model of building programs as a collection of data abstraction facilities. As a result, the time requirement for making changes to a program designed using object oriented design techniques has been shown to be substantially reduced.

What Are The Benefits Of An Object Oriented Approach?

Object Oriented Programming allows improved productivity primarily through software reuse. Using object oriented methods makes it significantly easier to reuse rather than rewrite programs, saving a tremendous amount of programming time not only in the development but also in the testing and proving of the reused code.

Secondly, Object Oriented Programming promotes better management of the development process. It encourages programming by the refinement rather than reinvention of existing data objects. In addition, it facilitates the development of more readable source code, through the use of overloaded functions and operators that allows developers to model the conceptual basis of data rather than traditional functional specifications, and through overt references to data objects.

Object Oriented Programming provides greatly enhanced portability characteristics by allowing developers to encapsulate the external description of objects. Finally, better tools can be provided to accelerate and ease the process of object oriented design.

A Note On C++

C++ was developed by Bjarne Stroustrup at AT&Ts Bell Laboratories in 1978. The language extends C by providing support for data abstraction facilities and object oriented programming. In 1984 AT&T released the so-called 'Release E' to educational institutions, and in 1985 made its first commercial release of the product. In 1986 Dr. Stroustrup published a C++ text which aroused wide general interest in the language. There were a number of OOPSLA conferences on Object Oriented Programming, Usenix introduced C++ workshops, and there was wide and increasing use of the language amongst a large number of commercial users. In Summer 1989, AT&T offered Release 2 of C++ as a fully supported source product.

The industry's acceptance of C++ is reflected in the large number of people interested in standardising the language. Interest has grown particularly in the last two years - ANSI for example formed the X3J16 subgroup whose first meeting was held in December 1989. The subgroup is using Bjarne Stroustrup's reference manual as the basis of their work on a C++ standard. ISO has also started formal standards activity, although initially it will monitor the ANSI X3J16 committee. UNIX International has also formed a special interest group, and X/Open has expressed interest, though for the moment it is also monitoring the other standards bodies.

Advantages of C++

C++ allows an easy transition for C programmers. Many C programmers start out using C++ merely as a 'better C' and later grow into using the data abstraction facility and some of the object oriented design facilities. C++ gives the programmer the option of using or not using these facilities as (s)he becomes more comfortable with them. C++ also preserves an investment in C software. C++ is link compatible with C, allowing applications to contain a mix of C and C++.

C++ is available on a wide variety of hardware systems, and most code is easily portable across variants. It supports multiple paradigms in the sense that application developers can opt to use more or less of its object oriented facilities.

C++ preserves the run time efficiency of C. In a number of applications that require very low level access similar to C, it is possible to achieve very high efficiency with C++.

C++ also provides strong type checking. When the ANSI C Committee introduced type checking for C, they borrowed most of their concepts straight from C++.

The language also supports inline functions, enabling the developer to avoid the overhead of a function call at the expense of some space trade offs. It provides for the specification of default arguments, and allows one to overload both functions and operators. In this way the developer is better able to model the conceptual data. At AT&T we have found that C programmers are able to start using the above features in almost no time.

Data abstraction facilities are provided through the notion of user defined objects or classes. Users define within a class the member functions or friend functions that are allowed to reference the data. There are also facilities for automatic initialisation and clean up of class objects.

Finally C++ offers object oriented facilities. Some of the object oriented facilities that are provided include inheritance and multiple inheritance, where data objects are derivable from existing objects, and the dynamic binding of virtual functions at run time.

To derive the full benefits of object oriented programming, a programmer must learn to approach a program in a different way from thinking of the algorithms. By allowing the programmer to ease into the world of OOP at their own pace, C++ has proved to be relatively easy to introduce, yet has measurable impact on productivity and maintainability, as seen in the following section.

C++ — Some Experiences Of The Language

C++ has been used on a wide variety of applications within AT&T, in such areas as telephone and networking products, systems software products, and the development of internally used tools. Currently there are over a hundred projects using C++, ranging in size from

a few people to about 75. These projects operate on a wide variety of platforms, from PCs to workstations, mini computers, Amdahls and Crays within the company.

One project, involving around 45 programmers, is a telephone Operation Support System. This project had been using C++ primarily as a 'better C' with some data abstraction, but had not originally followed object oriented design methodology. The programmers found that they were able to replace one commercial database system with another mid way through development with essentially no impact. Two senior members of the project redesigned the interfaces and the rest of the programmers had no impact on their productivity. Many of the implementation details were delayed until requirements were firmed up, but most programmers were nonetheless able to continue to design and develop their code from the interface specification.

Programmers on this particular project found also that minimal training was required. Three to four programmers were involved in designing the classes used by most of the rest of the project, so that the training for most of the programmers on the project was straightforward.

Users at AT&T have found the code to be much more modular and easily maintained. The use of data abstraction facilities in C++ allows fixes to be significantly more localised than in traditional programming languages. A second major benefit is that type checking has reduced syntactic errors significantly compared to C.

As mentioned above, C++ allows increased productivity. Because less code needs to be developed, programmers have found that projects can be completed within a significantly reduced time frame. The members of one particular project estimated that they had been able to save forty percent more of the code than would have been possible had they been using C.

Features of C++ Release 2.0

Release 2.0 has been available since June 1989. It offers a variety of enhanced features, including:

Multiple Inheritance
- a class can be derived from multiple ancestors.

Type Safe Linkage
- provides type checking across module boundaries.

Abstract Classes
other classes.
- an abstract definition of an object is used to derive

User Defined Freestore Management - gives the user greater control over data collection, and more compatibility with ANSI C.

Since ANSI evolved in parallel with C++ there were some differences that were syntactic rather than conceptual. When these differences were examined, anything that was not a required difference was changed to be compatible with ANSI C. C++ is not strictly upward compatible with ANSI C, but many of the features are very compatible. Type checking for example is almost identical.

Available with Release 2.0 is a reference manual which provides the de facto language definition. It is this definition to which more and more products are conforming, as reflected by announcements in the trade press.

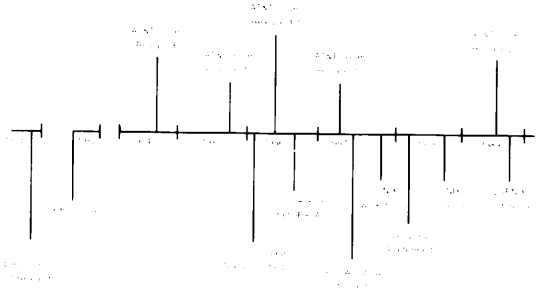
C++ Into The Nineties

C++ will develop in the 1990s into an industry standard product. The focus of activity will move from adding specific new language features to providing object oriented development environments as the language standardises and becomes stable. By the end of the decade, tools and integrated programming environments should be common place to provide for object oriented design.

Summary

C++ has evolved from a research prototype into a mature language suitable for production use. C++ has been shown to improve software and raise productivity, and use of the language is widespread and growing. Industry support of C++ is expanding, and the product should become fully standardised within this decade.

THE HISTORY OF C++



Calendar of UNIX Events

This is a combined calendar of planned conferences, workshops, or standards meetings related to the UNIX operating system. Most of this information came from the various conference organizers, although some was taken from ;login: (USENIX), 13, 1, Jan/Feb 1988, CommUNIXations (/usr/group), VII, 6, Nov/Dec 1987, and the /usr/group UNIX Resources Guide.

If you have a UNIX related event that you wish to publicise then contact either John Quarterman at *jsq@longway.tic.com* or Alain Williams at *addw@phcomp.co.uk* giving brief details as you see below.

Abbreviations:

APP	Application Portability Profiles
C	Conference
CT&LA	Conformance Testing & Laboratory Accreditation
S	Symposium
T	Tradeshow
U	UNIX
UG	User Group
W	Workshop

1990

mon days	conference	location
Mar 5-6	X3J11	New York City, NY, USA
Mar 14	CEN/CENELEC C Procurement	Brussels, Belgium
Mar 26-28	USING C	Dallas, Texas, USA
Mar 26-29	DECUS S	Vasteras, Sweden
Mar 27-30	AFUU C	Paris, France
Apr 9	POSIX APP W	NIST, G, MD, USA
Apr 9-11	USENIX C++ Converence	San Francisco, CA
Apr 23-27	EUUG	Munich, Germany
Apr 23-27	IEEE 1003	Salt Lake City, UT
Apr 26-28	UniForum NZ C	Aukland, New Zealand
May	U 8x/etc C&T	/usr/group/cdn, Toronto, ON, Canada
May 2-4	IETF	IAB, (U. Washington, Seattle, WA), USA
May 7-11	DECUS S	New Orleans, Louisiana, USA
May 16-18	i2u C	Milan, Italy
May 17	NLUUG C	Unix & Parallel Systems, Ede, Netherlands
May 30-Jun 1	UNIX/90	/usr/group/cdn
Jun 11-15	USENIX	Marriott Hotel, Anaheim, CA, USA
Jun 11-15	ISO WG15 (POSIX)	Paris, France
Jun 19-21	EMAP T	London, UK
Jul 9-11	15th JUS S	JUS, Tokyo, Japan
Jul 9-13	UKUUG C	London, UK
Jul 16-20	IEEE 1003	Danvers, MA
Jul 31-Aug 2	IETF	IAB, ?, not in North America
Aug 27-28	Security	Portland, OR, USA
Sept 4-6	GUUG C	Wiesbaden, Germany

Sept 25-28	AUUG Conference	Southern Cross, Melbourne, Australia
Autumn	USENIX C Software Development	
Oct 3-5	UNIX Solutions T	Anaheim, California, USA
Oct 4-5	USENIX W -MACH	Burlington, Vermont, USA
Oct 22-25	IPA C	Jerusalem, Israel
Oct 22-26	EUUG C	Nice, France
Oct 31-Nov 2	UNIX EXPO	New York, NY
Nov 5-9	10th Internat'l C on CC	ICCC, New Delhi, India
Nov 8	NLUUG C	Open Systems, Ede, Netherlands
Nov 15	POSIX APP W	NIST, G, MD, USA
Nov 15-16	16th JUS Symposium	Osaka, Japan
Dec 4-5	JUS UNIX Fair '90	Tokyo, Japan
Dec 10-12	Sinix C, Unix Asia '90	Singapore
Dec 10-14	DECUS S	Las Vegas, NV, USA
Dec 13-16	Sinix T, Unix Pavillion '90	Singapore
Dec 17-19	UKUUG C	Cambridge, UK

1991

Jan 21-25	USENIX	Grand Kempinski, Dallas, TX, USA
Jan 22-25	UniForum	Infomart, Dallas, TX, USA
Feb	Unix in Government C&T	Ottawa, ON, Canada
Feb 18-22	DECUS S	Ottawa, Canada
May	U 8x/etc C&T	/usr/group/cdn; Toronto, ON, Canada
May 6-10	DECUS S	Atlanta, GA, USA
May 20-24	EUUG	Tromso, Norway
Jun/Jul	UKUUG C	Liverpool, UK
Jun 10-14	USENIX	Opryland, Nashville, TN, USA
Sept 16-20	EUUG	Budapest, Hungary
Dec	UKUUG C	Edinburgh, UK
Dec 9-13	DECUS S	Anaheim, CA, USA

1992

Jan 20-24	USENIX	Hilton Square, San Francisco, CA, USA
Jan 21-24	UniForum	Moscone Center, San Francisco, CA, USA
Spring	EUUG	Jersey, UK
May 4-8	DECUS S	Atlanta, GA, USA
Jun 8-12	USENIX	Marriott, San Antonio, TX, USA
Autumn	EUUG	Amsterdam, Netherlands

1993

Jan	USENIX	Town & Country, San Diego, CA, USA
Mar 2-4	UniForum	Washington, DC, USA
Jun 21-25	USENIX	Cincinnati, OH, USA

Organising Bodies

NIST/NBS/POSIX
Roger Martin
National Institute of Standards
and Technology
Technology Building, Room B266
Gaithersburg, MD 20899
+1-301-975-3295
+1-301-975-3295
rmartin@swe.icst.nbs.gov

IEEE Computer Society
P.O. Box 80452
Worldway Postal Center
Los Angeles, Ca. 90080

/usr/group
4655 Old Ironsides Drive, Suite 200
Santa Clara, California 95054
U.S.A.
+1-408-986-8840
+1-408-986-1645 fax

/usr/group/cdn
241 Gamma St.
Etobicoke, Ontario M8W 4G7
Canada
+1-416-259-8122

Tracy MacIntyre
Exhibition Manager
EMAP International Exhibitions Ltd.
Abbot's Court
34 Farringdon Lane
London EC1R 3AU
United Kingdom
+44-1-404-4844

AUUG
P.O. Box 366
Kensington
N.S.W. 2033
Australia
uunet!munnari!auug
auug@munnari.oz.au
+61 3 344 5225

AMIX, c/o IPA
P.O. Box 919
Ramat-Gan
Israel, 52109
+972-3-715770
+972-3-715772
amix@bimacs.bitnet
amix@bimacs.biu.ac.il

Japan UNIX Society (JUS)
#505 Towa-Hanzomon Corp. Bldg.
2-12 Hayabusa-cho
Chiyoda-ku, Tokyo 102
Japan
bod%jus.junet@uunet.uu.net
+81-3-234-5058

UNIX Fair '88 Association
1-1-1 Hirakawa-chu,
Chiyoda-ku, Tokyo 102
Japan

Singapore Unix Association - Sinix
20 Bideford Road #11-05
Wellington Building
Singapore 0922
+65 734 3256

DECUS U.S. Chapter
219 Boston Post Road, BP02
Marlboro, Massachusetts 01752-1850
U.S.A.
+1-617-480-3418

USENIX Association Office
2560 Ninth St., Suite 215
Berkeley, CA 94710
USA
+1 415 528 8649
office@usenix.uucp

National Expositions Co., Inc. (UNIX EXPO)
15 West 39th Street
New York, NY 10018
U.S.A.
+1-212-391-9111
fax: +1-212-819-0755

Sun UK User Group
Sue Crozier
Sun Microsystems, UK
+44 276 62111

USING
P.O. Box 1077
Lisle,
Illinois 60532, USA

UniForum NZ Secretariat
PO Box 585
Hamilton
New Zealand

EUUG National group addresses can be found on the back cover of this newsletter.

Windows Column

William Roberts
liam@cs.qmc.ac.uk

Queen Mary & Westfield College

William Roberts won the 1988 BCS Wilkes Award for a paper on formal specification. He works for Queen Mary & Westfield College, trying to find ways of managing 140 Macintoshes running A/UX. None of the above explains why he is also the author of the Window Systems column for the EUUG Newsletter.

This "Window Systems Column" is more a collection of small odds and ends I'm afraid, but here goes.

X11 Release 4 is out

Release 4 of the X11 window system is now available from various sources on this side of the Atlantic (see below).

For those unfamiliar with X distributions, "X11 Release 4" is the full source code to the "sample server" complete with the device specific parts needed to build it for a number of different systems.

What's in X11 release 4

The release notes for X11 release 4 say the following:

"This is the fourth release of the X Window System, Version 11 from MIT. Substantial progress has been made in optimising the sample server, window manager, and programming libraries. In addition, major improvements to the user interface of several of the key applications (in particular, xmh, twm, xman, and xterm) should make release noticeably nicer to use. Sample implementations of the various new Consortium Standards are included as well as prototype implementations of several efforts currently under development. No incompatible changes have been made to either the core Protocol or to the Xlib programming library. The Xt Intrinsics should be source compatible with the previous release. Changes have been made to the Xaw widget set, but a configuration option for providing backwards compatibility interfaces is available.

Several new sets of fonts have been added: a new fixed width family of fonts, a Kanji and Kana font, the Lucida family from Bigelow & Holmes and Sun Microsystems, a terminal emulator font from Digital Equipment Corporation, and 100 dots-per-inch (dpi) versions of all 75dpi fonts.

This release has been built on the following operating systems: Ultrix 3.1 (both VAX and RISC), SunOS 4.0.3, HP-UX 6.5, Domain/OS 10.1, A/UX 1.1, AIX RT-2.2 and PS/2-1.1, AOS-4.3, UTEK 4.0, NEWS-OS 3.2, UNICOS 5.0.1, and UNIX System V, Release 3.2 (AT&T 6386 WGS). It should work correctly, or with a minor amount of work, on a variety of other systems as well."

There are some 17 pages or so of the release notes, so I'll stop quoting and start paraphrasing: The primary focus of this release has been optimisation of the server and improvements in the key applications; the sample server code has been made smaller, faster and more robust still

(they have been known to run for 3 months now without problems) and lots of bugs have been fixed in the various X libraries. Support has been added for System V (both with and without the STREAMS transport layers), and ANSI C function prototypes have been added to the Xlib and Xt header files; the include files should now also be usable from C++ without modification. The client libraries can be compiled as SunOS 4.0 shared libraries (good news for Sun filestores everywhere!)

A new Consortium standard extension has been added for non-rectangular windows, allowing oval buttons, round clock and all that sort of thing. There are two prototype extensions for handling multi-buffering (for animation) and alternate input devices.

The core (i.e. MIT supported) window manager has changed from the old Ultrix window manager (uwm) to the more popular twm. The xterm, xman and xmh utilities have been improved and supplemented by xditview for previewing the output of ditroff.

The user contributed tapes have swelled from 1 to 3 and include some new games, some new window managers and some user interface toolkits.

Where to get X11 release 4

The most convenient way to get X11 release 4 is to contact J. Watson who is again offering free distribution (as he did for X11 release 3). He has mailed me the following details, which I pass on to you:

> I believe it was you who mentioned in the EUUG newsletter last year that I was offering free distribution of the X Window System in Europe. You might be interested to know that your mention generated quite a bit of interest; I got quite a few requests for tapes from people who told me that they had read about my service in your column. I can't say exactly how many of these there were, but I sent a total of 109 tapes of V11R3, and I suspect that about half of them were requested after your column was printed.
>
> Now that V11R4 is out, I am offering the same service again. So, if you want to mention it in your column again, feel free. The conditions are the same as before: I will include here a brief description of the offer.

I offer distribution of the latest release of the X Window System, currently version 11 release 4, anywhere in Europe. I can only make cartridge tapes, either QIC-24/QIC-120/QIC-150 or TK-50/TK-70. No 9-track tapes, no HP cartridge tapes, no floppy disks (more than one person has asked for floppies over the years).

There is no charge for this distribution: the only requirement is that the tapes be returned. I do not require that tapes be sent in advance. I pay the postage to send the tapes from here; I expect the recipient to pay the postage to send them back. (One person sent back the tapes by Federal Express, *freight collect*. It takes all kinds, I guess.)

The distribution contains quite a bit more than just the MIT release:

— MIT X.V11R4, including core and all contributed tapes

— Official patches issued by MIT, if any

— Speedups/enhancements from reputable sources, if any (i.e. Purdue)

— Other interesting/useful software, and newer releases of software from the MIT contributed tapes, where appropriate. Currently this includes PBMplus and x11perf version 1.2. Sun has promised to send me a new release of XView which they call "R4+"; if it ever gets here, I will include it in the distribution.

— Selected postings from comp.sources.x.

— Source code for compress/uncompress, patch and tar.

The portion of this which is the MIT distribution is *identical* to what is sent out by MIT themselves. I read the MIT tapes and write my tapes with "dd", so I don't risk screwing things up trying to extract/archive everything. When I add newer releases of software that is already on the MIT tapes, I always put it in a separate tape record, rather than trying to merge it into the MIT distribution.

Anyone in Europe who wants to order a tape from me can contact me at:

J. Watson
Adasoft AG
Nesslerenweg 104
CH-3084 Wabern

Tel: +41 31 54.35.70
Fax: +41 62 61.41 30

mcsun!chx400!pan!jw
or jw@pan.uucp

For those of you in the UK with free access to the JANET network, try contacting the info-server at Imperial College, sending a message with no subject line and containing just the lines

request catalogue
topic xv11r4
request end

The distribution is MUCH TOO BIG to be sent by mail, so only people capable of using JANET NIFTP protocols should bother with the info server.

News of NeWS?

From the computing newspapers I read that AT&T UNIX System V.4 is now out, and that this includes X11/NeWS. I've seen X11/NeWS running on a Sun SparcStation 1 with a graphic accelerator and it was very nice indeed. However, if you have only got small Sun 3s then you can forget it (I'd welcome an article from anyone at Sun who'd like to comment on this).

On a smaller note, the MacNeWS implementation of NeWS 1.1 for A/UX is no longer available. Apparently the Grasshopper Group (who were selling it) sold so few in the 2-3 years they have been trying that they have given it up as a bad job.

Finally, Glenn Reid (the man who wrote the Adobe PostScript books) has left Adobe and gone to work for NeXT Inc., presumably on Display PostScript and the NeXTStep user interface.

Line Layout Manager delayed due to late running

When the Apple System 7.0 operating system is released later this year, it will not include the Line Layout Manager that I described in a previous edition of EUUGN (so Apple inform me). The Outline Fonts and new printer handling will be there, but the automatic handling of kerning, special ligatures and so on won't be available until "a future system software release".

A New X11 Users Group

The new European X User Group (EXUG) held its inaugural meeting at the Institute of Electrical Engineers in London, England on the 21st November 1989. Over 200 people attended and there was a related exhibition with 10 companies. The meeting elected Niall Mansfield of Unipalm Ltd as the Chairman of the group (I won a pound bet that Ray Anderson of IXI Ltd wouldn't stand) and Valerie Holt as the Secretary, plus deputy chairman, treasurer, newsletter editor and a committee of 4.

They have already produced a newsletter, and anyone interested in joining EXUG should contact:

The Secretary
European X User Group
Mitchell House
185 High Street
Cottenham
Cambridge
CB4 4RX
England

exug-committee@doc.ic.ac.uk

The proposed date for the next newsletter is 23rd March 1990, and there is talk of a Spring meeting but no date has yet been fixed.

It's All Writs being a Lawyer

More minor excitements over the long running Apple vs Microsoft legal battle about who owns what intellectual property rights in window systems. Apparently Xerox have entered the fray with a suit against Apple (recall that the Apple Lisa took a lot of ideas from work done at Xerox PARC, but that Apple claim to have some sort of licence from Xerox). Someone will get rich out of all this, but it won't be you or I, dear Reader.

A Threat

My working life at present has little to do with window systems; if this column is to continue then it will need considerable input from other people, so get writing.

Puzzle Corner

Mick Farmer
mick@cs.bbk.ac.uk

Mick is a lecturer at Birkbeck College (University of London) and the Secretary of the UKUUG. His interest is in all aspects of Distance Learning and he is the Senior Consultant (Software) for LIVE-NET, an interactive video network connecting London's colleges. He is also a member of the University's VLSI Consortium, mainly because the design tools draw such pretty pictures.

Hello peeps,

Solution to Puzzle Number 8

The reasoning needed to arrive at the unique solution is quite long so I will not reproduce it here. If you would like a copy, send me a mail message. Here is the solution. Note that I did not say that there were only seven sevens in the solution!

```
                          5 8 7 8 1
                  ---------------------
1 2 5 4 7 3 )7 3 7 5 4 2 8 4 1 3
              6 2 7 3 6 5
              -----------
              1 1 0 1 7 7 8
              1 0 0 3 7 8 4
              -------------
                  9 7 9 9 4 4
                  8 7 8 3 1 1
                  -----------
                  1 0 1 6 3 3 1
                  1 0 0 3 7 8 4
                  -------------
                      1 2 5 4 7 3
                      1 2 5 4 7 3
                      -----------
                                0
```

Solution to Puzzle Number 9

The basic number must be such that its square has seven digits and its cube eleven digits. This gives limits of 2155 and 3162 for the basic number (try 2154 and 3163 to see why). The second multiplication indicates that the penultimate digit of the square must be zero. Only numbers ending 00, 01, 02, 03, 47, 48, and 49 have squares with this property. The first multiplication indicates that the first three digits of the base number are below 5, while the last one must be 5 or more. This leads to a four-digit number beginning with the pair 22, 23, or 24 and ending with the pair 47, 48, or 49. Trial and error now yields the solution 2348!

Puzzle Number 10

One year (long ago :-) a number of my students failed their examinations. Two thirds of those failed on Compilers, three quarters on Graphics, and four fifths on Networks. Moreover, 26 failed on all three subjects. If this is the smallest number possible, how many students failed overall?

Puzzle Number 11

At a recent EUUG conference interpreters were hired for translating into Dutch, French, German, Hungarian, Italian, and Portugese. Furthermore, the interpreters had surnames corresponding to the six languages. Each of the six interpreters spoke two of the languages and no two of them spoke the same two languages. Each of the languages was spoken by just two people and none of them spoke the language of which they were the namesake.

Ms Hungarian could speak Dutch and German. Another interpreter spoke Dutch and Italian. Mr French and Mr Dutch, between them, spoke all of the four languages of which neither is the namesake. Of the two languages spoken by Mr Dutch, both the namesakes spoke French. Neither of the German-speaking interpreters had any knowledge of Italian. What two languages were offered by Ms Portugese?

A View of the Organisational Structure of POSIX Standards

Dominic Dunlop
domo@tsa.uucp

The Standard Answer Ltd

Equipped with an undergraduate degree in Electrical Engineering from the University of Bradford in England, Dominic sidled into the world of mini- and micro-computers. From there, he managed to effect an entry into the hallowed temples of Unix, and has hung around there ever since, writing the odd paper, contributing the odd standard, and starting the odd company. He became an independent consultant in January 1989, and his latest company, The Standard Answer Ltd, has just bought an Apple portable as the Unix machine is too heavy to carry around.

Just for a change, there has not been a meeting of the ISO POSIX working group to report for this issue of the newsletter, so I'm taking the opportunity to mount a small hobby-horse — a hobby-horse which, if harnessed, will help to distance POSIX away from its American roots.

This article provides an overview of the way in which the international standards community works, insofar as it affects POSIX and the incorporation into POSIX of internationalisation features. I'm not going to describe the technology underlying internationalisation other than to say that its aim is to make operating system and applications software independent of the user's spoken language and its representation (character sets, collation, text direction and so on). This done, localisations specific to each group of natural languages, users can tailor programs to their requirements without the need for expensive and legally-problematic hacking of source code. (If you want to know more, let me know, and I'll either expand on the topic, or give a few pointers.)

Figure 1 shows the relationship of standards bodies as far as POSIX is concerned. (The picture may look very different for other standards efforts, such as Open Systems Interconnection, but that need not concern us here.)

All standards must originate somewhere, whether in industry, in a professional association, in a national standards body, or in an international standards body. In the case of the POSIX family of standards, the Institute of Electrical and Electronic Engineers has assumed responsibility for the initial production of the documents. The IEEE is a professional association which is open to qualified engineers, no matter what their nationality. (It is not, as many people both inside and outside the U.S.A. believe, a solely North American organisation.) It has been involved for many years in the production of consensus standards — that is, standards arrived at through a formal process which gives ample opportunity for any interested party to comment and vote on proposals.

According to the standards procedures of the IEEE, the main group of interested parties is its membership, although non-members are also allowed to participate. Unusually among standards bodies, voting on IEEE standards is nominally "one member, one vote". (More typical standards bodies vote by corporation or by country.) The exception to the IEEE's individual voting scheme is that institutions can also participate, provided that they represent a broad constituency, rather than a single narrow commercial interest. Currently represented on the

POSIX effort are the Open Software Foundation, UniForum, UNIX International, Usenix and X/Open. None of these is an official standards body, although all are involved in the production of materials on which future standards may be based. In some cases, the organisations produce documents which look and smell like standards but which, because they are not produced by an open (and slow, and legalistic) consensus process, may well show some bias towards the interests of the originating organisation. Known broadly as industry standards, these documents appear before consensus standards, and must subsequently be brought into line if a consensus standard is to succeed.

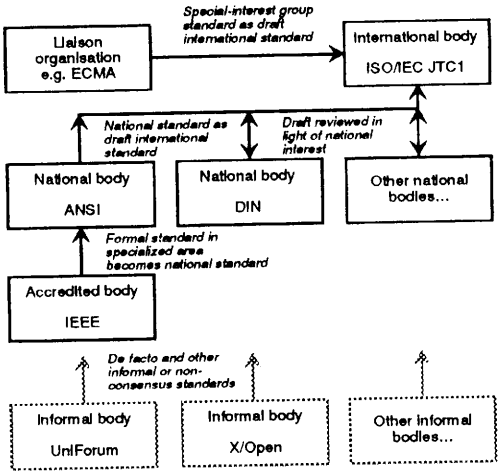


Figure 1

As figure 1 shows, in the hierarchy of standards organisations, the IEEE is near the bottom. Above it is firstly the national level, then the international. As the IEEE is based in the U.S.A., it has gained accreditation from the U.S. national standards body, ANSI (the American National Standards Institute). This means that ANSI considers the IEEE competent to produce national standards on behalf of ANSI. Of course, accreditation by ANSI gives rise to an anomaly: the IEEE, through a democratic process potentially involving an international membership, is creating national standards for the U.S.A. I shall return to this issue later.

ANSI, in turn, is a "member body" of Joint Technical Committee 1 (JTC1), an international standards body formed jointly by the International Organisation for Standardisation (ISO) and the International Electrotechnical Commission (IEC) to handle the standardisation of information technology. ANSI's role in JTC1 is nominally to represent U.S. interests in the "one nation, one vote" process by which international standards are ratified. Other member bodies such as DIN (West Germany), JISC (Japan) and IRISI (Iran), play a similar part, making sure that no standard conflicts with their own national interests.

Member bodies may sponsor draft standards at the JTC1 level. In the case of POSIX, ANSI is the sponsor. The international standards community expects that a draft standard sponsored by a national member body in this way is likely to show a bias towards the needs and culture of that member body, and so may require amendment and perhaps extension before it is suitable for adoption as an international standard. Certainly, both POSIX and the C language have come in for criticism at the international level for their lack of support for non-Roman alphabets.

In order to root out and correct any bias or omission in a draft standard sponsored by a particular member body, other member bodies are expected to pore over the proposal, and feed in changes which reflect their national needs. Obviously, this could take forever: approaching a hundred countries are represented on JTC1. Typically, the number of member bodies participating in a particular standards effort is limited, and of these few play a very active role. In the case of the POSIX effort, around a dozen member bodies are circulated with the working group's paperwork, and of these, perhaps half are regularly represented at its meetings. Even so, by the time a national standard has progressed to the level of becoming a JTC1 draft, it is rather late to begin making changes — particularly if, as is the case for POSIX and C, there is a pressing need for an international standard.

As presented so far, the standards world is strictly hierarchical: a standard such as POSIX progresses from an accredited special interest group within a country, firstly to national level, and finally to international status. Officially, it is not until the final stage that interests outside the originating country get to comment on it. The process could be made more efficient if interest groups outside the originating country had a means of commenting at an earlier stage, but the hierarchy seems to preclude such comment.

Interestingly, there is a "side door" at the international level which can be used to short-circuit the normal time-consuming process. The top level of figure 1 shows an organisation in

liaison with JTC1, the European Computer Manufacturers' Association (ECMA), which has gained the privilege of being allowed to propose and comment on standards at the international level. The process of obtaining liaison status is both difficult and lengthy, and is open only to international organisations with a valid claim to representing a specific broad area of interest. (Besides ECMA, the World Health Organisation and Mastercard International are among the sixty or so bodies in liaison with JTC1.) If the members of a liaison body can formulate a standard which is useful to them, liaison status allows that standard to be proposed for adoption as a formal international standard. Since all bodies with such status are themselves international (or at least regional), such proposals are likely to satisfy international needs without much need for amendment. (ECMA has sponsored several standards for magnetic media in JTC1; banking interests have been active in the standardisation of credit cards.) Indeed, JTC1 has developed a "fast track" approvals mechanism for use when member bodies agree that little review is necessary — although it has to be said that not every use of the fast track has resulted in a standard being approved.

The strict hierarchy imposed by ISO makes for easy and obvious management control, but is under some strain. Firstly, where emerging standards seek to accommodate international needs from their first drafting, the late review by national member bodies provided by ISO makes for unnecessary delay — delay which could be avoided if national bodies had an official means of providing input at an earlier stage. Secondly, regional standards organisations — most notably CEN, the European Standards Centre — are growing in importance, and do not fit well into a scheme which is set up according to strictly national guidelines.

These two problems combine to foster provincial attitudes on the part of standards makers — and politicians — involved with POSIX both inside and outside the U.S.A. Those inside reason that, since they are creating a U.S. national standard, international considerations are relatively unimportant, and can be left for later. Outside the U.S., standardisers reckon that it will be so long before they can mold a U.S.-produced standard to their own requirements that they might as well develop their own, probably incompatible, standards to fill their immediate needs. In Europe, a proposal to adopt issue 3 of X/Open's Portability Guide (XPG3) as a standard was strongly backed for a while, even though XPG3 is not wholly aligned with POSIX. (On the reasonable grounds that the 1003.1 standard had not been approved at the time of publication. XPG4 will be aligned with POSIX.) Interestingly, just as the IEEE is seen in Europe as representing U.S. interests, X/Open is seen by many U.S.-based observers as a European outfit, despite its many U.S. members.

Provincial attitudes among technical people and their managers outside the U.S.A. exacerbate the problems. Although the IEEE makes some effort to reach this constituency by holding one of the quarterly working group meetings outside U.S. every couple of years, the majority of attendees are always Americans. Europeans in particular seem, even if they have the inclination to attend, to find it difficult to justify the expense to their management. The interests of Arab countries and the Indian subcontinent are seldom represented at all. In contrast, delegates from Japan and other Pacific rim countries have been attending meetings in increasing numbers, even when lengthy and costly travel is involved.

Given the current structure of the international standardisation community, is it possible to work within it and yet overcome the two problems which face the POSIX effort: that of obtaining useful international input at an early stage; and the parallel problem of preventing divergence between POSIX and emerging industry, national and regional standards? Can the current structure accommodate formal mechanisms which provide for solutions, or will the problems remain unless the structure itself is changed?

Until now, practical international input to POSIX has come from two sources which are not a part of the formal hierarchy of international Standardisation: UniForum and X/Open. As I have already mentioned, X/Open is an international grouping seen by some as primarily European: its active membership has to date consisted of computer suppliers. UniForum, which was known as /usr/group until 1989, is a grouping of hardware suppliers, software authors, value-added resellers, and users. As with X/Open and other groupings, it is the suppliers which have played the largest part in the organisation — users have seldom made their voice heard. UniForum is U.S.-based, but has affiliates around the world.

These affiliates are largely autonomous, and, despite efforts to involve them, have played almost no part in Uniforum's standards activities — even when these are involved with internationalisation. (While UniForum's Technical Subcommittee on Internationalisation has active participation from outside the U.S.A., the people concerned became involved directly, rather than through their local UniForum affiliates.) USENIX, the other user grouping with institutional representation to the IEEE POSIX project has a better claim to providing a forum for users, but is almost exclusively North American, and, unlike UniForum, has no internal structures concerned with standardisation. The European UNIX systems User Group (EUUG) has a truly pan-European membership made up, like that of USENIX, primarily of computer programmers and technical users, but has not participated officially in any standards effort. Its involvement to date has been confined to the co-sponsorship with USENIX of a standards monitor service, which provides members with information about progress on POSIX and in related areas.

It is my view that, if international interests are to play a greater part in the drafting of POSIX standards, they must be represented formally within the IEEE. This is not to minimise the importance of the work done by UniForum, but rather to say that an official stamp of some sort is necessary in order that its importance receives a wider recognition both inside and outside the IEEE. Unlike other topics handled in the past by UniForum, real-time and transaction processing among them, internationalisation has never officially been incorporated into the POSIX effort because it cannot stand alone. There cannot usefully be such a thing as a standard for internationalisation: rather, internationalisation should be a consideration in the drafting of any standard for computer software.

The 1003.0 (POSIX Guide) working group is currently wrestling with the problem of handling internationalisation issues within POSIX. It may be possible to borrow a useful concept from ISO: that of the rapporteur group. Rapporteur groups cut across normal boundaries, bringing together those who are interested in some problem or activity which is common to a number of standards projects.

It is over-optimistic to hope that bringing internationalisation officially into the POSIX fold will result in immediate participation by those who currently wait until documents reach the ISO level before commenting through their national member bodies. One way to reach this audience might be to convince it that the IEEE is indeed an international, rather than strictly North American, grouping. A radical way of achieving this would be for the IEEE to seek liaison status with JTC1, so obtaining a means of submitting base documents directly, instead of through ANSI. To do this would involve the IEEE in the considerable expense and logistic complexity of sponsoring standards — a task for which resources are not currently in place in an organisation which seldom gives the appearance of being over-endowed with resources.

In any event, even if the IEEE were to apply for liaison status tomorrow, it would be a long time before it was granted. Unless or until this happens, it seems to me that it is the duty of user groups around the world to to encourage their members to play a part in the process through the IEEE. So that's what I've been doing in this article!

Instructions for EUUG Authors and Speakers

Stuart McRobert
Jan-Simon Pendry
Dave Edmondson

Imperial College
London

ABSTRACT

A list of instructions for authors of papers at EUUG technical conferences and speakers at those conferences is presented. Details of preparing papers for inclusion in the Proceedings and of preparing visual aids for use during the talk are discussed. Careful adherence to the instructions offered in this document will result in an improved interaction between submitters and conference organisers, to everyone's benefit.

Introduction

To help ensure a more uniform look to the Conference Proceedings, authors are requested to adhere to the following guidelines in preparing their *troff*-ready copy.

⇒ ALL SPEAKERS MUST PROVIDE A PAPER FOR THE PROCEEDINGS BY ⇐

⇒ **SATURDAY, JUNE 30th, 1990** ⇐

If we do not receive a copy of your paper in electronic or hardcopy form by the deadline indicated above, your talk will be struck from the conference and you will not be permitted to speak.

Troff-Ready Papers

The following instructions should be followed carefully when preparing your *troff*-ready copy.

Avoid excessive font changes. If you haven't used *troff* before, please experiment before making the final run, and consult your local *troff* and stylistics gurus.

If English is not your native tongue and you would like the EUUG to help check and possibly correct any difficulties, please do ask. Please submit early so that the paper can be read through and any suggestions returned to you.

Finally, please use good taste in any event.

Macro Package

The whole Proceedings will be produced using the *–ms* macro package. This gives the final book a consistent look which we believe is desirable. Papers which are not sent using the *–ms* package will simply be edited to conform. It will help speedy and accurate production if authors submit using this form.

If you do not have *–ms* but have *–mm* instead, then use that. We will then convert your paper to use *–ms*, so please **only** use the simplest formatting commands.

Authors are requested to use the following title layout macros,

.TL
Title of the paper (one or more lines)
.AU
Author(s) (may also be several lines)
.AI
Author's institution(s)
.AB
Abstract, which should be **no** more than half a page long.
.AE
Abstract End indicator.

Several interspersed .AU and .AI lines can be used for multiple authors, and a short abstract is **required.**

The paper should be set in single column mode without any extra headers or footers, since these will be produced by us throughout the Proceedings.

Do use†

√ Headings (.NH, .SH).

√ Left paragraphs (.LP, **not** .PP).

√ Indented and quoted paragraphs (.IP, .QP, .QS, .QE).

√ Italic and bold text (.I, .B).

√ Footnotes (.FS, .FE).

√ Displays, equations, pictures and tables (.DS, .DE, .PS, .PE, .EQ, .EN, .TS, .TE, .TH).

√ Keeps (.KS, .KE, .KF).

√ Use numbered figures inside floating keeps, rather than referring to an immediately following figure.

√ **Short** input lines with a newline after punctuation marks. Long lines stand a good chance of being truncated by various mail systems.

√ Remove all spaces at the end of lines since they confuse troff. If you do not put a newline after punctuation marks, remember to put exactly two spaces after full stops and exclamation marks and exactly one space after other punctuation characters and between words.

√ *refer* to prepare references for inclusion in your paper. This is important since all references are now produced in the same style throughout the conference proceedings. In the example below (taken from one of the past conference papers), we illustrate how to specify a reference and then how to refer to it. First of all the specification.

```
%A Robin Faichney
%T Dp: a System for Inter-Program Communication
%B Proceedings of the EUUG Spring 1989 Conference
%P 207-215
%D April 1989
%C Brussels
```

† Relevant *-ms* macros are in brackets.

and now a sample piece of text which refers to it.

```
The earlier paper
.[ [
faichney inter-program communication
.]],
contains an abstract analysis...
```

This would then appear as...

> The earlier paper [Fai89a], contains an abstract analysis...

The references will be collected together at the end of your paper and listed in in alphabetical order by author. We normally use `refer -k -e -n -l3,2 -s` to achieve this. We've included the above reference at the end of this paper to help illustrate this. If you do not have specific references you may include a bibliography at the very end of your paper.

√ Trademarks, which should be collected together and placed at the end of the text after all the references. Do not place marks in the text to show trademarks. The setting process will collect all the trademarks together and place them at the start of the proceedings.

√ *Spell -b*, it can find some spelling mistakes and errors.

√ A colleague familiar with your work, but preferably not involved in the production of the paper, to carefully read it through and check any technical points and provide you with constructive criticism and encouragement.

Do not

× Change internal registers (line lengths, page offsets, margins, etc.).

× Force page breaks to make the output look better on your output device as the text will undoubtedly fit into pages in a different way when we typeset it.

× Rewrite the macro package. If you need a special macro for something, please include it **fully** documented with troff comments so we can understand it. But remember, all the papers are set together one after the other using the *-ms* macro package, so your macro **must** be compatible and not upset other things.

× Use running titles, headers, footers or other annoyances. Don't worry about the default *-ms* header containing the page number, it will be useful on your hard copy submission.

× Use double quote characters (") to surround strings. Instead, use pairs of left (``) and right ('') quotes, the troff two character names \(lq and \(rq or the *–ms* strings *Q and *U.

× Use the .UX macro. Use the string "UNIX".

× Ignore deadlines.

× Ignore this advice.

The Proceedings will be typeset in portrait orientation (long axis vertically) on A4 pages (210×297mm).

Figures and Tables

All figures should have a reference number, e.g. Figure 1, and preferably a title, centred **under** the figure. This should be made to appear as:

Figure N: *The title of this figure is ...*

Any special requirements for reproducing imagery should accompany the manuscript on a separate markup sheet.

Diagrams should be drawn using *pic* which will allow for direct inclusion into the paper and typesetting. On the other hand, by special prior arrangement, complex images may possibly be included. These should

preferably be supplied in Postscript format or as monochrome Sun screen dump images. In this case, the images will be printed on a conventional laser printer (300 dpi) and inserted by hand. Finally, we may be able to directly include suitable high quality images. Please contact us **immediately** if you are interested in any of these options and we can discuss what can reasonably be produced and how to supply the data.

The production of images will require extra time and your paper will need to be received well before the deadline. Note that the printing process does not include colour plates.

Figures and tables should all be referred to by a specific reference, e.g. see Figure 2, and **not** by location, e.g. see the figure below. To avoid blank spaces, figures and tables should be allowed to *float* through the text using floating keeps — see the `.KF` macro.

When using *tbl* please set the tab character to something other than a TAB, which tends to be expanded when sent through mail systems. For example,

```
.TS
tab(:);
l n n.
Apples:23:45
Pears:3:442
.TE
```

uses a colon instead of a TAB to delimit the columns.

Page Count

We ask that authors limit the length of their papers to 15 pages (surfaces) total. Some dispensations *may* be available, but even at 15 pages per paper the Proceedings may rival many large city telephone books. Again, you need to apply for dispensations, but don't hold your breath.

Submission

The EUUG prefers papers to be submitted in *troff -ms* form by electronic mail using shar. If you do not have access to these macros or cannot send your paper by electronic mail please contact us immediately to make arrangements for submission of your paper.

Do not trust electronic mail to get your paper to the destination. If you have not received an acknowledgement in a reasonable period of time, think the worst and send the paper again. All submissions will be acknowledged on receipt.

The Proceedings will be formatted and then set on a Postscript typesetter (typically 1270 dpi). The following fonts are available **Roman**, *Italic*, **Bold**, Courier (constant width), and Special (please confine your use to the special symbols listed in the standard *troff* and *eqn* guides). Please use Courier font for program examples and listings. If you don't have a constant width font as font C, please indicate constant width font use with two new macros. .CW to start constant width text and .CE to end, returning to the previous font. For an example of Courier font, see the comments about setting the TAB character in *tbl* above.

We prefer to accept submissions using *troff, ditroff, -ms macros, pic, tbl, refer, eqn, psfig*.

To accept electronic submissions, we *must* have the following three things:

- Instructions for typesetting the paper.
- The source input for the paper itself, i.e. *tbl, eqn, troff...* text and commands.
- A hard copy of the paper so that we can verify that what we produce is the same as what you intended.

Everything but the hard copy should be sent via electronic mail to

```
euug-papers@ukc.ac.uk
or
..mcvax!ukc!euug-papers
```

The subject line should read:

```
Subject: title of your paper
```

Followed by,

- Your name.
- Your postal address.
- Your electronic mailing address, relative to mcvax (cwi.nl).
- Your telephone numbers, preferably both day and evening numbers (in international format e.g. +44 1 589 5111).
- Fax or telex numbers are also useful if you have them.
- Commands needed to format your paper, e.g.

```
tbl paper.t | eqn | troff -ms
```

- Any other information that might be useful to the people printing the Proceedings. Remember it is vitally important that we have a way of contacting you quickly if there are any problems with your submission.
- Last, but not least, the actual paper. If you have access to shar, it is desirable to use it. It is a good idea to use the *–a* switch, so that a prefix character added and the data is checked when unpacked.

Please send one **original,** single sided output, and a good photocopy of your paper. Use page numbers! If the imagery requires paste-up, please package the submission so that it will survive the journey. Please do **not** use staples, paper clips or anything else that may damage the papers in transit.

Submit the original hard copy to (including a copy of the information requested above *plus a passport sized black and white photograph* of the principal speaker):

Stuart McRobert
EUUG Conference Proceedings
Department of Computing
Imperial College
180 Queen's Gate
London, SW7 2BZ
UK

Phone: +44 1 589 5111 Ext 5022 Fax: +44 1 581 8024
Phone: +44 71 589 5111 Ext 5022 Fax: +44 71 581 8024 (After 5th May 1990)

and a photocopy to

EUUG Conference Proceedings
Owles Hall
Buntingford
Hertfordshire
SG9 9PL
UK

Phone: +44 763 73039 Fax: +44 763 73255

All papers will be acknowledged.

Content

The matter of publishing licensed material is of grave concern to EUUG. The final copy will be reviewed for such content, and if, in the opinion of those responsible for worrying about such things, the submission contains licensed material, it cannot be printed in the Proceedings. The deadlines are such that it is extremely unlikely one could amend a submission and get it back in time, so look before you send it. If there is **any** question on this matter, please contact Owles Hall immediately.

Your Presentation at the Conference

In general, you should limit yourself to a few important points in your presentation. You don't need to summarise the entire paper.

Visual Aids

By default, one overhead and a 35mm slide projector will be available. If your presentation requires any other equipment, please contact Owles Hall **immediately.** In particular, if you are bringing hardware which must be staged in the hall before your session, we must have the logistics worked out well before the beginning of the conference. Remember that you may also have difficulty with Customs Officials if you are moving equipment across International borders.

If you wish to show any videos, you must provide us with full details well in advance regarding the format and standard used (don't assume anything, there are many standards and problems).

If you are using overheads, please try to typeset them. Use at least 24-point type, which limits you to about 15 lines per overhead, although fewer lines would be better. If you must use handwritten overheads, write carefully using a good, dark pen and stick to the 15 line limit. Typewritten overheads are *not* acceptable.

On 35mm slides, use *very* high contrast. Use similar guidelines for point size as indicated for overhead transparencies. Remember, what previews well in your conference room will not be legible in a 600-seat auditorium.

Have printed copies of your materials when you give your talk so that you don't have to keep looking back at the screen. This is disorienting for both you and the audience.

If you are using overheads, you *must* arrange for someone else to turn slides. You cannot turn slides and be visible at the same time.

Include a title slide with the title of your talk, your name(s), and your electronic mail address(es). Duplicate this slide at the end of your talk so that you can display it during the Question and Answer portion of your presentation.

Avoid putting important information at the bottom of the page. It can be difficult for some people to see the entire screen.

Also please avoid using multilayer overheads, since the use of the "reveal" technique is discouraged unless you are confident that you can do it smoothly and well. Remember that the contents of your hidden overhead can often be faintly seen when only covered by a thin sheet of paper (the projection light is very bright). Use thick card if you really want to hide something, or better still, use another overhead.

Authors are reminded to keep their slides and overhead transparencies in their hand luggage when traveling. Also, a spare copy is always useful.

Timing

Each session is carefully planned to fill **all** the available time. Your presentation must be kept to the agreed-upon timetable otherwise you will be "stealing" time from other speakers. The session chairs will be ruthless with the clock.

Run through your talk at least once with your slides to get an idea of the timing. Remember, it will run somewhat longer in front of an audience than it does in front of the mirror. It is better to give a talk that is too short than too long. At this conference, all regular talks have 30 minute slots, which means you can

talk for 20 minutes. The remainder of the time will be used for your introduction and for answering questions after your talk.

Administrivia

Bring a short (one paragraph), legible biography of yourself written in English to the conference for the session chair to read during your introduction.

Contact your session chair at least a day before your session to ensure that you know where to go, where to turn in your slides, etc. Be sure your session chair knows how to pronounce your name.

You will have received notification of the length of time you have for your talk with the letter of acceptance.

Deadlines

The publication schedule is very tight, and to have the Proceedings available at registration we require your cooperation. To meet the necessary internal deadlines, we **must** have electronic submission in hand on **SATURDAY, JUNE 30th, 1990**. Manuscripts arriving after this date will be at risk.

Accommodation and expenses

Papers often have multiple authors, it is expected that there will be a single principal speaker for the paper. The principal speaker is entitled to free registration to the conference; this includes coffee, lunches and attendance at social events. EUUG is unable to pay for any hotel and travel expenses except in special circumstances. You should contact the Programme Chair as soon as possible (unless arrangements have already been made) if help is required. The attention of students is drawn to the section on Student Grants in the Pre-Conference registration pack.

The speaker is responsible for booking accommodation, please look in the Pre-Conference registration pack for a booking form. In cases of difficulty please contact Mrs. Gibbons at Owles Hall who will be pleased to help.

Sample Reference

Fai89a.Robin Faichney, "Dp: a System for Inter-Program Communication," in *Proceedings of the EUUG Spring 1989 Conference*, pp. 207-215, Brussels, April 1989.

Acknowledgements

Thanks to USENIX for allowing us to base this guide on their own *Instructions for USENIX Authors and Speakers*, by *Eric Allman* (Britton Lee), *Judy DesHarnais* (USENIX Association), and *Evi Nemeth* (University of Colorado).

Revised January 1990 by sm.

The Matrix: Computer Networks and Conferencing Systems Worldwide

The Matrix: Computer Networks and Conferencing Systems Worldwide, John S Quarterman, Digital Press, 1990, DP ISBN 1-55558-033-5, PH ISBN 0-13-565607-9. Price £53.95, Soft Back, 719 Pages pp, Size 25 cm x 18 cm.

This is a book which will be of interest to both new users and to those who think they know it all.

The first part of the book contains general background on networks, protocols, and most elements of communications between computers. If it doesn't tell you about what you want to know it usually has a reference to where to look for more information.

The index is fairly comprehensive and well cross referenced, for example UKC appears as UKC (University of Kent at Canterbury), University of Kent at Canterbury (UKC) and as Canterbury (University of Kent) The index also includes people, networks, protocols and standards bodies.

I would suggest it is the type of book to buy and put on your shelf to be dipped into as and when needed. It is described by John Quarterman in the Preface as "a random access book" and this is in fact an excellent description. It is also described (by Tracey L LaQuey in the foreword) as "a window to the world".

Thus if you wish to know about networks in Saudi Arabia you look in the index under Saudi Arabia and are directed to two pages in the book, one explaining that the PDN (Public Data Network) of this country is called IDAS, and the other directing you to the entry for GulfNet, seven of whose nodes are in Saudi Arabia. PDN, IDAS and GulfNet are also referenced in the index.

There are sections on etiquette on Computer Mediated Communication (CMC) and ethics which mentions viruses and worms.

The second half of the book describes the Matrix itself. This contains a list of networks in various countries and how they connect with other networks. Also in some cases why and when they were set up and how they have developed since then. There are maps of countries showing sites and connections, how to access various systems (ie who to contact), and many references to find more information about various items.

There is a chapter listing standards bodies, what they do and where to contact them.

The one problem I do see with this book is that the information it provides in the second half of the book could need to be updated fairly regularly, for instance it refers to mcvax (which has been replaced by mcsun) and gives Peter Collinson as the UKC contact (Peter Houlder is now the contact).

This has obviously been recognised by John Quarterman because included in the back of the book is a page detailing how to join a project which has been set up to provide an interactive relational database of information related to the Matrix. The prototype of this service was already being developed when the book went to press.

In conclusion, I would recommend this book as a reference for both naive and experienced users who want to get to grips with networking – both large and small. It is useful as a stepping stone to other books which describe specific areas in more detail but also for generally finding out about who else is out there.

UKUUG Winter Conference Abstracts

Here are the abstracts of the papers that were delivered at the UKUUG winter conference held in Cardiff.

Thanks are due to Robert Evans <robert@computing-maths.cardiff.ac.uk> who ran the conference and provided these abstracts.

The Development of an Internet Protocol Routing Gateway

Richard Almeida

The Computing Laboratory
The University
Canterbury, Kent CT2 7NF
England
rpa@ukc.ac.uk

UNIX networking is based around the TCP/IP protocol stack. To allow file transfer, login and NFS services across different network media, and between different sites using TCP/IP, an Internet Protocol Routing Gateway is required. This paper is split into two sections. The first section is an introduction to Internet Protocols, and describes the lower levels of the Internet Protocol stack, including IP, ICMP and UDP. The second section describes the reasons for development and the technical operation of the University of Kent's Internet Protocol Routing Gateway, a device which routes IP traffic between Cambridge Ring, X25 and Ethernet media. Also discussed are the features that are be provided by the gateway to prevent unauthorised access to the connected networks.

C News — Some Experience

Ian G Batten

BT Fulcrum
igb@fulcrum.bt.co.uk

C News is a re-implementation of the basic news software from Geoff Collyer and Henry Spencer of the University of Toronto. It provides most of the functionality news, like the difference between B and A News. It simply provides the mechanisms for unpacking and distributing news to sites running either B News or some package compatible with that.

C was actually written in parallel with the development of B 2.11. This means that much of the behaviour is documented in terms of the older B 2.10 News package[1], rather than the more widely used 2.11.

Document Authoring Tools in a Networked Workstation Environment

Martin D. Beer
Steven M. George
Roy Rada

Department of Computer Science
University of Liverpool
PO Box 147
Liverpool
L69 3BX

The availability of networks of UNIX-based graphical workstations has stimulated new developments in authoring software. This paper presents our experiences along several fronts. First, we discuss the lessons learnt from developing a simple authoring tool to run on the Atari-ST, using the GEM operating system. This was always intended to be used by a single author and was not tied to expensive computer networks. With the arrival of a large network of powerful graphical workstations in our department, developments have recently transferred to them. We discuss the development of software using 1) the X11 toolkit and one of the readily available widget sets, 2) a configurable editor (gnu-emacs) to develop prototype applications, and 3) the ANDREW toolkit to reimplement the original Atari authoring system, but this time providing a tool that will allow several authors to collaborate closely with each other. The practicalities of these approaches are discussed with reference to our own experiences.

A Model for Representing a University Organisational Structure in the X.500 Directory Service

Steve Benford

The University of Nottingham

Directory services will play a vital role in supporting future network users and applications. Services to be provided include:

- establishing a global namespace for humans, applications, devices and groups;
- mapping names to addresses (''white pages'' service)
- providing a powerful information service for users (''yellow pages'' service).

In order to establish a global Directory service for OSI applications, the ISO and CCITT jointly published the X.500 international Directory standard in 1988. Following this, October 1989 will see the start of a large scale Directory pilot experiment throughout the UK academic community. This experiment funded by the JNT, will involve twelve UK universities and will utilise the ''Quipu'' X.500 implementation which runs on UNIX (Quipu was developed at UCL). The aim of the pilot is to provide much needed experience with Directories and to pave the way for the JNT's migration to OSI protocols. The experiment therefore encourages as many interesting, large scale uses of the Directory as possible.

This paper presents some preliminary modelling work carried out at the University of Nottingham, prior to taking part in the

1. As run by cam-el

experiment. This work shows how the detailed structure of the University might be represented in the Directory service. This includes support for a wide range of information, including:

- The structure of academic departments, research groups and service departments.
- University employees (names, addresses, telephone and email)
- University roles and occupants (e.g. department heads, union reps ...)
- Committees, meeting dates and locations.
- Halls of residence
- Departmental publications

The aim of the model is to provide a detailed breakdown of the University hierarchy. This might be used to provide a variety of novel services:

- Replacement of the current publication circulation lists by an on-line mechanism.
- Calendar facility for committee meetings, dates and management of membership.
- Role to address mapping (e.g. "who is the admissions tutor for the department of computer science?")

In order to develop this model, the paper introduces a number of new X.500 "object classes" and "attribute types". A description is then given of how these are arranged into a University naming tree. Arguments for the choice of this specific tree structure are presented.

Although specifically describing the University of Nottingham, it is intended that this model will be applicable to most universities and perhaps even, in a broader sense, to some companies. At a time when experience with Directories is urgently required, discussion of, and experimentation with, such models should provide useful insights into the possible applications of Directory services.

X/DeskMaster - Developing a Sophisticated UNIX Interface with X and OSF/Motif.

Paul Bentley
Gary Walsh

Siemens SDG
Woodley
Reading
Berkshire
RG5 3JP

X/DeskMaster is an object based, rules driven interface to UNIX, running together with X window managers. It offers many advantages and interesting features over other desktop managers.

X/DeskMaster is derived from the successful Collage windowing system, supplied for the last two years by Europe's largest UNIX vendor. The main requirements of the product were to retain the existing product functionality, but to use the OSF/Motif widget set.

This paper begins by briefly introducing the main features of X/DeskMaster. These are used as a framework to describe the experiences of porting this sophisticated application to X, using OSF/Motif widgets.

Techniques for adding a globally accessible menu to an application program, and for preventing desktop icons covering application windows are included. A `custom` widget was created so that icons may appear any shape the user desires and not in a rectangular box. Problems of interaction with window managers are covered.

This is not a critique of OSF/Motif. Much of the the content is independent of OSF/Motif, and is therefore relevant to any X application.

Naming Services rather than Machines

Piete Brooks

Computer Laboratory
Cambridge University
pb@uk.ac.cam.cl

Many sites and most users tend to think in terms of the host with which they are communicating, rather than the service that it is providing. People ask me for the X.121 address of my machine, rather than of an individual service it provides. The NRS has made a clear indication that an address is associated with a name, context, network triple. It also gave recommendations about the registration of suitable services at the site level, rather than the host level (notably mail).

The talk will extend these ideas and show how much simpler and more flexible they can make the provision of a more available system. This document provides some of the technical background to the talk, surveying the provisions made by various "name servers" in use in the UNIX environment.

The COSINE Project – An Introduction and the IXI Network

Ian Smith
Bob Cooper

Rutherford Appleton Laboratory

The EUREKA COSINE (Co-operation for Open Systems Interconnection Networking in Europe) Project has recently progressed from the specification phase to the implementation phase. The aim of the project is to create a pan-European data communications infrastructure based on OSI standards to serve the European Research Community. The user community, which spans both academic and industrial research, is expected to be large and diverse.

19 European countries, together with the Commission of the European Communities (CEC) are participating in the project. Funding will be derived from the CEC and national contributions. The UK national contribution will be funded by the Computer Board, DTI, ESRC, NERC and SERC.

The implementation phase will last for three years and is expected to create an ongoing set of services for the Research Community. RARE (Reseaux Associes pour la Recherche Europeene) will be responsible for managing the project.

The implementation phase comprises a wide range of activities and sub-projects, many directly concerned with the provision of services. An important sub-project, which will provide the foundation for all the other activities, is a private international X.25 network called IXI. This is scheduled to begin operation as a pilot service early in 1990 and will act as a backbone interconnecting many national academic and public networks.

The presentation will be divided into two parts. A brief introduction to the the COSINE Project will be followed by a presentation on the IXI network.

UNIX and Object Oriented Distributed Systems

Donal Daly
Vinny Cahill
Chris Horn

Distributed Systems Group
Department of Computer Science
Trinity College
Dublin 2
Ireland
daly@cs.tcd.ie
vjcahill@cs.tcd.ie
horn@cs.tcd.ie

UNIX is a well established system interface, as can be seen from the work of POSIX and X/Open. It has been gradually extended to support distribution and embrace concepts such as object orientation. Systems like Mach try to make the kernel smaller while providing increased support for distribution. Object oriented systems promise the potential for re-usable software, along with higher level data modelling. The Esprit COMANDOS project is supporting distribution and object orientation. It intends to provide an integrated platform for the development and online management of distributed applications. Placing a UNIX interface on top of such a distributed object orientated kernel is a possible approach to integrating UNIX and distributed object systems, which is explored in this paper. The motivation for supporting UNIX in an object oriented distributed environment is presented. We describe then, the main features of the COMANDOS kernel. Finally, an approach to supporting UNIX with an object oriented kernel is outlined. Such an approach would provide a migration path for existing UNIX users towards a fully object oriented system. It would also provide to UNIX users not interested in object orientation access to the increased functionality available in a distributed system.

Are Standards the Answer?

Dominic Dunlop

The Standard Answer Ltd

Moves are afoot to standardise every aspect of the UNIX® world in order that the benefits of open systems can be realised. But what *needs* to be standardised? What *are* the benefits? And who really cares anyway? The answers to these questions turn out to be rather vague, and are not always a good fit onto the standardisation activity which has taken place to date.

This paper examines the forces behind standardisation, reaching the conclusion that, while standardisation is a necessary process, it cannot and should not hope to have a significant effect on the diversity of ideas in the field of computer technology — or in any other field.

Designing an X.500 User Interface: The Early Stages

Andrew Findlay
Damanjit Mahl

Brunel University
Andrew.Findlay@brunel.ac.uk
Damanjit.Mahl@brunel.ac.uk

The X.500/ISO 9594 Directory is briefly described, and the early stages of the design of a user interface are detailed. Examples are included that give an idea of the appearance of the proposed interface under the X Window System.

An Idiots Guide to OSI Inter-Computer Cooperation

John Henshall BSc FSA Scot.

Senior Computing Officer
EUCS, University of Edinburgh

This tutorial paper is intended as an introduction to the application orientated layers of the ISO reference model. It will introduce each of the end to end related layers discussing what each uniquely provides for an application. The internal architecture of the model will be examined to give the feel of how the communication functions are realised. This is strictly a "beginners" guide and so any delegate with a basic knowledge of OSI could take the opportunity offered by this tutorial to catch up on some sleep/walk the dog/phone the spouse.

The X.500 Directory Service - Data Gathering

Julia M Hill

Computer Centre
Heriot-Watt University
Edinburgh

The technical requirement is for a system to interface with the X.400 Message Handling Systems, but there are a number of uses: the Directory Service will provide a means for humans to communicate with other humans.

The most obvious use is for finding electronic mail addresses of colleagues at home and abroad. Initial internal use may be very simple, but still valuable: after all, what use is a telephone without a directory? Users of electronic mail systems presently have to write to colleagues to find out their electronic mail name and address. The Directory will provide a central means of storing data about people whom others wish to contact for various reasons, and also data about administrative units. People spend a lot of time searching manually for names, addresses and mailnames. They ask Postmasters, Computer Centre Advisers, and Administrators. With current administrative policies tending towards the 'terminal on every desk' approach, many users who are not computer-literate will need this service. It may even provide the answer to the well-known question "Where can I get a list of all the Organic Chemists in Europe?".

Eventually it will become the master copy for much of the data it contains.

UKnet Overview, Accounting and Future plans

Peter Houlder

Computer Laboratory
University of Kent
Canterbury, UK
uknet@ukc.ac.uk

This paper is in three parts. The first part is essentially a repeat of old information, which can be skipped by regular attendees. It deals with the internal aspects of UKnet as a network; what it does, how it is administered, services offered costs etc. The second part tries to shed some light on the complexities of costing network services. The final part covers the UKC viewpoint on TCP/IP and OSI developments.

ISODE — Who, What and Where Next

Julian P Onions

Communications Research Group
Nottingham University

ISODE is the ISO development environment. It is a collection of libraries and application programs which together provide the upper layers of OSI. ISODE currently runs on most versions of UNIX. ISODE is openly available, which means it is usable by anyone for anything except that none of the authors takes responsibility for use or misuse of the product. It is available on a tape for a copying charge or can be retrieved directly by a suitable transfer protocol. The principal author of the ISODE is Marshall T Rose now with Nysernet.

However, since the first release, there have been many contributors to the code and it is hoped that this will continue in the future.

An Overview of Application-Level Services in the Internet Protocol Suite

Jim Reid

Department of Computer Science
Strathclyde University
Glasgow
Scotland
G1 1XH
jim@cs.strath.ac.uk
jim@strath-cs.uucp
...!uunet!mcvax!ukc!strath-cs!jim

The Internet Protocol family, often referred to as the TCP/IP protocols, have been widely adopted for local area and wide area networking. This has been a result of the evolution of the US ARPAnet and the development of protocol implementations for most of the commonly used operating systems. The increasing popularity of UNIX workstations has also meant an increased use of the Internet protocols, often in environments that have previously used proprietary protocols.

This paper presents a short introduction to the Internet Protocol architecture. A brief description of the main services that are built on top of the available transport service protocols is given. These include the obvious facilities like mail and file transfer (FTP and SMTP respectively) and a virtual terminal service (telnet). UNIX-specific services including system logging, remote command execution and network routing will also be discussed.

The Internet protocols are also used to specify experimental facilities that can be developed into fully-fledged production systems. Two common examples are Sun's Network File System (NFS) and Remote Procedure Call (RPC) mechanism. More recently, the Internet has switched to a domain based name service for address and name lookups.

In addition to providing these general network services, the Internet protocols are also used as the basis of both the X-windows and NeWS windowing systems. Other "interesting" facilities that can be provided include a network time-of-day service and a face server.

AUSTRIA - UUGA
Friedrich Kofler
Schottenring 33/Hof
A-1010 Wien
AUSTRIA
Tel: +43 222 34 61 84
uuga@tuvie.at

BELGIUM - BUUG
Marc Nyssen
Department of Medical Informatics
VUB, Laarbeeklaan 103
B-1090 Brussels
BELGIUM
Tel: +32 2 477 4111
Fax: +32 2 477 4000
buug@vub.uucp

DENMARK - DKUUG
Mogens Buhelt
Kabbelejevej 27B
DK-2700 Bronshoj
DENMARK
Tel: +45 31 60 65 80
mogens@dkuug.dk

FUUG

FINLAND - FUUG
Anu Patrikka-Cantell
Finnish UNIX Users' Group
Arkadiankatu 14 B 45
00100 Helsinki
FINLAND
Tel: +358 0 494 371

AFUU

FRANCE - AFUU
Miss Ann Garnery
AFUU
11 rue Carnot
94270 Le Kremlin Bicetre
Paris
FRANCE
Tel: +33 1 46 70 95 90
Fax: +33 1 46 58 94 20
anne@afuu.fr

GUUG

GERMANY - GUUG
Dr Anton Gerold
GUUG-Vorstand
Elsenheimerstr 43
D-8000 MUNCHEN 21
WEST GERMANY
Tel: +49 89 570 7697
Fax: +49 89 570 7607
gerold@lan.informatik.tu-muenchem.dbp.de

HUNGARY - HUUG
Dr Elôd Knuth
Computer and Automation Institute
Hungarian Academy of Sciences
H-1502 Budapest 112, P.O. Box 63
HUNGARY
Tel: +361 1 665 435
Fax: +361 1 354 317

ICELAND - ICEUUG
Marius Olafsson
University Computer Center
Dunhaga 5
Reykjavik
ICELAND
Tel: +354 1 694747
marius@rhi.hi.is

IRELAND - IUUG
Norman Hull
Irish UNIX-systems User Group
Court Place
Carlow
IRELAND
Tel: +353 503 31745
Fax: +353 503 43695
iuug-exec@cs.tcd.ie

i2u

ITALY - i2u
Ing Carlo Mortarino
i2u
Viale Monza 347
20126 Milano
ITALY
Tel: +39 2 2520 2530

NLUUG

NETHERLANDS - NLUUG
Patricia Otter
p/a Xirion bv
Burg. Verderlaan 15 X
3454 PE De Meern
THE NETHERLANDS
Tel: +31 3406 61 990
nluug@hp4nl

NORWAY - NUUG
Jan Brandt Jensen
Unisoft A.S.
Enebakkvn 154
N-0680 Oslo 6
NORWAY
Tel: +47 2 688970

EUUG-S
N
I
X

SWEDEN - EUUG-S
Hans E. Johansson
NCR Svenska AB
Box 1206
S-164 28 KISTA
SWEDEN
Tel: +46 8 750 26 03
hans@ncr.se

UNITED KINGDOM - UKUUG
Bill Barrett
Owles Hall
Buntingford
Hertfordshire SG9 9PL
UNITED KINGDOM
Tel: +44 763 73039
Fax: +44 763 73255
ukuug@ukc.ac.uk

PORTUGAL - PUUG
Legatheaux Martens
Avenue 24 de Julho, nº 134, 7º
Lisboa
PORTUGAL
Tel: +351 1 673194/609822
Fax: +351 1 7597716
puug@inesc.pt

The European UNIX systems User Group
Owles Hall
Buntingford
Hertfordshire SG9 9PL
UNITED KINGDOM
Tel: +44 763 73039
Fax: +44 763 73255
euug@EU.net

YUGOSLAVIA - YUUG
Milan Palian
Iskra Delta Computers
Parmova 41
61000 Ljubljana
YUGOSLAVIA
Tel: +38 61 574 554
mpalian@idcyuug.uucp