Australian UNIX systems User Group Newsletter

AUUGN

Volume 11, Number 1

February 1990

AUUG Book Club

The Australian UNIX* systems User Group Newsletter

Volume 11 Number 1

February 1990

CONTENTS

---

* UNIX is a registered trademark of AT&T in the USA and other countries.

AUUG General Information

Memberships and Subscriptions

Membership, Change of Address, and Subscription forms can be found at the end of this issue.

All correspondence concerning membership of the AUUG should be addressed to:-

> The AUUG Membership Secretary,
> P.O. Box 366,
> Kensington, N.S.W. 2033.
> AUSTRALIA

General Correspondence

All other correspondence for the AUUG should be addressed to:-

> The AUUG Secretary,
> P.O. Box 366,
> Kensington, N.S.W. 2033.
> AUSTRALIA

AUUG Executive

President	**Greg Rose**	Secretary	**Tim Roper**
	greg@softway.sw.oz Softway Pty. Ltd. New South Wales		*timr@labtam.oz* Labtam Information Systems Pty. Ltd. Victoria
Treasurer	**Michael Tuke** *mjt@anl.oz* ANL Limited Victoria		
Committee Members	**Peter Barnes** *pdb@uqcspe.cs.uq.oz* Computer Science University of Queensland		**John Carey** *john@labtam.oz* Labtam Information Systems Pty. Ltd. Victoria
	Pat Duffy *pat@pta.oz* Pyramid Technology Corporation New South Wales		**Chris Maltby** *chris@softway.sw.oz* Softway Pty. Ltd. New South Wales

Next AUUG Meeting

The AUUG90 Conference and Exhibition will be held
from September 25th to September 28th 1990.
The venue will be the World Congress Centre, Melbourne.

AUUG Newsletter

Editorial

Late again. I am trying not to make a habit of this. Oh Well.

The Summer Technical meetings have come and gone, and by all accounts were a great success. Technical meetings were held in Perth, Hobart, Melbourne, Canberra and Sydney. The next issue of *AUUGN* will feature reports on these meetings and some of the papers that were presented.

This issue sees the start of the AUUG Book Club, an arrangement with Prentice Hall that will get AUUG members a 20% discount on computing titles. I hope you will all take advantage of it.

Also in this issue is information on the upcoming AUUG elections. I have enjoyed my year as Assistant Returning Officer, and will be running again. But, unfortunately, the same does not apply for a few members of the Management Committee. So we seek nominees for the election to be held in May. Don't forget to send in your completed nomination forms by the 1st of May. Do your bit to break the Labtam/Softway power block!

AUUGN Correspondence

All correspondence reguarding the AUUGN should be addressed to:-

> David Purdue
> AUUGN Editor
> Labtam Information Systems Pty. Ltd.
> 43 Malcolm Road
> Braeside Victoria 3195
> AUSTRALIA
>
> ACSnet: *davidp@labtam.oz*
>
> Phone: +61 3 587 1444
> Fax: +61 3 580 5581

Contributions

The Newsletter is published approximately every two months. The deadline for contributions for the next issue is Friday the 23rd of March 1990.

Contributions should be sent to the Editor at the above address.

I prefer documents to be e-mailed to me, and formatted with troff. I can process mm, me, ms and even man macros, and have tbl, eqn and pic preprocessors, but please note on your submission which macros and preprocessors you are using. If you can't use troff, then just plain text please.

Hardcopy submissions should be on A4 with 30 mm left at the top and bottom so that the AUUGN footers can be pasted on to the page. Small page numbers printed in the footer area would help.

Come on, everyone, contribute! If the muse is upon you, and you have to write, but you can't think of anything to write about, give me a call and I'll throw some ideas at you.

Advertising

Advertisements for the AUUG are welcome. They must be submitted on an A4 page. No partial page advertisements will be accepted. The current rate is AUD$ 200 dollars per inside page. More prominent positions are available but cost more. Contact the editor for details.

Mailing Lists

For the purchase of the AUUGN mailing list, please contact Tim Roper.

Back Issues

Various back issues of the AUUGN are available, details are printed at the end of this issue.

Acknowledgement

This Newsletter was produced with the kind assistance and equipment provided by Labtam Information Systems Pty Ltd.

Disclaimer

Opinions expressed by authors and reviewers are not necessarily those of AUUG Incorporated, its Newsletter or its editorial committee.

Secretary's Letter

Firstly I would like to record my pleasure at the 1990 summer meetings actually happening for the first time after a couple of false starts in previous years. Furthermore, they were not only highly successful but in all cases were made so by the efforts of *neither* present nor past committee members. My thanks go to all those responsible.

Increased involvement by members in the organisation of new events and activities is the key to AUUG increasing its range of services. Right now we could do with someone to distribute the latest USENIX conference proceedings and the 1990 UniForum UNIX products directory. Taking on relatively easy and self-contained tasks such as these is a good way of deciding whether or not you feel cut out for more involvement with AUUG, which leads me to ...

Annual elections of office bearers are due in May. A nomination form is included in this issue for your convenience. Please give serious thought to whether you would like to offer your particular skills to AUUG. Do not assume that the *status quo* will be maintained because it very likely will not. Also, you may know a member who just needs your encouragement to stand.

At the last meeting of the Management Committee it was suggested that AUUG explicitly expand its aims to include *open systems*† other than UNIX, such as Mach, NFS and the X Window System. Of course such things are often the subject of conference presentations, perhaps due to their generally being of interest to the same community as is UNIX. The question posed here is whether AUUG should move formally to claim and defend that territory. Please give it some thought.

Tim Roper <timr@labtam.oz>

---

† For want of a better term ...

AUUG Inc
1990 Annual Elections – Nomination Form

We, ...,

.... and

....,

being current financial members of AUUG Inc do hereby nominate

....

for the following position(s)**

> President
> Secretary
> Treasurer
> General Committee Member (4 to be elected)
> Returning Officer
> Assistant Returning Officer

Signed .... Date ....

....

....

I, ...
do hereby consent to my nomination to the above position(s).

Signed .... Date ....

** Strike out position(s) for which nomination is **not** desired. Each person may be elected to at most one position, and the ballot for positions shall be determined in the order in which the positions are listed on this nomination form.

Nominees may use the reverse side of the page to provide a short statement of policies, aims and/or objectives to be circulated to members of AUUG Inc at the time of the election.

Nominations must be lodged with the Secretary by the first day of May, 1990. Please post them to:

> The Secretary
> AUUG Inc.
> PO Box 366
> Kensington
> NSW 2033

WAUG

Western Australian Unix systems Group

The Western Australian UNIX systems Group (WAUG) was formed in late 1984, but floundered until after the 1986 AUUG meeting in Perth. Spurred on by the AUUG publicity and greater commercial interest and acceptability of UNIX systems, the group reformed and has grown to over 70 members, including 16 corporate members.

A major activity of the group are monthly meetings. Invited speakers address the group on topics including new hardware, software packages and technical dissertations. After the meeting, we gather for refreshments, and an opportunity to informally discuss any points of interest. Formal business is kept to a minimum.

Meetings are held on the third Wednesday of each month, at 6pm. The (nominal) venue is "University House" at the University of Western Australia, although this often varies to take advantage of corporate sponsorship and facilities provided by the speakers.

The group also produces a periodic Newsletter, **YAUN** (Yet Another UNIX Newsletter), containing members contributions and extracts from various UNIX Newsletters and extensive network news services. YAUN provides members with some of the latest news and information available.

For further information contact the Secretary, Skipton Ryper on (09) 222 1438, or Glenn Huxtable (glenn@wacsvax.uwa.oz) on (09) 380 2878.

Glenn Huxtable,
Membership Secretary, **WAUG**

SESSPOOLE

Who is SESSPOOLE?

SESSPOOLE is the South Eastern Suburbs Society for Programmers Or Other Local Enthusiasts. Here we are talking about the South Eastern Suburbs of Melbourne.

What is SESSPOOLE?

SESSPOOLE is a group of programmers and friends who meet every six weeks or so for the purpose of discussing UNIX, drinking wines and ales, and just relaxing and socialising over dinner. Anyone who subscribes to the aims of SESSPOOLE is welcome to attend our meetings, whether they come from the South Eastern Suburbs or not. The aims of SESSPOOLE are:

> To promote knowledge and understanding of the UNIX system, and of similar or related computer systems; and to promote knowledge and understanding of red and white wines, and of similar or related wines.

SESSPOOLE is also the first Chapter of the AUUG to be formed, and its members were involved in the staging of the AUUG Summer'90 Melbourne Meeting.

When is SESSPOOLE?

The next meeting of SESSPOOLE is on Wednesday, the 28th of March 1990, at 6:30pm.

Where is SESSPOOLE?

The next meeting of SESSPOOLE will be held in the Bistro of the Oakleigh Hotel, 1555 Dandenong Road, Oakleigh.

Want to know more?

To find out more about SESSPOOLE and SESSPOOLE activities, contact either David Purdue <*davidp@labtam.oz*> or John Carey <*john@labtam.oz*>. Their phone number is 587-1444 (bh). Or look for announcements in the newsgroup **aus.auug**.

AUUG Book Club
Book Reviews

AUUG Inc and Prentice Hall Australia have formed the AUUG Book Club to give AUUG members a chance to obtain Prentice Hall books at a significant discount.

To obtain copies of the books reviewed here, fill in the order form that appears on page 18. Don't forget to deduct 20% from the listed retail prices.

Review copies of these books were kindly provided by Prentice Hall.

If you would like to review books for further offers from the AUUG Book Club, please contact the AUUGN Editor, David Purdue. (Find his address on page 5.)

CONCEPTUAL SCHEMA AND RELATIONAL DATABASE DESIGN: A FACT ORIENTED APPROACH
by G.M. Nijssen and T.A.Halpin.
RRP $47.95, 340 pages.

Reviewed by
Amanda Moore
Australian Centre for
Unisys Software (ACUS)
<amanda@mlacus.oz.au>

This book by Nijssen and Halpin from the University of Queensland is an excellent buy for people who are involved in designing information systems that will be implemented using relational databases. The methodology described therein fits in very neatly with a software lifecycle that includes prototyping of applications in the requirements phase. So for systems where the customers are known end-users who are the experts on the application that is being modelled, it is particularly valuable.

The authors explore high level semantic data modelling and conceptual schema design from a fact oriented approach which they believe is a more powerful and yet simpler method than either entity-relationship modeling or normalisation. The conceptual schema as designed using this fact-oriented approach can then be mapped into SQL based relational databases for implementation using current technology. One idea behind the methodology is based in the fifth generation concept of using formal English as the means of communicating with the computer. The authors provide the designer with a design procedure that allows the starting point to be examples of relevant facts expressed in natural language which is then smoothly mapped from that human level down into relational database structures.

The methodology used for designing a conceptual schema and then for mapping it into relational tables is known as NIAM which stands for "Nijssen's Information Analysis Methodology". In this scheme a conceptual schema is seen to consist of stored fact types, constraints and derivation rules. The part of the schema which contains stored fact types lists the type of facts that may be stored in the database including how they are referenced and the roles played by them. The constraints sections contain static or dynamic constraints that apply to the fact types and the derivation rules section provides a list of functions, operators and rules that can be used to derive information that is not explicitly stored in the database.

The book contains a description of the NIAM design procedure which consists of three parts: CSDP (Conceptual Schema Design Procedure), ONF (Optimal Normal Form) Algorithm and IFD (Information Flow Diagrams).

CSDP is a nine step procedure for building a design by "starting with specific examples, and thereafter following a well-defined procedure using visually meaningful diagrams easily populated for validation purposes." It is necessary that these examples are extensive enough to represent a significant population and it is stressed in the book that consulting with an application expert (called the universe of discourse (UoD) expert) is useful for determining if the model does in fact represent the real world situation to the degree of accuracy required by the customer.

The nine steps are specified in recipe book fashion in chapters three through nine.

A good way to come to grips with CSDP is to sit down with pencil and paper when read-

ing chapter three and mark the new terms encountered on the diagrams and do the exercises. The time spent there pays off in later chapters. It is the crux of the entire book as it describes the fact types and basic diagramming conventions and notations used to describe the conceptual schema. The succeeding chapters specify constraints on the basic fact types as well as information on subtypes, references schemes and redundancy checking. Following these steps results in a conceptual schema.

Once the conceptual schema is complete it can be mapped onto a relational schema by applying the ONF algorithm. For anyone who has ever tangled with 1NF through 5NF normalisation, this algorithm is delightfully simple. Most of the hard work of getting a design that accurately represents the UoD, in such a way that it is easily implemented, has already been done in the CSDP method of building the conceptual schema. The algorithm itself consists of mapping the fact types isolated during the CSDP into tables and the role types into columns. The keys are derived from the uniqueness constraints. Chapter eleven, which describes this algorithm and its application, also specifies how to transform a conceptual schema, before applying the ONF algorithm, to obtain the most efficient relational schema.

The third part of the methodology is the IFD's which allow expression of the process-oriented aspect. This has graphical symbols for processes, functions, flow of information, screens, reports, users and databases. Since the methodology is very much data oriented, it is not surprising that the area of information flow is not tackled until chapter twelve. However, using the IFD's would result in a very clear picture of the information required in a system and also where it flows because the coupling between the Information Flow Diagrams (IFD) and the Conceptual Schema Diagrams (CSD) is strong. The reason being that the information manipulated by a function is defined on the diagram itself.

As a technical book, this book has some very good points. The subject matter in the book is taught as a course at the University of Queensland so it is essentially a student-oriented book. This is good news for the professional database designer who can easily pick up the book and learn the methodology without help, since the information is very clearly specified. Even someone who was very well versed in

designing relational databases using normalisation would benefit from the different perspective that NIAM covers. As well, it has ANSWERS to some of the exercises especially for the early chapters when it is worthwhile to do the exercises to check understanding. It has always escaped this readers understanding as to why people write technical books with lots of exercises and no answers!

One feature that the book is lacking is a glossary. Many technical terms are used throughout the book and until some familiarity is gained through their repetition, it is necessary to backtrack to the original mention of the term. A glossary would be an extremely valuable addition for the second edition.

As the preface points out, it is not necessary to read the entire book to get an understanding of the NIAM methodology. The first chapter, which gives a historical background, is interesting but not strictly necessary to the professional reader. Nor is the information in chapter six on set theory or the discussion on normalisation required reading if the reader is already familiar with these concepts. In a preliminary reading the conceptual metaschema information in chapter thirteen and the background discussion relating NIAM to other design methodologies could also be left out.

All in all, this is a book to be studied not skimmed, if real benefit is to be gained. However, once the effort has been made, this reader believes that the advantages gained from knowing this simple but effective method for designing conceptual schemas that map easily into well formed relational tables is worth the investment in time and effort.

ELECTRONIC LOGIC SYSTEMS
by A.E.A Almaini

Reviewed by
Charles Homes
Austek Microsystems Pty Ltd
<chomes@adl.austek.oz.au>

Electronic Logics Systems deals with design of electronic logic. It attempts to provide a text for the undergraduate as well as a reference for the engineer. The author admits to the

difficulty in tracking what is a fast changing field, and with this, the second edition, he tries to keep pace. The book is split into two sections. Part one deals with the core concepts of logic design. It introduces the mathematical methods, and gives examples of the physical devices and processes they apply to. Part two deals with more complicated and abstract logic techniques. It comes closer to the real word problems facing a logic designer. Each chapter contains numerous examples and end with a questions and answers section.

PART ONE:

The first chapter discusses combinatorial logic. Starting with a simple, and short (thankfully), introduction to boolean logic we move on to Karnaugh maps, gate conversion and factoring. The basic concepts are all introduced clearly and with little fuss.

Chapter two describe logic and memory devices. It talks about bipolar and MOS circuits and provides simple descriptions of the circuits and operation of RTL, DTL, TTL, ECL, I², nMOS, cMOS and GaAs. The topics covered range from noise immunity, fan in/out, power dissipation, gate delay and interfacing various logic families. These discussions are reasonably detailed and include sample calculations. Most memory devices are also discussed and PAL's, PLA's and gate arrays are introduced. As before circuit diagrams showing possible implementations of the various devices are also included. At the end of the chapter ULM's (Universal Logic Modules) are introduced. The ULM is heavily used throughout the rest of the text. Chapter is 2 already a little dated.

Next comes a short chapter comparing various design strategies. Example are given for combinatorial logic designs using discrete components, standard gates, ULM's, ROM's, PLA's and PAL's. The real gem here is a quick discussion of static logic hazards.

Chapter 4 delves into sequential logic. After a mercifully short (but thorough) introduction to clocked logic an overly long discussion of counters follows. Then general synchronous circuits and state reduction. The chapter ends with examples of state machine implementations.

Chapter 5 is a very good description of asynchronous sequential circuits. It is presented step by step and is easy to follow. Race conditions and hazards are explained.

In chapter 6 provides an adequate coverage of arithmetic logic circuits. It clearly sets out all the required information and gives some implementation details. Serial logic, binary division and floating point are all discussed.

PART TWO:

Chapters 7, 8, 9 and 10 discus partitioning, group coding, minimisation for combinatorial, and sequential circuits (both synchronous and asynchronous). These chapters delve into the details for each logic type using both manual and computer specific algorithms. Many design examples are included. Guidelines for state assignment in sequential logic can be found, and a discussion on sequential circuits using shift registers for state proved interesting.

In chapter 11 the realities of difficult sequential circuits are explored. Examples include systems with a large number of states and systems with many variables.

Modulo-2 logic circuits are discussed in chapter 12. This is a good, but rather brief, introduction to design with exclusive-or/and logic. Coding, decoding and error detection and correction are also mentioned.

Chapter 13 is devoted to computer aided design of electronic circuits. This is a difficult subject and by necessity the author chose one particular software package. This discussion introduces some of the principles of CAD, but a much more thorough approach is needed to be really beneficial.

The author himself provides an adequate summary of his work in the preface, as follows:

> "The book has been designed for use by undergraduates studying electronic or computer engineering, Students in other engineering fields [...] will also find it valuable, and its practical orientation will make it of use to practicing engineers."

I recommend this book if you find yourself in either of the above categories. The text provides plenty of grounding in basic logic and the abundant practical examples will make it useful for for those whose native language does not include such gems as "PAL" and "MOSFET". I must also add to the authors comments that section two may also provide some interesting reading to those well versed in logic design. But, the book I received did have one mis-cut page and half a dozen pages stuck together.

THE HELIOS OPERATING SYSTEM
by Perihelion Software.
RRP $59.50

Reviewed by
Huw Davies
La Trobe University Computer Centre
<cchd@latvax8.lat.oz.au>

With such a title and sight unseen, I thought that this book might be in the same vein as Bach: **The Design of the Unix Operation System**, that is, a description of the design and implementation of a distributed Unix like operating system. Once it arrived, it was obvious that this was not the case. According to the cover, "Divided into three main sections covering the user interface, the programmer interface, and a full technical reference guide..."

The User Interface section is best described as a simple introduction to shell usage, along with a description of each shell command, one to a page. To anyone with any exposure to Unix, this section can be skipped almost entirely, a saving of 145 pages. For those without Unix experience, this section provides a good introduction. Chapter 4 describes in considerable detail, the Input/Output Server, which is used to provide a communication path from a transputer network to a host processor. Unfortunately, this chapter illustrates one of the main weaknesses of the text, which is a lack of consistent depth of presentation. The description of the configuration file is very detailed, and is obviously documentation for someone configuring a transputer network, describing, for example, the System_Image which is "the system image to be booted into the root transputer." Chapter 5 further illustrates this problem, describing how to configure the transputer network providing unnecessary detail without providing understanding.

The second part of the book, entitled "The Programmer Interface", starts with an very general overview of transputers and the Helios operating system. A more detailed discussion of communications, processes and other parts of Helios are given. At this stage, I felt that the book was getting into the meat of the topic, but again at a rather superficial level. The next chapter is entitled "Programming under Helios" and starts with a section on Philosophy of programming and describes the use of Paral-

lelism. Following directly from this, is a detailed description of the assembler, including tokenised assembler format. It is very hard not to get frustrated by the rather erratic editing. Perseverance is rewarded (for a time) by a very good description of parallel programming using the Component Distribution Language and the extensions made to the shell to connect together individual processes to form a task force used to solve a particular problem. Unfortunately, Chapters 8 and 9 consist of 213 pages of description of the individual routines provided by the Helios resident library. Whilst this information is vitally useful to a programmer, I again question its presence in this book.

The final section "Technical Information" contains a more detailed description of the kernel, the Nucleus, servers, protection, Network Management and other related topics. Again, these areas could be of great interest to the non-Helios user, but this part is again of varying depth and detail and proved to be disappointing. This section concludes with yet another list, in this case message summaries, one to a page, occupying 22 pages.

It is hard to see where the publishers have aimed this book. My guess is that they see it sitting next to a Helios users' terminal, acting as a bible. If this is the case, there are descriptive sections which should be removed. If, on the other hand, it is aimed as a general introduction to Helios, much of the detail in the book can be compressed (or removed altogether) and certain sections rewritten to make them more complete and easier to understand. Notwithstanding these comments, given the dearth of published information on Helios (judging by the bibliography) if you want to know more, then this book may be the only place to go.

HANDS-ON VENTURA 2.0
by C.J. Wallia and LaVaughn Hart.
RRP $43.95

Reviewed by
Allan Charlton
Labtam Technical Publishing Department

I've been using Ventura Publisher on a daily basis for more than two years. It is a very powerful, mistake-tolerant tool of my trade, and

I like it. Unfortunately, many of the books about Ventura seem to be written as a first exercise by an enthusiastic new owner eager to air his knowledge of typesetting and page layout, but this one is an exception. It's a friendly, inviting book that didn't treat me as a moron. The exercises in 'Hands-on Ventura 2.0' produce presentable documents using most or all of Ventura's features, and will prompt you to healthy experimentation.

The book is a convenient size, and is comb bound. I found that when I wanted to lay it flat on my desk, it would. And when I wanted to settle down for a good read, it was good for that, too. The writing style is conversational, and the layout is clear. The book is structured as a series of tutorials followed by a function-by-function reference guide and an index. If you've worked the exercises, and made notes along the way, you should have no trouble. The table of contents at the front of the book is detailed enough to locate specific parts of exercises, so you can locate help that way if it's more comfortable.

One thing the book doesn't teach you is how to lay out a page to make it attractive and readable. That is a separate set of skills you will have to pick up somewhere else.

What do we want from a book like this? What I look for is some exercises to step me through my 'slow' areas, and a quick reference guide for times when all I need is a prompt. This should all be tied together in such a way that it is easy to quickly find a feature when I need it. Books like this get used a lot, and wind up all dog-eared, which is just the way they should be (the neatest books on my shelves are the ones that are hardest to read). 'Hands-on Ventura 2.0' will get dog-eared because it's useful and easy to use.

In the preface, Dr.Wallia says that each exercise can be comfortably completed in one hour. A little further on he says, "After completing all twenty exercises, you'll know how to work Ventura well." This, I'm afraid, is pure salesmanship. You will have been led through exercises that use most or all of the features of Ventura, but that doesn't mean that you know how to use them well. In fact, I don't believe that twenty hours of exercises will do very much more than give you enough confidence to launch yourself into your first 'real' project. There are so many features to Ventura that you probably

won't remember what they all are, let alone how to use them. The Ventura learning curve is long, no matter how you approach it, but most of us don't need all the features of the package all of the time. So for most of us, Dr. Wallia's book will be a companion, a crutch to lean on in times of memory lapse or creativity surges. It is a worthwhile, readable addition to the manuals supplied with the software.

The concept behind Ventura is that of a utility, rather than a writing program. You write in a word processor, illustrate in a graphics package, and draw graphs in a spreadsheet, then load Ventura and import those files. A chapter file keeps track of it all, while Ventura generates other files for captions, style sheets and graphics. Ventura does have some capabilities in word processing and simple drawing, but it is the DTP functions where it shines and that is what it is for. 'Hands-on Ventura 2.0' doesn't spell out this concept in so few words, but by the time you've done the first half dozen or so exercises, there will be little doubt in your mind. The text and exercises bring the reader along gently but firmly, introducing material in bite-sized chunks that are not difficult to digest. Then, at the end of the book, a table of keyboard shortcuts is provided. Once you have picked up some proficiency with Ventura, you may find the mouse to be an infuriatingly slow device for entering commands, and the keyboard shortcuts will save lots of aggravation. I'm grateful to Dr. Wallia for showing me a couple that I didn't know about.

I tried to find something inaccurate, some misinformation, or an error, perhaps. There aren't many.

In the preface, Dr. Wallia says that the minimum background is an understanding of the fundamentals of DOS (meaning MS-DOS), and with that I disagree. It's good to know a little about the operating system, but you can muddle along quite nicely without it. Ventura has a friendly approach to the handling of files, and if you can handle filename extensions, you know enough. As you work with Ventura you will find yourself generating large numbers of files, and you may need some expertise or a utility such as XTree to manage them.

Dr. Wallia assumes you have PostScript. The book doesn't say so until page 28, by which time you're into exercise 2, and that's a little unfortunate. Lots of readers will have purchased systems without PostScript, and they can make

good use of Ventura; it just looks a little different in places and some options may not be present. I think this should have been explained in the preface and during the section on installation of the software rather than casually tossing it in part way through exercise 2. I doubt that readers who don't have PostScript printers will have much difficulty provided that they remember that this will be the reason some screens don't look the way the book says they will. Sometimes a part of an exercise will be impossible, and the reader should have this explained and be encouraged to continue. For example, if you have an Epson dot matrix printer (so no PostScript), exercise 2 will explain that it assumes you have PostScript, and ask you to select Zapf Chancery font. It doesn't even tell you that you only have Swiss and Dutch (Helvetica and Times) fonts and that you should experiment with them. It's a small thing, but I think that it could annoy some readers.

On page 89, exercise 5 has a note to the effect that Ventura changes word processor files by inserting the paragraph tag names, and that if you don't want this to happen, you should use the Rename option in the Edit menu before saving. Dr. Wallia's explanation of just what the Rename function will do is on page 382, and perhaps it should be cross-referenced to exercise 5's special paragraph. When you use the Rename function, you will wind up with two copies of the word processor file - one with tag name, and one without. If you edit the document, be sure to make the same editing changes to both files!<R> These three are the worst faults I could find, and are not indicative of the rest of the book. Faults are few and minor - things that might be a little annoying, but you'll cope without difficulty.

Probably the worst mistake made by technical writers is that they don't manage to correctly identify and address their target readers. As a result, we get treated as boffins or buffoons, neither of which is true for most of us most of the time. Dr. Wallia is a little less guilty of this than most writers, so I warmed quickly, but I did get tired of the opening paragraphs of each chapter. They are a wise inclusion for the first few, but at the end of twenty exercises I didn't need detailed instructions on how to load Ventura, or how to quit. Of course, there is the option of skimming past those paragraphs, but there is the danger that something new is thrown into the one you're skimming. Let me assure

you they're all the same.

There is a danger in books of this type that you won't want to trouble yourself with the basic exercises, but get right down to the exercises you're interested in doing. For example, if you have to design a brochure, that's exercise 13, and at an average of an hour each exercise, it's almost two working days before you complete the brochure exercise. My advice is to take that time. As I said earlier, Ventura has a long learning curve any way you do it, and you will pick up things in those first 12 exercises that you will want to use in your brochure. Take the time, work all the exercises, then start your own 'real' project. As you go, you will get stuck, so rather than muddle on, go back and repeat some of the exercises. Wallia and Hart have thought them out well and presented them with illustrations of screens in the right places. You can read the text quickly and from a distance (that's important) so there's no excuse.

If you are somewhat entranced by the thought of becoming a desktop publisher, buy the book and have a read. If you are already wrestling with Ventura, buy the book. It will cost you some hours, but will save you heaps. If you already have the package, buy the book. It will tell you things you didn't know about Ventura, and you'll work better for it. It's so good, I wish I'd written it.

CONCURRENT PROGRAM STRUCTURES
by David Bustard, John Elder and Jim Welsh
RRP $132.95, 332 pages.

Reviewed by
David Purdue
Labtam Information Systems Pty Ltd
<davidp@labtam.labtam.oz>

A concurrent program is one that contains components that may be executed simultaneously. Knowledge of the principles of concurrent programming is becoming increasingly important, because of the applications concurrent programming has in operating system development (particularly distributed and multiprocessor operating systems), real time or embedded systems, and process-oriented simulation modeling. **Concurrent Program Structures** is very nearly an excellent textbook for a university course on

concurrency.

The book starts with an overview of concurrency. It uses analogies to introduce the notion of concurrent processes, and gives examples of inherently concurrent applications and of applications that can exploit concurrency to improve performance.

Chapter 2 examines models for the execution of concurrent programs, including single processor and multiprocessor architectures.

Chapter 3 gives an approach to the design of concurrent programs. It outlines a good methodology for designing programs and for documenting the program design, and illustrates the method with several examples. The method stresses modular design, and this chapter gives some tips on recognising areas of responsibility in a program that can be split off into a separately executing module.

Chapter 4 starts the discussion of implementing concurrent programs. It introduces the Pascal Plus language, which is ISO Pascal with extensions to support concurrency based on proposals made by C.A.R. Hoare. Pascal Plus is the language used for example programs in most of the rest of the book. A definition of Pascal Plus is given in the first appendix. This chapter also introduces the monitor concept for mutual exclusion.

After introducing modular design, chapter 5 discusses modular testing.

Chapters 6 and 7 discuss how processes can interact. Chapter 6 discusses resource management, including the classic case study: the "dining philosophers" problem; chapter 7 discusses inter-process communication.

Chapters 8, 9 and 10 look at the applications of concurrency. Chapter 8 looks at simulation, chapter 9 discusses real-time systems, and chapter 10 shows how concurrency applies to operating system design and development.

Finally, chapter 11 examines approaches other than the monitor concept for process coordination, and in doing so takes a brief overview of other languages that incorporate concurrency.

Overall, **Concurrent Program Structures** is a very good textbook. It is well written, the language being concise enough to give each page a high information content, yet not so terse as to render the text difficult to read. There are many examples and case studies to reinforce the ideas expressed. Each chapter ends with a summary of the material presented, a "further reading" list and a set of exercises, the answers for which are in the back of the book. And in all it presents a good methodology for the design and documentation of concurrent programs.

However, I would have two reservations in recommending this book as a textbook for a university course. First is the choice of Pascal Plus as the teaching language, as Pascal Plus does not have as many implementations as other suitable languages, for example Modula II. Second is the price: $133 is a big slice out of the average student's book budget.

Apart from those reservations, I feel that **Concurrent Program Structures** is a good book, and I enjoyed reading it.

AUUG BOOK CLUB
&
PRENTICE HALL AUSTRALIA.

20% DISCOUNT TO AUUG MEMBERS

Please send me copy/copies of the following books —

☐
Vesely / **Cobol**
RRP $80.50* ISBN: 1385-4050-0 Cloth 1989

☐
Nijssen / **Conceptual Schema and Relational Database Design**
RRP $47.95* ISBN: 1316-7263-0 Cloth 1989

☐
Bustard / **Concurrent Program Structures**
RRP $132.95* ISBN: 1316-7289-4 Cloth 1989

☐
Wallia / **Hands on Ventura 2.0**
RRP $43.95* ISBN: 1320-2276-1 Paper 1989

☐
Perihelion / **Helios Operating System**
RRP $59.50* ISBN: 1338-6004-3 Paper 1989

☐
Keller / **Operating Systems**
RRP $138.50* ISBN: 1363-8040-9 Cloth 1989

* Deduct 20% from listed retail prices.

Name:_____ Organisation: _____

Address: _____
(Street address only)
_____ Telephone: _____

Enclosed cheque for $ _____ (Payable to 'Prentice Hall Australia')
Please charge my: ☐ Bankcard ☐ Visa ☐ MasterCard
Credit Card No: ☐☐☐☐ ☐☐☐☐ ☐☐☐☐ ☐☐☐☐
Expiry Date: _____ Signature: _____

Mail completed order form to Prentice Hall Australia, PO Box 151, Brookvale NSW 2100

OR ☎ **Use our FAST PHONE SERVICE call Liz Guthrie.**
Have your credit card ready (9am to 5pm)
SYDNEY (02) 939 1333

Prentice Hall Australia Pty Ltd
7 Grosvenor Place, Brookvale NSW 2100
Tel: (02) 939 1333   Fax: (02) 938 6826

X TECHNICAL BIBLIOGRAPHY

compiled by Ken Lee
DEC Western Software Laboratory
Palo Alto, California
<klee@wsl.dec.com>

Edition of February 5, 1990

Reprinted with the permission of the author.

Asente, Paul

"Simplicity and Productivity," *UNIX Review*, vol. 6, no. 9, pp. 57-63. A discussion on the classing mechanism in the X Toolkit.

ASP, Inc.

X Manual Set (3 volumes), ASP, Inc., 1989. A copy of the MIT X11R3 manual set. Volume 1 covers popular client programs. Volume 2 covers Xlib. Volume 3 covers Xt and the Athena widgets. An upcoming Volume 4 covers Motif.

ASP, Inc.

X Reference Set (3 volumes), ASP, Inc., 1989. A 3 volume set of quick reference guides covering Xlib, Xt, Athena widgets, and popular clients.

Baldwin, Howard

"Why All The Shouting Over X Terminals?" *UNIX World*, Networking Supplement, pp. 75-81, 1989. A broad look at X terminals, with details on 6 vendor's products.

Brunet, James

"Using PC's as X Servers," *UNIX World*, Networking Supplement, pp. 83-85, 1989. An introduction to X server software for IBM PC's.

Brunhoff, Todd

"Pleasing The Eye," *UNIX Review*, vol. 7, no. 10, pp. 64-72. An introduction to VEX, one of the proposed video extensions to X.

Cashin, Jerry

"Many Struggle to Set Laws of Windows Game," *Software Magazine*, vol. 9, no. 2, pp. 74-79, February, 1989. A description of the players in the window system standards war. Includes a photo of Bob Scheifler.

Champine, George

Bob Scheifler, Jim Gettys, Georges Grinstein, and Bertram Herzog, "Panel discussion on the X Window System," in **SIGGRAPH'88 Panel Proceedings**, August, 1988. This is the transcript of an interesting panel discussion on X. George Champine was moderator. Panelists were Bob Scheifler, Jim Gettys, Georges Grinstein, and Bertram Herzog. The panelists discussed the history of X and current work at/on the X Consortium, PEX, and national and international standards for X. This was followed by a lively question and answer session on the present and future of X.

Farrow, Rik

"Before their time?" *UNIX World*, pp. 75-81, July, 1989. This article describes X terminals in general and compares two models. The comparison is, unfortunately, not especially meaningful.

Gancarz, M.

"UWM: A User Interface for X Windows," in **Proceedings of the Summer, 1986 USENIX Conference**, pp. 429-440. Describes UWM, the first good window manager for X.

Gettys, Jim

"Problems Implementing Window Systems in UNIX," in **Proceedings of the Winter, 1986 USENIX Conference**, pp. 89-97. Describes some of the early work on X, especially technical issues in implementing server-based window systems on UNIX systems.

Hancock, Bill

"DECwindows: X'ing With A Vengeance," *DEC Professional*, vol. 8, no. 2, pp. 70-84, February, 1989. A description of DECwindows, DEC's

product version of X. It includes enhanced X libraries, a tuned X server, a high-level user interface description language, and several X applications.

Hopgood, F. R. A.

Methodology of Window Management, Springer-Verlag, New York, 1986. ISBN 0-387-16116-3. The proceedings of the important 1985 Alvey Workshop on Window Management. Many of the big names apparently were here. It includes interesting material on problems with UNIX window systems and proposed solutions. It also includes descriptions of early versions of current UNIX window systems, such as X and Sun's NeWS.

Johnson, Eric

and Kevin Reichard, **X Window Applications Programming**, MIS: Press. ISBN 1-55828-016-2. Another tutorial on Xlib programming. This book has lots and lots of examples, but coverage of complex subjects is not as deep as Jones' book or as broad as the O'Reilly Xlib tutorial. Unfortunately, many of the example programs are not well behaved, so be careful.

Jones, Oliver

Introduction to the X Window System, Prentice-Hall, 1988. ISBN 0-13-499997-5. An excellent introduction to programming with Xlib. Written with the programmer in mind, this book includes many practical tips that are not written anywhere else. This book is not as broad as the O'Reilly Xlib tutorial and doesn't offer as many examples as the Johnson & Reichard book, but Jones is probably the most experienced X programmer of this group and this shows in the quality of the material in the book.

Lee, Ed

"Window of Opportunity," *UNIX Review*, vol. 6, no. 6, pp. 47-61. This article proposes a user interface reference model and shows how X attempts to meet it. In the process, it presents a good, broad overview of X, its advantages, and its shortcomings.

Leffler, Samuel J.

"A Window On The Future?" *UNIX Review*, vol. 6, no. 6, pp. 62-69. This article compares Sun's NeWS with other window systems, such as X, and concludes that, despite its problems, NeWS is the window system of the future. See Robin Schaufler's paper for a more technical comparison of the X and NeWS.

Lemke, David

and David S. H. Rosenthal, "Visualizing X11 Clients," in **Proceedings of the Winter, 1989 USENIX Conference**, pp. 125-138. A detailed look at "visuals", the X abstraction of display hardware capabilities. Proper use of visuals is necessary for an X client to take advantage of the capabilities of a server's hardware. Included in the MIT X distribution (X11R4/doc/tutorials/visuals).

Linton, Mark A.

John M. Vlissides, and Paul R. Calder, "Composing User Interfaces with InterViews," *IEEE Computer*, vol. 22, no. 2, pp. 8-22, February, 1989. Describes the InterViews user interface toolkit, written in C++ and based on the X Window System. Even if you don't plan to use InterViews as a toolkit, it provides a good C++ binding to the X protocol.

Macklem, Rick

Jim Linders, and Hugh Smith, "G Shell Environment," in **Proceedings of the Summer, 1988 USENIX Conference**, pp. 15-22. Describes a rudimentary graphical UNIX shell written as a X client.

McCormack, Joel

and Paul Asente, "Using the X Toolkit or How to Write a Widget," in **Proceedings of the Summer, 1988 USENIX Conference**, pp. 1-13. An excellent tutorial on writing basic X Toolkit widgets. Potential widget writers (and maybe users, too) should probably start by reading this paper. Unfortunately, it's based on X11R2, so some of it is out of date.

McCormack, Joel	and Paul Asente, "An Overview of the X Toolkit," in **Proceedings of the ACM SIGGRAPH Symposium on User Interface Software**, pp. 46-55, October, 1988. An excelent architectural overview of the X Toolkit, including its goals, how it accomplished them, and possible future directions.
McCormack, Joel	Paul Asente, and Ralph Swick, **X Toolkit Intrinsics - C Language Interface**, DEC Press, 1989. Not yet published, but should be a much improved version of the MIT X Toolkit intrinsics manual, by the same authors.
McCormack, Joel	"Smart Code, Stupid Memory: A Fast X Server for a Dumb ColorFrame Buffer," DEC Western Research Laboratory Technical Note 9, September, 1989. Describes the implementation of one of the best X servers currently available, that of the color DECstation 3100. Algorithms as well as performance measurements are discussed. For information on obtaining free copies of this report, send eletronic mail to "wrl-techreports@decwrl.dec.com". Your message should contain the single word "help".
McLoughlin, Lee	"A Simple Guide to Porting the X Window System," in **Proceedings of EUUG Spring 1989**, pp. 283-291, Brussels, April 3-7, 1989. A brief, but good, discussion of the interface between the X server and the underlying operating system.
Mehta, Sunil	"User Interfaces and the IEEE P1201 Committee," *UNIX Review*, vol. 8, no. 1, pp. 14-20. A description of the activities of the IEEE P1201 standards committee, by its chairman. P1201 is developing international standards for the the X Window System libraries (Xlib and above) and user interface development tools based on X. Lower levels of X (servers and protocol) are being standardized by ANSI Task Group X3H3.6.
MIPS Staff	"Graphical User Interfaces Now," *MIPS Magazine*, vol. 1, no. 11, pp. 42-50, November, 1989. A comparison of popular user interfaces including Microsoft Presentation Manager, NextStep, Macintosh, Motif, and Open Look. Motif and Open Look are both available for the X Window System. The IXI X.desktop and Visix Looking Glass UNIX file managers, both based on X, are also mentioned. See also the review by Alan Southerton.
Myers, Brad A.	"Window Interfaces: A Taxonomy of Window Manager User Interfaces," *IEEE Computer Graphics & Applications*, vol. 8, no. 5, pp. 65-84, September, 1988. A taxonomy of current window system user interfaces, including the UWM window manager for X. Discusses and compares the features of each user interface. Contrast this with Rob Pike's window system user interface guidelines. Myers is a well known user interface researcher.
Nadeau, David R.	"High-Performance 3-D Graphics In A Window Environment," *Computer Technology Review*, pp. 89-93, Fall, 1988. A discussion on integrating Megatek's high-performance 3D graphics hardware/software with X.
Nye, Adrian	"The X Window System Protocol," *UNIX World*, vol. 6, no. 9, pp. 105-113, September, 1989. An introduction to the X protocol. This is a short version of Nye's introduction to Volume 0 of O'Reilly's X series.
O'Reilly, Tim	Reilly, Tim'u>(1250u+1n) .br "The Toolkits (and Politics) of X Windows," *UNIX World*, vol. 6, no. 2, pp. 66-73, February, 1989. An introduction to some of the popular widget sets available for the X Toolkit.

O'Reilly and Associates	Reilly and Associates'u>(1250u+1n) .br **The X Window System Series (4 volumes)**, O'Reilly and Associates, 1988, 1989. ISBN 0-937175-26-9, 0-937175-27-7, 0-937175-38-2, etc. This is a 4 (and growing) volume set of books. Volume 1 is a tutorial on Xlib. It's broader than the others (Jones, Johnson & Reichard), but offers fewer examples or practical tips. Volumes 0, 2, and 3 are approximately the same as the MIT manuals (protocol manual, Xlib manual pages, and popular client manual pages). Older printings were based on X11R2, but newer ones are updated to X11R3. Future volumes will reportedly cover the X Toolkit intrinsics, XView, and Motif. Written by technical writers, these are probably the most professional looking of the X books.
Open Software Foundation	**OSF/Motif Series (5 volumes)**, Prentice Hall, 1990. ISBN 0-13-640491-X, 0-13-640525-8, 0-13-640517-7, 0-13-640509-6, 0-13-640483-9. The volumes include Motif Style Guide, Programmer's Guide, Programmer's Reference, User's Guide, and Application Environment Specification (AES) User Environment Volume, Motif is a user interface programming environment for X. It was designed and developed by members of the Open Software Foundation, including DEC, HP, IBM. These companies have committed to using Motif as their standard user interface in the near future.
Pike, Rob	"Window Systems Should Be Transparent," *USENIX Computing Systems*, vol. 1, no. 3, pp. 279-296, Summer, 1988. Some window system user interface guidelines, based on MUX window system for the BLIT terminal, with some comparisons to X. Contrast this to Myers' paper. Pike is the developer of the BLIT.
Pike, Rob	"A Concurrent Window System," *USENIX Computing Systems*, vol. 2, no. 2, pp. 133-153, Spring, 1989. A description of a multi-threaded window system written using a concurrent programming language. The design is shown to be simpler than the single threaded design of X.
Probst, Richard	"OPEN LOOK Toolkits," *SunTechnology*, vol. 1, no. 4, pp. 76-86, Autumn, 1988. OPEN LOOK is a user interface specification designed by AT&T and Sun. This paper discusses OPEN LOOK programming toolkits, including two for the X Window System.
Rao, R.	and S. Wallace, "The X Toolkit," in **Proceedings of the Summer, 1987 USENIX Conference**. Describes an early version of the X Toolkit.
Rochkind, Marc J.	"XVT: A Virtual Toolkit for Portability Between Window Systems," in **Proceedings of the Winter, 1989 USENIX Conference**, pp. 151-163. Describes the Extensible Virtual Toolkit (XVT), a user interface toolkit that provides a uniform application programming interface to several different window systems, while providing the "look and feel" appropriate for the window system. Supported window systems include X11, Microsoft Windows, OS/2 Presentation Manager, and the Macintosh.
Rosenthal, David S. H.	"Window Exchange," *UNIX Review*, vol. 7, no. 12, pp. 58-64. An excelent review of X client interoperability, by the author of the X11R4 Inter-Client Communication Conventions Manual (ICCCM). It discusses some common programming errors in this area and how proper programming and attention to the IC3M will make your programs more robust.
Rosenthal, David S. H.	"A Simple X11 Client Program," in **Proceedings of the Winter, 1988 USENIX Conference**, pp. 229-235. A version of the "hello, world" paper, presenting and comparing the basics of the X library and the X Toolkit. All potential X programmers (Xlib or X toolkit) should understand everything in this paper before they attempt to write any X

programs. Included in the MIT X distribution (X11R4/doc/tutorials/HelloWorld) and reprinted as "Going for Baroque" in the June or July, 1988 (vol. 6, no. 6) issue of *UNIX Review* magazine.

Rost, Randi — Jeffrey Friedberg, and Peter Nishimoto, "PEX: A Network-Transparent 3D Graphics System," *IEEE Computer Graphics & Applications*, pp. 14-26, July, 1989. A good overview of PEX, the PHIGS/PHIGS+ 3D extension to X. A complete PEX is currently being developed by Sun under contract to the MIT X Consortium and is scheduled to be publically available in 1990. See also the Thomas & Friedmann paper on PEX.

Schaufler, Robin — "X11/NeWS Design Overview," in **Proceedings of the Summer, 1988 USENIX Conference**, pp. 23-35. Discusses Sun's X/NeWS window system. Compares the functionality of X and NeWS and describes a server that merges the two.

Scheifler, Robert — and Jim Gettys, "The X Window System," *ACM Transactions on Graphics*, vol. 5, no. 2, pp. 79-109, April, 1986. The first published description of X. Although it discusses X10, it is still one of the most comprehensive overviews of X. An updated X11 version is reportedly being written by Jim Gettys.

Scheifler, Robert — James Gettys, and Ron Newman, **X Window System: C Library and Protocol Reference**, DEC Press, 1988. ISBN 1-55558-012-2. The Xlib bible by the authors of X11. It is shipped with the X11R3 tapes. Includes detailed descriptions of the X protocol and all Xlib functions and data structures. Required for all serious Xlib programmers, but may be rough going for those with little experience in interactive computer graphics. An X11R4 version of this book is in preparation.

Shein, Barry — "Primal Screens," *Sun Expert*, vol. 1, no. 3, pp. 56-69, January, 1990. An interesting, though somewhat religious, comparison of X and Sun's NeWS window system. It concludes, "even Scott McNealy has spoken of NeWS as a failure as a standard, although he hastens to add that it was a technological success."

Southerton, Alan — "Friendly Desktops," *UNIX World*, vol. 6, no. 11, pp. 68-73, November, 1989. Another comparison of X-based UNIX file managers, including Non Standard Logic InDepth, IXI X.Desktop, Visix Looking Glass, AT&T Open Look, Torch Technologies Y-OpenTop, and Softscience Convenience Plus. See also the review by MIPS.

Steven Mikes — **X Window System Technical Reference**, Addison-Wesley, 1990. ISBN 0-201-52370-1. A quick reference manual for X11R3, including Xlib, Xt, fonts, bitmaps, keysyms, Motif, and OPEN LOOK.

Stroyan, Michael — "Three-Dimensional Graphics Using the X Window System," **Dr. Dobb's Journal**, vol. 15, no. 2, pp. 28-36, February, 1990. A high level description of various approaches to developing 3D graphics tools for X, including those of the PHIGS Extension to X (PEX) and HP's Starbase-on-X11 (sox11).

Sun Microsystems — **OPEN LOOK Graphical User Interface Series**, Addison-Wesley, 1990. ISBN 0-201-52365-5, ISBN 0-201-42364-7. This series includes Functional Specifications and Application Style Guide. OPEN LOOK was designed by AT&T and Sun and implementations are available from both. AT&T's implementation uses the X Toolkit. Sun's implementation uses XView, a similar, but not compatible, toolkit.

Swick, Ralph R. — and Mark S. Ackerman, "The X Toolkit: More Bricks for Building User Interfaces," in **Proceedings of the Winter, 1988 USENIX Conference**,

pp. 221-233. An introduction to the X Toolkit. Describes the toolkit architecture and the basic Athena widgets.

Thomas, Spencer W.   and Martin Friedmann, "PEX - A 3-D Extension to X Windows," in **Proceedings of the Winter, 1989 USENIX Conference,** pp. 139-149. Describes the demonstration implementation of PEX, the PHIGS/PHIGS+ 3D extension to X. A complete PEX is currently being developed by Sun under contract to the MIT X Consortium and is scheduled to be publically available in 1990. See also the Rost, et al, paper on PEX.

Wood, Patrick   "Introduction to Xlib Programming (2 parts)," *UNIX World*, vol. 6, October-November, 1989. This is a two part tutorial article on Xlib programming. Not as good as Rosenthal's "hello, world" article.

X/Open Company, Ltd.   **X/Open Portability Guide: Window Management,** Prentice-Hall, 1988. X/Open is an international user group of UNIX hardware and software developers. It publishes portability guidelines with the goal that developers that follow these guidelines will be portable across other hardware and software supporting the guidelines. The X/Open window management guidelines are based on the X11R3 specifications.

Young, Doug   "Programming with the X Toolkit (3 parts)," *UNIX World*, vol. 7, no. 1-3, January-March, 1990. This three part tutorial article is based on Young's books on the X Toolkit.

Young, Douglas A.   **X Window Systems Programming and Applications With Xt,** Prentice-Hall, 1989. ISBN 0-13-972167-3. An excellent (and currently the only) tutorial on programming with the X Toolkit intrinsics. Examples in this book use the HP widgets, but a Motif edition is scheduled to be released soon.

Young, Douglas A.   **X Window System: Programming and Applications With Xt,** **OSF/Motif Edition,** Prentice-Hall, 1990. ISBN 0-13-497074-8. This is the Motif version of the above.

COMMONLY ASKED X QUESTIONS

compiled by the X User's Group
<xug@ics.com>

Edition of March 8, 1990

Reprinted with the permission of the author(s).

This article contains the answers to some Frequently Asked Questions often seen in comp.windows.x. It is printed here to help reduce volume in this newsgroup and to provide hard-to-find information of general interest.

This article includes answers to these questions (they are in no particular order yet):

0) Where can I obtain X source?
1) Where can I obtain X11R4?
2) Where can I obtain Motif?
3) Where can I obtain software implementing Open Look?
4) Where can I obtain other X sources?
5) What is the xstuff mail-archive?
6) Where can I obtain patches to X11R4?
7) Where can I find books and articles on X that are good for beginners?
8) What courses on X are available?
9) Is there a skeleton X program available?
10) What are these common abbreviations?
11) What is XUG?
12) What is the X Consortium and how do I join?
13) What conferences on X are coming up?
14) What is the current state of the world in X terminals?
15) How can I get an X server on a PC?
16) How can I get an X server on a Macintosh running MacOS?
17) Where can I obtain an X-based editor or word-processor?
18) Where can I obtain an X-based paint/draw program?
19) Where can I obtain an X-based spreadsheet?
20) Where can I get a PostScript previewer for X?
21) Where can I get a troff previewer for X?
22) How do I convert Mac/TIFF/GIF/Sun/PICT/Face/img/FAX/etc images to X?
23) How do I use an alternate window manager with DEC's session manager?
24) How do I build X with gcc?
25) What are these funny problems compiling X11R3 on the Sun4?
26) What are these funny problems installing X11R4 on the Sun running SunOS 4?
27) Where can I get a fast X server for a workstation?
28) How can I change the titlebar of my xterm window?
29) Why doesn't anything appear when I run this simple program?
30) What is the difference between a Screen and a screen?
31) Why do I get a BadDrawable error drawing to XtWindow(widget)?
32) Can I get the background pixel/pixmap of a window using XGetWindowAttributes?
33) Why does the pixmap I copy to the screen show up as garbage?
34) Why doesn't my program get the keystrokes I select for?
35) Why does the R3 xterm, et al, fail against the R4 server?
36) Where can I obtain alternate language bindings to Xlib?

If you have suggestions or corrections for any of these answers or any additional information, please send them directly to xug@expo.lcs.mit.edu; the information will be included in the next revision (or possibly the one after that; thanks for the many suggestions which haven't been incorporated yet). The answers in this iteration are acknowledged to be partial and to lean more towards products than to programming.

The information contained herein has been gathered from a variety of sources. In many cases attribution has been lost; if you would like to claim responsibility for a particular item, please let us know.

Conventions used below: telephone numbers are Bell-system unless otherwise noted; prices on items are generally not included.

0) Where can I obtain X source?

Intelligent Software Products, Inc., (516-766-2867) ships X11R3 [formats are unknown].

Integrated Computer Solutions, Inc., (617-547-0510) ships X11R3 on half-inch and quarter-inch formats.

The Free Software Foundation (617-876-3296) sells X11R3 on half-inch tapes and on QIC-24 cartridges.

Automata Design Associates (215-646-4894) sells X11R3 source on 5.25" high-density floppies and QIC-24 quarter-inch cartridge tapes.

European sites can obtain a free distribution from an unnamed Good Samaritan who may be reached at chx400!pan!jw or cernvax!pan!jw. [1/90]

IXI Limited (+44 223 462 131) is selling X11R3 source on quarter-inch cartridge formats and on 5.25" and 3.5" floppy, with other formats available on request. [IXI, 2/90]

[Note that some distributions are media-only and do not include docs.] [The MIT Software Center no longer distributes X11R3.] (See below for FTP sites.)

1) Where can I obtain X11R4?

The MIT Software Center is shipping X11R4 on four 1600bpi half-inch tapes. Call the X Hotline at (617) 258-8330 for prerecorded ordering information and a good product description.

Integrated Computer Solutions, Inc., ships X11R4 on half-inch, quarter-inch, and TK50 formats. Call 617-547-0510 for ordering information.

The Free Software Foundation (617-876-3296) sells X11R4 on half-inch tapes and on QIC-24 cartridges.

Yaser Doleh (doleh@math-cs.kent.EDU; P.O. Box 1301, Kent, OH 44240) is making X11R4 available on HP format tapes, 16 track, and Sun cartridges.

European sites can obtain a free X11R4 distribution from an unnamed Good Samaritan who may be reached at chx400!pan!jw or cernvax!pan!jw. [1/90]

IXI Limited (+44 223 462 131) is selling X11R4 source on quarter-inch cartridge formats and on 5.25" and 3.5" floppy, with other formats available on request. [IXI, 2/90]

Virtual Technologies (703-430-9247) provides the entire X11R4 compressed source release on a single QIC-24 quarter-inch cartridge and also on 1.2meg or 1.44 meg floppies upon request. [Conor Cahill (cpcahil@virtech.uu.net) 2/90]

[Note that some distributions are media-only and do not include docs.]

Canadian sites can send email to xhacks@csri.toronto.edu to arrange for the exchange of tapes; the offer is subject to "time availability". [information from Mark Moraes (moraes@csri.toronto.edu), 2/90]

UK sites can obtain R4 through the UKUUG Software Distribution Service, from the Department of Computing, Imperial College, London, in several tape formats. You may also obtain the source via Janet (and therefore PSS) using Niftp (Host: uk.ac.ic.doc.src Name: guest Password:

your_email_address). Queries should be directed to Lee McLoughlin, 01-589-5111#5037, or to ukuug-soft@uk.ac.ic.doc.

X11R4 is ftp-able from expo.lcs.mit.edu; these sites are preferable, though, and are more direct:

	Location	Machine Name	Internet Address	FTP Directory
(1)	West USA	gatekeeper.dec.com	16.1.0.2	pub/X11/R4
	Central USA	mordred.cs.purdue.edu	128.10.2.2	pub/X11/R4
(2)	Central USA	giza.cis.ohio-state.edu	128.146.8.61	pub/X.V11R4
	Southeast USA	uunet.uu.net	192.48.96.2	X/R4
(3)	Northeast USA	crl.dec.com	192.58.206.2	pub/X11/R4
(4)	UK Janet	src.doc.ic.ac.uk	129.31.81.36	X.V11R4
	UK niftp	uk.ac.ic.doc.src		<XV11R4>
(5)	Australia	munnari.oz.au	128.250.1.21	X.V11/R4

The giza.cis.ohio-state.edu site, in particular, is known to have much of the contrib stuff that can be found on expo. In addition, the same stuff is also available via anonymous UUCP from osu-cis, at TB+ and V.32 speeds. Write to uucp@cis.ohio-state.edu (same as osu-cis!uucp) for instructions. [thanks to Bob Sutterfield, bob@MorningStar.Com, 2/90]

The release is available to DEC Easynet sites as CRL::"/pub/X11/R4".

2) Where can I obtain Motif?

Various hardware vendors produce developer's toolkits of binaries, header files, and documentation; check your hardware vendor, particularly if that vendor is an OSF member. Systems known to be shipping now: HP (sans UIL), Apollo (sans UIL), SCO, ISC. In addition, independent binary vendors produce Motif toolkits. ICS makes several binary kits, notably for Sun, Apple; Quest (408-988-8880) sells kits for Suns, as well.

An OSF/Motif source license must be obtained from OSF before source can be obtained from the Open Software Foundation or any value-added vendor. Call the Motif Desk at OSF at 617-621-8835 for ordering information.

3) Where can I obtain software implementing Open Look?

Sun's XView has a SunView-style API. A new version is on the X11R4 tape; a newer version is also available (as of 2/90) on expo.lcs.mit.edu for anonymous ftp.

AT&T's Open Look GUI 2.0 Xt-based toolkit is now generally available [2/90]; contact 1-800-828-UNIX#544 for information.

Sun is shipping OpenWindows 1.0 for Sparc, Sun-3, and Sun386i machines; contact your local sales representative for more details.

Solbourne's extensible C++-based Object Interface Library will be distributed by AT&T [date of availability appx. 6/90].

4) Where can I obtain other X sources?

User-contributed software is distributed through the newsgroup comp.sources.x, moderated by Dan Heller (argv@sun.com); also check that group for posting information.

Miles O'Neal of Sales Technologies, Inc., started a Public Domain Widget Library in 11/88. The PDWL is a repository of widgets donated for the use of other X programmers and also of toolkit documentation, widget-writing documentation, better documentation for extant widgets and toolkits, and widget-writing tools.

In addition, the PDWL also stores information on commercially available toolkits and other sites from which public-domain widget-related and X-related stuff may be obtained. It is a place for non-commercial stuff not readily available elsewhere.

You can access the PDWL by sending electronic mail to the account gatech!stiatl!xwidgets. Send a Subject line of 'help' to obtain more information on retrieving widgets and on submitting sources.

The machine expo.lcs.mit.edu has a great deal of user-contributed software in the contrib/ directory; a good deal of it is present in current or earlier versions on the X11R3 and X11R4 contrib tapes. There is a new directory contrib/R4fixes/ for fixes to R4 contrib software. [Jim Fulton, 2/90]

A new west-coast UUCP X11 Archive is administered by Mark Snitily (mark@zok.uucp) and contains the full X11R4 distribution, the XTEST distribution, an entire archive of comp.sources.x and other goodies.

The machine zok has a TB+ modem which will connect to 19.2K, 2400, 1200 baud (in that order). The anonymous UUCP account is UXarch with password Xgoodies. The modem's phone number is 408-996-8285.

A sample Systems (or L.sys) entry might be:

```
zok Any ACU 19200 4089968285 in:--in: UXarch word: Xgoodies
```

To get a current listing of the files that are available, download the file "/usrX/ls-lR.Z".

A full subject index of the comp.sources.x files is available in the file "/usrX/comp.sources.x/INDEX".

The machine has just the one modem, so please do not fetch large amounts of data at one sitting. [courtesy Mark Snitily, 2/90]

FTP sites and software available (list as of X11R3; also see above):

brazos.rice.edu	128.42.42.2	pub/X11R3/core.src
charon.mit.edu	18.80.0.13	perl+patches, xdvi
cs.purdue.edu	128.10.2.1	rcs,xspeed
j.cc.purdue.edu	128.210.0.3	comp.sources.{unix,x,amiga}, elm, uupc
nl.cs.cmu.edu	128.2.222.56	Fuzzy Pixmap 0.84 in /usr/mlm/ftp
shambhala.berkeley.edu	128.32.132.54	xrn
terminator.cc.umich.edu	35.1.33.8	xscheme, msdos, atari
cayuga.cs.rochester.edu	192.5.53.209	Xfig,LaTeX styles,Jove
cfdl.larc.nasa.gov	128.155.24.55	gnu, rfc, sun, X, ucb, odu, vm
cheddar.cs.wisc.edu	128.105.2.113	Common Lisp stuff, X11 courier fonts
cs.orst.edu	128.193.32.1	Xlisp
dinorah.wustl.edu	128.252.118.101	X11R3/core.src
expo.lcs.mit.edu	18.30.0.212	a home of X, portable bitmaps
gatekeeper.dec.com	128.45.9.52	X11,etc...
giza.cis.ohio-state.edu	128.146.8.61	X11R3, PEX
hotel.cis.ksu.edu	129.130.10.12	XBBS, msdos, U3G toolkit
icarus.riacs.edu	128.102.64.1	SLIP, chkpt, macdump, Xpostit
interviews.stanford.edu	36.22.0.175	InterViews X toolkit
jpl-mil.jpl.nasa.gov	128.149.1.101	Tex, Mac, Gnu, Xv11R{2,3}
m9-520-1.mit.edu	18.80.0.45	Xim (X image viewer)
mordred.cs.purdue.edu	128.10.2.2	X11R3
polyslo.calpoly.edu	129.65.17.1	src/spaceout.tar.Z for X11
scam.berkeley.edu	128.32.138.1	X sources, etc.
sun.soe.clarkson.edu	128.153.12.3	X11 fonts, TeX
think.com	10.4.0.6	X11.2 Interviews 3d

vaxa.isi.edu	128.9.0.33	X, db
wheaties.ai.mit.edu	128.52.32.13	"tX11"
xanth.cs.odu.edu	128.82.8.1	comp.srcs.{x,unix,misc,games,amiga},X

[This is from a file posted in early July 1989; sorry: attribution is lost, and this list does need updating.]

In addition, UUNET Source Archives (703-876-5050) tracks comp.sources.x and provides 600MB+ of compressed programs on two 6250 bpi or five 1/4" tapes.

5) What is the xstuff mail-archive?

The xstuff server is a mail-response program. That means that you mail it a request, and it mails back the response.

Any of the four possible commands must be the first word on a line. The xstuff server reads your entire message before it does anything, so you can have several different commands in a single message (unless you ask for help). The xstuff server treats the "Subject:" header line just like any other line of the message.

The archives are organized into a series of directories and subdirectories. Each directory has an index, and each subdirectory has an index. The top-level index gives you an overview of what is in the sub-directories, and the index for each subdirectory tells you what is in it.

(1) The command "help" or "send help" causes the server to send you a more detailed version of this help file.

(2) if your message contains a line whose first word is "index", then the server will send you the top-level index of the contents of the archive. If there are other words on that line that match the name of subdirectories, then the indexes for those subdirectories are sent instead of the top-level index. For example, you can say "send index fixes" (or "index fixes"). A message that requests an index cannot request data.

(3) if your message contains a line whose first word is "send", then the xstuff server will send you the item(s) named on the rest of the line. To name an item, you give its directory and its name. For example

<div align="center">send fixes 1 3-4</div>

You may issue multiple send requests. The xstuff server contains many safeguards to ensure that it is not monopolized by people asking for large amounts of data. The mailer is set up so that it will send no more than a fixed amount of data each day. If the work queue contains more requests than the day's quota, then the unsent files will not be processed until the next day. Whenever the mailer is run to send its day's quota, it sends the requests out shortest-first.

(4) Some mailers produce mail headers that are unusable for extracting return addresses. If you use such a mailer, you won't get any response. If you happen to know an explicit path, you can include a line like

<div align="center">path foo%bar.bitnet@mitvma.mit.edu</div>

or

<div align="center">path bar!foo!frotz</div>

in the body of your message, and the daemon will use it.

The xstuff server itself can be reached at xstuff@expo.lcs.mit.edu. If your mailer deals in "!" notation, try sending to {someplace}!eddie!expo.lcs.mit.edu!xstuff.

[based on information from the MIT X Consortium, 8/89.]

6) Where can I obtain patches to X11R4?

The xstuff server now has five patches for X11R4 [3/90]. Send to xstuff the Subject line

send fixes 1-5

Patches are typically also distributed through the newsgroup comp.sources.x, with some lagtime. Some source re-sellers may be including patches in their source distributions of X11R4.

7) Where can I find books and articles on X that are good for beginners?

Ken Lee of the DEC Western Software Laboratory regularly posts to comp.windows.x and ba.windows.x a list of reference books and articles on X and X programming. This list appears on page 19 of this issue of AUUGN.

In addition, check the X11R4 core distribution in doc/tutorials for some useful papers and tutorials, particularly the file doc/tutorials/answers.txt. "Late Night's Top Ten X11 Questions" by Dave Lemke (lemke@ncd.com) and Stuart Marks (smarks@sun.com) answers other common questions and some of these here in more detail.

8) What courses on X are available?

Lurnix offers 4-day "type-along courses" on Xt; the course is being ported from Xaw to Xm. Information is available at 800-433-9337 (in CA: -9338).

Integrated Computer Solutions, Inc., offers several multi-day, hands-on courses on X, Xt, and the Xaw, HP, and Motif widget sets, in particular. Information is available at 617-547-0510 and info@ics.com.

Intelligent Visual Computing teaches several Xt-based lab courses on-site. IVC is at 919-481-1353 or at info@ivc.uu.net.

Various vendors are also beginning to offer X training, usually specific to Xt and a proprietary widget set; HP and DEC are also offering Xlib courses. Sun offers an XView course.

IXI Limited (+44 223 462 131) offers regular X training courses for both programmers and non-technical managers.

Advanced Computing Environments periodically offers at least a two-day Introduction course. Contact Susie Karlson at 415-941-3399 for information.

Various universities are offering short X courses or overviews: UCLA, Dartmouth, University of Lowell, University of Canberra (within Australia: 062-522422) ...

Among the best places to find courses are at the various Unix conferences — Uniforum, Usenix, Unix Expo, Xhibition, he MIT X Technical Conference, AUUG'90 Winter Conference, &c.

9) Is there a skeleton X program available?

There is no general framework such as the TransSkel program for the Macintosh which handles lots of the odds and ends and overhead of development under a window system and which can be used as a platform for additional development. In X, the problem is typically solved by using an interactive application builder tool or by using cut&paste on existing X applications. Good applications which you might look to manipulate when you want to "test just this one little thing" include contrib/clients/xskel, a simple program that puts up a window and allows sketching in it and offers a starting point for quick hacks, the Xaw examples in the examples/ directory in the R3 and R4 distributions, and the Xlib "Hello World" example in the R3 doc/HelloWorld and R4 doc/tutorials/HelloWorld; an updated version of this program which uses R4 Xlib calls and current ICCCM conventions was posted in 2/90 to

10) What are these common abbreviations?

Xt　　　The X Toolkit Intrinsics is a library layered on Xlib which provides the functionality from which the widget sets are built. An "Xt-based" program is an application which uses one of those widget sets and which uses Intrinsics mechanisms to manipulate the widgets.

Xmu　　The Xmu library is a collection of Miscellaneous Utility functions useful in building various applications and widgets.

Xaw　　The Athena Widget Set is the MIT-implemented sample widget set distributed with X11 source.

Xm　　　The OSF/Motif widget set from the Open Software Foundation.

XUI　　DEC's X-programmer's toolkit, including a widget set and a high-level widget description language, is being phased out.

Xhp　　The Hewlett-Packard Widget Set was originally based on R2++, but several sets of patches exist which bring it up to R3, as it is distributed on the X11R4 tapes.

dxwm　　The DECwindows window manager is part of DEC's current X release.

mwm　　The Motif Window Manager is distributed with OSF/Motif source and is available from vendors in binary form.

CLX　　The Common Lisp X Interface is a Common Lisp equivalent to Xlib.

XDMCP　The X Display Manager Protocol provides a uniform mechanism for a display such as an X terminal to request login service from a remote host.

XLFD　　The X Logical Font Description Conventions describes a standard logical font description and conventions to be used by clients so that they can query and access those resources.

ICCCM　The Inter-Client Communication Conventions Manual explains the set of standard conventions which X clients should follow to allow them to cooperate in the areas of selections, cut buffers, window management, session management, and resources. The latest version is on the X11R4 tape.

RTFM　　Common X-speak meaning "please locate and consult the relevant documentation".

UTSL　　A common expression meaning "take advantage of the fact that you aren't limited by a binary license — Use The Source, Luke".

11) What is XUG?

The X User's Group was formed in January of 1988. Its purpose is to encourage X development by providing information on the X Window System to all who are interested.

- Local Area Groups: [this list is in the process of being updated]:

Bay Area	Jim Turner, 415/960-0123
Boston	Mitch Trachtenberg, 617/621-8700
Cleveland	Mike Kolberg, 216/243-1198
New York City	#TBC#
Princeton, NJ	Joe Camaratta, 609/734-6500
Research Triangle Park	Steven Thiedke, 919/481-1353
Washington, DC	Thomas Fagre, 703/866-7425
England	Ray Anderson, +44 223 462131
France	Daniel Dardailler, +33 93 65 77 71
Singapore	Chee Keong Law, 772-3116

- XNextEvent: the several-times-yearly newsletter includes articles of general interest.

To join, form a local group, contribute to XNextEvent, or help out in any other way, contact Alex Fisher at XUG, c/o Integrated Computer Solutions, 163 Harvard Street, Cambridge, MA 02139, 617/547-0634, or email to xug@expo.lcs.mit.edu (please make sure to include a return address, particularly if you connect to the world via a UUCP connection)

12) What is the X Consortium and how do I join?

The MIT X Consortium was formed in January of 1988 to further the development of the X Window System and has as its major goal the promotion of cooperation within the computer industry in the creation of standard software interfaces at all layers in the X Window System environment. MIT's role is to provide the vendor-neutral architectural and administrative leadership required to make this work. Membership in the Consortium open to any organization. There are two categories of membership, Member (for large organizations) and Affiliate (for smaller organizations). Most of the Consortium's activities take place via electronic mail, with meetings when required. As designs and specifications take shape, interest groups are formed from experts in the participating organizations. Typically a small multi-organization architecture team leads the design, with others acting as close observers and reviewers. Once a complete specification is produced, it may be submitted for formal technical review by the Consortium as a proposed standard. The standards process typically includes public review (outside the Consortium) and a demonstration of proof of concept. Your involvment in the public review process or as a Member or Affiliate of the Consortium is welcomed. Write to: Bob Scheifler, MIT X Consortium, Laboratory for Computer Science, 545 Technology Square, Cambridge, MA 02139.

[For complete information see the XCONSORTIUM man page from the X11R4 distribution, from which this information is adapted.] [2/90]

13) What conferences on X are coming up?

Several booths at NCGA'90 (Anaheim, March 19-22) will be showing X. Call 1-800-225-NCGA for more information.

"Discover Unix 1990" (Tyngsboro, MA, April 2-4) has some high-powered X talks. Call Andree Fontaine at 508-649-9731 for attendee information.

The Xhibition90 X trade show and conference is being held in San Jose May 21-25. Xhibition is focused on the X Window System and offers tutorials, panels, presentations, and vendor exhibits. Call Xhibition at 617-547-0510 for information.

The MIT Technical Conference is typically held in January in Boston, mostly for historical reasons.

14) What is the current state of the world in X terminals?

Here is a selection of vendors with "impressions of consensus opinions".

Acer (408-922-0333) has the Xebra 1000, based on an 8086 cpu, with a 640x480 monochrome screen. "Low performance." "May not be sold anymore."

AT&T's (800-247-1212; ask for local dealer) 730X has a 1Kx1K b/w display with a 1:1 aspect ratio. The terminal supports multiple Telnet sessions and AT&T windowing in addition to X. [Starlan only.] "Very, very nice."

C. Itoh (714-660-1421) "Rumor: making a terminal that Sun will OEM, if C. Itoh doesn't pull out of the business first."

DEC (800-343-4040) is slated to ship in 3/90 the VT1000, a home-brew 15" 1024x864 monochrome terminal using the TI 34010.

Gipsi S.A. (+33 (1) 30.60.75.00 or Jeff Abramatic at jfa@gipsi.fr) in 10/89 announced "le tX", a line of 68030-based X terminals running X11R3. High-end models, at least, feature downloadable X servers.

Model	Memory	Resolution	Display	Refresh (Hz)	Price (FF)
M	2 MB	1280x960x1	19" B&W	66	32 400
Me	2 MB	1280x960x2	19" Greyscale	66	38 000
C4	2 MB	1280x768x4	16" Colour	60	59 900
C8	4 MB	1280x1024x8	19" Colour	60	79 400

Expansion is up to 8MB and 8 planes.

The exclusive US distributor is Peripheral Design, Inc (404-263-0067).

GraphOn (800-472-7466) OptimaX 200 runs a server on the host which translates from X protocol to a proprietary protocol which can run over a serial line. The screen is 14". The terminal is based on a 12MHz 68000. (See the December 1989 issue of XNextEvent for an informal review.) "Best available solution for RS232C lines."

HP (800-752-0900; ask for nearest sales office) offers the 700/X series of terminals using on the TI 34010. Bit-mapped graphics monitors with resolutions of 640x480, 800x600, and 1024x768 are supported. All units come standard with 1 Mbyte of RAM expandable up to 4 Mbytes and display 16 colors from a palette of 4096 or up to 16 levels of gray-scale.

Human Design Systems (800-437-1551) offers several combinations of 14", 16", and 19" color, grey-scale, and mono screens, at least 1Kx1K. All support thin and thick Ethernet. High-end models are expandable to 8.5MB. "Slow."

IBM's Xstation 120 starts with 512KB of memory and features support for simultaneous Token-Ring and Ethernet connections. [2/90]

Jupiter Systems (415-523-9000, 508-836-4400) produces the Model 310 which features a 19-inch 1280x1024 color monitor. "A price leader, but also a performance leader."

Network Computing Devices (415-694-0650) offers several terminals. The NCD16 has a 1Kx1K 16" square display, a 12.5MHz 68000, a non-optical mouse, a DEC-influenced keyboard, and an X11R3 server. The base configuration comes with 1.5M memory. There is an option for down-loading the server into RAM. (This is the terminal offered by MIPS and DG; it is also the Tektronix XN5.) There is also a 19" version with 1280x1024 resolution. The new 17c is an 8-bit color 1024x768 display. "Nice engineers' terminals."

NCR (513-445-2033) offers the Towerview with 1024x840 resolution and a PROM-based server. The Towerview supports serial connections. Fonts are down-loaded. The XL15 and XL19 have 15", 1024x800 and 19", 1280x1024 displays, respectively. "Seems to be designed for the PC office."

Princeton Graphic Systems (800-221-1490) has introduced the Ultra X line with monochrome up to 1024x768 and color up to 1024x1280. Expandable to 8MB.

Qume (408-942-4000) has announced an X terminal called the QXT 10 X.

Spectragraphics (619-450-0611) offers an X terminal with emulation for the IBM 3270 and related terminals.

Tektronix (203-877-1494; or Rick Kamp rickka@orca.WV.tek.com) offers the Model 4211 Graphics Netstation using the TI 34010 graphics processor. The 15" screen is 1024x768 color. The XN11 is a PseudoColor device with up to 8 planes. The 16" and 19" monitors have the 1024x768 resolution. There is also an XN10, which features a 19", 1024x768 color monitor.

Visual Technologies (800-VISUALC; MA 508-836-4400) offers the 640X based on a 12MHz 68000. The display is 1024x800 14" monochrome. The 719 increases speed to 16Mhz and offers a 19" display. The 719x Turbo offers grey-scale images. "Reasonable low-cost-per-seat performance station."

Digital Review's 2/26/90 issue evaluates a subset of these terminals.

15) How can I get an X server on a PC?

Locus Computing (800-955-6287; CA: 213-670-6500; UK: +44 296 89911) has a server called PC-Xsight which also appears in Acer's X terminal.

HP (800-752-0900) has the "HP Accelerated X Window Display Server" (HP AXDS/PC; HP part D2300B) which will run on any AT-class DOS machine with 640KB, MSDOS 3.1 or higher, and the HP Intelligent Graphics Controller 10 card, to which the X11R3-based server is downloaded (avoiding performance-limitations from PC RAM-size and processor speed). [from John Kempff (kempff@hppad.hp.com), 3/90]

Graphic Software Systems (GSS) (503-641-2200) makes PC-Xview, an MSDOS-based X server which interfaces with PC/TCP Plus networking software from FTP Software and Excelan's LAN WorkPlace for DOS. The server works with (a) 286, 386, 486 (b) EGA, VGA, DGIS displays. (c) DOS 3.2 and above (d) Microsoft, Logitech, Mouse Systems Mice (e) 640k memory up to 16 MB memory [the PC-Xview/16 is available for PCs with extended memory].

VisionWare's XVision is a Microsoft Windows-based X server which allows an IBM-compatible PC or PS/2 to display X clients running on a networked computer at the same time as local DOS programs. VisionWare is at 612-377-3627 or vision@vware.mn.org (UK: +44 532 788858 and vware@vision.uucp).

Integrated Inference Machines (714-978-6201 or -6776) is shipping X11/AT, an X server that runs under MS-windows. The server converts an IBM-AT into an X terminal which can simultaneously run MS-DOS and Microsoft Windows applications.

IBM is rumored to offer a product; part #5709-029.

Hummingbird Communications (Canada 416-470-1203) produces the HCL-eXceed and HCL-eXceed Plus for EGA, VGA, and VGA+ controllers.

Information Network Solutions also offers a product called HCL-eXceed for the *86. The fax is 02-4122079 inside Australia, 612-4122079 from overseas.

AGE (619-565-7373) offers the XoftWare TIGA.

Bell Technologies (Fremont, CA: 415-659-9097)

Intelligent Decisions, Inc. (Sunnyvale, CA: 408-734-3730)

16) How can I get an X server on a Macintosh running MacOS?

eXodus from White Pine Software (603-886-9050) runs on any Mac with at least 1MB of memory and runs the X server within a standard Macintosh window. eXodus II uses the math co-processor and other features of high-end Macs. [info current as of 6/89]

Apple's MacX runs on MacPlus or newer machines with >= 2MB of memory and system software 6.0.4 or later. It is an "X11R3.5" server that includes support for an optional built-in ICCCM-compliant window manager, X11R4 fonts and colors, a built-in BDF font compiler, and built-in standard colormaps, and it supports the X11R4 notion "all visuals that make sense" for color displays. Available 1Q90. [courtesy Alan Mimms (alan@apple.com], 3/90]

17) Where can I obtain an X-based editor or word-processor?

You can ftp the latest version of emacs, including X11 support, from prep.ai.mit.edu [18.71.0.38]. The file you probably want is ~ftp/pub/gnu/emacs-18.55.tar.Z, or similarly-named files.

Epoch is a modified version of Gnu Emacs with additional facilities useful in an X environment. Epoch is available by anonymous ftp from cs.uiuc.edu (128.174.252.1), in the directory pub/epoch-files. There are two subdirectories: epoch contains the epoch source, and gwm contains the source to the programmable window manager GWM, with which epoch works well.

The Andrew system on the X11R4 tape has been described as one of the best word-processing packages available. It supports word processing with multi-media embedded objects: rasters, tables/spread sheets, drawings, style editor, application builder, embedded programming language, &c. [Fred Hansen (wjh+@ANDREW.CMU.EDU)]

In addition:

FrameMaker and FrameWriter are available as X-based binary products for several machines. Frame is at 800-843-7263 (CA: 408-433-3311).

InDepthEdit is available from Non Standard Logics (+33 (1) 43 36 77 50).

DECwrite is available from DEC for some DEC hardware and SunWrite is available from Sun.

IslandWrite will soon be available from Island Graphics (415-491-1000) for some HP & Apollo platforms.

Interleaf is currently available from Interleaf (800-241-7700, MA: 617-577-9800) on all Sun and DEC platforms; others are under development.

The Alis office-productivity tool from Applix (1-800-8APPLIX, MA: 508-870-0300) includes a multi-font WYSIWG document composer; for several systems.

18) Where can I obtain an X-based paint/draw program?

xpic is an object-oriented drawing program. It supports multiple font styles and sizes and variable line widths; there are no rotations or zooms. xpic is quite suitable as an interactive front-end to pic, though the xpic-format produced can be converted into PostScript. (The latest version is on the R4 contrib tape in clients/xpic.)

xfig 1.4.3 is an object-oriented drawing program supporting compound objects. The text-handling is limited. The xfig-format can be converted in PostScript or other formats. The latest is on the R4 contrib tape in clients/xfig; it is one of the several 'xfig' programs which several groups independently developed parallel versions of from the R3 xfig.

idraw 2.5 supports numerous fonts and various line styles and arbitrary rotations. It supports zoom and scroll and color draws and fills. On the R4 tape; see also interviews-request@interviews.stanford.edu.

[courtesy Jim Helman (jim@kaos.Stanford.EDU) 7/89; some comments added by XUG]

In addition:

dxpaint is a bitmap-oriented drawing program most like MacPaint; it's good for use by artists but bad for drawing figures or drafting. dxpaint is part of the Ultrix 3.x release.

19) Where can I obtain an X-based spreadsheet?

Vendor	Product	Phone
Access Technology	20/20	(508) 655-9191
Informix	WingZ	(800) 331-1763
Quality Software Products	Q-Calc/eXclaim	800-628-3999 (CA:213-410-0303)
Unipress	Q-Calc	(201) 985-8000
Uniplex	Uniplex	(214) 717-0068, (800) 356-8063
Digital	DECdecision	1-800-DIGITAL
BBN Software Products	BBN/Slate	617-873-3984 (Scott Richardson)

(BBN Software Products includes WordProcessing, Spreadsheet, Graphics, Image Processing, Foreign Language WordProcessing, Electronic Mail, and Elecronic Conferencing)

The Alis office-productivity tool from Applix (1-800-8APPLIX, MA: 508-870-0300) includes a spreadsheet.

20) Where can I get a PostScript previewer for X?

xps is available from almost everywhere that the X11 contributed source can be found. The version currently on expo is based on Crispin Goswell's PostScript interpreter with fixes and speedups by John Myers and Barry Shein and an X11 driver by Terry Weissman. There are known problems with fonts. The package is good for lowering the edit-print-edit cycle in experimenting with particular PostScript effects.

Ghostscript is distributed by the Free Software Foundation (617-876-3296) and includes a PostScript interpreter and a library of graphics primitives. The README for the current version, 1.3, points out that it doesn't take advantage of many of the facilities offered by X but that this is intended to change in the future. The software can probably be found on prep.ai.mit.edu. A 1.4beta may be found on uunet. [2/90]

In addition:

ScriptWorks is Harlequin's software package for previewing and printing PostScript(R) descriptions of text and graphics images; previewers for X are available. For information call +44-223-872522 or send email to scriptworks-request@uk.co.harlqn.

Digital's dxpsview runs on UWS 2.1 and 2.2.

21) Where can I get a troff previewer for X?

X11R4 has two previewers for device-independent troff: the supported client xditview, and the contributed-but-well-maintainted xtroff. An earlier version of xtroff also appeared on the R3 contributed source.

In addition:

Elan Computer Group (CA: 415-964-2200) produces eroff, a modified troff implementation, and Elan/Express, an X11 eroff previewer (misleadingly) labeled "WYSIWYG".

SoftQuad (416-963-8337; USA only 800-387-2777, mail@sq.uu.net or mail@sq.com) offers SoftQuad Publishing Software, including a substantially rewritten troff formatter, a better intermediate language with backwards compatibility, and an X11[R3,R4] previewer. (This is the package adopted by AT&T's own MIS department, and used in and re-sold by many parts of AT&T). [information from Ian Darwin, SoftQuad (ian@sq.com) 3/90]

Image Network (1-800-TOXROFF; CA: 415-967-0542) has the 'xroff' package, which includes a fine modified troff implementation and a set of X11-based page previewers. (This is the package OEM'ed by several hardware vendors.)

[mostly courtesy moraes@cs.toronto.edu (Mark Moraes)] [2/90]

22) How do I convert Mac/TIFF/GIF/Sun/PICT/Face/img/FAX/etc images to X?

The likeliest program is an incarnation of Jef Poskanzer's useful++ Portable Bitmap Toolkit, which includes a number of programs for converting among various image formats. It includes support for many types of bitmaps, gray-scale images, and full-color images. The latest version, PBMPLUS, was posted to the net about 11/22/89; it is also on the R4 tape under contrib/clients/pbmplus. The package has been independently updated to support XPM images. Useful for viewing some image-formats is Jim Frost's xloadimage, a version of which is in the R4 directory contrib/clients/xloadimage. [Both PBMPLUS and xloadimage are under active development; watch for updated versions.]

23) How do I use an alternate window manager with DEC's session manager?

DEC's session manager will start dxwm up by default. To override this, add to your .Xdefaults file something like this line, naming the full pathname:

```
sm.windowManagerName:    /usr/bin/X11/your_favorite_wm
```

24) How do I build X with gcc?

MIT is now using regularly the Free Software Foundation's GNU-CC to build the X distribution and uses gcc-built servers to test performance increases.

[These options are gathered from several descriptions of building X with gcc 1.34, 1.35, and 1.36]:

Use the options

-O -fstrength-reduce -fpcc-struct-return

"-traditional" may also be necessary if your version of gcc is sufficiently old.

Do not use -finline-functions, particularly on the R4 server.

Make sure to run "fixincludes" from the gcc distribution before doing anything.

HOWEVER, there is a bug in gcc 1.34 and 1.36 (but not in 1.35 or 1.37) which miscompiles things of the form (expr == 0 ? exp1 : exp2). The fix needed in X11R4 (and probably X11R3) is to change the definition of XtNewString in Intrinsic.h to:

```
#define XtNewString(str) x
((str) != NULL ? (strcpy(XtMalloc((unsigned)strlen(str) + 1), str)) : NULL)
```

A work-around is also in fix-2 to X11R4.

25) What are these funny problems compiling X11R3 on the Sun4?

```
cc -c -O -I. -I../../include -I../../../.X11 -I../mfb cfbbitblt.c
cc: Fatal error in iropt: Illegal instruction (core dumped)
```

Known problems with the Sun4 optimizer render the -O flag unusable on this file.

In addition, there is a problem in all of the procedures that return a parameter that was never referenced. Instead of returning the string, the compiler with optimization seems to be returning the last value computed. You can compile lib/Xt/TMparse.c without optimization; alternatively, you can replace the "return str" in various routines to use that parameter [courtesy of Jim Fulton, MIT X Consortium]:

```
#ifdef sparc
/*
 * The silly optimizer in SunOS 4.0.3 and below generates
 * bogus code that causes the value of the most recently
 * used variable to be returned instead of the value
 * passed in.
 */
static String silly_optimizer_kludge;
#define BROKEN_OPTIMIZER_HACK(val) silly_optimizer_kludge = (val)
#else
#define BROKEN_OPTIMIZER_HACK(val) val
#endif
```

and have routines end with

```
return BROKEN_OPTIMIZER_HACK(str);
```

Note also that the SPARCstation1 has a bug in its use of -misalign; a fix to cc should be obtained from Sun.

26) What are these funny problems installing X11R4 on the Sun running SunOS 4?

All of the executables that I try to run have the following results:

```
ld.so: libXmu.so.4: not found
```

or even:

```
ld.so: call to undefined procedure __GetHostname from 0xf776a96c
```

If you are building with shared libraries on a Sun, remember that you need to run "ldconfig" as root after installing the shared libraries (if you've installed X on a file-server, run it on the server's clients, too). While building and installing the distribution, you need to be careful to avoid linking against any existing X shared libraries you might have (e.g. those distributed with OpenWindows). You should make sure you do not have LD_LIBRARY_PATH set in your environment during the build or the installation. If you are going to keep xterm and xload as setuid programs, please note that the shared libraries must be installed in /usr/lib or /usr/5lib for these programs to work (or else those programs must be linked statically). [courtesy MIT X Consortium]

27) Where can I get a fast X server for a workstation?

The R4 server should be among the fastest available for most machines.

The "Purdue" speedups significantly speed up the X11R3 server. Look on expo.lcs.mit.edu:contrib/Purdue.2.[01]-tar.Z. (You'll also need gcc.)

International Quest Corporation (408-988-8289) has an optimized R3 server for Sun3/4/386i under SunOS 4.0 and also an optimized R4 server.

Torch Technology Ltd makes several R3-based tuned servers, most notably for Sun 3 and Sun 4. Call (UK) +44 223 841000 for information. (Note: Torch is about to go out of business; so far the fate of its products is unclear.)

28) How can I change the titlebar of my xterm window?

The solution involves sending an escape sequence to xterm which will cause it to update the property which the window manager relies upon for the string which appears in the window titlebar. A solution is as easy as typing this in an xterm running a shell:

```
echo "ESC]2;TEXT^G"
```

where ESC is the escape key, TEXT is the string you wish to have displayed, and ^G is a Control-G (the BEL character).

Here is a more complicated csh alias which changes the titlebar to the current working directory when you change directories:

```
alias newcd 'cd * ; echo ESC]2;$cwd^G'
```

The digit '2' in these strings indicates to xterm that it should change only the title of the window; to change both the title and the name used in the icon, use the digit '0' instead, and use '1' to change only the icon name.

29) Why doesn't anything appear when I run this simple program?

```
the_window = XCreateSimpleWindow(the_display,
        root_window,size_hints.x,size_hints.y,
        size_hints.width,size_hints.height,BORDER_WIDTH,
        BlackPixel(the_display,the_screen),
        WhitePixel(the_display,the_screen));
XSelectInput(the_display,the_window,ExposureMask|ButtonPressMask|
        ButtonReleaseMask);
XMapWindow(the_display,the_window);
XDrawLine(the_display,the_window,the_GC,5,5,100,100);
```

You are right to map the window before drawing into it. However, the window is not ready to be drawn into until it actually appears on the screen — until your application receives an Expose event. Drawing done before that will generally not appear. You'll see code like this in many programs; this code would appear after window was created and mapped:

```
while (!done) {
        XNextEvent(the_display,&the_event);
        switch (the_event.type) {
        case Expose:    /* On expose events, redraw */
            XDrawLine(the_display,the_window,the_GC,5,5,100,100);
            break;
        ...
        }
}
```

Note that there is a second problem: some X servers don't set up the default graphics context to have reasonable foreground/background colors, and your program should not assume that the server does, so this program could previously include this code to prevent the case of having the foreground and background colors the same:

```
the_GC_values.foreground=BlackPixel(the_display,the_screen);    /* e.g. */
the_GC_values.background=WhitePixel(the_display,the_screen);    /* e.g. */
the_GC = XCreateGC(the_display,the_window,
                GCForeground|GCBackground,&the_GC_values);
```

Note: the code uses BlackPixel and WhitePixel to avoid assuming that 1 is black and 0 is white or vice-versa. The relationship between pixels 0 and 1 and the colors black and white is implementation-dependent. They may be reversed, or they may not even correspond to black and white at all.

30) What is the difference between a Screen and a screen?

The 'Screen' is an Xlib structure which includes the information about one of the monitors or virtual monitors which a single X display supports. A server can support several independent screens. They are numbered unix:0.0, unix:0.1, unix:0.2, etc; the 'screen' or 'screen_number' is the second digit — the 0, 1, 2 which can be thought of as an index into the array of available Screens on this particular Display connection. The macros which you can use to obtain information about the particular Screen on which your application is running typically have two forms — one which takes a Screen and one with takes both the Display and the screen_number. In Xt-based programs, you typically use XtScreen(widget) to

determine the Screen on which your application is running, if it uses a single screen. (Part of the confusion may arise from the fact that some of the macros which return characteristics of the Screen have "Display" in the names — XDisplayWidth, XDisplayHeight, etc.)

31) Why do I get a BadDrawable error drawing to XtWindow(widget)?

I'm doing this in order to get a window into which I can do Xlib graphics within my Xt-based program:

```
canvas = XtCreateManagedWidget( ...,widgetClass,...) /* drawing area */
window = XtWindow(canvas); /* get the window associated with the widget */
XDrawLine (...,window,...); /* produces error */
```

The window associated with the widget is created as a part of the realization of the widget. Using a window id of NULL ("no window") could create the error that you describe. It is necessary to call XtRealizeWidget() before attempting to use the window associated with a widget.

32) Can I get the background pixel/pixmap of a window using XGetWindowAttributes?

No. Once set, the background pixel or pixmap of a window cannot be re-read by clients. The reason for this is that a client can create a pixmap, set it to be the background pixmap of a window, and then free the pixmap. The window keeps this background, but the pixmap itself is destroyed. If you're sure a window has a background pixel (not a pixmap), you can use XClearArea() to clear a region to the background color and then use XGetImage() to read back that pixel. However, this action alters the contents of the window, and it suffers from race conditions with exposures. [courtesy Dave Lemke of NCD and Stuart Marks of Sun]

Note that the same applies to the border pixel/pixmap. This is a (mis)feature of the protocol which allows the server is free to manipulate the pixel/pixmap however it wants. By not requiring the server to keep the original pixel or pixmap, some (potentially a lot of) space can be saved. [courtesy Jim Fulton, MIT X Consortium]

33) Why does the pixmap I copy to the screen show up as garbage?

The initial contents of pixmaps are undefined. This means that most servers will allocate the memory and leave around whatever happens to be there — which is usually garbage. You probably want to clear the pixmap first using XFillRectangle() with a function of GXcopy and a foreground pixel of whatever color you want as your background (or 0L if you are using the pixmap as a mask). [courtesy Dave Lemke of NCD and Stuart Marks of Sun]

34) Why doesn't my program get the keystrokes I select for?

The window manager controls how the input focus is transferred from one window to another. in order to get keystrokes, your program must ask the window manager for the input focus. To do this, you must set up what are called "hints" for the window manager. If your applications is Xlib-based, you can use something like the following:

```
XWMHints wmhints;
wmhints.flags = InputHint;
wmhints.input = True;
XSetWMHints(dpy, window, &hints)
```

If your application is based on the Xt Intrinsics, you can set the XtNinput resource to be True (as you probably want to in any case). [courtesy Dave Lemke of NCD and Stuart Marks of Sun]

35) Why does the R3 xterm, et al, fail against the R4 server?

The value given to a window's do_not_propagate mask is the likely culprit. R3 allowed bogus values to be set, and early version of both Andrew and Interviews did, as well. Similar problems also occurs in the R3 Xm PanedWindow widget.

If it is impossible to fix client source, use 'xset bc' to put the X11R4 server into bug-compatibility mode.

36) Where can I obtain alternate language bindings to Xlib?

Versions of the CLX Lisp bindings are part of the X11R3 and X11R4 core source distributions.

Ada bindings were written by Mark Nelson and Stephen Hyland at SAIC for the DOD. The bindings can be found on hapo.sei.cmu.edu or on wsmr-simtel20.army.mil and are also in the Ada Software Repository (ASR). R3 bindings should be available by the end of 1/90.

Prolog bindings (called "XWIP") written by Ted Kim at UCLA while supported in part by DARPA are available by anonymous FTP from expo.lcs.mit.edu:contrib/xwip.tar.Z or ftp.cs.ucla.edu:pub/xwip.tar.Z. These prolog language bindings depend on having a Quintus-type foreign function interface in your prolog. The developer has gotten it to work with Quintus and SICStus prolog. Inquiries should go to xwip@cs.ucla.edu. [3/90]

AUUG Institutional Members

ANL Limited
Adept Business Systems Pty Ltd
Alcatel STC Australia
Alcatel STC Australia
Aldetec Pty Ltd
Alliance Computer Centre Pty. Ltd
Apple Computer Australia
Apscore International Pty Ltd
Australian Electoral Commision
Australian Nuclear Science &;Technology Organisation
Australian Wool Corporation
BHP Melbourne Research Labs
Ballarat Base Hospital
Basser Department of Computer Science
Bond University Library Service
Bond University;School of Information and Computing Science
CADAD Support Section - SECWA
Capricorn Coal Management Pty. Ltd.
Civil Aviation Authority
Colonial Mutual
Commodore Business Machines Pty. Ltd.
Comperex (NSW) Pty Ltd
Computer Power IR+D, NSW Branch
Computer Power Today IR+D
Computer Software Packages
Computerscene International
Corinthian Engineering Pty Ltd
Cybergraphic Systems Pty Ltd
DBA Limited
DMR Group
Data General
Davey Products Pty Ltd
Dept of Agricultural + Rural Affairs
Dept of Industry, Technology and;Resources, Victoria
Digital Equipment Corporation; (Australia) Pty. Limited
Earth Resource Mapping Pty Ltd
Elxsi Australia Ltd
Epson Australia Pty Ltd
Flinders University;Discipline of Computer Science
Genasys II Pty Ltd
Golden Casket Art Union
Gould Electronics Pty Ltd
Harris & Sutherland Pty Ltd
Hewlett Packard Australia Limited
Honeywell Software Centre
IBM Australia Limited
ICL Australia Pty. Ltd.
IPS Radio and Space Services

AUUG Institutional Members

Ipec Transport Group
Labtam Information Systems Pty Ltd
Macquarie Bank Limited
Macquarie University
Main Roads Department, Queensland
Mincom Pty. Ltd.
Motorola Communications Australia
NEC Information Systems Australia Pty Ltd
NSW Parliament
National Engineering Information Services P/L
Nixdorf Computer Pty Limited
Olivetti Australia Pty Ltd
Olympic Amusements P/L
Overseas Telecommunications Corporation
Performance Systems
Prentice Computer Centre
Prime Computer of Australia Ltd
Pyramid Technology Corporation
Q. H. Tours Limited
Queensland Department of Mines
RMIT Computer Centre
Roads and Traffic Authority NSW
SEQEB
Sigma Data Corporation Pty Ltd
Sola International Holdings Ltd
South Australian Institute of Technology
Sphere Systems Pty Ltd
State Bank Victoria
State Bank of NSW
State Library of Tasmania
Sugar Research Institute
Sun Microsystems Australia
Swinburne Institute of Technology
Sybase Australia Pty Ltd
Tattersall Sweep Consultation
Telecom Network Engineering - C.S.S
The Australian Artificial Intelligence Institute
The Australian National University
The Mathematics and Computing Department - BCAE (Kelvin Grove Campus)
The University of Adelaide
The University of New South Wales
The University of Wollongong
University of New England
University of Technology Sydney;Computing Services Division
Vicomp
Wang Australia Pty Ltd
Wyse Technology Pty. Ltd.
Yartout Pty Ltd

EUROPEAN
UNIX® SYSTEMS USER GROUP
NEWSLETTER

EUUG

Volume 9, Number 4
Winter 1989

Editorial

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

Parliament Hill Computers Ltd

Protect Your Software

How do you do it? You can use clever serialisations or use a dongle, but you should also do this through legal means, i.e., copyright. It is not as difficult as it seems, and if done properly gives you a means of redress if you do discover a theft.

Alicia Gronke delivered an excellent paper on this topic at the EUUG Vienna conference in September. This is reprinted on page 2 – you now have no excuse for not copyrighting your software.

Conference Mail

Did you send or receive mail when you were at the Vienna conference? The story of the activity behind the scenes is brought to you on page 10.

Those guys deserve a big "thank you" from all of us who made use of the service, why don't you mail them and say so.

Unix Networking Group

The USING group has thrown its doors wide open and is inviting membership from Europeans who are interested in Networking.

Pat Eves introduces you to the group on page 20, and invites you to their conference in March 1990.

More Conferences

Other groups also announce their forthcoming conferences to you, the AUUG (Australia) in September 1990 on page 26, USENIX (USA) in January on page 37, UKUUG (UK) in December and February on page 33, the NLUUG (Netherlands) in May on page 30, and most important of all the EUUG conference in April on page 16.

There is a summary of UNIX related meetings in the Calendar on page 71.

Regular Columns

Janet Davis has left UEL and now we have Gill Mogg hosting their column. She has started out with a detailed look at the OPEN LOOK Graphical User Interface on page 61.

Group Reports

Did you know about the UNIX group in Czechoslovakia? On page 13 Peter Pronay introduces the latest UNIX user group and discusses some of the problems that they have, not least of which is the non availability of source.

You will also find a report from Iceland (page 28) as well ass the regular reports from the Netherlands (page 30), the UK (page 33), and from the USA (page 35).

Advertise in the EUUG Newsletter

You too can reach a pan European audience of UNIX professionals. Make your product or service widely known.

Page 23 will tell you all that you need to know.

Future EUUGN Dates

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.

Please mail or phone me to discuss an idea. Remember these important dates:

Copy Date	Publication Date
22 January	1 March
23 April	1 June
23 July	1 September
22 October	1 December

How to Protect Your Software...

Alicia Dunbar Gronke
epg@siegfried.sun.com

USENIX Association
Berkeley, California, USA

Alicia Dunbar Gronke is a native of San Francisco, California. Although formally trained and educated as a classical musician, she began exhibiting strange inclinations towards studying copyright law while working as a legal aide and research assistant for a small law firm about three years ago.

This is her first attempt at publically unleashing this, until now, "closet obsession".

How to Protect Your Software Through International Copyright Laws: Step-By-Step Instructions

This article will attempt to clarify some common misconceptions about copyright formalities as applied to intellectual properties. Due to the fact that these laws vary from country to country, the examples presented here should be viewed only as basic approaches to copyright protection, not as a legal reference.

Taking these few necessary precautions to protect your work from theft or unauthorised alteration isn't nearly as much of a task as one might believe. Here, we set about taking the mystery out of what should be only a few simple procedures. The idea being stressed here is taking as many preventive measures as possible to protect your work. The effort required to do this is minimal.

Introduction

Copyrights, by my definition, were designed with the idea that if a person is willing to share their works and expressions with the interested populace by distributing 'copies' of said works, that person has an exclusive 'right' to protection under the law from those ever present scavengers that watch for the drop of the proverbial ball, only to profit at that person's expense. That person is also entitled to the right to reap any rewards or benefits which may become evident upon publication. This type of law provides protection for the form an expression takes, not the ideas being expressed.

Step Number One: The Copyright Notice

Let's say that you've just put the finishing touches on your new C++ program. You're on top of the world and you feel like a new parent. That's fine, but what's to deter anyone from stealing your new offspring? A Notice of Copyright. The first step in the posting of a copyright notice entails the use of the copyright symbol, namely ©. This should be followed by the word Copyright, date of first publication and your name. This simple step fulfils the minimum formality requirements set by the UCC, the Universal Copyright Convention, as well as all signatory nations of the Berne Convention. Example:

© Copyright [DATE OF PUBLICATION] [YOUR NAME]

However, this may be your second or perhaps third revision of this work, in which case your statement will look something like this:

© Copyright 1985, 1989 [YOUR NAME]

In the case of multiple authors, the practice of showing the names of individual contributors (as well as the date of publication of their contribution) is often used:

© 1988 Modesia Fleming
© 1986 Elissa Metterhausen
© 1983 Daniel Karrenberg

These notices are not mandatory requirements of the UCC, however, the use of them makes the owner exempt from any other copyright formalities imposed by any other UCC nation. Although it is implied by the copyright notice, it would also be to your advantage to add the fact that this publication is not to be copied in any way, shape or form without the express written permission of the author. There is no such thing as too much insurance. To make it easier for the user to comply with your wishes, you may want to include the name and address of your publisher. The extra information volunteered in this instance will not invalidate the copyright. As long as the four major requirements are clear and properly placed, the notice is considered valid.

On works containing trade secrets, such as computer program source code, a notice indicating that the work contains non-public proprietary information should also be added.

As for [DATE OF PUBLICATION], the moment you give, sell, rent or lend the work, consider it 'published'. The exceptions to this rule are if the people using it are your employees or are part of a limited group of individuals of the understanding that further distribution of said work is unacceptable. In such instances, this could be viewed as a 'limited publication'. To keep a tighter rein, a friendly legend should be added, such as:

"This copy is for private circulation only and may not be used in any other manner."

Or you may choose a more formal approach:

"The material within is an unpublished copyrighted work containing trade secrets or otherwise sensitive information of [THE COMPANY]. This copy has been provided on the basis of strict confidentiality and on the express understanding that it may not be reproduced or revealed to any person, in whole or in part, without express written permission from [THE COMPANY], which is the sole owner of copyright and all other rights therein."

To deter any later arguments, your notice of copyright should be affixed in obvious places throughout the work making sure it is not concealed from view upon reasonable examination. For example, in source or object code, the notice should be placed at the beginning as well as the end of all printouts, not forgetting occasional insertions in the program itself on disk or tape. It is preferable for most users that this notice be placed solely at the very beginning or at the top of the menu. This is also acceptable. There are few things more irritating than having a copyright notice on constant display. And don't forget to affix this notice to all packaging as well, on the outside of boxes and on tapes. If a proper copyright notice is not placed on all 'publicly distributed' copies of a work, the work with which the copyright is associated may in time enter the 'public domain', as an improper copyright is an *invalid* copyright.

Allow me to stress the fact that under the Berne Convention (more on Berne later), the use of the copyright notice on published copies of a work is optional. However, the use of it is still encouraged in the U.S. As an incentive to do so, a new section was added to the U.S. Copyright Act. This section [401(d)] prevents claims of innocent infringement by defendants where notices have been placed on stolen works.

Step Number Two: Registering Your Work

This is the second most vital step toward protecting your work. Although this is no longer mandatory in most countries, it still carries considerable weight in courts of law throughout the world in cases of infringement. As each country has its own individual approach to intellectual property law, it is *extremely* important that you seek the advice of legal counsel before

making any final decisions. One advantage of this step is that your work, should it be distributed without notice of copyright (heaven forbid!), will still have some form of protection. Your registry notice is proof that you took precautionary measures to insure against this sort of thing. Another plus: bringing suit to enforce your copyright sans registration makes the job just that more difficult.

For the sake of argument, say a writer of a non UCC country, say Upper Volta, first publishes their work in Japan, a UCC nation. That work will be automatically protected as in any other UCC nation.

What is the UCC?

The UCC, otherwise known as the Universal Copyright Convention, first came into force 16 September 1955. The two other major copyright conventions are the Berne Convention and the Buenos Aires Convention. Of the three, the UCC is the most widely adopted treaty. It is the 'umbrella' providing protection known as the 'national treatment' doctrine. By employing this doctrine, each member country is obliged to grant the same protection to the works of other member countries as works first published within its own boundaries, as long as that protection meets the minimum requirements of the UCC. This rule applies to both published and unpublished works. So far, it sounds like the panacea for all those 'copyright protection blues'. Not true. The problem with this is the fact that there are still many countries not belonging to the UCC, and even for the ones who do, there are limitations to the protective requirements imposed beyond national treatment. The minimum standards set by the UCC aren't nearly as comprehensive as the alternatives.

Note:
Depending on the country, you may have trouble protecting object code, since some countries still view it as an adaptation or derivative of source code.

What is the Berne Convention?

The Berne Convention for the Protection of Literary and Artistic works is the oldest and most respected of international treaties. It was concluded at Berne, Switzerland in 1886 and is organised as a Union which is open to all countries of the world provided these minimum protective requirements are met:

- the granting of national treatment,

- the granting of certain 'moral rights/droit morale' to authors with regard to the exploitation of their works,

- the granting of certain 'economic rights', such as the exclusive rights of translation, reproduction, performance, adaptation, arrangement or alteration, regarding protected works, and

- the adoption of certain minimum terms of protection, generally the life of the author plus 50 years, for various works.

In essence, Berne provides a substantially higher level of protection. For example, copies accidentally distributed without copyright notice are given protection under Berne. This is not the case in most UCC countries. Should any conflict in protection arise, an author is assured to receive the most favourable protection offered under either treaty. Both conventions stipulate that any disputes between member States concerning the interpretation or application of the Convention that has not been settled via negotiation should be referred to the International Court of Justice.

Here is a quote from Dr Henry Olsson, Director of Copyright and Public Information Department, 21 November 1986 at an international conference on copyright in s'Gravenhage, The Netherlands, speaking on the Berne Conference:

"The Assembly of the Berne Union in September of 1986 declared *inter alia* that copyright is based on human rights and that authors, as creators of beauty, entertainment and learning, deserve that their rights in their creations be recognised and effectively protected both in their own country and in all other countries of the world."

What is the Buenos Aires Convention?

This treaty provides copyright protection for sixteen South and Central American countries as well as the United States. According to Article 3 of the Buenos Aires Convention, no formalities have to be observed in any country other than the country of origin, *provided* a statement appears in the work that indicates the reservation of the property right. The phrase 'All Rights Reserved' or its equivalent (e.g., 'Derechos Reservados') are commonly used for this purpose. The three nations of this convention choosing not to sign the UCC treaty are Uruguay, Honduras and Bolivia.

To gain equal protection in these countries, the formalities of the Buenos Aires Convention require your notice to look something like this:

© 1989 Jaap Akkerhuis All Rights Reserved

Between contracting states which are also signatories of the UCC, the notice provision, as well as any other conflicting provisions, are superseded by those of the UCC. When distributing software in Latin American countries, take into account the technology transfer legislations which can limit the years during which protection can be provided in contracts. An excellent reference on this subject is *Trade Secrets and Know-how Throughout the World, (1981)* by Aaron N. Wise.

The following lists contain the names of independent nations which have signed the UCC treaty, the Berne Convention, the Buenos Aires Convention, as well as those without any major treaties. These Conventions are voluntary agreements made by the governments of these signatory countries. (As only three of the sixteen member countries of the Buenos Aires Convention are UCC signatories, time will be spent rather on the two larger Conventions.)

Countries signing the Universal Copyright Convention:
Algeria, Andorra, Argentina, Australia, Austria, The Bahamas, Bangladesh, Belgium, Brazil, Bulgaria, Cameroon, Canada, Chile, Colombia, Costa Rica, Cuba, Czechoslovakia, Democratic Kampuchea, Denmark, Ecuador, El Salvador, Fiji, Finland, France, German Democratic Republic, Federal Republic of Germany, Ghana, Greece, Guatemala, Haiti, Holy See, Hungary, Iceland, India, Ireland, Israel, Italy, Japan, Kenya, Laos, Lebanon, Liberia, Liechtenstein, Luxembourg, Madagascar, Mali, Malta, Mauritania, Mexico, Monaco, Morocco, The Netherlands, New Zealand, Norway, Pakistan, Peru, Poland, Portugal, Romania, Senegal, Spain, Sweden, Switzerland, Trinidad & Tobago, Tunisia, United Kingdom, United States, Venezuela, and Yugoslavia.

Countries signing the Berne Convention:
(as of November 1, 1988)

Argentina, Australia, Austria, Bahamas, Barbados, Belgium, Benin, Brazil, Bulgaria, Burkina Faso, Cameroon, Canada, Central African Republic, Chad, Chile, Philippines, Colombia, Congo, Costa Rica, Côte d'Ivoire, Cyprus, Czechoslovakia, Denmark, Egypt, Fiji, Finland, France, Gabon, German Democratic Republic, Federal Republic of Germany, Greece, Guinea, Holy See, Hungary, Iceland, India, Ireland, Israel, Italy, Japan, Lebanon, Libya, Liechtenstein, Luxembourg, Madagascar, Mali, Malta, Mauritania, Mexico, Monaco, Morocco, The Netherlands, New Zealand, Niger, Norway, Pakistan, Peru, Poland, Portugal, Romania, Rwanda, Senegal, South Africa, Spain, Sri Lanka, Suriname, Sweden, Switzerland, Thailand, Togo, Trinidad and Tobago, Tunisia, Turkey, United Kingdom, United States, (as of 1 March 1989), Uruguay, Venezuela, Yugoslavia, Zaire, and Zimbabwe.

Countries signing the Buenos Aires Convention:
Argentina, Bolivia, Brazil, Chile, Columbia, Costa Rica, Dominican Republic, Ecuador, Guatemala, Haiti, Honduras, Nicaragua, Panama, Paraguay, Peru, United States, and Uruguay.

Countries Without Conventions:
Afghanistan, Albania, Angola, Antigua, Barbuda, Bahrain, Belau, Belize, Bhutan, Botswana, Burundi, Cape Verde, Comoros, Djibouti, Dominica, Equatorial Guinea, Ethiopia, Gambia, Grenada, Guinea-Bissau, Guyana, Indonesia, Iran, Iraq, Jamaica, Jordan, Kiribati, Korea, Kuwait, Lesotho, Malaysia, Maidives, Myanmmar, Mongolia, Mozambique, Nauru, Nepal, Oman, Papua New Guinea, Phillippines, Qatar, Saint Lucia, Saint Vincent and the Grenadines, San Marino, Sao Tome and Principe, Saudi Arabia, Seychelles, Sierra Leone, Singapore, Soloman Islands, Somalia, Sudan, Swaziland, Syria, Tanzania, Tuvalu, Uganda, Upper Volta, Vanuatu, Western Samoa, Yemen (Aden) and Yemen (San'a).

Protection is automatic anywhere as long as you are a citizen of a UCC or Berne country and the work is unpublished. However, once the work is published, you must adhere to the rules imposed by the member country with which the work is registered.

Here are a few prime examples of how various European countries, the U.S. and Japan apply their own rules and formalities regarding the copyrighting of computer software.

Austria:

No provision has been made with respect to the protection of software in Austrian law. A court in Wien confirmed the position that under certain conditions, software is protected under the Austrian Copyright Act. *[Oberlandesgerichts Wien, 8 August 1985, GRUR Int. 793 1987]*

Belgium:

Like Austria, no special provision has been allotted in Belgian copyright law for the protection of software. A bill was introduced in Parliament in June of 1988 for the enactment of a new Copyright Act. The proposal provides that software is protected by copyright with a term of protection lasting 25 years. This Act is not expected to become Law until 1990.

There are currently two laws in Belgium for the protection of copyrightable works. Only programs that are original and express the personal creativity of their authors are eligible for this protection. Belgian law does not protect ideas or opinions, only the form in which they are expressed. Protection begins with creation. The term of copyright spans the life of the author plus a period ending 50 years after January 1st of the following year of death.

France:

Seeing as France has no formal copyright office, paperwork here is not a problem, even though software is indeed copyrightable. Acceptable evidence of copyright date, in some cases, can be proven by showing the work was registered in the U.S. A new law introducing the principle that software is an intellectual property protectable by copyright law came into effect January 1st, 1986. Article 1 of this law expressly confers copyright protection on software. The term is 25 years from the date of the program's creation. France requires the program to be an original work, marked by a personal and intellectual contribution of the author.

West Germany:

Like the rest of Europe, filing in West Germany is unnecessary. They have what is called the Copyright Revision Act, which includes "programs for the processing of data", ("Programme für die Datenverarbeitung"),

Section 2(1)(1) of the German Copyright Act. The term of protection lasts for the lifetime of the author plus 70 years. A computer program is copyrightable only if:

- it is not just a simple program, i.e., the answer to the problem being solved is not obvious,

- while developing the program, various solutions could freely determine variables,

- the program is not restricted to mechanical-technical continuation and development of generally known subject matter, and

- one can perceive an important, creative and original ability of selection, assembly, reviewing, arranging and classification of information and instructions which surpasses the general average ability.

In the eyes of the German court, only an individual can own a copyright, not a company, which temporarily dissolves the idea of 'work-for-hire'. The common way around this is for companies to request the worker to execute a license agreement, giving the company exclusive user and marketing rights to the program.

Japan:

According to the 1982 decision of the Tokyo District Court, only source code is copyrightable and object code is merely a copy which is also protected under the Japanese Copyright Act. The period of protection is for the life of the author plus 50 years. Copyright is affirmed at the moment of creation. As with other signatory countries, registration is not a prerequisite for filing a lawsuit, although it makes proving infringement a much less painful ordeal. It is presumed that the date of creation corresponds with the date of registration. Your best move, should you be creating something in Japan, is to register with the Commissioner of the Agency for Cultural Affairs in order to establish the name of the copyright owner and the date of publication.

Their work-for-hire applications are quite similar to that of the U.S. In cases of infringement, the damages collected are equal to the profit made by the infringer. On the other hand, damages could equal the possible amount accrued had the author sold the program.

The Netherlands:

There are really no formal requirements for software copyright protection and only the outward form of the work is protected. Computer

programs in Holland are protected under Dutch law only after meeting the general requirements of originality and perceptibility. Original in the sense that the work be the result of creative activity, perceptible to the point of the work being perceptible to the senses. Protection is recognised at creation and terminates 50 years from the 1st of January of the year following the death of the author.

Sweden:
Computer programs in Sweden are protected under the Swedish Copyright Act which states: "Catalogues, tables and similar compilations, in which a large number of particulars have been summarised, may not be reproduced without the consent of the producer..." Future legislation will expressly provide for software protection under copyright law.

United Kingdom:
Section 1(1) of the Copyright Amendment Act confirms that programs are to be treated in the same way as literary works under the 1956 Copyright Act. An Act which states that a copyright work first comes into existence when it is reduced to writing or some other material form. Section 2 of the same Act states that computer programs comply with the material form requirement, even if the programs are not printed or written on paper but simply stored on a computer.

Once again, registration here is not required. So please keep careful records of program creations and changes by date. It is still unclear as to whether or not object code is seen by the United Kingdom as copyrightable. Here the author of the program is the owner of the copyright unless it was composed under a work-for-hire situation, in which case the employer would be the owner. Under a new Act passed in November of 1988, (the Copyright, Designs and Patents Act), the electronic copying of a program constitutes infringement.

United States:
All it takes to obtain a copyright registration is filing a copy of the program with the Copyright Office, along with a two sided form and a $10.00 filing fee. There is only a cursory examination to see that the form is filled out completely, the money paid and the deposit made. Once issued, the copyright is assumed valid and the term of protection is essentially 75 years from the date of publication, or 100 years from the date of

creation, or the life of the author plus 50 years, whichever is shorter. For more detailed information, contact:

> Register of Copyrights
> Copyright Office
> Library of Congress
> Washington, D.C. 20559
> U.S.A.

Registration of a work with the Copyright Office used to be required as a prerequisite to a suit for infringement. Section 411 of the Copyright Act has been changed to a two-part system where American authors are required to register while authors of other countries are not.

Judging by Congress' litigation 'track record', they still favour the registration system. Works accompanied by a registration certificate continue to be given prima facie standing when the question of copyright validity is in infringement litigation.

Registration provides the copyright owner with a broader range of infringement remedies. For these reasons, registration within 3 months of first publication is generally advisable. (Known as 'timely registration'.) You'll be needing Form TX for software. Take note that the U.S. Copyright Office now accepts copies of object code under the 'rule of doubt' theory. So, in granting registration, no opinion is expressed as to whether the code actually embodies the registered work. The alternatives listed are available should you feel uncomfortable submitting only object code:

- Submit only the first and last ten pages of source code,

- Submit all of the source code (referring to programs consisting of 25 pages or less) with up to half of it blocked out as necessary to protect trade secrets,

- Submit only the first and last 25 pages of source code with up to half of the code blocked out, making sure to block out just enough code in order to protect trade secrets, or

- Submit the first and last 25 pages of object code with an additional 10 consecutive pages of source code.

What are the differences between Copyright, Trade Secret and Patent Law?

Trade Secret is information. It is a way to 'protect something of economic value by keeping it a secret'. This something generally provides some sort of an edge, otherwise known as a 'competitive advantage'. Taking careful measures to keep source code under wraps and away from prying eyes and sticky fingers is of utmost importance. If by chance, in spite of all your precautions, your work still falls into the wrong hands, you will have a very strong case against the infringer if you can prove your diligence and persistence in defending your work. Here is a list of steps which may prove helpful:

• Store working copies of source code in a safe, a locking filing cabinet (one that is difficult to carry away), or a safe deposit box. Remember to keep backup copies in separate places.

• When you're finished working with the code or are stepping away from it for a break, lock it up.

• Stamp ALL copies of the source code with a rubber stamp (preferably in red ink) reading "CONFIDENTIAL". This includes diskettes and their jackets.

• Include a notice, along the same lines of the formal notice shown on page 2, at the top of the menu. You of course can use your own discretion as to how many times you'd like to list this throughout the work.

• When showing the source code, have that party sign a Trade Secret Nondisclosure Agreement form *before* viewing it.

There is a veritable plethora of precautionary measures one can take regarding Trade Secrets. I've named but a few. Most countries appear to allow trade secret information to be transferred via license agreement or contract. There is a possibility that in your particular country the transfer of trade secrets may result in tax consequences. Consult with local counsel.

Patent deals with the right of protection to an inventor for the sole monopolisation of use and commercial exploitation of an 'invention'. So far, patent is the most costly particularly in terms of the application process, and it takes up to three years to obtain. Now that we've covered the less attractive aspects, here are the positive points. Should your software qualify for a patent, it cannot be used by anyone without your permission. And that holds true for 17 years. Of course there's no telling how long it will take before the software becomes obsolete.

Note:
There is an international agreement that covers patents by the name of the Patent Cooperation Treaty (PCT) which establishes procedures for obtaining uniform patent protection. Their aim is to guarantee national treatment internationally, to enhance the creation of national protective systems and to avoid a far-reaching legal disintegration in this somewhat limited field outside the Berne and UCC conventions. For more information, a booklet called the PCT Applicant's Guide can be obtained from the World Intellectual Property Organisation.

   WIPO
   34, chemin des Colombettes
   CH-1211 Geneva 20   SWITZERLAND

Piracy: How to Fight it From the U.S. Vantage Point

An acquaintance brings up in casual conversation, this great new C++ program she just bought for a mere fraction of what you're charging for yours. Upon careful investigation, you find yourself face to face with a slightly modified version of YOUR program. Being a victim of piracy is bad enough, but how can you fight the pirate if he's living in Switzerland? (Piracy seems to have paid off handsomely in his case.) Besides having a screaming fit and tearing your hair out by the roots, what do you do? You've done everything your lawyer told you to do in the first place. You slathered warnings and copyright notices on the box, on inside labels, in the source code. In the words of Douglas Adams, "Don't Panic".

First, call a *good*, (please note the emphasis on the word *good*) lawyer. Not all who advertise themselves as 'copyright lawyers' know what they're talking about. Try to get a referral from someone whose judgement you trust. Be sure you have all the necessary documentation on hand, i.e., notice of registration or other proof of creation, such as sealed self-addressed envelopes with dated cancellations. Once you've found your attorney, she will clarify the procedure for you and make sure all is in order before moving on to the next step.

This next step entails making sure that your copyright registration is recorded with the U.S.

Customs Service, [19 CFR Part 133 Subpart (D)]. This should slow down any imports bearing a suspicious likeness to your work until your case makes it into court and the court can issue an order preventing any further importation.

Now you're ready to move into the final stretch. You must seek an order from the International Trade Commission to bar the counterfeit work from entering the country on the grounds that its importation would prove to be an unfair act or would constitute an unfair method of competition [19 USC. Section 1337]. A similar rule applies to trademarks as well. When filing this complaint, you must be armed with the following information:

— Is this product produced in another country?

— Does it threaten to destroy an existing industry in the U.S.?

— Is the threatened industry efficiently and economically operated?

After the answers to these questions are reviewed and the formal complaint is filed, the case will be heard by an Administrative Law Judge, empowered to grant immediate relief and to ban the imposter work in whole or in part.

Conclusion

Last and most importantly, I must stress the importance of seeking legal counsel on a country-by-country basis to protect your intellectual property. Granted, there is no worldwide panacea for the piracy menace as yet. Every effort has been made to assure all the facts stated in this paper are current and accurate, giving a brief overview of what to expect when preparing a notice of copyright. As needs change, so will the laws, and no one is better qualified to keep you up to date than a lawyer who specialises in the area of Intellectual Properties.

Acknowledgements

The author would like to express her deepest appreciation to all those who have been so giving of their time, patience, expertise and support. Edward Gronke, husband, devil's advocate and friend; Dan Appelman, whose insight proved to be most invaluable, and Michael Kerekes, both friends and lawyers extraordinaire; Peter Salus and Shelly Anderson, true inspirations when it comes to dealing with the human animal; and lastly, the Data Communications group of Sun

Microsystems.

Bibliography

[1] Neil Boorstyn.
     Copyright Law.
     The Lawyers Co-operative, 1981

[2] Brad Bunnin & Peter Beren.
     Author Law & Strategies.
     Nolo Press, 1983

[3] R.M. Gadbaw & T.J. Richards.
     Intellectual Property Rights.
     Westview Press, 1988

[4] R. Lee Hagelshaw.
     The Computer User's Legal Guide.
     Chilton Book Company, 1985

[5] Jozef A. Keustermans & Ingrid M. Arckens.
     International Computer Law
     Mathew Bender, 1989

[6] Melville B. Nimmer.
     Nimmer on Copyright.
     Mathew Bender, 1988

[7] Daniel Remer.
     Legal Care for Your Software.
     Nolo Press, 1984

[8] Robert H. Rines, et al.
     Computer Software: A New Proposal.
     IDEA: The Journal of Law & Technology,
     Vol.29
     Franklin Pierce Law Center, 1988

[9] M.J. Salone, Stephen Elias.
     How to Copyright Software.
     Nolo Press, 1988

[10] William S. Strong.
     The Copyright Book.
     MIT Press, 1986

The EUUG Conference Mailing System

Torsten Beyer

tb@unido.uucp

tb@marvin.Informatik.Uni-Dortmund.de

Computer Science Department
University of Dortmund
West Germany

I'm currently working at University of Dortmund's Computer Science Department as part of the System Administration group. Right now I'm working on a User Administration System. When I'm not fooling around with UNIX computers I like flying with sailplanes and drinking Guinness (although not at the same time :-). Furthermore I love to go to EUUG conferences as I love to meet these funny UNIX people.

Michael Kuschke

mk@unido.uucp

mk@marvin.Informatik.Uni-Dortmund.de

Computer Science Department
University of Dortmund
West Germany

I'm studying computer science at the University of Dortmund (Germany). To afford this I work at the university in system administration, mainly Sun's, and networking. A few may know me from events like the EUUG conferences. I like drinking beer, meeting people all over the world and having fun. I hope I'll finish my studies in a finite time :-)

Those of you who attended the last EUUG conference in Vienna will have seen a new service on offer: the ability to send and receive electronic mail at the conference. This article is about the project, how it worked, how it was received and what we are planning for the next conference.

Software

For the EUUG Mailing System we had to alter the normal way of accessing electronic mail to a certain extent. Usually everybody has their own account from which s/he can read and write letters. As we didn't know the exact number of attendees in advance we decided not to provide personal accounts but instead to have one general mail-account that could be used by anyone to send mail. This of course produced one major problem: if there are no personal accounts available, how is it possible to actually send mail to a conference attendee? We solved this problem by modifying the UNIX mailing software (*sendmail* and friends) to accept letters with any address. Thus people without a personal account on the mail machine could still receive mail. To make mail addresses as unique as possible we proposed that mail to a conference attendee be sent to *Firstname_Lastname_Organization@euug-conf.EU.net*. All incoming mail that could not be delivered locally (i.e., mail to us :-) was spooled in a special directory. This directory was scanned by a daemon that printed mail out onto a laser-printer. This printed mail was then put in a public place where people could look for their mail. To maintain as much privacy as possible, all mail was folded in a way that showed the address but not the contents.

To send mail a means had to be provided to let people insert their real 'home-address' in the e-mail. Usually mail agents construct the sender's address from its login name. As all people had the same login name, this had to be changed. Instead of identifying the sender by the general account the actual home mail-address of the sender had to be put into the mail header. We used *comp* from the *mh* mail package to get this job done. It is possible to configure *comp* in a way that it asks for the senders address instead of constructing its own. Furthermore a simple front-end to *comp* was built that allowed users to choose their favourite editor to write the mail.

All this worked pretty well, except for the very first mail that arrived (Sorry Dan). When the printing software was set up the laser-writer still had some problems. The printing software printed the mail—at least it thought so—and deleted it afterwards. Unfortunately the laser-writer wasn't yet working ...

As an additional service net-news was offered. Initially this didn't work too well. Again a general account was installed to be used by people to read news. Due to some faults we had done in preparing and tailoring *rn* some problems occurred. We forgot that *rn* gets pretty upset about more than one *rn* process running from the same user-id. So for the first few days it was not possible for more than one attendee to read news at one time. This led to some pretty disappointing experiences for some of the delegates. In the middle of reading articles their *rn* process was killed by another delegate who had been told by *rn* to do so... This problem however was soon fixed and for the rest of the week the news service ran smoothly as well.

Hardware

All this ran on a Sun 3/260 that was kindly provided by Sun Germany together with Bacher GMBH Vienna. The machine itself ran well and our software was installed easily. The peripherals, however, were initially troublesome. It took quite a while to get the terminal server box running—this connected the terminals via Ethernet to the machine. Basically this was due to two problems: the weird way that the box had to be configured and the quality of the terminals. These were kindly provided by Technical University of Vienna (thanks to Fritz Plank), where they had been lying in some damp cellar for quite a while. The effect of this was most of the keyboards refused to work either at all or just partly. It took us quite a while—and Fritz quite a bit of diesel as he had to drive between the conference place and Technical University to get new terminals—so that at last four terminals were available.

The biggest problem however was introduced by the modem. We had a leased line across the city to our mail- and news-feed (again the Technical University). When we set things up on Saturday afternoon all worked pretty well. On Monday however nothing worked at all. We wanted 4800 Baud but didn't even manage to get a reliable connection at 1200 Baud. The problem seemed to be line quality. On Saturday afternoon when we set things up, the line had been OK. On Monday much more happening on the phone system which gave more line noise. It was Daniel Karrenberg who finally solved the problem by magically tuning the modem. First we connected an ordinary phone speaker to the line and listened to it. All sounded pretty quiet. So Daniel opened the modem and started meditating over the circuits. Finally he pointed to some arbitrary selected

jumpers and said: "Let's play around with these jumpers and see what happens. Could be that these control the amplification rate". We were pretty sceptical especially as the modem was only rented. Anyway, he tried and it worked. From this time we had no more line problems (but perhaps the lines next to ours started to get problems :-). Thanks to Daniel.

Acceptance

The whole project was greatly appreciated by most attendees (see tables 1 and 2). During the five days that the system was running, about 1000 mails were sent and about another 1000 were received. Using an estimate of about 300 delegates: everyone sent and received about three mails. Not bad for the first time. It was fun watching delegates discovering the system. Monday morning the tutorial attendees started using the system. Probably most of their mails were like: "Hi, I can be reached at the conference. Try answering to...". By the end of Monday we received the first bunch of responses. The same happened on Wednesday when the conference itself started. Lots of people were hanging around in front of the terminals, probably sending their first hello's to their friends. The first answers arrived on Wednesday afternoon, but the main flood arrived on Thursday. Apart from these "Hello" sort of mails there was some obviously pretty serious correspondence going on as well.

Table 1: Total number of logins: 905

Account	Mon	Tue	Wed	Thu	Fri	Total
netnews	23	51	90	93	50	316
mail	75	92	167	92	30	473
news&mail	98	143	257	185	80	789
other						116

As they told us the Swedish postmaster team solved some major problems at home via mail. So all in all the new service was widely accepted.

Our experience in organising and maintaining the project together with feedback we got at the conference however suggest some ideas for future improvement.

Future

First of all we need more terminals. The 'terminal-pool' was pretty crowded—especially during coffee breaks. At the next conference we'll provide significantly more terminals. Although the mail-printing software worked well, having personal accounts for each attendee would be desirable. Currently we are investigating this. Probably we will come up with some software to make attendees create their own accounts. And of course InterEUnet access would be very desirable. We are investigating this as well. However this is probably the most complicated part of our plans. The more people who get involved in this project the more complicated the organisation will get. Providing InterEUnet access however means having to coordinate lots of different people and organisations.

Anyway, if anyone wants to make suggestions to improve the Conference Mailing System, feel free to contact us at the net addresses above.

Table 2: Number of sent and received mail

Mails received	
Total	1559
Non system mail	941
Mails sent	
Total	1771
Non system mail	1150

UNIX in Czechoslovakia

Peter Pronay

peter@iaccs.uucp

Institute of Applied Cybernetics
Bratislava
Czechoslovakia

Peter Pronay is a researcher in the Institute of Applied Cybernetics in Bratislava, Czechoslovakia. He was a systems programmer, responsible for LAN and WAN networking in the International Institute of Applied Systems Analysis (IIASA)—a UNIX stronghold—in Laxenburg, Austria from 1981 to 1986. He has 8 years of UNIX experience. His main fields of interest comprise LAN and WAN networking and UNIX system administration. He also does part-time consulting to the users of SINIX(UNIX from SIEMENS).

Surprised? How could it possibly get there?

Well, I can't tell you the exact and complete answer either. One thing is for sure: country borders and even bloc borders do not seem to be very tight when it comes to software in general and especially, when it comes to embargoed SW. But jokes aside. UNIX of different flavours has been legally available in Czechoslovakia for a couple of years. Only as binaries—as you may have guessed. Different manufacturers, like SIEMENS, NCR, HP, Olivetti and others offer and deliver their company versions of UNIX together with their hardware. Of course, the number of installations of such machines is quite low.

The reasons for it are, first of all, the limited amount of hard currency the country and the companies can spend on western technology. But that is not the only reason. The influence of IBM and the like, on mainframes, affects the way an average computing department head thinks. This is as true here just as in other countries.

Another—at least partly justified—reason for the delayed appearance of UNIX is the lower reliability of domestic hardware. For coping with it, the computer professionals preferred other operating systems with error logging features and better HW-fault tolerance. Recently, with the improving reliability of the Czechoslovak computers the somewhat discouraging ignorance of UNIX towards HW problems is losing its importance. However, to our knowledge only a negligible portion of the installed base of the VAX-compatible machines, produced in Czechoslovakia for several years, run UNIX.

Nevertheless, UNIX is becoming strong in Czechoslovakia. The main vehicle at the moment is multiuser PC-based systems. With the recent shift of the embargo limitations, allowing 32-bit PCs with up to 20 MHz clock rates to be imported, the HW base for this category of UNIX-ing or XENIX-ing becomes more reasonable. And here the conditions for quick growth of the interest in UNIX/XENIX are certainly better than in the West. Although the VAX-compatibles offer higher processing rates, their higher price makes in many cases the 80386s the preferred choice.

Since 80286-based ATs still dominate the scene, most potential users of UNIX/XENIX are discouraged by the lack of application software. With 80386 based systems and their ability to run

DOS applications under UNIX, these hesitations will fade away.

One direction of software development remains completely closed for the Czechoslovak UNIX community. It is the graphics user interface X-Windows or alternative solutions, which all require powerful graphic stations, all of which are banned by the COCOM.

The high flexibility of UNIX in communications with minimum HW requirements (uucp) is another strong argument for luring people away from other systems. The relatively good telephone network in Czechoslovakia would allow certain progress here. However, on the imported UNIX based systems, this kind of communication is still banned by the COCOM regulations (would you believe this kind of insanity?). Certainly, this is another obstacle in moving towards open communication between the computer user community in Czechoslovakia and in the western world.

My institute—the Institute for Applied Cybernetics in Bratislava*—has been using UNIX and XENIX for several years and has contributed to the growing popularity of these systems in Czechoslovakia. I was lucky enough to be able to meet with the members of the EUUG board in Vienna during the last EUUG conference. Cautiously, I asked Teus Hagen and others, how would they look at an attempt to extend the EUUG's reign to Czechoslovakia. Of course I had in mind the access to EUNET, too. To my big surprise the answers were positive. I immediately acted so as not to loose any time at all. I applied in the name of our institute for direct academic membership of the EUUG and right after coming back from Vienna (what a long journey—all of 60 km!) I began to undertake steps toward founding the National UNIX Systems User Group in Czechoslovakia.

Once this is done, the next step should be the establishment of contacts to the EUUG on national UUG level. My institute might become a suitable backbone network site in the future—it is the principal developer of the networking products for the Czechoslovak PTTs, running its own

---

* Bratislava is the capital of Slovakia—one of the two republics in Czechoslovakia

experimental network, covering several towns in Czechoslovakia. This is especially important, since the country doesn't have its public data network yet.

But this is all easier to say than to do. At the forthcoming meeting of the UNIX users there will be one major point that needs clarification. It is the relationship of the UNIX copyright owners toward Czechoslovak users. It would be foolish to pretend, that all instances of UNIX in Czechoslovakia (and in other countries—not only of the socialist bloc) have been legally acquired. The sooner we—and the West—accept this fact, and the sooner that we do something positive to bring the situation in order, the better for both sides. The embargo is certainly not doing any good to anybody (or is it?). Certainly, countries or blocs will try to maintain exclusive ownership of high-tech technologies, closely related to military applications. But trying to exclude a number of nations from a whole large section of technological 'culture' for decades is hardly justifiable. Unavoidably, it leads to mutual mistrust, speculation and 'back-door' approaches.

Shortly after I obtained the membership certificate from the EUUG, I established contacts with UNIX Europe Ltd., in order to get AT&T's clear statement about the company's policy regarding licensing UNIX to Czechoslovakia. If there is to be no way of getting source licenses, the activities of the UNIX community in Czechoslovakia will certainly remain limited, and the interest in integrating the national UNIX community to the EUUG will be smaller. On the other hand, we realise, that our group will have to act to correct the attitude of so many in our country on the idea of software copyright. There is a lot happening in this area in our country—as in the sphere of law.

I personally have no illusions about a smooth solution of these intricate problems. But I would like to hope of good will on both sides. I also expect a hand from the EUUG in establishing the contacts necessary for constructive negotiations. Likewise, our institute is ready to act as a mediator in contacts to the official places in Czechoslovakia. Should the contacts fail, fertile ground for illegal distribution and use of UNIX will be further provided and this is something that we would like to avoid.

When Alain Williams asked me to write this information about UNIX in the CSSR, he stressed that I should report about ongoing research or

software projects related to UNIX. Well, in our situation and in spite of the above, this is particularly difficult. I could tell you something about what my institute is working on—perhaps I'll do so in a future report. But before we establish the CSUUG, I will not be able to give you a good picture about the whole scene. Those who buy western minicomputers with UNIX are well placed commercial companies, that are not especially fond of developing interesting software. The majority of these purchases goes into industry, some part goes to trading companies as well.

Those, who are strong in SW development (research institutes and universities) do not have enough hard currency to buy better HW, and if they do, they still do not succeed in purchasing UNIX sources. So they do what they can, but prefer not to talk too much about it. The projects I know about cover a broad spectrum—from AI,

through multiprocessor based systems and networking to distributed database systems. There are also some pretty strong individuals and teams in the different branches of the theory of computer science. All these people would love to get UNIX sources—for the reasonable prices normally offered to the academic community in the West. I am an optimist. I believe that the barriers will be loosened and that the software folks in Czechoslovakia will gain an equal opportunity to show how good they really are! Even now the many western companies that employ Czechoslovak programmers know about it.

Let me finish with a plea: as with other standards, UNIX is a strong means to bring like-minded people together. The computer users and programmers in Czechoslovakia certainly do not want to stand aside—they would like to make their contribution.

Position Offered

The UKUUG wishes to produce a UK Newsletter and seeks an editor to collect material, produce camera ready artwork, and coordinate printing and distribution.

It is envisaged that there will initially be 6 newsletters a year, the style and contents are to be agreed with potential editors.

Candidates should:

- Live in the United Kingdom.
- Have reasonable net connectivity.
- Have available equipment and experience to allow them to typeset a newsletter.

The UKUUG will pay all costs and a fee (to be agreed) upon successful production of each newsletter. Help will be available in the form of advice, contact with a printer, and generally setting up.

For more information please contact:

Sunil K Das,
Computer Science Dept,
The City University,
Northampton Square,
London EC1V 0HB,
England
Tel: +44 1 253 4399 X 3725
Fax: +44 1 250 0837
Net: sunil@cs.city.ac.uk

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 17.30 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.

These new developments also require better tools than the known ones like new source code control systems.

The current submissions of papers deal with all facets of a mature UNIX and all the interesting developments possible on a stable system base.

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

Tutorials

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

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

Monday 23rd April

Tutorial M1.	Programming with OSF/Motif (TM) Tutor: Hans-Joachim Brede
Tutorial M2.	Advanced Network Programming Tutor: Richard Stevens
Tutorial M3	Chorus Tutors: Marc Rozier & Francois Armand

Tuesday 24th April

Tutorial T4.	Project Athena Tutor: Dan Geer
Tutorial T5.	An Introduction to 4.3 BSD Internals Tutor: Tom Doeppner
Tutorial T6.	System V Kernel Structure and Flow Tutor: Bill Rieken

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.

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 will be printed and distributed to all EUUG Members early in January, 1990. If you would like a copy of the booklet sent to a Non-Member who you think may be interested please complete the form on the next page.

Costs will be as follows:

Tutorials

Tutorial per person if booked before	26th Feb.	Members Only £180
Tutorial per person if booked after	25th Feb.	Members Only £280
* Tutorial per person if Registration and payment on the door		Members only £380

Conference

3 Day Conference if booked before	26th Feb.	Members £180
		Non-Members £260
3 Day Conference if booked after	25th Feb.	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 Enquiries

Mrs Helen Gibbons
The Secretariat,
EUUG,
Owles Hall,
BUNTINGFORD,   Tel: +44 763 73039
Herts. SG9 9PL   Fax: +44 763 73255
U.K.   euug@EU.net

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, HERTSSG9 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   £ _____

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   ☐   £ _____

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 ....

USING—UNIX Systems Information Networking Group

Patrick J Eves

bpa!bpats!peves

peves@bpats.BELL-ATL.com

Information Systems Support
Philadelphia, PA, USA

Pat Eves is a Systems Analyst in the Informations Systems Department of the Bell Atlantic Corporation. Pat is responsible for UNIX Planning and Standardisation.

What is USING?

The UNIX System Information Networking Group (USING) is a non-profit association of users interested in fostering innovations and sharing ideas and experiences about UNIX system based solutions in the telecommunications industry. The objective of this organisation is to examine UNIX system issues and to provide a forum to discuss those issues pertinent to the telecommunications industry. Specific objectives include, but are not limited to, the following:

— Conducting workshops and common interest forums to discuss current telecommunication industry issues and concerns.

— Providing two-way communication between end users and industry.

— Providing a forum for discussion and solution of business problems though use of the UNIX operating system.

— Providing a forum for users and vendors to demonstrate UNIX system products and solutions.

— Facilitating the evolution, definition and communication of telecommunication

standards.

— Providing a forum for the dissemination of information relating to the development of national and international standards of UNIX-based systems.

Background

Prior to the breakup of AT&T in 1984, UNIX system professionals (end-users, developers, managers and system administrators) throughout the Bell System had an information network to communicate issues, ideas and concerns relating to all aspects of the UNIX environment. After divestiture, AT&T had spun off the local telephone companies into 7 Regional Bell Operating Companies. The resources and the information network that were available within the Bell operating companies and AT&T prior to divestiture were no longer available. In 1986 the USING organisation was formed in an attempt to bridge the information gap between UNIX system professionals within the regional operating companies and to expand the scope to include other members of the UNIX system community in the fast growing telecommunications industry.

International Presence

More recently, USING members have demonstrated a growing interest for UNIX system products from outside the United States. Based on this interest, USING has decided to broaden its geographical scope to include Europe as well as The Far East. As a result, USING is encouraging membership and conference participation from international telecommunications professionals with an interest in the UNIX Operating System.

Sponsored Events

Since its inception in 1986, USING membership has grown to approximately 500 members. The first two annual conferences were held in Denver, Colorado. The third USING Conference will be held 26-28 March 1990 at the Hyatt Regency DFW Hotel in Dallas, Texas, USA. The conferences feature technical sessions, a solutions show, a vendor exhibit, and birds-of-a-feather gatherings. The first USING Workshop, focusing on UNIX System Administration, was held 18-19 September 1989 in Somerset, New Jersey, USA. In addition to the conferences and workshops, the USING Newsletter was created to further enhance communications between the organisation and its members and is published three times a year: April, August, and December.

Membership Information

USING membership is open to all telecommunications industry employees utilising or planning to utilise the UNIX operating system. The annual membership fee is $10.00 and the membership expires at the end of each calendar year. Membership is subject to the review and acceptance by the USING Executive Board. Send your request for a membership application to:

USING Membership
Dave Kledzik
Ameritech Applied Technologies
Room 5A
150 East Gay Street
Columbus
Ohio, 43215, USA

Additional Information

For additional information, please call +1 312 987 2323 or send your correspondence to:

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

Board Members

The current Executive Board Members and their present employers are:

President	Fred Stables	NYNEX Service Corp
Vice President	Chuck Dean	Bell Atlantic Corp
Secretary/Treasurer	Don Ackerson	AT&T
Membership Chair	Dave Kledzik	Ameritech Applied Technologies
Event Chair	Tom Raleigh	Bellcore
Media Chair	Sukan Makmuri	Teknekron Communications
Board Members	Kevin Bray	Pacific Bell
	Jerry Carlin	Pacific Bell
	Jeff Conner	Southwestern Bell
	Pat Eves	Bell Atlantic Corp
	Ashley Ewing	BellSouth Services
	Tom Foss	AT&T
	Hans Heilborn	Ericsson Telecom AB
	Tom Pottanat	NYNEX Science and Technology

Call for Papers

USING Conference—1990

Dallas, Texas
March 26-28, 1990

The UNIX Systems Information Networking Group (USING) is a non-profit association of users interested in fostering innovations and sharing ideas and experience about telecommunications industry UNIX system based solutions.

The 1990 conference will be held in Dallas, Texas March, 26-28, at the Hyatt Regency DFW Hotel. It will feature technical sessions, a solutions and vendor show, and birds-of-a-feather gatherings. The technical sessions will run from Monday through Wednesday March, 26-28, 1990. The overall theme of the conference will highlight networking and connectivity. The RBOC Solutions Show and the Vendor Exhibit will run from Tuesday through Wednesday, March, 27-28, 1990. A exhibition highlighting connectivity is planned. The primary intent of the vendor exhibit is to provide the opportunity to display advanced technology (software and hardware) relevant to the UNIX operating system in the telecommunications industry. For details on exhibiting at the Dallas vendor exhibition, please contact:

> Chuck Dean
> Bell Atlantic
> 2221-L Broadbirch Drive
> Silver Springs, MD 20904-1985
> 301-989-5080

Conference Programme

The conference program is a forum for the presentation of new developments and uses of the UNIX operating system in the telecommunications industry. Papers and presentations in all areas of UNIX systems related research and development involving networking and the telecommunications industry are solicited for formal review for the technical program of USING Conference—1990. Accepted papers and presentations will be presented during the 3 days of the conference and published in the conference proceedings.

Appropriate topics for technical presentations include, but are not limited to:

- High-Speed Networks (FDDI, etc.)
- ISDN, BISDN, etc.
- Network Monitoring & Security
- UNIX System Networking
- Protocols
- Distributed Filesystems
- File Servers
- Video and Voice Applications
- Teleconferencing
- Hypermedia
- Distributed Applications
- Mail Systems
- Videotext
- UNIX Systems Based Network Services
- Standard Operating Environments
- Field Trials & User Experience
- Fault Tolerant Systems & Applications
- Telco Applications
- Switching Software & Support Systems

Submit either an extended abstract of 300 to 1000 words detailing the major points of the talk and its relevance or a full paper which contains a 100 to 300 word abstract. Acceptance or rejection of a paper will be based solely on the technical merits of the work as submitted, so preference will go to full papers 4-12 pages). An award will be give for the best paper. To qualify, the author must adhere to all deadlines and formats. The following schedule will be followed.

> Jan. 5, 1990 Submission due
> Jan. 19, 1990 Notification to authors
> Feb. 2, 1990 Final camera ready presentation due

Send four copies of program submissions to our Program Chair:

> Fred Staples
> NYNEX Service Company
> 1166 A/A Room 8D1
> New York City, New York 10036
> 212-395-3492

Guide to Authors

Submissions of extended abstracts instead of papers will be considered for the conference. For details about the required format and method of submission please contact Fred Staples.

EUUG Executive Report

Helen M W 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 Executive Committee met on 4th September in Paris in order to finalise reports for the meeting of the full Governing Board which took place at the end of the month on 23rd September 1989.

Both meetings reviewed the current membership status and financial situation in some detail. Membership, through the sixteen national groups, continues to grow steadily, though there has been no dramatic change in numbers recently. Future growth may however be expected as it is hoped that new groups may join in the near future from countries such as Spain, Switzerland and the USSR.

The accounts which are produced in detail by the Secretariat at Owles Hall for each Executive and Governing Board meeting, highlighted the fact that EUUG is now in its strongest financial position ever. This is primarily due to good cost control, profit from conferences, and excellent response from most national groups to the revised contribution collection mechanisms introduced this year. The reserves built up are now adequate and necessary for a stable base and for underwriting conference, networking, publications and other activities.

As a result it was agreed that there would be no rise in the subscription rates for 1990, which means that they have remained stable since 1986. When inflation is taken into account this represents reduced contributions in real terms.

Relationships with other groups continue to thrive. The EUUG has been approached by USING, a UNIX special interest group of Telecommunications Manufacturers, and several discussions have been held with OSF, UNIX International and X/Open.

During the Vienna conference there was a full meeting of the Executive Board of USENIX and joint meetings were held. Interrelations are planned on newsletters, journals, conferences, software distributions and other schemes.

Results of the membership survey recently undertaken by IDC have been coming through although the final report has not yet been completed. The questionnaire received a very good response and showed that the majority of members are senior professionals connected with Industry and are big UNIX users. The results seen so far show that 25% of the membership attend conferences and that the newsletter is popular. About 3.7 people on average read one copy.

The EUUG EUnet E-mail directory is now sold out. A new edition is being planned and will be available in January 1990. Orders may be sent beforehand to the EUUG Secretariat.

The Vienna conference was a success with approximately 400 participants. The tutorials were particularly popular and well attended. For the first time at an EUUG conference an E-mail link was running and was well used. It is hoped to continue the service. Conference proceedings are now available from the Secretariat at a cost of 20 GBP including postage and packing, cheque with order.

The next conference will be held in Munich 23rd-27th April 1990. There will be two days of tutorials and three days of technical conferences. Further details appear elsewhere in this newsletter. Offers of papers may still be accepted.

The EUnet backbone managers recently held a workshop at Breukelen in Holland and met again during the Vienna conference. The networking service is seen by members as very important and there are plans for its expansion. The Chairman of the backbone managers is Yves Devilliers of INRIA, France.

Public Relations activities within EUUG are increasing and two extra people have been co-opted to the Executive Committee to cope with the increased workload. This means that Kim Biel-Nielsen will retain primary responsibility for public and external relations and will be directly responsible for press relations, EUUG presentations and special projects, but he will now be helped by Johan Helsingius who will undertake relations with external groups and Norman Hull, who will concentrate on internal relations with the national groups.

Groups are reminded that a portable EUUG stand and slides describing the activities of the EUUG are available on request from Owles Hall.

For your reference here is a list of EUUG officials:

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.uucp
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

Report from ICEUUG

Marius Olafsson
marius@rhi.hi.is

University of Iceland, Computing Services
Reykjavik
Iceland

Marius obtained a BSc Math from the University of Iceland in 1977, and a MSc Comp. Sci. in 1981 from the University of Alberta. He has been involved in computing since 1975, using various systems. He started working with UNIX back in 1979, using V.6 and has never looked back. He returned to Iceland in 1986 and is currently administering UNIX systems at the University's Computing Services.

The Icelandic UNIX Users Group (ICEUUG) was founded in November 1986. Now, at the end of the third year membership stands at 24 institutional members and 6 individual members.

Activities

ICEUUG holds monthly meetings during the winter months (there are quite a few of those here), but activity is low during the summer. Meetings on various topics of interest to the Icelandic UNIX community have been held and this winter representatives of vendors offering UNIX-related products in Iceland will be invited to discuss their products.

ICEUUG publishes a local newsletter, two or three times a year. Articles range from simple information to our members to tutorials on various aspects of using and administering UNIX systems.

Of continuing interest to us are the evolving standards on native language processing within UNIX systems. Our language makes it impossible for us to use the 7-bit standards as used by many (most) other European countries. Instead we have to use 8-bit character sets and thus traditionally have to replace many of the standard UNIX

utilities with public-domain 8-bit-transparent versions. Members can obtain these from ICEUUG to enable their systems to conform to the Icelandic standards. There are of course vendors that actually offer 8-bit-clean UNIX systems, but there are many more that don't and even some that claim that they do, but when certain areas (e.g., networking) are looked at closely things break down. Few vendors seem to realise that 8-bit-clean must include such beasts as awk and sed. Initially, many seem to think that supporting 8-bit filenames is enough.

EUnet in Iceland

Currently there are 19 sites on the Icelandic portion of EUnet. We do not expect much increase in that number as the cost of the trans-Atlantic communications is prohibitive for us and it is therefore very hard to recruit new sites. The Icelandic EUnet uses RFC-822 addressing internally and uses the ISO-8859/1 character set for communications. This means that much of the mail/usenet-software must be modified, and ICEUUG assists sites wishing to make these modifications to join the net.

There are very close ties between ICEUUG and the Association for Research networking in Iceland (SURIS) and the Icelandic EUnet backbone is in the process of being moved to a machine owned by this organisation and operated by the University of Iceland. Iceland now has a connection to NORDUnet, but is no longer a member of EARN, and the previous EARN node in Iceland (ISEARN) no longer exists. We therefore rely solely on EUnet (and to some extent NORDUnet) for our network connections to the rest of the world.

ICEUUG Executive

Einar Kjartansson (Chairman)
National Energy Authority, Reykjavik (eik@os.is)

Marteinn Sverrisson
Science Institute, University of Iceland (matti@raunvis.hi.is)

Marius Olafsson
University of Iceland (marius@rhi.hi.is)

Earlier this year Andrew Mc Phearson <andrew@stl.co.uk> bravely travelled from London to Canterbury on his bicycle to raise money for charity. This event is recorded below:

Date: Sept 2nd 1989
Miles: 60
Time to complete: 10 hrs
Charity: Crisis (aka Crisis at Christmas)
£ raised: £250.90

News from the Netherlands

Frances Brazier
frances@psy.vu.nl

Dept of Cognitive Psychology
Vrije Universiteit
Amsterdam
The Netherlands

Frances has a master's degree in Mathematics and Computer Science, and has been doing research at the Department of Cognitive Psychology for the past 7 years. Human-machine interfaces and information retrieval are her major fields of interest.

Membership

Although the NLUUG does lose a few members every year the increase in numbers still continues as shown in the table below.

April 88	Nov 88	April 89	Nov 89	
59	71	80	89	academic members (@ Dfl 300)
133	157	164	178	industrial members (@ Dfl 600)
20	31	35	37	individual members (@ Dfl 100)
212	259	279	304	TOTAL

UNIX and Connectivity—Our Next Conference

Our next conference on UNIX and Connectivity will have been held by the time this newsletter is printed—on Nov 9th. The proceedings will also have been published. Copies are available to EUUG members at cost price—Dfl 30 (approx 13 ecu). Please contact Frances Brazier at the above address.

Abstracts of the technical presentations follow below as does a list of the tutorials.

XAA UNIX Application Architecture
Willy Witsel,

NCR Nederland BV, Amsterdam
The concept of XAA will be introduced and compared to IBM's SAA. SAA's architecture and communication facilities will be briefly described, as will the XAA concept, based on standard SVR4 functionality, X/Open, POSIX 1003 and other generally accepted standards.

Network Management
in a Heterogeneous Environment
Larry Kluger,
Sun Microsystems Europe,
Bagshot, England
Network Management is a basic aspect for any distributed system. It need changes from 'nice

to have' to 'must have' as the number of workstations, servers, networks and applications grows.

Sun's network management philosophy, architecture and tools will be discussed, focusing on Sun's network management architecture.

X.400 The Realisation of Electronic Messaging within an OSI Framework
Richard van Schelven,
EGS, UNIX Software Development,
Erkrath, Germany

X.400 defines a Message Handling System with which it is possible to transmit information (text, data, graphics, digitised voice messages) worldwide from one communication system to another, independent of the underlying hardware architecture. The international X.400 standards and the corresponding European standards, their implementations and the present situation will be discussed.

SURFnet: DECnet-stronghold, TCP/IP Bastion or OSI-bulwark?
Erik Huizer, SURF-net BV, Utrecht

The technical developments of the past, present and future will be discussed. The possibilities and limitations of the network service (telecom switches, Cisco-routers), the transport service (Connectionless Network Services vs. Connection-Oriented Network Services) and the applications (mail routing, address translation, DECnet-, RSCS-, and OSI-services/protocols) will be presented.

Directory Navigation in the QUIPU X.500 System
Paul Barker, Colin J. Robbins,
University College London

OSI Directory Services have recently been standardised following the X.500/IS 9594 standard. An overview of the Directory Services model and the services offered will be provided.

QUIPU is one of the first implementations of the X.500 standard and was developed at UCL as part of the ESPRIT INCA project. The approach taken in representing Directory knowledge and handling Directory navigation across heterogeneous networks will be discussed.

EUnet Now and in the Future
Daniel Karrenberg (EUUG),
CWI, Amsterdam

EUnet's present situation will be described in detail as will the plans for the future.

Network Computing and its Applications
Bouwe van der Eems,
Apollo (subsidiary of Hewlett-Packard),
Houten

Network Computing allows users to value from various computer resources in the network: from execution of a part of an application on another machine to parallel execution of parts of an application on different machines. The system manager's tasks can be decentralised and/or executed in parallel in the network. The concept of Network Computing, its implementation and the availability will be presented as will a number of real cases.

PC LAN Integration in a UNIX Environment
Joop Veenis,
Geveke Electronics, Amsterdam

The recent history of network development, the different PC/host possibilities on Ethernet, and the PC LAN integration in a UNIX system will be discussed in turn. The architecture and functionality of PC LAN operating systems such as Novell's Advanced Netware, Microsoft's LAN Manager and other systems will be described.

MFTS/TRANSIF: Heterogeneous Communications Software in Europe
Ramesh Rao, M. B Abecasis,
R. Arnold & P. Day, SG2-B, Brussels

Multilateral File Transfer System is communication software for distribution of data and resources of many different computers. It uses the NIFTP protocol. The techniques chosen for inter-process communication in the implementation of MFTS on UNIX—TRANSIX—will be explained.

Amoeba— High Performance Distributed Computing
Sape Mullender,
Centrum voor Wiskunde en Informatika,
Amsterdam

The Amoeba Project is a project in which a

distributed operated system is being developed. This project started in 1980 and now involves the CWI, the Vrije Universiteit and Cambridge University (UK). The developments within this project will be described.

Revolution 89 or "Distributing UNIX Brings it Back to its Original Virtues"
Michel Gien, Frc Rozier,
Chorus Systems,
Saint-Quentin-en-Yvelines, France

CHORUS-V3 is the current version of the CHORUS Distributed Operating System. The CHORUS Architecture has been designed for building 'new generations' of open, distributed, scalable Operating Systems. In the presentation the architecture and the facilities provided by the Nucleus and by CHORUS/MIX, a UNIX system built with the CHORUS technology that provides a binary compatibility with UNIX System V and extended UNIX services supporting distributed applications, light-weight processes and real-time facilities, will be summarised.

Who Wants a Network?
Martijn de Lange,
ACE, Associated Computer Experts,
Amsterdam

While connectivity is just beginning to be appreciated and accepted by the 'real' end users, the R&D community is struggling with the newly introduced chaos and the inconsistencies in the swamp of interconnected systems. An assessment is given of current engineering practice and the challenges associated with the usage of networks and network related tools and interfaces in an engineering environment. The considerations that are background to the approach followed in the development of the SPIRIT Workstation, a high performance technical workstation, largely extend beyond simple peer-to-peer connectivity. It is highlighted that the exhausting and disaster-prone networking techniques of today will lead to utter chaos in the mid nineties when large scale networking will be common in end-user environments.

The following topics will be covered in the tutorials:

Bird's eye view of UNIX OS,
Rob Westerman,
West Consulting, Delft

Security in a (UNIX) Network,
Frans Schippers,
Xirion bv, Amsterdam

LANs on TCP/IP,
Hans van Staveren,
Vrije Universiteit, Amsterdam

Network File System (NFS),
Jos van der Meer,
Sun Microsystems Nederland,
Amersfoort.

In addition to the (parallel) technical sessions and tutorials an exhibition will be held. At the end of the day a forum will be held in which Theo de Ridder (SERC), Michel Gien (Chorus), Andy Tanenbaum (Vrije Universiteit), Sape Mullender (CWI), Martijn de Lange (ACE) and Marten van Gelderen (Xirion), will discuss the future of UNIX: "Beyond UNIX...".

UNIX and Parallel Systems

Our next conference, to be held on May 17th, will be on UNIX and Parallel Systems. Anyone interested in contributing to this event is invited to contact the NLUUG (address on the back cover of the Newsletter).

The Backbone

The Stichting NLnet, the organisation responsible for NLnet, has had a change of chairman: Ted Lindgreen is now responsible for the daily activities. Marten van Gelderen, the former chairman is still actively involved as a member of the Board.

IP-connectivity within Europe, InterEUnet is a topic which is receiving a lot of attention.

UKUUG Report

Mick Farmer

mick@cs.bbk.ac.uk

Birkbeck College
London
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.

Membership

Our membership continues to increase in number, particularly commercial members. We are approaching saturation point in respect of academic members, most of which are also connected to JANET and UKnet. The table below shows the breakdown of our membership at the beginning of October.

Category	Number
Academic	98
Commercial	221
Honorary	5
Individual	48
Total	372

The UKUUG Executive is currently investigating the possibility of restructuring our membership categories, but no decision has been reached so far.

Winter 1989 Technical Meeting

Our next technical meeting is being held at the University of Wales College of Cardiff on 11/13 December, 1989 and will be hosted by Robert Evans from the Department of Computing Mathematics. As usual our Winter meeting has a strong networking flavour. Presentations already agreed include:

- ISO higher-level protocol developments
- ISODE latest developments
- X400 developments
- X500 directory service
- NFS/FTAM
- OSI transition plans for ac.uk
- RARE/COSINE developments
- IXI
- X windows over OSI
- Fidonet developments
- TCP/IP protocol stack

Also included are talks on the future of UKnet, EUnet, and Usenet. We are continuing the recent policy of running popular tutorials at our meetings, this time Jim Craigie will introduce X400 fundamentals. For further details contact Robert (robert@computing-maths.cardiff.ac.uk)

or the UKUUG Secretariat (ukuug@ukc.ac.uk).

UNIX System Administration Workshop

We are holding this one-day Workshop at the Institute of Education (Central London, England) on 14 February, 1990. It will be hosted by Neil Todd from GiD Ltd. Topics likely to be discussed include:

- Smart shells and menu systems

- Administration in a distributed environment

- System audit procedures

- Software integration and test procedures

- Data Protection Act

➤ Yellow Pages—friend or foe?

Further suggestions for topics and talks are welcome. Please contact Neil (neil@gid.co.uk) or the UKUUG Secretariat (ukuug@ukc.ac.uk).

As with our previous successful UNIX Security Workshop, a video of this event will be available (see the UKUUG Report in the Autumn '89 EUUG Newsletter).

Summer 1990 Conference

This meeting will be held at the Royal Lancaster Hotel (Central London, England) on 9/13 July, 1990 and it will be hosted by the UKUUG chair, Sunil Das (if he's not off somewhere Suniling :-). This major conference, entitled *UNIX—The Legend Evolves*, will feature many of the major workers in the UNIX and operating systems community. Those who have agreed to speak include:

- Dave Presotto, Rob Pike, and Ken Thompson from Bell Labs unveiling *Plan 9*, the next generation of UNIX at Bell Labs.

- Brian Kernighan from Bell Labs talking about the lessons we have learned for future software developments.

- Kirk McKusick and Mike Karels from the University of California, Berkeley talking about 4.3BSD Internals.

- John Bentley from Bell Labs (Programming Pearls).

- Andrew Tanenbaum from Vrije University (MINIX).

- Stuart Feldman from Bell Labs (make and f77).

- James Gosling from Sun Microsystems (Emacs).

- Doug Comer from Purdue University (XINU).

- Piers Dick-Lauder from the University of Sydney (Share scheduler).

This meeting will follow the traditional EUUG format and consist of two days of tutorials (Monday-Tuesday) followed by the three-day conference (Wednesday-Friday).

London UNIX User Group (LUUG)

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

FaceSaver Project

We are in the process of implementing a FaceSaver system. It will be used for our Summer 1990 Meeting (see above) with a trial run during our Winter 1989 Technical Meeting (see above). We intend to offer this service to the EUUG and other affiliated groups.

Stop Bit

That's all for now folks!

USENIX Association News for EUUG Members

Donnalyn Frey
donnalyn@frey.com

Frey Communications
Fairfax, VA USA

Ms Frey 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 will be held in sunny Washington, DC on January 20-26, 1990. The first two days will be devoted to tutorials, with the next three days for technical sessions. Topics which may be presented include UNIX and Artificial intelligence, Ada and UNIX, Software Release Systems and Servers, Architectures and Compilers, File Systems and Servers, Distributed Systems and Servers, Distributed Systems and Services, User Interfaces, and Novel Applications. For further information on the conference, contact the USENIX conference office.

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 Call for Papers will be announced in November 1989. For a copy of the Call for Papers, contact the USENIX Conference office.

1990 C++ Conference

The Call for Papers has been issued for the 1990 C++ Conference. The conference will be held in

San Francisco, California on April 9-11, 1990. The conference is expected to be of interest to a broad range of C++ users and potential users. Tutorials are planned on April 9 to teach C++ to novices, as well as more advanced tutorials. Technical sessions are planned for April 10 and 11. Registration information on the conference will be available from the USENIX conference office in January.

The technical sessions will cover a spectrum of work on and with C++, including a diversity of users and applications. Papers are solicited on all aspects of C++, including:

Applications
Libraries
Programming Environments
Case Studies
New or Improved Implementations

Extended abstracts (up to 2 pages) or papers (9 - 12 pages) must be received, either electronically or on paper, by **Friday, January 12, 1990.** Authors will be notified of acceptance by February 5 and must submit a full paper by April 9.

Queries should be directed to:
 Jim Waldo
 CHR 03 DE
 Apollo Computer
 300 Apollo Drive
 Chelmsford, MA 01824
 USA

 waldo@apollo.com, decvax!apollo!waldo

 (last resort) +1 508 256 6600 x 5747

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 or telephone at +1 714 588 8649. The conference office can provide you with information on the annual Computer Graphics, Large Installation Systems Administration, UNIX Security, and UNIX and Supercomputers workshops, as well as other workshops. The office can also provide information on the 1990 C++ conference and the semi-annual technical conferences. A schedule of upcoming events includes:

USENIX Association Future Meetings

Date	Location	Topic
11/16-17/89	Monterey, CA	Graphics Workshop V
1/23-26/90	Washington, DC	Semi-Annual Conference
4/9-11/90	San Francisco, CA	C++ Conference
6/11-15/90	Anaheim, CA	Semi-Annual Conference and Exhibition
Fall 90	Honolulu, HI	Software Development Environments joint workshop with SIGMA Project, Japan
Fall 90	Northeast US	MACH Workshop
1/22-25/91	Dallas, TX	Semi-Annual Conference
6/10-14/91	Nashville, TN	Semi-Annual Conference & Exhibition
1/20-24/92	San Francisco,CA	Semi-Annual Conference
6/8-12/92	San Antonio, TX	Semi-Annual Conference & Exhibition

X User Group

 Niall Mansfield
 Unipalm Ltd
 145 St Neots Rd
 Hardwick
 Cambridge CB3 7QJ
 UK
 nmm@unipalm.co.uk
 +44 954 211797

An inaugrial meeting was held on 21st Nov in IEE Savoy Place, London.

Those interested please contact the above.

USENIX Winter Conference

This is being held from 22nd to 26th January 1990 at the Omni Shoreham hotel, Washington, DC, USA.
Conference programme:

	Wednesday 24 January	
9:00-10:30	Introductory Remarks Remarks from the Board	Dan Klein Ellie Young
Keynote Address	NASA's Manned Spacecraft Computers	Jim Tomayko
10:30-11:00	Morning Break	
11:00-12:30	Virtual Memory	Chair - Chet Juszczak
	A Dynamic File System Inode Allocation and Reclaim Policy	Ron Barkley, T. Paul Lee AT&T Bell Laboratories
	Insuring Improved VM Performance: Some No-Fault Policies	Danny Chen, *et al* AT&T Bell Laboratories
	An External Pager Implemented as a UNIX System V Release 4 Virtual File System	Dean Thomas Unisys Corporation
12:30-2:00	Lunch Break	
2:00-3:30	Architecture & Debuggers	Chair - John Mashey
	Implementing a Mach Debugger for Multithreaded Applications	Deborah L. Caswell, *et al* Hewlett-Packard Company
	pdb: A Network Oriented Symbolic Debugger	Paul Maybee Solbourne Computer, Inc.
	Some Efficient Architecture Simulation Techniques	Robert Bedichek University of Washington
3:30-4:00	Afternoon Break	
4:00-5:30	Applications	Chair - Susanne Smith
	Software Tickerplants on UNIX	Mark Luppi, *et al* Fusion Systems Group
	GENESIS and XODUS General Purpose Neural Network Simulation Tool	John Uhley, *et al* California Institute of Technology
	Keynote - A Language and Extensible Graphical Editor for Music	Tim Thompson AT&T Bell Laboratories

	Thursday 25 January	
9:00-10:30	Utilities	Chair - John Devitofranceschi
	Integrated Interactive Access to Heterogeneous Distributed Services	William E. Weihl Massachussets Institute of Technology
	The UNIX System Math Library, a Status Report	Steve Sommars AT&T Bell Laboratories
	Tcl: An Embeddable Command Language	John K. Ousterhout University of California, Berkeley
10:30-11:00	Morning Break	
11:00-12:30	Kernel Internals	Chair - Charlie Perkins
	An Event-based Fair Share Scheduler	Raymond B. Essick Prisma, Inc.
	Parallel STREAMS: a Multi-Processor Implementation	Arun Garg Sequent Computer Systems
	Implementing Berkeley Sockets in System V Release 4	Ian Vessey, Glenn Skinner Sun Microsystems
12:30-2:00	Lunch Break	
2:00-3:30	Networks	Chair - Alix Vasilatos
	Two Network Management Tools -or- (How Many Packets Would a Packet Router Route if a Packet Router Could Route Packets?)	Jeff Okamoto Hewlett-Packard Company
	Packet Trains on NSFNET National Backbone A Traffic Characterization	Steven A. Heimlich University of Maryland
	Pseudo-Network Drivers and Virtual Networks	Steven Bellovin AT&T Bell Laboratories
3:30-4:00	Break	
4:00-5:30	Special Session on Ethics and Computers	Moderator - Rob Kolstad

	Friday January 26	
9:00-10:30	User Interface Management Systems	Chair - Dan Geer
	The Serpent User Interface Management System	Brian Clapper, *et al* Software Engineering Institute
	Parallel Object-Oriented UIMS with Macro and Micro Stubs	Masami Hagiya, *et al* Kyoto University
	MTX - A Shell that permits dynamic rearrangement of process connections and Windows	Stephen A. Uhler Bell Communications Research
10:30-11:00	Morning Break	
11:00-12:30	File Systems	Chair - Kirk McKusick
	Using UNIX as One Component of a Lightweight Distributed Kernel for Multiprocessor File Servers	David Hitz, *et al* Auspex Systems Inc.
	A Highly-Parallelized Mach-based Vnode File System	Alan Langerman, *et al* Encore Computer Corporation
	Disk Scheduling Revisited	Margo Seltzer, *et al* University of California, Berkeley
12:30-2:00	Lunch Break	
2:00-4:00	Languages and Software Engineering	Chair - Dan Klein
	Postloading for Fun and Profit	Stephen C. Johnson Ardent Computer Corp.
	Multiple Site Source Reconciliation	Dodi Francisco, Lois C. Price TRW Financial Systems, Inc.
	CVS-II: Parallelizing Software Development	Brian Berliner Prisma, Inc.
	Ada and Binary UNIX Standards	Mitchell Gart Alsys Incorporated

Further details may be obtained from:

USENIX Association
2560 Ninth St., Suite 215
Berkeley, CA 94710
USA
office@usenix.com
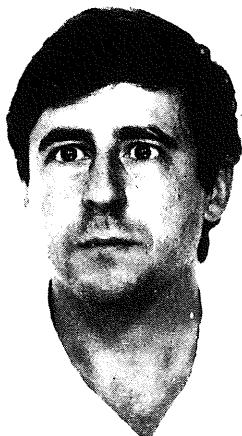
EUnet Growing Up in Spain

José A. Mañas

jmanas@dit.upm.es

Postmaster@Spain.EU.net

Dept. Ingeniería Telemática
ETSI Telecomunicación
Madrid, Spain

José A. Mañas is an Associate Professor at the Telematics Engineering Dept. (DIT) of the Technical University at Madrid (UPM). Since late 1987 he has been in charge of mail services provided by the Department to the research community. He is very fond of UNIX and is currently trying to become a guru, but it seems to be a difficult goal to reach. In the meantime, he spends his time lecturing on protocol engineering.

Introduction

The DIT (Dept. de Ingeniería Telemática) has been involved in mail services for some years. To a large extent, this department has been directly related to the evolution of a major part of the networking in Spain. In addition, it currently runs the Spanish backbone for EUnet, hosting the central node goya.

This paper presents an overview of the current status as seen from the EUnet world from the standpoint of: members, topology, distribution, resources, traffic and accounting. There are other networks in Spain, and cooperation is constructive in that it enables actual network topology to be hidden under unified addressing. The interoperability with these other networks is also considered, along with some figures on the amount of exchanged mail.

The given name of the Spanish entry host is goya. Goya offers three services: 'mail', 'news', and 'archive service'. The first two are fairly well known and deserve no further comment. The archive service is similar to other services around EUnet and other networks. It

implements a mail accessible data base. Users send a request via mail, and something is sent back via mail in response. The first request is usually

```
To: Archive Server <info@dit.upm.es>
Subject: help
```

The amount of available information is about 30 Mbytes (after compressing); not very impressive, but it seems to be really useful in view of the number of users[1] So far we do not know whether this service will be subsumed by the EUnet distributed archive-service that is being set up, or whether perhaps both will be coordinated, or...
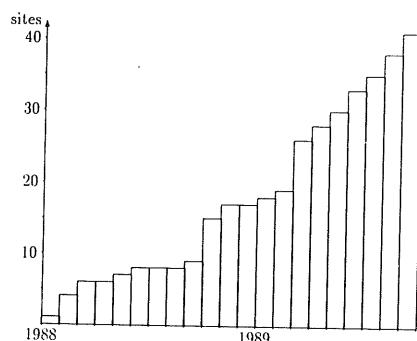
History

Goya began operation in the first quarter of 1986 with a link to mcvax. The net remained quite small for the first several years, and by early 1988

---

1. During last July, 92 requests were processed, and 2.520 Kbytes of information were sent.
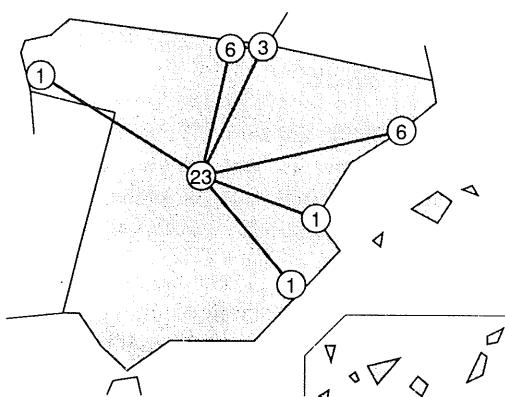
there was still only one registered site, namely dit.upm.es. Since January 1988 the net has evolved considerably and there are currently 41 registered sites providing service to more than 68 hosts. The actual number of hosts is difficult to estimate since domain addressing is strongly recommended to Eunet members, and most sites have only registered a frontend host. The number of users is even harder to estimate.

The following diagram shows the evolution of registered sites for the last two years. The growth is roughly linear; about 2 new sites per month.
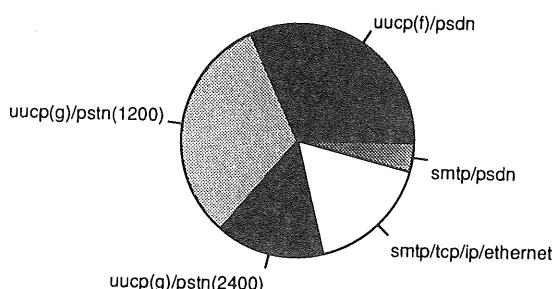
¹During last July, 92 requests were processed, and 2.520 Kbytes of information were sent.

The network topology is basically a star, centered at Madrid, where goya lives. The net is highly concentrated within Madrid, although the number of sites around the country is growing quickly. The following picture shows the distribution of sites in the country (figures denote the number of sites in each province)

With respect to the communication topology, most of the nodes work at 1200 on the PSTN. A few of them succeed in running 2400 modems, but this is usually restricted to Madrid due to the poor quality of public lines. An increasing number of nodes work on PSDN (x.25). Lastly, a few sites are connected to a common LAN, where SMTP/TCP/IP is used. For international transfers, there is a switched x.25 line working at 9600 bps. The following picture shows the distribution of links running each protocol stack.

Connection Types

Resources

The Spanish 'backbone', goya belongs to the DIT which takes care of its operation. Goya is a SUN-3/160 running at 3 MIPS, with 12 Mb of main memory and 600 Mb of disk storage.

There are two phone lines for incoming connections, accepting 2400, 1200, 600 and 300 bps transfers. There is also an x.25 line at 9600, that is used for incoming connections and to link with other backbones in Europe.

Human resources are harder to estimate since most of the department is more or less directly involved in communication issues, and everyone plays a rôle when problems crop up. It can be estimated that 1 effective person is fully devoted to the running of EUnet, with about 7 people hiding behind <postmaster@Spain.EU.net>. In addition, there is approximately 1 more effective person for gatewaying issues, relation to other networks, and so on.

These are all the resources available so far. With them, goya currently handles over 170 Mbytes per month. Let's try to get an idea of how it is distributed.

Layer 4 Traffic

First, focusing on layer 4 (transport in ISO/OSI), the following table shows the UUCP traffic classified per direct link to goya. Countries with a direct link are denoted by their top-level domain

(e.g., .fr for France). Cernvax, mcsun and mcvax are grouped apart since they usually handle much more traffic than what is expected

for their respective host countries (.ch, .nl). The entry NAT reflects national traffic, that is, .es transactions.

1989	uucp traffic (Kbytes)								
	Jan	**Feb**	**Mar**	**Apr**	**May**	**Jun**	**Jul**	**Ago**	**1989**
.de	2133	1821	1110	1833	1716	1659	2145	1644	14061
.dk	96	615	17	3307	108	63	167	39	4412
.fr	2892	2744	2794	2471	2382	2419	3301	1669	20672
.gr	0	7	174	177	352	38	708	103	1559
.it	96	32	11	11	45	876	42	1119	
.pt	38	23	30	22	46	25	18	1	203
.se	118	57	63	89	335	158	129	361	1310
.uk	399	1150	1851	2074	1738	763	1430	608	10013
cernvax	0	42	415	2904	1470	790	823	618	7062
mcsun	0	0	0	0	0	0	0	40678	40678
mcvax	47735	53710	65996	75407	69540	76037	79434	70321	538180
NAT	14868	21812	12965	51963	66018	58137	58142	58930	342835
TOTAL	68285	82077	85447	140258	143716	140134	147173	175014	982104

Layer 7 Traffic

Climbing up to layer 7 (application in ISO/OSI), and therefore concentrating on end-to-end mail transactions, the following table shows a classification of mail handled by goya. The data correspond to July 1989, when 9.505 messages circulated via goya. The numbers represent total number of Kbytes between zones.

from ^{to}	eunet	rare	earn	EUR	ITC
eunet	4,393	1,490	715	3,494	432
rare	767	202	0	400	10,118
earn	4,648	445	2	0	0
EUR	4,534	1,385	67	-	-
ITC	5,071	4,586	598	-	-

National nodes are classified according to network membership. EUnet is the Spanish part of EUnet; RARE is the Spanish part of RARE MHS. EARN is a bit more complicated, and covers the whole BITnet. The problem is that there is a national gateway into EARN where everything goes, whether hosts are national recipients or not. In fact, goya does not hold a list of EARN registered hosts.

EUR covers European sites. Geographically membership to Europe is extracted from top level domain or from EUnet host tables. Lastly, ITC covers intercontinental sites.

Obviously, there must not exist any international transit traffic, either European or Intercontinental. There should not be any traffic passing between any other two nets in Spain either, however the traffic exsists and is quite noticeable. It deserves some attention.

First of all, as a general comment, notice that domain addressing hides real network membership under a common .es top-level domain. Furthermore, EUnet is a quite well known entry point for Europe, and within EUnet, goya is the entry point for top-level domain .es. That is the reason for so much traffic entering Spain via goya, even traffic that is to be gatewayed into other networks.

Now, for the details:

1. rare → rare

   This traffic is due to a heavily used distribution list that is expanded at the EUnet side, but it is used by both EUnet and Rare people.

2. earn → rare

   This traffic is due to the fact (fault?) that we are grouping together senders having a .bitnet top-level domain, either national or international.

3. earn → earn

   This traffic results from the hiding of real network under common .es addressing.

Although within BITnet a Spanish node is declared as entry point (EB0UB011), some BITnet sites do not have very up to date tables.

4. rare → Europe and Intercontinental

This traffic is a result of the service provided to Spanish RARE sites on an equal rights basis. We handle part of their mail for sites where EUnet connectivity is better, and they do the same for us.

5. European and Intercontinental → rare and earn

Again, it is due to the hiding under .es.

There is an important conclusion to be derived from these comments: the current level of interoperability between networks makes it infeasible that any of the nets could survive without good cooperation with the others.

The Other End

Lastly, it may be interesting to know how is traffic distributed when one end is .es. Last July's traffic has been classified according to the other end. Numbers are Kbytes, once again. European UUCP hosts have been assigned to their respective host country. Extra-european ones are collapsed under .uucp. Hosts in BITnet are not classified either, and therefore they all show up under .bitnet.

domain	from	to
.ar	5	9
.au	7	91
.be	1	43
.ca	398	1
.ch	604	40
.cl	0	7
.de	508	468
.dk	45	82
.fi	90	246
.fr	1641	589
.ie	139	82
.it	28	248
.jp	8	2
.nl	590	141
.no	81	1
.nz	9	5
.pt	1	4
.se	48	7
.uk	1068	439
.us	25	5

domain	from	to
.arpa	1499	3
.com	2150	145
.edu	7435	10458
.gov	1138	26
.mil	59	4
.net	1048	94
.org	1610	7
..cern	7	4
.irl	0	1
.oz	0	1
.uninett	443	0
.bitnet	6514	2130
.uucp	3001	1789

Other Networks
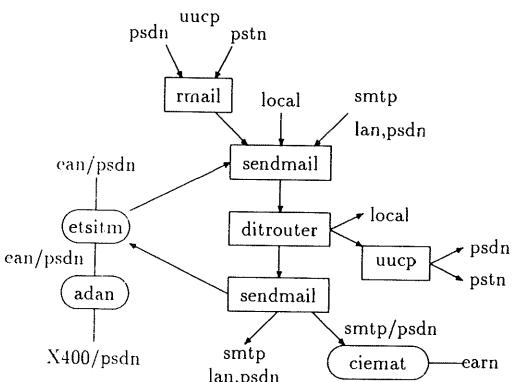
There are close relationships with EARN (BITnet), FAEnet (Spanish branch of HEPnet), and RARE MHS. All these networks are integrated under domain addressing (.es), and routing is handled entirely within the country.

For gatewaying into EARN and FAEnet, an agreement exists with ciemat.es, (a HEPnet site with a variety of computers) that is registered both at EARN and HEPnet. SMTP over PSDN is used between goya and ciemat.es.

With respect to RARE MHS, goya is directly registered in it. The EAN package permits straightforward gatewaying.

Software Architecture

There is a complex net of packages running to provide the services described above. The following picture describes their relationship.

Sendmail is the heart of everything. It accepts incoming messages via different media: UUCP (g) over PSTN, UUCP (f) over PSDN, UUCP (t) over TCP/IP/ethernet, SMTP over TCP/IP/ethernet, SMTP over PSDN, and from the EAN gateway into RARE MHS. This sendmail doesn't deliver mail, but rather concentrates on normalising addresses. Normalised messages are handed to ditrouter, a local version of Piet's router [2]. This package accesses data bases looking for delivery routes, checks for acceptable sender and recipients, logs traffic for later accounting, and delivers mail either locally, or via UUCP, or to a second sendmail if SMTP or further rewriting is required.

Accounting and Charging

There is a national project named IRIS that currently funds EUnet costs, which consists of public lines, and backbone and gateways operation. This program funds the other operational networks in Spain on an equal basis; namely EARN, FAEnet, and the pilot RARE MHS service. IRIS started last January 1988, and will run till the end of 1991.

This situation makes the everyday operation very comfortable, since it virtually runs for free. But this will not be the case for long; most notably for companies. Therefore, an accounting package is being developed from which we shall be able to distribute the costs when the funding stops.

Accounting is done at layer 7 (of ISO/OSI), that is, we don't care about transport transfers, focussing instead on end-to-end transfers. The routing software (sendmail) pays plenty of attention to removing routes and finding out real senders and real recipients. Messages are then classified according to 4 coarse cost areas: local, national, european, and intercontinental. Different weights are assigned to each area to get an overall figure that reflects the use of the net resources. This package providing the traffic data given above.

Accounting is also used for coping with fixed costs (lines, equipment, staff, ...) in such a way that these costs are paid proportionately to the amount of use of the net, but having an upper

boundary for heavy users. One of the objectives is to keep a low hook-up fee, while at the same time keeping it reasonable for heavy users.

These accounting procedures provide a means of distributing costs. The idea is to have a number of fixed cost bands, and use the accounting to allocate users on the corresponding band. This avoids the details of exact billing, and permits easy pre-estimates of costs, something highly appreciated for planning.

News is charged on a fixed-cost basis. There are three levels: (1) no news, no cost; (2) any news you wish, full price; and (3) up to 10% volume of news (approximately), 10% price (exact). These bands permits easy billing, quasi-fixed costs, and somehow help new news' subscribers to keep costs small.

These accounting procedures are currently subject to discussion within the net, and they may be subject to modifications before actual implementation; but users have informally agreed on them for the time being. Now we have to look at the exact figures.

Problems

The major problems are the following:

● Lack of experience in running UNIX[TM]. Most Spanish computer users are used to VMS[TM] systems, and UNIX makes them feel uneasy. I believe that gurus are really needed for running UNIX, and the lack of expertise at the site (and in the environment) provokes a negative reaction from [system] managers. This reaction gradually disappears as they become more expert and they see that things are actually running.

● Poor quality of public lines. Both phone lines and data lines perform poorly in Spain. Installation delays are on the order of months, or even years! High speed data lines are extremely difficult to get. On phone lines, it is pretty difficult to work at more than 1200 bps when you are to connect two cities. The system functions better during the night (of course: modems much prefer not hearing neighbour's conversations).

This situation may drastically change when Trailblazers arrive (some day). Currently there are problems to get them at a reasonable price, and there are only a few of them in

2. Piet Beertema <piet@EU.net>

Spain (none in EUnet; all for private BBS).

● Absence of a national UNIX users group. There have been several attempts to one up, but all of them have failed so far. There seems to be nobody with the resources and abilities to run it. This fact prevents the more rapid introduction of UNIX in Spain, as well as a higher level of cooperation.

● Lack of source licenses. There have traditionally existed licensing problems from AT&T, due to certain flaws in the Spanish laws with respect to proprietary rights. The problems seem to be partly solved since Spain joined the EEC. But not completely, and the fact is that there are only three (3) source licenses for UNIX in Spain, which is really inadequate for a country with almost 40 million people. Out of these three licenses, only one is held by a university.

This certainly prevents much enthusiasm for running UNIX at the universities, a fact which negatively impacts on the companies which recruit graduates from those universities. Nevertheless, the growing involvement of Spanish researchers in European projects is rapidly introducing a significant level of UNIX knowledge.

Expectations

The future can be reasonably foreseen. I expect mail sites to continue to grow in a linear fashion for some time. The number of News sites (currently only 3) should rise quickly as the quality of lines improves. There are a number of reasons to justify these expectations. Some of them were already outlined in the section on problems. I believe it is sensible to say that participation in Europe is greatly helping researchers to become aware of UNIX, and the growing number of UNIX machines in universities (something that is currently taking place) will mean a greater impact of UNIX on industry in the near future.

The unavailability of UNIX, together with very favorable treatment for universities (high discounts, basically), helped DEC to virtually fill up the university computing centers with VAXes running VMS. This situation has been changing more recently, however. Current DEC interests seem to be in other directions, and the university market is being taking over by Sun Microsystems

and Hewlett-Packard, both strongly promoting UNIX.

This vision is somewhat oversimplified, but the the fact remains that the final outcome is a growing number of UNIX users, which directly results in a growing number of EUnet members.

There is an additional reason for expecting linear growth: IRIS, the National Program for networking in Spain, is strongly encouraging the connection of new sites. Having this service for free (although only for some time) greatly helps people to decide to start. Some decrease may be expected when the project ends.

We'll have to wait and see what the future has in store for EUnet as it continues to grow up in Spain.

Acknowledgements

There are many individuals that help to keep goya running. A few of them are heavily involved, and their work should be acknowledged: Inmaculada Pindado, José R. Alonso, José M. Rivero, and José L. Jiménez.

From: Inge Arnesen <ingea@ifi.uio.no>
Subject: Trailblazer in Norway
To: uknet@ukc.ac.uk

Re: Article in EUUG Newsletter— Trailblazer & TCP/IP services in Europe

For your information, the Trailblazer is approved by the Norwegian authorities and is currently available from among others, the former state owned monopoly, TBK. The trailblazer is increasing in popularity in Norway and is doing very well vs. the V.32 modems, but there are still few around, mainly because of the high cost and rather low advertising profile.

More information is available from:

　　TBK Comms. systems: +47 2 63 22 00.

　　Inge (BoB) ingea@ifi.uio.no

　　Inge Arnesen,
　　University of Oslo,
　　Norway.

EUUG Software Distribution

Frank Kuiper
euug-tapes@cwi.nl

Centre for Mathematics and Computer Science
Amsterdam

```
        _] [__|  |
  <_____ |-1
       O-O-O
```

To start with, the picture you see on this page is not neccessarily mine! It certainly was not so in the last issue of the EUUGN :-)

Comments I received about it ranged from: "Did you have an accident?", through: "Did you have plastic surgery done?", to: "I like your new looks!". (Ahem!)

But, if in the future Alain can more accurately describe my mug shot to a hasty printer, I will feel much better!

Since the last issue of the EUUGN there is one new tape available. It is the 'Vienna, autumn 1989 conference tape', EUUGD18 We decided to try a small experiment, the result of which is a tape absolutely loaded with games. All kinds you can think of: adventure games, tactical games, screen oriented, etc, etc. There are different sets, for the Sun architecture, the X environment, and a general one.

Some people thought this "a waste of bandwidth, effort, time and money", but most others seemed to be quite happy with it. At CWI we have a so-called 'Game Keeper <play@cwi.nl>'. He maintains most of the on-line available games at CWI. He helped in making this tape, by supplying bug-free games. So the games you find on this tape have an added value, above that of the versions you can get from the news network, archives, or bulletin boards. They work, and sometimes have added features. How about that for service.

For the Munich conference, spring next year, we also plan to have a conference tape. A very serious one this time. The theme of this tape will be 'Graphics'. If you have special request items, please contact me and I will try to get hold of the software.

If you have other things you would like to see become (part of) an EUUG Software Distribution, please tell me. I know that some people would like to have the X Test Suite, the latest version of Kermit, etc. And although it is rather time consuming for me to try and keep updates for every program ever distributed on an EUUG tape, I will try to do so. Especially with items like X, 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 (August 1988) 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.

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

The first price listed is for reel-tapes in tar 1600 bpi format, the second one is for distributions on cartridge tapes in 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 120,-/180,-

EUUGD2:   Early Pascal compiler of the Free University of Amsterdam. V7 source licence minimum.
Price: DFl 120,-/180,-

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 150,-/180,-

EUUGD5:   A collection of benchmark programs made up by EUUG.
Price: DFl 60,-/180,-

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 240,-/300,-

EUUGD7:   UNIXISTAT Version 5.2. A collection of about 25 data manipulation and analysis programs written in C by Gery Perlman.
Price: DFl 60,-/180,-

EUUGD8:   A collection of useful software, based on the so called Copenhagen tape (EUUG Unix conference Autumn 1985).

EUUGD9:   A collection of useful software, based on the so called Florence tape (EUUG UNIX conference Spring 1986). Price: DFl 150,-/210,-

EUUGD10:   MMDFIIb. Multichanel 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 90,-/180,-

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 120,-/180,-

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 120,-/180,-

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 120,-/180,-

EUUGD14:   -> NOW A NEW VERSION <- This is version 5.0 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 systems 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 120,-/150,-

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: Søren Sørensen
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 X11 Windowing system material, release 3: X11R3 This is one tape, containing the entire distribution from MIT in compressed format. Uncompressed this is some 87 Megabytes. This includes the core system, as well as much user contributed software.
Price : DFl 120,-/180,-

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 the 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 120,-/180,-

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 120,-/180,-

EUUGD18: This is the 'Vienna EUUG 1989 autumn conference tape', and consists entirely of games! There is a SUN specific set, a set for the X Windowing System environment, and a general useable 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 120,-/180,-

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)
> Fascimile: +31 20 5924199
> Telex: 12571 mactr nl
> Internet: euug-tapes@cwi.nl

Please note that for distributions D1, D2 and D4 (and in some cases also for D8) 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 HFl 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 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.

> 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: ....

This page may be photocopied for use.

ISO/IEC JTC1/SC22/WG15 (POSIX) Meeting October, 1989

Dominic Dunlop

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 , and has hung around there ever since, writing the odd paper, contributing to 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 its own UNIX computer.

Report to EUUG and Usenix on ISO/IEC JTC1/SC22/WG15 (POSIX) Meeting 11th-13th October, 1989

Introduction

Working Group 15 of Subcommittee 22 of Joint Technical Committee 1 of the International Organization for Standardization and the International Electrotechnical Commission (ISO/IECJTC1/SC22/WG15") met in Brussels, Belgium, from the 11th to the 13th of October in order to further the POSIX standardisation effort. I was present at the meeting as an observer with the brief of reporting back to you. This report is the second jointly commissioned by the European UNIX systems User Group (EUUG) and Usenix. If you have any comments, or need clarification or further information, please contact me at the mail address above.

First, a summary of the most important aspects of the meeting:

Summary

- The big news is that the working group has recommended that ISO accepts the POSIX operating system interface in its current form as international standard (IS) 9945-1. Assuming that this recommendation is accepted, an international standard which is identical to IEEE Std 1003.1-1988 should be registered by ISO in the next few months.

- During the balloting of the standard at the international level, a number of comments were raised. These will be addressed by the

production of a revised International Standard on short order—by next June according to the current schedule. The result will be a version of POSIX in which known problems are fixed, but which is not extended in any way.

- Extensions, such as real-time facilities, transparent network file access, and security features will be added in future releases of the international standard.

- The cooperation of the IEEE POSIX project in producing standards which are acceptable to ISO and to its members is critical to the timely production of ISO standards. Steps were taken to make sure that IEEE documents are produced in a format that is acceptable to ISO, and that IEEE work on the revision of its 1003.1 standard is synchronised with the work of the ISO working group.

- Draft 9 of IEEE 1003.2, the proposed IEEE shell and utilities standard, has been accepted as Draft Proposal (DP) 9945-2. This means that the movement towards an international standard in this area is now officially under way.

- The problems raised by the suggested adoption of the whole of issue 3 of the X/Open Portability Guide as a European prestandard (see report on May, 1989 meeting) seem to have receded: European

alignment with a number of formal international standards is finding acceptance as a viable and more useful alternative.

- The working group has set up 'rapporteur groups' on conformance testing, internationalisation, and security in order to ensure that future international standards for POSIX take account of the developments in, and of the requirements of, these important areas.

- The next meeting of the working group does not take place until June, 1990. Making a virtue of necessity, the group hopes to achieve much before that time.

POSIX as an International Standard

The international ballot period for Draft International Standard (DIS) 9945, *Portable operating system interface for computer environments*, closed at the beginning of September. The DIS is identical to draft 13 of the IEEE 1003.1 POSIX standard, which in turn is identical, except in details of layout, to Std 1003.1-1988 published by the IEEE.

Of 26 national standards bodies entitled to vote, 19 approved the standard, one (South Africa) abstained, and one (Japan) voted against. (The five remaining countries did not vote.) Broadly speaking, ISO rules require only 75% of those voting to vote in favour in order that a standard is accepted. Where there are only one or two votes against, as in this case, the situation is even more clear-cut. Nevertheless, ISO rules require the technical committee responsible for the standard to show that it has considered the concerns of the objectors, even if it has decided not to address them by amending the draft standard.

Japan's major worry was simply that the document did not *look* like an International Standard—a matter on which France, despite voting in favour, and ISO's Central Secretariat, had also voiced concern. Instead, DIS 9945 looks like what it is—the draft of an IEEE standard—and may consequently be difficult to navigate for those used to ISO's standard format for standards.

This editorial issue could be handled simply by instructing ISO's Central Secretariat to re-enter the document text, and set it in the required format. This would take perhaps a year, and would not address the large number of 'non-normative' changes already known to be required

in the document as a result of work done by the IEEE over the past year. These changes are currently under discussion within the IEEE as P1003.1a. They are thought not to affect substance of the standard, merely clarifying it, fixing a number of small errors, and adding standard C function prototypes. However, ISO procedures sensibly require that any change to a draft standard must result in a new vote on the amended document, and consequently a further delay to the acceptance of a final standard.

Judging that it was more important to get a POSIX standard out in the field as soon as possible, rather than to ensure that its format and content was perfect in every way, the working group decided on a two step process:

1. Recommend that DIS 9945 is accepted in its current form as IS 9945-1. (The request to split the POSIX standard into multiple documents came as the draft standard was being balloted, with the result that its number has sprouted a −1.) ISO may decide to reprint the existing document, adding cover material to say that it is a standard. Alternatively, the standard may be published as a reference document: a few pages which tell the reader to go and look at a particular ANSI standard. (There is a precedent for this: the International Standards for COBOL and PL/1 simply point to ANSI documents.)

   If ISO accepts the recommendation, POSIX should become an International Standard within the next six months.

   ISO may turn down the request if it judges that the working group's plans to resolve outstanding issues are inadequate. Hopefully, this will not happen, because

2. The working group has undertaken to produce and ballot an amendment to the standard by 1st June, 1990. The amendment—actually 1003.1a produced by the IEEE—will fix all issues raised during the balloting of DIS 9945. What is more, the working group—or rather, the hard-pressed editor for the IEEE's POSIX project—will merge the addendum with the existing standard, producing a single document in a format acceptable to ISO. This, it is hoped, will be published as a revised standard late next year.

The Future of International Standards for POSIX

In my last report, I noted that the working group had requested that its project was split into several parts, resulting in several standards, numbered 9945-1, 9945-2 and so on, rather than a single standard 9945. This has happened, with the result that the operating system interface will be covered by 9945-1; shell and utilities by 9945-2, and system administration by 9945-3. No other numbers have yet been allocated. It is important to note that the apparent one-for-one correspondence between 1003.1 and 9945-1 will grow more tenuous as time goes on: facilities for real-time processing (1003.4), security control (1003.6) and transparent file access (1003.8) will be added to future versions of 9945-1. While 9945-2 corresponds to 1003.2, there is no connection between 1003.3 (Test Methods) and 9945-3. Instead, 9945-3—when it gets off the

ground—will be based on the IEEE's 1003.7 work.

I also mentioned last time that ISO standards are supposed to be independent of any particular computer language. 9945-1 will probably lose its ties to C with its second amendment (that is, the amendment after the one described in the previous section). This will introduce a need for a new standard to describe its C bindings, and further standards to describe bindings for Ada, FORTRAN, and so on. While the IEEE language bindings are part of the 1003 project (1003.5 for Ada, and 1003.9 for FORTRAN), ISO practice is to allocate a completely new standard number for bindings work. Consequently, a request for a new number, with three designated parts, has been made. We will not know this number until next June.

Table 1 summarises correspondence between ISO and IEEE standards.

TABLE 1. Correspondence between ISO and IEEE Activities

ISO	IEEE	Topic	Notes
9945-1	1003.1	OS interface	Now
	1003.1a	Clean-up	1990
	1003.1b	Extensions, language independence etc.	Future
	1003.4	Real-time	Future
	1003.6	Security	Future
	1003.8	Transparent file access	Future
9945-2	1003.2	Shell & tools	First release
	1003.2a	User Portability Extension	Future
9945-3	1003.7	System administration	First release
1xxxx-1	—	C bindings	Future (probably to be done by new 1003 working group)
1xxxx-2	1003.5	Ada bindings	Future
1xxxx-3	1003.9	FORTRAN bindings	Future
—	1003.0	POSIX environment	Some overlap with ISO DP 10000. *International Standardized Profiles*
—	1003.3	Test methods	Under consideration by rapporteur group
—	1003.8	(Aspects besides T.F.A.)	Work elsewhere in ISO on RPC
—	1003.10	Supercomputing	Profile: relevant to DP 10000
—	1003.11	Transaction Processing	Profile; also relevant to SC21/WG3 database work
—	1201.x	X Window	See below
—	1224.x	Interfaces to OSI services	Not clear where these fit in ISO work: SC21 (OSI) seems to be against working on bindings

A word about windowing is in order. Work in a number of JTC1 SCs nibbles at the edges of the issue:

SC2 (Code sets):

Encoding of pictures. There is no connection between this work and X's bitmap distribution format.

SC18 (Office systems):

> Office system user interface; Font and character information interchange (lots of this); page layout and document structure (even more of this)

SC22 (Languages):

> Form interface management system—a new project involving interactive screen forms and such

SC24 (Graphics):

> No work—even though SC24 looks like the obvious place to put windowing standardisation.

It is an article of faith that no international standard may encroach on another's territory, and that the terms of reference of each SC do not overlap. This presents difficulties in dealing with new (well, new in ISO terms) and widely-applicable technologies such as windowing. Perhaps it may be possible to hand the issue to SC24 without upsetting other SCs. Alternatively, it may be necessary for JTC1 to set up a whole new SC to run with it, and bring the currently fragmented work together. (This recently happened on security issues—see below.) Again, watch this space for more news.

9945-2 Shell and Tools Standard

The majestic machinery of JTC1/SC22 1has sanctioned the use of draft 9 of IEEE 1003.2 as a draft proposal (DP), which embarks forthwith on a six-month balloting period. This period is to be synchronised with the IEEE's ballot, with the result that 1003.2 and 9945-2 move forward in lock-step, and should hit the streets simultaneously as identical American and international standards.

Document Format

In order to avoid future wrangles over document format with ISO's Central Secretariat, and to avoid time wasted in recasting IEEE standards into ISO's mould, *all* 1003 standards are to be created and balloted in a format acceptable to ISO. (And to the IEEE. And to the POSIX working groups. But mostly to ISO.)

WG15 is concerned that ISO's standards for standards were drawn up with relatively short documents in mind. For example, ISO's Central Secretariat objects to the line numbers which appear in draft 13 of 1003.1—even though it used the line numbers in referencing other changes that

it wanted! Hopefully, an acceptable compromise will be reached. Working group chairs and editors will be told what the changes mean to them just as soon as a decision is reached.

Rapporteur Groups

The concept of rapporteur groups is an ISO invention. It refers to a group of "technical experts" (another ISO term) from a number of related standards efforts, or concerned with a specialised topic within a single standards effort, which meets to discuss its area of interest. Members of the group then report back to their own groups, in order to integrate the work of the rapporteur group and the standards efforts that it links.

WG15 has three rapporteur groups: Conformance, Internationalisation, and Security. Each addresses areas known to have applicability in fields broader than POSIX itself. For example, JTC1 has just created a whole new subgroup (SC27) to handle security, bringing together separate developments in SC18 (Office systems), SC20 (Data encryption), SC21 (Open Systems Interconnection), SC22 (Languages)* — and anything else which turns out to have security implications. (I mentioned this development in my last report, but managed to garble some of the references. Sorry about that...) Similarly, there is work on conformance testing and internationalisation both inside and outside ISO.

In Brussels, the rapporteur groups all held informal meetings separate from the main business of WG15. Since all three have only just got off the ground, there is little to report as yet, but watch this space!

X/Open Portability Guide as a European Standard?

At the May meeting of WG15, our minds were much exercised by a proposal from CEN (Comité Européen pour la Normalisation—The European Committee for Standardization) that the whole of the third edition of the X/Open Portability Guide (XPG3) should become a draft European prestandard. The arguments against doing this

---

*   Why is the POSIX project a subdivision of the languages subgroup? Because it was the least unsuitable place in the ISO structure to put it at the time...

centre on the fact that the XPG is not a formal standard reached (slowly) through consensus, but an informal document which references formal standards where it can, but which then goes on to fill the gaps with *de facto* and suggested standards material. Increasingly, the European countries which form CEN's membership have come to realise that a document of this type, while useful in its own right (arguably more useful than existing formal standards, in fact), cannot be adopted as a European standard for both legal and practical reasons.

XPG3 has, however, helped to focus European minds on areas where formal standards are lacking. At the moment it looks as though the CEN project charged with producing a POSIX standard will build on the output of WG15. In addition to this, Germany is in favour of adopting as prestandards those parts of XPG3 which do *not* correspond to existing or emerging international standards—for example, ISAM, curses and X Window. The argument for this is that some kind of standard is urgently needed in these areas. The argument against, coming from Britain, Denmark, the Netherlands and others, is that CEN can only adopt standards which are public—*de facto* just isn't good enough, and besides, such things are outside the scope of the original work-order for a POSIX standard. At the moment, it looks as if this point of view will prevail.

As a sidelight to this issue, it seems that ISAM will eventually make it into the POSIX standard, as X/Open has expressed a desire to submit a base document to the 1003.1 working group.

Harmonisation and Synchronisation

The three previous headings—9945-2, rapporteur groups, and the CEN standard for POSIX— highlight a couple of important issues identified by JTC1:

Harmonisation:
> Standards covering identical or related topics should be in agreement; and

Synchronisation:
> Development work on standards covering identical or related topics should be developed in step with one another, both so that there is no unnecessary delay between the appearance of one standard and the appearance of another, and to avoid duplication of work—for example, the same ballot objection being made to and fielded

by two separate groups.

WG15 has taken steps to synchronise its activities with those of the IEEE 1003 working groups, its main feeder. In some cases this means that WG15 will set IEEE timetables—almost a case of the tail wagging the dog, but necessary in order to arrive at international standards as quickly as possible.

To address the issue of harmonisation, WG15 discussed a new category of liaison to JTC1. Liaison is a mechanism which allows transnational and international setters and users of standards to monitor or to contribute to the work of ISO. Participation is otherwise the province of national standards bodies such as ANSI, JISC and DIN—ISO is currently bad at dealing with regional standards bodies such as CEN. The proposal embodied a combination of sticks and carrots which would allow other types of standards bodies to participate on condition that they undertook to align with relevant international standards within some reasonable time after publication. The working group reached no conclusion on this radical idea, and will discuss it again at its next meeting.

It will be a while before JTC1 gets around to considering any proposal of this nature. In the mean time, WG15 will continue to invite observers such as myself to its meetings.

Language Independence

As at the previous meeting, this topic was discussed at some length. The policy of JTC1 is that, ultimately, in the interests of precision and verifiability, all base standards should be written in some formal language which is itself the subject of an ISO standard. There is a small problem here: no formal language suitable for use in the POSIX project is yet the subject of a standardisation effort. (Although IEEE P1003.7, System Administration, is making use of ASN-1, a standardised formal language developed for use in describing communications systems.) If POSIX were to wait for a formal language to be standardised before breaking the current links between POSIX and C, nothing could be done for a couple of years. However, it is necessary to break the links with C as soon as possible, in order that additional bindings Ada and FORTRAN can be defined. The break will be made informally, by using English along with language-independent data types, and so on.

In parallel with this development by WG15, a research project funded by the European Community (EC) looks like funding the development of a description of the POSIX operating system interface in VDM-SL (Vienna Definition Method Specification Language). SC22 is actually thinking about standardising this formal language, which is already being used in the production of an ISO standard for Modula 2 by SC22/WG13. Welcoming what is, in effect, an offer to discover the problems involved in defining POSIX using a formal language, WG15 has sent a message of encouragement to the EC, while emphasising to SC22 that, as far as POSIX is concerned, the coming language-independent description is a necessary step on the path towards a formal definition.

The Portable Common Tools Environment

Another research project supported by the EC concerns the Portable Common Tools Environment, PCTE. Essentially a very sophisticated and all-encompassing object-based workbench for the support of Computer-Assisted Software Engineering (CASE), PCTE is the result of six years' work, and the investment of several million European Currency Units (ECUs) by government and industry—with more years and mega-ECUs to come. Among other organisations, NATO is a strong champion of the technology. The European Confederation of Computer Manufacturers (ECMA) has, over the last couple of years, been working on a PCTE standard which may (just) be ready in 1991, and which may then be offered to ISO.

What has this to do with POSIX? Well, PCTE was originally aggressively host-independent—independent, that is, both of hardware and, on systems where it was not to run native, of operating system. This made excellent sense six years ago when development started—using UNIX as a development host. Versions are currently available for several UNIX hosts, with VMS and IBM mainframe versions on the way. Times move on, however, and there is now (ISO Central Secretariat permitting) an international standard hardware-independent operating system which looks like becoming the predominant host for PCTE. It makes sense, therefore, for PCTE to align itself closely with POSIX, so avoiding unnecessary duplication or conflict of functionality. Following a morning of presentations by PCTE experts, WG15 agreed to

keep members of the ECMA PCTE working group informed of its activities.

Date of Next Meeting

The next meeting of WG15 is to be held in Paris, France from the 11th to the 15th of June, 1990, and is to be hosted by AFNOR, the French national standards body.

The OPEN LOOK™ Graphical User Interface

Gillian Mogg

gill@uel.uucp

UNIX Europe Limited (UEL)
International House
Ealing Broadway
London

Guest columnists in this issue are Ruth Klein and Bill Sherman.

Ruth Klien is the Supervisor of Systems Engineering for user interfaces at AT&T Bell Laboratories.

Bill Sherman is a Distinguished Member of Technical Staff, working on user interface development at AT&T Bell Laboratories.

User Interface Technology: The OPEN LOOK™ Graphical User Interface

The concept of a graphical user interface was originally pioneered at Xerox PARC in the early 1970's. The objective was to provide users with a more natural means of interacting with the computer system, using icons and visual metaphors to 'drive' the system. Although these concepts were originally developed over 15 years ago, it's just within the past few years that they've emerged successfully in the commercial market place. They first gained popularity for single tasking operating system environments such as the Apple Mactintosh™ System and later for the MS-DOS™ System. Recently, they've begun to be popular for the multi-tasking UNIX® Operating System environment.
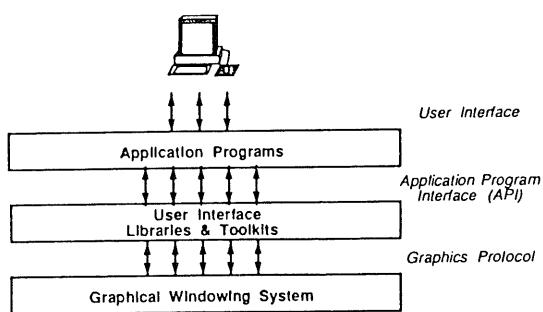
With the growing popularity of UNIX System V in the business market, a simplified user interface is considered to be a necessary element to gain acceptance by commercial end users. In response to this need, several user interfaces have emerged in the past year, and others are promised before year end 1989. Among these are the currently available OPEN LOOK and DECwindows™ User Interfaces ('UI'). And, the NextStep™ and OSF/Motif™ interfaces are targeted for availability later this year. This sudden proliferation of available and promised UNIX System user interfaces is creating an overwhelming situation, we describe the OPEN LOOK User Interface as an example of how the needs of this market can be met.

User Interface Software Architecture

The predominant architecture for UNIX System graphical user interface software is the client/server model. The client/server model gets its name from the fact that the window system has been taken out of the operating system and implemented as a user-level process, called a display server. 'Client' applications request graphics services of the display server via a byte stream protocol. Because of this simple interface, client applications may be distributed across machines throughout a local area network. In many client/server window systems (such as the X

Window System™), the window management functions are largely split out of the display server into a separate process, called a window manager. X and NeWS are examples of client/server window systems. There are 3 basic architectural levels shared by the client/server systems (Figure 1):

Figure 1. User Interface Architecture

- **The Graphical Windowing System:** This level provides the primitives for graphics operations as well as for windowing, i.e., displaying multiple applications, executing simultaneously, sharing a display device. Some of these systems, such as the X Window System, have no inherent user interface, and allow user interface software to be layered on top of it. OPEN LOOK UI, DECwindows XUI and OSF/Motif are all user interface systems that can be layered on top of X. X is the common platform choice since it provides the benefit of a standard and distributed architecture.

- **User Interface Libraries and Toolkits:** Toolkits simplify the task of writing applications that adhere to a specified user interface, and provide portability across platforms. They consist of a pre-defined user interface objects, such a menus, scroll-bars, buttons, etc. These objects are used by the application program to create the user interface. The programming interface used by the application developer is referred to as the Application Programmer Interface (API). The

APIs for the various X Window System based user interfaces vary, although many of them have a common substructure (to be described later). Toolkits communicate with the underlying X Window System using the X graphics protocol. This graphics protocol is the same, no matter which toolkit is being used.
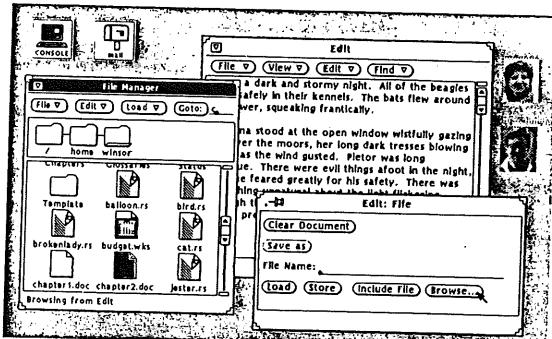
- **Application Programs:** Application programs written with a specific user interface Toolkit will present a consistent user interface to the end user. There is generally a User Interface Style Guide document that helps application designers to use the user interface elements in a common fashion. In addition to programs written by application developers, there are generally several system-level applications that come along with the user interface system. Minimally, this includes a Window Manager.

In this article, we'll describe:

- OPEN LOOK UI elements and design principles

- System-Level applications for a user interface environment.

- User Interface Toolkits and the OPEN LOOK API.

OPEN LOOK User Interface Elements and Design Principles
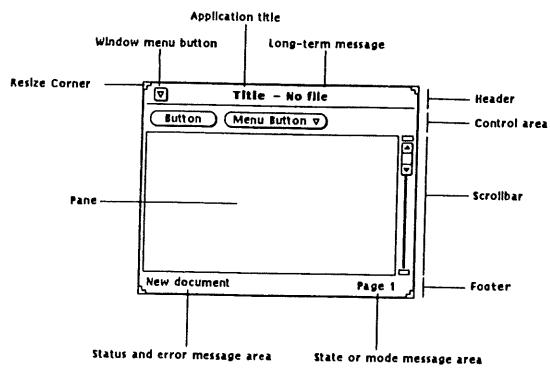
Features of the OPEN LOOK UI were designed by Sun Microsystems and AT&T, based on initial technology licensed from Xerox Corp. The overall design objective for the OPEN LOOK UI is to define a user interface that matches the computing environment needs of the next decade. That is, a user interface that meets the needs of evolving operating system and display device technology, yet provides interoperability with existing user interfaces prominent in the marketplace. We'll first describe the basic elements of the OPEN LOOK UI, and then discuss some of the design principles of the OPEN LOOK UI.

Figure 2. Typical OPEN LOOK UI Workspace

Figure 3. An OPEN LOOK UI Base Window

OPEN LOOK UI Elements
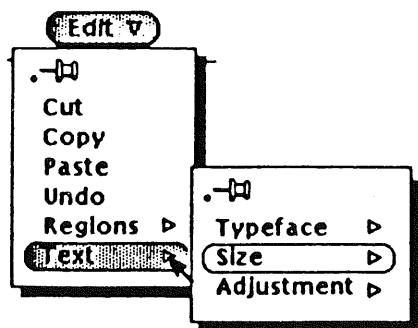
The screen display area in an OPEN LOOK UI environment is referred to as the Workspace (see Figure 2). Windows and icons are displayed on the Workspace. Windows are used to display applications to the user, and allow the user to interact with applications. Icons represent objects that the user can act upon, e.g., files or other applications. Multiple windows and icons can appear on the Workspace, and a mouse pointing device is used to select and operate on these objects. There are three mouse functions. SELECT selects objects and commands, ADJUST adjusts a selection, i.e., adds more objects to a set of selected objects, or toggles off selected objects from a set. MENU displays a menu associated with an object. We'll use the terms SELECT, ADJUST, and MENU to represent depressing the respective mouse button. Although there are three basic mouse button functions, these operations can be executed on systems with one, two or three button mice. On the OPEN LOOK UI Workspace, applications appear within a Base Window. These Base Windows have distinctive visual elements, (see Figure 3), including:

- **Resize Corners**, like pictures mounts, in the four corners of the window, used to resize the window.

- At the top of the window, the **Window Header**, which contains information about the application, and the Window Menu Button. The Window Menu Button is an accelerator for quickly executing a default window operation, such as closing a window into an icon.

- Below the Window Header is the **Control Area** with buttons that are used to 'drive' the applications. These buttons represent application specific operations. The user can activate an operation (depress SELECT on a button) or display a menu (depress MENU on a button). Buttons that have menus associated with them are designated by a triangle in the button.

- Below the Control Area is the window **Pane** which is the application display area. The OPEN LOOK UI window layout allows an application to have multiple control areas and multiple window panes. For example, a mail application which could have one control area and pane for incoming mail, and another set for outgoing mail.

- If there is not enough space in the window Pane to display the application data, then Scrollbars appear (either horizontal or vertical scrollbars). The OPEN LOOK UI scrollbar is visually modeled after an elevator, i.e., an elevator car riding on a cable with upper and lower cable anchors.

- At the bottom of the window, a **Footer** area for display of messages.

Figure 4. A Cascading Menu

When working with an application in a Base Window, the user gives commands to the application by operating on the buttons in the Control Area. The user can SELECT a button for immediate execution of a function, or can choose from a menu of subfunctions by doing a MENU operation on the button. Menus can 'cascade' into submenus, as in Figure 4. This example shows an Edit button, the menu for the Edit button, and a cascaded submenu. One thing worth noting in this example is the the menu and submenu each show one item with a ring around it. This ring designates the default choice for the menu and submenu. The ability to designate defaults simplifies the learning process for users inexperienced with a given application. Also, the OPEN LOOK UI provides accelerators to allow experienced users to quickly execute the default operation without having to cascade through the sequence of submenus.
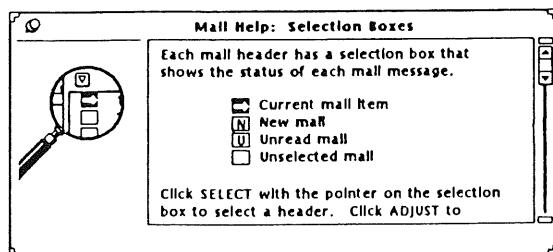
In an OPEN LOOK UI environment, menus are not just associated with buttons in a Control Area, but may be associated with any object on the workspace. To see the menu, the user depresses the MENU mouse button while the mouse pointer is over the object. These menus are referred to as 'pop-up' menus, simply because they 'pop-up' on the display wherever the mouse pointer is positioned. For example, a Drawing application could have a pop-up menu that allows users to

duplicate or delete a specific object. Pop-up menus minimise the mouse movement required for a given operation, which makes the interface more efficient for experienced users.

Once the user has chosen an operation from a Control Area Button or a menu, the application may require additional information to complete the operation. In such cases, the application can display a pop-up window to the user, requesting additional information needed to execute the command. These are referred to as **Command Windows**. For example, an application could display a command window to request that the user type in a file name for a save operation.
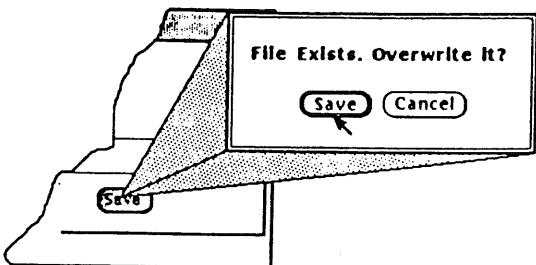
In addition to the command window, the OPEN LOOK UI provides three additional types of pop-up windows: Help Windows, Notice Windows, and Property Windows.

Help Windows display help information, as in Figure 5. The OPEN LOOK UI provides context sensitive help. The user places the mouse pointer over an item on the Workspace and depresses a Help Key on the keyboard. The Help Window pops up and the magnifying glass visual displays the object for which help has been requested.

Figure 5. An OPEN LOOK UI Help Window
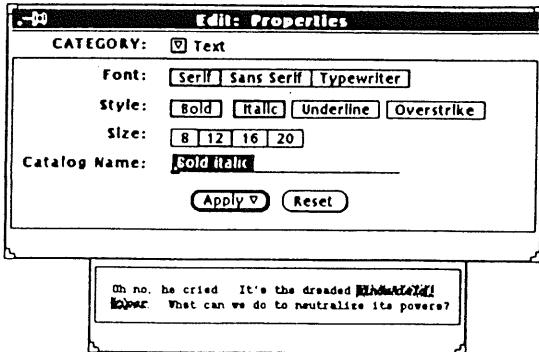
Notice Windows provide warning information to the user, as in Figure 6. In addition to the warning information, Notice Windows provide options for what to do next. In this example, the Save operation is designated as the default. As with menu items, the default is designated by a ring around the item. In fact, the ring visual is used throughout the OPEN LOOK UI to designate the default among a set of options.

Figure 6. An OPEN LOOK UI Notice Window

Property Windows set characteristics for objects. All objects in an OPEN LOOK UI environment can have properties associated with them—the Workspace, application windows, and objects within an application's Pane. Figure 7 shows a Property Window for setting text characteristics. OPEN LOOK UI Property Windows provide a consistent method for setting object characteristics. Most other user interfaces do not provide such a capability, and as a result users need to hunt around to find out how a specific application's characteristics are set.

One thing to note about pop-up windows is that they are transient. That is, they pop-up to perform a specific operation., and then disappear. The OPEN LOOK UI provides a mechanism for the user to control these pop-up windows. With the exception of the Notice Window, all OPEN LOOK UI pop-up windows have the Pushpin visual in the upper left hand corner of the window. The user can SELECT the pushpin and 'pin' the pop-up window onto the workspace, similar to pinning an item onto a bulletin board. Once pinned, the pop-up window is no longer transient, but will stay up for multiple operations. This is another of the efficiency characteristics of the OPEN LOOK UI, freeing the user from going through a repetitive sequence of operations to get to the function they want to perform. For example, the Property Window shown in Figure 7 can be pinned to the Workspace and then used to set characteristics for multiple text objects. The user does not have to go through repetitive command sequences to bring up the pop-up window for each text object. In addition to appearing on pop-up windows, pushpins may appear on menus, as seen previously in Figure 4.

Figure 7. Property Window with Pushpin

Thus far, we've described how users control applications in the OPEN LOOK UI environment by operating on buttons in a Control Area. And, how these operations could result in the appearance of menus to choose subfunctions, or the appearance of pop-up windows to provide additional input needed to complete an operation. Although buttons are the most prevalent type of control element, the OPEN LOOK UI provides a wide variety of Controls. Controls are user interface elements which can appear within application Control Areas or pop-up windows, and provide users with distinctive visuals for different types of input. A sample set of OPEN LOOK UI Controls appears in Figure 8. These Controls include visuals such as Exclusive settings (letting a user know when they can select only one item from a list), and Non exclusive settings (letting the user know when they can select multiple items from a list), check boxes, gauges, etc.

EUUG

Buttons	(Eagle) (Hawk...)
Menu buttons	(Insects ▽) (Marsupials ▷) (Platypus) Kangaroo Koala Opossom
Items	Command Item / Window Item... / Menu Item ▷
Abbreviated buttons	▣ ▣
Abbreviated menu buttons	Marsupials: ▽ Platypus
Text field	
Text field with scrolling buttons	◁me▷
Multi-line text field	
Numeric field with increment/ decrement buttons	1234 ▲▼
Exclusive settings	French German English
Nonexclusive settings	Aquarium Displays: Minnow / Whale / Dolphin / Tuna
Check boxes	☐ Shoes / ☑ Socks
Slider	
Gauge	Disk Usage: 0 ... 100
Read-only message	Size: 1997 bytes

Figure 8. Some OPEN LOOK UI Controls

OPEN LOOK UI Design Principles

The design principles for the OPEN LOOK UI environment can be summarised as:

- Simplicity, consistency and efficiency of the user interface

- A thorough user interface specification document that unambiguously defines the elements of the OPEN LOOK UI.

- Portability and consistency across hardware platforms.

- A rich repertoire of visual elements and user interface functionality resulting in a more efficient application development process.

- Interoperability with other prominent user interface environments.

- Free from legal encumbrance.

The need for simplicity, consistency and efficiency in the user interface design is self-

evident. These criteria were used throughout the OPEN LOOK UI design process and manifested themselves in characteristics such as an uncluttered visual appearance; the use of real world metaphors as visual objects (e.g., buttons, elevators, scrollbars, pushpins); context sensitive Help; a consistent method for setting Properties for objects, applications and the overall Workspace; the existence of defaults and consistent visual distinction of defaults; and the efficiency characteristics provided by pop-up menus, pushpins, and accelerators. One other significant consistency characteristic is the OPEN LOOK UI concept of input virtualisation. In an OPEN LOOK UI programming environment, inputs via the keyboard and mouse are defined by their function. The user can tailor their environment by re-mapping these functions using the Workspace Properties Window. For example, a left-handed user can re-assign the mouse functions of SELECT, ADJUST and MENU to whichever mouse buttons (or even keyboard keys) are most comfortable for them. Since the OPEN LOOK UI toolkits and programming interface deal with these as functions, not explicit mouse button numbers, all OPEN LOOK UI applications run by that user then respond appropriately to the re-mapped functions. No programming intervention is required.

Another key characteristic of the OPEN LOOK UI design process was the existence of a thorough user interface specification. This allowed the user interface to be designed independently of any specific hardware or operating system base. The OPEN LOOK User Interface Specification document describes the elements of the user interface and functionality to be provided by toolkits. The Specification document was distributed for industry review in July 1988, and has resulted in valuable industry feedback which has been incorporated into the May 1989 final draft. The existence of the Specification document facilitates portability and consistency of the user interface across various hardware platforms. For example, all visual elements are described in a device independent manner, as engineering drawings. To our knowledge, the OPEN LOOK UI is the only major user interface in the market which has a publicly available specification document.

The OPEN LOOK UI Specification defines a rich repertoire of user interface functionality and associated visual elements. This was motivated by

the desire to have a consistent user interface that was applicable to a broad spectrum of applications. The availability of this rich set of user interface elements in the programming environment results in the reduction of time for applications development.

Another of the OPEN LOOK UI design principles was interoperability with other prominent user interface environments. By 'interoperability', we mean the positive transfer of learned experience from one user to interface environment to another, so that users don't make drastic or frustrating errors. This required insuring that none of the elements of the OPEN LOOK UI conflicted with the two existing popular user interface environments in the market today.

Given the 'Look and Feel' lawsuits in the industry today, another of the design criteria was to make the user interface free from legal encumbrance. The OPEN LOOK UI is based on initial technology licensed from Xerox, and the specification was under legal review throughout the design process.

System-Level Applications

In and of itself, the specification of user interface elements alone do not provide an integrated user interface environment to the end user. There are basic system-level functions that users expect to perform as part of the overall user interface environment. And, users expect these functions to be consistent across various hardware platforms (i.e., multiple vendor environment). For this reason, the OPEN LOOK UI Specification document includes definition of the following system-level applications.

- **Workspace manager**: the workspace manager maintains the properties and state information of a user's environment between login sessions. Such properties include workspace colours, favourite programs list, where windows are placed, etc. The OPEN LOOK UI workspace manger is the first client application to execute, and it spawns other applications that the user desires to run.

- **Window manager**: The Window Manager controls the placement and positioning of windows on the Workspace. The Window Manager offers a window menu and controls (often called 'decorations') around the window border to allow the user to interact with the window manager. In addition, for the X Window System platform, the Window

Manager is compliant with MIT's Inter-Client Communications Convention Manual (ICCCM). The ICCCM ensures that client applications from various environments communicate with window managers (and each other) in a consistent fashion. This is extremely important to software vendors who may be writing applications that interface to applications that interface to applications developed elsewhere.
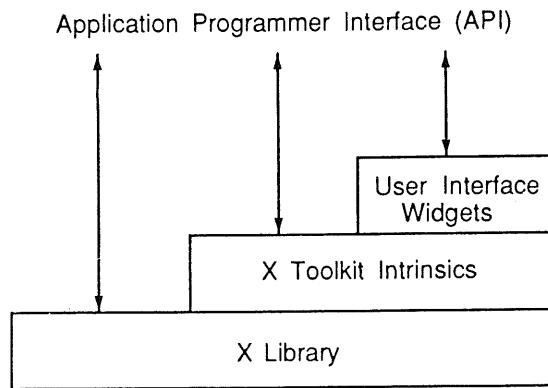
- **File manager**: The rôle of a file manager is to allow users to view the file system and in applications graphically. Simple operations such as displaying, editing, copying and printing a file are typically accomplished by dragging icons around the work space with a pointing device.

The incorporation of basic functionality for these system level applications into the OPEN LOOK UI Specification is the direct result of the industry review process. End-users considered these system level functions to be integral elements of the overall user interface environment. The specification of basic functionality for these system level applications is among the differentiating factors in the OPEN LOOK UI design. This does not preclude the development of 'value-added' system-level applications that are not only OPEN LOOK UI compliant but have additional functionality (such as the ability to handle dynamic data pipes) as well.

Toolkits and APIs

User interface toolkits provide a set of building modules (library routines) for application programmers, freeing them from having to understand the window system details, and allowing them to concentrate on developing the application itself. There are several Toolkits that provide the OPEN LOOK UI. They vary based on the underlying graphical windowing system technology and specific customer base needs, e.g., the need to migrate an existing graphics customer base to the OPEN LOOK UI environment. In this article, we'll focus only on the OPEN LOOK UI Toolkit, commonly referred to as XT+, developed by AT&T.

The OPEN LOOK XT+ Toolkit is implemented as a layer on top of the X Window System Intrinsics. This means that the Application Programmer Interface (API) has three layers (Figure 9).

Application Programmer Interface (API)

Figure 9. X Toolkit Architecture

- The X Library (Xlib) is a C-language procedural interface to the X protocol providing access to the low-level drawing and input-event operations.

- The Toolkit Intrinsics provide a common substructure and programming style for managing the object-oriented user interface environment. For example, all widgets are created and destroyed with calls to Intrinsics routines, regardless of which widget set is being used.

- The User Interface Widgets are the actual objects for the specified user interface.

These three levels make up the complete user interface Application Programmer Interface (API). Many applications will deal with only with the Widget-level API. Other applications may choose to interface with the Xlib layer for graphics drawing functions or with the Toolkit Intrinsics layer to create new widgets for an application or set of applications.

The OPEN LOOK XT+ Toolkit is implemented as a widget layer on top of the X Window System Toolkit Intrinsics. In fact, several of the user interface systems are implemented in this fashion. DECwindows is implemented as a widget layer, as will OSF/Motif. All three user interface systems have a common substructure in the Toolkit Intrinsics layer. And, this Toolkit Intrinsics layer mandates a new, common, programming style for

applications development. When using any of the Intrinsics based user interface systems, a more rigorous division between application code and user interface code is enforced. In fact, it can be said that the user interface 'drives' the application. That is, an application function is executed as a result of a specific user interface operation. The Toolkit informs the application specific code that an operation has been performed via the use of callback routines. The applications programming style therefore typically consists of:

- First, laying out the user interface aspects of the application, i.e., which widgets will be used, and what application functions will be activated. For example, if you were building a spreadsheet application, you would lay out all of the menus, forms, scroll-bars, and cell arrangements first.

- Second, building up the application by filling in the callback routines. These define the application functions to be executed in response to user interface actions. For example, in the case of the spreadsheet application, the callback routines would include the logic for how cells interact, how fields are tallied, etc.

Of course, this is oversimplifying the design process for a complex spreadsheet application, but it provides a flavour of the object-oriented programming style enforced by the Toolkit Intrinsics. This object-oriented style is common to programming with the OPEN LOOK XT+ Toolkit, the DECwindows XUI Toolkit, and will be common for the OSF/Motif Toolkit. This programming style is new to many application developers and requires a reorientation of the typical program design process. It takes a while to learn and become proficient in this new programming style, although this rigorous separation of user interface from application semantics usually results in cleaner and more maintainable code. Our recommendation to application developers is to begin this learning process. Use one of the currently available toolkits, e.g., the OPEN LOOK XT+ Toolkit, and become familiar with the X and Toolkit Intrinsics programming environment.

The top layer of the Intrinsics-based APIs is the widget level. The programmer interfaces to the widget level differ among the various Intrinsics-based toolkits. The widget level embodies the

specific user interface objects and programming parameters for these objects. The widgets available with the OPEN LOOK XT+ Toolkit are shown in Figure 10.

Primitive Widgets:	Complex Widgets:
CheckBox	AbbrevButtonStack
OblongButton	ButtonStack
RectButton	Menu
ScrollBar	ScrollingList
Slider	Notice
Static Text	PopupWindow
Text	
TextField	**Convenience Routines:**
Composite Widgets:	OlMMToPixel
	OlScreenMMToPixel
BulletinBoard	OlPointToPixel
Caption	OlScreenPointToPixel
ControlArea	OlPixelToMM
Exclusives	OlScreenPixelToMM
FooterPanel	OlPixelToPoint
Form	OlScreenPixelToPoint
NonExclusives	OlRegisterHelp
ScrolledWindow	

Figure 10. OPEN LOOK XT+ Widgets & Convenience Routines.

It is worth noting that the major design principles of the user interface specification are also design principles for the XT+ Toolkit implementations. Among the differentiating factors here are:

- **Device independence.** The toolkit provides convenience routines that allow for implementation of display resolution independent objects. In addition, the currently available OPEN LOOK UI fonts and bitmaps support execution on displays that cover the range from about 50 to just under 100 DPI (Dots Per Inch), in aspect ratios of 1:1. 5:6, and 3:4.

- **Ease of application implementation.** One of the direct implications of having a fully defined user interface specification document, is that the Toolkit widgets are rich in

functionality. The user interface style is implemented in the toolkit, requiring fewer callback routines to the application. This makes it easier for software developers to implement in the OPEN LOOK UI environment.

Summary

We've focused on describing characteristics of the OPEN LOOK UI as an example of current graphical user interface technology. Specifically, the basic user interface elements, system-level applications, and the XT+ Toolkit implementation were described. Our intent has been to provide useful information to aid in the decision of which user interface is appropriate for your environment.

References

AT&T and Sun Microsystems, Inc.,
OPEN LOOK Graphical User Interface Functional Specification Release 1.0.
May 1, 1989.

Klein, Ruth and Sherman, William,
Is Yet Another User Interface Really Necessary? How OPEN LOOK Got that Way.
UNIX Technology Advisor, Volume 1 Number 3. May, 1989.

Rosenthal, David S.H.,
X Window System, Version 11 Inter-Client Communications Conventions Manual. May 9, 1989.

Varhol, Peter D.,
OPEN LOOK.
MIPS Magazine, Volume 1 Number 7, July 1989.

For further information on the contents of this column, please contact Gillian Mogg <*gill@uel.uucp*>.

A Directory of Electronic Mail !%@:: Addressing and Networking
Book Review

A Directory of Electronic Mail !%@:: Addressing and Networking, Donnalyn Frey and Rick Adams, A Nutshell Handbook, O'Reilly & Associates, Inc, 1989, ISBN 0-937175-93-0 (USA), Paper Back, 286 Pages pp.
Reviewed by Peter Houlder of University of Kent
uknet@ukc.ac.uk

This is a book for people, who fail to understand how people survive without electronic mail (email). It is the first book that endeavours to explain how all the diverse and ever-growing email networks interrelate.

The book starts with a Preface outlining how to use the book. This should be read carefully. It is followed by the first chapter adapted, courtesy of Daniel Karrenberg and Anke Goos, from the original in the European R & D E-mail Directory, published by the EUUG. This "User Introduction to Electronic Mail" is a valuable guide to the fundamentals of how email works.

The second chapter, all 207 pages of it, covers over 100 major world networks. Each entry is for a single network and is clearly and consistently structured. The first thing I like about each network entry is the accompanying map, showing the scope of the network and, where appropriate, its links to other networks. The text is on a separate page opposite the accompanying map. The text contains: *a description of the network, address structure and format, architecture, connections to other networks or sites, facilities available to users, contact, cross references to other networks, future plans and the date that*

information was last updated. The last sentence is courtesy of the back of the book. Unfortunately it is not yet true. This however is not intended as a criticism, as the book is intended to be regularly updated and the information template will undoubtedly become more complete with each update. The lack of a certain completeness of information does not however detract from the book's worth.

On a personal level the book represents my first real job threat. As a network administrator, the formal specification of such whispered folklore is worrying; the user might actually be able to solve their own problems. The book is only 2 chapters long, but it also contains 5 appendices, a glossary and 3 sets of indices. These sections are an essential part of the book's structure. How else would you know that CINECA was an Italian network, or that the main South Korean network was called SDN.

As a summary I would highly recommend this book to all serious email users. It is a reference book, but it also contains sections, that intentional or not, are amusing. The Addressing, Architecture and Future Plans for South Africa are all described by one word unknown, no other entry is so incomplete ....

At present the book is only available over the counter in the USA, although it will soon be published directly in Europe. In the meantime it can be obtained from the EUUG, Owles Hall, Buntingford, Hertfordshire SG9 9PL, United Kingdom, price on application.

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

year mon days	conference	(sponsor,) (hotel,) location
1989 Dec 5-6	JUS UNIX Fair 89	Tokyo, Japan
1989 Dec 6-8	Sun UG C	Hilton, Anaheim, CA, USA
1989 Dec 8-9	Sinix UNIX Asia'89 C	World Trade Center, Singapore
1989 Dec 11-15	OSI Implementors W	NIST, G, MD
1989 Dec 11-13	UKUUG C	Cardiff, Wales, UK
1990 Jan	U in Gov. C&T	Ottawa, ON, Canada
1990 Jan 8-12	IEEE 1003	New Orleans, LA
1990 Jan 9-10	U in Gov. C&T	Ottawa, ON, Canada
1990 Jan 20-26	DECUS S	Toronto, Canada
1990 Jan 22-26	USENIX	Omni Shoreham, Washington, DC, USA
1990 Jan 23-26	UniForum	Washington Hilton, Washington, DC, USA
1990 Jan 29	IEEE 1003	New Orleans, LA, USA
1990 Feb 6-8	IETF	IAB, (FSU, Talahassee, FL), USA
1990 Feb 14	UKUUG W System Admin	Inst of Education, London, UK
1990 Mar 5-6	X3J11	New York City, NY, USA
1990 Mar 26-28	USING C	Dallas, Texas, USA
1990 Mar 26-29	DECUS S	Vasteras, Sweden
1990 Mar 27-30	AFUU C	Paris, France
1990 Apr	IEEE 1003	Montreal, Quebec, Canada
1989 Apr 9	POSIX APP W	NIST, G, MD, USA
1990 Apr 9-11	USENIX C++ Converence	San Francisco, CA
1990 Apr 23-27	EUUG	Munich, Germany
1990 Apr 23-27	IEEE 1003	Salt Lake City, UT
1990 May	U 8x/etc C&T	/usr/group/cdn, Toronto, ON, Canada
1990 May 2-4	IETF	IAB, (U. Washington, Seattle, WA), USA
1990 May 7-11	DECUS S	New Orleans, Louisiana, USA

EUUG

1990 May 17	NLUUG C	Unix & Parallel Systems, Ede, Netherlands
1990 May 30-Jun 1	UNIX/90 C&T	/usr/group/cdn, Toronto, ON
1990 Jun 11-15	USENIX	Marriott Hotel, Anaheim, CA, USA
1990 Jul 9-11	15th JUS S	JUS, Tokyo. Japan
1990 Jul 9-13	UKUUG C	London, UK
1990 Jul 16-20	IEEE 1003	Danvers, MA
1990 Jul 31-Aug 2	IETF	IAB, ?, not in North America
1990 Sept 25-28	AUUG Conference	Southern Cross, Melbourne, Australia
1990 Autumn	USENIX C Software Development	Honolulu
1990 Autumn	USENIX W -MACH	Northeast USA
1990 Oct 22-26	EUUG	Nice, France
1990 Oct 31-Nov 2	UNIX EXPO	New York, NY
1990 Nov 5-9	10th Internat'l C on CC	ICCC, New Delhi, India
1990 Nov 8	NLUUG C	Open Systems, Ede, Netherlands
1990 Nov 15	POSIX APP W	NIST, G, MD, USA
1990 Dec 4-5	JUS UNIX Fair '90	JUS, Tokyo, Japan
1990 Dec 10-14	DECUS S	Las Vegas, NV, USA
1991 Jan 21-25	USENIX	Grand Kempinski, Dallas, TX, USA
1991 Jan 22-25	UniForum	Infomart, Dallas, TX, USA
1991 Feb	U in Government C&T	Ottawa, ON, Canada
1991 Feb 18-22	DECUS S	Ottawa, Canada
1991 May	U 8x/etc C&T	/usr/group/cdn; Toronto, ON, Canada
1991 May 6-10	DECUS S	Atlanta, GA, USA
1991 May 20-24	EUUG	Tromso, Norway
1991 Jun 10-14	USENIX	Opryland, Nashville, TN, USA
1991 Sept 16-20	EUUG	Budapest, Hungary
1991 Dec 9-13	DECUS S	Anaheim, CA, USA
1992 Jan 20-24	USENIX	Hilton Square, San Francisco, CA, USA
1992 Jan 21-24	UniForum	Moscone Center, San Francisco, CA, USA
1992 Spring	EUUG	Jersey, UK
1992 May 4-8	DECUS S	Atlanta, GA, USA
1992 Jun 8-12	USENIX	Marriott, San Antonio, TX, USA
1992 Autumn	EUUG	Amsterdam, Netherlands
1993 Jan	USENIX	Town & Country, San Diego, CA, USA
1993 Mar 2-4	UniForum	Washington, DC, USA
1993 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

EUUG

/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

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

Puzzle Corner

Mick Farmer
mick@cs.bbk.ac.uk

Hello peeps,

Solution to Puzzle Number 5

Arithmetically, this puzzle presents no difficulty. B gains twelve hours and C loses twelve hours, both at the rate of one minute per day. This is achieved after 720 days when A is indicating twelve noon on the 720th day from 1st April 1898. Now 1900 was not a leap year so the date was noon on 22nd March, 1900.

Note that we get a different answer if the year had been 1998 because the year 2000 *is* a leap year!

Solution to Puzzle Number 6

The majority of solutions received followed the form that 1 expected, invoking the C compiler as `cc -D'program=/*'` .... Typical of these was the program from jomp@uucp.mentor. The Pascal program contains the C program as a trailing comment. The C program contains the Pascal program as a leading comment, provided we can invoke the preprocessor:

```
program hybrid(output);
begin
    writeln('Hello Niklaus!');
end. {*/

int main(void)
{
    printf("Hello Dennis!\n");
}
```

However, what a delight to receive a solution from ch@uucp.dce which didn't require such trickery! This solution uses the fact that the construct (*... is valid C and also the alternative start of comment token in Pascal. It's all in the first line:

```
(*hello)();  /*)

program wirth(output);
begin
    writeln('Hello, Niklaus!');
end. {*/

int main(void)
{
    printf("Hello, Dennis!\n");
}
```

Solution to Puzzle Number 7

The solution obviously involves a piece of C code together with a copy of this C code stored as an array of strings. The difficulty involves a representation of the double quote character. The simplest solution I've got is given below, though I'd like to receive other solutions that you think are better.

```
#define QUOTE '"'
#include <stdio.h>

int main(void)
{
    int i, j;
    extern char *s[];
    s[0][15] = QUOTE;
    for (i = 0; *s[i] != '$'; i++)
        puts(s[i]);
    s[0][15] = '$';
    for (j = 0; s[j]; j++) {
        putchar(QUOTE);
        fputs(s[j], stdout);
        putchar(QUOTE);
        putchar(',');
        puts(s[2]);
    }
    for (i++; s[i]; i++)
        puts(s[i]);
}
```

```
char *s[] = (
"#define QUOTE '$'",
"#include <stdio.h>",
"",
"int main(void)",
"{",
"    int i, j;",
"    extern char *s[];",
"    s[0][15] = QUOTE;",
"    for (i = 0; *s[i] != '$'; i++)",
"        puts(s[i]);",
"    s[0][15] = '$';",
"    for (j = 0; s[j]; j++) {",
"        putchar(QUOTE);",
"        fputs(s[j], stdout);",
"        putchar(QUOTE);",
"        putchar(',');",
"        puts(s[2]);",
"    }",
"    for (i++; s[i]; i++)",
"        puts(s[i]);",
"}",
"",
"char *s[] = {",
"$",
"NULL };",
NULL };
```

Puzzle Number 8

The idea underlying the next two 'faded documents' puzzles is that an ancient document has been found containing an arithmetic problem. Only a few figures are still recognizable together with faint marks where other figures have been. The first is the classical *Seven sevens* first published in 1906. Reconstruct the division.

```
                    * * 7 * *
                 -------------------
* * * * 7 * ) * * 7 * * * * * * *
             * * * * * *
             -----------
             * * * * * 7 *
             * * * * * * * *
             -------------
               * 7 * * * *
               * 7 * * * *
               -----------
               * * * * * * *
               * * * * 7 * *
               -----------
                 * * * * * *
                 * * * * * *
                 -----------
                          0
```

Puzzle Number 9

This problem consists of two multiplication skeletons, with no figures whatsoever. The first represents the multiplication of a number by itself, the second the multiplication of the resulting square by the first number to produce the cube of the first number.

```
      * * * *
      * * * *
      -------
    * * * * *
    * * * *
  * * * *
* * * *
- --------------
* * * * * * *
```

```
        * * * *
* * * * * * * *
------------
        * * * *
      * * * *
    * * * *
  * * * *
* * * * *
* * * * *
-------------------
* * * * * * * * * *
```

Loads-a-puzzles,

Mick

EUUG Autumn Conference Abstracts

Here are the abstracts of the papers delivered at the EUUG Autumn conference held in Vienna this year.

Copies of the proceedings are available from Owles Hall at £20 each including post and packing (there is an order form elsewhere in this newsletter);.

Thanks are due to Stuart McRoberts <sm@ic.doc.ac.uk> and Jan-Simon Pendry <jsp@ic.doc.ac.uk> who organised the typesetting.

Are Standards the Answer?

Dominic Dunlop

The Standard Answer Ltd.
4, St. Rumbolds Road
Wallingford
OXO.OX10 0DL
UK
domo@sphinx.co.uk

Moves are afoot to standardize every aspect of the UNIX world in order that the benefits of open systems can be realised. But what *needs* to be standardized? 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 standardization 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.

Engineering a (Multiprocessor) UNIX Kernel

Michael H. Kelley
Andrew R. Huber

Data General Corporation
Research Triangle Park.
North Carolina 27709
USA
kelleymh@dg-rtp.dg.com
huber@dg-rtp.dg.com

This paper describes the software engineering aspects of the DG/UX™ kernel, a symmetric multiprocessor UNIX kernel that runs on the Data General AViiON™ and ECLIPSE™ MV families of computers. The DG/UX kernel is a completely new implementation (not based on AT&T or Berkeley source code) that has applied modern software engineering techniques to improve the structure and modularity of the code. The result is a kernel that is more reliable, easier to maintain, and easier to enhance.

The paper discusses four major themes that have guided the engineering of the kernel to achieve the reliability, maintainability, and enhanceability goal. It describes how the themes of *hierarchy*, *information hiding*, *conventions*, and *tools* have been applied throughout the kernel and implemented using the standard C language and UNIX development environment. Finally some practical experiences in using this method to produce the finished DG/UX product are described.

An Implementation of STREAMS for a Symmetric Multiprocessor UNIX Kernel

Philippe Bernadat

O.S.F. Research Institute
c/o Bull-Imag†
2 Avenue Vignate, Z.I. Mayencin
38610 Gieres
France
bernadat@gu.bull.fr

This paper describes a solution to running UNIX Streams in a multiprocessor environment. As for other layers of the UNIX kernel, every part of the streams code and every stream module can be executed on any CPU. The design is adaptable to various kernel implementations and only relies on the streams message mechanism.

Developing Writing Tools for UNIX Workstations

Martin D. Beer, Steven M. George and Roy Rada

Department of Computer Science,
University of Liverpool.
P. O. Box 147,
LIVERPOOL
L69 3BX.
UK
mdb@mva.cs.liv.ac.uk

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 re-implement 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.

Implementation of a Window Manager under X11R3

Hans-Joachim Brede
Nicolai Josuttis
Achim Lörke

BREDEX GmbH
D-3300 Braunschweig
nico@bredex.uucp

A window manager in the X11 environment is responsible for the appearance of all X applications. It does the placement (moving, resizing, stacking etc.) of windows and can add some "decoration" (titlebars) to the top-level X windows. Additionally, it will enforce some policy on iconifying windows.

After a brief overview over the duty of a window manager and the responsibility of conforming clients, we will describe the design and implementation of a specific window manager. The implementation follows the guidelines given in the "Inter-Client Communication Conventions Manual" [ICCCM].

Special effort will be given to some of the less trivial aspects of a window manager. This includes discussion of colormap policy, icon concepts and some special X requests (synthetic events, save-set handling and additional properties).

The whole window manager is part of a bigger system that emulates an existing window system for SINIX† called COLLAGE‡ This system was realized using X toolkit widgets, and therefore the window manager is integrated in the toolkit environment. Problems that arose from this context are those with the reparenting of windows and the handling of events regarding the root window.

Teaching a Spreadsheet How to Access Big Databases

Michael Haberler
Martin Ertl

Hewlett-Packard Austria, Vienna
Technical University Vienna
mah@hpuviea.at
martin@hpuviea.at

While PCs are becoming commodity items, few attempts have been made to tap their processing power for *cooperative programs* in typical business applications. We view a cooperative program as set of distributed communicating processes with specialized tasks. In the context of a typical data entry/lookup application, the PC "process" might provide the user interface while a process running on a mainframe is responsible for database access. This might result in a fast, responsive user interface while lowering the mainframe and communications requirements. To explore the feasibility of this approach, we have built a sample cooperative program: a PC spreadsheet which can access a relational database running on a UNIX host. We describe different approaches to problem partitioning as well as our experience with using Sun Remote Procedure Call and BSD sockets as a programming interface for cooperative programs.

---

† SINIX is a trademark of Siemens AG, München.

‡ COLLAGE is a trademark of Siemens AG, München.

System Security – Administration Through Automation

Dale A. Moir

Lachman Associates, Inc.
1901 N. Naper Boulevard
Naperville, IL 60540
USA
dale@laidbak.uucp

A suggested approach to implementing an automated computer security policy is presented. Policy considerations ranging from physical security to file permissions are discussed in detail. In each case, methods of integrating the security policy considerations into an automated procedure are described. The cost effectiveness of an automated policy is defined in terms of user education, administrator training, and relative advantages over alternative methods. A brief section on disaster recovery is also included, as the same approach to automation may be applied in this area as well. Finally, the resultant computing environment, with full security measures in place, is described from the user and administrator perspectives.

Lettermatrix for the Selection of Passwords Through the User

Ernst Piller

BULL Austria
Linke Wienzeile 192
A-1150 Vienna, Austria
piller@atbull.at

In UNIX a user authenticates himself by entering a secret password. Most users choose simple passwords (short words and/or words with a special meaning), which are changed frequently. The result is a lack of security. In this paper, a new method for the selection of passwords by the user is beeing introduced. The method is based on a usual (traditional) secret password and on a lettermatrix, which is displayed on the screen of the user. The user has to find his way through the lettermatrix, based on his usual secret password, and he will find during and at the end of these steps, his actual password. These passwords constantly change (one-time passwords) and stand out because of their security.

User Experience with Security in a Wide-area TCP/IP Environment

Peder Chr. Nørgaard

Computer Science Department,
Aarhus University,
Denmark
pcnorgaard@daimi.dk

EUUG is currently considering establishing a TCP/IP based network on leased lines between its European members, similar to the North American Internet.

We know from stories from the USA that the Internet is breeding ground for things like the Internet Worm of November 1988. This gives natural basis for concerns about the security problems in the wide-area TCP/IP environment.

In the Scandinavian countries the NORDUnet has been operational since January 1989. NORDUnet connects most Nordic universities offering several services, among those TCP/IP. The NORDUnet is connected to the John v. Neumann

EUUG

center in USA through a 64 Kb transatlantic line, effectively integrating it into the Internet.

I will here report on those considerations of security that we have been focusing on since it became evident that any computer science student of USA and Canada can connect to any TCP/IP-based service on all our machines.

How to Protect Your Software Through International Copyright Laws: Step-By-Step Instructions[1]

Alicia Dunbar Gronke

USENIX Association
Berkeley, California
epg@sun.com

This article will attempt to clarify some common misconceptions about copyright formalities as applied to intellectual properties. Due to the fact that these laws vary from country to country, the examples presented here should be viewed only as basic approaches to copyright protection, not a legal reference.

Taking these few necessary precautions to protect your work from theft or unauthorized alteration isn't nearly as much of a task as one might believe. Here, we set about taking the mystery out of what should be only a few simple procedures. The idea being stressed here is taking as many preventative measures as possible to protect your work. The effort required to do this is minimal.

UNIX in German Speaking Countries

Wolfgang Christian Kabelka

Hockegasse 17
A-1180 Vienna

With hardware becoming less lucrative the next great battle in computers may be over software. Customers increasingly see versatile software as strategic tool to gain competitive advantages. This paper should help to answer the question if UNIX grants the application coverage in German speaking countries?

Using an Object-Oriented Model of UNIX for Fault Diagnosis

Anita Lundeberg

Department of Artificial Intelligence

Gail Anderson, Paul Chung

Artificial Intelligence Applications Institute

Alex Zbyslaw

Centre for Speech Technology Research

University of Edinburgh
80 South Bridge
Edinburgh EH1 1HN
United Kingdom

---

1. This paper is reprinted in this newsletter.

gail.anderson@ed.ac.uk

The modification of an existing system based on a model of UNIX is described. Object-oriented programming techniques are employed in a new version of the model. This is combined with code for diagnosis of problems with UNIX and with a program to generate parts of the model automatically, producing a general diagnostic system. A demonstrator application for the system is developed.

Modelling the NFS Service on an Ethernet Local Area Network

Floriane Dupre-Blusseau

Centre National d'Etudes des Télécommunications
Service LAA/ITP/GMI
Route de Trégastel
BP 40
22301 Lannion Cédex
France
blusseaf@lannion.CNET.FR

This article contributes to the field of performance evaluation based on simulation of queueing networks. Obtaining information on the performance of a local area network supporting the distributed file system traffic of NFS (Network File System) is the main goal of our work. We want to find out beforehand, with scientific methods and no more empirically, whether excess load brought about by the addition of connected workstations on the network, saturates the network. To achieve this goal, we studied NFS and the underlying protocols with a view to performance. Based on the generated load on the network and the processing delays, we retained the most relevant features in order to complete an NFS model. Then we chose the most appropriate software to build our model, which was QNAP (Queueing Network Analysis Package). Finally we designed and solved the NFS model, using simulation and by varying entry parameters.

One of the main results relates to the acquisition of an NFS model which does not yet exist, and of a methodology also usable in other environments. We are able to determine, by solving the model, the greatest number of connected workstations working together on the local area network. One also can use the model to choose between several network utilization policies, the most appropriate one, provided that the criteria were first fixed.

RISC vs. CISC From the Perspective of Compiler/Instruction Set Interaction

Daniel V. Klein

Software Engineering Institute
Carnegie Mellon University
Pittsburgh, PA 15217
dvk@sei.cmu.edu

This paper compares the utilization of a number of different computer instruction sets by a collection of compilers. Wherever possible, several compilers were used for each architecture. This paper demonstrates that CISC instruction sets are underutilized by compilers, while RISC instruction sets are nearly completely utilized. We observe that if an instruction exists on a computer, it should be usable by the compilers for that computer. Because CISC computers have large numbers of instructions which are not effectively used by

compilers, the instructions are superfluous. By eliminating superfluous and redundant instructions from architectures, future systems can run more efficiently, and algorithms can be executed with greater celerity.

On the Evaluation of the Performance of RISC Systems

Kurt P. Judmann

Technical University of Vienna

Modern computer systems from the PC-level up to Supercomputers claim to gain performance from using of a 'Reduced Instruction Set'. From the users and system analysts point of view it is hardly visible whether increased performance over other systems results from a reduced instruction set or from other design features. Such features can be cache hierarchies, parallel processing, pipelining or different implementation of the memory interface. This paper describes a model which can be used to calculate an average instruction execution time for instruction groups, and mixes of instruction groups of a given processor. It allows comparisons of different processors by using a code substitution method and helps in evaluating the effects of the different architectural features used in RISC systems. Although the model can be used for processors in general it is tailored to describe the techniques used in RISC processors.

SPARC – Scalable Processor Architecture

Dr. Martin Lippert

Sun Microsystems GmbH
Bahnhofstrasse 27
D–8043 Unterfoehring
mlippert@sunmuc.uucp

This paper gives an overview over the technical concept of a new RISC architecture, called SPARC, developed by Sun Microsystems. SPARC stands for Scalable Processor ARChitecture – the design specification is published and licensable. Sun has licensed SPARC processors to several semiconductor vendors and over 50 computer manufacturers.

Its well known that RISC based microprocessors offer substantially more cpu power than systems with processors based on a traditional architecture with complex instruction sets. The term 'scalable' refers to the size of the smallest lines on a chip. As lines become smaller, chips get faster. However, some chips designs do not shrink well – they do not scale properly – because the architecture is too complicated. Because of its simplicity, SPARC scales well. Consequently, SPARC systems will get faster as better chip–making techniques are perfected.

In combination with efforts of unifying UNIX, SPARC based systems will allow binary compatibility among systems from different vendors like in the PC world today.

TeamSo:
Team Software Development Support System

Eva Strausz & Janos Szel

Department of Electronics Computer and Automation Institute
Hungarian Academy of Sciences

A software tool is introduced to help software development in team-work. This paper provides an overview of this package called TeamSo, and describes some of the most important design and implementation decisions.

A SQL Programming Interface for the
Relational Database System Db++

Ralph Zainlinger
Thomas Hadek

Technical University Vienna
Institut für Techn. Informatik
Treitlstr. 3/182/1
A-1040 Vienna
Austria
ralph@vmars.at
tom@vmars.at

Database systems in the UNIX world are still a controversial topic splitting the research community. Today's users can choose between systems such as ORACLE, well known, broadly accepted, and expensive, or typical UNIX databases, such as the relational database system db++, integrated into the UNIX framework at a reasonable price but with rather low capabilities compared with the first group. The major shortcomings of db++ are the low-level programming interface, the lack of mechanisms for concurrent multiple access (multi-user) and the neglect of transaction based processing. This paper describes various extensions of the relational database system db++ based upon a high-level SQL programming interface. The main purpose is to maintain the obvious advantages of db++ by simultaneously increasing its functionality and general applicability with standardised mechanisms and concepts.

Processable Multimedia Document Interchange
Using ODA*

Jaap Akkerhuis
Ann Marks
Jonathan Rosenberg
Mark S. Sherman

Information Technology Center
Carnegie Mellon University
Pittsburgh, USA
jaap+@andrew.cmu.edu
annm+@andrew.cmu.edu
jr+@andrew.cmu.edu
mss+@andrew.cmu.edu

The EXPRES (Experimental Research in Electronic Submission) project promotes the electronic interchange of multi-media documents among the scientific research community. For this project we concentrate on the problem of effective interchange of processable multi-media documents. In particular, we are ignoring the transfer method. Instead we concern ourselves with the question of how a multi-media document created on one system can be viewed and edited on another system.

The obvious technique of performing translations between each pair of systems is impractical. In order to attack the problems efficiently, we make use of a standard representation. We have settled on the international standard Office Document Architecture (ODA)* as the intermediate format. This paper discusses how we implemented ODA for interchange.

TCP/UDP Performance as Experienced by User-Level Processes

Josef Matulka

Department of Applied Computer Science
Institute of Information Processing and Information Economics
Vienna University of Economics and Business Administration
Augasse 2-6,
A-1090 Vienna,
Austria
matulka@awiwuw11.earn

This paper is concerned with the presentation of empirical data on the performance of the internet protocols TCP and UDP as experienced by a distributed application. After a short summary of the main services provided by the internet protocols IP, UDP, and TCP, an abstract data type built on top of the well-known Berkeley IPC sockets is presented, which has been used to implement a benchmark experiment. The experimental design of this experiment is described and performance data are given for UDP and TCP under two different environments ('loaded' hosts and 'unloaded' hosts).

Resource Management System for UNIX Networks

D. Schenk, G. Reichelt, A. Pikhard

UNISYS Austria
Mariahilferstrasse 20
A-1070 Vienna
Austria

Although there are comfortable tools for UNIX administration on single, i.e. non-networked systems, there is a lack of such tools for (large) UNIX networks.

File sharing utilities such as RFS or NFS offer file-sharing and, in the case of NFS, automated updating of distributed data. A tool for central resource administration is missing.

UNISYS RMS (Resource Management System) allows central maintenance of distributed resources. A resource is any part of any component of the network that may be (re)configured, e.g. a user-id, a terminal port or a software product.

The 'heart' of RMS is a relational data base which runs on one specific node in the network called the RMS Server. The database application generates requests which are forwarded to the other systems in the network called the RMS Clients. Every update to the data base causes a chain of actions to be started on the clients.

A prototype of RMS is currently evaluated: it runs both on LAN (Ethernet) and WAN (X.25). The full RMS is currently under development at UNISYS Vienna.

---

* *Office Document Architecture (ODA) and Interchange Format (ISO 8613)* International Organization for Standardization (ISO), 1988.

Interconnection of LANs, Using ISDN, in a TCP/IP Architecture

Philippe Blusseau

OST-DRD/CMC
CNET LAA/ITP/RIC
Route de Trégastel
B.P. 40
22301 LANNION CEDEX
blusseau@cnetlu.uucp

Network interconnection is not really a recent problem: network architectures currently used, or planned, specify a level structure, inside which one can insert new wide or local area networks. Of course, ISDN is one of these networks that can be included in those architectures.

In this paper, we will point out which interconnections can be done between LANs and ISDN. We will then focus on the new services offered by the ISDN, especially the basic "S0" access, and how they can be efficiently used in this context.

System Administration of UNIX Networks: Two Approaches to Supporting the Management of Large, Distributed, Multi-Vendor Networks

Georg-Michael Raabe
Carol J. Rosenstock

Apollo Computer GmbH
Hahnstrasse 37–39
D-6000 Frankfurt 71
West Germany
raabe_g@apollo.com

This paper discusses two aspects of the support of large, distributed, multi-vendor networks. First we will address user account management and show how a single system can manage a heterogeneous network. Second we will address file system backup and restore and present a system designed to coordinate backup for an entire network.

Porting Applications to the XVIEW Toolkit and the OPEN LOOK Graphical User Interface

Nannette Simpson

Sun Microsystems
2550 Garcia Avenue
Mountain View
CA 94043
nannette@sun.com

The OPEN LOOK Graphical User Interface Functional Specification has evolved over the past two years with the final draft, Revision 18, becoming available for review May 1989. Sun has implemented several prototype toolkits which captured the progress of the user interface design through its infancy and adolescence. With the impending maturity of the design, Sun offers product toolkits and a suite of applications built on those toolkits. This paper details the issues of transforming existing tools that run on SunView (Sun's kernel-based window system), into integrated applications running on XView, Sun's first OPEN LOOK, X toolkit.

X Display Servers:
Comparing their Functionality
and Architectural Differences
to Diskless Workstations

Timothy L. Ehrhart

Ericsson Telecommunicatie BV
Haansebergseweg 1
P.O. Box 8
5120 AA Rijen
The Netherlands
+31 1612 29308
Tim.Ehrhart@ericsson.se

X Display Servers are a new generation of computing devices arriving on the scene. They consist of the now familiar high resolution bitmap screen, keyboard and mouse, replacing the single font tubes and keyboard as 'stdout' and 'stdin'. These devices contain a powerful microprocessor, local memory, network interfaces, built-in fonts, and run only the X11 server code (i.e., not running UNIX). They don't utilise any of the local compute power in application support. The applications, or clients, in X terminology, all run remotely on another host-based computer. The output of the remote client is displayed on the local display, managed by the X11 server. These devices appear to offer the same functionality as a Diskless Workstation. Can the two be compared in an apples-to-apples fashion?

A System for the Redirection
of Graphical User-Interaction

Robin Faichney

Computing Laboratory
University of Kent at Canterbury
Canterbury
Kent
CT2 7NF
UK
rjf@ukc.ac.uk

The redirection of user-interaction is complicated in a graphical environment due to the nature and variety of graphical actions, compared with character stream input/output. The solution is to redirect at a high level within the application. Accordingly, this system is designed to allow the redirection of all traffic between the user-interface and the underlying functionality of an application, to and/or from another program. This facility is quite flexible and has a variety of uses. These include facilitation of the automatic testing of the underlying functionality of graphics-interfaced applications, and allowing extensibility of user-interfaces, via the recording and replay of command macros. A prototype and some applications which use it have been implemented, demonstrating the main features of the system.

Performance Analysis
for Shared Oracle Database in UNIX Environment

C. Boccalini

I & O – Informatica e Organizzazione S.r.L.
Genova
Italy

J. Marino, M. Paolucci

Department of Communications,
Computer and Systems Sciences (DIST),
Via Opera Pia 11A
16145 Genova
Italy
paolucci@dist.unige.it

The methodology used in order to execute a performance evaluation in UNIX environment, in a case of concurrent access to a common Oracle database, is presented. The reason of such an evaluation, as well as the model adopted to structure the benchmarks, are discussed. Some results obtained are presented, pointing out their meaning in relation to the different hardware configurations tested and the model itself.

A Transaction Monitor for SINIX

Heike von Lützau-Hohlbein

GBI
Gesellschaft Beratender
Informatiker mbH
Schieggstr. 9 D
D-8000 München 71

The use of transaction monitors is widespread in the mainframe arena. Information retrieval-, booking-, warehouse control-, stock control- or personnel administration systems only work efficiently due to their use. Taking into account that the commercial EDP market is becoming more and more penetrated by UNIX systems, there is a rising demand for DB/DC applications. It is in this context that transaction oriented program to program communication gains importance in both homogeneous and heterogeneous computer networks including those containing UNIX systems.

The problems of the implementation of transaction monitors in UNIX systems in general and known implementations in particular are discussed.

As an example UTM (SINIX), Universal Transaction Monitor for SINIX (SINIX is the SIEMENS derivative of UNIX), is presented in more detail. In particular its interfaces to the user, interfaces to the system and how it is embedded in the SINIX system.

Connectivity in a heterogeneous environment is explained together with what types of partner can be addressed and how. It will be shown how UTM(SINIX) can help to integrate SINIX computers and their users into distributed DB/DC applications, how mainframes can be accessed and their power used.

An Approach to Reliability In Distributed Programming

Giandomenico Spezzano and Domenico Talia

CRAI
Localitá S. Stefano
87036 Rende (CS)
Italy
dot@crai.uucp

This paper proposes the use of a distributed concurrent language for the implementation of fault-tolerant distributed systems. In our approach, distributed software systems are composed of a set of cooperating processes, which communicate using the message passing model, and are placed on the various hosts of the distributed architecture. The interesting aspects of our approach are presented in terms of modularity, concurrency, portability and reliability. This approach is bound to the use of a high-level concurrent language in contrast to the traditional approaches for the implementation of the reliable distributed systems.

XEiffel: An Object-Oriented Graphical Library and an OPEN LOOK Based on it

Marco Menichetti

UniRel
Firenze
Italy

Object-oriented languages can be extended by user-defined classes libraries. In this paper I present XEiffel. It is a graphical extension of the object-oriented language Eiffel. It is based on the X Window system but adds to it some interesting object-oriented features. As XEiffel application I present an OPEN LOOK graphical interface toolkit. The whole work has been developed using SUN 3/60 workstations at UniRel.

Efficient Implementation of Low-Level Synchronization Primitives in the UNIX-Based GUIDE Kernel

D.Decouchant

Laboratoire de Genie Informatique

E.Paire, M.Riveill

Centre de Recherche BULL
Unite Mixte BULL-IMAG Systemes
Z.I. de Mayencin
2. Rue de Vignate
38610 Gieres, France
decoucha@imag.imag.fr
paire@imag.imag.fr
riveill@imag.imag.fr

When developing new and complex applications on top of UNIX system, implementors are usually faced with synchronization problems whose solution is not simple. This is especially true when such applications are in fact a new system level which defines a different model of synchronization. Standard mechanisms normally provided are simple, general but not efficient enough when heavily used. Synchronization implementation should be of low cost with respect to other system components, but this is not usually the case. This paper first summarizes the synchronization mechanism required by our object-oriented environment, then describes the implementation of our final solution, which was derived in several steps, and finally presents experience and performance

measurements of different progressive improvements.

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.

Social Aspects of EUUG and USENIX

John S. Quarterman

Texas Internet Consulting
701 Brazos Suite 500
Austin, TX 78701
U.S.A.
jsq@longway.tic.com

EUUG and USENIX conferences are quite similar in many ways, but they also differ, particularly in their social aspects. This paper presents some comments about technical and social events scheduled as part of the conference proper, and about more informal activities. The continental network cultures are contrasted. The purpose is not to show that one conference is better than another; many of the more basic features are not reproducible (and probably not desirable) in a different environment. But the organisers of each conference have benefitted in the past by adopting some features of the other, and still could do so.

Management Committee Meeting

27th October, 1989

MINUTES

The meeting opened at 10:00am with the following committee members present: President Greg Rose (GR), Secretary Tim Roper (TR), Treasurer Michael Tuke (MT), Pat Duffy (PD), and John Carey (JC). Also present was the AUUGN Editor David Purdue (DP) and the Summer '90 meeting coordinator Glenn Huxtable (GH). Wael Foda (WF) attended for some of the time.

1. Apologies
   From Chris Maltby who was overseas and Peter Barnes who was had been unable to get a flight from Brisbane due to airline disruptions.

2. Minutes of last meeting (8th August, 1989)
   Moved PD/JC that **The minutes be accepted.** Carried.

4. AUUG89
   There was discussion of tutorials. They were agreed to be a good thing. There had been some complaints about the quality of at least one tutorial.

   WF tabled a financial report, delegates list, and exhibition visitors list. The cost of the free registration list was discussed and agreed to be too great.

5. AUUG90
   The World Congress Centre had been inspected by TR and JC in September. They had recommended that it be used instead of the Southern Cross. ACMS has booked it on AUUG's behalf for 25th - 28th September, 1990. Specifically, the large exhibition hall (about 100 stands), the smaller auditorium (500 seats), the banquet area and some side rooms.

   The venue had been inspected by the committee prior to the meeting. The main concern was the capacity of the smaller auditorium. A possible fallback was to use the larger auditorium (1500 seats) if numbers warranted it and if it had not been booked by someone else. A definite fallback was to use the banquet area which could seat over 1000.

   WF presented a budget, a planner, an exhibition layout and a list of current bookings.

   There was discussion about publicity methods. WF suggested more press advertising.

   The Call for Papers would appear in AUUGN Vol 10 No 5 and be mailed to the full mailing list within a month with a covering letter. JC responsible for CFP, PD for letter. It was suggested that it be advertised in the press.

   It was agreed that membership information should be mailed with the brochure and that the brochure should be redesigned by PD and WF, contain more programme information and be earlier than usual.

WF reported that the exhibition space promotional brochure had been reprinted for the new venue. Exhibitors were being asked to choose their space in the order that they had booked it for the old venue. Once this process was complete the remaining space would be offered for sale.

Tutorial fees were set at $100 for one and $160 for two. An amount of $600 to be paid to presenters but with quality control, ie. materials to be approved by subcommittee before final acceptance.

Free registration was to be restricted to speakers (one per paper, not transferable), the Programme Chair and the Convenor. In particular, tutorial presenters were not to received complimentary registration and were to be paid more money instead.

MT volunteered to be Convenor. Ken McDonnel was invited by phone to coordinate guest speakers and accepted. DP volunteered to coordinate tutorials.

It was agreed that the delegate list should state AUUG's ownership of copyright as insurance against people who used it as a mailing list.

Suggestions for guest speaker topics included the X Window System and Mach.

There was discussion of after dinner speakers. WF to follow up suggestions.

WF has alternative proposals for AUUG91 in progress as requested by AUUG: proposed venues were the Canberra Convention Centre and Darling Harbour (Sydney).

11. Secretarial Assistance
   ACMS had indicated interest in taking on secretarial duties. TR had supplied them with duties list. WF to quote. GR said decisions on these things should be delegated. TR was delegated to decide.

3. Business arising from Minutes
   Re 5(c), $400 had been refunded as we had been overcharged for freight. The directories were in the hands of the direct mail house.

   Re 8, a letter had been received from the author protesting the reprinting of the paper in question in ComputerWorld and requesting that there be no further publication of it. It was agreed that no permission had been granted to ComputerWorld to reprint from AUUGN Vol 10 No 4 and the Secretary was requested to write to ComputerWorld pointing out the breach of copyright and drawing attention to the copyright notice in AUUGN.

   Re 15(c), the Secretary reported that this had not been done.

Re 7, DP had consulted Tony Blackmore, editor of Professional Computing, who had expressed the view that a formal release form was unnecessary.

6. **President's Report**
   GR reported that he had been very busy and had not done much.

   He had spoken to Al Nemeth (President USENIX, on governing body of UNIX International, on standards bodies, employed by Prime R&D) as a possible Summer 90 guest speaker. Prime to pay.

   Had spoken to the DMR Group who were undertaking study of UNIX market in Australia similar to that done in Canada. [See Other Business.]

   Moved PD/JC That the President's report be accepted. Carried.

7. **Secretary's Report**
   TR reported as follows.

   Memberships as at 16/10/89 stood at 104 Institutional (up 24), 399 Ordinary (up 86), 12 Student (up 2). There were 23 (up 5) newsletter subscribers.

   JC had brought database up to date, temporarily relieving a large backlog of forms. This allowed mailing lists to be generated for AUUGN Vol 10 No 5, Summer 90, Product Directories and Computing Systems.

   TR stressed need for secretarial assistance. For example, correspondance had been recently neglected due to employer committments.

   Tabling of correspondance was deferred again by agreement.

   Moved JC/MT That the Secretary's report be accepted. Carried.

8. **Treasurer's Report**

   MT reported as follows.

   He was now emplyed by ANL Ltd.

   As at 2/10/89 the cheque account balance was $37880.72. Cash at hand was about $50000 but about $25000 was payable in the next 1-2 months. GR suggested short term (at call) investment of $20000.

   A rough budget suggested net income of $30000 for 1989/90. It was agreed that this should be incorporated in the minute.

   Moved TR/JC That the Treasurer's report be accepted. Carried.

9. **AUUGN Editor's Report**
   DP reported that AUUGN Vol 10 No 5 was at the printers.

Moved JC/TR that new members be sent a copy of the current issue. Carried.

PD suggested that if ACMS take on secretarial assistance they could send a card on receipt of fees.

10. 1990 Summer Meetings
GH reported as follows.

Committments to organise meetings had been received from Perth (GH), Hobart (S. Bittinger), Melbourne (S. Prince), Canberra (Peter Wishart and Ross Hand), Sydney (Softway). In Brisbane PB and Tom Crawley were looking for a convenor, in Townsville there was interest from GBRMPA and James Cook Uni.

Al Nemeth had been invited as guest speaker subject to funding by Prime. He had indicated positive interest but not formally accepted. Ken McDonnel was to speak at Perth, Melbourne and possibly Hobart. GR to speak at Melbourne and possibly Perth.

JC suggested video tape of meetings be sent to Darwin etc. in lieu of local meeting.

GH suggested extra speakers for remote locations: Hobart, Townsville and Canberra. Perth should be OK due to strong local group although AUUG memebership could benefit from a speaker like GR. Airline disruption could be a problem.

JC reported on the Melbourne meeting as follows. It was aiming to serve technical community. Stephen Prince was looking after programme, Kathy Ching the site details. A Call for Speakers ahd been mailed. So far received 5 expressions of interest and may membership enquiries. Had booked two alternative lecture theatres at Monash Uni. Lunch was optional at about $20. Features to include car parking and air-conditioning.

GH reported on Perth meeting as follows. Himself was convenor. Chris McDonald was programme chair. A Call for Speakers had been plaguarised from Melbourne. Had booked the Orchard Hotel at $25 per head for 120 heads. Lunch was $40 per head.

GR stated that as we had yet to make summer meetings work we should leave the more academic details until next time.

Moved TR/PD That AUUG contribute to the extent of financing a speaker from within Australia for every meeting. Carried with GR abstaining.

12. Constitutional Changes
CM was absent. GR suggested that he had probably forgotten about it and that the Secretary should remind him.

4. AUUG89 (cont.)
Further suggestions for guest speakers: RAC, Salamon Brothers, market analyst, Mach person, MIT X Window System person.

PB's report was distributed and discussed.

It was suggested that vendor and user companies be invited to offer papers.

13. Other Business.
GR described the DMR proposal. GR had been invited to sit on steering committee. DMR would like AUUG assistance particularly in contacting Institutional members in search of investors. It was agreed that AUUG should ensure equitable treatment of all members.

Moved TR/PD That AUUG assist DMR provided that (a) executive summary be supplied to all members (b) all members have equal opportunity to invest (c) DMR pay AUUG $10000 plus costs (d) DMR write an article for AUUGN (e) DMR offer a paper for AUUG90. Carried.

GR and PD volunteered for AUUG90 Programme Commitee. It was suggested that PB be invited. It was agreed that we should require CV and photo with abstracts.

GH presented a petition from WAUG to become a chapter. Moved JC/MT That it be accepted. Carried.

14. Next Meeting
It was resolved that the next meeting should be held in Melbourne on Wednesday 7th February, 1990.

The meeting was closed at 4:30pm.

AUUGN Back Issues

Here are the details of back issues of which we still hold copies. All prices are in Australian dollars and include surface mail within Australia. For overseas surface mail add $2 per copy and for overseas airmail add $10 per copy.

pre 1984	Vol 1-4	various	$10 per copy
1984	Vol 5	Nos. 2,3,5,6	$10 per copy
		Nos. 1,4	unavailable
1985	Vol 6	Nos. 2,3,4,6	$10 per copy
		No. 1	unavailable
1986	Vol 7	Nos. 1,4-5,6	$10 per copy
		Nos. 2-3	unavailable
			(Note 2-3 and 4-5 are combined issues)
1987	Vol 8	Nos. 1-2,3-4	unavailable
		Nos. 5,6	$10 per copy
1988	Vol 9	Nos. 1,2,3	$10 per copy
		Nos. 4,5,6	$15 per copy
1989	Vol 10	Nos. 1-6	$15 per copy

Please note that we do not accept purchase orders for back issues except from Institutional members. Orders enclosing payment in Australian dollars should be sent to:

AUUG Inc.
Back Issues Department
PO Box 366
Kensington NSW
Australia 2033

AUUG

Membership Categories

Once again a reminder for all "members" of AUUG to check that you are, in fact, a member, and that you still will be for the next two months.

There are 4 membership types, plus a newsletter subscription, any of which might be just right for you.

The membership categories are:

Institutional Member
Ordinary Member
Student Member
Honorary Life Member

Institutional memberships are primarily intended for university departments, companies, etc. This is a voting membership (one vote), which receives two copies of the newsletter. Institutional members can also delegate 2 representatives to attend AUUG meetings at members rates. AUUG is also keeping track of the licence status of institutional members. If, at some future date, we are able to offer a software tape distribution service, this would be available only to institutional members, whose relevant licences can be verified.

If your institution is not an institutional member, isn't it about time it became one?

Ordinary memberships are for individuals. This is also a voting membership (one vote), which receives a single copy of the newsletter. A primary difference from Institutional Membership is that the benefits of Ordinary Membership apply to the named member only. That is, only the member can obtain discounts an attendance at AUUG meetings, etc. Sending a representative isn't permitted.

Are **you** an AUUG member?

Student Memberships are for full time students at recognised academic institutions. This is a non voting membership which receives a single copy of the newsletter. Otherwise the benefits are as for Ordinary Members.

Honorary Life Membership is not a membership you can apply for, you must be elected to it. What's more, you must have been a member for at least 5 years before being elected.

It's also possible to subscribe to the newsletter without being an AUUG member. This saves you nothing financially, that is, the subscription price is greater than the membership dues. However, it might be appropriate for libraries, etc, which simply want copies of AUUGN to help fill their shelves, and have no actual interest in the contents, or the association.

Subscriptions are also available to members who have a need for more copies of AUUGN than their membership provides.

To find out if you are currently really an AUUG member, examine the mailing label of this AUUGN. In the lower right corner you will find information about your current membership status. The first letter is your membership type code, N for regular members, S for students, and I for institutions. Then follows your membership expiration date, in the format exp=MM/YY. The remaining information is for internal use.

Check that your membership isn't about to expire (or worse, hasn't expired already). Ask your colleagues if they received this issue of AUUGN, tell them that if not, it probably means that their membership has lapsed, or perhaps, they were never a member at all! Feel free to copy the membership forms, give one to everyone that you know.

If you want to join AUUG, or renew your membership, you will find forms in this issue of AUUGN. Send the appropriate form (with remittance) to the address indicated on it, and your membership will (re-)commence.

As a service to members, AUUG has arranged to accept payments via credit card. You can use your Bankcard (within Australia only), or your Visa or Mastercard by simply completing the authorisation on the application form.

AUUG Incorporated
Application for Institutional Membership
Australian UNIX* systems Users' Group.
*UNIX is a registered trademark of AT&T in the USA and other countries.

To apply for institutional membership of the AUUG, complete this form, and return it with payment in Australian Dollars, or credit card authorisation, to:

AUUG Membership Secretary
P O Box 366
Kensington NSW 2033
Australia

● Foreign applicants please send a bank draft drawn on an Australian bank, or credit card authorisation, and remember to select either surface or air mail.

This form is valid only until 31st May, 1990

.... does hereby apply for

☐ New/Renewal* Institutional Membership of AUUG $325.00

☐ International Surface Mail $ 40.00

☐ International Air Mail $120.00

Total remitted AUD$_____
(cheque, money order, credit card)

* Delete one.

I/We agree that this membership will be subject to the rules and by-laws of the AUUG as in force from time to time, and that this membership will run for 12 consecutive months commencing on the first day of the month following that during which this application is processed.

I/We understand that I/we will receive two copies of the AUUG newsletter, and may send two representatives to AUUG sponsored events at member rates, though I/we will have only one vote in AUUG elections, and other ballots as required.

Date: __/__/__   Signed: _____

Title: _____

☐ Tick this box if you wish your name & address withheld from mailing lists made available to vendors.

For our mailing database - please type or print clearly:

Administrative contact, and formal representative:

Name: ....   Phone: .... (bh)

Address: .... (ah)

....

....   Net Address: ....

....   *Write "Unchanged" if details have not*

....   *altered and this is a renewal.*

Please charge $_____ to my/our ☐ Bankcard ☐ Visa ☐ Mastercard.

Account number: __ __ __ __ __ __ __ __ __ __ __ __ __ __ __ __.   Expiry date: __/__.

Name on card: _____   Signed: _____

Office use only: **Please complete the other side.**

Chq: bank _____ *bsb* _____ - _____ *a/c* _____ *#* _____

Date: __/__/__ *$*   *CC type* ___ *V#* _____

Who: _____   *Member#* _____

Please send newsletters to the following addresses:

Name: ....   Phone: .... (bh)

Address: .... (ah)

....   Net Address: ....

....

....

....

Name: ....   Phone: .... (bh)

Address: .... (ah)

....   Net Address: ....

....

....

....

Write "unchanged" if this is a renewal, and details are not to be altered.

Please indicate which Unix licences you hold, and include copies of the title and signature pages of each, if these have not been sent previously.

Note: Recent licences usally revoke earlier ones, please indicate only licences which are current, and indicate any which have been revoked since your last membership form was submitted.

Note: Most binary licensees will have a System III or System V (of one variant or another) binary licence, even if the system supplied by your vendor is based upon V7 or 4BSD. There is no such thing as a BSD binary licence, and V7 binary licences were very rare, and expensive.

☐ System V.3 source   ☐ System V.3 binary

☐ System V.2 source   ☐ System V.2 binary

☐ System V source   ☐ System V binary

☐ System III source   ☐ System III binary

☐ 4.2 or 4.3 BSD source

☐ 4.1 BSD source

☐ V7 source

☐ Other *(Indicate which)* ....

AUUG Incorporated
Application for Ordinary, or Student, Membership
Australian UNIX* systems Users' Group.
*UNIX is a registered trademark of AT&T in the USA and other countries

To apply for membership of the AUUG, complete this form, and return it with payment in Australian Dollars, or credit card authorisation, to:

AUUG Membership Secretary
PO Box 366
Kensington NSW 2033
Australia

● Please don't send purchase orders — perhaps your purchasing department will consider this form to be an invoice.

● Foreign applicants please send a bank draft drawn on an Australian bank, or credit card authorisation, and remember to select either surface or air mail.

This form is valid only until 31st May, 1990

I, ... do hereby apply for

☐ Renewal/New* Membership of the AUUG $78.00

☐ Renewal/New* Student Membership $45.00 (note certification on other side)

☐ International Surface Mail $20.00

☐ International Air Mail $60.00 (note local zone rate available)

Total remitted   **AUD$_____**

(cheque, money order, credit card)

* Delete one.

I agree that this membership will be subject to the rules and by-laws of the AUUG as in force from time to time, and that this membership will run for 12 consecutive months commencing on the first day of the month following that during which this application is processed.

Date: __/__/__   Signed: _____

☐ Tick this box if you wish your name & address withheld from mailing lists made available to vendors.

For our mailing database - please type or print clearly:

Name: ....

Phone: .... (bh)

Address: ....

.... (ah)

....

Net Address: ....

....

....

Write "Unchanged" if details have not

....

altered and this is a renewal.

Please charge $_____ to my ☐ Bankcard ☐ Visa ☐ Mastercard.

Account number: __ __ __ __ __ __ __ __ __ __ __ __ __ __ __ __ __.   Expiry date: __/__.

Name on card: _____   Signed: _____

Office use only:

Chq: bank _____ bsb _____-_____ a/c _____ # _____

Date: __/__/__ $ *CC type ___ V# _____*

Who: _____   *Member# _____*

Student Member Certification *(to be completed by a member of the academic staff)*

I, ... certify that

.... *(name)*

is a full time student at .... *(institution)*

and is expected to graduate approximately ___/___/___ .

Title: _____   Signature: _____

AUUG Incorporated
Application for Newsletter Subscription
Australian UNIX* systems Users' Group.

*UNIX is a registered trademark of AT&T in the USA and other countries

Non members who wish to apply for a subscription to the Australian UNIX systems User Group Newsletter, or members who desire additional subscriptions, should complete this form and return it to:

AUUG Membership Secretary
P O Box 366
Kensington NSW 2033
Australia

● Please don't send purchase orders — perhaps your purchasing department will consider this form to be an invoice.

● Foreign applicants please send a bank draft drawn on an Australian bank, or credit card authorisation, and remember to select either surface or air mail.

● Use multiple copies of this form if copies of AUUGN are to be dispatched to differing addresses.

This form is valid only until D

Please *enter / renew* my subscription for the Australian UNIX systems User Group Newsletter, as follows:

Name: ....

Address: ....

....

....

....

....

Phone: .... (bh)

.... (ah)

Net Address: ....

Write "Unchanged" if address has

not altered and this is a renewal.

For each copy requested, I enclose:

☐ Subscription to AUUGN $ 90.00

☐ International Surface Mail $ 20.00

☐ International Air Mail $ 60.00

Copies requested (to above address)   _____

Total remitted   AUD$_____

(cheque, money order, credit card)

☐ Tick this box if you wish your name & address withheld from mailing lists made available to vendors.

Please charge $_____ to my ☐ Bankcard ☐ Visa ☐ Mastercard.

Account number: __ __ __ __ __   __ __ __ __   __ __ __ __   __ __ __ __ .   Expiry date: __/__ .

Name on card: _____   Signed: _____

Office use only:

Chq: bank _____ *bsb* _____-_____ *a/c* _____ # _____

Date: __/__/__ $ *CC type* ___ *V#* _____

Who: _____   *Subscr#* _____

AUUG

Notification of Change of Address
Australian UNIX* systems Users' Group.

*UNIX is a registered trademark of AT&T in the USA and other countries.

If you have changed your mailing address, please complete this form, and return it to:

AUUG Membership Secretary
P O Box 366
Kensington NSW 2033
Australia

Please allow at least 4 weeks for the change of address to take effect.

Old address (or attach a mailing label)

Name: ....   Phone: .... (bh)

Address: .... (ah)

....

....   Net Address: ....

....

....

New address (leave unaltered details blank)

Name: ....   Phone: .... (bh)

Address: .... (ah)

....

....   Net Address: ....

....

....