The USENIX Association Newsletter

Volume 15, Number 6

November/December 1990

CONTENTS

Tom Strong .................................................. 3
Call for Papers: Summer 1991 USENIX Conference .................. 4
USENIX Student Attendee Grant ........................................ 4
USENIX Winter Conference Program ................................... 6
Call for Proposals to Chair the Winter 1993 USENIX Technical Conference ..... 10
Request for Proposals to Chair the Large Installation Systems Administration (LISA) Conference ........................................... 10
Report from the UKUUG ........................................... 11
Long-Term Calendar of UNIX Events ..................................... 12
Summary of Board of Directors' Meeting ................................ 13
Book Reviews
  Designing Object-Oriented Software ................................. 15
    Edward Gordon
  Software Engineering: Concepts and Management .................. 16
    Robert C. Birss
  An Update on UNIX-Related Standards Activities .................. 17
    Jeffrey S. Haemer
Local User Groups .............................................. 40

The closing date for submissions for the next issue of ;login: is January 2, 1991.
;login: (ISSN 1044-6397) is published bi-monthly by the USENIX Association.

The USENIX Association is a not-for-profit organization of those interested in UNIX and UNIX-like systems. It is dedicated to fostering and communicating the development of research and technological information and ideas pertaining to advanced computing systems, to the monitoring and encouragement of continuing innovation in advanced computing environments, and to the provision of a forum where technical issues are aired and critical thought exercised so that its members can remain current and vital.

The officers of the Association are:

President M. Kirk McKusick
mckusick@usenix.org
Vice-President Michael D. O'Dell
mo@usenix.org
Secretary Rob Kolstad
kolstad@usenix.org
Treasurer Sharon Murrel
eowyn@usenix.org
Directors Rick Adams
rick@usenix.org
Ed Gould
ed@usenix.org
Evi Nemeth
evi@usenix.org
Barry Shein
bzs@usenix.org
Executive Director Ellie Young
ellie@usenix.org

USENIX Association Office
2560 Ninth Street, Suite 215
Berkeley, CA 94710
(415) 528-8649
office@usenix.org

The editorial staff of ;login: is:

Editor Ellie Young
Managing Editor Carolyn S. Carr
Copy Editor Melinda Shore
Advisor Barry Shein

Judith F. DesHarnais, Conference Coordinator
USENIX Conference Office
22672 Lambert Street, Suite 613
El Toro, CA 92630
(714) 588-8649
judy@usenix.org

Cynthia Deno, Exhibit Manager
(408) 335-5646
cynthia@usenix.org

Daniel V. Klein, Tutorial Coordinator
(412) 421-2332
dvk@usenix.org

Contributions Solicited

Members of the UNIX community are encouraged to contribute articles to ;login: . Contributions may be sent electronically to login@usenix.org or through the U.S. mail to the Association office. The USENIX Association reserves the right to edit submitted material.

UUNET Subscriptions

UUNET Communications
3110 Fairview Park Drive, Suite 570
Falls Church, VA 22042
(703) 876-5050
uunet-request@uunet.uu.net

Acknowledgments

The Association uses a Sun* 3/180S running SunOS for support of office and membership functions, preparation of ;login:, and other Association activities. Connected to the Sun is a QMS Lasergrafix* 800 Printer System donated by Quality Micro Systems of Mobile, Alabama. It is used for general printing and draft production of ;login: and Computing Systems with ditroff software provided by mt Xinu. The membership and mailing lists are maintained using Sybase.*

This newsletter is for the use of the members of the USENIX Association. Any reproduction of this newsletter in its entirety or in part requires written permission of the Association and the author(s).

;login: (ISSN 1044-6397) Volume 15, Number 6 (November/December 1990). Published 6 times a year. Mailing address: USENIX Association, 2560 Ninth Street, Suite 215, Berkeley, CA 94710.

SUBSCRIPTION for non-members: $40 per year.
Second-class application pending at Berkeley, California and additional offices. Postmaster: Send address changes to USENIX Association, 2560 Ninth Street, Suite 215, Berkeley, CA 94710.
Tom Strong died in September, after a lengthy battle with lung cancer and a brain tumor.

In 1982, when the USENIX Association moved to the Bay Area from New York, Tom was part of the management company retained to manage the office (he was the Strong of Penny, Penny and Strong). When the Association decided to actually hire a staff of its own, Tom stayed on and acted as Executive Director for several months prior to the appointment of Jim Ferguson in 1985.

From 1982 through the July/August 1990 issue, Tom has been the Managing Editor, the Production Editor and the typesetter of ;login:. Tom was the lead rapporteur and compositor for the January 1983 [San Diego] UNICOM Proceedings. Every USENIX workshop proceedings from Graphics 1985 through C++ 1990 passed through his hands. Computing Systems from its inception through 3.3 was set by Tom.

Through my three years as Executive Director, I worked closely with Tom: membership renewals, election materials (for the 1984, 1986, 1988, and 1990 elections), manual and tape order forms—even the most trivial task was treated with a seriousness and care that was lavished on major efforts like the journal.

Tom was an outdoorsman who would periodically appear in my office with a fish; it was he who established the opening day of trout season as a USENIX holiday. For many years, I thought that Tom smoked the worst cigars available; perhaps they acted as a fish lure.

Tom did most of the maintenance of the Association's data base for many years, on the VAX 11/730 and on the Sun 3/180S, using awk scripts. Much of the merging of conference and workshop attendees' names into the list over half a decade was his work.

The Association (and the UNIX community as a whole) will miss Tom. So will his widow and family to whom I, on behalf of the many thousands who knew Tom's work, offer sincere condolences. Future issues of ;login: and Computing Systems will stand as continuing memorials to Tom's efforts and his dedication.

Peter H. Salus
Call for Papers: Summer 1991 USENIX Conference
Opryland Hotel, Nashville, TN, June 10–14, 1991

Multimedia—Interfaces for Now and the Future

Systems designers and multimedia developers must face the challenge of how to support and deliver the new types of interfaces—voice, video, animated graphics, touch and music—that users are demanding now. While adding immeasurably to the power of the computer for communication and expression of ideas, the new media multiply system complexity and magnify the challenge of providing easy access to computer resources.

What are the technical engineering requirements to enable your operating systems to process effectively the new types of data? What must you, and your organization, do to prepare?

How can we design new multimedia interfaces to improve information handling? What projects are underway now that offer insight into the directions to be taken in developing fully integrated multimedia systems with a coherent and meaningful framework? These are some of the questions tackled by presenters and attendees at the USENIX Summer 1991 Conference.

Formats for Presentations

The USENIX Summer 1991 Conference will provide a variety of forums in which participants can explore multimedia issues, as well as more general operating system and environment questions.

We invite submissions of your papers and multimedia presentations for the technical track. Please target a sophisticated technical audience, particularly knowledgeable of operating systems issues but keenly interested in new, interesting projects in many areas. Suggested topics include, though are not limited to:

Multimedia, applications and research
  systems integrating voice, video, audio, touch, or music
  data compression technology
  user interface/human factors

Hypermedia
  authoring systems
  hypermedia/multimedia documents

Operating systems issues
  multiprocessor systems
  distributed systems
  secure systems
  fault tolerant systems
  systems for novel architectures
  distributed file systems

Communications and Networking
  protocols
  performance
  administration and security

Programming environment
  user interfaces, windowing, graphics
  compilers and language technology
  software development and other support tools
  testing and debugging

Sophisticated Applications
  databases
  transaction processing
  instructional
  scientific, biological, medical, etc.

Form of Submissions

Submissions to the technical track should represent new work and be in the form of an abstract and outline. Be complete enough to provide details of the approach and give the committee confidence in the final paper. Full papers are accepted as well. A submission should be from 3–5 pages and include:

1. Author name(s), postal addresses, telephone numbers, and e-mail addresses.
2. Abstract: 100–300 words.
3. Outline: 2–5 pages giving enough details of the approach or algorithms to allow the committee to understand and judge the submission.
4. References and citations to relevant literature. Please show you are aware of previous work (and not reinventing the wheel).
5. Time needed for presentation. Slots are usually 30 minutes but adjustment can be made when in-depth background or audio-visual support is desirable.
6. Audio-visual presentation requirements. We are happy to provide assistance and equipment in making your presentation as audio and visually appealing as possible.
Authors whose submissions are accepted will receive instructions for the preparation of final papers to be published in the conference proceedings. We are looking into possibilities for making audio and video materials as well.

**Relevant Dates**

Abstracts and outlines due: **February 6, 1991**
Notifications to authors: **March 4, 1991**
Final papers due: **April 19, 1991**

Please submit one hard copy and one electronic copy to the address below:

Deborah K. Scherrer  
Nashville USENIX Technical Program  
mt Xinu  
2560 Ninth Street  
Berkeley, CA 94710

Internet: nashville@usenix.org  
UUCP: uunet!usenix.org!nashville

Telephone (415) 644–0146  
FAX (415) 644–2680

Be sure to include your postal and electronic mail addresses in all correspondence.

**USENIX Student Attendee Grant**

The Association will award a limited number of travel and accommodation grants to full-time students interested in attending the Dallas Technical Conference (January 21–25, 1991).

Interested **full-time students** should contact the Association’s Executive office for an application form soon. Applications must be returned no later than January 2, 1991.

Ellie Young  
Executive Director
USENIX Winter Conference Program
Grand Kempinski Hotel, Dallas, Texas, January 21–25, 1991

Tutorials

Monday, January 21

An Introduction to the TCP/IP Protocol Suite
Richard Stevens, Consultant

An Introduction to C++
Robert Murray, AT&T Bell Laboratories

UNIX System V Release 4.0 Internals I – File, VM, and Process Subsystems
Steve Buroff, AT&T; Michael Scheer, ProLogic Corporation.

Programming on The X Window System, Ver. 11
Oliver Jones, Saber Software, Inc.

An Introduction to 4.3/4.4BSD Internals
Thomas W. Doeppner Jr., Brown University

Mach Overview
Avadis Tevanian, Jr., NeXT, Inc.

An Introduction to the Internals of the GNU C Compiler (GCC)
Richard M. Stallman, GNU Project

UNIX on Modern Architectures
Curt F. Schimmel, Amdahl Key Computer Labs

An Introduction to Object-Oriented Programming
Dave Taenzer, US West Advanced Technologies

UNIX Technologies of Japan
Jun Murai, Keio University
Hiromichi Kogure, UNIX System Laboratories Pacific, Ltd.

Network Security
Dan Geer, Digital Equipment Corporation;
Jon A. Rochlis & Jeffrey I. Schiller, MIT

Tuesday, January 22

UNIX Network Programming
Richard Stevens, Consultant

Using C++ Effectively
Andrew Koenig, AT&T Bell Labs

UNIX System V Release 4.0 Internals II – Session & Streams and Subsystems and Code
Steve Buroff, AT&T; Mike Scheer, ProLogic Corporation

An Introduction to Programming With the X Toolkit Intrinsics
Paul Kimball & Chuck Price, Digital Equipment Corporation

New Kernel Facilities in 4.3BSD-Reno
Marshall Kirk McKusick and Michael J. Karels, U.C. Berkeley

Mach Virtual Memory Internals
Nawaf Bitar, Hewlett-Packard Company

Advanced Topics in Systems Administration
Evi Nemeth, University of Colorado
Rob Kolstad, Sun Microsystems

Programming in Perl
Tom Christiansen, CONVEX Computer Corporation

Parallel Programming and Scalable Software
Stephen C. Johnson, nCUBE

Network Computing System and Architecture: Overview and Tutorial in Writing Distributed Applications
Nathaniel Mishkin & Paul J. Leach, Hewlett Packard; Richard Mackey, Open Software Foundation

Tuesday, January 22—Half day

C++ Programming Style
Tom Cargill, Consultant

C++ Tactics
Robert Murray, AT&T Bell Laboratories

Special Note to Full Time Students: A limited number of spaces in each class have been reserved for full time students at a special fee. Please contact the Conference office for full details.
Preliminary Technical Program

Wednesday, January 23

9:00–10:30 Opening Remarks and Announcements
Lori S. Grob, Chorus systèmes

Keynote Address
Eben Ostby, Pixar

Eben Ostby joined the Pixar Animation Research and Development Group (then the Lucasfilm Computer Graphics Project) in 1983. With a background in computer science in design, Mr. Ostby has designed and implemented animation and modelling systems for three-dimensional computer graphics. He has also worked on a number of films. He was Technical Director on Knickknack, Tin Toy and Red’s Dream, and a technical contributor to Luxo jr, Young Sherlock Holmes, Flags and Waves and The Adventures of Andre and Wally B. He produced and directed the film Beach Chair, a computer animated mini-travelogue, which is considered a classic in its genre. His current research areas include the procedural generation of plaids.

11:00–12:30 Kernels 1 Chair: Barry Gleeson
Processors, Priority and Policy: Mach Scheduling for New Environments
David L. Black, Carnegie Mellon University
A 2nd Generation Kernelized UNIX
Marc Guillemont, Jim Lipkis, Doug Orr, Marc Rozier, Chorus Systemes
Partitioned Multiprocessors and the Coexistence of Heterogeneous Operating System Environments
Nick Vasilatos, Concurrent Computer Corporation

11:00–12:00 Invited Talk: Toolkit Graphics
Doug Blewett, AT&T Bell Laboratories

2:00–3:30 File System Performance Chair: Trent Hein
Extent-like Performance from a UNIX File System
Larry McVoy and Steve Kleiman, Sun Microsystems
Smart Filesystems
C. Staelin and H. Garcia-Molina, Princeton University
Lessons Learned Tuning the 4.3BSD Reno Implementation of the NFS Protocol
Rick Macklem, University of Guelph

2:00–3:00 Invited Talk: Troff Macro Programming
Sharon Murrel, AT&T Bell Laboratories, Jaap Akkerhuis, mt Xinu

4:00–5:30 Threads & Networks Chair: Deborah Scherrer
Sun OS Multi-thread Architecture
Bringing the C Libraries With Us Into A Multithreaded Future
Michael B. Jones, Carnegie Mellon University
A Tree-Based Packet Routing Table for Berkeley UNIX
Keith Sklower, CSRG, University of CA—Berkeley
4:00–5:30  **Invited Talk: UNIX Security Today and Tomorrow Panel**

*Pat Bahn*, organizer, GTE Government Systems
*Bill Cheswick*, moderator, AT&T Bell Laboratories

**Thursday, January 24**

9:00–10:30  **Interface Tools**

Chair: Tom Duff

An X11 Toolkit Based on the Tcl Language
*John K. Ousterhout*, University of CA—Berkeley

User Interface Construction Based On Parallel and Sequential Execution Specification
*Toshiyuki Masui*, Carnegie Mellon University

$Home Movie—A Home Movie System for Producing Demos on a Sun
*Stephen A. Uhler*, Bellcore Computer Systems Research Division

9:00–10:30  **Invited Talk: Systems Administration Forum—Part 1**

*Rob Kolstad*, Sun Microsystems

9:00–12:30  **AWK Paper and Kernel Panel**

Chair: TBA

Awk As A Major Systems Programming Language
*Henry Spencer*, University of Toronto

Panel—Kernel Directions (1 Hour)

11:00–12:30  **Invited Talk: Systems Administration Forum—Part 2**

*Rob Kolstad*, Sun Microsystems

12:30–2:00  **Lunch**

2:00–3:30  **Programming Tools**

Chair: Marc Donner

Program Loading in OSF/1
*Harminder G. Singh, Larry W. Allen, Kevin G. Wallace and Melanie B. Weaver*, Open Software Foundation

Compiling from Saved State: Fast Incremental Compilation with Traditional UNIX Compilers
*Alastair Fyfe, Ivan Soleimanipour and Vijay Tatkar*, Sun Microsystems

A New Hash Package for UNIX
*Margo Seltzer*, University of CA—Berkeley; *Ozan Yigit*, York University

2:00–3:00  **Invited Talk: Using Distributed Objects**

*Vinny Cahill*, University of Dublin

4:00–5:30  **File Systems**

Chair: Steve Bourne

Evolutionary Path to Network Storage Management
*Antony W. Foster, Robert K. Israel, Arun Taylor, Tracy M. Taylor, Neil Webber*, Epoch Systems

A Highly Available Network File Server
*Anupan Bhide and Stephen P. Morgan*, IBM Research; *Elmootazbellah N. Elnozahy*, Rice University

The OSF/1 UNIX Filesystem (UFS)
*Susan LoVerso, Noemi Paciorek and Alan Langerman*, Encore Computer Corp; *George Feinberg*, Open Software Foundation

4:00–5:30  **Work in Progress Session**

Chair: Lisa Bloch

November/December 1990
Friday, January 25

9:00–10:30 Objects in Action
Advancing Files to Attributed Software Objects
Andreas Lampen, Technische Universität Berlin

Organizing Tools in a Uniform Environment Framework
Axel Mahler, Technische Universität Berlin

The Process File System and Process Model in UNIX System V
Roger Faulkner and Ron Gomes, Sun Microsystems

Chair: Michel Gien

11:00–12:30 Insecurity
Limitations of the Kerberos Authentication System
Michael Merritt and Steven Bellovin, AT&T Bell Laboratories

UNIX Password Encryption Considered Insecure
Philip Leong and Chris Tham, University of Sydney

An Authentication Mechanism for USENET
Matt Bishop, Dartmouth College

Chair: Michael Karels

11:00–12:00 Distributed File Systems Panel
Mike Kazar, Transarc; John Ousterhout, University of CA—Berkeley; Rafael Alonso, Princeton University; Brian Palowski, Sun Microsystems, Moderator: Peter Honeyman, IFS/University of Michigan

Chair: Jan Edler

2:00–3:30 Kernel II
An Experimental Implementation of Draft POSIX
Asynchronous I/O
A. Lester Buck and Robert A. Coyne, Jr., IBM Federal Sector Div.

The Parallelization of UNIX System V Release 4.0
Mark Campbell, Richard Barton, Jim Browning, et.al., NCR Corporation

An Overview of the Integrity S2 NonStop–Ux Operating System
Peter Norwood, Tandem Computers

Chair: Max Meredith Vasilatos

2:00–3:00 Invited Talk: Debugging X and X Toolkit Applications
Paul E. Kimball, Digital Equipment Corporation

4:00–5:30 Distributed Processing
Drums: A Distributed Statistical Server for STARS
Andy Bond and John H. Hine, Victoria University of Wellington

Experience Building a Process Migration Subsystem for UNIX
Dan Freedman, University of Calgary

A Modular Architecture for Distributed Transaction Processing
Michael Wayne Young, Dean Thompson and Elliot Jaffe, Transarc Corporation

For additional USENIX conference information, or pre-registration materials contact:

USENIX CONFERENCE OFFICE
22672 Lambert St., Suite 613
El Toro, CA 92630
Tel: (714) 588–8649
FAX: (714) 588–9706
Call for Proposals to Chair the Winter 1993 USENIX Technical Conference

The USENIX Association is seeking proposals from people interested in serving as the Technical Program Chair for the 1993 Winter Conference, to be held January 25-29 in San Diego, California.

We are seeking an energetic person with the following qualifications:

Excellent administrative and public speaking skills
Knowledge of the timely and appropriate topics in the field
Ability to solicit good panel members and appropriate speakers
Attendance at previous USENIX conferences

Proposals should be brief (one page) including the following points:

a. conference: date and location
b. statement of purpose
c. preference for member of the Board of Directors who will serve as liaison
d. form of submissions: abstracts, extended abstracts, or full papers.

Proposals are subject to approval by the USENIX Board of Directors. Details concerning the schedule, site and call for papers will be worked out with the Association after the appointment of the chair has been made.

Proposals are due: January 4, 1991

Please address all inquiries and proposals to the Association's Executive Director, Ellie Young (ellie@usenix.org).

Request for Proposals to Chair the Large Installation Systems Administration (LISA) Conference

The USENIX Association is once again seeking proposals from people interested in chairing its fifth LISA conference, to be held sometime in the Fall of 1991, or Spring 1992.

We are seeking an energetic person with the following qualifications:

Good administrative skills
A lot of experience in the administration of large installations
Good public speaking skills
Knowledge of what are the timely/appropriate topics in the field
Ability to solicit good panel members/appropriate speakers
Attendance at previous LISA workshops/conferences

Proposals should be brief (1 page) and might include the following:

Statement of Purpose
(e.g., why should we have another one?)
Form of submissions: (e.g., abstracts, extended abstracts or full papers?)
Format (e.g., 2 days of technical sessions, panel sessions, etc.)
List of topics to be addressed, as in the call for papers
Special features, such as having tutorials, BOFs, vendor demos.
List of potential program committee members and/or a co-chair*
Biography and references

Proposals are subject to approval by the Board of Directors. Details concerning the schedule, site and Call for Papers will be worked out with the Association after the appointment of the chair.


Please address all inquiries and proposals to the Association's executive director, Ellie Young (ellie@usenix.org).

*While most USENIX conferences have had an individual chair, proposals requesting a co-chair and/or small program committee are welcome. The chair of the 1990 LISA Conference (Steve Simmons) has offered to be a part of the 1991 program committee as well.
Membership Figures

The last eighteen months has been a period of rapid growth for the UKUUG. Our membership is now over 600 people making us the second largest group in Europe after the French. Okay, these numbers may sound small to you but we're a smaller country!

The Legend Evolves

We held a very successful conference last July at the Royal Lancaster Hotel in London (England). Over 440 people attended the conference, which naturally attracted the interest of the computing press. The likes of Brian Kernighan, Rob Pike, Dennis Ritchie and Ken Thompson were forever fending off questions like "How did you start to ...". Alas, one journal managed to get the event wrong and called it the annual meeting of the European User Group!

I'll conclude by mentioning the time-honoured competition quiz. Rob Pike invited us to invent a suitable insult for the X Window system. From a mountain of entries the judges (Rob, Dennis, Ken) chose the following: X is a terminal illness. X—a trip down "memory" lane enabling the user to experience the performance of a decade ago, but with no discernable advantages. X is just bad NeWS. Light travels half as fast through an X-window. Our thanks to book publishers Addison-Wesley, Prentice-Hall, and Wiley for donating books as the prizes.

Conference Proceedings

Copies of our London conference proceedings (undoubtedly a collector's item for the future) are available from the UKUUG Secretariat (address below) at 50 pounds each.

UKUUG Secretariat Tel: +44 763 73039
Owles Hall Fax: +44 763 73255
Buntingford Net: ukuug@ukc.ac.uk
Hertfordshire SG9 9PL
England

Workshop Videos

We have produced two video programmes on relevant material to those working in the community. Both of these are the result of successful one-day workshops organised by the UKUUG.

- UNIX Security
  A three hour video discussing the following topics: The HACKMAN project; System V/MLS; An analysis of the Internet worm. The Sun Yellow Pages system; Secure RPC; Some myths and facts about security. And more . . .

- UNIX System Administration
  A four hour video discussing the following topics: POSIX developments; System management; Managing X.400 mail systems. Project Athena; System administration in a heterogeneous environment. And more . . .

Each video costs 60 pounds (plus VAT in the UK) and can be ordered from the UKUUG Secretariat (address above) or directly from:

Birkbeck College Tel: +44 71 631 6351
Video Services Fax: +44 71 636 4971
Department of Computer Science
Birkbeck College
Malet Street
London WC1E 7HX England

FaceServer Project

This service is now in full swing and we intend taking it to most major conferences in the future. Note that the name has changed ever so slightly due to a registered trademark problem.

This project is being supported by Acorn Computers Ltd. of Cambridge (England) as part of their on-going commitment to UNIX.

November/December 1990
Long-Term Calendar of UNIX Events

<table>
<thead>
<tr>
<th>Year</th>
<th>Month</th>
<th>Event Description</th>
<th>Location</th>
</tr>
</thead>
<tbody>
<tr>
<td>1990</td>
<td>Dec</td>
<td>UKUUG</td>
<td>Cambridge, UK</td>
</tr>
<tr>
<td>1991</td>
<td>Jan</td>
<td>IEEE 1003</td>
<td>New Orleans, LA</td>
</tr>
<tr>
<td>1991</td>
<td>Jan</td>
<td>Multi-User C Show for Gov't UniForum</td>
<td>Canada, Ottawa, ON</td>
</tr>
<tr>
<td>1991</td>
<td>Jan</td>
<td>*Software Devel. Environ. in UNIX</td>
<td>Grand Kemphinski, Dallas, TX</td>
</tr>
<tr>
<td>1991</td>
<td>Jan</td>
<td>USENIX</td>
<td>Grand Kemphinski, Dallas, TX</td>
</tr>
<tr>
<td>1991</td>
<td>Jan</td>
<td>UniForum</td>
<td>Infomart, Dallas, TX</td>
</tr>
<tr>
<td>1991</td>
<td>Feb</td>
<td>DECUS</td>
<td>Ottawa, Ont.</td>
</tr>
<tr>
<td>1991</td>
<td>Mar</td>
<td>CeBIT 91</td>
<td>Hannover, Germany</td>
</tr>
<tr>
<td>1991</td>
<td>Apr</td>
<td>AFUU CNET</td>
<td>Paris La Defense, France</td>
</tr>
<tr>
<td>1991</td>
<td>Apr</td>
<td>IEEE 1003</td>
<td>Chicago, IL</td>
</tr>
<tr>
<td>1991</td>
<td>Apr</td>
<td>USENIX C++</td>
<td>Washington, D.C.</td>
</tr>
<tr>
<td>1991</td>
<td>Apr</td>
<td>DECUS Muenchien Symposium</td>
<td>Hannover, West-Germany</td>
</tr>
<tr>
<td>1991</td>
<td>May</td>
<td>NCR Unix User Group C</td>
<td>San Antonio, Texas</td>
</tr>
<tr>
<td>1991</td>
<td>May</td>
<td>U 8x/etc C&amp;T /usr/group/cdn;</td>
<td>Toronto, ON, Canada</td>
</tr>
<tr>
<td>1991</td>
<td>May</td>
<td>DECUS</td>
<td>Atlanta, GA</td>
</tr>
<tr>
<td>1991</td>
<td>May</td>
<td>APP/OSE Users Forum</td>
<td>NIST, Gaithersburg, MD</td>
</tr>
<tr>
<td>1991</td>
<td>May</td>
<td>Multi-User C Show UniForum</td>
<td>Toronto, Canada</td>
</tr>
<tr>
<td>1991</td>
<td>May</td>
<td>IEEE TCOS Cpr Workstations</td>
<td>Falmouth, MA</td>
</tr>
<tr>
<td>1991</td>
<td>May</td>
<td>EurOpen</td>
<td>Tromso, Norway</td>
</tr>
<tr>
<td>1991</td>
<td>May</td>
<td>USENIX</td>
<td>Opryland, Nashville, TN</td>
</tr>
<tr>
<td>1991</td>
<td>May</td>
<td>Sun User Group</td>
<td>Atlanta, GA</td>
</tr>
<tr>
<td>1991</td>
<td>Jun</td>
<td>IEEE 1003</td>
<td>Liverpool, UK</td>
</tr>
<tr>
<td>1991</td>
<td>Jun</td>
<td>USENIX</td>
<td>Budapest, Hungary</td>
</tr>
<tr>
<td>1991</td>
<td>Jun</td>
<td>Sun User Group</td>
<td>Sydney, Australia</td>
</tr>
<tr>
<td>1991</td>
<td>Jul</td>
<td>UKUUG</td>
<td>Canada, Montreal, Quebec</td>
</tr>
<tr>
<td>1991</td>
<td>Jul</td>
<td>USENIX</td>
<td>Edinburgh, UK</td>
</tr>
<tr>
<td>1991</td>
<td>Jul</td>
<td>DECUS</td>
<td>San Jose, CA</td>
</tr>
<tr>
<td>1991</td>
<td>Jul</td>
<td>IEEE 1003</td>
<td>Anaheim, CA</td>
</tr>
<tr>
<td>1991</td>
<td>Aug</td>
<td>USENIX</td>
<td>Hilton Square, San Francisco, CA</td>
</tr>
<tr>
<td>1991</td>
<td>Aug</td>
<td>UniForum</td>
<td>Moscone Center, San Francisco, CA</td>
</tr>
<tr>
<td>1991</td>
<td>Aug</td>
<td>EurOpen</td>
<td>Jersey, UK</td>
</tr>
<tr>
<td>1991</td>
<td>Sep</td>
<td>USENIX</td>
<td>Atlanta, GA</td>
</tr>
<tr>
<td>1991</td>
<td>Sep</td>
<td>Sun Users Group</td>
<td>San Antonio, TX</td>
</tr>
<tr>
<td>1991</td>
<td>Sep</td>
<td>IEEE 1003</td>
<td>Washington, DC</td>
</tr>
<tr>
<td>1991</td>
<td>Sep</td>
<td>USENIX</td>
<td>Alaska</td>
</tr>
<tr>
<td>1991</td>
<td>Oct</td>
<td>EurOpen</td>
<td>Amsterdam, Netherlands</td>
</tr>
<tr>
<td>1991</td>
<td>Oct</td>
<td>USENIX</td>
<td>Melbourne, Australia</td>
</tr>
<tr>
<td>1991</td>
<td>Nov</td>
<td>USENIX</td>
<td>San Diego, CA</td>
</tr>
<tr>
<td>1991</td>
<td>Dec</td>
<td>USENIX</td>
<td>San Francisco, CA</td>
</tr>
<tr>
<td>1992</td>
<td>Jan</td>
<td>USENIX</td>
<td>Cincinnati, OH</td>
</tr>
<tr>
<td>1992</td>
<td>Jan</td>
<td>UniForum</td>
<td>Dallas, TX</td>
</tr>
<tr>
<td>1992</td>
<td>Feb</td>
<td>UniForum</td>
<td>Boston, MA</td>
</tr>
<tr>
<td>1992</td>
<td>Mar</td>
<td>USENIX</td>
<td></td>
</tr>
<tr>
<td>1992</td>
<td>Apr</td>
<td>USENIX</td>
<td></td>
</tr>
<tr>
<td>1992</td>
<td>May</td>
<td>USENIX</td>
<td></td>
</tr>
<tr>
<td>1992</td>
<td>Jun</td>
<td>USENIX</td>
<td></td>
</tr>
<tr>
<td>1992</td>
<td>Jul</td>
<td>USENIX</td>
<td></td>
</tr>
<tr>
<td>1992</td>
<td>Aug</td>
<td>USENIX</td>
<td></td>
</tr>
<tr>
<td>1992</td>
<td>Sep</td>
<td>USENIX</td>
<td></td>
</tr>
<tr>
<td>1992</td>
<td>Oct</td>
<td>USENIX</td>
<td></td>
</tr>
<tr>
<td>1992</td>
<td>Nov</td>
<td>USENIX</td>
<td></td>
</tr>
<tr>
<td>1992</td>
<td>Dec</td>
<td>USENIX</td>
<td></td>
</tr>
</tbody>
</table>

*Compiled with the assistance of Alain Williams of the EUUG, Susanne Smith of Windsound Consulting, and John Quarterman of Texas Internet Consulting.

*USENIX Workshops
Summary of Board of Directors’ Meeting
Berkeley, CA, September 24–25, 1990

The regular quarterly meeting of the USENIX Board of Directors was convened in Berkeley, CA on September 24, 1990. Attendance: Rick Adams, Ed Gould, Rob Kolstad, Kirk McKusick, Sharon Murrel, Evi Nemeth, Michael O’Dell, Barry Shein, Alan G. Nemeth, Deborah K. Scherrer, Ellie Young, Judy DesHarnais, Cynthia Deno, Dan Appelman.

**Opening Remarks**

McKusick said that the Board shapes policy of where the organization is going, while still having oversight responsibility to ensure that actions are going on at the executive level. He stressed that the Board is collectively ‘the boss’ of the organization, and should work in concert with the staff and not on an individual basis.

**Anaheim Conference**

DesHarnais reported that revenue was down (a 10% drop in attendance compared to Baltimore).

There was a long discussion on such issues as the need to focus on our changing community, and that perhaps USENIX has remained static. Alan Nemeth said that the needs of the skilled application developers and systems developers are met with our tutorials, but not with the technical sessions, and that more tutorials that address their needs are needed as well. Kolstad felt that the workshops drain conferences, since most companies only allow one meeting per year. Young was requested to pull a sample from our database to see who returns to the conferences.

**Dallas ’91 Winter Conference.**

Scherrer reported on behalf of Grob that 84 papers were submitted and 31 had been accepted. There were a lot of papers on security, file systems, and threads. The committee had a great desire to see application papers but none were accepted. Gould said that we need to expand the program.

**Nashville ’91 Summer Conference.**

Scherrer said that the program would have a multi-media focus with a theme of “Interfaces of the Future”. She, as program chair, would be working on getting the committee working proactively on sessions. We should emphasize to the media people that they need to reach the people designing these platforms and environments. DesHarnais said we had space for 22 tutorials. McKusick asked if we should consider reducing the number of tutorials offered? Alan Nemeth cautioned that we should not lose our role as providing tutorials that a commercial organization would not offer. After more discussion, the general consensus was that the maximum number of quality tutorials offered is subject to the physical limits of the site, and we should not work to fill the rooms.

**Security Workshop.**

Young reported that 260 people attended, and the event was no longer a ‘workshop.’ She would be working with Bishop and a group of people who are interested in putting together a conference.

**Long-Term Conference Planning.**

DesHarnais said that we need to decide something now regarding a potential move to the mid-March dates with UniForum in 1995. If we move, prices would be higher. There was a discussion about changing the number of conferences. Alan Nemeth said we need to have a long-term worldview of who our audience is: a broad conference once a year model argues that our audience is the same as always, and that a two conferences per year model asks: do we intend to really broaden our audience, and what are we going to be in five years? We need to attract the application programmers coming from different systems. Shein suggested that we look at subsets of our audience, and pick 2–3 application areas and promote them, e.g., transaction processing, small business applications. The multimedia theme in Nashville will attempt to do this, and if it’s successful we need to do follow-up in the same general vein. Everyone indicated a desire to stay with the January and June format for 1995, and none favored going to single meeting per year.

**Executive Director’s Report.**

Young went over her report on staffing disruptions over the past few months, the hiring of
the Cynthia Deno as exhibit manager, Dan Klein as tutorial coordinator, Alain Hénon as managing editor of monographs/series program, and her intent to hiring a full-time deputy executive director. It also included a report on the selection of the MIT Press as the publisher for the USENIX monograph series.

Journal Report.

O'Dell reported that issue 3:3 was just out which put the journal back on schedule, and 3:4 was in limbo pending arrangements for a new typesetter. Gould said that the response to the special music issue has been loud and positive, and O'Dell was congratulated.

Budget 1990.

Young went over the current budget that reflected activity through July, as well as projections for the year-end. While income from Security and LISA conferences would be more than projected, revenue from C++ conference, both technical conferences, and the PDS program was less than projected, while expenses at the conferences were higher. A discussion ensued regarding the projected deficit at year-end.

Budget 1991.

Young presented the preliminary draft budget which reflected income would be flat, and that the Board would need to decide ways in which to reduce the projected deficit, so that a final budget could be prepared for the coming year. Alan Nemeth suggested that the Board target a budget figure for fiscal year 1991 that gives -0- excess revenue over expenses (balanced).

McKusick suggested that we go through revenues and expenses line by line.

Membership dues.

Young's analysis presented various scenarios for raising dues and comparisons with other organizations. Nemeth pointed out that the journal had been a large extra expense in the past three years, while dues had remained constant. After much discussion, it was decided to change dues to: (Individual: $50; Student $15; Supporting: $1,000; Educational $150; Corporate: $300).

It was agreed that we should target a small increase in prices for the proceedings to members and a larger one to non-members. Young said she would work on reducing expenses for the executive office and conferences.

Face Saver Project.

It was decided that the FaceSaver project would most likely be funded for the Nashville conference, with final approval being made at the January board meeting.

Press Relations Proposal.

It was decided not to fund a press room in Dallas, that Young should work with Frey on new ways to expend the monies in her proposal for press releases, and funding activity for Nashville would be decided at the next meeting.

Young congratulated the Board on their efforts thus far, since it looked as though the projected deficit had been cut by at least half. She felt that in the short run small raises regularly of member dues, tutorial, workshop and conference fees, along with cutting costs for conferences might produce a balanced budget.

Workshop and conference fees.

After much discussion concerning workshop, tutorial, and tech session fees, it was decided that conference technical fees be raised to $225 for members and $275 for non-members beginning with the Winter '91 technical conference and include the mini-conferences.

Standards.

McKusick stated that the ISO monitor project was the cornerstone of our cooperative efforts with the EUUG, and we should approve the project if matching funds come from the EUUG. It was agreed to do this.

McKusick presented Quarterman's proposal. Alan Nemeth affirmed the value of the snitch reports. After much discussion, it was generally felt that the proposal in its present form was not acceptable, and that if he and others want to submit other proposals for next year we would entertain them. Funding for the snitch reports was approved.

Next Meeting.

It will be held Sunday, January 20, at the Grand Kempinski Hotel in Dallas, Texas.
Book Review
Designing Object-Oriented Software
by Rebecca Wirfs-Brock, Brian Wilkerson, and Lauren Wiener
Reviewed by Edward Gordon
Data Systems Associates

Introduction
In an effort to create a more useful methodology for designing systems, the academic community has created the concept of object-oriented design. Initially, engineers would design their systems using flow charts for software, and state diagrams for hardware. In an interesting synthesis, Ed Yourdon created the “Yourdon” method, which takes the best features of flow charts and the best features of state diagrams. But, the available methods were much too linear, not allowing for the free flow of ideas necessary for performing system design. The need for a coherent system design methodology has spawned the development of object-oriented design.

Object-oriented design has many proponents and has been popular for the last five years. Two of the more notable proponents have been Bertrand Meyer, in the work Object-Oriented Software Construction and Brad J. Cox in Object Oriented Programming: An Evolutionary Approach. This book is the latest in a series of works that explain the methodology.

The book presents an evolving view of object-oriented design. The authors first present the motivation behind object-oriented design, defining their terms, and introducing the graphing mechanisms. There is a strong emphasis on describing how the classes interact, and the relationship between the different class types. In order to explain the use of the object-oriented mechanisms the authors evolve their description of their methodology using case study. The first case study is of an automated teller machine.

This study presents the design methodology to the reader. When the software design is completely developed and the case study complete, the authors present a second case study that assumes the reader’s understanding of object-oriented design and shows the thought process that a skilled designer would use when producing a system with the object-oriented methodology. This case study describes an online documentation system.

The appendices are valuable. The first appendix contains a synopsis of the design methodology, and the tools, graphing methods, and terms necessary for utilizing the system. The second and third appendices provide the graphs for the case studies. Finally, the fourth section presents some exercises for the student to use to practice object-oriented design techniques.

Conclusion
It should be remembered that a clear, concise methodology is necessary for providing system and software designs. Wirfs-Brock, Wilkerson, and Wiener provide a set of techniques for producing an integrated object-oriented design that relies on the fundamental concepts upon which the object-oriented design school of thought has been built. The authors proceed to expand upon the basic techniques and produce a cohesive whole with which system software design can be performed.
Book Review

Software Engineering: Concepts and Management
by Allen Macro

Reviewed by Robert C. Birss
Sun Microsystems
sun.com/bbirss

Software Engineering: Concepts and Management is the first of a five-volume series on “practical software engineering topics,” to be followed by volumes on specification and feasibility, design, implementation, and software estimating and technical quality. Allen Macro is both general editor of the series and author of the first and last volumes. The series is intended as a basis for guidelines in software engineering for practitioners, “for the comprehension of others involved in software development” (page ix), and as a text for academic and industrial courses in software engineering.

As in his earlier volume, The Craft of Software Engineering, Macro defines software engineering as

the establishment and use of sound engineering principles and good management practice, and the evolution of applicable tools and methods and their use as appropriate, in order to obtain software that is of high quality in an explicitly defined sense. (page 31)

He attempts a synoptic exposition of concepts and definitions, the modalities of software development, and software management—with strong emphasis on quality [“Software quality is the whole of the matter, so far as the process and outcome of software engineering are concerned.” page 412]. The sections on managing for change, managing for quality, and organization and personnel factors are particularly good. However, the book is surprisingly superficial on implementation issues. Take, for example, code reviews. Presumably, they will be covered in depth in the forthcoming implementation or quality volumes. But that they rate only passing mention in the one paragraph on static testing in this volume makes it a questionable choice as a “stand-alone” book for any audience.

The writing is literate and witty, as can be expected of someone who writes that “solemnity and software are sad bedfellows” (p. 471). It is also rather British, which may sometimes make things a bit opaque for the American reader.

The book contains four appendices: a consolidated case study on a chess-playing program, sample exam questions, a glossary of terms, and a list of references. Macro sees the questions serving as either an exam for students, a tool for measuring “the scope of subject awareness in a department” (page 517), or individual questions requiring short written answers at interviews. The list of references would, perhaps, be more useful if it were a general bibliography on software engineering. It is hard to see how Zipf's The Psychology of Language or Russell’s A History of Western Philosophy will give the curious or the perplexed reader much help with sorting out just what software engineering is or how to make it happen—even though our author cites both works to good effect.

I was not familiar with the author, so when I unwrapped the book, I thought “What an appropriate name for someone writing about software.” Then I wondered if it was a typo—for “Marco.” Unfortunately, the text does not resolve the question, since some of the references to the author’s earlier book give his name name as “Macro” and some give it as “Marco.” Of course, “McCabe” is sometimes “McCable”, so at least our author isn’t the object of a typesetter’s personal vendetta.

The five volumes together may well provide a thorough examination of software engineering. This book alone is not satisfactory.
An Update on UNIX–Related Standards Activities

Jeffrey S. Haemer
Report Editor, USENIX Standards Watchdog Committee

Reports are done quarterly for the USENIX Association by volunteers from the individual standards committees. The volunteers are familiarly known as snitches and the reports as snitch reports. The band of snitches, John Quarterman and I make up the working committee of the USENIX Standards Watchdog Committee. Our job is to let you know about things going on in the standards arena that might affect your professional life — either now or down the road a ways.

We don’t yet have active snitches for all the committees and sometimes have to beat the bushes for new snitches when old ones retire or can’t make a meeting, but the number of groups with active snitches continues to grow (as, unfortunately, does the number of groups).

If you’re active in any standards-related activity that you think you’d like to report on, please drop me a line. We need snitches in several 1003 groups, and nearly all of the 1200-series groups. We currently have snitches in X3J16 (C++) and X3B11 (WORM file systems), but there are probably X3 groups that USENIX members would like to know about that we don’t even know to look for watchdogs in. I also take reports from other standards activities. This quarter, you’ve seen reports from the WG-15 TAG (the U.S.’s effort in the ISO POSIX arena), from the NIST Shell-and-Tools FIPS meeting, and from the USENIX Standards BOF.

If you have comments or suggestions, or are interested in snitching for any group, please contact me (jsh@usenix.org) or John Quarterman, USENIX Standards Liaison (jsq@usenix.org).

The USENIX Standards Watchdog Committee also has both a financial oversight committee — Ellie Young, Alan G. Nemeth, and Kirk McKusick (chair); and a policy committee — the financial committee plus John S. Quarterman (chair).

An official statement from John Quarterman:

The basic USENIX policy regarding standards is:
• Collect and publish contextual and technical information such as the snitch reports that otherwise would be lost in committee minutes or rationale appendices or would not be written down at all.
• Encourage appropriate people to get involved in the standards process.
• Hold forums such as Birds of a Feather (BOF) meetings at conferences and standards workshops.
• Write and present proposals to standards bodies in specific areas.
• Occasionally sponsor other standards-related activities, including as White Papers in particularly problematical areas, such as IEEE 1003.7, and contests, such as the current Weirdnix contest.
• Very occasionally lobby organizations that oversee standards bodies regarding new committee, documents, or balloting procedures.
• Sponsor a representative to the ISO/IEC JTC1 SC22 WG15 (ISO POSIX) standards committee, jointly with EUUG (the European UNIX systems Users Group).

There are some things we do not do:
• Form standards committees. It’s the USENIX Standards Watchdog Committee, not the POSIX Watchdog Committee, not part of POSIX, and not limited to POSIX.
• Promote standards.
• Endorse standards.

Occasionally we may ask snitches to present proposals or argue positions on behalf of USENIX. They are not required to do so and cannot do so unless asked by the USENIX Standards Watchdog Policy Committee.

Snitches mostly report. We also encourage them to recommend actions for USENIX to take.
Report on IEEE 1003.2: Shell and tools

Randall Howard <rand@mks.com> reports on the July 16–20, 1990 meeting in Danvers, MA:

Background on POSIX.2

The POSIX.2 standard deals with the shell programming language and utilities. Currently, it is divided into two components:

- POSIX.2, the base standard, deals with the basic shell programming language and a set of utilities required for application portability. Application portability essentially means portability of shell scripts and thus excludes most features that might be considered interactive. In an analogy to the ANSI C standard, the POSIX.2 shell command language is the counterpart to the C programming language, while the utilities play, roughly, the role of the C library. In fact, because POSIX.2 provides an interface to most of the features (and possibly more) of POSIX.1, it might also be thought of as a particular language binding to the soon-to-be language independent version of that standard. POSIX.2 also standardizes command-line and function interfaces related to certain POSIX.2 utilities (e.g., popen(), regular expressions, etc.), as discussed in detail in the snitch report for the Snowbird meeting. This part of POSIX.2, which was developed first, is also known as "Dot 2 Classic."

- POSIX.2a, the User Portability Extension or UPE, is a supplement to the base POSIX.2 standard. Not a stand-alone document, it will eventually be an optional chapter and a small number of other revisions to a future draft of that base document. This approach allows the adoption of the UPE to trail Dot 2 Classic without delaying it. The UPE standardizes commands, such as vi, that might not appear in shell scripts but are important enough that users must learn them on any real system. It is essentially an interactive standard that attempts to reduce retraining costs caused by system-to-system variation.

Some utilities have interactive as well as non-interactive features. In such cases, the UPE defines extensions from the base POSIX.2 utility. An example is the shell, for which the UPE defines job control, history, and aliases. Features used both interactively and in scripts tend to be defined in the base standard.

Together, Dot 2 Classic and the UPE will make up the International Standards Organization’s IS 9945/2 — the second volume of the proposed ISO three-volume standard related to POSIX.

Status of POSIX.2 Balloting

Draft 10 of Dot 2 Classic was sent out during July in a recirculation ballot. Recirculation means that objections need only be considered if they are existing unresolved objections or are based on new material. Other objections will be considered at the whim of the Technical Editor.

Draft 10 is an imposing, if not intimidating, 780 pages, made even denser by the decision to remove much white space in a (vain) attempt to save paper. Ballots are due by September 10. Unfortunately, the recirculation ballot materials arrived at my organization on August 17th, giving our group barely three weeks to review this massive document.

The technical editors and others working behind the scenes (Hal Jespersen, Don Cragun, and others) have done an admirable job of diff-marking changes and producing personalized lists of unresolved objections for each balloter. In addition, all 96 pages of unresolved objections are provided. However, the amount of new material that has never been reviewed and the major reorganization means that Draft 10 bears much less resemblance to Draft 9 than one might hope. That, combined with balloting on the UPE, has put many balloters — myself included — in balloting overload.

If a recirculation simply means forming opinions on my (and other) unresolved objections, then the time period is quite reasonable. However, as I shall describe below, Draft 10 is so changed from the previous drafts that it deserves to be read practically from cover to cover, and the recirculation deadline does not provide adequate time for that task. The changes fall into a number of categories:
New Utilities: For example, a superset of the traditional `od` replaced the Draft 9 `hexdump` which was `xd` in Draft 8. "Pathchk" and "set -o noclobber" have replaced `create` from Draft 9 and `validfnam` and `mktemp` from Draft 8. Such examples demonstrate that Draft 10 is not mature and needs more consideration to achieve consensus.

Expanded Material: Previous descriptions of such utilities as `awk`, `sh`, `bc`, etc., were neither sufficiently comprehensive nor sufficiently complete to be of the quality demanded of a standard. In the latest draft, these descriptions have been fleshed out, and include much more detail on operator precedence, interactions, subtle semantics, and so on. This is clearly a step in the right direction, but adds to the job of reviewing Draft 10.

Internationalization: While the `localedef` and `locale` utilities remain, they have changed substantially. I personally support including these features, but am concerned that these are being designed during the balloting process which is, if anything, worse than design-by-committee. Overall, balloting-group reaction to these utilities ranges from impassioned pleas for their removal to requests for greater functionality (complexity) to handle ever more arcane aspects of the internationalization problem.

Chapter 2: Chapter 2's front matter is substantially reorganized and more voluminous. This chapter contains definitions, utility syntax information, requirements imported from POSIX.1, the definition of a locale, description of basic and extended regular expressions, etc.. Utility descriptions seem to be getting shorter, with more and more pointers to Chapter 2. This is a good trend, as long as balloters adequately consider the chapter's technical contents.

Status of POSIX.2a Balloting

The first formal ballot on POSIX.2a UPE Draft 5 was due in the IEEE offices by August 16th. Unfortunately, the UPE is laced with references to definitions and concepts largely defined in Chapter 2 of Draft 10. I did not receive my Draft 10 until after the UPE balloting was due to be returned. This hinders any attempt to review these two documents as a single entity — which is what they will eventually become.

The UPE is starting to mature: it's converging. The major controversy is scope — as it has been throughout the UPE's entire life. This draft aligns itself more closely to Dot-2-Classic in many ways, which leads me to believe that combined review is essential to its understanding.

A few utilities remain contentious:

- `nice`, `renice`: These require underlying functionality absent from POSIX.1, although POSIX.4 has `setscheduler()`, which allows applications to set priority and scheduling algorithms.

Some working and balloting group members adamantly resist any attempt to add utilities that are not implementable on top of a bare POSIX.1. Others view the UPE as addressing what users type, regardless of underlying implementation. I am in the latter camp, not the least because other working groups, such as POSIX.4, have not yet standardized a utility interface, leaving a void which the much-maligned UPE group is most able to fill. (It is telling that implementing `df` and `ps` is impossible using only POSIX.1 functions, yet there is little opposition to including either utility.

- `ps`: The description for this utility was an interesting amalgam of two incompatible visions of how `ps` output should be formatted — that in Draft 4 and that in Draft 5. A correction should have been issued during balloting, so that balloters could concentrate on the real issues of what should be the scope of the `ps` utility.

- `patch`: This utility differs from many others; its origins are in the public domain rather than in a traditional UNIX variants. As a result, many people feel that `patch` is worthwhile, but not mature enough to standardize.

- `lint89`: This utility is optional, largely because it is controversial for a number of reasons. Obviously, the very name `lint89` is painfully bureaucratic. Furthermore, many feel that ANSI C makes it unnecessary; moreover, any remaining required functionality rightfully belongs as an additional option in the `c89 (cc)` utility. Some point to
existing practice. But what is existing practice when the utility’s name is lint89?

[Editor: On the other hand, it may prove indispensable in detecting portability problems in lex89- and yacc89-generated code. Parenthetically, Draft 10 calls these lex and yacc, but that must just be a temporary oversight; the utilities obligatorily have ANSI C input and output. (One assumes we'll escape c89tags because ctags can be made to work with both flavors.)]

- compress: The inclusion of this utility remains controversial because of the Unisys patent on the particular variable of Lempel-Ziv compression used by traditional implementations of this utility. The working group appears to be divided on the subject of basing a standard on patented material — no matter what the licensing fees are. There is, however, general agreement that it is preferrable to remove compress entirely rather than “invent” some new compression algorithm. Therefore, it appears that a pax-like compromise, of having a single interface to a number of competing formats or algorithms, is not widely supported. [Editor: see Andrew Hume's X3B11 report for another wrinkle on data compression.] Clearly, this issue will have to be resolved with further information from Unisys lawyers during the balloting process.

Status of the Danvers Meeting

The Danvers working group dealt with neither Dot 2 Classic nor the UPE. Instead, at POSIX.3.2’s request (that’s the subgroup of Dot 3 producing test assertions for Dot 2), we met jointly to co-develop test assertions for Dot 2 Classic. This work is a consequence of the SEC’s recent decision requiring each POSIX working group to develop its own test assertions and ballot them with the standard. It also stems from Dot 3’s frustration over the (inadequate) way Dot 2 addressed testing. For example, automated testing of Ip is impossible; it can only be tested by a human test procedure. Our working group should have explored the implications of this before subjecting POSIX.3 to that task. (Some utilities can only be tested manually, but the working group defining that utility should likely put something to that effect in the Rationale or History of Decisions Made to confirm to the testing people that they knew this.)

The three days of working with Dot 3 were a real learning experience for our working group. Nonetheless, many of us had our fill of test assertions that week. I’m also concerned that a three-day meeting cost my company nearly as much as a five-day meeting would have. In the future, I would prefer to see schedules that make productive use of the entire working week.

Report on IEEE 1003.3: Test Methods

Doris Lebovits <lebovits@attunix.att.com> reports on the July 16–20, 1990 meeting in Danvers, MA:

Overview

Dot three’s job is to do test methods for all of the other 1003 standards. The group’s work, whose first parts are now in ballot, specifies the requirements for OS conformance testing for our industry and for NIST. This makes our balloting group, our technical reviewers, and our schedules worth watching. Pay attention, also, to what comes out of the Steering Committee on Conformance Testing (SCCT). Their projects and decisions will be interesting and important.

This was the working group’s seventeenth meeting. As usual, we reviewed the ballot status of P1003.1 test methods, worked on P1003.2 test methods and reviewed steering committee activities. Technical reviews were done on parts I and II and the group developed assertions for part III. Participants from the usual companies attended (AT&T, NIST, OSF, Mindcraft, IBM, DEC, HP, Data General, Cray Research, Unisys, Perennial, and Unisoft, Ltd.), as did an assortment of P1003.2 members (see below).

Document structure

Currently, our evolving document has three parts: Part I is generic test methods, Part II is test methods for measuring P1003.1 conformance, including test assertions, and Part III contains test methods and assertions for measuring P1003.2 conformance.

After the ballot, each part will become a separate standard. Part I will be published as IEEE P1003.3, Part II as IEEE P1003.3.1, and Part III as IEEE P1003.3.2.
Ballot status

Draft 11 of the current ballot, which was recirculated to the (approximately) ninety-member balloting group late in February, closed balloting March 23. Of the respondents, 19 disapproved with substantive negative comments. This met the two-thirds response requirement, but falls short of the needed two-thirds approval.

A recirculation ballot for P1003.3 Draft 12, which is the revision of Part I of Draft 11, began August 28 and is expected to close September 28, 1990. The recirculation of P1003.3.1 Draft 12 (Part II) will be conducted at a later date.

On the first and last days, the technical reviewers worked on ballot objections to Part I and Part II. All Part I objections and most Part II objections were resolved. The definition of an untested assertion was reviewed and a permanent rationale will be included in Part I.

P1003.2 verification

This was our fifth meeting working on the verification standard for the P1003.2 standard. The assertion writing and review were done jointly with the P1003.2 working group.

The whole P1003.3 and P1003.2 working groups worked jointly on defining test assertions based on P1003.2 Draft 10. They worked in three small breakout groups. The joint group (P1003.2 plus P1003.3) also met in plenary session several times to discuss progress and small-group issues. Progress was slow in the beginning, since most of the P1003.2 working group were not familiar with test assertions, but by the end of the week we had discussed and resolved several issues. Some examples:

• Do we need to state assertions in P1003.3.2 explicitly that duplicate P1003.3.1? (Yes.)
• Must we test locale variables for every locale-sensitive interface? (They should be tested when their behavior is clearly stated for a utility.)
• Should assertions for multiple operands be consistent? (Yes.)

Lowell Johnson (Unisys) is Secretary of the P1003.2 Test Methods activities, and Andrew Twigger (Unisoft Ltd) is Technical Editor. Ray Wilkes, the former Chair, has changed jobs and is no longer able to attend regularly, so Roger Martin is actively looking for a replacement.

Steering Committee on Conformance Testing (SCCT)

The SCCT is supposed to alleviate the increasing dot-three work load that all the other proliferating groups are creating. Their job is coordinating the activities of all test-methods groups, monitoring their conformance to test methods, and writing Project Authorization Requests (PARs). Currently, its members are Roger Martin (NIST, Steering Committee Chair), Anita Mundkur (HP), Andrew Twigger (Unisoft Ltd), Bruce Weiner (Mindcraft), Lowell Johnson (Unisys) and the newest member, John Williams (GM). That there is a new member in the steering committee is very important, especially because John is from GM, the largest user voice other than the U.S. government.

The steering committee did not have anything for the working group to review. It is still documenting procedures, and Roger is still clarifying which standards the working group will address.

Report on IEEE 1003.5: Ada bindings

Jayne Baker <cgb@d74sun.mitre.org> reports on the July 16-20, 1990 meeting in Danvers, MA:

Introduction and Overview

P1003.5 completed the last touches on Draft 6 of the Ada Language Binding, before sending it to ballot, and considered our options for P1003.5 work beyond balloting. We also addressed the International Standards Organization's (ISO's) refusal to accept and register our draft and revised our balloting schedule.

Final Document Modifications

This meeting was our last chance to modify our document without a formal IEEE ballot to justify that change. We spent a large portion of the meeting editing Draft 5, chapter by chapter. Draft 6 will ballot in less than two months, so document stability was guarded, but we considered a few proposals for changes.
• David Emery's Process Group ID as a Separate Type proposal addresses the P1003.1 intention and underlying semantics with respect to Process_Group_ID. Specifically, the proposal recommends that Process_Group_ID be a separate type, or a derived type at a minimum, rather than a part of Process_ID. Dave believes that P1003.1 intended Process_ID and Process_Group_ID to be treated as separate types. This perception is supported by a few operations, such as Wait_for_Process_Group, which suggest the two types are indeed separate. Representing the two types separately would help prevent confusing them. Making them separate would also allow function overloading. For the most part, the group agreed, but felt that the types really do behave more like derived types than separate types.

There was some resistance to adopting this proposal because of the number of changes it would require in sections 3 and 4 (Process Primitives and Process Environment), but there was also opposition to handing the problem off to the balloting group. We finally decided to consult with the Language Independence group.

• A proposal submitted by Mars Gralia, of Applied Physics Laboratory, Clarify Functional Option 'FIFO', addressed a topic presented in section 8 (Language-Specific Services for Ada). This proposal was accepted because it introduced flexibility that makes it easier for P1003.5 to support the P1003.4 work in the future.

• Mars also offered a Simplify and Unify proposal, which provoked lengthy, somewhat heated discussion. Specifically, the section 8, Is_append, function returns yes/no, to support an existing application, but there is a naming convention P1003.5 supports that requires Is_Append to return a boolean; indeed, the append function in section 6 (Input and Output Primitives) already returns boolean.

Our priorities are

• Consistency with the Ada language.
• Consistency between the Ada and POSIX portions of the document;
• Consistency with existing implementations.

Unfortunately, some of these conflict with others in this case. The good news is we may not have to decide what to do: Ada Interpretation (AI) 544 addresses this issue. However, we did not know, and could not find out, the complete resolution of the AI in Danvers. Moreover, Dave Emery and Hal Jespersen, who are preparing the document for ballot, don't have time to make all the changes Mars's proposal would require between now and ballot circulation. Jim Lonjers suggested that Mars submit a negative ballot on this issue, which would let the ballot-resolution group construct a decision consistent with the AI during ballot resolution.

Future Work

When Draft 6 enters the IEEE ballot process, the ballot resolution group becomes responsible for ballot coordination and resolution, and the working group is freed to submit new Program Authorization Requests (PARs). IEEE policy lets a group operate for six months without a PAR, so we have to do our job quickly.

We listed possible new work areas, then ranked them based on our effectiveness in the area, the work's importance, and the effort required. Here is our list.

• Test Assertions for P1003.5
• A straw-man vote shows the test assertions work as the number one issue, though we suspect neither our corporations nor our individual bosses will be very interested in the work. However, test assertions are a National Institute of Standards and Technologies (NIST) requirement, which may increase corporate interest levels. We do have total control over the test assertions work, and have been directed by the SEC to address it prior to our first round of IEEE ballot. To prevent a delay to the first round of IEEE ballot, the SEC has allowed us to include a “plan” for identifying and accomplishing the test assertions portion of the document, rather than the actual test assertions.
• Shells & Utilities (Ada binding to P1003.2)
• Language Independence (Helping P1003.1 create a language-independent specification for 1003.1–1988 and 1003.1–1990.)
The Shell and Tools work and language independence ran close seconds. The Shells & Tools work received a high ranking in the straw-man vote because we feel that the work is do-able and that our effectiveness in the area would be high. Moreover, compared to other areas (e.g., the P1003.4 work), the level of P1003.5 effort required would be low. Language-independence ranked high as it is critical to both the current P1003.5 work (see ISO Acceptance and Registration, below) and the POSIX effort as a whole. The people working the language-independent issues are asking for our input now. Moreover, without our input the resulting language-independent work could adversely impact us, and P1003.5 might not have the voting clout during balloting to block anything particularly awful. Several members interested in these issues are already holding Birds-of-a-Feather meetings with the P1003.1 language-independent group.

- Threads issues (Ada binding to P1003.4a) and Real-Time Extensions (Ada binding to P1003.4)

This area generates the most interest among working group members, several of whom have been working with P1003.4 for some time. Ted Baker, former P1003.5 snitch, has written a document on the subject, Real-time Extension for Portable Operating System Ada Binding - Version 0.0 for the U.S. Army HQ CECOM Center for Software Engineering, and provided us with copies for review and consideration. Group consensus is that if we rush into this area, we are likely to stumble over language-independence issues, so we will work with the P1003.4 language-independence small group until their specification is well along, and then begin work on the Ada binding in parallel with its completion.

ISO Acceptance and Registration

Jim Isaak, Technical Committee on Operating Systems (TCOS) Chairman, reported to P1003.5 that ISO declined to accept and register P1003.5 at the recent Subcommittee 22 (SC22) Paris meeting. Their primary reason was the lack of a language-independent specification for P1003.1. How, they asked, can a language-dependent binding exist without a base, language-independent specification? We had also failed to use Working Group 11’s procedure-calling mechanism to generate our language bindings. (The WG11 approach produces a direct, language-dependent binding to a language-independent specification.) P1003.9, FORTRAN binding to P1003.1, suffered the same fate for the same reasons.

For now, we will provide a copy of P1003.5 Draft 5 to SC22 for their review and comments regarding potential registration problems in the future. To address WG11 concerns, Jim Isaak, POSIX Strategy Director — note the different hat — recommended we also forward a copy of Draft 5 to WG11 for review. David Emery and I, both of MITRE, will follow up with a white paper explaining, with examples, why a one-to-one, direct mapping of the functionality described in the language-independent specification to the language-dependent binding is not always optimal, and that a complete (i.e., thick) language-independent specification and a reference-type (i.e., thin) language-dependent binding is neither practical nor possible for some languages.

Finally, we will formally submit Draft 7 (or later) to SC22, requesting they recommend it for ISO acceptance/registration as a Committee Document (CD). (CD has replaced “Draft Proposal” or DP.) The earliest this could happen is January 1991.

Why not Drafts 5 or 6? A new policy, intended to promote document stability requires one IEEE ballot cycle before submitting a draft for ISO registration.

IEEE Ballot Issues/Schedule

We met with Jim Isaak and Lorraine Kevra, the new TCOS Balloting vice-chair, to discuss the IEEE balloting process and our balloting schedule.

P1003.5 produced a schedule for achieving simultaneous IEEE and ISO ballot at the April/Salt Lake City meeting (see my report from last quarter), but because of the problems with ISO, described above, we have revised this schedule.

Approximately 450 people joined the P1003.5 ballot group. Only 61 of those people are
POSIX participants; that is, only one-sixth of all POSIX participants (from all working groups) signed up for our ballot group! The other 390-odd participants are SIGAda members. We are very pleased with this response.

Ballot-group formation closed on August 6. Confirmation to applicants was originally scheduled for August 8. Because of the large number of non-POSIX balloters, this date was pushed back to about August 17, but anyone who signed up and has still not received confirmation should contact Bob Pritchard at the IEEE Standards Office, 445 Hoes Lane, Piscataway, NJ 08855, (908) 562-3811.

Now that ballot group formation has closed, the group cannot expand. Only people who fail to respond to the initial ballot can be removed (“abstain” is not a non-response); ballot group members are not required to respond to re-circulation ballots.

Bob Pritchard will mail Draft 6 to the P1003.5 ballot group on September 10, 1990. The distribution takes a minimum of two weeks.

The ballot period officially begins on September 24, 1990, and closes October 24, 1990. This allows the ballot group at least four weeks for review. Being realistic, we imagine that not everyone will complete their document review. To prevent the uneven coverage that would result from 450 reviewers reading the document from front to not-quite-back, our cover letter requests that reviewers begin their reviews at different spots, using a scheme based on the first letter of the reviewer’s last name.

If people do not return their ballots by October 24, the IEEE office may send a follow-up letter to the ballot group members requesting that they return their ballots.

Steve Deller, of Verdix, will do all necessary coordination with organizations listed on our PAR. Jim Lonjers, of Unisys, with Lorraine Kevra’s help, will coordinate ballot resolution. Each chapter will have someone responsible for its resolution, but alternates for each chapter are absolutely critical. Jim Isaak says that, based on his experience, we should assume 20% of the people who do ballot resolution will, for some reason, prove unable to complete their portion of the task.

Jim Lonjers will provide the last ballot to the technical reviewers by December 5, 1990. The ballot resolution group will meet at the Tri-Ada meeting in early December to determine how close we are to achieving the 75% minimum acceptance. At that same meeting they will also coordinate ballot responses to objections which cover multiple chapters and objections which produce conflicting responses. We believe they will have resolved the last ballot by January 11, 1991, and a re-circulation ballot is tentatively scheduled for the April 1991 POSIX meeting time frame.

In IEEE re-circulation ballot, two sets of material are returned to the ballot group:

- the changes made to the document (either a set of changes, or a new document with change bars), and
- the unresolved objections.

IEEE policy does not allow the balloters’ names, companies, or company locations to be returned with the unresolved objections packet; to maintain anonymity, ballot comments are numbered, and individual balloters notified of their own ballot comment numbers. (IEEE and ANSI do maintain balloters’ names, companies, and company locations to detect corporate ballots, where and if they occur.) The balloting group gets at least ten days to review the re-circulation ballot, though they can be given more time if the size of the re-circulation material and the document being balloted warrant it.

Miscellany

Eight Next Generation Computer Resources (NGCR) representatives gave working-group participation quite a boost. Although NGCR people have the bond of all being NGCR representatives, they are not employed by a single employer, but are from all over the United States, and they possess individual interests and strengths. In the past, our core group has only been about a dozen people, so we are pleased by NGCR’s interest and participation, and eager to work with them.

In April 1990, David Emery went to Sweden, to meet with the Ada 9x committee group dealing with secondary standards and setting priorities of those standards. Secondary standards are those standards not contained within the language itself (i.e., not in the Ada Language Reference Manual). POSIX was a very high priority secondary
standard. The next Ada 9x committee meeting will be at the SIGAda meeting in Los Angeles in August. Dave is heading a panel presentation on the P1003.5 Binding at this meeting. The chapter authors will also be a part of this panel.

At July POSIX meeting, P1003.5 expressed its special thanks to Dave for his better-than-excellent job as our Technical Editor. He has contributed significant time (much of it his own) and effort to the P1003.5 work, and we appreciate it.

Report on ANSI X3B11.1: WORM File Systems

Andrew Hume <andrew@research.att.com> reports on the July 17–19, 1990 meeting in Murray Hill, NJ:

Introduction

X3B11.1 is working on a standard for file interchange on write-once media (both sequential and non-sequential (random access)): a portable file system for WORMs. The fifth meeting was held at Murray Hill, NJ on July 17–19, 1990. We adopted a working paper and set to work on a list of issues suggested by the chair.

Data Compression

Despite the huge capacities of WORM disks, people always want more. Data compression is an easy way to supply more, and on current machine architectures, probably can speed data access by trading CPU cycles for I/O bandwidth. Its main problem is that you need to support more than one algorithm and thus, you need some way to specify algorithms. This is a purely administrative issue, but luckily, it appears that X3 may soon act as a registry for compression algorithms (driven by the need to register compression algorithms for IBM 3840 cartridge tape work in X3B5). (How does this fit in with the rumblings about compress from POSIX.2? I’m not certain. I think part of becoming part of the registry means giving up patent rights or allowing liberal licensing, but maybe not. After all, the CD formats are now an ISO standard, but I still think you have to be licensed to make them.)

Path Tables and Extended Attributes

Path tables were removed from the working paper. We agreed to support hard and symbolic links. The next question was how to handle “secret” files: files primarily intended for system use. Examples might include the file describing free space, associated files (like the resource fork of a Macintosh file), and extended attributes (of a Microsoft s-1HP file). We agreed that the latter two cases should be handled by regular files that probably are not in the directory tree but are pointed to by the “inode” for a file. (Note that this implies there is a way to scan all the files in a volume set without traversing the directory tree(s), analogous to running down the inodes in UNIX.)

Given this, we have decided to support extended attributes as a “secret” or system file (and probably include pointers to things like resource forks as those attributes). This also gives us an extensible way of handling non-standard or non-essential inode fields. One of the important tasks remaining is to decide which fields are more-or-less mandatory (such as modify time, owner) and which can safely be pushed off into the extended attributes (access control lists, file valid after date). Please send us your suggestions!

Space Allocation and Management

We agreed that we have to support preallocating space for files, freeing some or all of that space and then reusing that space for other files. After much discussion about extent lists and bit maps, we compromised on a scheme based on extent lists (the details to be worked by the working paper editor). The idea is that is that the free space is described by an extent list (of small but specifiable size) of the “best” (probably largest) free spaces, and if this overflows, “worst” free spaces are added to a system file representing all the free spaces not in the above extent list.

Checksums

It was decided that all system data structures would include a 16 bit checksum (CRC-16). We anticipate that most errors would be transient (cabling or memory) and not be media errors.

Multi-Volume Sets

I had thought the last meeting had settled just about all the questions about multi-volume sets, but I was wrong. It took most of a day to agree on these.
• You have to have the last volume in order to grok the whole volume set (access any/all of the directories and files).
• You can extend volume sets at any time. This and the last item taken together imply the existence of “terminal” volumes (which can act as master volumes of a volume set) and “nonterminal” volumes (the rest). For example, if I extend a single-volume volume set by two volumes, then volumes 1 and 3 are terminal and volume 2 is not.
• You can extract file data from any volume by itself. This is meant only for disaster recovery (I dropped the master volume down the stairwell) and doesn’t imply any requirements on directory tree information (much as fsck restores unattached inodes to /lost + found).
• Volumes can refer to data (say, extents) on other volumes (both earlier and later volumes). Preallocated space on any volume in a volume set can be returned for future reuse.
• The address space of logical blocks for the volume set will be 48 bits; 16 bits for the volume number and 32 bits for the logical block number within a volume. Media can be big (200GB helical scan media exist now) so 32 bits may seem barely big enough, but in such cases you can use a big logical block size. For example, a logical block size of 16KB implies a limit of 64 terabytes per volume; this should be ample for a few years.

Defect Management

We spent a lot of time on this and learned a lot, but basically put it off to the next meeting. What we mean by “defect management” is “How do we deal with write errors from the file system’s point of view?” (We ignore the disk controller and the device driver, both of which do some unknown amount of more-or-less transparent error management.)

We discussed the “sane” approach: insert a layer between the file system that handles errors, allowing the file-system code to assume an error-free interface. This apparently good idea is ruled out by slip-sectoring, a (to my mind bogus) technique, which says, “if writing block \( n \) fails, then try subsequent blocks \( (n+1, n+2, \ldots) \) until we succeed.” Slip-sectoring is mainly used to enhance performance (it does ensure that blocks are more-or-less contiguous), and some disk controllers use it as their error-management technique. (This really screws up your logical address space; it is legitimate for a SCSI disk, your typical error-free, logical-address-space disk interface, to write logical block 5 at physical block 5, then logical block 1 at physical block 4 (1-3 were write errors), then disallow I/O to logical blocks 2, 3, and 4 because there is no place to put them — these blocks just vanish!)

As preparation for the next meeting, Don Crouse, who deals mainly with high-end machines like Crays and large IBMs, is writing a position paper on performance, and members of the committee, many of whom are drive manufacturers or integrators, are collecting estimates of error rates we have to deal with. (This matters; I see one bad block out of 100,000, but some people have used drives with a bad block in every 100.) The problem is that WORMs have really slow seek times, and when you are pouring a 50MB/s Cray channel at a set of WORMs, you can’t afford to spend 1-2 seconds seeking to the bad block area. I personally think we should just do regular bad-block mapping (like most SMD disk drivers) out of a special system file, and people with performance concerns should arrange to have this space spread over the disk.

Endian-ness

A poll was taken of who really cared which way integer fields were stored; the results were LSB – 1, MSB – 1, Don’t Care – 11. It is awkward to specify one of LSB and MSB; this puts half the systems out there at a competitive (performance) disadvantage (though I am skeptical of whether it’s significant). Even though we’re specifying an interchange standard, the group felt that most interchange would be between systems of the same endian-ness, so we should, somehow, allow native byte order. Accordingly, we agreed that endian-ness will be specified in the volume header (for the whole volume set). In retrospect, I think this was silly; we should have just picked one way. In order that everyone important be evenly disadvantaged, we could have used some byte order like 3-0-1-2 that no one uses.
Finale

The committee is trying to nail down a firm proposal for balloting. We anticipate a substantial amount of change at the next meeting (Oct 16-18 in Nashua, NH) and have reserved time (Dec 11-13, but no place) for an additional meeting so that we can ballot after the following meeting (Jan 29–31, Bay area). We now have a working paper (available by the end of September or so); I think it likely we can meet this schedule, but who knows.

Anyone interested in attending any of the above meetings should contact either the chairman, Ed Beshore (edb@hpgrla.hp.com), or me (andrew@research.att.com, research!andrew, tel: 908/582-6262). I am also soliciting your comments on necessary inode fields and defect management. I will present anything you give me at the next meeting.

Report on X3J16: C++

Mike Vilot <mjv@objects.mv.com> reports on the July meeting in Seattle, WA:

Standard C++?

The C++ programming language has been gaining popularity at a remarkable rate (an informal estimate is that the C++ population doubles every nine months). One reaction to the growing popularity has been a call to stabilize the language's definition and achieve some consistency across implementations. C++ is popular enough that larger corporations are considering adopting it as an officially endorsed development language, but some cannot make such a move unless the language is defined by a standard.

For these and other reasons, the ANSI secretariat agreed to establish the X3J16 committee to formulate a standard for C++. Dmitry Lenkov, of HP, made the official proposal and serves as chairman of the committee. To date, X3J16 has met three times: an organizational meeting last December, the first technical meeting in March to get organized, and a meeting in July to really get started.

The December meeting was purely administrative: over 50 attendees received lectures and tons of paper on X3 rules and procedures. The highlight of the day was an invited presentation by Bjarne Stroustrup on "the spirit of C++." The transcript is available as committee document X3J16/90-0022 or from Greg Comeau at Comeau Computing, 91-34 120th Street, Richmond Hill, NY 11418, (718) 849-2355.

March meeting

AT&T hosted the meeting in New Jersey. Most of the week was spent on administrative matters, while the group got organized and accustomed to The Bureaucratic Way. Since most of the members are engineers, the highlight of the week was the evening technical sessions on implementing exception handling for C++. (The week was sort of a mini-USENIX conference, as most members had gone without a substantial C++ gathering since the October '88, Denver conference.)

The week's major activities were discussing and preparing a goals document, and describing the committee's activities and priorities.

Goals

Here is a brief outline of the goals document, which is available as X3J16/90-0023:

- Standardize syntax and semantics of the language as a token sequence without the presence of preprocessing directives.
- Define and standardize a minimum set of C++ libraries, their contents, and interfaces.
- Standardize elements of a C++ environment.
- Consider proposed major changes: parameterized types and exceptions.
- Ensure that the standard is suitable for the international community.
- Ensure a very high level of compatibility with ANSI C.
- Establish coordinating liaisons with X3J11 (ANSI C) and Numerical C Extensions Group.
- Produce two deliverables: draft proposed standard and rationale. Priorities:
  - clear & unambiguous
  - C++ reference manual
  - other base documents
  - consistency
  - user/implementer experience
portability, efficiency, expressiveness
* ease of implementation (including translation to C)

There was some confusion over the multiple base documents. Most members had seen the AT&T C++ version 2.0 reference manual, but in preparation for standardization, the language and its reference manual had suffered a number of subsequent, small changes. AT&T made the 2.1 reference manual available to X3J16; it was essentially the text of the book *The Annotated C++ Reference Manual* by Margaret Ellis and Bjarne Stroustrup.

My naive suggestion to remove the ANSI C standard as a base document in favor of a single base provoked the most intense and emotional discussion of the week. At stake was compatibility between C++ and C.

While most members of X3J16 feel that the existence of a separate committee implies the standardization of a new language, some former members of X3J11, which just finished the C standard, want to eliminate any and all incompatibilities with C. (There was even a threat to sabotage the C++ standard in balloting if they are not removed.)

This issue is obviously important and has two sides. Make your preferences known to the committee. For detailed reference material, both “C++: As Close as Possible to C—But No Closer,” (Bjarne Stroustrup and Andy Koenig, *The C++ Report*, 1(7), 1989) and Chapter 18 of *The Annotated C++ Reference Manual* document and explain differences and incompatibilities between the languages as they stand today.

Focusing on a language without preprocessing directives continues the de-emphasis of the C preprocessor. This is particularly favored by C++ vendors looking into more powerful development environments.

[Editor: Admittedly, improper preprocessor use can sink us in deep and dirty bath water, but let's make sure to save the baby. When writing portable C, I personally find #ifdefs extremely valuable; I suspect they will remain valuable in C++, and I would hate to see the working group neglect this valuable porting tool.]

The libraries effort includes asking what to do about the ANSI-C library, and investigating what can be standardized in a more C++-like approach.

The environment work addresses the linking and executing of C++ code with non-C++ code (i.e., linkage and program execution models), rather than development environment tools.

There are thousands of suggested “improvements” proposed as extensions to C++, but there is consensus on two named in the goals document: parameterized types and exception handling. Their proposals are detailed, and both have been implemented (in some form) in a few C++ implementations.

The emphasis on international concerns reflects the lessons learned from the difficulties of C standardization. X3J16 has some fences to mend, particularly in the international community. Rather than waiting until the last minute to spring a standard on the ISO, the C++ committee is involving itself with the international community right from the start.

**July meeting**

Microsoft, Inc. hosted the second meeting in Seattle. Sub-groups focused on the key topics listed in the goals statement from the March meeting, and reported their progress here.

**International Concerns**

Steve Carter, of Bellcore, presented the major International Concerns (particularly character sets and formal specification) and asked the other groups to work on these issues. He also suggested various sites overseas where future X3J16 meetings could help cooperation with international standardization efforts.

**Editorial**

Jonathan Shopirio, of AT&T, presented the Editorial group’s proposal for organizing and formatting the standard. He is also working on an abstract machine model and a way to define the semantics in the standard precisely and consistently.

**Formal Syntax**

James Roskind, an independent consultant, presented the work of the Formal Syntax group. He has developed (and published on the net) a yacc-able grammar for C++, and is concerned about ambiguities in the the language. Most of the discussion was spent trying to discover whether C++ can (or should) be made LALR(1).
Core Language

Andy Koenig, of AT&T, presented the Core Language group’s work. Initially, they identified and classified difficulties in the working document.

Environment

John Vasta, of HP, presented the work of the Environment group. A key issue addressed by this group is the interaction of C++ with other programming languages. Among the important topics are linkage of C++ and non-C++ translation units, especially the construction and destruction of static C++ objects.

Libraries

I presented the Library group’s work. There were many suggestions, from both inside and outside of the committee. (Interested outside suggesters were James Coggins, Keith Gorlen, and Doug Lea, who have each developed large C++ libraries.) A few people noted similarity with topics covered by other standards (notably POSIX). Initially, the library group will focus on a few commonly-used components. Parameterized types and exception handling will significantly affect the way we design libraries in C++.

Language Extensions

Bjarne Stroustrup, of AT&T, presented the work of the Extensions group, which was by far the most active. The technical sessions presented experience with implementation and use of the template facility.

The most active and emotional debate of the week was on exception handling, which discussed the proposal outlined by Andy Koenig and Bjarne Stroustrup in their paper “Exception Handling for C++” presented at the USENIX C++ Conference in April. Martin O’Riordan, of Microsoft, and Mike Miller, of Glockenspiel, presented arguments in favor of extending the current proposal (which defines termination semantics for exceptions) to include resumption semantics. Andy and Bjarne explained their reasons for not including resumption — the most important being the complexity and cost of implementation.

To their credit, the group worked hard to find a proposal that provided both kinds of exceptions with acceptably small time/space overhead. However, at the end of the week, Bjarne declared the debate deadlocked and refused to impose his proposal while substantial disagreement remained. This is another topic where you should make your opinions heard.

C Compatibility

Mike Miller presented the work of the C Compatibility group. Tom Plum, of Plum-Hall, produced a list of every section of the C++ reference manual that was not C. Much of the group’s near-term activity will be devoted to explaining the many items on the list.

The Seattle meeting produced tangible progress on the language standard. X3J16 voted to accept the proposed document outline and format. They also agreed to incorporate the template proposal (the text from Chapter 14 of The Annotated Reference Manual, minus the annotations — it was literally a scissors-and-tape job). We hope C++ vendors will regard templates as now officially in the language, and provide users an opportunity to work with this feature.

Next events

A few substantial issues lie ahead. The next meeting should see some resolution on the exception proposal. We should see some progress on the review of language ambiguities and inconsistencies, and have some idea of how difficult it will be to ANSify the document. We should also see some specific proposals on library contents. The most substantial will be a simplified version of iostreams by Jerry Schwarz of Stardent.

Our target date for delivering a draft standard is the end of 1992. X3J16 meets three times per year. The next three meetings (and their hosts) will be:

- November 12-26, Cupertino CA (Hewlett Packard)
- March 11-15, Nashua NH (Digital)
- June 17-21, Lund Sweden (Lund Institute of Technology)

Membership on an X3 committee is open to any individual or organization with expertise and material interest in the topic addressed by the committee. The cost for membership is $250. Contact the chair or vice chair for details.
Chair: Dmitry Lenkov  
HP California Language Lab  
19447 Pruneridge Avenue MS 47 LE  
Cupertino, CA 95014  
(408)447-5279  
FAX (408)447-4924  
email dmitryhplabs.hp.com  

Vice Chair: William M. Miller  
Glockenspiel, Ltd  
P.O. Box 366  
Sudbury, MA 01776-0003  
(508)443-5779  
email wmmiller@cup.portal.com  

Report on U.S. TAG to ISO/IEC/JTC1/SC22 WG15

Susanne Smith <sws@calvin.wa.com> reports on the July 19, 1990 meeting in Danvers, MA:  

Overview

Before you ask, ISO/IEC/JTC1/SC22 WG15 is ISO POSIX. The U.S. TAG is the United States Technical Advisory Group, which formulates the U.S. position on WG15 issues, and chooses the members of the U.S. delegation to the international WG15 meetings.

This meeting began at 8:00 A.M. and ended before noon. This must be a record — not just for the TAG, but for any standards group meeting. There were three major business items:

- language independence,
- document circulation procedures, and
- rapporteurs.

ISO POSIX: Winners and Losers

The vote for 9945-1.2 (1003.1a draft 5) was unanimously in favor without substantive comments. If all goes well there just may be an IEEE version of 9945-1 available in Seattle.

My last report mentioned the formatting problems with the 9945-1 document. The TAG had decided to request the formation of an ad hoc committee in Paris to try to resolve these problems. WG15 resolved to instruct the WG15 convener, Jim Isaak, to request written editorial requirements from the ITTF (formerly the Central Secretariat) and IEEE, and forward these to SC22. The emphasis here should be on written requirements.

WG15 refused to register 1003.4, real-time extensions, as a CD (committee document, formerly DP, draft proposal) because it is not a language-independent specification. They were also concerned that the standard might have to change once there is a language independent version of 1003.1.

1003.5, Ada binding, and 1003.9, FORTRAN binding, suffered a similar fate for different reasons. 1003.5 and 1003.9 were held off until at least the October WG15 meeting because G15 had not seen the 1003.5 and 1003.9 documents, and were reluctant to register something they hadn't seen before. And again, they were concerned that these standards might have to be rewritten once there is a language independent version of 1003.1.

Administrivia

Skip to the next section if you're easily bored or just not interested in bureaucracy. Why, you ask, was WG15 being asked to register something they had not seen before? Here are the steps that have to be completed before a document gets circulated:

- The committee and SEC approve its release.
- The TAG approves its circulation.
- The committee chair delivers the document to the TAG chair, Donn Terry.
- The TAG chair forwards the document to the WG15 convener, Jim Isaak.
- The WG15 convener distributes the document.

1003.5 and 1003.9 were approved by the TAG for circulation to WG15 during the April meeting in Snowbird. This left six weeks for for the documents to be circulated and read. The problem was that the TAG chair did not receive the documents in time to have them circulated before the meeting. To avoid this problem in the future, the TAG is going to ask the SEC to assign an action item to the committee chair so that there is a method to track this task.

In other news:

- The TAG procedures were entered and marked up, and will be included in the next mailing.
Are You Ready for UNIX in VDM?

We cannot delay language independence for 1003.1 any longer. It’s now really holding up international progress on important POSIX efforts. But what format or technique should we use? ISO rules seem to require an ISO-standard method, which could restrict us to VDM (Vienna Definition Method), but no one thinks VDM will work. Paul Rabin and Steve Walli have been working on a method, but the TAG worries that a non-standard method will create problems like those we’ve suffered through with document formats (see last TAG report). In order to avoid rejection later we will circulate the new method in SC22 and WG15 for review and comment. To make this circulation useful, Donn Terry is listing specific questions for SC22 and WG15 to answer.

[Editor: I believe that ISO rules only restrict us to VDM if we produce a formal definition, i.e., something from which one could do correctness proofs. Of course, rules and politics are not always the same thing and using VDM might help grease the skids. Still, we should stop and ask if not using VDM would hold us up any more than using VDM.]

The TAG will also ask the WG15 convener to schedule an ad hoc meeting on language independence during the October WG15 meeting to help move it along.

Rap, a-rap, a-rap, they call me the rapporteur.

Rapporteurs are technical experts on specialized aspects of a particular standards effort. Their scope is usually broader than an individual standard, and they usually coordinate efforts in several standards bodies. WG15 has three rapporteur groups, one each for conformance, internationalization, and security. We send a representative to each.

The conformance-testing rapporteur group will be looking at 1003.3 draft 12 (conformance testing), and the OSF-UI-X/Open Phoenix project as potential base documents for the ISO 9945-series documents. The Phoenix project is developing a conformance-testing platform. We will not have to decide whether we want to submit 1003.3 as a new work item in this area until 1991.

Ralph Barker asked that UniForum be allowed to send him and one UniForum Internationalization Technical Committee member to the next internationalization rapporteur group meet-

ing. This person would be subject to subcommittee approval but selected by UniForum. Worry about the fact that the TAG would not choose this person evaporated when it became clear that Donn Terry would continue as internationalization rapporteur and that the UniForum members would just be an addition.

The TAG appointed AI Weaver security rapporteur to fill the vacancy Terry Dowling left when he resigned in January.

Summary

The most important development is that the synchronization proposal discussed in the last report is already dead. This proposal was to have fed balloting responses from IEEE into WG15, and vice-versa, allowing WG15 approval to follow on the heels of IEEE approval. Now, while the IEEE is advancing, everything in WG15 is blocked by 1003.1 language independence.

Report on NIST Shell-and-Tools FIPS Workshop

Donald Lewine
<lewine@cheshirecat.woob.dg.com> reports on the September 6, 1990 meeting in Gaithersburg, MD:

The Federal Government publishes Federal Information Processing Standards (FIPS) for use in buying and using computers. One set of FIPS deal with systems with “POSIX-like interfaces.” The government will purchase about $7 Billion worth of POSIX systems in FY91. Standards let the government avoid vendor-specific requirements like UNIX or SVID. The theory is that the larger the number of vendors that can meet the specification the lower the cost to the taxpayer. Whether that’s true or not, using standards makes it harder to protest a purchase decision.

On September 6, the National Institute of Standards and Technology (NIST) held a workshop to gather input from industry and federal agencies on the wisdom of adopting Draft 9 of the IEEE Standard for POSIX Shell and Utility Application Interface (P1003.2) as a Federal Information Processing Standard (FIPS).

The meeting was attended by about a dozen system vendors and about half that many federal agencies.
Roger Martin of NIST opened the meeting with what was to be a three-minute introduction. NIST’s agenda was to collect specific comments on the FIPS as printed on Page 23959 of the Federal Register. The vendors’ agenda was to get NIST to give up the idea of adopting a FIPS until after the IEEE standard is final. Not surprisingly, given this clash, Roger’s opening remarks ran over by a factor of 20.

Here is NIST’s case for adopting a FIPS based on POSIX.2/D9:

- The federal government is going to purchase about $17 billion worth of systems with “POSIX-like interfaces.” NIST wants to give the agencies as much help as possible. Draft 9 is a good enough standard to serve this purpose.
- It takes about a year to get a FIPS adopted. If POSIX.2 is not approved until mid-1991, a FIPS based on draft 9 will have a significant lifespan.
- If NIST were to publish a FIPS, it would accelerate the production of the P1003.2 standard. (just as FIPS 151 accelerated IEEE 1003.1-1988).
- No agency is going to be stupid enough to demand draft 9 if a vendor can supply a system conforming to a later draft or to the final standard, so the FIPS will do no harm. (This was hotly debated.)

After that introduction, and before the next attack on Roger Martin, Sheila Frankel and Rick Kuhn described the technical content of the FIPS. Basically, the idea is to adopt draft 9 minus the parts that might change. There are about 25 items that may change.

Roger Martin came back for another round of target practice. He went over the general policy of NIST, which is to adopt standards from outside and at the highest possible level. The levels are, highest to lowest:

- International Standards
- National Standards
- Draft Standards
- de facto Standards

NIST could be convinced to change from POSIX.2/D9 to POSIX.2/D10. Here are the factors it will consider:

How much delay is introduced (Three months may be OK. One year is unacceptable.)

Is Draft 10 that much better than Draft 9? Is this just a delaying action?

Shane McCarron, of UNIX International, made a great speech pointing out how much wasted effort would occur if every vendor had to rush out and implement POSIX.2/D9. The NIST people seemed shocked at how different POSIX.2/D9 is from existing practice.

[Editor: See Randall Howard’s POSIX.2 report for some examples of just how different Draft 9 is from Drafts 8 and 10.] Nevertheless, the argument seemed to fall on deaf ears, because NIST claimed that a promise to meet the FIPS should be good enough, and everyone can still wait for AT&T USL to write the code.

It was pointed out that Congress did not allocate enough funding for NIST to do much testing for POSIX.2 conformance. This means that vendors will have to “self certify” and coverage may vary. After some discussion this item was placed into the “write your representative” category, because only Congress can allocate the money.

NIST pointed out that they are under a great deal of pressure to “advise” federal agencies who want to move to open systems. A large percentage of RFPs for POSIX-like systems will be coming from groups who know nothing about such systems. Vendors were worried that this “advice” would end up in court cases and be read by judges as “regulations.”

In my opinion, NIST is going to go ahead and publish a flawed FIPS in the belief that it will drive the IEEE to pick up the pace of POSIX. The government has a burning need for a standard, they find it politically unacceptable to use UNIX System V as that standard, and they strongly prefer action over waiting for the IEEE.
Recent Standards Activities
Jeffrey S. Haemer <jsh@ico.isc.com>.

Summer-Quarter Standards Activities

This editorial addresses some of the summer-quarter standards activities covered by the US-ENIX Standards Watchdog Committee. In it, I've emphasized non-technical issues, which are unlikely to appear in official minutes and mailings of the standards committees. Previously published watchdog reports give more detailed, technical summaries of these and other standards activities. If my comments encourage you to read one of those earlier reports that you wouldn't have read otherwise, I've done what I set out to do. Of course, on reading that report you may discover the watchdog's opinions differ completely from the ones you see here. As watchdog editor I just edit the reports, I do not determine their contents. The opinions that follow, in contrast, are mine.

Profiles

There's an explosion of activity in the profiles world, bringing with it an explosion of problems, and dot zero, the POSIX guide group, is at ground zero. The first problem is, "What's a profile?" Everyone has a rough idea: it's a document that specifies an application-specific set of standards (or pieces of standards). The best informal illustration I've heard is from Michelle Aden, of Sun Microsystems. Imagine, she says, you have to write a guideline for buying lamps for Acme Motors. You might require that the lamps have ANSI-standard, three-prong plugs, accept standard one-way, hundred-watt bulbs, have cords no shorter than five feet, and stand either two to three feet tall (desk models) or five to seven feet tall (floor-standing models). This combination of pointers to standards, additional specifications, and detailed options, which gives purchasing agents guidelines to help them make choices without tying their hands to a specific vendor, is a profile — in this case, an Acme Motors lamp profile. Dot zero now sees itself as a group writing a guide to help profile writers pick their way through the Open-Systems' standards maze.

But that rough agreement is as far as things go. And the standards world is never informal. For "profile" to graduate from a hallway conversation buzzword to an important organizing principle, it needs a precise definition. And since there are already four groups writing profiles — real-time, transaction processing, multiprocessing, and supercomputing — TCOS needs to figure out what a profile is quickly. ISO already has IAPs (International Applications Profiles). The ISO document TR 10K describes these in detail. Unfortunately, TR 10K was developed for OSI-related profiles and shows it. Cut-down extracts of the standard appear in the document. Someone needs to define a PAP (POSIX Application Profile).

But that's just the first problem. Even thornier is "What does it mean to say that something conforms to the POSIX transaction-processing profile?" If I want to write assertions for a profile or tests to verify those assertions, how do I do it? Does it suffice to conform to the individual components? What about their interactions? The first principle of management is "If it ain't somebody's job, it won't get done." Dot zero has done such a good job of promoting The Profile as an organizing principle for addressing standards issues that people are beginning to press dot zero for answers to questions like these. Unfortunately, that's a little like killing the messenger. It's just not dot zero's job. So the fundamental profile question is "Who's in charge?" Right now, I think the question sits squarely, if uncomfortably, in the lap of the SEC (the Sponsors Executive Committee), which oversees the IEEE's operating-systems activities.

In the meantime, the various working groups writing profiles are making headway by just trying to define profiles and seeing where they get stuck. Dot twelve, the real-time profile group, is busily making various sorts of tables, to try to find a reasonable way to specify the pieces that make up a profile, their options, and their interactions. Dot ten, the supercomputing profile group, is seeking an overall structure for a profile document that

1. The introduction to this series of reports provides a general overview of the committee itself.
2. I use "dot zero" to refer both to the P1003.0 working group and to the document it's producing. These are common conversational conventions among standards goers, and which of the two I mean is usually obvious from context.
makes sense. Dot eleven, the transaction-processing profile group, is trying to steal from dots twelve and ten, an important test of the generality of the other two groups' solutions. Dot fourteen, the multiprocessing profile group, isn't far enough along to make theft worth their while, but will eventually provide a second generality test. Think of it as a problem in portable ideas.

**Will I Win My Beer?**

In my last editorial, I announced a beer bet with John Gertwagen over whether threads will ballot and pass before the base dot-four (real-time) ballot objections are resolved. I'm still betting on threads, but it looks like the bet is still anyone's to win. Some folks assure me that I'll win my beer handily, others say I don't have a chance.

At the summer POSIX meetings in Danvers, Massachusetts the dot-four chair, Bill Corwin, challenged the threads folks to come up with a ballotable draft by the end of the week, and they very nearly did. (I hear complaints from some quarters that the vote to go to ballot was 31 to 7 in favor, and that attempts to move toballoting were only blocked because of filibusters from those opposed.) On the other hand, technical reviewers are now resolving ballot objections to the base with machetes. They've thrown away asynchronous events altogether and have discarded real-time files and adopted the fmmmap model that the balloting group suggested.

**Innovation**

Hoare once said, "One thing [the language designer] should not do is to include untried ideas of his own." We have followed that precept closely. The control flow statements of Ratfor are shamelessly stolen from the language C, developed for the UNIX operating system by D. M. Ritchie. — Kernighan and Plauger.

Should standards groups just standardize existing practice or should they be solving known problems? And if they solve known problems, how much innovation is allowed? Shane McCarron's September UNIX/Review article uses the real-time group, dot four, as a focus for an essay on this subject. His thesis is that standards bodies should only be allowed to standardize what's boring. I've already seen John Gertwagen's reply, which I assume will be printed in the next issue. I find myself agreeing (and disagreeing) with both and recommend you read them.

This battle will rage brighter in some of the groups less far along, but sporadic fighting still breaks out in the shell and tools group, dot two. Right now, collation and character classification are seeing a lot of skirmishing. Some want to stay relatively close to the existing practice, while others want to grow a mechanism to deal with the Pandora's box of internationalization. My favorite current example, though, is make. Bradford's augmented make is almost a decade old. Stu Feldman's original is a couple of years older than that. That decade has seen a number of good make replacements, some of them wildly successful: Glenn Fowler's nmake has virtually replaced make for large projects in parts of AT&T. Still, many of these upgrades maintain the original make model, just patching up some of make's more annoying craters and painting over its blemishes. At this point, there is real consensus among make augmentors about some patches. Most upgrades expand make's metarules. For example,

```
.c.o:
$(CC) $(CFLAGS) <
```

might become

```
%.c : %.o
$(CC) $(CFLAGS) <
```

Not much of a change, but it also gives us

```
.s : .
$(GET) $(GFLAGS) -p $< >$
```

in place of the current, baroque

```
.c.o:
$(GET) $(GFLAGS) -p $< >$
```

3. Dot four's real-time files are currently a part of the supercomputing profile. If they disappear from dot four, they may reappear elsewhere. Dot four has moved from "design by working committee" to "design by balloting committee."


Make's successors don't agree on syntax, but they all agree that "..." rules are the wrong solution to a real problem. Should dot two standardize a newer solution? Existing-practice dogmatists would say, "No. It's not make." Here's a place I say, "Yes, if we can do it in a way that doesn't break too many makefiles." The prohibition should be against untried ideas, and I don't see those here. A year or so ago, Stu Feldman (make), Glenn Fowler (nmake), Andrew Hume (mk), and a handful of other make luminaries presented a proposal to add four extensions to dot two's make. Not one is yet in the draft. I hope that changes.

SCCT Faces Serious Problem

At Danvers, the testing group, dot three, worked with dot two on test assertions to try to avoid the kinds of problems created by the P1003.1 test assertions, which dot one had no input into until the assertions were in ballot.

A side effect of the collaboration, which is taking place before dot two is finished, is that it may reveal that parts of dot two are imprecise enough to require a rewrite. Dot two, draft eight had around four-hundred ballot objections, draft nine saw fewer than half that number. There was hope that draft ten would halve that again, bringing it within striking distance of being a standard. The assertion work may point out and clear up rough spots that might otherwise have escaped the notice of battle-fatigued balloters. (Paradoxically, NIST, which is heavily represented in dot three and painfully familiar with dot two's status and problems, is currently pushing for a shell-and-tools FIPS based on the now-out-of-date draft nine.)

The exercise of trying to construct assertions for dot two before it goes to ballot may bring some new testing problems into focus, too. Before I explain what I mean, I'll give you a little background.

The POSIX effort has outgrown dot three, which did test assertions for dot one and is in the process of constructing test assertions for dot two. Dot three has, at most, a couple of dozen mem-

bers, and the document for dot two alone may swell to one- or two-thousand pages. If dot three were to continue to do all test assertion work, it would have to produce a similar document for at least a dozen other standards.

Reacting to this problem, the SEC created a steering committee, the SCCT, to oversee conformance testing. The committee's current plan is to help guide standards committees to write their own assertions, which will be part of the base document. Test assertions, like language independence, are about to become a standards requirement (a standards standard).

With this change, the current process — write a base document, evolve the base document until it's done, write test assertions for the result, evolve the test assertions until they're done — would become: write a base document with test assertions, then evolve the base document modifying the test assertions as you go. A sensible-enough idea on the surface, but after the joint dot-two, dot-three meeting I have questions about how deep that sense runs.

First, does it really make sense to write assertions early? Working-group members should be exposed to assertion writing early. When working-group members understand what a testable assertion is, it's easier to produce a testable document. Still, substantive, major draft revisions are normal, (see the real-time group's recent ballot, for example) and keeping test assertions up-to-date can be as much work as writing them from scratch. This meeting saw a lot of review of draft-nine-based assertions to see which ones had to change for draft ten.

Second, if you make the assertions part of the standard, they're voted on in the same ballot. Are the same people who are qualified to vote on the technical contents qualified to vote on the test assertions?

Third, writing good assertions is hard, and learning to write them takes time. How eager will people in working groups be to give up time they

7. It didn't reach that goal. Keith Bostic tells me he submitted 132 objections himself.

8. Any imagined glamour of POSIX meetings fades rapidly when one is picking nits in a several-hundred-page standards document. When asked where the next meeting was, one attendee replied, "some hotel with a bunch of meeting rooms with oversized chandeliers and little glasses full of hard candies on the tables."
now spend writing and revising document content in order to do assertions?

Fourth, is the time well-spent? Not everything merits the time and expense of a standard. If only a small number of organizations will ever develop test suites for a particular standard (with none being a special, but important case) does it make sense for folks to spend time developing standards for those test suites? Wouldn’t it make more sense to develop it after there is a clear need? (This is a perverse variant of the “existing practice” doctrine. Even if you don’t think standards should confine themselves to existing practice, does it make sense to innovate if there’s never going to be an existing practice?)

**Stay Tuned for This Important Message**

If you haven’t yet had the pleasure of internationalizing applications, chances are you will soon. When you do, you’ll face messaging: modifying the application to extract all text strings from external data files. The sun is setting on

```c
main ()
{
    printf("hello, world\n");
}
```

and we’re entering a long night of debugging programs like this:

```c
#include <stdio.h>
#include <nl_types.h>
#include "msg.h" /* decls of catname(), etc. */
#define GRTHG "hello, world:n"
nl_catd catd;
main ()
{
    setlocale(LC_ALL, "");
    catd = catopen(catname(argv[0]), 0);
    printf(catgets(catd, SETID, HSGID, GRTHG));
    catclose(catd);
    exit(0);
}
```

This, um, advance stems from a desire to let the program print

```
chào các ông
```

instead of

```
hello, world
```

when `LANG` is set to “Vietnamese.”

Most programs use text strings, so the system services interface group, dot one, has been thinking about portable library calls to supply such strings and portable formats for the files that contain them.

Actually, “re-thinking” is probably more accurate than “thinking about.” 1003.1a Draft 9, specified a design by the UniForum Technical Committee on Internationalization. At Danvers, X/Open counter-proposed a variant of its existing XPG3 specification, arguing that the X/Open scheme may have problems but it also has users, while the UniForum proposal is still in the laboratory. (It brings to mind the apocryphal story of Stu Feldman’s wanting to improve the design of `make`, but feeling he couldn’t because he already had seven users.) Someone from Unisys also brought a proposal, different from both UniForum’s and X/Open’s.

That no one even showed up to defend the UniForum proposal shows that there is something wrong with standardizing messaging. In one instance there is enough support for a messaging scheme to get it into the draft standard; in the next, there’s none at all. In the end, the working group agreed that a messaging standard was premature and that the free market should continue to operate in the area for a while.

Given the relative sizes of the organizations concerned, this outcome probably sticks us with the X/Open scheme for a while, which I find the ugliest of the lot. Still, it’s not a standard, and there’s room for innovation and creativity if we’re quick about it. The “existing practice” criterion is supposed to help avoid a requirement for massive, murderous source code changes. We should be looking for the messaging scheme that doesn’t require changes in the first place, not the one with the most existing victims.

**Language Independence Stalls ISO Progress**

Internationally, 1003.4 (real-time), 1003.5 (Ada bindings), and 1003.9 (FORTRAN bindings) are being held hostage by ISO, which refuses to loosen them on the world until we come up with a language-independent binding for 1003.1. The question is, who will do the work? “Not I,” says dot four, whose travel vouchers are signed by companies caught up in the glamour of real-time and threads; “Not I,” say dot five and dot nine, who seldom have even ten working-group members apiece; “Not I,” say the tattered remnants of dot one, exhausted from struggling with 1003.1-1988, FIPS-151 and 151-1, and (almost) 1003.1-1990, before any other groups have even a first standard passed. Where is the Little Red Hen when we need her?
Should We Ballot POSIX the Way We Ballot Three-Phase Power?

In the meantime, we progress inexorably toward balloting on several IEEE/ANSI standards. The sizes of the drafts (and several contributors to comp.std.unix) raise real questions about whether the IEEE's balloting process make sense for the sort of standards work POSIX is performing. A month or so might be enough to review a few-page hardware standard. But is it enough for the nearly 800 pages in the latest recirculation of dot two? Does it really make sense to review the standard for grep in hard copy? Many would like to see longer balloting times and on-line access to drafts. Some argue that the final standard should be available only from the IEEE, both to insure authenticity and to provide the IEEE with income from its standards efforts; even that argument seems weak. Checksums can guarantee authenticity, and AT&T's Toolchest proves that electronic distribution works: I'll bet ksh has paid for itself several times over.

"We handed 1201.1 its head and asked it to comb its hair."

Moving away from POSIX, we come upon 1201.1, still in search of an officially sanctioned mission that the group wants to take on. The group currently has a PAR (charter) to standardize various aspects of X-based windowing, principally the toolkit-level API but any hope of compromise between the OPEN LOOK and OSF/Motif factions died at the winter-quarter Utah meetings. In a moment of responsible behavior, the group recovered by switching to a dark horse: a window-system-independent API that could be implemented on top of either product. Marc Rochkind's XVT, which already allows users to write programs that are portable across several, unrelated window systems including X, the Mac, and MS-Windows, was offered as a proof-of-concept.

While the original charter could probably encompass the new XVT work, the group seemed to feel that this direction change, together with the fragmenting of the original group into separate toolkit, drivability, UIMS, and X intrinsics efforts, required that they ask the SEC for a new charter. (The drivability group has already had a separate PAR approved and is now 1201.2.) The group convened a pair of interim meetings in Milpitas, California, and Boulder, Colorado, to forge a PAR that would meet the SEC's new, stricter standards for PAR approval by the summer Danvers meeting. They didn't succeed.

Most of the problems seem to have been administrative missteps. Some examples:

- Working-group members complained that the Milpitas meeting took place without enough notice for everyone to attend, and issues that had been resolved at the interim meetings were re-opened in Danvers.
- The PAR was so broadly written that at least one technology (Serpent) was advanced as a candidate that almost no one thought should even be considered.
- Some working-group members hadn't even received copies of the XVT reference manual before they reached Danvers.
- Many SEC members appeared not to have seen a copy of the PAR until the afternoon before the SEC meeting, and some saw the final PAR for the first time at the SEC meeting itself.

Many people who weren't familiar with the proposal ended up uneasy about it, not because they'd read it and didn't like it, (they'd not been given much chance to read it) but because a lack of attention to administrative details in the proposal's presentation sapped their confidence in the group's ability to produce a sound standard. After all, standards is detail work. In the end, the SEC tactfully thanked the group and asked them to try again. One SEC member said, "We handed 1201.1 its head and asked it to comb its hair."

I believe all of this is just inexperience, not a symptom of fundamental flaws in the group or its approach. If 1201.1 can enlist a few standards lawyers — POSIX has no shortage of people who know how to dot all the i's and cross all the t's — and can muster the patience to try to move its PAR through methodically and carefully, I think the group will give us a standard that will advance our industry. If it doesn't do so soon, though, the SEC will stop giving it its head back.
Local User Groups

The Association will support local user groups by doing a mailing to assist the formation of a new group and publishing information on local groups in `login:`. At least one member of the group must be a current member of the Association. Send additions and corrections to login@usenix.org.

CA - Fresno: the Central California UNIX Users Group consists of a uucp-based electronic UNIX Users mailing list to which members may post questions or information. For connection information:

Educational and governmental institutions:
Brent Auernheimer (209) 294-4373
brent@CSUFresno.edu or csufresbrent

Commercial institutions or individuals:
Gordon Crumal (209) 875-8755
csufresgordon (209) 298-8393

CA - Irvine: the UNIX Users Association of Southern California meets the 2nd Monday of each month.
Rich Bergstedt, AT&T (714) 727-5231
8001 Irvine Center Dr., Suite 224
Irvine, CA 92718-2900

UUASC Information Line (714) 727-5232

CO - Boulder: the Front Range UNIX Users Group meets monthly at different sites.
Steve Gaede gaede@sa.com
Software Design (303) 444-9100
& Analysis, Inc.
1113 Spruce St., Ste. 500
Boulder, CO 80302

FL - Coral Springs:
S. Shaw McQuinn (305) 344-8686
8557 W. Sample Road
Coral Springs, FL 33065

FL - Fort Lauderdale/Miami: The South Florida UNIX Users Group meets the 2nd Tuesday of each month.
Tony Vincent, John McLaughlin (305) 776-7770
{sun, novavax, gould}@sunvice@tony
jmclaughlin@sun.com

John O'Brien (305) 475-7633
gatechluflorida@novavax@john

FL - Jacksonville/Northeast: UNIX Users of Jacksonville meets the 2nd Thursday of each month.
Tom Blakely (904) 646-2820
ufloridalutfb

Emilie Olsen (904) 390-3621

FL - Melbourne: the Space Coast UNIX Users Group meets at 8pm on the 3rd Wednesday of each month at the Florida Institute of Technology.

Bill Davis bill@ccd.harris.com
(407) 242-4449

FL - Orlando: the Central Florida UNIX Users Group meets the 3rd Thursday of each month.
Mike Geldner code@sunflamiike
Ben Goldfarb goldfarb@bcx9.ucf.edu
Mikel Manitius (407) 869-2462

FL - Tampa Bay: the Tampa UNIX Users Group meets the 1st Thursday of each month in Largo.
Bill Hargen (813) 530-8655
uunet!pdn!hargen
George W. Leach (813) 530-2376
uunet!pdn!reggie

GA - Atlanta: meets on the 1st Monday of each month in White Hall, Emory University.
Atlanta UNIX Users Group
P.O. Box 12241
Atlanta, GA 30355-2241
Marc Merlin (404) 442-4772
Mark Landry (404) 365-8108

MI - Detroit/Ann Arbor: The South Eastern Michigan Sun Local Users Group meets jointly with the Nameless UNIX Group on the 2nd Thursday of each month in Ann Arbor.
Steve Simmons home: (313) 426-8981
scs@korr Dexter mi.us office: (313) 769-4086
K. Richard McGill Bill Bulley
rich@sendai.ann-arbor.mi.us web@applga.uucp

MI - Detroit/Ann Arbor: dinner meetings the 1st Wednesday of each month.
Linda Mason (313) 855-4220
michigan!usr/group
P.O. Box 189602
Farmington Hills, MI 48018-9602

MN - Minneapolis/St. Paul: meets the 1st Wednesday of each month.
UNIX Users of Minnesota Robert A. Monio
17130 Jordan Court pessutt@dmsiaq.mn.org
Lakeville, MN 55044 (612) 220-2427
MO - St. Louis:
St. Louis UNIX Users Group    Eric Kiebler
Plus Five Computer Services    plus5@sluug
765 Westwood, 10A    (314) 725-9492
Clayton, MO 63105

NE - Omaha: meets the 2nd Thursday of each month.
/usr/group nebraska    Kent Landfield
P.O. Box 44112    kent@ugn.uucp
Omaha, NE 68144    (402) 291-8300

New England - Northern: meets monthly at different sites.
Peter Schmitt
Peter.Schmitt@dartvax!dartmouth.edu
Kiewit Computation Center    (603) 646-2085
Dartmouth College
Hanover, NH 03755

Peter J. Holsberg    mccc!pjh
Mercer County
Community College
1200 Old Trenton Road
Trenton, NJ 08690

NY - New York City: Unigroup of New York City meets every other month in Manhattan.
Unigroup of New York City
G.P.O. Box 1931
New York, NY 10116
Peter Gutmann    (212) 618-0973
peterg@murphy.com

OH - Columbus: The Columbus Local UNIX Group meets the 1st Monday of each month.
Mark Verber    verber@mps.ohio-state.edu
Physics Department    (614) 292-8002
Ohio State University
Columbus, OH 43210

OK - Tulsa: the Tulsa UNIX Users Group, $USR, meets the 2nd Wednesday of each month.
Stan Mason    (918) 560-5329
tulsix!smason@drd.com
Mark Lawrence    (918) 743-3013
mark@drd.com

PA - Philadelphia: the UNIX SIG of the Philadelphia Area Computer Society meets the morning of the 3rd Saturday of each month.
G. Baun, UNIX SIG    rutgers!{bpa,cbmvax}!
c/o PACS    temvax!pacabb!{gbaun,whutchi}
Box 312
La Salle University
Philadelphia, PA 19141

TX - Austin: CACTUS meets the 3rd Thursday of each month.
Capital Area Central Texas UNIX Society
P.O. Box 9786
Austin, TX 78766-9786
officers@peyote.cactus.org
James Johnson    (512) 331-3781
jhnsmn@cs.utexas.edu

TX - Dallas/Fort Worth:
Dallas/Fort Worth UNIX Users Group
Sdeny Systems, Inc.
5327 N. Central, #320
Dallas, TX 75205
Jim Hummel    (214) 522-2324

TX - Houston: the Houston UNIX Users Group (Hounix) meets the 3rd Tuesday of each month.
Hounix answering machine    (713) 684-6590
Bob Marcum, president    (713) 270-8124
Chuck Bentley, vice-president    (713) 789-8928
chuckb@hounix.uucp

TX - San Antonio: the San Antonio UNIX Users meets the 3rd Thursday of each month.
Jeff Mason    gatech!petrol!hpsatb!jeff
Hewlett Packard    (512) 494-9336
14100 San Pedro
San Antonio, TX 78232

WA - Seattle: meets monthly.
Bill Campbell    (206) 232-4164
Seattle UNIX Group Membership Information
P.O. Box 820
Mercer Island, WA 98040-0820
uw-beaver!tikall!camcol!bill

Washington, D.C.: meets the 1st Tuesday of each month.
Washington Area UNIX Users Group
2070 Chain Bridge Road, Suite 350
Vienna, VA 22182
Alan Fedder    (703) 448-1908

CANADA - Toronto:
Evan Leibovich    (416) 452-0504
143 Baronwood Court
evan@telly.on.ca
Brampton, Ont. Canada L6V 3H8

November/December 1990
Winter Conference Tutorials and Program
Standards Reports
Book Reviews