Draft Standard for Information Technology— Portable Operating System Interface (POSIX®)

Prepared by the Austin Group

(http://www.opengroup.org/austin/)

Technical Standard

Shell and Utilities, Issue 6

The Open Group
The Institute of Electrical and Electronics Engineers, Inc.

Portions of IEEE Std 1003.1-200x are derived with permission from copyrighted material owned by Hewlett-Packard Company, International Business Machines Corporation, Novell Inc., The Open Software Foundation, and Sun Microsystems, Inc.

Technical Standard

Shell and Utilities, Issue 6

Document Number:

Published in the U.K. by The Open Group, 2001.

See http://www.opengroup.org/austin/bugreport.html for instructions on commenting on this unapproved draft document.

Contents

List of Figures

List of Tables

Preface

IEEE Std 1003.1-200x has been jointly developed by the IEEE and The Open Group. It is both an IEEE standard and an Open Group Technical Standard.

Background

The developers of IEEE Std 1003.1-200x represent a cross-section of hardware manufacturers, vendors of operating systems and other software development tools, software designers, consultants, academics, authors, applications programmers, and others.

Conceptually, IEEE Std 1003.1-200x describes a set of fundamental services needed for the efficient construction of application programs. Access to these services has been provided by defining an interface, using the C programming language, a command interpreter, and common utility programs that establish standard semantics and syntax. Since this interface enables application writers to write portable applications—it was developed with that goal in mind—it has been designated POSIX,[1] an acronym for Portable Operating System Interface.

Although originated to refer to the original IEEE Std 1003.1-1988, the name POSIX more correctly refers to a *family* of related standards: IEEE Std 1003.*n* and the parts of ISO / IEC 9945. In earlier editions of the IEEE standard, the term POSIX was used as a synonym for IEEE Std 1003.1-1988. A preferred term, POSIX.1, emerged. This maintained the advantages of readability of the symbol ‘‘POSIX’’ without being ambiguous with the POSIX family of standards.

Audience

The intended audience for IEEE Std 1003.1-200x is all persons concerned with an industry-wide standard operating system based on the UNIX system. This includes at least four groups of people:

1. Persons buying hardware and software systems
2. Persons managing companies that are deciding on future corporate computing directions
3. Persons implementing operating systems, and especially
4. Persons developing applications where portability is an objective

Purpose

Several principles guided the development of IEEE Std 1003.1-200x:

- Application-Oriented

  The basic goal was to promote portability of application programs across UNIX system environments by developing a clear, consistent, and unambiguous standard for the interface specification of a portable operating system based on the UNIX system documentation. IEEE Std 1003.1-200x codifies the common, existing definition of the UNIX system.

[1] The name POSIX was suggested by Richard Stallman. It is expected to be pronounced *pahz-icks*, as in *positive*, not *poh-six*, or other variations. The pronunciation has been published in an attempt to promulgate a standardized way of referring to a standard operating system interface.

- Interface, Not Implementation

  IEEE Std 1003.1-200x defines an interface, not an implementation. No distinction is made between library functions and system calls; both are referred to as functions. No details of the implementation of any function are given (although historical practice is sometimes indicated in the RATIONALE section). Symbolic names are given for constants (such as signals and error numbers) rather than numbers.

- Source, Not Object, Portability

  IEEE Std 1003.1-200x has been written so that a program written and translated for execution on one conforming implementation may also be translated for execution on another conforming implementation. IEEE Std 1003.1-200x does not guarantee that executable (object or binary) code will execute under a different conforming implementation than that for which it was translated, even if the underlying hardware is identical.

- The C Language

  The system interfaces and header definitions are written in terms of the standard C language as specified in the ISO C standard.

- No Superuser, No System Administration

  There was no intention to specify all aspects of an operating system. System administration facilities and functions are excluded from IEEE Std 1003.1-200x, and functions usable only by the superuser have not been included. Still, an implementation of the standard interface may also implement features not in IEEE Std 1003.1-200x. IEEE Std 1003.1-200x is also not concerned with hardware constraints or system maintenance.

- Minimal Interface, Minimally Defined

  In keeping with the historical design principles of the UNIX system, the mandatory core facilities of IEEE Std 1003.1-200x have been kept as minimal as possible. Additional capabilities have been added as optional extensions.

- Broadly Implementable

  The developers of IEEE Std 1003.1-200x endeavored to make all specified functions implementable across a wide range of existing and potential systems, including:

  1. All of the current major systems that are ultimately derived from the original UNIX system code (Version 7 or later)
  2. Compatible systems that are not derived from the original UNIX system code
  3. Emulations hosted on entirely different operating systems
  4. Networked systems
  5. Distributed systems
  6. Systems running on a broad range of hardware

  No direct references to this goal appear in IEEE Std 1003.1-200x, but some results of it are mentioned in the Rationale (Informative) volume of IEEE Std 1003.1-200x.

- Minimal Changes to Historical Implementations

  When the original version of IEEE Std 1003.1 was published, there were no known historical implementations that did not have to change. However, there was a broad consensus on a set of functions, types, definitions, and concepts that formed an interface that was common to most historical implementations.

The adoption of the 1988 and 1990 IEEE interface standards, the 1992 common standards, the various Open Group (formerly X/Open) versions, and the subsequent revisions and addenda to all of them have consolidated this consensus, and this revision reflects the significantly increased level of consensus arrived at since the original versions. The earlier standards and their modifications specified a number of areas where consensus had not been reached before, and these are now reflected in this revision. The authors of the original versions tried, as much as possible, to follow the principles below when creating new specifications:

1. By standardizing an interface like one in an historical implementation; for example, directories

2. By specifying an interface that is readily implementable in terms of, and backwards compatible with, historical implementations, such as the extended *tar* format defined in the *pax* utility

3. By specifying an interface that, when added to an historical implementation, will not conflict with it; for example, the *sigaction*( ) function

This revision tries to minimize the number of changes required to implementations which conform to the earlier versions of the approved standards to bring them into conformance with the current standard. Specifically, the scope of this work excluded doing any ‘‘new’’ work, but rather collecting into a single document what had been spread across a number of documents, and presenting it in what had been proven in practice to be a more effective way. Some changes to prior conforming implementations were unavoidable, primarily as a consequence of resolving conflicts found in prior revisions, or which became apparent when bringing the various pieces together.

However, since it references the 1999 versions of the ISO C standard, and no longer supports ‘‘Common Usage C’’, there are a number of unavoidable changes. Applications portability is similarly affected.

IEEE Std 1003.1-200x is specifically not a codification of a particular vendor’s product.

It should be noted that implementations will have different kinds of extensions. Some will reflect ‘‘historical usage’’ and will be preserved for execution of pre-existing applications. These functions should be considered ‘‘obsolescent’’ and the standard functions used for new applications. Some extensions will represent functions beyond the scope of IEEE Std 1003.1-200x. These need to be used with careful management to be able to adapt to future IEEE Std 1003.1-200x extensions and/or port to implementations that provide these services in a different manner.

- Minimal Changes to Existing Application Code

  A goal of IEEE Std 1003.1-200x was to minimize additional work for the developers of applications. However, because every known historical implementation will have to change at least slightly to conform, some applications will have to change.

IEEE Std 1003.1-200x

IEEE Std 1003.1-200x defines the Portable Operating System Interface (POSIX) requirements and consists of the following volumes:

- Base Definitions
- Shell and Utilities (this volume)
- System Interfaces

- Rationale (Informative)

This Volume

The Shell and Utilities volume of IEEE Std 1003.1-200x describes the commands and utilities offered to application programs on POSIX-conformant systems. Readers are expected to be familiar with the Base Definitions volume of IEEE Std 1003.1-200x.

This volume of IEEE Std 1003.1-200x is structured as follows:

- Chapter 1 explains the status of this volume of IEEE Std 1003.1-200x and its relationship to other formal standards. It also describes the defaults used by the utility descriptions in Chapter 4.
- Chapter 2 describes the command language used in POSIX-conformant systems.
- Chapter 4 consists of reference pages for all utilities available on POSIX-conformant systems.

Comprehensive references are available in the index.

Typographical Conventions

The following typographical conventions are used throughout IEEE Std 1003.1-200x.

The typographical conventions listed here are for ease of reading only. Editorial inconsistencies in the use of typography are unintentional and have no normative meaning in IEEE Std 1003.1-200x.

Reference	Example	Notes
C-Language Data Structure	**aiocb**	
C-Language Data Structure Member	*aio_lio_opcode*	
C-Language Data Type	**long**	
C-Language Function	*system*( )	
C-Language Function Family	*exec*	
C-Language Function Argument	*arg1*	
C-Language External Variable	*errno*	
C-Language Header	**<sys/stat.h>**	
C-Language Keyword	**#define**	
C-Language Macro with Argument	*assert*( )	
C-Language Macro with No Argument	INET_ADDRSTRLEN	
Commands within a Utility	**a**, **c**	
Conversion Specification, Specifier / Modifier Character	%A, g, E	1
Environment Variable	*PATH*	
Error Number	[EINTR]	
Example Output	**Hello, World**	
Filename	**/tmp**	
Literal Character	'c'	2
Literal String	"abcde"	2
Optional Items in Utility Syntax	**[ ]**	
Parameter	<*directory pathname*>	
Special Character	<newline>	3
Symbolic Limit, Configuration Value	{LINE_MAX}	4

Reference	Example	Notes
Symbolic Constant	_POSIX_VDISABLE	
Syntax	`#include <sys/stat.h>`	
User Input and Example Code	`echo Hello, World`	5
Utility Name	*awk*	
Utility Operand	*file_name*	
Utility Option	−**c**	
Utility Option with Option-Argument	−**w** *width*	

Notes:

1. Conversion specifications, specifier characters, and modifier characters are used primarily in date-related functions and utilities and the *fprintf* and *fscanf* formatting functions.
2. Unless otherwise noted, the quotes shall not be used as input or output. When used in a list item, the quotes are omitted.
3. The style selected for some of the special characters, such as <newline>, matches the form of the input given to the *localedef* utility. Generally, the characters selected for this special treatment are those that are not visually distinct, such as the control characters <tab> or <newline>.
4. Names surrounded by braces represent symbolic limits or configuration values which may be declared in appropriate headers by means of the C **#define** construct.
5. Brackets shown in this font, `"[ ]"`, are part of the syntax and do *not* indicate optional items. In syntax the `'|'` symbol is used to separate alternatives, and ellipses (`"..."`) are used to show that additional arguments are optional.

Shading is used to identify extensions and options; see Section 1.8.1 (on page 2210).

Footnotes and notes within the body of the normative text are for information only (informative).

Informative sections (such as Rationale, Change History, Application Usage, and so on) are denoted by continuous shading bars in the margins.

Ranges of values are indicated with parentheses or brackets as follows:

— (*a*,*b*) means the range of all values from *a* to *b*, including neither *a* nor *b*.

— [*a*,*b*] means the range of all values from *a* to *b*, including *a* and *b*.

— [*a*,*b*) means the range of all values from *a* to *b*, including *a*, but not *b*.

— (*a*,*b*] means the range of all values from *a* to *b*, including *b*, but not *a*.

Trademarks

The following information is given for the convenience of users of IEEE Std 1003.1-200x and does not constitute an endorsement by The Open Group or IEEE of these products. There may be other products mentioned in the text that might be covered by trademark protection and readers are advised to verify them independently.

AIX® is a registered trademark of IBM Corporation.

AT&T® is a registered trademark of AT&T in the U.S.A. and other countries.

BSD™ is a trademark of the University of California, Berkeley, U.S.A.

Hewlett-Packard®, HP®, and HP-UX® are registered trademarks of Hewlett-Packard Company.

IBM® is a registered trademark of International Business Machines Corporation.

Motif®, OSF/1®, UNIX®, and the ‘‘X Device’’ are registered trademarks and IT DialTone™ and The Open Group™ are trademarks of The Open Group in the U.S. and other countries.

POSIX® is a registered trademark of the Institute of Electrical and Electronic Engineers, Inc.

Sun® and Sun Microsystems® are registered trademarks of Sun Microsystems, Inc.

/usr/group® is a registered trademark of UniForum, the International Network of UNIX System Users.

Acknowledgements

The contributions of the following organizations to the development of IEEE Std 1003.1-200x are gratefully acknowledged:

- AT&T for permission to reproduce portions of its copyrighted System V Interface Definition (SVID) and material from the UNIX System V Release 2.0 documentation.
- The SC22 WG14 Committees.

This document was prepared by the Austin Group, a joint working group of the IEEE, The Open Group, and ISO SC22 WG15.

Austin Group Common Standards Committee

At the time IEEE Std 1003.1-200x was approved, the membership was as follows:

Chair
Andrew Josey

Organizational Representatives
Donald W. Cragun [IEEE PASC]
Nicholas Stoughton [ISO / SC22 WG15]
Mark Brown [The Open Group]

Editor
Cathy Hughes

Technical Reviewers

Peter Anvin
Bouazza Bachar
Theodore P. Baker
Walter Briscoe
Dave Butenhof
Mark Brown
Geoff Clare
Donald W. Cragun
Lee Damico
Ulrich Drepper
Paul Eggert
Joanna Farley
Clive D.W. Feather
Andrew Gollan
Michael Gonzalez
Joseph M. Gwinn
Yvette Ho Sang
Cathy Hughes
Lowell G. Johnson
Andrew Josey
Michael Kavanaugh
David Korn
Marc Aurele La France
Jim Meyering
Gary Miller
Finnbarr P. Murphy
Joseph S. Myers
Sandra O'Donnell
Frank Prindle
Curtis Royster Jr.
Glen Seeds
Keld Simonsen
Raja Srinivasan
Nicholas Stoughton
Donn S. Terry
Fred Tydeman
Peter Van Der Veen
James Youngman
Jim Zepeda
Jason Zions

Working Group

Harold C. Adams
Pierre-Jean Arcos
Jay Ashford
Peter Anvin
Karen D. Gordon
Joseph M. Gwinn
Steven A. Haaser
Charles E. Hammons
Frank Prindle
Francois Riche
John D. Riley
Andrew K. Roach

Bouazza Bachar
Theodore P. Baker
Robert Barned
Joel Berman
Andries Brouwer
David J. Blackwood
Shirley Bockstahler-Brandt
James Bottomley
Walter Briscoe
Mark Brown
Eric W. Burger
Alan Burns
Dave Butenhof
Keith Chow
Geoff Clare
Donald W. Cragun
Lee Damico
Juan Antonio De La Puente
Ming De Zhou
Steven J. Dovich
Richard P. Draves
Ulrich Drepper
Paul Eggert
Philip H. Enslow
Joanna Farley
Clive D.W. Feather
Pete Forman
Mark Funkenhauser
Lois Goldthwaite
Andrew Gollan
Michael Gonzalez
Chris J. Harding
Barry Hedquist
Vincent E. Henley
Karl Heubaum
Jon Hitchcock
Yvette Ho Sang
Niklas Holsti
Thomas Hosmer
Cathy Hughes
Jim D Isaak
Lowell G. Johnson
Michael B. Jones
Andrew Josey
Michael J. Karels
Michael Kavanaugh
David Korn
Steven Kramer
Thomas M. Kurihara
Marc Aurele La France
C. Douglass Locke
Roger J Martin
Nick Maclaren
Craig H. Meyer
Jim Meyering
Gary Miller
Finnbarr P. Murphy
Joseph S. Myers
John Napier
Peter E. Obermayer
James T. Oblinger
Sandra O'Donnell
Helmut Roth
Jaideep Roy
Curtis Royster Jr.
Stephen C. Schwarm
Richard Seibel
David L. Shroads Jr.
Curtis Smith
W. Olin Sibert
Glen Seeds
Keld Jorn Simonsen
Raja Srinivasan
Nicholas Stoughton
Donn S. Terry
Marc J. Teller
Fred Tydeman
Mark-Rene Uchida
Scott A. Valcourt
Michael W. Vannier
Frederick N. Webb
Paul A.T. Wolfgang
Garrett Wollman
Peter Van Der Veen
Eric Vought
X/Open Company Ltd.
James Youngman
Oren Yuen
Janusz Zalewski
Jim Zepeda
Jason Zions

Referenced Documents

Normative References

Normative references for IEEE Std 1003.1-200x are defined in the Base Definitions volume of IEEE Std 1003.1-200x.

Informative References

The following documents are referenced in IEEE Std 1003.1-200x:

1984 /usr/group Standard
/usr/group Standards Committee, Santa Clara, CA, UniForum 1984.

Almasi and Gottlieb
George S. Almasi and Allan Gottlieb, *Highly Parallel Computing*, The Benjamin/Cummings Publishing Company, Inc., 1989, ISBN: 0-8053-0177-1.

ANSI C
American National Standard for Information Systems: Standard X3.159-1989, Programming Language C.

ANS X3.226-1994
American National Standard for Information Systems: Standard X3.226-1994, Programming Language Common LISP.

Brawer
Steven Brawer, *Introduction to Parallel Programming*, Academic Press, 1989, ISBN: 0-12-128470-0.

DeRemer and Pennello Article
DeRemer, Frank and Pennello, Thomas J., *Efficient Computation of LALR(1) Look-Ahead Sets*, SigPlan Notices, Volume 15, No. 8, August 1979.

Draft ANSI X3J11.1
IEEE Floating Point draft report of ANSI X3J11.1 (NCEG).

FIPS 151-1
Federal Information Procurement Standard (FIPS) 151-1.

FIPS 151-2
Federal Information Procurement Standards (FIPS) 151-2, Portable Operating System Interface (POSIX)—Part 1: System Application Program Interface (API) [C Language].

HP-UX Manual
Hewlett-Packard HP-UX Release 9.0 Reference Manual, Third Edition, August 1992.

IEC 60559:1989
IEC 60559:1989, Binary Floating-Point Arithmetic for Microprocessor Systems (previously designated IEC 559:1989).

IEEE Std 754-1985
Standard for Binary Floating-Point Arithmetic.

IEEE Std 854-1987
Standard for Radix-Independent Floating-Point Arithmetic.

IEEE Std 1003.9-1992
Standard for Information Technology — POSIX FORTRAN 77 Language Interfaces — Part 1: Binding for System Application Program Interface API.

IETF RFC 791
Internet Protocol, Version 4 (IPv4), September 1981.

IETF RFC 819
The Domain Naming Convention for Internet User Applications, Z. Su, J. Postel, August 1982.

IETF RFC 822
Standard for the Format of ARPA Internet Text Messages, D.H. Crocker, August 1982.

IETF RFC 919
Broadcasting Internet Datagrams, J. Mogul, October 1984.

IETF RFC 920
Domain Requirements, J. Postel, J. Reynolds, October 1984.

IETF RFC 921
Domain Name System Implementation Schedule, J. Postel, October 1984.

IETF RFC 922
Broadcasting Internet Datagrams in the Presence of Subnets, J. Mogul, October 1984.

IETF RFC 1034
Domain Names — Concepts and Facilities, P. Mockapetris, November 1987.

IETF RFC 1035
Domain Names — Implementation and Specification, P. Mockapetris, November 1987.

IETF RFC 1123
Requirements for Internet Hosts — Application and Support, R. Braden, October 1989.

IETF RFC 1886
DNS Extensions to Support Internet Protocol, Version 6 (IPv6), C. Huitema, S. Thomson, December 1995.

IETF RFC 2045
Multipurpose Internet Mail Extensions (MIME), Part 1: Format of Internet Message Bodies, N. Freed, N. Borenstein, November 1996.

IETF RFC 2373
Internet Protocol, Version 6 (IPv6) Addressing Architecture, S. Deering, R. Hinden, July 1998.

IETF RFC 2460
Internet Protocol, Version 6 (IPv6), S. Deering, R. Hinden, December 1998.

Internationalisation Guide
Guide, July 1993, Internationalisation Guide, Version 2 (ISBN: 1-859120-02-4, G304), published by The Open Group.

ISO C (1990)
ISO / IEC 9899: 1990: Programming Languages — C, including Amendment 1: 1995 (E), C Integrity (Multibyte Support Extensions (MSE) for ISO C).

ISO 2375: 1985
ISO 2375: 1985, Data Processing — Procedure for Registration of Escape Sequences.

ISO 8652: 1987
ISO 8652: 1987, Programming Languages — Ada (technically identical to ANSI standard 1815A-1983).

ISO / IEC 1539: 1990
ISO / IEC 1539: 1990, Information Technology — Programming Languages — Fortran (technically identical to the ANSI X3.9-1978 standard [FORTRAN 77]).

ISO / IEC 4873: 1991
ISO / IEC 4873: 1991, Information Technology — ISO 8-bit Code for Information Interchange — Structure and Rules for Implementation.

ISO / IEC 6429: 1992
ISO / IEC 6429: 1992, Information Technology — Control Functions for Coded Character Sets.

ISO / IEC 6937: 1994
ISO / IEC 6937: 1994, Information Technology — Coded Character Set for Text Communication — Latin Alphabet.

ISO 7-bit or 8-bit coded character set for text communication using public communication networks, private communication networks, or interchange media, such as magnetic tapes and discs.

ISO / IEC 8802-3: 1996
ISO / IEC 8802-3: 1996, Information Technology — Telecommunications and Information Exchange Between Systems — Local and Metropolitan Area Networks — Specific Requirements — Part 3: Carrier Sense Multiple Access with Collision Detection (CSMA / CD) Access Method and Physical Layer Specifications.

ISO / IEC 8859
ISO / IEC 8859, Information Technology — 8-Bit Single-Byte Coded Graphic Character Sets:

Part 1: Latin Alphabet No. 1
Part 2: Latin Alphabet No. 2
Part 3: Latin Alphabet No. 3
Part 4: Latin Alphabet No. 4
Part 5: Latin / Cyrillic Alphabet
Part 6: Latin / Arabic Alphabet
Part 7: Latin / Greek Alphabet
Part 8: Latin / Hebrew Alphabet
Part 9: Latin Alphabet No. 5
Part 10: Latin Alphabet No. 6
Part 13: Latin Alphabet No. 7
Part 14: Latin Alphabet No. 8
Part 15: Latin Alphabet No. 9

This standard character set comprises 191 graphic characters covering the requirements of most of Western Europe.

ISO POSIX-1: 1996
ISO / IEC 9945-1: 1996, Information Technology — Portable Operating System Interface (POSIX) — Part 1: System Application Program Interface (API) [C Language] (identical to ANSI / IEEE Std 1003.1-1996). Incorporating ANSI / IEEE Stds 1003.1-1990, 1003.1b-1993, 1003.1c-1995, and 1003.1i-1995.

ISO POSIX-2: 1993
ISO / IEC 9945-2: 1993, Information Technology — Portable Operating System Interface

(POSIX) — Part 2: Shell and Utilities (identical to ANSI / IEEE Std 1003.2-1992, as amended by ANSI / IEEE Std 1003.2a-1992).

Issue 1
X / Open Portability Guide, July 1985 (ISBN: 0-444-87839-4).

Issue 2
X / Open Portability Guide, January 1987:

- Volume 1: XVS Commands and Utilities (ISBN: 0-444-70174-5)
- Volume 2: XVS System Calls and Libraries (ISBN: 0-444-70175-3)

Issue 3
X / Open Specification, 1988, 1989, February 1992:

- Commands and Utilities, Issue 3 (ISBN: 1-872630-36-7, C211); this specification was formerly X / Open Portability Guide, Issue 3, Volume 1, January 1989, XSI Commands and Utilities (ISBN: 0-13-685835-X, XO / XPG / 89 / 002)
- System Interfaces and Headers, Issue 3 (ISBN: 1-872630-37-5, C212); this specification was formerly X / Open Portability Guide, Issue 3, Volume 2, January 1989, XSI System Interface and Headers (ISBN: 0-13-685843-0, XO / XPG / 89 / 003)
- Curses Interface, Issue 3, contained in Supplementary Definitions, Issue 3 (ISBN: 1-872630-38-3, C213), Chapters 9 to 14 inclusive; this specification was formerly X / Open Portability Guide, Issue 3, Volume 3, January 1989, XSI Supplementary Definitions (ISBN: 0-13-685850-3, XO / XPG / 89 / 004)
- Headers Interface, Issue 3, contained in Supplementary Definitions, Issue 3 (ISBN: 1-872630-38-3, C213), Chapter 19, Cpio and Tar Headers; this specification was formerly X / Open Portability Guide Issue 3, Volume 3, January 1989, XSI Supplementary Definitions (ISBN: 0-13-685850-3, XO / XPG / 89 / 004)

Issue 4
CAE Specification, July 1992, published by The Open Group:

- System Interface Definitions (XBD), Issue 4 (ISBN: 1-872630-46-4, C204)
- Commands and Utilities (XCU), Issue 4 (ISBN: 1-872630-48-0, C203)
- System Interfaces and Headers (XSH), Issue 4 (ISBN: 1-872630-47-2, C202)

Issue 4, Version 2
CAE Specification, August 1994, published by The Open Group:

- System Interface Definitions (XBD), Issue 4, Version 2 (ISBN: 1-85912-036-9, C434)
- Commands and Utilities (XCU), Issue 4, Version 2 (ISBN: 1-85912-034-2, C436)
- System Interfaces and Headers (XSH), Issue 4, Version 2 (ISBN: 1-85912-037-7, C435)

Issue 5
Technical Standard, February 1997, published by The Open Group:

- System Interface Definitions (XBD), Issue 5 (ISBN: 1-85912-186-1, C605)
- Commands and Utilities (XCU), Issue 5 (ISBN: 1-85912-191-8, C604)
- System Interfaces and Headers (XSH), Issue 5 (ISBN: 1-85912-181-0, C606)

Knuth Article
Knuth, Donald E., *On the Translation of Languages from Left to Right*, Information and Control,

Volume 8, No. 6, October 1965.

KornShell
Bolsky, Morris I. and Korn, David G., *The New KornShell Command and Programming Language*, March 1995, Prentice Hall.

MSE working draft
Working draft of ISO/IEC 9899:1990/Add3:draft, Addendum 3 — Multibyte Support Extensions (MSE) as documented in the ISO Working Paper SC22/WG14/N205 dated 31 March 1992.

POSIX.0:1995
IEEE Std 1003.0-1995, Guide to the POSIX Open System Environment (OSE) (identical to ISO/IEC TR 14252).

POSIX.1:1988
IEEE Std 1003.1-1988, Standard for Information Technology — Portable Operating System Interface (POSIX) — Part 1: System Application Program Interface (API) [C Language].

POSIX.1:1990
IEEE Std 1003.1-1990, Standard for Information Technology — Portable Operating System Interface (POSIX) — Part 1: System Application Program Interface (API) [C Language].

POSIX.1a:2000
IEEE Std 1003.1a-2000, Standard for Information Technology — Portable Operating System Interface (POSIX) — Part 1: System Application Program Interface (API) — Amendment ????: TITLE?? [C Language].

POSIX.1d:1999
IEEE Std 1003.1d-1999, Standard for Information Technology — Portable Operating System Interface (POSIX) — Part 1: System Application Program Interface (API) — Amendment 4: Additional Realtime Extensions [C Language].

POSIX.1g:2000
IEEE Std 1003.1g-2000, Standard for Information Technology — Portable Operating System Interface (POSIX) — Part 1: System Application Program Interface (API) — Amendment 6: Protocol-Independent Interfaces (PII).

POSIX.1j:2000
IEEE Std 1003.1j-2000, Standard for Information Technology — Portable Operating System Interface (POSIX) — Part 1: System Application Program Interface (API) — Amendment 5: Advanced Realtime Extensions [C Language].

POSIX.1q:2000
IEEE Std 1003.1q-2000, Standard for Information Technology — Portable Operating System Interface (POSIX) — Part 1: System Application Program Interface (API) — Amendment 7: Tracing [C Language].

POSIX.2b:-2000
IEEE Std 1003.2b:2000, Standard for Information Technology — Portable Operating System Interface (POSIX) — Part 2: Shell and Utilities — Amendment ???: TITLE??.

POSIX.2d:-1994
IEEE Std 1003.2d:1994, Standard for Information Technology — Portable Operating System Interface (POSIX) — Part 2: Shell and Utilities — Amendment 1: Batch Environment.

POSIX.13:-1998
IEEE Std 1003.13:1998, Standard for Information Technology — Standardized Application Environment Profile (AEP) — POSIX Realtime Application Support.

Sarwate Article

Sarwate, Dilip V., *Computation of Cyclic Redundancy Checks via Table Lookup*, Communications of the ACM, Volume 30, No. 8, August 1988.

Sprunt, Sha, and Lehoczky

Sprunt, B., Sha, L., and Lehoczky, J.P., *Aperiodic Task Scheduling for Hard Real-Time Systems*, The Journal of Real-Time Systems, Volume 1, 1989, Pages 27-60.

SVID, Issue 1

American Telephone and Telegraph Company, System V Interface Definition (SVID), Issue 1; Morristown, NJ, UNIX Press, 1985.

SVID, Issue 2

American Telephone and Telegraph Company, System V Interface Definition (SVID), Issue 2; Morristown, NJ, UNIX Press, 1986.

SVID, Issue 3

American Telephone and Telegraph Company, System V Interface Definition (SVID), Issue 3; Morristown, NJ, UNIX Press, 1989.

The AWK Programming Language

Aho, Alfred V., Kernighan, Brian W., and Weinberger, Peter J., *The AWK Programming Language*, Reading, MA, Addison-Wesley 1988.

UNIX Programmer's Manual

American Telephone and Telegraph Company, *UNIX Time-Sharing System: UNIX Programmer's Manual*, 7th Edition, Murray Hill, NJ, Bell Telephone Laboratories, January 1979.

XNS, Issue 4

CAE Specification, August 1994, Networking Services, Issue 4 (ISBN: 1-85912-049-0, C438), published by The Open Group.

XNS, Issue 5

CAE Specification, February 1997, Networking Services, Issue 5 (ISBN: 1-85912-165-9, C523), published by The Open Group.

XNS, Issue 5.2

Technical Standard, January 2000, Networking Services (XNS), Issue 5.2 (ISBN: 1-85912-241-8, C808), published by The Open Group.

X/Open Curses, Issue 4, Version 2

CAE Specification, May 1996, X/Open Curses, Issue 4, Version 2 (ISBN: 1-85912-171-3, C610), published by The Open Group.

Yacc

Yacc: Yet Another Compiler Compiler, Stephen C. Johnson, 1978.

Parts of the following documents were used to create the base documents for IEEE Std 1003.1-200x:

AIX 3.2 Manual

AIX Version 3.2 For RISC System/6000, Technical Reference: Base Operating System and Extensions, 1990, 1992 (Part No. SC23-2382-00).

OSF/1

OSF/1 Programmer's Reference, Release 1.2 (ISBN: 0-13-020579-6).

OSF AES

Application Environment Specification (AES) Operating System Programming Interfaces

Volume, Revision A (ISBN: 0-13-043522-8).

System V Release 2.0

— UNIX System V Release 2.0 Programmer's Reference Manual (April 1984 - Issue 2).

— UNIX System V Release 2.0 Programming Guide (April 1984 - Issue 2).

System V Release 4.2
Operating System API Reference, UNIX SVR4.2 (1992) (ISBN: 0-13-017658-3).