
**Information technology — Portable
Operating System Interface (POSIX®)
Base Specifications, Issue 7**

*Technologies de l'information — Spécifications de base de l'interface
pour la portabilité des systèmes (POSIX®), Issue 7*



Reference number
ISO/IEC/IEEE 9945:2009(E)

Copyright © 2001-2008, IEEE and The Open Group.
All rights reserved

PDF disclaimer

This PDF file may contain embedded typefaces. In accordance with Adobe's licensing policy, this file may be printed or viewed but shall not be edited unless the typefaces which are embedded are licensed to and installed on the computer performing the editing. In downloading this file, parties accept therein the responsibility of not infringing Adobe's licensing policy. Neither the ISO Central Secretariat nor IEEE accepts any liability in this area.

Adobe is a trademark of Adobe Systems Incorporated.

Details of the software products used to create this PDF file can be found in the General Info relative to the file; the PDF-creation parameters were optimized for printing. Every care has been taken to ensure that the file is suitable for use by ISO member bodies and IEEE members. In the unlikely event that a problem relating to it is found, please inform the ISO Central Secretariat or IEEE at the address given below.



COPYRIGHT PROTECTED DOCUMENT

© IEEE 2001-2008

All rights reserved. Unless otherwise specified, no part of this publication may be reproduced or utilized in any form or by any means, electronic or mechanical, including photocopying and microfilm, without permission in writing from either ISO or IEEE at the respective address below.

ISO copyright office
Case postale 56 • CH-1211 Geneva 20
Tel. + 41 22 749 01 11
Fax + 41 22 749 09 47
E-mail copyright@iso.org
Web www.iso.org

Institute of Electrical and Electronics Engineers, Inc.
3 Park Avenue, New York • NY 10016-5997, USA
E-mail stds.ipr@ieee.org
Web www.ieee.org

Published in Switzerland

Foreword

ISO (the International Organization for Standardization) and IEC (the International Electrotechnical Commission) form the specialized system for worldwide standardization. National bodies that are members of ISO or IEC participate in the development of International Standards through technical committees established by the respective organization to deal with particular fields of technical activity. ISO and IEC technical committees collaborate in fields of mutual interest. Other international organizations, governmental and non-governmental, in liaison with ISO and IEC, also take part in the work. In the field of information technology, ISO and IEC have established a joint technical committee, ISO/IEC JTC 1.

IEEE Standards documents are developed within the IEEE Societies and the Standards Coordinating Committees of the IEEE Standards Association (IEEE-SA) Standards Board. The IEEE develops its standards through a consensus development process, approved by the American National Standards Institute, which brings together volunteers representing varied viewpoints and interests to achieve the final product. Volunteers are not necessarily members of the Institute and serve without compensation. While the IEEE administers the process and establishes rules to promote fairness in the consensus development process, the IEEE does not independently evaluate, test, or verify the accuracy of any of the information contained in its standards.

The main task of ISO/IEC JTC 1 is to prepare International Standards. Draft International Standards adopted by the joint technical committee are circulated to national bodies for voting. Publication as an International Standard requires approval by at least 75 % of the national bodies casting a vote.

Attention is called to the possibility that implementation of this standard may require the use of subject matter covered by patent rights. By publication of this standard, no position is taken with respect to the existence or validity of any patent rights in connection therewith. ISO/IEEE is not responsible for identifying essential patents or patent claims for which a license may be required, for conducting inquiries into the legal validity or scope of patents or patent claims or determining whether any licensing terms or conditions provided in connection with submission of a Letter of Assurance or a Patent Statement and Licensing Declaration Form, if any, or in any licensing agreements are reasonable or non-discriminatory. Users of this standard are expressly advised that determination of the validity of any patent rights, and the risk of infringement of such rights, is entirely their own responsibility. Further information may be obtained from ISO or the IEEE Standards Association.

ISO/IEC/IEEE 9945 was prepared by The Open Group (as The Open Group Technical Standard Base Specifications, Issue 7) and the Portable Applications Standards Committee of the Computer Society of the IEEE (as IEEE Std 1003.1™-2008). It was adopted by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 22, *Programming languages, their environments and system software interfaces*, in parallel with its approval by the ISO/IEC national bodies, under the “fast-track procedure” defined in the Partner Standards Development Organization cooperation agreement between ISO and IEEE. IEEE is responsible for the maintenance of this document with participation and input from ISO/IEC national bodies.

This first edition of ISO/IEC/IEEE 9945 cancels and replaces ISO/IEC 9945-1:2003, ISO/IEC 9945-2:2003, ISO/IEC 9945-3:2003 and ISO/IEC 9945-4:2003, which have been technically revised. It also incorporates the Technical Corrigenda ISO/IEC 9945-1:2003/Cor.1:2004, ISO/IEC 9945-2:2003/Cor.1:2004, ISO/IEC 9945-3:2003/Cor.1:2004 and ISO/IEC 9945-4:2003/Cor.1:2004.

(blank page)



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

Base Specifications, Issue 7

IEEE Computer Society

Sponsored by the
Portable Applications Standards Committee
and
The Open Group

IEEE
3 Park Avenue
New York, NY 10016-5997, USA
1 December 2008

IEEE Std 1003.1™-2008
(Revision of
IEEE Std 1003.1-2004)

(blank page)

IEEE Std 1003.1™ -2008
(Revision of
IEEE Std 1003.1-2004)

The Open Group Technical Standard
Base Specifications, Issue 7

Standard for Information Technology— Portable Operating System Interface (POSIX®) Base Specifications, Issue 7

Sponsor

Portable Applications Standards Committee

of the

IEEE Computer Society

and

The Open Group

Approved 26 September 2008

IEEE-SA Standards Board

Approved 24 July 2008

The Open Group



Abstract

POSIX.1-2008 is simultaneously IEEE Std 1003.1™-2008 and The Open Group Technical Standard Base Specifications, Issue 7.

POSIX.1-2008 defines a standard operating system interface and environment, including a command interpreter (or “shell”), and common utility programs to support applications portability at the source code level. POSIX.1-2008 is intended to be used by both application developers and system implementors and comprises four major components (each in an associated volume):

- General terms, concepts, and interfaces common to all volumes of this standard, including utility conventions and C-language header definitions, are included in the Base Definitions volume.
- Definitions for system service functions and subroutines, language-specific system services for the C programming language, function issues, including portability, error handling, and error recovery, are included in the System Interfaces volume.
- Definitions for a standard source code-level interface to command interpretation services (a “shell”) and common utility programs for application programs are included in the Shell and Utilities volume.
- Extended rationale that did not fit well into the rest of the document structure, which contains historical information concerning the contents of POSIX.1-2008 and why features were included or discarded by the standard developers, is included in the Rationale (Informative) volume.

The following areas are outside the scope of POSIX.1-2008:

- Graphics interfaces
- Database management system interfaces
- Record I/O considerations
- Object or binary code portability
- System configuration and resource availability

POSIX.1-2008 describes the external characteristics and facilities that are of importance to application developers, rather than the internal construction techniques employed to achieve these capabilities. Special emphasis is placed on those functions and facilities that are needed in a wide variety of commercial applications.

Keywords

application program interface (API), argument, asynchronous, basic regular expression (BRE), batch job, batch system, built-in utility, byte, child, command language interpreter, CPU, extended regular expression (ERE), FIFO, file access control mechanism, input/output (I/O), job control, network, portable operating system interface (POSIX®), parent, shell, stream, string, synchronous, system, thread, X/Open System Interface (XSI)

The Institute of Electrical and Electronics Engineers, Inc.
3 Park Avenue, New York, NY 10016-5997, USA

The Open Group
Thames Tower, Station Road, Reading, Berkshire, RG1 1LX, U.K.

Copyright © 2008 by the Institute of Electrical and Electronics Engineers, Inc. and The Open Group
All rights reserved.

Published 1 December 2008 by the IEEE. Printed in the United States of America by the IEEE.

PDF: ISBN 978-0-7381-5798-6 STD95820

CDROM: ISBN 978-0-7381-5799-3 STD95820

Published 1 December 2008 by The Open Group. Printed in the United Kingdom by The Open Group.

Doc. Number: C082

ISBN: 1-931624-79-8

No part of this publication may be reproduced in any form, in an electronic retrieval system or otherwise, without the prior written permission of the publisher. Permission to reproduce all or any part of this standard must be with the consent of both copyright holders and may be subject to a license fee. Both copyright holders will need to be satisfied that the other has granted permission. Requests should be sent by email to austin-group-permissions@opengroup.org.

This standard has been prepared by the Austin Group. Feedback relating to the material contained within this standard may be submitted by using the Austin Group web site at www.opengroup.org/austin/defectform.html.

IEEE

IEEE Standards documents are developed within the IEEE Societies and the Standards Coordinating Committees of the IEEE Standards Association (IEEE-SA) Standards Board. The IEEE develops its standards through a consensus development process, approved by the American National Standards Institute, which brings together volunteers representing varied viewpoints and interests to achieve the final product. Volunteers are not necessarily members of the Institute and serve without compensation. While the IEEE administers the process and establishes rules to promote fairness in the consensus development process, the IEEE does not independently evaluate, test, or verify the accuracy of any of the information contained in its standards.

Use of an IEEE Standard is wholly voluntary. The IEEE disclaims liability for any personal injury, property, or other damage, of any nature whatsoever, whether special, indirect, consequential, or compensatory, directly or indirectly resulting from the publication, use of, or reliance upon this, or any other IEEE Standard document.

The IEEE does not warrant or represent the accuracy or content of the material contained herein, and expressly disclaims any express or implied warranty, including any implied warranty of merchantability or fitness for a specific purpose, or that the use of the material contained herein is free from patent infringement. IEEE Standards documents are supplied “AS IS”.

The existence of an IEEE Standard does not imply that there are no other ways to produce, test, measure, purchase, market, or provide other goods and services related to the scope of the IEEE Standard. Furthermore, the viewpoint expressed at the time a standard is approved and issued is subject to change brought about through developments in the state of the art and comments received from users of the standard. Every IEEE Standard is subjected to review at least every five years for revision or reaffirmation. When a document is more than five years old and has not been reaffirmed, it is reasonable to conclude that its contents, although still of some value, do not wholly reflect the present state of the art. Users are cautioned to check to determine that they have the latest edition of any IEEE Standard.

In publishing and making this document available, the IEEE is not suggesting or rendering professional or other services for, or on behalf of, any person or entity. Nor is the IEEE undertaking to perform any duty owed by any other person or entity to another. Any person utilizing this, and any other IEEE Standards document, should rely upon the advice of a competent professional in determining the exercise of reasonable care in any given circumstances.

Interpretations: Occasionally questions may arise regarding the meaning of portions of standards as they relate to specific applications. When the need for interpretations is brought to the attention of IEEE, the Institute will initiate action to prepare appropriate responses. Since IEEE Standards represent a consensus of concerned interests, it is important to ensure that any interpretation has also received the concurrence of a balance of interests. For this reason, IEEE and the members of its societies and Standards Coordinating Committees are not able to provide an instant response to interpretation requests except in those cases where the matter has previously received formal consideration. At lectures, symposia, seminars, or educational courses, an individual presenting information on IEEE Standards shall make it clear that his or her views should be considered the personal views of that individual rather than the formal position, explanation, or interpretation of the IEEE.

Comments for revision of IEEE Standards are welcome from any interested party, regardless of membership affiliation with IEEE.^A Suggestions for changes in documents should be in the form of a proposed change of text, together with appropriate supporting comments. Comments on standards and requests for interpretations should be addressed to:

Secretary, IEEE-SA Standards Board
445 Hoes Lane
Piscataway, NJ 08854
USA

Authorization to photocopy portions of any individual standard for internal or personal use is granted by the Institute of Electrical and Electronics Engineers, Inc., provided that the appropriate fee is paid to Copyright Clearance Center. To arrange for payment of licensing fee, please contact Copyright Clearance Center, Customer Service, 222 Rosewood Drive, Danvers, MA 01923 USA; +1 978 750 8400. Permission to photocopy portions of any individual standard for educational classroom use can also be obtained through the Copyright Clearance Center.

^A For this standard please send comments via the Austin Group, as indicated on page ii.

The Open Group

The Open Group is a vendor-neutral and technology-neutral consortium, whose vision of Boundaryless Information Flow™ will enable access to integrated information within and between enterprises based on open standards and global interoperability. The Open Group works with customers, suppliers, consortia, and other standards bodies. Its role is to capture, understand, and address current and emerging requirements, establish policies, and share best practices; to facilitate interoperability, develop consensus, and evolve and integrate specifications and Open Source technologies; to offer a comprehensive set of services to enhance the operational efficiency of consortia; and to operate the industry's premier certification service, including UNIX® certification.

Further information on The Open Group is available at www.opengroup.org.

The Open Group has over 20 years' experience in developing and operating certification programs and has extensive experience developing and facilitating industry adoption of test suites used to validate conformance to an open standard or specification.

The Open Group publishes a wide range of technical documentation, the main part of which is focused on development of Technical and Product Standards and Guides, but which also includes white papers, technical studies, branding and testing documentation, and business titles. Full details and a catalog are available at www.opengroup.org/bookstore.

As with all *live* documents, Technical Standards and Specifications require revision to align with new developments and associated international standards. To distinguish between revised specifications which are fully backwards compatible and those which are not:

- A new *Version* indicates there is no change to the definitive information contained in the previous publication of that title, but additions/extensions are included. As such, it *replaces* the previous publication.
- A new *Issue* indicates there is substantive change to the definitive information contained in the previous publication of that title, and there may also be additions/extensions. As such, both previous and new documents are maintained as current publications.

Readers should note that Corrigenda may apply to any publication. Corrigenda information is published at www.opengroup.org/corrigenda.

Introduction

This introduction is not part of POSIX.1-2008, Standard for Information Technology – Portable Operating System Interface (POSIX).

This standard was developed, and is maintained, by a joint working group of members of the IEEE Portable Applications Standards Committee, members of The Open Group, and members of ISO/IEC Joint Technical Committee 1. This joint working group is known as the Austin Group.^B

The Austin Group arose out of discussions amongst the parties which started in early 1998, leading to an initial meeting and formation of the group in September 1998. The purpose of the Austin Group is to develop and maintain the core open systems interfaces that are the POSIX[®] 1003.1 (and former 1003.2) standards, ISO/IEC 9945, and the core of the Single UNIX Specification.

The approach to specification development has been one of “write once, adopt everywhere”, with the deliverables being a set of specifications that carry the IEEE POSIX designation, The Open Group's Technical Standard designation, and an ISO/IEC designation.

This unique development has combined both the industry-led efforts and the formal standardization activities into a single initiative, and included a wide spectrum of participants. The Austin Group continues as the maintenance body for this document.

Anyone wishing to participate in the Austin Group should contact the chair with their request. There are no fees for participation or membership. You may participate as an observer or as a contributor. You do not have to attend face-to-face meetings to participate; electronic participation is most welcome. For more information on the Austin Group and how to participate, see www.opengroup.org/austin.

Background

The developers of POSIX.1-2008 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, POSIX.1-2008 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 developers to write portable applications – it was developed with that goal in mind – it has been designated POSIX^C, 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 POSIX.1-2008 is all persons concerned with an industry-wide standard operating system based on the UNIX system. This includes at least four groups of people:

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

^B The Austin Group is named after the location of the inaugural meeting held at the IBM facility in Austin, Texas in September 1998.

^C 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.

Purpose

Several principles guided the development of POSIX.1-2008:

- **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. POSIX.1-2008 codifies the common, existing definition of the UNIX system.
- **Interface, Not Implementation** – POSIX.1-2008 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** – POSIX.1-2008 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. POSIX.1-2008 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 this standard, and functions usable only by the superuser have not been included. Still, an implementation of the standard interface may also implement features not in POSIX.1-2008. POSIX.1-2008 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 POSIX.1-2008 have been kept as minimal as possible. Additional capabilities have been added as optional extensions.
- **Broadly Implementable** – The developers of POSIX.1-2008 endeavored to make all specified functions implementable across a wide range of existing and potential systems, including:
 - All of the current major systems that are ultimately derived from the original UNIX system code (Version 7 or later)
 - Compatible systems that are not derived from the original UNIX system code
 - Emulations hosted on entirely different operating systems
 - Networked systems
 - Distributed systems
 - Systems running on a broad range of hardware

No direct references to this goal appear in POSIX.1-2008, but some results of it are mentioned in the Rationale (Informative) volume.

- **Minimal Changes to Historical Implementations** – When the original version – IEEE Std 1003.1-1988 – 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 system interface standards, the 1992 IEEE shell and utilities standard, the various Open Group (formerly X/Open) specifications, and IEEE Std 1003.1-2001 and its technical corrigenda have consolidated this consensus, and this version reflects the significantly increased level of consensus arrived at since the original versions. The authors of the original versions tried, as much as possible, to follow the principles below when creating new specifications:

- By standardizing an interface like one in an historical implementation; for example, directories
- 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
- By specifying an interface that, when added to an historical implementation, will not conflict with it; for example, the *sigaction()* function

POSIX.1-2008 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 POSIX.1-2008. These need to be used with careful management to be able to adapt to future extensions of POSIX.1-2008 and/or port to implementations that provide these services in a different manner.

- Minimal Changes to Existing Application Code – A goal of POSIX.1-2008 was to minimize additional work for application developers. However, because every known historical implementation will have to change at least slightly to conform, some applications will have to change.

POSIX.1-2008

POSIX.1-2008 defines the Portable Operating System Interface (POSIX) requirements and consists of the following topics arranged as a series of volumes within the standard:

- Base Definitions
- System Interfaces
- Shell and Utilities
- Rationale (Informative)

Base Definitions

The Base Definitions volume provides common definitions for this standard, therefore readers should be familiar with it before using the other volumes.

This volume is structured as follows:

- Chapter 1 is an introduction.
- Chapter 2 defines the conformance requirements.
- Chapter 3 defines general terms used.
- Chapter 4 describes general concepts used.
- Chapter 5 describes the notation used to specify file input and output formats in this volume and the Shell and Utilities volume.
- Chapter 6 describes the portable character set and the process of character set definition.
- Chapter 7 describes the syntax for defining internationalization locales as well as the POSIX locale provided on all systems.
- Chapter 8 describes the use of environment variables for internationalization and other purposes.
- Chapter 9 describes the syntax of pattern matching using regular expressions employed by many utilities and matched by the *regcomp()* and *regexexec()* functions.
- Chapter 10 describes files and devices found on all systems.
- Chapter 11 describes the asynchronous terminal interface for many of the functions in the System Interfaces volume and the *stty* utility in the Shell and Utilities volume.
- Chapter 12 describes the policies for command line argument construction and parsing.
- Chapter 13 defines the contents of headers which declare the functions and global variables, and define types, constants, macros, and data structures that are needed by programs using the services provided by the System Interfaces volume.

Comprehensive references are available in the index.

System Interfaces

The System Interfaces volume describes the interfaces offered to application programs by POSIX-conformant systems. Readers are expected to be experienced C language programmers, and to be familiar with the Base Definitions volume.

This volume is structured as follows:

- Chapter 1 explains the status of this volume and its relationship to other formal standards.
- Chapter 2 contains important concepts, terms, and caveats relating to the rest of this volume.
- Chapter 3 defines the functional interfaces to the POSIX-conformant system.

Comprehensive references are available in the index.

Shell and Utilities

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

This volume is structured as follows:

- Chapter 1 explains the status of this volume and its relationship to other formal standards. It also describes the defaults used by the utility descriptions.
- Chapter 2 describes the command language used in POSIX-conformant systems, and special built-in utilities.
- Chapter 3 describes a set of services and utilities that are implemented on systems supporting the Batch Environment Services and Utilities option.
- Chapter 4 consists of reference pages for all utilities, other than the special built-in utilities described in Chapter 2, available on POSIX-conformant systems.

Comprehensive references are available in the index.

Rationale (Informative)

The Rationale volume is published to assist in the process of review. It contains historical information concerning the contents of this standard and why features were included or discarded by the standard developers. It also contains notes of interest to application programmers on recommended programming practices, emphasizing the consequences of some aspects of POSIX.1-2008 that may not be immediately apparent.

This volume is organized in parallel to the normative volumes of this standard, with a separate part for each of the three normative volumes.

Within this volume, the following terms are used:

- Base standard – The portions of POSIX.1-2008 that are not optional, equivalent to the definitions of *classic* POSIX.1 and POSIX.2.
- POSIX.0 – Although this term is not used in the normative text of POSIX.1-2008, it is used in this volume to refer to IEEE Std 1003.0™-1995.
- POSIX.1b – Although this term is not used in the normative text of POSIX.1-2008, it is used in this volume to refer to the elements of the POSIX Realtime Extension amendment. (This was earlier referred to as POSIX.4 during the standard development process.)
- POSIX.1c – Although this term is not used in the normative text of POSIX.1-2008, it is used in this volume to refer to the POSIX Threads Extension amendment. (This was earlier referred to as POSIX.4a during the standard development process.)
- Standard developers – The individuals and companies in the development organizations responsible for POSIX.1-2008: the IEEE P1003.1 working groups, The Open Group Base working group, advised by the hundreds of individual technical experts who balloted the draft standards within the Austin Group, and the member bodies and technical experts of ISO/IEC JTC 1/SC 22.
- XSI option – The portions of POSIX.1-2008 addressing the extension added for support of the Single UNIX Specification.

Typographical Conventions

The following typographical conventions are used throughout this standard. In the text, this standard is referred to as POSIX.1-2008, which is technically identical to The Open Group Base Specifications, Issue 7.

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 POSIX.1-2008.

Reference	Example	Notes
C-Language Data Structure	aiocb	
C-Language Data Structure Member	<i>aio_lio_opcode</i>	
C-Language Data Type	long	
C-Language External Variable	<i>errno</i>	
C-Language Function	<i>system()</i>	
C-Language Function Argument	<i>arg</i>	
C-Language Function Family	<i>exec</i>	
C-Language Header	<sys/stat.h>	
C-Language Keyword	return	
C-Language Macro with Argument	<i>assert()</i>	
C-Language Macro with No Argument	NET_ADDRSTRLEN	
C-Language Preprocessing Directive	#define	
Commands within a Utility	a, c	
Conversion Specifier, Specifier/Modifier Character	%A, g, E	1
Environment Variable	<i>PATH</i>	
Error Number	[EINTR]	
Example Output	Hello, World	
Filename	/tmp	
Literal Character	'c', '\r'	2
Literal String	"abcde"	2
Optional Items in Utility Syntax	[]	
Parameter	<directory pathname>	
Special Character	<newline>	3
Symbolic Constant	_POSIX_VDISABLE	
Symbolic Limit, Configuration Value	{LINE_MAX}	4
Syntax	#include <sys/stat.h>	
User Input and Example Code	echo Hello, World	5
Utility Name	<i>awk</i>	
Utility Operand	<i>file_name</i>	
Utility Option	-c	
Utility Option with Option-Argument	-w width	

Note that:

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. The literal characters <apostrophe> (also known as single-quote) and <backslash> are either shown as the C constants **' '** and **'\ '**, respectively, or as the special characters <apostrophe>, single-quote, and <backslash> depending on context.
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.

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:

1. (a,b) means the range of all values from a to b , including neither a nor b .
2. $[a,b]$ means the range of all values from a to b , including a and b .
3. $[a,b)$ means the range of all values from a to b , including a , but not b .
4. $(a,b]$ means the range of all values from a to b , including b , but not a .

Note: A symbolic limit beginning with POSIX is treated differently, depending on context. In a C-language header, the symbol `POSIXstring` (where *string* may contain underscores) is represented by the C identifier `_POSIXstring`, with a leading underscore required to prevent ISO C standard name space pollution. However, in other contexts, such as languages other than C, the leading underscore is not used because this requirement does not exist.

Notice to Users

Laws and Regulations

Users of this document should consult all applicable laws and regulations. Compliance with the provisions of this standard does not imply compliance to any applicable regulatory requirements. Implementers of the standard are responsible for observing or referring to the applicable regulatory requirements. IEEE and The Open Group do not, by the publication of standards, intend to urge action that is not in compliance with applicable laws, and these documents may not be construed as doing so.

Copyrights

This document is copyrighted by the IEEE and The Open Group. It is made available for a wide variety of both public and private uses. These include both use, by reference, in laws and regulations, and use in private self-regulation, standardization, and the promotion of engineering practices and methods. By making this document available for use and adoption by public authorities and private users, the IEEE and The Open Group do not waive any rights in copyright to this document.

Updating of IEEE Documents

Users of IEEE standards should be aware that these documents may be superseded at any time by the issuance of new editions or may be amended from time to time through the issuance of amendments, corrigenda, or errata. An official IEEE document at any point in time consists of the current edition of the document together with any amendments, corrigenda, or errata then in effect. In order to determine whether a given document is the current edition and whether it has been amended through the issuance of amendments, corrigenda, or errata, visit the IEEE Standards Association web site at ieeexplore.ieee.org/xpl/standards.jsp, or contact the IEEE at the address listed previously.

For more information about the IEEE Standards Association or the IEEE standards development process, visit the IEEE-SA web site at standards.ieee.org.

Errata

Errata, if any, for this and all other standards can be accessed at the following web site: standards.ieee.org/reading/ieee/updates/errata. Users are encouraged to check this URL for errata periodically.

Feedback

POSIX.1-2008 has been prepared by the Austin Group. Feedback relating to the material contained in POSIX.1-2008 may be submitted using the Austin Group web site at www.opengroup.org/austin/defectform.html.

Interpretations

Current interpretations can be accessed at the following web site: standards.ieee.org/reading/ieee/interp.

Patents

Attention is called to the possibility that implementation of this standard may require use of subject matter covered by patent rights. By publication of this standard, no position is taken with respect to the existence or validity of any patent rights in connection therewith. A patent holder or patent applicant has filed a statement of assurance that it will grant licenses under these rights without compensation or under reasonable rates, with reasonable terms and conditions that are demonstrably free of any unfair discrimination to applicants desiring to obtain such licenses. Other Essential Patent Claims may exist for which a statement of assurance has not been received. The IEEE and The Open Group are not responsible for identifying Essential Patent Claims for which a license may be required, for conducting inquiries into the legal validity or scope of Patents Claims, or determining whether any licensing terms or conditions are reasonable or non-discriminatory. Further information may be obtained from the IEEE Standards Association and The Open Group.

Contents

Volume	1	Base Definitions, Issue 7.....	1
Chapter	1	Introduction.....	3
	1.1	Scope	3
	1.2	Conformance.....	4
	1.3	Normative References	4
	1.4	Change History	5
	1.5	Terminology	5
	1.6	Definitions and Concepts.....	6
	1.7	Portability	6
	1.7.1	Codes	7
	1.7.2	Margin Code Notation	13
Chapter	2	Conformance.....	15
	2.1	Implementation Conformance	15
	2.1.1	Requirements	15
	2.1.2	Documentation	16
	2.1.3	POSIX Conformance.....	16
	2.1.4	XSI Conformance	19
	2.1.5	Option Groups.....	20
	2.1.6	Options	26
	2.2	Application Conformance.....	29
	2.2.1	Strictly Conforming POSIX Application.....	29
	2.2.2	Conforming POSIX Application	30
	2.2.3	Conforming POSIX Application Using Extensions.....	30
	2.2.4	Strictly Conforming XSI Application	30
	2.2.5	Conforming XSI Application Using Extensions	31
	2.3	Language-Dependent Services for the C Programming Language	31
	2.4	Other Language-Related Specifications.....	31
Chapter	3	Definitions.....	33
	3.1	Abortive Release.....	33
	3.2	Absolute Pathname.....	33
	3.3	Access Mode	33
	3.4	Additional File Access Control Mechanism.....	33
	3.5	Address Space.....	33
	3.6	Advisory Information.....	34
	3.7	Affirmative Response	34
	3.8	Alert	34
	3.9	Alert Character (<alert>).....	34
	3.10	Alias Name.....	34
	3.11	Alignment	35
	3.12	Alternate File Access Control Mechanism	35

Contents

3.13	Alternate Signal Stack.....	35
3.14	Ancillary Data.....	35
3.15	Angle Brackets.....	35
3.16	Apostrophe Character (<apostrophe>).....	35
3.17	Application.....	35
3.18	Application Address.....	36
3.19	Application Program Interface (API).....	36
3.20	Appropriate Privileges.....	36
3.21	Argument.....	36
3.22	Arm (a Timer).....	36
3.23	Asterisk Character (<asterisk>).....	36
3.24	Async-Cancel-Safe Function.....	36
3.25	Asynchronous Events.....	37
3.26	Asynchronous Input and Output.....	37
3.27	Async-Signal-Safe Function.....	37
3.28	Asynchronously-Generated Signal.....	37
3.29	Asynchronous I/O Completion.....	37
3.30	Asynchronous I/O Operation.....	37
3.31	Authentication.....	37
3.32	Authorization.....	38
3.33	Background Job.....	38
3.34	Background Process.....	38
3.35	Background Process Group (or Background Job).....	38
3.36	Backquote Character.....	38
3.37	Backslash Character (<backslash>).....	38
3.38	Backspace Character (<backspace>).....	38
3.39	Barrier.....	39
3.40	Basename.....	39
3.41	Basic Regular Expression (BRE).....	39
3.42	Batch Access List.....	39
3.43	Batch Administrator.....	39
3.44	Batch Client.....	39
3.45	Batch Destination.....	40
3.46	Batch Destination Identifier.....	40
3.47	Batch Directive.....	40
3.48	Batch Job.....	40
3.49	Batch Job Attribute.....	40
3.50	Batch Job Identifier.....	40
3.51	Batch Job Name.....	41
3.52	Batch Job Owner.....	41
3.53	Batch Job Priority.....	41
3.54	Batch Job State.....	41
3.55	Batch Name Service.....	41
3.56	Batch Name Space.....	41
3.57	Batch Node.....	42
3.58	Batch Operator.....	42
3.59	Batch Queue.....	42
3.60	Batch Queue Attribute.....	42
3.61	Batch Queue Position.....	42
3.62	Batch Queue Priority.....	42
3.63	Batch Rerunability.....	43
3.64	Batch Restart.....	43

Contents

3.65	Batch Server	43
3.66	Batch Server Name.....	43
3.67	Batch Service	43
3.68	Batch Service Request.....	43
3.69	Batch Submission	43
3.70	Batch System	44
3.71	Batch Target User	44
3.72	Batch User	44
3.73	Bind	44
3.74	Blank Character (<blank>).....	44
3.75	Blank Line.....	44
3.76	Blocked Process (or Thread)	44
3.77	Blocking	44
3.78	Block-Mode Terminal	45
3.79	Block Special File.....	45
3.80	Braces	45
3.81	Brackets.....	45
3.82	Broadcast	45
3.83	Built-In Utility (or Built-In).....	46
3.84	Byte.....	46
3.85	Byte Input/Output Functions	46
3.86	Carriage-Return Character (<carriage-return>)	46
3.87	Character	47
3.88	Character Array.....	47
3.89	Character Class.....	47
3.90	Character Set.....	47
3.91	Character Special File	47
3.92	Character String.....	47
3.93	Child Process	48
3.94	Circumflex Character (<circumflex>).....	48
3.95	Clock	48
3.96	Clock Jump.....	48
3.97	Clock Tick.....	48
3.98	Coded Character Set	48
3.99	Codeset	49
3.100	Collating Element.....	49
3.101	Collation	49
3.102	Collation Sequence.....	49
3.103	Column Position.....	50
3.104	Command.....	50
3.105	Command Language Interpreter	50
3.106	Composite Graphic Symbol.....	50
3.107	Condition Variable	50
3.108	Connected Socket.....	51
3.109	Connection	51
3.110	Connection Mode.....	51
3.111	Connectionless Mode	51
3.112	Control Character.....	51
3.113	Control Operator.....	51
3.114	Controlling Process	51
3.115	Controlling Terminal	52
3.116	Conversion Descriptor	52

Contents

3.117	Core File.....	52
3.118	CPU Time (Execution Time)	52
3.119	CPU-Time Clock.....	52
3.120	CPU-Time Timer.....	52
3.121	Current Job	52
3.122	Current Working Directory.....	53
3.123	Cursor Position.....	53
3.124	Datagram.....	53
3.125	Data Segment.....	53
3.126	Deferred Batch Service	53
3.127	Device	53
3.128	Device ID.....	53
3.129	Directory.....	53
3.130	Directory Entry (or Link)	53
3.131	Directory Stream	54
3.132	Disarm (a Timer)	54
3.133	Display	54
3.134	Display Line	54
3.135	Dollar-Sign Character (<dollar-sign>)	54
3.136	Dot.....	54
3.137	Dot-Dot.....	55
3.138	Double-Quote Character.....	55
3.139	Downshifting.....	55
3.140	Driver	55
3.141	Effective Group ID	55
3.142	Effective User ID	55
3.143	Eight-Bit Transparency	55
3.144	Empty Directory	56
3.145	Empty Line.....	56
3.146	Empty String (or Null String).....	56
3.147	Empty Wide-Character String.....	56
3.148	Encoding Rule	56
3.149	Entire Regular Expression.....	56
3.150	Epoch	57
3.151	Equivalence Class.....	57
3.152	Era.....	57
3.153	Event Management	57
3.154	Executable File.....	57
3.155	Execute.....	58
3.156	Execution Time	58
3.157	Execution Time Monitoring.....	58
3.158	Expand.....	58
3.159	Extended Regular Expression (ERE)	58
3.160	Extended Security Controls	58
3.161	Feature Test Macro	59
3.162	Field.....	59
3.163	FIFO Special File (or FIFO)	59
3.164	File	59
3.165	File Description	59
3.166	File Descriptor	60
3.167	File Group Class	60
3.168	File Mode.....	60

Contents

3.169	File Mode Bits	60
3.170	Filename	60
3.171	File Offset	60
3.172	File Other Class	61
3.173	File Owner Class	61
3.174	File Permission Bits	61
3.175	File Serial Number	61
3.176	File System	61
3.177	File Type	61
3.178	Filter	62
3.179	First Open (of a File)	62
3.180	Flow Control	62
3.181	Foreground Job	62
3.182	Foreground Process	62
3.183	Foreground Process Group (or Foreground Job)	62
3.184	Foreground Process Group ID	62
3.185	Form-Feed Character (<form-feed>)	63
3.186	Graphic Character	63
3.187	Group Database	63
3.188	Group ID	63
3.189	Group Name	63
3.190	Hard Limit	63
3.191	Hard Link	64
3.192	Home Directory	64
3.193	Host Byte Order	64
3.194	Incomplete Line	64
3.195	Inf	64
3.196	Instrumented Application	64
3.197	Interactive Shell	64
3.198	Internationalization	65
3.199	Interprocess Communication	65
3.200	Invoke	65
3.201	Job	65
3.202	Job Control	65
3.203	Job Control Job ID	65
3.204	Last Close (of a File)	66
3.205	Line	66
3.206	Linger	66
3.207	Link	66
3.208	Link Count	66
3.209	Local Customs	66
3.210	Local Interprocess Communication (Local IPC)	66
3.211	Locale	67
3.212	Localization	67
3.213	Login	67
3.214	Login Name	67
3.215	Map	67
3.216	Marked Message	67
3.217	Matched	68
3.218	Memory Mapped Files	68
3.219	Memory Object	68
3.220	Memory-Resident	68

Contents

3.221	Message	68
3.222	Message Catalog	68
3.223	Message Catalog Descriptor	69
3.224	Message Queue	69
3.225	Mode	69
3.226	Monotonic Clock	69
3.227	Mount Point	69
3.228	Multi-Character Collating Element	69
3.229	Mutex	69
3.230	Name	70
3.231	Named STREAM	70
3.232	NaN (Not a Number)	70
3.233	Native Language	70
3.234	Negative Response	70
3.235	Network	70
3.236	Network Address	70
3.237	Network Byte Order	71
3.238	Newline Character (<newline>)	71
3.239	Nice Value	71
3.240	Non-Blocking	71
3.241	Non-Spacing Characters	71
3.242	NUL	72
3.243	Null Byte	72
3.244	Null Pointer	72
3.245	Null String	72
3.246	Null Wide-Character Code	72
3.247	Number-Sign Character (<number-sign>)	72
3.248	Object File	72
3.249	Octet	72
3.250	Offset Maximum	73
3.251	Opaque Address	73
3.252	Open File	73
3.253	Open File Description	73
3.254	Operand	73
3.255	Operator	73
3.256	Option	74
3.257	Option-Argument	74
3.258	Orientation	74
3.259	Orphaned Process Group	74
3.260	Page	74
3.261	Page Size	74
3.262	Parameter	75
3.263	Parent Directory	75
3.264	Parent Process	75
3.265	Parent Process ID	75
3.266	Pathname	75
3.267	Pathname Component	76
3.268	Path Prefix	76
3.269	Pattern	76
3.270	Period Character (<period>)	76
3.271	Permissions	76
3.272	Persistence	76

Contents

3.273	Pipe.....	77
3.274	Polling.....	77
3.275	Portable Character Set.....	77
3.276	Portable Filename Character Set.....	77
3.277	Positional Parameter.....	78
3.278	Preallocation.....	78
3.279	Preempted Process (or Thread).....	78
3.280	Previous Job.....	78
3.281	Printable Character.....	78
3.282	Printable File.....	78
3.283	Priority.....	79
3.284	Priority Band.....	79
3.285	Priority Inversion.....	79
3.286	Priority Scheduling.....	79
3.287	Priority-Based Scheduling.....	79
3.288	Privilege.....	79
3.289	Process.....	80
3.290	Process Group.....	80
3.291	Process Group ID.....	80
3.292	Process Group Leader.....	80
3.293	Process Group Lifetime.....	80
3.294	Process ID.....	81
3.295	Process Lifetime.....	81
3.296	Process Memory Locking.....	81
3.297	Process Termination.....	81
3.298	Process-To-Process Communication.....	81
3.299	Process Virtual Time.....	82
3.300	Program.....	82
3.301	Protocol.....	82
3.302	Pseudo-Terminal.....	82
3.303	Radix Character.....	82
3.304	Read-Only File System.....	82
3.305	Read-Write Lock.....	82
3.306	Real Group ID.....	83
3.307	Real Time.....	83
3.308	Realtime Signal Extension.....	83
3.309	Real User ID.....	83
3.310	Record.....	83
3.311	Redirection.....	83
3.312	Redirection Operator.....	84
3.313	Referenced Shared Memory Object.....	84
3.314	Refresh.....	84
3.315	Regular Expression.....	84
3.316	Region.....	84
3.317	Regular File.....	84
3.318	Relative Pathname.....	85
3.319	Relocatable File.....	85
3.320	Relocation.....	85
3.321	Requested Batch Service.....	85
3.322	(Time) Resolution.....	85
3.323	Robust Mutex.....	85
3.324	Root Directory.....	85

Contents

3.325	Runnable Process (or Thread)	85
3.326	Running Process (or Thread)	86
3.327	Saved Resource Limits	86
3.328	Saved Set-Group-ID	86
3.329	Saved Set-User-ID	86
3.330	Scheduling	86
3.331	Scheduling Allocation Domain	86
3.332	Scheduling Contention Scope	86
3.333	Scheduling Policy	87
3.334	Screen	87
3.335	Scroll	87
3.336	Semaphore	87
3.337	Session	88
3.338	Session Leader	88
3.339	Session Lifetime	88
3.340	Shared Memory Object	88
3.341	Shell	88
3.342	Shell, the	88
3.343	Shell Script	89
3.344	Signal	89
3.345	Signal Stack	89
3.346	Single-Quote Character	89
3.347	Slash Character (<slash>)	89
3.348	Socket	89
3.349	Socket Address	89
3.350	Soft Limit	90
3.351	Source Code	90
3.352	Space Character (<space>)	90
3.353	Spawn	90
3.354	Special Built-In	90
3.355	Special Parameter	91
3.356	Spin Lock	91
3.357	Sporadic Server	91
3.358	Standard Error	91
3.359	Standard Input	91
3.360	Standard Output	91
3.361	Standard Utilities	91
3.362	Stream	92
3.363	STREAM	92
3.364	STREAM End	92
3.365	STREAM Head	92
3.366	STREAMS Multiplexor	92
3.367	String	92
3.368	Subshell	93
3.369	Successfully Transferred	93
3.370	Supplementary Group ID	93
3.371	Suspended Job	93
3.372	Symbolic Constant	93
3.373	Symbolic Link	94
3.374	Synchronized Input and Output	94
3.375	Synchronized I/O Completion	94
3.376	Synchronized I/O Data Integrity Completion	94

Contents

3.377	Synchronized I/O File Integrity Completion	94
3.378	Synchronized I/O Operation	94
3.379	Synchronous I/O Operation.....	95
3.380	Synchronously-Generated Signal	95
3.381	System.....	95
3.382	System Boot.....	95
3.383	System Clock.....	95
3.384	System Console	95
3.385	System Crash	95
3.386	System Databases.....	96
3.387	System Documentation	96
3.388	System Process.....	96
3.389	System Reboot	96
3.390	System Trace Event	96
3.391	System-Wide	96
3.392	Tab Character (<tab>).....	97
3.393	Terminal (or Terminal Device)	97
3.394	Text Column.....	97
3.395	Text File.....	97
3.396	Thread.....	97
3.397	Thread ID	97
3.398	Thread List	98
3.399	Thread-Safe	98
3.400	Thread-Specific Data Key.....	98
3.401	Tilde Character (<tilde>).....	98
3.402	Timeouts	98
3.403	Timer	98
3.404	Timer Overrun.....	98
3.405	Token.....	99
3.406	Trace Analyzer Process.....	99
3.407	Trace Controller Process.....	99
3.408	Trace Event.....	99
3.409	Trace Event Type	99
3.410	Trace Event Type Mapping.....	99
3.411	Trace Filter.....	99
3.412	Trace Generation Version	99
3.413	Trace Log	100
3.414	Trace Point.....	100
3.415	Trace Stream	100
3.416	Trace Stream Identifier	100
3.417	Trace System	100
3.418	Traced Process.....	100
3.419	Tracing Status of a Trace Stream	100
3.420	Typed Memory Name Space	100
3.421	Typed Memory Object.....	101
3.422	Typed Memory Pool	101
3.423	Typed Memory Port.....	101
3.424	Unbind	101
3.425	Unit Data	101
3.426	Upshifting	101
3.427	User Database	101
3.428	User ID.....	102

Contents

3.429	User Name	102
3.430	User Trace Event.....	102
3.431	Utility	102
3.432	Variable.....	103
3.433	Vertical-Tab Character (<vertical-tab>).....	103
3.434	White Space.....	103
3.435	Wide-Character Code (C Language)	103
3.436	Wide-Character Input/Output Functions	103
3.437	Wide-Character String.....	103
3.438	Word.....	104
3.439	Working Directory (or Current Working Directory)	104
3.440	Worldwide Portability Interface	104
3.441	Write.....	104
3.442	XSI	104
3.443	XSI-Conformant	105
3.444	Zombie Process.....	105
3.445	±0	105
Chapter 4	General Concepts	107
4.1	Concurrent Execution.....	107
4.2	Directory Protection.....	107
4.3	Extended Security Controls	107
4.4	File Access Permissions.....	108
4.5	File Hierarchy	108
4.6	Filenames.....	109
4.7	Filename Portability.....	109
4.8	File Times Update	109
4.9	Host and Network Byte Orders	110
4.10	Measurement of Execution Time	110
4.11	Memory Synchronization	110
4.12	Pathname Resolution.....	111
4.13	Process ID Reuse	112
4.14	Scheduling Policy.....	112
4.15	Seconds Since the Epoch	113
4.16	Semaphore.....	113
4.17	Thread-Safety.....	114
4.18	Tracing	114
4.19	Treatment of Error Conditions for Mathematical Functions	116
4.19.1	Domain Error	116
4.19.2	Pole Error.....	117
4.19.3	Range Error	117
4.20	Treatment of NaN Arguments for the Mathematical Functions	118
4.21	Utility	118
4.22	Variable Assignment.....	118
Chapter 5	File Format Notation.....	121
Chapter 6	Character Set	125
6.1	Portable Character Set	125
6.2	Character Encoding	128

Contents

6.3	C Language Wide-Character Codes	129
6.4	Character Set Description File	129
6.4.1	State-Dependent Character Encodings	132
Chapter 7	Locale	135
7.1	General	135
7.2	POSIX Locale	136
7.3	Locale Definition	136
7.3.1	LC_CTYPE	139
7.3.2	LC_COLLATE	146
7.3.3	LC_MONETARY	154
7.3.4	LC_NUMERIC	157
7.3.5	LC_TIME	158
7.3.6	LC_MESSAGES	164
7.4	Locale Definition Grammar	165
7.4.1	Locale Lexical Conventions	165
7.4.2	Locale Grammar	166
Chapter 8	Environment Variables	173
8.1	Environment Variable Definition	173
8.2	Internationalization Variables	174
8.3	Other Environment Variables	177
Chapter 9	Regular Expressions	181
9.1	Regular Expression Definitions	181
9.2	Regular Expression General Requirements	182
9.3	Basic Regular Expressions	183
9.3.1	BREs Matching a Single Character or Collating Element	183
9.3.2	BRE Ordinary Characters	183
9.3.3	BRE Special Characters	183
9.3.4	Periods in BREs	184
9.3.5	RE Bracket Expression	184
9.3.6	BREs Matching Multiple Characters	186
9.3.7	BRE Precedence	187
9.3.8	BRE Expression Anchoring	187
9.4	Extended Regular Expressions	188
9.4.1	EREs Matching a Single Character or Collating Element	188
9.4.2	ERE Ordinary Characters	188
9.4.3	ERE Special Characters	188
9.4.4	Periods in EREs	189
9.4.5	ERE Bracket Expression	189
9.4.6	EREs Matching Multiple Characters	189
9.4.7	ERE Alternation	190
9.4.8	ERE Precedence	190
9.4.9	ERE Expression Anchoring	190
9.5	Regular Expression Grammar	191
9.5.1	BRE/ERE Grammar Lexical Conventions	191
9.5.2	RE and Bracket Expression Grammar	192
9.5.3	ERE Grammar	194

Chapter	10	Directory Structure and Devices	197
	10.1	Directory Structure and Files.....	197
	10.2	Output Devices and Terminal Types.....	198
Chapter	11	General Terminal Interface	199
	11.1	Interface Characteristics	199
	11.1.1	Opening a Terminal Device File.....	199
	11.1.2	Process Groups	200
	11.1.3	The Controlling Terminal.....	200
	11.1.4	Terminal Access Control	201
	11.1.5	Input Processing and Reading Data	201
	11.1.6	Canonical Mode Input Processing.....	202
	11.1.7	Non-Canonical Mode Input Processing.....	202
	11.1.8	Writing Data and Output Processing	203
	11.1.9	Special Characters	203
	11.1.10	Modem Disconnect	205
	11.1.11	Closing a Terminal Device File.....	205
	11.2	Parameters that Can be Set	205
	11.2.1	The termios Structure	205
	11.2.2	Input Modes.....	206
	11.2.3	Output Modes.....	207
	11.2.4	Control Modes	209
	11.2.5	Local Modes	210
	11.2.6	Special Control Characters.....	212
Chapter	12	Utility Conventions.....	213
	12.1	Utility Argument Syntax.....	213
	12.2	Utility Syntax Guidelines.....	215
Chapter	13	Headers	219
Volume	2	System Interfaces, Issue 7.....	463
Chapter	1	Introduction.....	465
	1.1	Relationship to Other Formal Standards	465
	1.2	Format of Entries.....	465
Chapter	2	General Information	467
	2.1	Use and Implementation of Interfaces	467
	2.1.1	Use and Implementation of Functions.....	467
	2.1.2	Use and Implementation of Macros	468
	2.2	The Compilation Environment	468
	2.2.1	POSIX.1 Symbols.....	468
	2.2.2	The Name Space	469
	2.3	Error Numbers.....	477
	2.3.1	Additional Error Numbers	484
	2.4	Signal Concepts	484
	2.4.1	Signal Generation and Delivery.....	484
	2.4.2	Realtime Signal Generation and Delivery	485
	2.4.3	Signal Actions	486
	2.4.4	Signal Effects on Other Functions.....	490

Contents

2.5	Standard I/O Streams	490
2.5.1	Interaction of File Descriptors and Standard I/O Streams	491
2.5.2	Stream Orientation and Encoding Rules	493
2.6	STREAMS	494
2.6.1	Accessing STREAMS	495
2.7	XSI Interprocess Communication	496
2.7.1	IPC General Description	496
2.8	Realtime	497
2.8.1	Realtime Signals	497
2.8.2	Asynchronous I/O	497
2.8.3	Memory Management	499
2.8.4	Process Scheduling	501
2.8.5	Clocks and Timers	505
2.9	Threads	507
2.9.1	Thread-Safety	507
2.9.2	Thread IDs	508
2.9.3	Thread Mutexes	508
2.9.4	Thread Scheduling	509
2.9.5	Thread Cancellation	511
2.9.6	Thread Read-Write Locks	515
2.9.7	Thread Interactions with Regular File Operations	516
2.9.8	Use of Application-Managed Thread Stacks	516
2.10	Sockets	517
2.10.1	Address Families	517
2.10.2	Addressing	517
2.10.3	Protocols	517
2.10.4	Routing	518
2.10.5	Interfaces	518
2.10.6	Socket Types	518
2.10.7	Socket I/O Mode	519
2.10.8	Socket Owner	519
2.10.9	Socket Queue Limits	519
2.10.10	Pending Error	519
2.10.11	Socket Receive Queue	520
2.10.12	Socket Out-of-Band Data State	520
2.10.13	Connection Indication Queue	521
2.10.14	Signals	521
2.10.15	Asynchronous Errors	521
2.10.16	Use of Options	522
2.10.17	Use of Sockets for Local UNIX Connections	525
2.10.18	Use of Sockets over Internet Protocols	525
2.10.19	Use of Sockets over Internet Protocols Based on IPv4	526
2.10.20	Use of Sockets over Internet Protocols Based on IPv6	526
2.11	Tracing	529
2.11.1	Tracing Data Definitions	531
2.11.2	Trace Event Type Definitions	535
2.11.3	Trace Functions	539
2.12	Data Types	540
2.12.1	Defined Types	540

Contents

	2.12.2	The char Type.....	541
	2.12.3	Pointer Types	541
Chapter	3	System Interfaces.....	543
Volume	3	Shell and Utilities, Issue 7	2277
Chapter	1	Introduction.....	2279
	1.1	Relationship to Other Documents	2279
	1.1.1	System Interfaces.....	2279
	1.1.2	Concepts Derived from the ISO C Standard	2283
	1.2	Utility Limits.....	2285
	1.3	Grammar Conventions.....	2287
	1.4	Utility Description Defaults.....	2288
	1.5	Considerations for Utilities in Support of Files of Arbitrary Size.....	2295
	1.6	Built-In Utilities	2296
Chapter	2	Shell Command Language	2297
	2.1	Shell Introduction.....	2297
	2.2	Quoting.....	2298
	2.2.1	Escape Character (Backslash)	2298
	2.2.2	Single-Quotes.....	2298
	2.2.3	Double-Quotes.....	2298
	2.3	Token Recognition.....	2299
	2.3.1	Alias Substitution	2300
	2.4	Reserved Words.....	2301
	2.5	Parameters and Variables.....	2301
	2.5.1	Positional Parameters	2301
	2.5.2	Special Parameters	2302
	2.5.3	Shell Variables.....	2302
	2.6	Word Expansions	2305
	2.6.1	Tilde Expansion	2305
	2.6.2	Parameter Expansion.....	2306
	2.6.3	Command Substitution	2309
	2.6.4	Arithmetic Expansion.....	2310
	2.6.5	Field Splitting	2311
	2.6.6	Pathname Expansion	2311
	2.6.7	Quote Removal.....	2311
	2.7	Redirection	2312
	2.7.1	Redirecting Input	2312
	2.7.2	Redirecting Output	2313
	2.7.3	Appending Redirected Output	2313
	2.7.4	Here-Document.....	2313
	2.7.5	Duplicating an Input File Descriptor	2314
	2.7.6	Duplicating an Output File Descriptor	2314
	2.7.7	Open File Descriptors for Reading and Writing.....	2315
	2.8	Exit Status and Errors.....	2315
	2.8.1	Consequences of Shell Errors	2315
	2.8.2	Exit Status for Commands	2315
	2.9	Shell Commands	2316

Contents

	2.9.1	Simple Commands.....	2316
	2.9.2	Pipelines	2318
	2.9.3	Lists	2319
	2.9.4	Compound Commands.....	2321
	2.9.5	Function Definition Command	2324
	2.10	Shell Grammar.....	2325
	2.10.1	Shell Grammar Lexical Conventions.....	2325
	2.10.2	Shell Grammar Rules.....	2325
	2.11	Signals and Error Handling.....	2330
	2.12	Shell Execution Environment.....	2331
	2.13	Pattern Matching Notation	2332
	2.13.1	Patterns Matching a Single Character.....	2332
	2.13.2	Patterns Matching Multiple Characters.....	2332
	2.13.3	Patterns Used for Filename Expansion.....	2333
	2.14	Special Built-In Utilities.....	2334
Chapter	3	Batch Environment Services	2375
	3.1	General Concepts	2375
	3.1.1	Batch Client-Server Interaction	2375
	3.1.2	Batch Queues	2376
	3.1.3	Batch Job Creation.....	2376
	3.1.4	Batch Job Tracking.....	2376
	3.1.5	Batch Job Routing.....	2377
	3.1.6	Batch Job Execution	2377
	3.1.7	Batch Job Exit	2378
	3.1.8	Batch Job Abort.....	2378
	3.1.9	Batch Authorization.....	2378
	3.1.10	Batch Administration	2378
	3.1.11	Batch Notification.....	2379
	3.2	Batch Services	2379
	3.2.1	Batch Job States.....	2380
	3.2.2	Deferred Batch Services.....	2381
	3.2.3	Requested Batch Services.....	2390
	3.3	Common Behavior for Batch Environment Utilities.....	2397
	3.3.1	Batch Job Identifier.....	2397
	3.3.2	Destination	2398
	3.3.3	Multiple Keyword-Value Pairs.....	2399
Chapter	4	Utilities.....	2401
Volume	4	Rationale (Informative), Issue 7.....	3407
Part	A	Base Definitions	3409
Appendix	A	Rationale for Base Definitions.....	3411
	A.1	Introduction	3411
	A.1.1	Scope	3411
	A.1.2	Conformance.....	3414
	A.1.3	Normative References	3414
	A.1.4	Change History	3414
	A.1.5	Terminology	3414

Contents

A.1.6	Definitions and Concepts.....	3416
A.1.7	Portability.....	3416
A.2	Conformance.....	3417
A.2.1	Implementation Conformance	3417
A.2.2	Application Conformance.....	3421
A.2.3	Language-Dependent Services for the C Programming Language.....	3421
A.2.4	Other Language-Related Specifications.....	3422
A.3	Definitions.....	3422
A.4	General Concepts	3443
A.4.1	Concurrent Execution.....	3443
A.4.2	Directory Protection.....	3444
A.4.3	Extended Security Controls	3444
A.4.4	File Access Permissions.....	3444
A.4.5	File Hierarchy	3444
A.4.6	Filenames.....	3445
A.4.7	Filename Portability.....	3446
A.4.8	File Times Update	3446
A.4.9	Host and Network Byte Order	3447
A.4.10	Measurement of Execution Time	3447
A.4.11	Memory Synchronization	3447
A.4.12	Pathname Resolution.....	3449
A.4.13	Process ID Reuse	3450
A.4.14	Scheduling Policy	3450
A.4.15	Seconds Since the Epoch	3450
A.4.16	Semaphore.....	3452
A.4.17	Thread-Safety	3452
A.4.18	Tracing	3452
A.4.19	Treatment of Error Conditions for Mathematical Functions	3452
A.4.20	Treatment of NaN Arguments for Mathematical Functions	3452
A.4.21	Utility	3452
A.4.22	Variable Assignment.....	3452
A.5	File Format Notation	3452
A.6	Character Set.....	3453
A.6.1	Portable Character Set	3453
A.6.2	Character Encoding	3454
A.6.3	C Language Wide-Character Codes	3454
A.6.4	Character Set Description File.....	3454
A.7	Locale	3456
A.7.1	General.....	3456
A.7.2	POSIX Locale	3457
A.7.3	Locale Definition	3457
A.7.4	Locale Definition Grammar	3464
A.7.5	Locale Definition Example.....	3464
A.8	Environment Variables.....	3467
A.8.1	Environment Variable Definition.....	3467
A.8.2	Internationalization Variables	3468
A.8.3	Other Environment Variables.....	3469
A.9	Regular Expressions	3470
A.9.1	Regular Expression Definitions.....	3471

Contents

A.9.2	Regular Expression General Requirements.....	3471
A.9.3	Basic Regular Expressions	3472
A.9.4	Extended Regular Expressions.....	3475
A.9.5	Regular Expression Grammar.....	3477
A.10	Directory Structure and Devices	3478
A.10.1	Directory Structure and Files.....	3478
A.10.2	Output Devices and Terminal Types.....	3478
A.11	General Terminal Interface	3478
A.11.1	Interface Characteristics.....	3479
A.11.2	Parameters that Can be Set.....	3483
A.12	Utility Conventions.....	3485
A.12.1	Utility Argument Syntax.....	3485
A.12.2	Utility Syntax Guidelines.....	3486
A.13	Headers.....	3488
A.13.1	Format of Entries.....	3488
A.13.2	Removed Headers in Issue 7	3489
Part B	System Interfaces.....	3491
Appendix B	Rationale for System Interfaces.....	3493
B.1	Introduction	3493
B.1.1	Change History	3493
B.1.2	Relationship to Other Formal Standards.....	3496
B.1.3	Format of Entries.....	3496
B.2	General Information	3497
B.2.1	Use and Implementation of Interfaces.....	3497
B.2.2	The Compilation Environment	3498
B.2.3	Error Numbers.....	3503
B.2.4	Signal Concepts	3507
B.2.5	Standard I/O Streams	3517
B.2.6	STREAMS.....	3517
B.2.7	XSI Interprocess Communication	3518
B.2.8	Realtime	3519
B.2.9	Threads	3564
B.2.10	Sockets	3592
B.2.11	Tracing	3594
B.2.12	Data Types.....	3620
B.3	System Interfaces.....	3622
B.3.1	System Interfaces Removed in this Version.....	3622
B.3.2	System Interfaces Removed in the Previous Version.....	3625
B.3.3	Examples for Spawn	3625
Part C	Shell and Utilities	3635
Appendix C	Rationale for Shell and Utilities.....	3637
C.1	Introduction	3637
C.1.1	Change History	3637
C.1.2	Relationship to Other Documents	3638
C.1.3	Utility Limits.....	3639
C.1.4	Grammar Conventions.....	3642
C.1.5	Utility Description Defaults.....	3642

Contents

C.1.6	Considerations for Utilities in Support of Files of Arbitrary Size	3645
C.1.7	Built-In Utilities	3646
C.2	Shell Command Language	3648
C.2.1	Shell Introduction	3648
C.2.2	Quoting	3648
C.2.3	Token Recognition	3650
C.2.4	Reserved Words	3651
C.2.5	Parameters and Variables	3651
C.2.6	Word Expansions	3654
C.2.7	Redirection	3660
C.2.8	Exit Status and Errors	3662
C.2.9	Shell Commands	3662
C.2.10	Shell Grammar	3669
C.2.11	Signals and Error Handling	3671
C.2.12	Shell Execution Environment	3671
C.2.13	Pattern Matching Notation	3671
C.2.14	Special Built-In Utilities	3673
C.3	Batch Environment Services and Utilities	3673
C.3.1	Batch General Concepts	3676
C.3.2	Batch Services	3678
C.3.3	Common Behavior for Batch Environment Utilities	3679
C.4	Utilities	3679
C.4.1	Utilities Removed in this Version	3679
C.4.2	Utilities Removed in the Previous Version	3679
C.4.3	Exclusion of Utilities	3679
Part	D Portability Considerations	3683
Appendix	D Portability Considerations (Informative)	3685
D.1	User Requirements	3685
D.1.1	Configuration Interrogation	3686
D.1.2	Process Management	3686
D.1.3	Access to Data	3686
D.1.4	Access to the Environment	3686
D.1.5	Access to Determinism and Performance Enhancements	3686
D.1.6	Operating System-Dependent Profile	3687
D.1.7	I/O Interaction	3687
D.1.8	Internationalization Interaction	3687
D.1.9	C-Language Extensions	3687
D.1.10	Command Language	3687
D.1.11	Interactive Facilities	3687
D.1.12	Accomplish Multiple Tasks Simultaneously	3687
D.1.13	Complex Data Manipulation	3688
D.1.14	File Hierarchy Manipulation	3688
D.1.15	Locale Configuration	3688
D.1.16	Inter-User Communication	3688
D.1.17	System Environment	3688
D.1.18	Printing	3688
D.1.19	Software Development	3688

Contents

D.2	Portability Capabilities.....	3689
D.2.1	Configuration Interrogation	3689
D.2.2	Process Management	3690
D.2.3	Access to Data.....	3690
D.2.4	Access to the Environment	3691
D.2.5	Bounded (Realtime) Response	3692
D.2.6	Operating System-Dependent Profile	3692
D.2.7	I/O Interaction	3692
D.2.8	Internationalization Interaction	3693
D.2.9	C-Language Extensions.....	3693
D.2.10	Command Language	3693
D.2.11	Interactive Facilities	3694
D.2.12	Accomplish Multiple Tasks Simultaneously	3694
D.2.13	Complex Data Manipulation	3694
D.2.14	File Hierarchy Manipulation	3695
D.2.15	Locale Configuration	3695
D.2.16	Inter-User Communication.....	3695
D.2.17	System Environment	3696
D.2.18	Printing.....	3696
D.2.19	Software Development.....	3696
D.2.20	Future Growth	3696
D.3	Profiling Considerations	3697
D.3.1	Configuration Options	3697
D.3.2	Configuration Options (Shell and Utilities)	3697
D.3.3	Configurable Limits.....	3699
D.3.4	Configuration Options (System Interfaces).....	3699
D.3.5	Configurable Limits.....	3704
D.3.6	Optional Behavior.....	3707
Part E	Subprofiling Considerations.....	3709
Appendix E	Subprofiling Considerations (Informative)	3711
E.1	Subprofiling Option Groups.....	3711
	Index	3717
List of Figures		
B-1	Example of a System with Typed Memory	3537
B-2	Trace System Overview: for Offline Analysis	3600
B-3	Trace System Overview: for Online Analysis	3601
B-4	Trace System Overview: States of a Trace Stream	3603
B-5	Trace Another Process	3613
B-6	Trace Name Space Overview: With Third-Party Library	3614
List of Tables		
3-1	Job Control Job ID Formats.....	66
5-1	Escape Sequences and Associated Actions	121
6-1	Portable Character Set	125
6-2	Control Character Set	130

Contents

7-1	Valid Character Class Combinations.....	142
10-1	Control Character Names	198
2-1	Value of Level for Socket Options.....	522
2-2	Socket-Level Options	523
2-3	Trace Option: System Trace Events.....	537
2-4	Trace and Trace Event Filter Options: System Trace Events.....	537
2-5	Trace and Trace Log Options: System Trace Events	538
2-6	Trace, Trace Log, and Trace Event Filter Options: System Trace Events	538
2-7	Trace Option: User Trace Event.....	539
1-1	Actions when Creating a File that Already Exists.....	2281
1-2	Selected ISO C Standard Operators and Control Flow Keywords	2284
1-3	Utility Limit Minimum Values.....	2285
1-4	Symbolic Utility Limits	2286
1-5	Regular Built-In Utilities	2296
3-1	Batch Utilities.....	2375
3-2	Environment Variable Summary	2379
3-3	Next State Table	2381
3-4	Results/Output Table	2383
3-5	Batch Services Summary	2390
A-1	Historical Practice for Symbolic Links	3440

Trademarks

The following information is given for the convenience of users of POSIX.1-2008 and does not constitute an endorsement by the IEEE or The Open Group of these products. Equivalent products may be used if they can be shown to lead to the same results.

There may be other products mentioned in the text that might be covered by trademark protection and readers are advised to verify them independently.

754TM, 854TM, 1003.0TM, 1003.1TM, 1003.1dTM, 1003.1gTM, 1003.1jTM, 1003.1qTM, 1003.2TM, 1003.2aTM, 1003.2dTM, 1003.9TM, and 1003.13TM are trademarks of the Institute of Electrical and Electronic Engineers, Inc.

AIX[®] is a registered trademark of IBM Corporation.

AT&T[®] is a registered trademark of AT&T in the USA and other countries.

Boundaryless Information FlowTM and TOGAFTM are trademarks and Motif[®], Making Standards Work[®], OSF/1[®], The Open Group[®], UNIX[®], and the “X” device are registered trademarks of The Open Group in the United States and other countries.

BSDTM is a trademark of the University of California, Berkeley, USA.

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

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

IEEE[®] is a registered trademark of the Institute of Electrical and Electronic Engineers, Inc.

Linux[®] is a registered trademark of Linus Torvalds.

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 POSIX.1-2008 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.
- Hewlett-Packard Company, International Business Machines Corporation, Novell Inc., The Open Software Foundation, and Sun Microsystems Inc. for permission to reproduce portions of their copyrighted documentation
- ISO/IEC JTC 1/SC 22/WG 14 C Language Committee
- Red Hat Inc. for permission to reproduce portions of its copyrighted documentation

POSIX.1-2008 was prepared by the Austin Group, a joint working group of the IEEE, The Open Group, and ISO/IEC JTC 1/SC 22.

Referenced Documents

Normative References

Normative references for POSIX.1-2008 are defined in [Section 1.3](#) (on page 4).

Informative References

The following documents are referenced in POSIX.1-2008:

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.

ANSI 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. Portable Operating System Interface (POSIX)—Part 1: System Application Program Interface (API) [C Language].

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 Standards Terms

IEEE 100, The Authoritative Dictionary of IEEE Standards Terms, Seventh Edition.

Referenced Documents

- IEEE Std 754TM-1985
IEEE Std 754-1985 (Reaff 1990), IEEE Standard for Binary Floating-Point Arithmetic.
- IEEE Std 854TM-1987
IEEE Std 854-1987, IEEE Standard for Radix-Independent Floating-Point Arithmetic.
- IEEE Std 1003.9TM-1992
IEEE Std 1003.9-1992, IEEE 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 (available at: www.ietf.org/rfc/rfc0791.txt).
- IETF RFC 819
The Domain Naming Convention for Internet User Applications, Z. Su, J. Postel, August 1982 (available at: www.ietf.org/rfc/rfc0819.txt).
- IETF RFC 822
Standard for the Format of ARPA Internet Text Messages, D.H. Crocker, August 1982 (available at: www.ietf.org/rfc/rfc0822.txt).
- IETF RFC 919
Broadcasting Internet Datagrams, J. Mogul, October 1984 (available at: www.ietf.org/rfc/rfc0919.txt).
- IETF RFC 920
Domain Requirements, J. Postel, J. Reynolds, October 1984 (available at: www.ietf.org/rfc/rfc0920.txt).
- IETF RFC 921
Domain Name System Implementation Schedule, J. Postel, October 1984 (available at: www.ietf.org/rfc/rfc0921.txt).
- IETF RFC 922
Broadcasting Internet Datagrams in the Presence of Subnets, J. Mogul, October 1984 (available at: www.ietf.org/rfc/rfc0922.txt).
- IETF RFC 1034
Domain Names — Concepts and Facilities, P. Mockapetris, November 1987 (available at: www.ietf.org/rfc/rfc1034.txt).
- IETF RFC 1035
Domain Names — Implementation and Specification, P. Mockapetris, November 1987 (available at: www.ietf.org/rfc/rfc1035.txt).
- IETF RFC 1123
Requirements for Internet Hosts — Application and Support, R. Braden, October 1989 (available at: www.ietf.org/rfc/rfc1123.txt).
- IETF RFC 1886
DNS Extensions to Support Internet Protocol, Version 6 (IPv6), C. Huitema, S. Thomson, December 1995 (available at: www.ietf.org/rfc/rfc1886.txt).
- IETF RFC 2045
Multipurpose Internet Mail Extensions (MIME), Part 1: Format of Internet Message Bodies, N. Freed, N. Borenstein, November 1996 (available at: www.ietf.org/rfc/rfc2045.txt).

Referenced Documents

IETF RFC 2181

Clarifications to the DNS Specification, R. Elz, R. Bush, July 1997 (available at: www.ietf.org/rfc/rfc2181.txt).

IETF RFC 2373

Internet Protocol, Version 6 (IPv6) Addressing Architecture, S. Deering, R. Hinden, July 1998 (available at: www.ietf.org/rfc/rfc2373.txt).

IETF RFC 2460

Internet Protocol, Version 6 (IPv6), S. Deering, R. Hinden, December 1998 (available at: www.ietf.org/rfc/rfc2460.txt).

Internationalisation Guide

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

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 1539:1991

ISO/IEC 1539:1991, 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 Graphic Character Set for Text Communication — Latin Alphabet.

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 11: Latin/Thai Alphabet

Referenced Documents

- Part 13: Latin Alphabet No. 7
- Part 14: Latin Alphabet No. 8 (Celtic)
- Part 15: Latin Alphabet No. 9
- Part 16: Latin Alphabet No. 10

ISO/IEC 9899: 1990

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

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 Std 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)

Referenced Documents

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)

Issue 6

Technical Standard, April 2004, published by The Open Group:

- Base Definitions (XBD), Issue 6 (ISBN: 1-931624-43-7, C046)
- System Interfaces (XSH), Issue 6 (ISBN: 1-931624-44-5, C047)
- Shell and Utilities (XCU), Issue 6 (ISBN: 1-931624-45-3, C048)

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, IEEE Guide to the POSIX Open System Environment (OSE) (identical to ISO/IEC TR 14252).

POSIX.1: 1988

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

POSIX.1a

P1003.1a, Standard for Information Technology — Portable Operating System Interface (POSIX) — Part 1: System Application Program Interface (API) — (C Language) Amendment.

POSIX.1d: 1999

IEEE Std 1003.1d™-1999, IEEE Standard for Information Technology — Portable Operating System Interface (POSIX) — Part 1: System Application Program Interface (API) —

Referenced Documents

Amendment 4: Additional Realtime Extensions [C Language].

POSIX.1g: 2000

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

POSIX.2b

P1003.2b, Standard for Information Technology — Portable Operating System Interface (POSIX) — Part 2: Shell and Utilities — Amendment.

POSIX.2d: 1994

IEEE Std 1003.2d™-1994, IEEE 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, IEEE 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.

Referenced Documents

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.

Source Documents

Parts of the following documents were used to create the base documents for POSIX.1-2008:

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).

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

Technical Standard: Base Specifications, Issue 7

Prepared by the Austin Group (www.opengroup.org/austin).

IMPORTANT NOTICE: This standard is not intended to assure safety, security, health, or environmental protection in all circumstances. Implementors of the standard are responsible for determining appropriate safety, security, environmental, and health practices or regulatory requirements.

This IEEE document is made available for use subject to important notices and legal disclaimers. These notices and disclaimers appear in all publications containing this document and may be found under the heading “Important Notice” or “Important Notices and Disclaimers Concerning IEEE Documents”. They can also be obtained on request from IEEE or viewed at <http://standards.ieee.org/IPR/disclaimers.html>.



Technical Standard

Vol. 1:

Base Definitions, Issue 7

The Open Group

The Institute of Electrical and Electronics Engineers, Inc.

(blank page)

Introduction

1.1 Scope

POSIX.1-2008 defines a standard operating system interface and environment, including a command interpreter (or “shell”), and common utility programs to support applications portability at the source code level. It is intended to be used by both application developers and system implementors.

POSIX.1-2008 comprises four major components (each in an associated volume):

1. General terms, concepts, and interfaces common to all volumes of POSIX.1-2008, including utility conventions and C-language header definitions, are included in the Base Definitions volume of POSIX.1-2008.
2. Definitions for system service functions and subroutines, language-specific system services for the C programming language, function issues, including portability, error handling, and error recovery, are included in the System Interfaces volume of POSIX.1-2008.
3. Definitions for a standard source code-level interface to command interpretation services (a “shell”) and common utility programs for application programs are included in the Shell and Utilities volume of POSIX.1-2008.
4. Extended rationale that did not fit well into the rest of the document structure, containing historical information concerning the contents of POSIX.1-2008 and why features were included or discarded by the standard developers, is included in the Rationale (Informative) volume of POSIX.1-2008.

The following areas are outside of the scope of POSIX.1-2008:

- Graphics interfaces
- Database management system interfaces
- Record I/O considerations
- Object or binary code portability
- System configuration and resource availability

POSIX.1-2008 describes the external characteristics and facilities that are of importance to application developers, rather than the internal construction techniques employed to achieve these capabilities. Special emphasis is placed on those functions and facilities that are needed in a wide variety of commercial applications.

The facilities provided in POSIX.1-2008 are drawn from the following base documents:

- IEEE Std 1003.1, 2004 Edition (POSIX-1) (incorporating IEEE Std 1003.1-2001, IEEE Std 1003.1-2001/Cor 1-2002, and IEEE Std 1003.1-2001/Cor 2-2004)

- The Open Group Technical Standard, 2006, Extended API Set Part 1
- The Open Group Technical Standard, 2006, Extended API Set Part 2
- The Open Group Technical Standard, 2006, Extended API Set Part 3
- The Open Group Technical Standard, 2006, Extended API Set Part 4
- ISO/IEC 9899:1999, Programming Languages — C, including ISO/IEC 9899:1999/Cor.1:2001(E), ISO/IEC 9899:1999/Cor.2:2004(E), and ISO/IEC 9899:1999/Cor.3

Emphasis has been placed on standardizing existing practice for existing users, with changes and additions limited to correcting deficiencies in the following areas:

- Issues raised by Austin Group defect reports, IEEE Interpretations against IEEE Std 1003.1, and ISO/IEC defect reports against ISO/IEC 9945
- Issues raised in corrigenda for The Open Group Technical Standards and working group resolutions from The Open Group
- Issues arising from ISO TR 24715:2006, Conflicts between POSIX and the LSB
- Changes to make the text self-consistent with the additional material merged
- Features, marked Legacy or obsolescent in the base documents, have been considered for removal in this version
- A review and reorganization of the options within the standard
- Alignment with the ISO/IEC 9899:1999 standard, including ISO/IEC 9899:1999/Cor.2:2004(E)

1.2 Conformance

Conformance requirements for POSIX.1-2008 are defined in [Chapter 2](#) (on page 15).

1.3 Normative References

The following standards contain provisions which, through references in POSIX.1-2008, constitute provisions of POSIX.1-2008. At the time of publication, the editions indicated were valid. All standards are subject to revision, and parties to agreements based on POSIX.1-2008 are encouraged to investigate the possibility of applying the most recent editions of the standards listed below. Members of IEC and ISO maintain registers of currently valid International Standards.

ANS X3.9-1978

(Reaffirmed 1989) American National Standard for Information Systems: Standard X3.9-1978, Programming Language FORTRAN.¹

ISO/IEC 646:1991

ISO/IEC 646:1991, Information Processing — ISO 7-Bit Coded Character Set for Information Interchange.²

1. ANSI documents can be obtained from the Sales Department, American National Standards Institute, 1430 Broadway, New York, NY 10018, USA.