

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

Prepared by the Austin Group
(<http://www.opengroup.org/austin/>)

Copyright © 2001 The Institute of Electrical and Electronics Engineers, Inc.
3 Park Avenue, New York, NY 10016-5997, USA.

Copyright © 2001 The Open Group
Apex Plaza, Forbury Road, Reading, Berkshire RG11AX, UK.

All rights reserved.

Except as permitted below, no part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, photocopying, recording or otherwise, without the prior permission of the copyright owners. This is an unapproved draft, subject to change. Permission is hereby granted for Austin Group participants to reproduce this document for purposes of IEEE, the Open Group, and JTC1 standardization activities. Other entities seeking permission to reproduce this document for standardization purposes or other activities must contact the copyright owners for an appropriate license. Use of information contained within this unapproved draft is at your own risk.

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

1 / *Technical Standard*

2 **System Interfaces, Issue 6**

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



5



6 © 2001, *The Open Group*

7 © 2001, *The Institute of Electrical and Electronics Engineers, Inc.*

8 All rights reserved.

9 Except as permitted below, no part of this publication may be reproduced, stored in a retrieval
10 system, or transmitted, in any form or by any means, electronic, mechanical, photocopying,
11 recording or otherwise, without the prior permission of the copyright owners.

This is an unapproved draft, subject to change. Permission is hereby granted for Austin Group participants to reproduce IEEE Std 1003.1-200x for purposes of IEEE, The Open Group, and JTC1 standardization activities.

Other entities seeking permission to reproduce IEEE Std 1003.1-200x for standardization purposes or activities must contact the copyright owners for an appropriate license. Use of information contained within this unapproved draft is at your own risk.

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

12 Technical Standard

13 System Interfaces, Issue 6

14 Document Number:

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

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

Contents

18

19	Chapter 1	Introduction.....	451
20	1.1	Scope.....	451
21	1.2	Conformance	451
22	1.3	Normative References	451
23	1.4	Change History	451
24	1.5	Terminology	451
25	1.6	Definitions.....	453
26	1.7	Relationship to Other Formal Standards	453
27	1.8	Portability	453
28	1.8.1	Codes.....	453
29	1.9	Format of Entries.....	461
30	Chapter 2	General Information	463
31	2.1	Use and Implementation of Functions	463
32	2.2	The Compilation Environment	463
33	2.2.1	POSIX.1 Symbols	463
34	2.2.1.1	The <code>_POSIX_C_SOURCE</code> Feature Test Macro.....	464
35	2.2.1.2	The <code>_XOPEN_SOURCE</code> Feature Test Macro.....	464
36	2.2.2	The Name Space.....	464
37	2.3	Error Numbers.....	471
38	2.3.1	Additional Error Numbers.....	478
39	2.4	Signal Concepts.....	478
40	2.4.1	Signal Generation and Delivery.....	478
41	2.4.2	Realtime Signal Generation and Delivery	479
42	2.4.3	Signal Actions	480
43	2.4.4	Signal Effects on Other Functions	484
44	2.5	Standard I/O Streams.....	484
45	2.5.1	Interaction of File Descriptors and Standard I/O Streams.....	485
46	2.5.2	Stream Orientation and Encoding Rules	486
47	2.6	STREAMS	488
48	2.6.1	Accessing STREAMS.....	489
49	2.7	XSI Interprocess Communication	489
50	2.7.1	IPC General Description.....	490
51	2.8	Realtime	491
52	2.8.1	Realtime Signals.....	491
53	2.8.2	Asynchronous I/O	491
54	2.8.3	Memory Management	493
55	2.8.3.1	Memory Locking.....	493
56	2.8.3.2	Memory Mapped Files.....	493
57	2.8.3.3	Memory Protection.....	494
58	2.8.3.4	Typed Memory Objects	494
59	2.8.4	Process Scheduling	494

60	2.8.5	Clocks and Timers	498
61	2.9	Threads.....	500
62	2.9.1	Thread-Safety.....	500
63	2.9.2	Thread IDs.....	501
64	2.9.3	Thread Mutexes.....	501
65	2.9.4	Thread Scheduling.....	502
66	2.9.5	Thread Cancellation	504
67	2.9.5.1	Cancelability States	504
68	2.9.5.2	Cancellation Points	505
69	2.9.5.3	Thread Cancellation Cleanup Handlers.....	507
70	2.9.5.4	Async-Cancel Safety.....	507
71	2.9.6	Thread Read-Write Locks.....	508
72	2.9.7	Thread Interactions with Regular File Operations	508
73	2.10	Sockets.....	508
74	2.10.1	Address Families.....	508
75	2.10.2	Addressing	509
76	2.10.3	Protocols	509
77	2.10.4	Routing.....	509
78	2.10.5	Interfaces.....	509
79	2.10.6	Socket Types.....	509
80	2.10.7	Socket I/O Mode.....	510
81	2.10.8	Socket Owner.....	510
82	2.10.9	Socket Queue Limits	510
83	2.10.10	Pending Error.....	510
84	2.10.11	Socket Receive Queue.....	511
85	2.10.12	Socket Out-of-Band Data State	511
86	2.10.13	Connection Indication Queue	512
87	2.10.14	Signals	512
88	2.10.15	Asynchronous Errors	512
89	2.10.16	Use of Options.....	513
90	2.10.17	Use of Sockets for Local UNIX Connections.....	516
91	2.10.17.1	Headers	516
92	2.10.18	Use of Sockets over Internet Protocols.....	516
93	2.10.19	Use of Sockets over Internet Protocols Based on IPv4.....	517
94	2.10.19.1	Headers	517
95	2.10.20	Use of Sockets over Internet Protocols Based on IPv6.....	517
96	2.10.20.1	Addressing	517
97	2.10.20.2	Compatibility with IPv4.....	518
98	2.10.20.3	Interface Identification.....	518
99	2.10.20.4	Options.....	519
100	2.10.20.5	Headers	520
101	2.11	Tracing.....	520
102	2.11.1	Tracing Data Definitions	521
103	2.11.1.1	Structures.....	521
104	2.11.1.2	Trace Stream Attributes.....	525
105	2.11.2	Trace Event Type Definitions	526
106	2.11.2.1	System Trace Event Type Definitions.....	526
107	2.11.2.2	User Trace Event Type Definitions.....	529

Contents

108	2.11.3	Trace Functions.....	529	
109	2.12	Data Types.....	530	
110	Chapter 3	System Interfaces	533	
111		Index.....	2171	
112	List of Tables			
113	2-1	Value of Level for Socket Options.....	513	
114	2-2	Socket-Level Options.....	514	
115	2-3	Trace Option: System Trace Events.....	527	
116	2-4	Trace and Trace Event Filter Options: System Trace Events.....	528	
117	2-5	Trace and Trace Log Options: System Trace Events.....	528	
118	2-6	Trace, Trace Log, and Trace Event Filter Options: System Trace		
119		Events.....	529	
120	2-7	Trace Option: User Trace Event	529	

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

125 **Background**

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

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

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

140 **Audience**

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

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

148 **Purpose**

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

- 150 • Application-Oriented

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

155

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

- 159 • Interface, Not Implementation
- 160 IEEE Std 1003.1-200x defines an interface, not an implementation. No distinction is made
161 between library functions and system calls; both are referred to as functions. No details of the
162 implementation of any function are given (although historical practice is sometimes
163 indicated in the RATIONALE section). Symbolic names are given for constants (such as
164 signals and error numbers) rather than numbers.
- 165 • Source, Not Object, Portability
- 166 IEEE Std 1003.1-200x has been written so that a program written and translated for execution
167 on one conforming implementation may also be translated for execution on another
168 conforming implementation. IEEE Std 1003.1-200x does not guarantee that executable (object
169 or binary) code will execute under a different conforming implementation than that for
170 which it was translated, even if the underlying hardware is identical.
- 171 • The C Language
- 172 The system interfaces and header definitions are written in terms of the standard C language
173 as specified in the ISO C standard.
- 174 • No Superuser, No System Administration
- 175 There was no intention to specify all aspects of an operating system. System administration
176 facilities and functions are excluded from IEEE Std 1003.1-200x, and functions usable only by
177 the superuser have not been included. Still, an implementation of the standard interface may
178 also implement features not in IEEE Std 1003.1-200x. IEEE Std 1003.1-200x is also not
179 concerned with hardware constraints or system maintenance.
- 180 • Minimal Interface, Minimally Defined
- 181 In keeping with the historical design principles of the UNIX system, the mandatory core
182 facilities of IEEE Std 1003.1-200x have been kept as minimal as possible. Additional
183 capabilities have been added as optional extensions.
- 184 • Broadly Implementable
- 185 The developers of IEEE Std 1003.1-200x endeavored to make all specified functions
186 implementable across a wide range of existing and potential systems, including:
- 187 1. All of the current major systems that are ultimately derived from the original UNIX
188 system code (Version 7 or later)
- 189 2. Compatible systems that are not derived from the original UNIX system code
- 190 3. Emulations hosted on entirely different operating systems
- 191 4. Networked systems
- 192 5. Distributed systems
- 193 6. Systems running on a broad range of hardware
- 194 No direct references to this goal appear in IEEE Std 1003.1-200x, but some results of it are
195 mentioned in the Rationale (Informative) volume of IEEE Std 1003.1-200x.
- 196 • Minimal Changes to Historical Implementations
- 197 When the original version of IEEE Std 1003.1 was published, there were no known historical
198 implementations that did not have to change. However, there was a broad consensus on a set
199 of functions, types, definitions, and concepts that formed an interface that was common to
200 most historical implementations.

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

- 208 1. By standardizing an interface like one in an historical implementation; for example,
209 directories
- 210 2. By specifying an interface that is readily implementable in terms of, and backwards
211 compatible with, historical implementations, such as the extended *tar* format defined in
212 the *pax* utility
- 213 3. By specifying an interface that, when added to an historical implementation, will not
214 conflict with it; for example, the *sigaction()* function

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

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

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

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

234 • Minimal Changes to Existing Application Code

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

238 **IEEE Std 1003.1-200x**

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

- 241 • Base Definitions
- 242 • Shell and Utilities
- 243 • System Interfaces (this volume)

- 244 • Rationale (Informative)

245 **This Volume**

246 The System Interfaces volume of IEEE Std 1003.1-200x describes the interfaces offered to
247 application programs by POSIX-conformant systems. Readers are expected to be experienced C
248 language programmers, and to be familiar with the Base Definitions volume of
249 IEEE Std 1003.1-200x.

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

- 251 • Chapter 1 explains the status of this volume of IEEE Std 1003.1-200x and its relationship to
252 other formal standards.
- 253 • Chapter 2 contains important concepts, terms, and caveats relating to the rest of this volume
254 of IEEE Std 1003.1-200x.
- 255 • Chapter 3 defines the functional interfaces to the POSIX-conformant system.

256 Comprehensive references are available in the index.

257 **Typographical Conventions**

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

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

262

263

264

265

266

267

268

269

270

271

272

273

274

275

276

277

278

279

280

281

282

283

284

285

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 Function	<i>system()</i>	
C-Language Function Family	<i>exec</i>	
C-Language Function Argument	<i>arg1</i>	
C-Language External Variable	<i>errno</i>	
C-Language Header	<sys/stat.h>	
C-Language Keyword	#define	
C-Language Macro with Argument	<i>assert()</i>	
C-Language Macro with No Argument	INET_ADDRSTRLEN	
Commands within a Utility	a, c	
Conversion Specification, Specifier/Modifier Character	<i>%A, g, E</i>	1
Environment Variable	PATH	
Error Number	[EINTR]	
Example Output	Hello, World	
Filename	/tmp	
Literal Character	'c'	2
Literal String	"abcde"	2
Optional Items in Utility Syntax	[]	
Parameter	<directory pathname>	
Special Character	<newline>	3

286
287
288
289
290
291
292
293
294
295

Reference	Example	Notes
Symbolic Limit, Configuration Value	{LINE_MAX}	4
Symbolic Constant	_POSIX_VDISABLE	
Syntax	#include <sys/stat.h>	
User Input and Example Code	echo Hello, World	5
Utility Name	awk	
Utility Operand	file_name	
Utility Option	-c	
Utility Option with Option-Argument	-w width	

296
297
298
299
300
301
302
303
304
305
306
307
308
309

Notes:

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

310
311
312
313
314
315
316
317
318
319
320

Shading is used to identify extensions and options; see Section 1.8.1 (on page 453).

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

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

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

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

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

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

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

321
322
323
324
325

Notes:

1. Symbolic limits are used in this volume of IEEE Std 1003.1-200x instead of fixed values for portability. The values of most of these constants are defined in the Base Definitions volume of IEEE Std 1003.1-200x, <limits.h> or <unistd.h>.
2. The values of errors are defined in the Base Definitions volume of IEEE Std 1003.1-200x, <errno.h>.

Trademarks

327

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

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

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

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

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

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

337 Motif[®], OSF/1[®], UNIX[®], and the “X Device” are registered trademarks and IT DialTone[™] and
338 The Open Group[™] are trademarks of The Open Group in the U.S. and other countries.

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

340 Sun[®] and Sun Microsystems[®] are registered trademarks of Sun Microsystems, Inc.

341 /usr/group[®] is a registered trademark of UniForum, the International Network of UNIX System |
342 Users.

Acknowledgements

343

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

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

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

351 **Austin Group Common Standards Committee**

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

353 Chair

354 Andrew Josey

355 Organizational Representatives

356 Donald W. Cragun [IEEE PASC]

357 Nicholas Stoughton [ISO/SC22 WG15]

358 Mark Brown [The Open Group]

359 Editor

360 Cathy Hughes

361 Technical Reviewers

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

376 Working Group

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

Acknowledgements

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

Referenced Documents

412

413 Normative References

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

416 Informative References

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

418 1984 /usr/group Standard

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

420 Almasi and Gottlieb

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

423 ANSI C

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

426 ANS X3.226-1994

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

429 Brawer

430 Steven Brawer, *Introduction to Parallel Programming*, Academic Press, 1989, ISBN:
431 0-12-128470-0.

432 DeRemer and Pennello Article

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

435 Draft ANSI X3J11.1

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

437 FIPS 151-1

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

439 FIPS 151-2

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

442 HP-UX Manual

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

444 IEC 60559: 1989

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

447 IEEE Std 754-1985

448 Standard for Binary Floating-Point Arithmetic.

449 IEEE Std 854-1987

450 Standard for Radix-Independent Floating-Point Arithmetic.

Referenced Documents

- 451 IEEE Std 1003.9-1992
452 Standard for Information Technology — POSIX FORTRAN 77 Language Interfaces — Part
453 1: Binding for System Application Program Interface API.
- 454 IETF RFC 791
455 Internet Protocol, Version 4 (IPv4), September 1981.
- 456 IETF RFC 819
457 The Domain Naming Convention for Internet User Applications, Z. Su, J. Postel, August
458 1982.
- 459 IETF RFC 822
460 Standard for the Format of ARPA Internet Text Messages, D.H. Crocker, August 1982.
- 461 IETF RFC 919
462 Broadcasting Internet Datagrams, J. Mogul, October 1984.
- 463 IETF RFC 920
464 Domain Requirements, J. Postel, J. Reynolds, October 1984.
- 465 IETF RFC 921
466 Domain Name System Implementation Schedule, J. Postel, October 1984.
- 467 IETF RFC 922
468 Broadcasting Internet Datagrams in the Presence of Subnets, J. Mogul, October 1984.
- 469 IETF RFC 1034
470 Domain Names — Concepts and Facilities, P. Mockapetris, November 1987.
- 471 IETF RFC 1035
472 Domain Names — Implementation and Specification, P. Mockapetris, November 1987.
- 473 IETF RFC 1123
474 Requirements for Internet Hosts — Application and Support, R. Braden, October 1989.
- 475 IETF RFC 1886
476 DNS Extensions to Support Internet Protocol, Version 6 (IPv6), C. Huitema, S. Thomson,
477 December 1995.
- 478 IETF RFC 2045
479 Multipurpose Internet Mail Extensions (MIME), Part 1: Format of Internet Message Bodies,
480 N. Freed, N. Borenstein, November 1996.
- 481 IETF RFC 2373
482 Internet Protocol, Version 6 (IPv6) Addressing Architecture, S. Deering, R. Hinden, July
483 1998.
- 484 IETF RFC 2460
485 Internet Protocol, Version 6 (IPv6), S. Deering, R. Hinden, December 1998.
- 486 Internationalisation Guide
487 Guide, July 1993, Internationalisation Guide, Version 2 (ISBN: 1-859120-02-4, G304),
488 published by The Open Group.
- 489 ISO C (1990)
490 ISO/IEC 9899:1990: Programming Languages — C, including Amendment 1:1995 (E), C
491 Integrity (Multibyte Support Extensions (MSE) for ISO C).
- 492 ISO 2375:1985
493 ISO 2375:1985, Data Processing — Procedure for Registration of Escape Sequences.

- 494 ISO 8652:1987
 495 ISO 8652:1987, Programming Languages — Ada (technically identical to ANSI standard
 496 1815A-1983).
- 497 ISO/IEC 1539:1990
 498 ISO/IEC 1539:1990, Information Technology — Programming Languages — Fortran
 499 (technically identical to the ANSI X3.9-1978 standard [FORTRAN 77]).
- 500 ISO/IEC 4873:1991
 501 ISO/IEC 4873:1991, Information Technology — ISO 8-bit Code for Information Interchange
 502 — Structure and Rules for Implementation.
- 503 ISO/IEC 6429:1992
 504 ISO/IEC 6429:1992, Information Technology — Control Functions for Coded Character
 505 Sets.
- 506 ISO/IEC 6937:1994
 507 ISO/IEC 6937:1994, Information Technology — Coded Character Set for Text
 508 Communication — Latin Alphabet.
- 509 ISO 7-bit or 8-bit coded character set for text communication using public communication
 510 networks, private communication networks, or interchange media, such as magnetic tapes
 511 and discs.
- 512 ISO/IEC 8802-3:1996
 513 ISO/IEC 8802-3:1996, Information Technology — Telecommunications and Information
 514 Exchange Between Systems — Local and Metropolitan Area Networks — Specific
 515 Requirements — Part 3: Carrier Sense Multiple Access with Collision Detection
 516 (CSMA/CD) Access Method and Physical Layer Specifications.
- 517 ISO/IEC 8859
 518 ISO/IEC 8859, Information Technology — 8-Bit Single-Byte Coded Graphic Character Sets:
- 519 Part 1: Latin Alphabet No. 1
 520 Part 2: Latin Alphabet No. 2
 521 Part 3: Latin Alphabet No. 3
 522 Part 4: Latin Alphabet No. 4
 523 Part 5: Latin/Cyrillic Alphabet
 524 Part 6: Latin/Arabic Alphabet
 525 Part 7: Latin/Greek Alphabet
 526 Part 8: Latin/Hebrew Alphabet
 527 Part 9: Latin Alphabet No. 5
 528 Part 10: Latin Alphabet No. 6
 529 Part 13: Latin Alphabet No. 7
 530 Part 14: Latin Alphabet No. 8
 531 Part 15: Latin Alphabet No. 9
- 532 This standard character set comprises 191 graphic characters covering the requirements of
 533 most of Western Europe.
- 534 ISO POSIX-1:1996
 535 ISO/IEC 9945-1:1996, Information Technology — Portable Operating System Interface
 536 (POSIX) — Part 1: System Application Program Interface (API) [C Language] (identical to
 537 ANSI/IEEE Std 1003.1-1996). Incorporating ANSI/IEEE Stds 1003.1-1990, 1003.1b-1993,
 538 1003.1c-1995, and 1003.1i-1995.
- 539 ISO POSIX-2:1993
 540 ISO/IEC 9945-2:1993, Information Technology — Portable Operating System Interface

Referenced Documents

- 541 (POSIX) — Part 2: Shell and Utilities (identical to ANSI/IEEE Std 1003.2-1992, as amended
542 by ANSI/IEEE Std 1003.2a-1992).
- 543 Issue 1
544 X/Open Portability Guide, July 1985 (ISBN: 0-444-87839-4).
- 545 Issue 2
546 X/Open Portability Guide, January 1987:
- 547 • Volume 1: XVS Commands and Utilities (ISBN: 0-444-70174-5)
 - 548 • Volume 2: XVS System Calls and Libraries (ISBN: 0-444-70175-3)
- 549 Issue 3
550 X/Open Specification, 1988, 1989, February 1992:
- 551 • Commands and Utilities, Issue 3 (ISBN: 1-872630-36-7, C211); this specification was
552 formerly X/Open Portability Guide, Issue 3, Volume 1, January 1989, XSI Commands
553 and Utilities (ISBN: 0-13-685835-X, XO/XPG/89/002)
 - 554 • System Interfaces and Headers, Issue 3 (ISBN: 1-872630-37-5, C212); this specification
555 was formerly X/Open Portability Guide, Issue 3, Volume 2, January 1989, XSI System
556 Interface and Headers (ISBN: 0-13-685843-0, XO/XPG/89/003)
 - 557 • Curses Interface, Issue 3, contained in Supplementary Definitions, Issue 3
558 (ISBN: 1-872630-38-3, C213), Chapters 9 to 14 inclusive; this specification was formerly
559 X/Open Portability Guide, Issue 3, Volume 3, January 1989, XSI Supplementary
560 Definitions (ISBN: 0-13-685850-3, XO/XPG/89/004)
 - 561 • Headers Interface, Issue 3, contained in Supplementary Definitions, Issue 3
562 (ISBN: 1-872630-38-3, C213), Chapter 19, Cpio and Tar Headers; this specification was
563 formerly X/Open Portability Guide Issue 3, Volume 3, January 1989, XSI Supplementary
564 Definitions (ISBN: 0-13-685850-3, XO/XPG/89/004)
- 565 Issue 4
566 CAE Specification, July 1992, published by The Open Group:
- 567 • System Interface Definitions (XBD), Issue 4 (ISBN: 1-872630-46-4, C204)
 - 568 • Commands and Utilities (XCU), Issue 4 (ISBN: 1-872630-48-0, C203)
 - 569 • System Interfaces and Headers (XSH), Issue 4 (ISBN: 1-872630-47-2, C202)
- 570 Issue 4, Version 2
571 CAE Specification, August 1994, published by The Open Group:
- 572 • System Interface Definitions (XBD), Issue 4, Version 2 (ISBN: 1-85912-036-9, C434)
 - 573 • Commands and Utilities (XCU), Issue 4, Version 2 (ISBN: 1-85912-034-2, C436)
 - 574 • System Interfaces and Headers (XSH), Issue 4, Version 2 (ISBN: 1-85912-037-7, C435)
- 575 Issue 5
576 Technical Standard, February 1997, published by The Open Group:
- 577 • System Interface Definitions (XBD), Issue 5 (ISBN: 1-85912-186-1, C605)
 - 578 • Commands and Utilities (XCU), Issue 5 (ISBN: 1-85912-191-8, C604)
 - 579 • System Interfaces and Headers (XSH), Issue 5 (ISBN: 1-85912-181-0, C606)
- 580 Knuth Article
581 Knuth, Donald E., *On the Translation of Languages from Left to Right*, Information and Control,

- 582 Volume 8, No. 6, October 1965.
- 583 KornShell
- 584 Bolsky, Morris I. and Korn, David G., *The New KornShell Command and Programming*
- 585 *Language*, March 1995, Prentice Hall.
- 586 MSE working draft
- 587 Working draft of ISO/IEC 9899:1990/Add3:draft, Addendum 3 — Multibyte Support
- 588 Extensions (MSE) as documented in the ISO Working Paper SC22/WG14/N205 dated 31
- 589 March 1992.
- 590 POSIX.0: 1995
- 591 IEEE Std 1003.0-1995, Guide to the POSIX Open System Environment (OSE) (identical to
- 592 ISO/IEC TR 14252).
- 593 POSIX.1: 1988
- 594 IEEE Std 1003.1-1988, Standard for Information Technology — Portable Operating System
- 595 Interface (POSIX) — Part 1: System Application Program Interface (API) [C Language].
- 596 POSIX.1: 1990
- 597 IEEE Std 1003.1-1990, Standard for Information Technology — Portable Operating System
- 598 Interface (POSIX) — Part 1: System Application Program Interface (API) [C Language].
- 599 POSIX.1a: 2000
- 600 IEEE Std 1003.1a-2000, Standard for Information Technology — Portable Operating System
- 601 Interface (POSIX) — Part 1: System Application Program Interface (API) — Amendment
- 602 ????: TITLE?? [C Language].
- 603 POSIX.1d: 1999
- 604 IEEE Std 1003.1d-1999, Standard for Information Technology — Portable Operating System
- 605 Interface (POSIX) — Part 1: System Application Program Interface (API) — Amendment 4:
- 606 Additional Realtime Extensions [C Language].
- 607 POSIX.1g: 2000
- 608 IEEE Std 1003.1g-2000, Standard for Information Technology — Portable Operating System
- 609 Interface (POSIX) — Part 1: System Application Program Interface (API) — Amendment 6:
- 610 Protocol-Independent Interfaces (PII).
- 611 POSIX.1j: 2000
- 612 IEEE Std 1003.1j-2000, Standard for Information Technology — Portable Operating System
- 613 Interface (POSIX) — Part 1: System Application Program Interface (API) — Amendment 5:
- 614 Advanced Realtime Extensions [C Language].
- 615 POSIX.1q: 2000
- 616 IEEE Std 1003.1q-2000, Standard for Information Technology — Portable Operating System
- 617 Interface (POSIX) — Part 1: System Application Program Interface (API) — Amendment 7:
- 618 Tracing [C Language].
- 619 POSIX.2b:-2000
- 620 IEEE Std 1003.2b: 2000, Standard for Information Technology — Portable Operating System
- 621 Interface (POSIX) — Part 2: Shell and Utilities — Amendment ???: TITLE??.
- 622 POSIX.2d:-1994
- 623 IEEE Std 1003.2d: 1994, Standard for Information Technology — Portable Operating System
- 624 Interface (POSIX) — Part 2: Shell and Utilities — Amendment 1: Batch Environment.
- 625 POSIX.13:-1998
- 626 IEEE Std 1003.13: 1998, Standard for Information Technology — Standardized Application
- 627 Environment Profile (AEP) — POSIX Realtime Application Support.

Referenced Documents

- 628 Sarwate Article
629 Sarwate, Dilip V., *Computation of Cyclic Redundancy Checks via Table Lookup*, Communications
630 of the ACM, Volume 30, No. 8, August 1988.
- 631 Sprunt, Sha, and Lehoczky
632 Sprunt, B., Sha, L., and Lehoczky, J.P., *Aperiodic Task Scheduling for Hard Real-Time Systems*,
633 The Journal of Real-Time Systems, Volume 1, 1989, Pages 27-60.
- 634 SVID, Issue 1
635 American Telephone and Telegraph Company, System V Interface Definition (SVID), Issue
636 1; Morristown, NJ, UNIX Press, 1985.
- 637 SVID, Issue 2
638 American Telephone and Telegraph Company, System V Interface Definition (SVID), Issue
639 2; Morristown, NJ, UNIX Press, 1986.
- 640 SVID, Issue 3
641 American Telephone and Telegraph Company, System V Interface Definition (SVID), Issue
642 3; Morristown, NJ, UNIX Press, 1989.
- 643 The AWK Programming Language
644 Aho, Alfred V., Kernighan, Brian W., and Weinberger, Peter J., *The AWK Programming
645 Language*, Reading, MA, Addison-Wesley 1988.
- 646 UNIX Programmer's Manual
647 American Telephone and Telegraph Company, *UNIX Time-Sharing System: UNIX
648 Programmer's Manual*, 7th Edition, Murray Hill, NJ, Bell Telephone Laboratories, January
649 1979.
- 650 XNS, Issue 4
651 CAE Specification, August 1994, Networking Services, Issue 4 (ISBN: 1-85912-049-0, C438),
652 published by The Open Group.
- 653 XNS, Issue 5
654 CAE Specification, February 1997, Networking Services, Issue 5 (ISBN: 1-85912-165-9, C523),
655 published by The Open Group.
- 656 XNS, Issue 5.2
657 Technical Standard, January 2000, Networking Services (XNS), Issue 5.2
658 (ISBN: 1-85912-241-8, C808), published by The Open Group.
- 659 X/Open Curses, Issue 4, Version 2
660 CAE Specification, May 1996, X/Open Curses, Issue 4, Version 2 (ISBN: 1-85912-171-3,
661 C610), published by The Open Group.
- 662 Yacc
663 *Yacc: Yet Another Compiler Compiler*, Stephen C. Johnson, 1978.
- 664 Parts of the following documents were used to create the base documents for
665 IEEE Std 1003.1-200x:
- 666 AIX 3.2 Manual
667 AIX Version 3.2 For RISC System/6000, Technical Reference: Base Operating System and
668 Extensions, 1990, 1992 (Part No. SC23-2382-00).
- 669 OSF/1
670 OSF/1 Programmer's Reference, Release 1.2 (ISBN: 0-13-020579-6).
- 671 OSF AES
672 Application Environment Specification (AES) Operating System Programming Interfaces

- 673 Volume, Revision A (ISBN: 0-13-043522-8).
- 674 System V Release 2.0
- 675 — UNIX System V Release 2.0 Programmer's Reference Manual (April 1984 - Issue 2).
- 676 — UNIX System V Release 2.0 Programming Guide (April 1984 - Issue 2).
- 677 System V Release 4.2
- 678 Operating System API Reference, UNIX SVR4.2 (1992) (ISBN: 0-13-017658-3).