INTERNATIONAL STANDARD

ISO/IEC 14519

ANSI/IEEE Std 1003.5b

First edition
1999-04-01

Information technology — POSIX® Ada Language Interfaces — Binding for System Application Program Interface (API) — Realtime Extensions

Technologies de l'information — Interfaces de langage POSIX® Ada — Boucle pour interface de programme d'application système (API) — Extensions temps réel

Reference number
ISO/IEC 14519:1999(E)
ANSI/IEEE
Std 1003.5b, 1996 edition

Abstract: This standard is part of the POSIX series of standards for applications and user interfaces to open systems. It defines the Ada language bindings as package specifications and accompanying textual descriptions of the application program interface (API). This standard supports application portability at the source code level through the binding between ISO 8652: 1987 (Ada) and ISO/IEC 9945-1: 1990 (IEEE Std 1003.1-1990) (POSIX), as amended by IEEE Std 1003.1b-1993, IEEE Std 1003.1c-1995, and IEEE Std 1003.1i-1995. Terminology and general requirements, process primitives, the process environment, files and directories, input and output primitives, device- and class-specific functions, language-specific services for Ada, system databases, synchronization, memory management, execution scheduling, clocks and timers, and message passing are covered. It also specifies behavior to support the binding that must be provided by the Ada compilation system.

Keywords: Ada, API, application portability, computer language bindings, open systems, operating systems, portable application, POSIX, POSIX language bindings, real-time, thread.

The Institute of Electrical and Electronics Engineers, Inc.
345 East 47th Street, New York, NY 10017-2394, USA

ISBN 0-7381-1570-3

IEEE Standards documents are developed within the IEEE Societies and the Standards Coordinating Committees of the IEEE Standards Board. Members of the committees serve voluntarily and without compensation. They are not necessarily members of the Institute. The standards developed within IEEE represent a consensus of the broad expertise on the subject within the Institute as well as those activities outside of IEEE that have expressed an interest in participating in the development of the standard.

Use of an IEEE Standard is wholly voluntary. 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.

Comments for revision of IEEE Standards are welcome from any interested party, regardless of membership affiliation with IEEE. Suggestions for changes in documents should be in the form of a proposed change of text, together with appropriate supporting comments.

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 all 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.

Comments on standards and requests for interpretations should be addressed to:

Secretary, IEEE Standards Board
445 Hoes Lane
P.O. Box 1331
Piscataway, NJ 08855-1331
USA

Note: 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. The IEEE shall not be responsible for identifying all patents for which a license may be required by an IEEE standard or for conducting inquiries into the legal validity or scope of those patents that are brought to its attention.

Contents

PAGE

Introduction xi

Section 1: General 1
1.1 Scope 1
1.2 Normative References 2
1.3 Conformance 3
1.3.1 Implementation Conformance 3
1.3.2 Application Conformance 6

Section 2: Terminology and General Requirements 7
2.1 Editorial Conventions 7
2.2 Definitions 8
2.2.1 Terminology 8
2.2.2 General Terms 10
2.2.3 Abbreviations 26
2.3 General Concepts 27
2.3.1 Process/Active Partition Relationship 27
2.3.2 Task/Thread Relationship 27
2.3.3 Ada Character Differences 27
2.3.4 Posix Signals Are Not Interrupts 28
2.3.5 System Call Exception Errors 28
2.3.6 Extended Security Controls 28
2.3.7 File Access Permissions 28
2.3.8 File Hierarchy 29
2.3.9 Filename Portability 29
2.3.10 File Times Update 29
2.3.11 Pathname Resolution 30
2.3.12 Allocated Storage 31
2.4 Package POSIX 31
2.4.1 Constants and Static Subtypes 36
2.4.2 POSIX Characters 41
2.4.3 POSIX Strings 42
2.4.4 String Lists 43
2.4.5 Option Sets 45
2.4.6 Error Codes and Exceptions 46
2.4.7 System Identification 52
2.4.8 Time Types 53
2.5 Package POSIX_Options 55
2.5.1 Implementation Options 55

2.6 Package POSIX_Limits 58
2.6.1 Implementation Limits 60
2.7 Package Ada_Streams 64
2.8 Package System 65
2.9 Package System_Storage_Elements 66
2.10 Package POSIX_Page_Alignment 67
2.10.1 Description 67
2.10.2 Error Handling 68
2.11 Environment Description 68
2.11.1 Time-Zone Information 69

Section 3: Process Primitives 71
3.1 Package POSIX_Process_Primitives 71
3.1.1 Process Template 73
3.1.2 Process Creation 75
3.1.3 Process Exit 79
3.1.4 Termination Status 80
3.1.5 Wait for Process Termination 82
3.2 Package POSIX_Unsafe_Process_Primitives 84
3.2.1 Process Creation 85
3.2.2 File Execution 87
3.3 Package POSIX_Signals 91
3.3.1 Signal Model 94
3.3.2 Signal Type 98
3.3.3 Standard Signals 98
3.3.4 Default Signal Actions 102
3.3.5 Tasking Safety 103
3.3.6 Interruptibility 103
3.3.7 Signal Sets 105
3.3.8 Block and Unblock Signals 106
3.3.9 Ignore Signals 107
3.3.10 Controlling Generation of Signal for Child Process 108
3.3.11 Examine Pending Signals 109
3.3.12 Signal Event Notification 109
3.3.13 Signal Information 110
3.3.14 Control Signal Queueing 112
3.3.15 Wait for Signal 113
3.3.16 Wait for Signal with Information 115
3.3.17 Signal Entries 116
3.3.18 Send a Signal 118
3.3.19 Queue a Signal 119
3.3.20 Interrupt a Task 120

Section 4: Process Environment 123
4.1 Package POSIX_Process_Identification 123
4.1.1 Process Identification Operations 124
4.1.2 Process Group Identification 125
4.1.3 User Identification 127

4.1.4 User and Group Identification . . . 128
4.2 Package POSIX_Process_Times . . . 130
4.2.1 Process Time Accounting . . . 130
4.3 Package POSIX_Process_Environment . . . 132
4.3.1 Argument List . . . 133
4.3.2 Environment Variables . . . 133
4.3.3 Process Working Directory . . . 137
4.4 Package POSIX_Calendar . . . 138
4.4.1 Time Information . . . 139
4.4.2 Operations on POSIX Times . . . 140
4.5 Package POSIX_Configurable_System_Limits . . . 142
4.5.1 Get Configurable System Options . . . 143
4.5.2 Get Configurable System Limits . . . 145

Section 5: Files and Directories . . . 149
5.1 Package POSIX_Permissions . . . 149
5.1.1 File Permissions . . . 150
5.1.2 Process Permission Set . . . 151
5.2 Package POSIX_Files . . . 152
5.2.1 Create and Remove Files . . . 153
5.2.2 Inquiries on File Types . . . 156
5.2.3 Modify File Pathnames . . . 156
5.2.4 Directory Iteration . . . 159
5.2.5 Update File Status Information . . . 161
5.2.6 Check File Accessibility . . . 163
5.3 Package POSIX_File_Status . . . 164
5.3.1 Access File Status . . . 165
5.3.2 Access Status Information . . . 166
5.4 Package POSIX_Configurable_File_Limits . . . 168
5.4.1 File Limits . . . 170
5.4.2 File Restrictions . . . 174
5.4.3 Pathname-Specific Options . . . 176

Section 6: Input and Output Primitives . . . 179
6.1 Package POSIX_IO . . . 179
6.1.1 Open or Create a File . . . 181
6.1.2 I/O Buffer Type . . . 187
6.1.3 Read from a File . . . 187
6.1.4 Write to a File . . . 190
6.1.5 File Position Operations . . . 195
6.1.6 Terminal Operations . . . 196
6.1.7 File Control . . . 196
6.1.8 Update File Status Information . . . 198
6.1.9 Truncate File to A Specified Length . . . 199
6.1.10 Synchronize a File's State . . . 200
6.1.11 Data Synchronization . . . 201
6.2 Package POSIX_File_Locking . . . 202
6.2.1 Lock and Unlock a Region of a File . . . 203

6.3 Package POSIX_Asynchronous_IO 204
6.3.1 AIO Descriptor Type 206
6.3.2 Attributes of AIO Control Blocks 208
6.3.3 Asynchronous Read 211
6.3.4 Asynchronous Write 212
6.3.5 List Directed I/O 214
6.3.6 Retrieve Status of Asynchronous I/O Request 216
6.3.7 Retrieve Bytes Transferred by Asynchronous I/O Request . . 218
6.3.8 Cancel Asynchronous I/O Request 219
6.3.9 Wait for Asynchronous I/O Request to Complete 220
6.3.10 Asynchronous File and Data Synchronization 221

Section 7: Device- and Class-Specific Functions 225
7.1 General Terminal Interface 225
7.1.1 Interface Characteristics 225
7.2 Package POSIX_Terminal_Functions 232
7.2.1 Terminal Characteristics 234
7.2.2 Input Modes 237
7.2.3 Output Modes 239
7.2.4 Control Modes 239
7.2.5 Local Modes 240
7.2.6 Retrieve and Define Terminal Modes and Bits per Character . 242
7.2.7 Special Control Characters 243
7.2.8 Noncanonical Controls 245
7.2.9 Baud Rate Subprograms 245
7.2.10 Line Control Operations 246
7.2.11 Foreground Process Group ID 248
7.2.12 Generate Terminal Pathname 249

Section 8: Language-Specific Services for Ada 251
8.1 Interoperable Ada I/O Services 251
8.1.1 Requirements 251
8.1.2 Additional Interpretation of the POSIX.1 Standard 255
8.2 Package POSIX_Supplement_to_Ada_IO 256
8.2.1 Parse Form values 257
8.2.2 Flush Files 258

Section 9: System Databases 259
9.1 Package POSIX_User_Database 259
9.1.1 Access Contents of a User Database Item 259
9.1.2 Access User Database Items 260
9.2 Package POSIX_Group_Database 261
9.2.1 Access Contents of a Group Database Item 262
9.2.2 Access Elements of the Group Item List of Members 262
9.2.3 Access Group Database Items 263

Section 10: Data Interchange Format 265

Section 11: Synchronization 267

11.1 Package POSIX_Semaphores 267
11.1.1 Semaphore and Semaphore Descriptor Types 268
11.1.2 Initialize an Unnamed Semaphore 269
11.1.3 Finalize an Unnamed Semaphore 270
11.1.4 Create/Open a Named Semaphore 271
11.1.5 Close a Named Semaphore 273
11.1.6 Remove a Named Semaphore 274
11.1.7 Decrement a Semaphore 275
11.1.8 Increment a Semaphore 276
11.1.9 Get the Value of a Semaphore 276
11.2 Package POSIX_Mutexes 277
11.2.1 Mutex Ownership 278
11.2.2 Mutex and Mutex Descriptor Types 279
11.2.3 Mutex Attributes Type 279
11.2.4 Mutex Process Shared Attribute 280
11.2.5 Mutex Locking Policy Attributes 281
11.2.6 Initialize and Finalize a Mutex 283
11.2.7 Change the Ceiling Priority of a Mutex 285
11.2.8 Lock and Unlock a Mutex 286
11.3 Package POSIX_Condition_Variables 287
11.3.1 Condition and Condition Descriptor Types 288
11.3.2 Condition Variable Attributes Type 288
11.3.3 Condition Process Shared Attribute 289
11.3.4 Initialize and Finalize a Condition 290
11.3.5 Broadcast and Signal a Condition 292
11.3.6 Wait on a Condition 293
Section 12: Memory Management 295
12.1 Package POSIX_Memory_Locking 296
12.1.1 Lock/Unlock a Process's Address Space 296
12.2 Package POSIX_Memory_Range_Locking 298
12.2.1 Lock/Unlock a Range of Process Address Space 298
12.3 Package POSIX_Memory_Mapping 300
12.3.1 Map Process Addresses to a Memory Object 301
12.3.2 Unmap Memory 305
12.3.3 Change Memory Protection 306
12.3.4 Memory Object Synchronization 308
12.4 Package POSIX_Shared_Memory_Objects 309
12.4.1 Open a Shared Memory Object 310
12.4.2 Remove a Shared Memory Object 312
12.5 Package POSIX_Generic_Shared_Memory 313
12.5.1 Open Shared Memory 314
12.5.2 Access Shared Memory 317
12.5.3 Close Shared Memory 317
12.5.4 Remove Shared Memory 318
12.5.5 Lock/Unlock Shared Memory 318

Section 13: Execution Scheduling 321

13.1 Scheduling Concepts and Terminology 321
13.2 Package POSIX_Process_Scheduling 321
13.2.1 Scheduling Parameters 322
13.2.2 Scheduling Policies 323
13.2.3 Modify Process Scheduling Policy and Parameters 323
13.2.4 Process Yield CPU 325
13.2.5 Get Scheduling Limits 325
13.3 Task Scheduling 326
13.3.1 Dynamic Priorities 326
13.3.2 Task Dispatching Policy Pragma 326
13.3.3 Task Creation Attributes Pragma 327
13.3.4 Task Yield CPU 327
13.4 Synchronization Scheduling 328

Section 14: Clocks and Timers 329
14.1 Package POSIX_Timers 329
14.1.1 Clock and Timer Types 330
14.1.2 Realtime Clock 330
14.1.3 Timer State and Timer Options 331
14.1.4 Clock Operations 332
14.1.5 Create a Timer 333
14.1.6 Delete a Timer 335
14.1.7 Timer Operations 335
14.2 High Resolution Delay 338

Section 15: Message Passing 339
15.1 Package POSIX_Message_Queues 339
15.1.1 Message Queue Attributes 341
15.1.2 Open a Message Queue 343
15.1.3 Close a Message Queue 346
15.1.4 Remove a Message Queue 347
15.1.5 Send a Message to a Message Queue 348
15.1.6 Receive a Message from a Message Queue 349
15.1.7 Generic Message Passing 350
15.1.8 Notify Process that a Message is Available 352
15.1.9 Set/Query Message Queue Attributes 353

Section 16: Task Management 355
16.1 Package Ada_Task_Identification 355
16.1.1 Description 355

Annex A (informative) Bibliography 357

Annex B (informative) Rationale and Notes 359
B.1 General 359
B.1.1 Purpose and Audience 359
B.1.2 Relation to Other POSIX Standards 359
B.1.3 Ada Language 360
B.1.4 Implementation Model 361

B.1.5 Level of Binding . . . 361
B.1.6 Form of Document . . . 362
B.1.7 Global Issues . . . 363
B.1.8 Naming . . . 364
B.1.9 Mapping C Features to Ada . . . 365
B.1.10 Conformance . . . 370
B.2 Terminology and General Requirements . . . 370
B.2.1 Conventions . . . 370
B.2.2 Definitions . . . 370
B.2.3 General Concepts . . . 371
B.2.4 Package POSIX . . . 374
B.2.5 Package POSIX_Options . . . 380
B.2.6 Package POSIX_Limits . . . 381
B.2.7 Package Ada_Streams . . . 381
B.2.8 Package System . . . 381
B.2.9 Package System_Storage_Elements . . . 381
B.2.10 Package POSIX_Page_Alignment . . . 381
B.2.11 Environment Description . . . 382
B.3 Process Primitives . . . 382
B.3.1 Package POSIX_Process_Primitives . . . 382
B.3.2 Process Creation . . . 383
B.3.3 Process Exit . . . 384
B.3.4 Wait for Process Termination . . . 386
B.3.5 Package POSIX_Unsafe_Process_Primitives . . . 386
B.3.6 Package POSIX_Signals . . . 386
B.3.7 Signal Model . . . 387
B.3.8 Signal Masking and Related Concepts . . . 387
B.3.9 Tasking Safety . . . 391
B.3.10 Signal Type . . . 392
B.3.11 Standard Signals . . . 392
B.3.12 Signal Sets . . . 393
B.3.13 Examine and Change Signal Action . . . 393
B.3.14 Signal Queueing . . . 394
B.3.15 Signal Notification Model . . . 395
B.3.16 Examine Pending Signals . . . 396
B.3.17 Send a Signal . . . 396
B.3.18 Binding for *sigsuspend* . . . 396
B.3.19 Synchronously Accept a Signal . . . 396
B.3.20 Queue a Signal to a Process . . . 397
B.3.21 Send a Signal to a Thread . . . 397
B.3.22 Schedule Alarm . . . 398
B.3.23 Suspend Process Execution . . . 398
B.3.24 Delay Process Execution . . . 398
B.3.25 Task Signal Entries . . . 398
B.3.26 Composability Considerations . . . 399
B.4 Process Environment . . . 400
B.4.1 Organization . . . 400
B.4.2 Package POSIX_Process_Identification . . . 400

B.4.3 Package Process_Times . . . 401
B.4.4 Package POSIX_Process_Environment . . . 402
B.4.5 Package POSIX_Calendar . . . 404
B.4.6 Package POSIX_Configurable_System_Limits . . . 405
B.5 Files and Directories . . . 405
B.5.1 Organization . . . 405
B.5.2 Package POSIX_Permissions . . . 406
B.5.3 Package POSIX_Files . . . 407
B.5.4 Package POSIX_File_Status . . . 408
B.5.5 Package POSIX_Configurable_File_Limits . . . 408
B.5.6 Directory Operations . . . 409
B.6 Input and Output Primitives . . . 409
B.6.1 Package POSIX_IO . . . 409
B.6.2 Package POSIX_File_Locking . . . 414
B.6.3 Package POSIX_Asynchronous_IO . . . 415
B.7 Device- and Class-Specific Functions . . . 417
B.8 Language-Specific Services for Ada . . . 418
B.8.1 General Rationale . . . 418
B.8.2 Limitations on Interoperability . . . 420
B.8.3 Rationale for Form Parameter . . . 421
B.8.4 Unaffected Implementation Dependencies . . . 426
B.8.5 Notes on Specific Topics . . . 426
B.9 System Databases . . . 428
B.9.1 Requirements from POSIX.1 . . . 428
B.9.2 Rationale for the Current Design . . . 429
B.10 Data Interchange Format . . . 430
B.11 Synchronization . . . 430
B.11.1 Package POSIX_Semaphores . . . 430
B.11.2 Mutexes & Condition Variables . . . 433
B.12 Memory Management . . . 436
B.12.1 Process Memory Locking . . . 436
B.12.2 Package POSIX_Memory_Mapping . . . 437
B.12.3 Package POSIX_Shared_Memory_Objects . . . 438
B.12.4 Package POSIX_Generic_Shared_Memory . . . 439
B.13 Priority Process Scheduling . . . 441
B.13.1 Scheduling Concepts and Terminology . . . 442
B.13.2 Package POSIX_Process_Scheduling . . . 442
B.13.3 Task Scheduling . . . 443
B.13.4 Synchronization Scheduling . . . 444
B.13.5 Thread Scheduling Pragmas vs. Environment Variables . . . 445
B.14 Clocks and Timers . . . 446
B.14.1 Types and Constants . . . 446
B.14.2 Clock Operations . . . 447
B.14.3 Timer Creation . . . 447
B.14.4 Timer Operations . . . 447
B.14.5 High Resolution Sleep . . . 448
B.15 Message Passing . . . 448
B.15.1 Message Queue Attributes . . . 449

B.15.2 Priorities 449
B.15.3 Generic Message Passing 449
B.15.4 Notification 450
B.16 Task Identification 450
B.17 Thread-Specific Data 450

Annex C (informative) Ada/C Cross-References 451
C.1 Ada-to-C Cross-Reference 451
C.2 C-to-Ada Cross-Reference 469

Alphabetic Topical Index 489

TABLES

Table2.1 – Typographical Conventions 7
Table2.2 – Constant and Subtype Correspondences 37
Table2.3 – Option Set Comparisons 46
Table2.4 – Static Subtypes and Options 59
Table2.5 – Portable Constants and Limits 63
Table2.6 – Static Subtypes and Limits 64
Table3.1 – Default Actions for Job Control Signals 103
Table4.1 – Functions for System-Wide Options 145
Table4.2 – Configurable System Limits 146
Table6.1 – Standard File Descriptors 182
Table6.2 – Error Codes and AIO Status Values 217
Table7.1 – `Terminal_Characteristics` Components 235
Table7.2 – `Terminal_Modes` Values for Input Control 237
Table7.3 – `Terminal_Modes` Values for Output Control 239
Table7.4 – `Terminal_Modes` Values for Hardware Control 240
Table7.5 – `Terminal_Modes` Values for Local Control Modes 241
Table7.6 – Special Control Character Usage 244

International Standard ISO/IEC 14519:1999(E)

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. 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.

International Standard ISO/IEC 14519 was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 22, *Programming languages, their environments and system software interfaces*.

Annexes A to C of this International Standard are for information only.

International Organization for Standardization/International Electrotechnical Commission
Case postale 56 • CH-1211 Genève 20 • Switzerland

Introduction

(This introduction is not a part of IEEE Std 1003.5b-1996, IEEE Standard for Information Technology—POSIX Ada Language Interfaces—Part 1: Binding for System Application Program Interface (API) Amendment 1: Realtime Extensions but is included for information only.)

This standard is an amended version of IEEE Std 1003.5-1992. The basic goal of this standard is to provide an Ada application program interface for the language-independent services made accessible to C-language applications programs by the interfaces defined in ISO/IEC 9945-1: 1990 (IEEE Std 1003.1-1990), as amended by IEEE Std 1003.1b-1993, IEEE Std 1003.1c-1995, and IEEE Std 1003.1i-1995.

The intent is to support portability of Ada applications via a standard binding to the services provided by a POSIX-conforming operating system. POSIX is defined by the standard C-language interfaces cited above. Therefore, much of the work in producing this standard was deciding what features of those C-language interfaces represented POSIX functionality, as opposed to C-language specific features.

This standard provides package specifications and accompanying textual description for a set of Ada packages that represent the POSIX system. This standard also specifies behavior to support the binding that must be provided by the Ada compilation system, and further defines behavior specified as implementation defined in the Ada Language standard (particularly in the area of Text_IO) for use in a POSIX environment.

The emphasis in POSIX is on application program portability, so the interfaces in this standard are not sufficient to implement an Ada compilation system or a POSIX shell (as defined in IEEE Std 1003.2-1992). For an application, the intent is that a Strictly Conforming POSIX.5 Application (one that uses only the facilities in this standard and that does not depend on implementation defined behavior) can be ported to any Conforming Implementation of these interfaces, and that the binding makes it easy to identify where a program is not strictly conforming and makes such programs easier to port.

Organization of This Standard

The standard is divided into three parts:

— Statement of scope, list of normative references, and conformance information (Section 1)

— Definitions and global concepts (Section 2)

— The various interface facilities (Sections 3 through 16)

The content of the sections parallels that of the correspondingly numbered sections of the base standards (IEEE Std 1003.1-1990, as amended by IEEE Std 1003.1b-1993,

IEEE Std 1003.1c-1995, and IEEE Std 1003.1i-1995), with some changes required to fit the style of IEEE Std 1003.5-1993. There is no Section 10, since there is no Ada binding for that section (Data Interchange Formats) of the base standards.

This introduction, any footnotes, notes accompanying the text, and the *informative* annexes are not considered part of this standard.

Related Standards Activities

Activities to extend this standard to address additional requirements are in progress, and similar efforts can be anticipated in the future.

The following areas are under active consideration at this time, or are expected to become active in the near future:[1]

(1) Language-independent descriptions of the services defined by IEEE-Std 1003.1-1990 and its amendments.

(2) C and FORTRAN language bindings to (1)

(3) Shell and utility facilities

(4) Verification testing methods

(5) Secure/Trusted system considerations

(6) Network interface facilities

(7) System administration

(8) Graphical user interfaces

(9) Profiles describing application- or user-specific combinations of open systems standards for: supercomputing, multiprocessor, and batch extensions; transaction processing; realtime systems; and multiuser systems based on historical models

(10) An overall guide to POSIX-based or related Open Systems standards and profiles

Extensions are approved as “amendments” or “revisions” to this standard, following IEEE and ISO/IEC procedures.

Approved amendments are published separately until the full standard is reprinted and such amendments are incorporated in their proper positions.

If you have an interest in participating in the PASC working groups addressing these issues, please send your name, address, and phone number to the Secretary, IEEE Standards Board, Institute of Electrical and Electronics Engineers, Inc., P.O. Box 1331, 445 Hoes Lane, Piscataway, NJ 08855-1331, USA, and ask to have this forwarded to the chair of the appropriate PASC working group. If you have an interest in participating in this work at the international level, contact your ISO/IEC national body.

[1] A *Standards Status Report* that lists all current IEEE Computer Society standards projects is available from the IEEE Computer Society, 1730 Massachusetts Avenue NW, Washington, DC 20036-1903, USA; Telephone: +1 202 371-0101; FAX: +1 202 728-9614.