

## 1.10 Unix Standards

At the time of this writing, the most interesting Unix standardization activity was being done by The Austin Common Standards Revision Group (CSRG). Their efforts have produced roughly 4,000 pages of specifications covering over 1,700 programming interfaces [Josey 2002]. These specifications carry both the IEEE POSIX designation as well as The Open Group's Technical Standard designation. The net result is that you'll likely encounter references to the same standard by various names: ISO/IEC 9945:2002, IEEE Std 1003.1-2001, and the Single Unix Specification Version 3, for example. In this text, we will refer to this standard as simply *The POSIX Specification*, except in sections like this one where we are discussing specifics of various older standards.

The easiest way to acquire a copy of this consolidated standard is to either order it on CD-ROM or access it via the Web (free of charge). The starting point for either of these methods is

<http://www.UNIX.org/version3>

## Background on POSIX

POSIX is an acronym for Portable Operating System Interface. POSIX is not a single standard, but a family of standards being developed by the Institute for Electrical and Electronics Engineers, Inc., normally called the *IEEE*. The POSIX standards have also been adopted as international standards by ISO and the International Electrotechnical Commission (IEC), called ISO/IEC. The POSIX standards have an interesting history, which we cover only briefly here:

- IEEE Std 1003.1-1988 (317 pages) was the first POSIX standard. It specified the C language interface into a Unix-like kernel and covered the following areas: process primitives (`fork`, `exec`, signals, and timers), the environment of a process (user IDs and process groups), files and directories (all the I/O functions), terminal I/O, system databases (password file and group file), and the `tar` and `cpio` archive formats.

The first POSIX standard was a trial-use version in 1986 known as "IEEE-IX." The name "POSIX" was suggested by Richard Stallman.

- IEEE Std 1003.1-1990 (356 pages) was next, and it was also known as ISO/IEC 9945-1: 1990. Minimal changes were made from the 1988 to the 1990 version. Appended to the title was "Part 1: System Application Program Interface (API) [C Language]," indicating that this standard was the C language API.
- IEEE Std 1003.2-1992 came next in two volumes (about 1,300 pages). Its title contained "Part 2: Shell and Utilities." This part defined the shell (based on the System V Bourne shell) and about 100 utilities (programs normally executed from a shell, from `awk` and `basename` to `vi` and `yacc`). Throughout this text, we will refer to this standard as *POSIX.2*.
- IEEE Std 1003.1b-1993 (590 pages) was originally known as IEEE P1003.4. This was an update to the 1003.1-1990 standard to include the real-time extensions developed by the P1003.4 working group. The 1003.1b-1993 standard added the following items to the 1990 standard: file synchronization, asynchronous I/O, semaphores, memory management (`mmap` and shared memory), execution scheduling, clocks and timers, and message queues.
- IEEE Std 1003.1, 1996 Edition [IEEE 1996] (743 pages) came next and included 1003.1-1990 (the base API), 1003.1b-1993 (real-time extensions), 1003.1c-1995 (`pthread`s), and 1003.1i-1995 (technical corrections to 1003.1b). This standard was also called ISO/IEC 9945-1: 1996. Three chapters on threads were added, along with additional sections on thread synchronization (mutexes and condition variables), thread scheduling, and synchronization scheduling. Throughout this text, we will refer to this standard as *POSIX.1*. This standard also contains a Foreword stating that ISO/IEC 9945 consists of the following parts:
  - Part 1: System API (C language)
  - Part 2: Shell and utilities
  - Part 3: System administration (under development)

[Parts 1](#) and [2](#) are what we call POSIX.1 and POSIX.2.

Over one-quarter of the 743 pages are an appendix titled "Rationale and Notes." This appendix contains historical information and reasons why certain features were included or omitted. Often, the rationale is as informative as the official standard.

- IEEE Std 1003.1g: Protocol-independent interfaces (PII) became an approved standard in 2000. Until the introduction of The Single Unix Specification Version 3, this POSIX work was the most relevant to the topics covered in this book. This is the networking API standard and it defines two APIs, which it calls Detailed Network Interfaces (DNIs):
  1. DNI/Socket, based on the 4.4BSD sockets API
  2. DNI/XTI, based on the X/Open XPG4 specification

Work on this standard started in the late 1980s as the P1003.12 working group (later renamed P1003.1g). Throughout this text, we

will refer to this standard as *POSIX.1g*.

The current status of the various POSIX standards is available from

<http://www.pasc.org/standing/sd11.html>

## Background on The Open Group

The Open Group was formed in 1996 by the consolidation of the X/Open Company (founded in 1984) and the Open Software Foundation (OSF, founded in 1988). It is an international consortium of vendors and end-user customers from industry, government, and academia. Here is a brief background on the standards they produced:

- X/Open published the *X/Open Portability Guide*, Issue 3 (XPG3) in 1989.
- Issue 4 was published in 1992, followed by Issue 4, Version 2 in 1994. This latest version was also known as "Spec 1170," with the magic number 1,170 being the sum of the number of system interfaces (926), the number of headers (70), and the number of commands (174). The latest name for this set of specifications is the "X/Open Single Unix Specification," although it is also called "Unix 95."
- In March 1997, Version 2 of the Single Unix Specification was announced. Products conforming to this specification were called "Unix 98." We will refer to this specification as just "Unix 98" throughout this text. The number of interfaces required by Unix 98 increases from 1,170 to 1,434, although for a workstation this jumps to 3,030, because it includes the Common Desktop Environment (CDE), which in turn requires the X Window System and the Motif user interface. Details are available in [Josey 1997] and at <http://www.UNIX.org/version2>. The networking services that are part of Unix 98 are defined for both the sockets and XTI APIs. This specification is nearly identical to POSIX.1g.

Unfortunately, Unix 98 referred to networking standards as XNS: X/Open Networking Services. The version of this document that defines sockets and XTI for Unix 98 ([Open Group 1997]) is called "XNS Issue 5." In the networking world XNS has always been an abbreviation for the Xerox Network Systems architecture. We will avoid this use of XNS and refer to this X/Open document as just the Unix 98 network API standard.

## Unification of Standards

The above brief backgrounds on POSIX and The Open Group both continue with The Austin Group's publication of The Single Unix Specification Version 3, as mentioned at the beginning of this section. Getting over 50 companies to agree on a single standard is certainly a landmark in the history of Unix. Most Unix systems today conform to some version of POSIX.1 and POSIX.2; many comply with The Single Unix Specification Version 3.

Historically, most Unix systems show either a Berkeley heritage or a System V heritage, but these differences are slowly disappearing as most vendors adopt the standards. The main differences still existing deal with system administration, one area that no standard currently addresses.

The focus of this book is on The Single Unix Specification Version 3, with our main focus on the sockets API. Whenever possible we will use the standard functions.

## Internet Engineering Task Force (IETF)

The Internet Engineering Task Force (IETF) is a large, open, international community of network designers, operators, vendors, and researchers concerned with the evolution of the Internet architecture and the smooth operation of the Internet. It is open to any interested individual.

The Internet standards process is documented in RFC 2026 [Bradner 1996]. Internet standards normally deal with protocol issues and not with programming APIs. Nevertheless, two RFCs (RFC 3493 [Gilligan et al. 2003] and RFC 3542 [Stevens et al. 2003]) specify the sockets API for IPv6. These are informational RFCs, not standards, and were produced to speed the deployment of portable applications by the numerous vendors working on early releases of IPv6. Although standards bodies tend to take a long time, many APIs were standardized in The Single Unix Specification Version 3.

[ [Team LiB](#) ]

◀ PREVIOUS    NEXT ▶