INTERNATIONAL STANDARDIZED PROFILE

ISO/IEC ISP 15287-2

IEEE Std 1003.13

First edition
2000-03-15

Information technology — Standardized Application Environment Profile —

Part 2: **Posix® Realtime Application Support (AEP)**

Technologies de l'information — Profil d'environnement d'application normalisée —

Partie 2: Support d'application en temps réel POSIX® (AEP)

Reference number
ISO/IEC ISP 15287-2:2000(E)
IEEE
Std 1003.13, 1998 edition

Abstract: This standard is part of the POSIX® series of standardized profiles for open systems. It defines environment profiles for portable realtime applications.

Keywords: AEP, application portability, data processing, environment, open systems, operating system, portable application, POSIX profiles, realtime application environments, realtime environment

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

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

Print: ISBN 0-7381-2319-6 SH94843
PDF: ISBN 0-7381-2320-X SS94843

International Standard ISO/IEC ISP 15287-2:2000
IEEE Std 1003.13-1998

Information Technology—
Standardized Application Environment Profile—
POSIX® Realtime Application Support (AEP)

Sponsor

Portable Applications Standards Committee
of the
IEEE Computer Society

Approved 19 March 1998
IEEE-SA Standards Board

Approved 4 January 2000 by the
International Organization for Standardization
and by the
International Electrotechnical Commission

Adopted as an International Standard by the
International Organization for Standardization
and by the
International Electrotechnical Commission

Published by
The Institute of Electrical and Electronics Engineers, Inc.

American National Standard

International Standardized Profile ISO/IEC ISP 15287-2:2000(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. In addition to developing International Standards, ISO/IEC JTC 1 has created a Special Group on Functional Standardization for the elaboration of International Standardized Profiles.

An International Standardized Profile is an internationally agreed, harmonized document which identifies a standard or group of standards, together with options and parameters, necessary to accomplish a function or a set of functions.

Draft International Standardized Profiles are circulated to national bodies for voting. Publication as an International Standardized Profile requires approval by at least 75 % of the national bodies casting a vote.

Attention is drawn to the possibility that some of the elements of this part of ISO/IEC ISP 15287 may be the subject of patent rights. ISO and IEC shall not be held responsible for identifying any or all such patent rights.

International Standardized Profile ISO/IEC ISP 15287-2 was prepared with the collaboration of

— Asia-Oceania Workshop (AOW);

— European Workshop for Open Systems (EWOS);

— Open Systems Environment Implementors' Workshop (OIW).

ISO/IEC ISP 15287 consists of the following parts, under the general title *Information technology — Standardized Application Environment Profile*:

— *Part 1: PSE 10-HIP — Posix Supercomputing Application Environment Profile*

— *Part 2: Posix® Realtime Application Support (AEP)*

Annexes A and B form a normative part of this part of ISO/IEC ISP 15287. Annex C is for information only.

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

IEEE Standards documents are developed within the IEEE Societies and the Standards Coordinating Committees of the IEEE Standards Association (IEEE-SA) 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-SA 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 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

TABLES

Introduction

The purpose of this standard is to define realtime application environments based on the ISO/IEC 9945 series of standards. It is intended for realtime systems implementors and realtime applications software developers.

This standard is designed to support building systems where not all the interconnected boxes use the same profile, for example, a hierarchical system where the bottom-level device controllers use the "minimal" profile, the next level up follows the larger "control" profile, and so on. There are interfaces called out for the smaller profiles that make no sense in an isolated box; those interfaces are there solely to support the construction of heterogeneous systems, and systems of communicating peers. Such systems are very common in practice.

To summarize, this standard is embedded in a much larger and widely supported set of standards, which yields benefits during code development, as much development and testing is done on the larger and more comfortable systems. It also may be used in the construction of large and heterogenous systems.

Four profiles have been defined to reflect the wide range of system requirements presented by realtime designs. The intent is to provide a meaningful and coherent set of interfaces that will provide software vendors and consumers with a uniform framework for describing and specifying operating system capabilities. This allows an application writer to construct an application that may be easily moved to a different system that supports the same profile. Similarly, it allows a vendor to claim conformance with an established standard, even if that vendor's implementation does not support the full POSIX feature set.

Initially, the focus of this standard is to provide standardized environments supporting the C language. Options are provided for bindings to the Ada programming language as well as for the C language. Bindings for other languages to these services may be developed and this standard will be updated as appropriate.

Within this document, the term "POSIX/RT-AEP" refers to this standard.

Organization of This Part of IEEE Std 1003.13-1998 (POSIX.13)

This standard is divided into eight elements:

(1) General (Section 1)

(2) Normative references (Section 2)

(3) Definitions (Section 3)

(4) Conventions and abbreviations (Section 4)

(5) Conformance (Section 5)

(6) The various realtime profiles (Sections 6 through 9)

(7) ISPICS requirements (C) (Annex A)

(8) ISPICS requirements (Ada) (Annex B)

References are used to direct the reader to other related sections.

Informative annexes are not part of the standard and are provided for information only. They are provided for guidance and to facilitate understanding.

In publishing this standard, the developers simply intend to provide a yardstick against which various realtime application environments can be measured for conformance. It is not the intent of the developers to measure or rate any products, to reward or sanction any vendors of products for conformance or lack of conformance to this standard, or to attempt to enforce this standard by these or any other means. The responsibility for determining the degree of conformance or lack thereof with this standard rests solely with the individual who is evaluating the product claiming to be in conformance with this standard.

Base Documents

The various realtime application environments described herein are based on the ISO/IEC 9945 and IEEE 1003 family of documents as well as ISO 9899, 1539, 8652, and 8859.

Scenario

This standard is based directly on existing small and/or realtime (typically non-UNIX™) kernel practice as well as the growing body of practice with POSIX conformant kernels having realtime features. The general approach taken in this standard is to specify interfaces, taken from POSIX, sufficient to deliver the functionality typical of current realtime systems, (see Table 1-19 through Table 1-27).

Each profile is specified with full features, to give users clear direction. Vendors may provide means to configure out those parts that are not needed by specific applications. Vendors wishing to expand on the specified profiles are strongly encouraged to take the added interfaces from other POSIX.13 profiles or from the base standards, rather than invent new interfaces.

For each profile, the minimum hardware typically required is specified. This is the hardware assumed to be present; implementations may, of course, have more, but nothing in the profile requires—either directly or indirectly—more than the specified minimum hardware model.

Audience

The intended audience for this class of profiles is all persons concerned with an industry-wide standard realtime application environment based on the POSIX suite of standards. This includes at least four groups of people:

(1) Persons buying hardware and software systems.

(2) Persons managing companies that are deciding on future corporate computing directions.

(3) Persons implementing realtime operating systems.

(4) Persons developing realtime applications where portability is a primary objective.

Background

The developers of POSIX/RT-AEP 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. In the course of their deliberations, the developers reviewed related U.S. and international standards, both published and in progress.

Conceptually, POSIX/RT-AEP describes a set of application environment profiles needed for the construction and execution of portable realtime application programs.

The developers of this standard have tried to capture the functionality of existing realtime systems in a reasonable number of profiles that specify predominate application environments. It is felt that these profiles, although not optimum, are a best fit to existing classes of applications and systems.

Features of several commercial realtime kernels were considered. These included **pSOS**™[1)], **VRTX32**™[2)], and **VxWorks**™[3)]. Since these products were commercially successful, they must have addressed a significant market segment. In addition, the uniprocessor subset of VITA's **ORKID** specification, NGCR's "***Tiny Real Time***" (TRT), and the **µITRON** specification were examined. These were proposed standard interfaces for small realtime embedded systems.

The following is a list of features that are representative of current realtime systems and highlights the range of system requirements. While some concepts are common to virtually all implementations (e.g., preemptive, priority-based scheduling), some only apply to smaller systems (e.g., a single address space), and some only to more full-featured systems (e.g., network support, self-hosting).

Basic Realtime Multitasking and Synchronization

- Multiple flows of control
- Preemptive priority scheduling of flows of control
- One address space for all flows of control
- Direct control of location of memory areas
- Inter-thread communications mechanism via message passing (queues)
- Binary and counting semaphores, without priority inheritance
- Mutual exclusion, with optional priority inheritance
- Local or global event flags (one thread awaits multiple things)
- Multiple memory areas, with both fixed- and variable-sized blocks allocation policies
- System time in units of clock ticks
- Timeouts on all blocking services in units of clock ticks

---

1) **pSOS** *is a registered trademark of Integrated Systems, Inc.*

2) **VRTX32** *is a registered trademark of Micro Research, Inc.*

3) **VxWorks** *is a registered trademark of Wind River Systems, Inc.*

— Hardware interrupt control and support for user interrupt handlers

— Signals

— Exception handling

— Minimal synchronous I/O interface: *open*(), *close*(), *read*(), *write*(), *ioctl*()

— Debugger interface

— No memory protection

— Application runs in privileged (supervisor) mode, if applicable

— Direct I/O, rather than via kernel

— System executable size and memory requirements are major constraints

I/O

Realtime systems supporting I/O generally provide the following features:

— Named I/O devices

— Support for serial I/O lines

— Pipes

— Installable user device drivers

— Memory mapped I/O

Local File System

Realtime systems supporting a file system generally provide the following features:

— Named files

— Hierarchical filesystem (directories)

— Contiguous preallocation of disk space

— May provide media compatibility with another filesystem (e.g. **MS-DOS**™[4] or **RT-11**™[5])

— No user IDs or file protection

Historically, filesystems for embedded realtime systems typically have had a one-level name space, contiguous allocation of disk space, and relatively short filenames. They have not supported an arbitrary hierarchy of named directories, non-contiguous allocation of disk space, or long filenames. They may have had numbered directories (e.g. **RSX-11M**™[6]), or only contiguous allocation of disk space (e.g. **RT-11**™).

However, recent commercial offerings have supported multilevel named directories and both contiguous and non-contiguous disk space allocation. In these implementations, the support of these features with potentially non-

4) **MS-DOS** *is a registered trademark of Microsoft Corporation.*

5) **RT-11** *is a registered trademark of Digital Equipment Corporation.*

6) **RSX-11M** *is a registered trademark of Digital Equipment Corporation.*

deterministic performance does not preclude an application from restricting itself to features with deterministic performance. For example, it is still possible to use contiguous files exclusively. Because it is relatively easy to implement both, and need not interfere with deterministic performance, the working group did not make a distinction between realtime and time-sharing filesystems in this AEP.

Traditional implementations of POSIX.1 filesystems employ a disk buffer cache to improve performance by reducing the number of physical media accesses, and by reordering the accesses to take advantage of the characteristics of rotating media. These implementations have not made a distinction between the buffering of data transfers [*read*() and *write*()], and directory operations [*creat*(), *link*(), *unlink*(), *mkdir*(), *rmdir*(), *rename*()]. A result of this is that a system crash at an unexpected moment can leave the filesystem in a corrupted state. This situation is usually corrected at the next system reboot by a filesystem checker and recovery program, such as *fsck*(). The checking and correcting of a corrupted filesystem may take a long and variable amount of time to perform, and may require a human operator to monitor its progress. Either of these characteristics would make a filesystem check unacceptable for some embedded realtime applications. It was suggested therefore, that such applications limit their use of directory operations to *safe* times, and that implementations maintain the filesystem in such a way that a filesystem check during reboot is avoided. This was considered, but rejected on the grounds that not all applications would require the capability, and that it was neither specifiable nor testable.

Network Communication

Realtime systems supporting networking generally provide the following features:

— Compatibility with a protocol stack (e.g. TCP/IP)

— May support applications such as FTP, TELNET, TFTP, rcp

Distributed File System

Realtime systems supporting a distributed (non-local) file system generally provide the following features:

— Remote access to a filesystem

— Performance not realtime

Memory Protection

Realtime systems supporting memory protection (typically requiring a memory management unit) generally provide the following features:

— Memory mapping and protection

— Ability to map to special areas of memory (I/O page, frame buffer)

— Typically do not have demand paging for realtime parts

Multiprocessor Support

Realtime systems supporting multiprocessing generally provide one of the following methods:

— **network**
Non-transparent access to remote objects, RPC

— **distributed**
Transparent access to objects, no load-balancing

— **symmetric**
Presence of a global task scheduling queue (may also have local scheduling queues)

Self-Hosting
Realtime systems supporting the capability for program development, text editing, compilation, etc. generally provide the following features:

— Shell

— Text editor

— Compiler, assembler, linker, debugger

— May have user ID protection

Related Standards Activities

Activities to extend this family of POSIX standards 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:[7)]

(1) Language-independent service descriptions of ISO/IEC 9945-1:1996.

(2) C, Ada, and FORTRAN Language bindings to (1).

(3) Shell and Utility facilities.

(4) Verification testing methods.

(5) Additional realtime facilities.

(6) Secure/Trusted System considerations.

(7) Network interface facilities.

(8) System Administration.

(9) Graphical User Interfaces.

(10) Profiles describing application- or user-specific combinations of open systems standards for supercomputing, multiprocessor, and batch extensions; and multiuser systems based on historical models.

(11) An overall guide to POSIX-based or related open systems standards and profiles.

---

7) 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; Telephone: +1 202 371-0101; FAX: +1 202 728-9614. Working drafts of POSIX standards under development are available from the Institute of Electrical and Electronics Engineers, 445 Hoes Lane, P.O. Box 1331, Piscataway, NJ 08855-1331 (http://www.standards.ieee.org/).

Extensions are approved as "amendments" or "revisions" to a base document, following the IEEE and ISO/IEC Procedures. This standard will be extended to accomodate those extensions.

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

If you have 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

and ask to have this forwarded to the chairperson of the appropriate PASC working group. If you have interest in participating in this work at the international level, contact your ISO/IEC national body.

ISO/IEC ISP 15287-2 was prepared by the Systems Services Working Group—Realtime, sponsored by the Portable Applications Standards Committee of the IEEE Computer Society. When the IEEE Standards Board approved this standard on March 19, 1998, the membership of the System Services Working Group was as follows:

Portable Applications Standards Committee
(PASC)

Chair:	Lowell Johnson
Vice Chair:	Joseph M. Gwinn
Functional Chairs:	Jay Ashford
	Andrew Josey
	Curtis Royster
Secretary:	Nick Stoughton

IEEE System Services Working Group — Realtime

Chair:	Joseph M. Gwinn
	Susan Corwin (to 1995)
Secretary:	Karen D. Gordon
	Frank Prindle (1996)
	Lee Schermerhorn (to 1994)
Editor:	Bob Luken
Technical Reviewers:	Andrew E. Wheeler, Jr.
	Joseph M. Gwinn
	Karen D. Gordon
Ballot Coordinators:	Andrew E. Wheeler, Jr.
	James T. Oblinger

Working Group

Ray Alderman
Larry Anderson
Pierre-Jean Arcos
Charles R. Arnold
V. Raj Avula
Theodore P. Baker
Todd Bargorek
Robert Barned
Richard M. Bergman
Nawaf Bitar
Steve Brosky
David Butenhof
Hans Petter Christiansen
Dave Cooper
Susan Corwin
Bill Cox
June R. Curtis
Peter Dibble
Christoph Eck
Maryland R. Edwards
Michael Feustel
Bill Gallmeister
Michael González
Karen D. Gordon
Randy Greene
Rick Greer
Joseph M. Gwinn
Steven A. Haaser
Barbara Haleen
Geoffrey R. Hall
Patrick Hebert
Mary R. Hermann
David Hughes
Duane Hughes
Michael B. Jones
Steven Kleiman
Robert Knighten
C. Douglass Locke
Kent Long
Robert D. Luken
Dave Lunger
Bill Maes
James T. Oblinger
Offer Pazy
Carolyn Petersen
Dave Plauger
Arlan Pool
Franklin C. Prindle
Francois Riche
Robert Rose
Gordon W. Ross
Barry Ruzek
Webb Scales
Lee Schermerhorn
Del Swanson
Barry Traylor
Stephen R. Walli
Andrew E. Wheeler, Jr.
David Wilner
John Zolnowsky

The following persons were members of the 1003.13 Balloting Group that approved the standard for submission to the IEEE Standards Board:

Norman Aaronson
Alejandro Alonso-Muñoz
Pierre-Jean Arcos
Charles R. Arnold
Theodore P. Baker
Robert Barned
Jason Behm
Richard M. Bergman
Andy R. Bihain
Shirley Bockstahler-Brandt
Steve Case
Hu Cheng
Hans Pietter Christiansen
Susan Corwin
Donald Cragun
June R. Curtis
Lee Damico
Christoph Eck
James A. Eiler
Philip H. Enslow
Donna K. Fisher
Michel Gien
John Gilbert
Michael González
Karen D. Gordon
Mars J. Gralia
Randy Greene
Joseph M. Gwinn
Steven A. Haaser
Geoffrey R. Hall
Patrick Hebert
Hans H. Heilborn
Duane Hughes
Hal Jespersen
Michael B. Jones
Joe Kelsey
Judy Kerner
Lawrence J. Kilgallen
Martin J. Kirk
Thomas M. Kurihara
Kevin Lewis
C. Douglass Locke
Kent Long
James P. Lonjers
Lee W. Lucas
Robert D. Luken
Dave Lunger
Marshall McKusick
Craig B. Meyers
Diana Norwood
James T. Oblinger
Dave Plauger
Arlan Pool
Franklin C. Prindle
Paul Rabin
Wendy Rauch
Henry H. Robbins
Steven Schwarm
Del Swanson
Sandra Swearingen
James G. Tanner
Mark-Rene Uchida
Andrew E. Wheeler, Jr.
David Wilner
Oren Yuen
John J. Zenor
John Zolnowsky

When the IEEE-SA Standards Board approved this standard on 19 March 1998, it had the following membership:

Richard J. Holleman, *Chair*
Donald N. Heirman, *Vice Chair*
Judith Gorman, *Secretary*

Satish K. Aggarwal
Clyde R. Camp
James T. Carlo
Gary R. Engmann
Harold E. Epstein
Jay Forster*
Thomas F. Garrity
Ruben D. Garzon
James H. Gurney
Jim D. Isaak
Lowell G. Johnson
Robert Kennelly
E.G. "Al" Kiener
Joseph L. Koepfinger*
Stephen R. Lambert
Jim Logothetis
Donald C. Loughry
L. Bruce McClung
Louis-François Pau
Ronald C. Petersen
Gerald H. Peterson
John B. Posey
Gary S. Robinson
Hans E. Weinrich
Donald W. Zispe

*Member emeritus

Noelle Humenick
IEEE Standards Project Editor

IEEE Information Technology — Standardized Application Environment Profile — POSIX® Realtime Application Support (AEP)

Section 1: General

1.1 Overview

This standard establishes a set of Realtime Environment Profiles based on ISO/IEC 9945-1:1990 {3} as amended by IEEE Std 1003.1b-1993 {5}, IEEE Std 1003.5b-1996 {9}, and ISO/IEC 9945-2:1993 {4} (POSIX) work and related standards specifying foundations for realtime applications.

The Application Environment Profiles specified herein are appropriate for the development and execution of realtime applications using the services and utilities provided by standards called out in this document.

1.2 Taxonomy Position

Editor's Note: The following taxonomy is consistent with the 8-25-1993 draft of TR10000-3.

- P — OSE Profiles
  - AEP — Application Environment Profiles
    - PS — System Profiles
      - PSE — Generic Environment Profiles
        - PSE5 — Realtime Environments
          - PSE51 — Minimal Realtime System Profile
          - PSE52 — Realtime Controller System Profile

PSE53 — Dedicated Realtime System Profile

PSE54 — Multi-Purpose Realtime System Profile

1.2.1 Rationale for Positioning (informative)

(This subclause is not a part of ISO/IEC ISP 15287-2: 2000)

This document contains requirements for Application Program Interfaces and Units of Functionality necessary to support four instances of the Generic Realtime Environment class of applications. It specifies the behavior to be observed at the interfaces of the Application Platform on which the class of applications can run. This subset of an OSE profile is complete and coherent within the context of the class of applications supported. As such, it is a System Profile class of Application Environment Profile (AEP).

1.3 Realtime System Profiles

This document describes four realtime profiles and their minimum hardware requirements.

1.3.1 Minimal Realtime System Profile

These systems are typically embedded in systems dedicated to unattended control of one or more special I/O devices. Neither user interaction nor a file system (mass storage) is required. The programming model is that of a single POSIX process (corresponding to the processor's hardware address space) containing one or more threads of control (POSIX.1c threads or Ada tasks). Although there is only one process, a Message Passing interface is provided for communications among threads of control and between PSE5X instantiations. Special devices are operated and controlled either by memory-mapped I/O or by the basic I/O interface, which provides a standard way to access the intrinsically nonstandard I/O hardware and its nonportable control code.

The hardware model for this profile assumes a single processor with its memory, but no memory management unit (MMU) or common I/O devices are required. (If there are in fact multiple processors, typically there are multiple instantiations of the operating system, perhaps communicating via shared memory or a backplane channel, perhaps isolated).

1.3.2 Realtime Controller System Profile

These systems are an extension of the Minimal Realtime System Profile. Support for a file system interface and asynchronous (non-blocking) I/O interfaces has been added.

The hardware model for this profile assumes a single processor and memory (MMU is not required). Mass storage devices are not required; the file system may, for instance, be implemented in memory (RAM disk).