Copyright © 1991 by the
Institute of Electrical and Electronics Engineers, Inc.
345 East 47th Street
New York, NY 10017, USA
All rights reserved as an unpublished work.

This is an unapproved and unpublished IEEE Standards Draft, subject to change. The publication, distribution, or copying of this draft, as well as all derivative works based on this draft, is expressly prohibited except as set forth below.

Permission is hereby granted for IEEE Standards Committee participants to reproduce this document for purposes of IEEE standardization activities only, and subject to the restrictions contained herein.

Permission is hereby also granted for member bodies and technical committees of ISO and IEC to reproduce this document for purposes of developing a national position, subject to the restrictions contained herein.

Permission is hereby also granted to the preceding entities to make limited copies of this document in an electronic form only for the stated activities.

The following restrictions apply to reproducing or transmitting the document in any form: 1) all copies or portions thereof must identify the document's IEEE project number and draft number, and must be accompanied by this entire notice in a prominent location; 2) no portion of this document may be redistributed in any modified or abridged form without the prior approval of the IEEE Standards Department.

Other entities seeking permission to reproduce this document, or any portion thereof, for standardization or other activities, must contact the IEEE Standards Department for the appropriate license.

Use of information contained in this unapproved draft is at your own risk.

IEEE Standards Department
Copyright and Permissions
445 Hoes Lane, P.O. Box 1331
Piscataway, NJ 08855-1331, USA
+1 (908) 562-3800
+1 (908) 562-1571 [FAX]

3.5 POSIX Open System Environment Profiles

The results of Open System specification projects are collected into an expanding set of ''Base Standards,'' addressing a growing subset of functional requirements.

Profile projects then select among these base standards to create a tailored, consistent set of standards addressing a more specific type (or instance) of system or set of application software. Profiles satisfy the requirements of application ''domains'' such as office or industrial automation, transaction processing, or real-time control systems.

This framework provides a way to characterize the functionality of profile activities. The current OSI profiles tend to focus strictly on the communications EEI. Other profiles might focus on a single component or span multiple interface types.

3.6 Application Platform Implementation Considerations

Profile writers need to be aware that in an open system environment, the application platform can be decomposed into independently procurable components. While standards are interface specifications, and as such are independent of implementation, there are aspects of platform implementation or construction that may affect the specification of standards, and that profile writers may want to address.

For each case, the portion of the application platform that implements any particular independently procurable service is described as the service component. Figure 3-6 shows an application platform made up of several service components. If components interact, the specification of the interface between service components within the application platform may be standardized or nonstandard (including proprietary).

An intercomponent interface is labeled in Figure 3-6 as ''System Internal Interface'' because it may be used to assemble an application platform from multiple components. Figure 3-6 shows how a System Internal Interface is shown in the reference model.

A standards-based SII between the application platform service components addresses portability and interoperability of the application platform service components, not portability and interoperability of application software and systems.

Development of an SII would also require a consensus to emerge on the ''best'' design and implementation of system software/hardware. Very little consensus has developed on the partitioning of the platform into components and consequent allocation of function to each. In fact, this aspect of system design has been in a constant and accelerating state of innovation for decades. One of the major objectives of the API is to provide a more stable interface that decouples application software from the constantly changing platform. This enables the migration of application software to platforms based on constantly upgraded technology. (See 3.1 ''Accommodation of New Information System Technology''.)

Copyright © 1991 IEEE. All rights reserved.
This is an unapproved IEEE Standards Draft, subject to change.