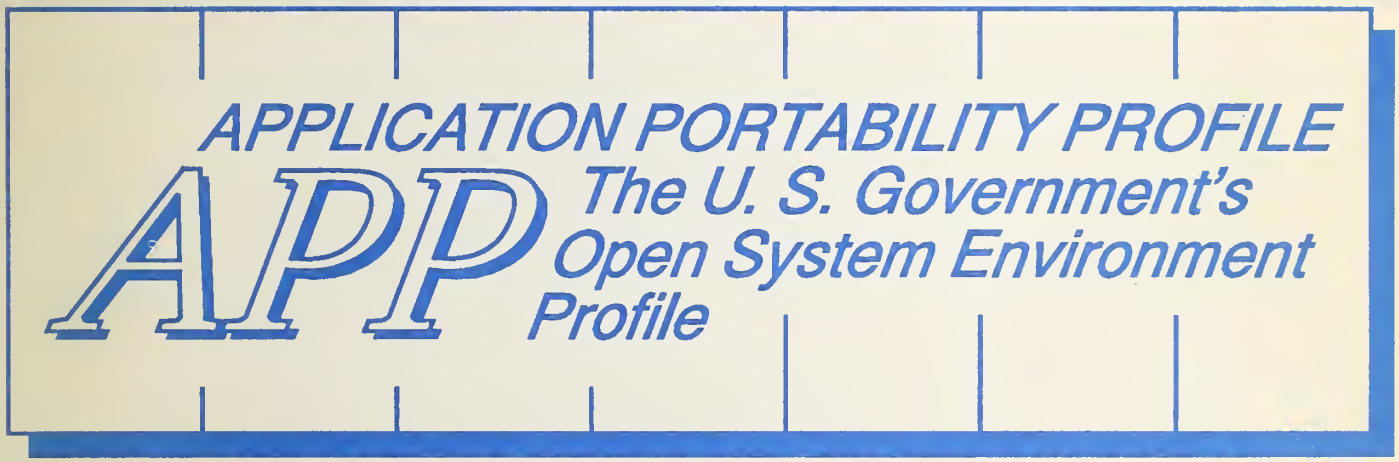
NIST Special Publication 500-187

Computer Systems Technology

U.S. DEPARTMENT OF COMMERCE
National Institute of Standards and Technology

APPLICATION PORTABILITY PROFILE (APP) The U.S. Government's Open System Environment Profile OSE/1 Version 1.0

Application Software



External Environment

NIST Special Publication 500-187

APPLICATION PORTABILITY PROFILE (APP) The U.S. Government's Open System Environment Profile OSE/1 Version 1.0

Computer Systems Laboratory
National Institute of Standards and Technology
Gaithersburg, MD 20899

April 1991

U.S. DEPARTMENT OF COMMERCE
Robert A. Mosbacher, Secretary
NATIONAL INSTITUTE OF STANDARDS
AND TECHNOLOGY
John W. Lyons, Director

Reports on Computer Systems Technology

The National Institute of Standards and Technology (NIST) has a unique responsibility for computer systems technology within the Federal government. NIST's Computer Systems Laboratory (CSL) develops standards and guidelines, provides technical assistance, and conducts research for computers and related telecommunications systems to achieve more effective utilization of Federal information technology resources. CSL's responsibilities include development of technical, management, physical, and administrative standards and guidelines for the cost-effective security and privacy of sensitive unclassified information processed in Federal computers. CSL assists agencies in developing security plans and in improving computer security awareness training. This Special Publication 500 series reports CSL research and guidelines to Federal agencies as well as to organizations in industry, government, and academia.

National Institute of Standards and Technology Special Publication 500-187
Natl. Inst. Stand. Technol. Spec. Publ. 500-187, 74 pages (Apr. 1991)
CODEN: NSPUE2

U.S. GOVERNMENT PRINTING OFFICE
WASHINGTON: 1991

For sale by the Superintendent of Documents, U.S. Government Printing Office, Washington, DC 20402

PREFACE

Federal information systems technology is migrating toward open system environments that consist of heterogeneous networked systems, databases, and hardware. An integral part of open system environments is adherence to consensus-based specifications. The focus of this guide is on Open System Environments (OSE) and the U. S. Government's Application Portability Profile (APP). The APP integrates federal, national, international, and other specifications to provide the functionality necessary to accommodate the broad range of Federal information technology requirements.

The APP is not a standard. At least one procurement has specified that proposed systems "shall conform to the APP." This will not suffice since the APP is not designed to cover every case. In some instances, the selection of one specification recommended in the APP will obviate the need for other specifications that are also recommended, but for somewhat different user requirements. In areas where the APP does not meet all of a user's requirements, users must augment the recommended specifications to ensure that proposed systems meet their requirements. This report is designed to help users determine which specifications to use.

The guidance is intended to assist Federal agencies in making informed choices regarding the selection and use of OSE specifications, and in the development of application profiles based on the APP. It is directed toward managers and project leaders who have the responsibilities of procuring, developing, and maintaining information systems supported by heterogeneous application platforms.

The mention of specification names in certain instances should not be interpreted to mean that the National Institute of Standards and Technology endorses the procurement of any specific products based on these specifications. NIST has endeavored to separate references to the specifications from products and services, and has provided evaluation criteria, where applicable, to enable users to make their own judgements of the applicability of the recommended specifications to their requirements. In each recommendation, the evaluations provided by NIST indicate that no other specification currently exceeds the general applicability of the recommended one. For specific individual and organizational requirements, other specifications not mentioned here may be more applicable.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ABSTRACT

An Open System Environment (OSE) encompasses the functionality needed to provide interoperability, portability, and scalability of computerized applications across networks of heterogeneous hardware/software platforms. The OSE forms an extensible framework that allows interfaces, services, protocols, and supporting data formats to be defined in terms of nonproprietary specifications that evolve through open (public), consensus-based forums. A selected suite of specifications that define these interfaces, services, protocols, and data formats for a particular class or domain of applications is called a profile. The Application Portability Profile (APP) is the U. S. Government's OSE profile. It was developed to provide functionality across a broad range of Federal applications. This report describes the service areas and components included in the APP and provides evaluations of recommended specifications for the majority of the service area components. Organizations should use this report to assist in determining which specifications may be applicable to their particular environments.

1. INTRODUCTION

Federal agencies are under increasing pressure to use information technology to improve efficiency and delivery of services to the public. At the same time Federal agencies are struggling with finding ways to use information technology to improve efficiency and service delivery, there is a new reality that is becoming increasingly evident. Key aspects of this new reality are that Federal agencies—

- now recognize that they no longer can create de facto standards and enforce them on the commercial market as they were able to do with the early standards;
- must rely on the commercial market for information technology products and services; and
- must establish strategies and plans for acquiring information technology products and services based upon open system standards that support applications software portability and interoperability.

The climate within Federal agencies is changing. Whereas before there were isolated islands of computing, now there is interdependence of users across the entire organization. This interdependence has served to highlight enterprise-wide needs for common application architectures, communication networks, and databases. This interdependence has also raised concerns about computer security issues and the need to address those issues from policy, management, and technical perspectives.

One of the biggest factors underlying the changing federal climate is that federal and nonfederal users now recognize that no single vendor can supply all of their needs for information technology systems and services. Since homogeneity is no longer practical, users need open systems that

provide interoperability of products and portability of people, data, and applications across heterogenous computing environments.

The need to improve portability and interoperability has resulted in widespread interest in standards such as POSIX (Portable Operating System Interface for Computer Environments) and GOSIP (Government Open Systems Interconnection Profile). While these are important milestones in the effort to achieve portability and interoperability, POSIX and GOSIP are not sufficient to address the full spectrum of needs, even in their range of application.

The focus of this guide is on open system environments (OSE) which integrate POSIX with GOSIP and provide the additional functionality necessary to accommodate the broad range of federal requirements. The guidance is intended to assist Federal agencies in making informed choices regarding the selection and use of OSE specifications, and in the development of application/organizational profiles. This guidance is directed toward managers and project leaders who have the responsibilities of procuring, developing, and maintaining information systems supported by heterogeneous hardware/software platforms.

Ideally, specifications would be expressed in terms of international standards. Unfortunately, there are areas of OSE functionality for which formal standards, much less international standards, do not exist. Although this situation will be rectified over time, users who have requirements for those functions are faced with the question, "What specifications should I use now?".

This document is directed toward assisting users in making an informed judgement regarding the choice of specifications to meet current requirements, particularly in those areas where formal standards do not exist. There are two dimensions of the assistance provided. First of all, specifications are provided for each functional area of the APP. The specifications represent the collective judgement of the National Institute of Standards and Technology (NIST) staff regarding the most appropriate specification for each functional area. Second, and equally as important, evaluation criteria to assist in making a qualitative assessment of the recommended specifications are defined and applied. Application of these evaluation criteria provides the NIST assessment of the quality of the specifications recommended.

Users should use the evaluation criteria to make their own assessments of the recommended specifications. Further, users should consider assigning weighted values to elements of the criteria based on their judgement of the relative importance of each element. Users should also consider requiring vendors to use the evaluation criteria to assess specifications that they choose to propose as an alternative to the specifications recommended in this document.

The following sections briefly describe the meaning of open system environment (OSE), the OSE reference model, and specific components of the Application Portability Profile. Later sections provide recommended specifications for specific APP components. They are identified by the component name followed by the title of the recommended specification. Toward the end of this report, we have included references for further information and addresses of organizations that distribute documents on the recommended specifications.

1.1 Open System Environment

An Open System Environment (OSE) is a conceptual framework that provides a context for user requirements and standards specification. It provides a set of information system building blocks with associated interfaces, services, protocols, and data formats. OSE is a key aspect of a worldwide movement in which the U. S. Government and other information intensive organizations are working to—

- protect their investments in application software;
- reduce dependence on single sources of supply for information technology products and services;
- stimulate the availability and quality of open system products in the commercial marketplace; and
- provide a stable base for the evolutionary development of large, complex systems.

The OSE movement is an outgrowth of efforts to establish the groundwork for computing environments that—

- are based on an architectural framework that allows an extensible collection of capabilities to be defined;
- define their capabilities in terms of nonproprietary specifications that are available to any vendor for use in developing commercial products; and
- evolve through an open (public), consensus-based process for defining, specifying, and coordinating standards related to the computing environment.

Computing environments having the above characteristics are referred to as "open". The original developers of the concept of open computing were concerned primarily with interoperability of computers communicating over a network. The technology that resulted from their work is commonly referred to as Open System Interconnection (OSI).

An OSE extends the OSI concept to the broader problems of applications portability and interoperability. Efforts are currently underway by both vendors and users to—

- establish an architectural framework for an OSE;
- define OSE interfaces, protocols, services, and supporting formats; and
- provide a forum for consensus-based agreements on OSE issues.

Although the OSE concept is relatively new, it has matured to the status of an emerging international consensus on the functionality (i. e., the collection of interfaces, protocols, services,

and supporting formats) that should be included in an OSE (see Guide to the POSIX Open Systems Environment, Draft 11, IEEE Working Group P1003.0).

1.2 OSE Reference Model

A reference model is a generally accepted representation of a particular application domain. It allows people who are interested in that domain to agree on definitions and build a fundamental understanding within the scope of the model. (Note: Other reference models are also described in this report. How they relate to one another will be covered in a future version of the APP Guide, or in a companion report.) An OSE reference model is necessary to establish a context for understanding how the disparate technologies required as part of an OSE relate to each other, and to provide a mechanism for identifying the key issues associated with applications portability and interoperability. The development and acceptance of an OSE reference model is a critical success factor for any meaningful evolution of an OSE beyond the current stage.

The OSE reference model must allow consideration of OSE issues from five fundamental, mutually supportive perspectives:

- **Management policy issues** are pervasive across the OSE. These are the fundamental issues of OSE management (e. g., operational control, maintenance, service quality, etc.), and are supported by application software as the tools of the management process.

- **User interface issues** are associated with how applications within an OSE are delivered to the user, and the definition of a consistent look-and-feel for the dialogue between the human user and the application platform. User perspective embraces all aspects of human/computer interaction (e. g., window style guide, character representation, internationalization, commands, and input devices).

- **System issues** relate to how services provided by the application platform are delivered to application programs.

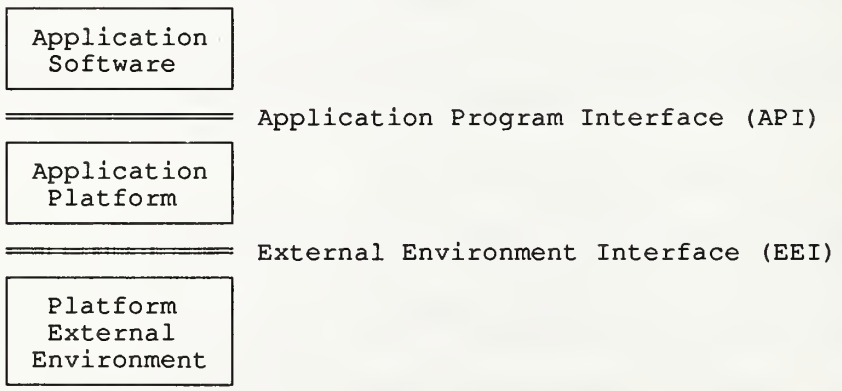


Figure 1. Open System Environment Reference Model.

- **Information interchange** issues deal with formats and related attributes required to support information interchange among application programs.

- **Communications issues** focus on the functions required to handle a wide range of information interchange needs through basic network services and associated transfer syntax.

The reference model shown in figure 1 is expected to achieve international acceptance as the OSE reference model. It serves as the framework for the NIST effort to evolve the APP during the next decade. Two types of elements are used in the model: entities and interfaces.

1.2.1 Model Entities

Figure 2 expands figure 1 to illustrate the components in the (1) application software, (2) application platform, and (3) platform external environment entities. These three classes of OSE reference model entities are described in the following:

- **Application Software** — Most users consider application software to be the computing element supporting their particular business needs (e. g., the payroll, accounting, spreadsheets, and other systems that provide information to the users in the course of conducting business). The application software includes data, documentation, and training, as well as programs.

- **Application Platform** — The application platform is composed of the collection of hardware and software components that provide the services used by application programs. Application platforms facilitate portable application programs through services accessed by application programming interfaces (API) that make the specific characteristics of the platform transparent to the application. The application platform components include the hardware and the software that interfaces directly with the hardware (i. e., the hardware drivers).

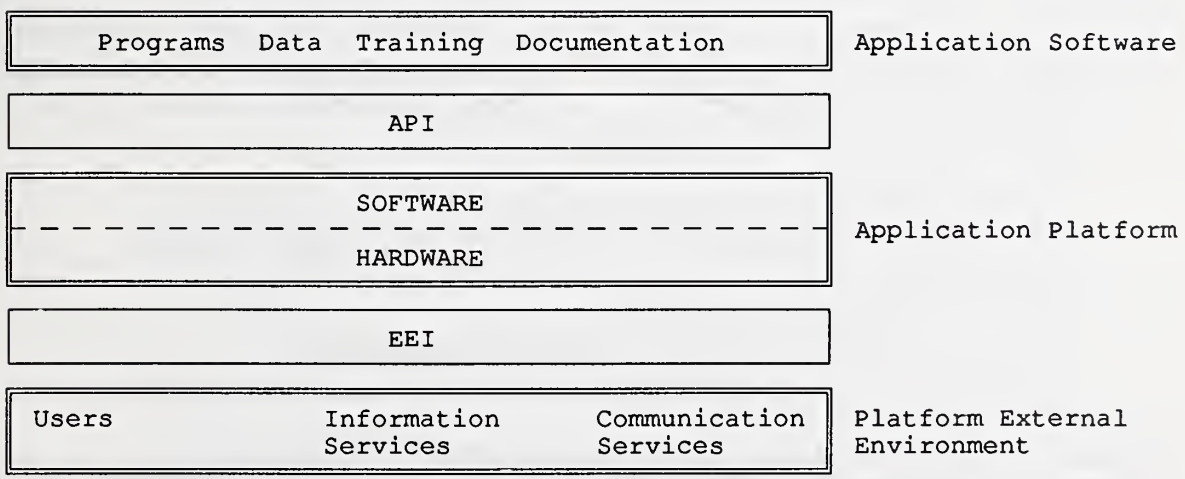


Figure 2. OSE Reference Model Entities.

- **Platform External Environment** — The platform external environment consists of those system elements which are external to the application program and the application platform (e. g., systems executing on other platforms).

1.2.2 Model Interfaces

There are two classes of interfaces in the OSE reference model as described in the following paragraphs.

- **Application Program Interface (API)** — The API is the interface, or set of functions, between the application software and the application platform. An API is categorized according to the types of service accessible via that API. There are four types of API services:

  - User Interface Services
  - Information Interchange Services
  - Communications Services
  - Internal System Services

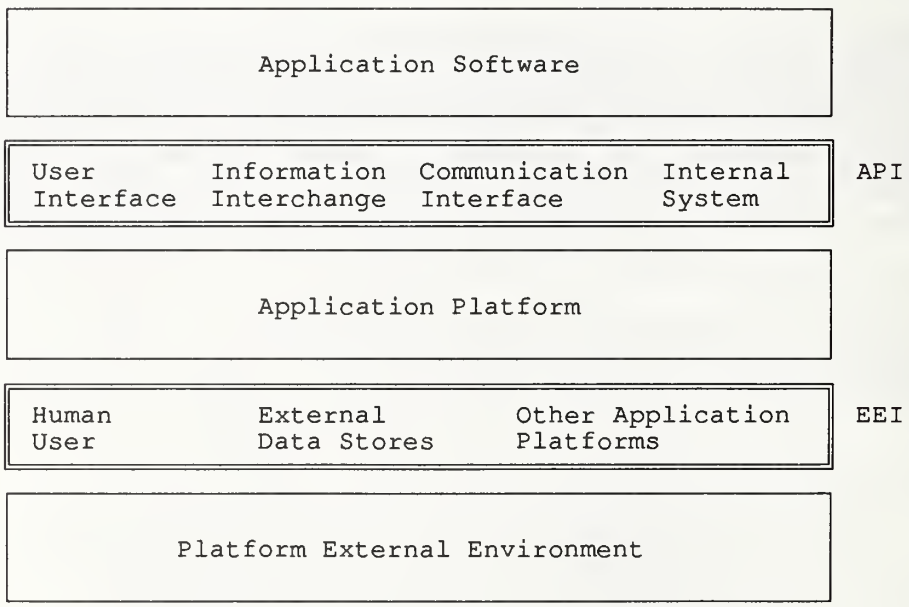


Figure 3. OSE Reference Model Interfaces.

- **External Environment Interface (EEI)** — The EEI is the interface which supports information transfer between the application platform and the external environment. An EEI

is categorized according to the type of information transfer services provided. There are three types of information transfer services. These are transfer services to and from:

- Human users
- External data stores
- Other application platforms

The two classes of interfaces are represented and partitioned in figure 3. Note that APIs primarily support portability while EEIs primarily support interoperability.

1.3 Application Portability Profile

An OSE profile is a suite of open system specifications including options and parameters needed to support the requirements of a specific domain of applications. An OSE profile typically reflects—

- the functions that are required by the applications of interest;
- the organization's view of the viability of a particular specification for migration to an international standard when that standard is established; and
- the availability of commercial off-the-shelf products that conform to the specifications.

The Application Portability Profile (APP) is an OSE profile developed for use by the U. S. Government. The OSE functions included as part of the APP are those that have been identified as important to a broad spectrum of Federal agencies. The APP is defined in terms of open system specifications organized into major service areas. These service areas, along with examples of specific services in each area, are defined in following sections.

The APP provides a framework for a Federal agency to develop individual application profiles that reflect specific needs. Application profiles are useful to gain an understanding of open system requirements and to identify targets of opportunity for implementing open system standards. The APP and application profiles play complementary roles in federal information technology plans.

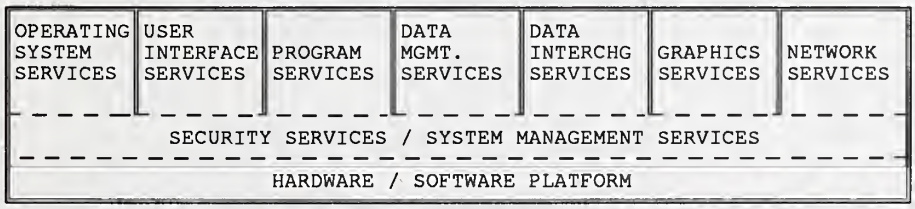


Figure 4. Application Platform and Service Areas.

The application platform entity from figure 3 is shown again in figure 4, but with the APP service areas identified. The broken lines below the service areas indicate that each service area is composed of more than just the specific services noted by name. They also include supporting services, such as security and system management, that cut across and are integrated in each of the other service areas.

1.4 APP Services

APP service areas provide the support necessary, *from an application's perspective*, for a broad range of applications within the U. S. Government. Figure 5 illustrates this perspective. It is up to individual organizations to determine which of these service areas are necessary to support their particular mix of applications.

Each of these service areas addresses specific components around which interface specifications have been or will be defined. Two service components, security services and system management services, are common to all of the other service areas, and pervade these areas in one or more forms. Currently, specifications for security can be recommended in operating system services, network services, and access control and integrity constraints in data management services. Specifications for security in the other service areas are not sufficiently advanced to warrant inclusion at this time.

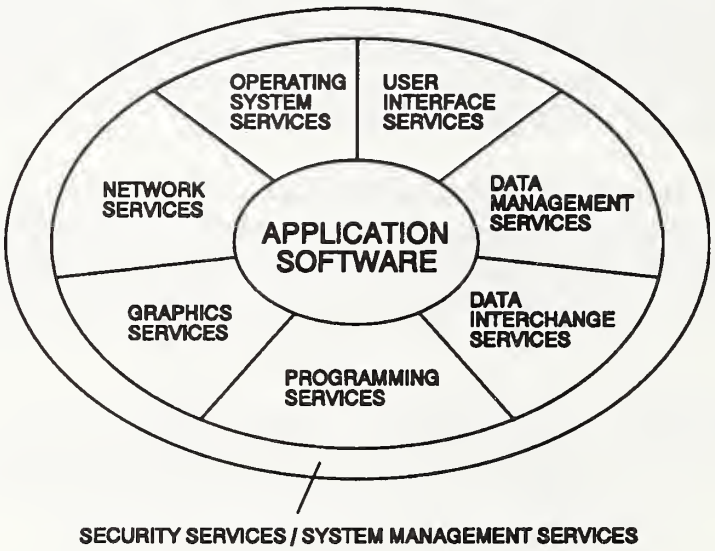


Figure 5. APP Services from an Application's Perspective.

System management services are partially defined. They are based on the Open System Interconnection (OSI) network management framework which applies mainly to networks and

individual nodes on networks. There is, however, an overlap among certain types of network management functions and individual system management functions. This overlapping area applies equally to networks and individual systems and forms the basis for the OSI approach to systems and network management. Other system management functions in the typical operating system sense (e. g., user profiles, resource administration, etc.) will be added over time. As these specifications mature and stabilize, they will be reviewed and appropriate ones may be selected for use in the APP.

1.4.1 Operating System Services

Operating system services are the core services needed to operate and administer the application platform and provide an interface between application software and the platform.

- Kernel operations provide low level services necessary to create and manage processes, execute programs, define and communicate signals, define and process system clock operations, manage files and directories, and control input-output processing to and from peripheral devices.

- Commands and utilities include mechanisms for operations at the operator level, such as comparing, printing, and displaying file contents, editing files, pattern searching, evaluating expressions, logging messages, moving files between directories, sorting data, executing command scripts, scheduling signal execution processes, and accessing environment information.

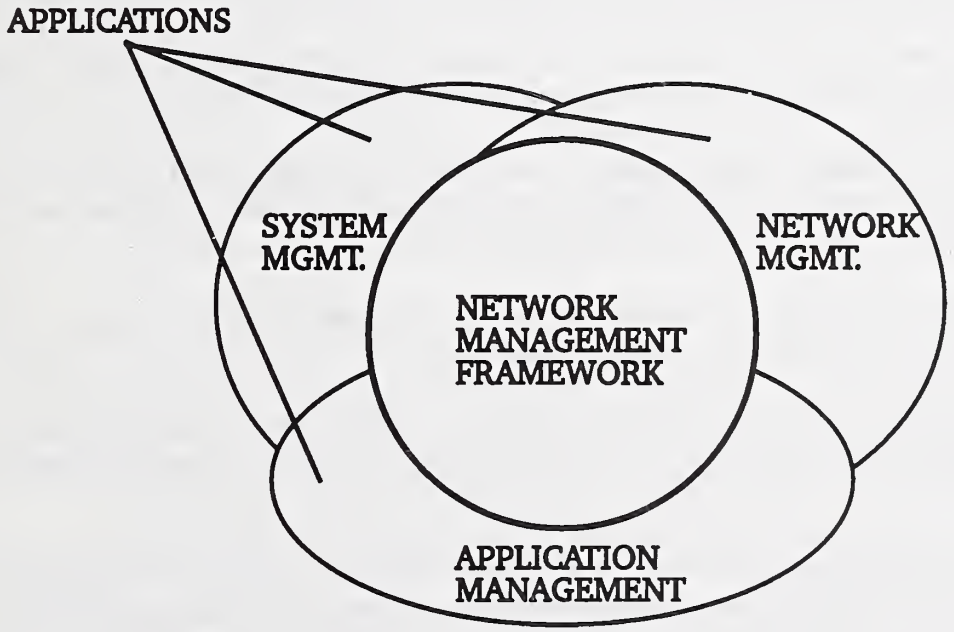


Figure 6. OSI Network Management Framework.

- System management includes capabilities to define and manage user access, devices, file systems, administrative processes (job accounting), queues, machine/platform profiles, authorization of resource usage, and system backup.

- Operating system security services are specified in terms of controlling the access of users and processes representing users to data, functions, hardware, and software resources of a system.

Of particular interest is the area of system management. System management affects not only operating system services, but also network services, and will eventually affect all of the services as specifications are defined. A model of how system management relates to both service areas is illustrated in figure 6. The area entitled "OSI Network Management Framework" encompasses services that are common to both operating system and network administration. Any specifications applying to either area should require specifications based on this central framework. The specification recommended in the APP is the same for both areas. Additional functionality specific to operating system management and network administration where they do not overlap will become available when consensus is achieved.

1.4.2 User Interface Services

User interface services define how users may interact with an application. Depending on the capabilities required by users and the applications, these interfaces may include the following:

- Client-server operations define the relationships between client and server processes operating within a network, in particular, graphical user interface display processes. In this case, the program that controls each display unit is a server process, while independent user programs are client processes that request display services from the server.

- Object definition and management includes specifications which define characteristics of display elements: color, shape, size, movement, graphics context, user preferences, interactions among display elements, etc.

- Window management specifications define how windows are created, moved, stored, retrieved, removed, and related to each other.

- Dialogue support includes specifications that define the relationships between what is displayed on the screen (e. g., cursor movements, keyboard data entry, external data entry devices), and how the display changes depending on the data entered.

- User interface security services include the definition and execution of types of user access to objects within the purview of user interface systems such as access to windows, menus, etc.; and the functions that provide user interface services such as user interface management systems. (APP specifications for this service are currently not available.)

User interfaces are often the most complex part of system development and maintenance. Within the past few years, significant advances have been made in user interfaces, in both ease-of-use and in reducing the development effort required.

The principal components of a window system are a CRT interface that contains one or more windows or panels; a pointing device such as a mouse or touch screen; and a set of objects on the screen that can be directly manipulated by the user through the pointing device or through keyboard responses. The specifications define interfaces between service components taken from the User Interface System Reference Model (UISRM) developed by NIST.

The User Interface System Reference Model (figure 7) is a representation of the components of a window system defined in terms of layers. Systems based on this reference model will have one or more layers represented in the model. In particular, many applications are built directly on the Toolkit layer, with no Dialogue/Presentation layer support from a user interface management system (UIMS). In some systems the Toolkit, Subroutine Foundation, and/or Data Stream Interface layers may be combined.

The layers are arranged from bottom to top in ascending complexity. The bottom layers contain primitive constructs on which the upper layer functions are built. Each layer is described individually as follows:

- Layer 0: Data Stream Encoding defines the format and sequencing of byte streams passed between client processes that require services and server processes on the same or separate platform that provide the services required. The Data Stream Encoding is the "wire" or "network" protocol. As a specification of message formats, the Data Stream Encoding is independent of operating system, programming language, or network communication.

- Layer 1: The Data Stream Interface specifies a function call interface to build the messages defined in the Data Stream Encoding layer. In programs, this interface is a function library. The Data Stream Interface converts parameters passed from a program into the bit stream that is transmitted over the network, and converts messages from the server into values passed to the program. The Data Stream Interface provides access to basic graphic functions from Layer 0, and may support system functions such as error handling and synchronization.

- Layer 2: The Subroutine Foundation uses features of the Data Stream Interface to provide the means to build components of window interfaces, such as scroll bars. Functions often provided by the Subroutine Foundation include initialization and destruction of objects, management of events and object hierarchy, and the saving and restoration of the interface state. The Subroutine Foundation can be thought of as a toolkit with which to build toolkits.

- Layer 3: Toolkit Components such as menus, pushbuttons, scroll bars, or help boxes can be used to build an application interface. These "prefabricated" components make up the Toolkit. The components of Toolkits vary with vendors, but they typically contain most of the common user interface elements.

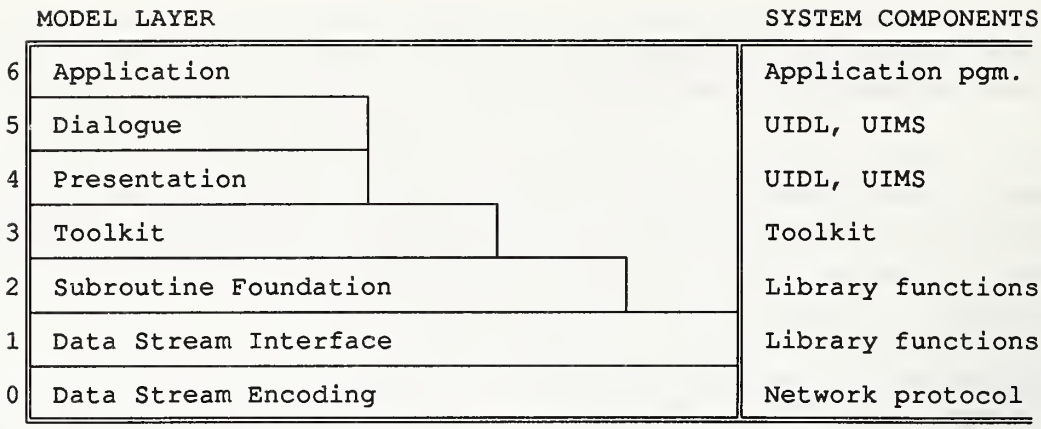


Figure 7. User Interface System Reference Model.

- Layer 4: The Presentation layer determines the appearance of the user interface, including aspects such as size, style, and color. It specifies how the components in the Toolkit should be composed to create windows. The appearance may be specified using a User Interface Definition Language (UIDL) and may be enforced by a window manager, which controls the size and location of windows, and decorates windows in the style specified by the user.

- Layer 5: The Dialogue layer coordinates the interaction between the computer system and the user. It can be thought of as a mediator between the user and the application program. Communication between user and application program is through the Dialogue layer, which may be implemented by a User Interface Management System (UIMS). The user/application interaction is specified by a "dialogue" that associates user actions, such as clicking on a menu item, with application actions. Some UIMS tools can accept a dialogue and a presentation style from which to generate an instance of the UIMS that controls the interaction between user and application.

- Layer 6: The application program implements the functions required by the user. Its interaction with the user is through the Dialogue layer. The application may call routines at the Toolkit, Subroutine Foundation, or Data Stream Interface levels as well, but portability may be reduced.

The services provided in these layers are spread across the user interface portion of the OSE Reference Model in the API and EEI, depending on a specific implementation.

1.4.3 Programming Services

The procedural aspect of an application is embodied in the programming languages used to code it. Additionally, professional system developers require tools appropriate to the development and maintenance of applications. These capabilities are provided by programming services which include the following:

- Programming languages and language bindings for COBOL, Fortran, Ada, C, and Pascal.

- Integrated software engineering environments (ISEE) and tools include systems and programs that assist in the automated development and maintenance of software. These include, but are not limited to, tools for requirements specification and analysis, for design work and analysis, for creating program code, for testing, for documenting, for prototyping, and for group communication. The interfaces among these tools include services for storing and retrieving information about systems and exchanging this information among the various system development environment components. An adjunct to these capabilities is the ability to manage and control the configuration of software components, test data, and libraries.

- Programming security services provide the means to control access to and integrity of programming objects such as libraries, program code, etc., and the tools or information that provide the infrastructure for development of software. (APP specifications for this service are currently not available.)

1.4.4 Data Management Services

Central to most systems is the management of data that can be defined independent of the processes that create or use it, maintained indefinitely, and shared among many processes. Data management services include the following:

- Data dictionary/directory services allow users and programmers to access and modify data about data (i. e., metadata). Such data may include internal and external formats, integrity and security rules, and location within a distributed system.

- Database management system (DBMS) services provide controlled access and modification of structured data. To manage the data, the DBMS provides concurrency control and facilities to combine data from different schemas. DBMS services are accessible through a programming language interface or an interactive/fourth generation language interface. For efficiency, database management systems generally provide specific services to create, populate, move, backup, restore, and archive databases, although some of these services could be provided by general file management capabilities described in operating system services.

- Distributed data services provide access to and modification of data in a remote database.

- Data management security services include control of access to and integrity of data stored in a system through the use of specific mechanisms such as privileges, database views, assertions, user profiles, verification of data content, data labels, etc.

1.4.5 Data Interchange Services

Data interchange services provide specialized support for the interchange of data between applications. These services are designed to handle data interchange between applications on the same platform and applications on different (heterogeneous) platforms.

- Document services include specifications for encoding the data (e. g., text, pictures, numerics, special characters, etc.), and both the logical and visual structures of electronic documents.

- Graphics data services include device independent descriptions of picture elements.

- Product data interchange services encompass those specifications that describe technical drawings, documentation, and other data required for product design and manufacturing, including geometric and nongeometric data such as form features, tolerances, material properties, and surfaces.

- Data interchange security services are used to verify and validate the integrity of specific types of data interchange. Examples of such services include nonrepudiation, encryption, access, data security labeling, etc. (APP specifications for this service are currently not available.)

1.4.6 Graphics Services

Graphics services provide functions required for creating and manipulating pictures. These services include display element definition and management, and graphical object attribute definition and management. These services are defined in specifications for describing multi-dimensional graphic objects in a form that is independent of output devices, and managing hierarchical database structures containing graphics data. Graphics security services included in this area are access to and integrity of functions that support the development of graphics software and graphical data. (APP specifications for graphics security services are currently not available.)

1.4.7 Network Services

Network services are provided to support distributed applications requiring data access and applications interoperability in heterogeneous or homogeneous, networked environments.

- Data communications includes protocols for reliable, transparent, end-to-end data transmission across communications networks.

- Transparent file access to local and remote files.

- Personal/micro computer support for interoperability with systems based on MS-DOS.

- Distributed computing services include specifications for extending the local procedure call to a distributed environment.

- Network security services include access, authentication, confidentiality, integrity, and nonrepudiation controls and management of communications between senders and receivers of information in a network.

2. APP SPECIFICATIONS

Ideally, specifications would be expressed in terms of international standards. Unfortunately, there are areas of OSE functionality for which formal standards, much less international standards, do not exist. Although this situation will be rectified over time, users who have requirements for those functions are faced with the question, "What specifications should I use now?".

In some cases, there are no publicly available specifications that pertain directly to a specific service area component. In those cases, we have tried to recommend a specification that at least partially covers the required functionality. In other cases, we have recommended specifications that are not entirely open, recognizing the fact that users need guidance now. NIST does not advocate that organizations should use the specifications in these cases without knowledge of what adverse effects might be in store (e. g., difficulty in porting applications later in a system's life, justifying the use of nonopen specifications, etc.). If another specification appears to meet an organization's requirements more fully, then we recommend that the organization choose the one that meets those requirements the best. NIST is constrained by the same limitations in selecting specifications as all other organizations, however, for a broad range of federal applications and organizations, we can offer some insight in minimizing problems and managing those that cannot be solved directly at this time.

The following section describes the current recommended specifications for each of the APP services. The information is provided to users to assist them in evaluating these specifications for inclusion in application and/or organizational profiles. The information includes evaluation criteria for each specification. These evaluations may be used to compare specifications listed in this guide to other specifications that an organization may be considering.

Each service area is preceded by a summary status report (see fig. 8) of all specifications reviewed in this report for that service area. Subsections of each service area describe specific evaluation criteria for the specification.

	LOC	PAV	CMP	MAT	STB	DFU	PRL
POSIX.1	●	○	○	●	●	●	

Legend: ● -high evaluation
○ -average evaluation
blank -low evaluation

LOC -- Level of consensus
PAV -- Product availability
CMP -- Completeness
MAT -- Maturity
STB -- Stability
DFU -- De facto usage
PRL -- Problems/limitations

Figure 8. Summary Status Report.

The summary status report relates the results of major evaluation criteria (e. g., level of consensus, completeness, etc.) to a graphic representation. With one view, all of the specifications in a particular service area can be compared to determine relative coverage of the area. Users may use this information to determine where they should concentrate their efforts in tailoring and augmenting application and organizational profiles.

Each of the specifications is evaluated according to how well it meets the requirements of a specific criterion. The criteria are defined as follows:

- Level of consensus—A low evaluation is given to specifications that are proprietary or are used by a very limited or specialized group of users; a high evaluation is given for a specification that has already become an international standard; average evaluations are assigned for public domain specifications that are not standard, or that may be in the process of becoming a standard (i.e., standards committee work-in-progress), or that are widely available across various hardware/software platforms.

- Product availability—A low evaluation is given to specifications for which only a very few proprietary products are available; high evaluations are given to specifications for which there is a wide variety of products available from various vendors across different application platforms; average evaluations are assigned to specifications that may be proprietary but have many products available from a variety of vendors, or that are public domain specifications with products readily available.

- Completeness—A specification is evaluated on the degree to which it defines and covers key features necessary in supporting a specific functional area or service.

- Maturity—According to the underlying technology of a specification, a high evaluation indicates that it is well-understood (e. g., a reference model is well-defined, appropriate concepts of the technology are in widespread use, the technology may have been in use for many years, a formal mathematical model is defined, etc.). A low evaluation indicates that it may be based on technology that has not been well-defined and may be relatively new.

- Stability—A high evaluation means that the specification is very stable, that no changes are expected within the next 2 years. A low evaluation indicates that significant or many changes are expected within a relatively short time, or that incompatibilities exist between current and expected releases of the specification. An average evaluation is given to those specifications that may have changes forthcoming to replace or deprecate features in the existing specifications.

- De facto usage—This evaluation criterion estimates the likelihood that a vendor will independently propose products that conform to this specification whether or not a reference specification is stated in the procurement documents. A high evaluation indicates that most proposed products will conform to the specification. A low evaluation indicates that it is unlikely that the vendor will propose products based on the specifications. An average evaluation indicates that vendors are just as likely to propose products based on the specifications as not (i. e., no clear determination exists). In the cases of low or average evaluations, it is imperative that users include a specification in procurement documentation.

A low evaluation does not necessarily mean that products implemented on the specification do not exist. It can also mean that some vendors would rather provide products that are not based on the recommended specifications.

- Problems/limitations—Lower evaluations are assigned to specifications with severe restrictions on use or capabilities (e. g., licensing restrictions), or known problems tend to be too difficult or too numerous to overcome (e. g., new releases of the specification are not compatible with previous releases, or not enough is covered in the standard to be useful). An average evaluation is given to those specifications that require some minor additional facility in order to be fully effective in their intended environment.

Additional informational items including the following are provided where pertinent:

- Specification available from—Organization from which the specification can be ordered.

- Publication date—Date on which the publication was released for general use (usually designated on the specification's title page.)

- Sponsoring organization—Organization responsible for developing and/or maintaining the specification. (In the case of certain Federal Information Processing Standards [FIPS] that adopt existing national or international standards, the organization responsible for the existing standard is listed.)

- Rationale—In a very few cases, a rationale section has been included to describe the reasoning behind a specific recommendation. The intent of this section is to show that a validation process was undertaken before a recommendation was made.

- Applicability—Description of the OSE service area that covers the recommended specification.

- Conformance testing—Provides information about current and future plans for conformance testing of products based on the recommended specification.

- Future plans—Known directions and long-term plans for individual specifications.

- Alternative specifications—In some instances, other specifications exist besides the recommended specification. Users may want to review these alternatives before selecting a specification on which to standardize.

- Suggested reference—Suggested procurement terminology is included at the end of each specification section.

2.1 Operating System Services

Operating system (OS) services include kernel operations, commands and utilities, system management, and security.

	LOC	PAV	CMP	MAT	STB	DFU	PRL
POSIX.1	●	○	●	●	●	●	●
POSIX.2				●			
GNMP	○						
POSIX.6				○			

LOC -- Level of consensus
PAV -- Product availability
CMP -- Completeness
MAT -- Maturity
STB -- Stability
DFU -- De facto usage
PRL -- Problems/limitations

Legend: ● -high evaluation
○ -average evaluation
blank -low evaluation

2.1.1 Kernel Operations — Portable Operating System Interface for Computer Environments (POSIX.1) FIPS PUB 151-1

Specification available from: National Technical Information Service (NTIS)

Publication date: March 28, 1990

Sponsoring organization: The Institute of Electrical and Electronics Engineers, Inc. (IEEE)

Applicability: Kernel operations provide low level services necessary to create and manage processes, execute programs, define and communicate signals, define and process system clock operations, manage files and directories, and control input-output processing to and from external devices.

Level of consensus: The U. S. Government's Federal Information Processing Standard Publication (FIPS PUB) is based on IEEE Standard 1003.1-1988. The FIPS PUB makes certain optional capabilities mandatory for Federal procurements. ANSI approved IEEE standard 1003.1-1988 on November 10, 1989. IEEE Standard 1003.1-1990 is proposed as an international standard, ISO 9945-1.

Product availability: A rapidly growing number of vendors claim POSIX conformance for their products. POSIX products are being delivered as part of several Federal procurements.

Completeness: The FIPS PUB has undergone change to bring it in line with IEEE Standard 1003.1-1988. This standard does not, however, include other kernel operations that are widely understood as part of the operating system kernel, such as realtime operations or kernel security capabilities. These capabilities will become parts of related standards. The FIPS PUB is complete as written.

Maturity: Antecedents of POSIX have existed for 20 years. The current standard was developed over an eight year period. Much research based on POSIX antecedents has been pursued which has led to various improvements in the POSIX specification.

Stability: FIPS PUB 151-1 is expected to be revised as FIPS PUB 151-2 which will bring it into line with the current national (IEEE 1003.1-1990) and international (ISO 9945-1) standards.

De facto usage: Even if users do not reference this specification in procurement documents, vendors will likely propose products that meet the specification or are compatible with it.

Known problems/limitations: POSIX consists of a family of related specifications, some of which are still in draft stages (e. g., IEEE P1003.2 Shell and Utilities, IEEE P1003.4 Realtime, etc.) FIPS PUB 151-1 is complete in itself. The other pieces mentioned herein will augment FIPS PUB 151-1 usability as additional FIPS PUBs.

Conformance testing: NIST has developed a conformance test suite and will offer third-party testing services via Accredited POSIX Testing Laboratories.

Future plans: Existing kernel operations will not change, although additional operations are on the horizon. Related standards for other service area components, such as realtime operations, system management, etc., will be developed over the next 1 to 3 years.

Alternative specifications: None (All other known specifications that provide these services are compatible with POSIX.)

Suggested reference: Operating system environments offered as a result of the requirements of which this is a part shall implement FIPS PUB 151-1, Portable Operating System Interface for Computer Environments (POSIX) as well as any additional elements specified elsewhere in this requirements document, and shall require validation in accordance with provisions contained in FIPS PUB 151-1. (NOTE: Users may additionally require a list of test exceptions attached to validation certificates from individual vendors.)

2.1.2 Commands and Utilities — NIST Planned FIPS on POSIX Shell and Utility Application Interface for Computer Operating System Environments — IEEE P1003.2 Draft 11

Specification available from: IEEE

Publication date: February 1991

Sponsoring organization: IEEE

Applicability: Commands and utilities include mechanisms for operations at the operator level, such as comparing, printing, and displaying file contents, editing files, pattern searching, evaluating expressions, logging messages, moving files between directories, sorting data, executing command scripts, scheduling signal execution processes, and accessing environment information.

Level of consensus: This specification is still in a draft stage and is currently undergoing a preliminary ballot by IEEE working group members. Changes are planned in the near future, but these are characterized as fine-tuning types of changes and additions where needed.

Product availability: Implementations of commands and utilities capabilities are available in proprietary operating systems that are very similar to the specification.

Completeness: The functional specifications are still subject to modification, but major features are already included in the draft.

Maturity: Antecedents and similarly-specified implementations have existed for ten to twenty years.

Stability: Drafts 9, 10, and 11 each included major revisions. Draft 11, or perhaps Draft 12, appears to be the one that will become the basis for the standard. All sections are still subject to change based on a continuing IEEE balloting process. The consensus-building process appears to have coalesced around Draft 11.

De facto usage: If users do not reference this specification in procurement documents, vendors will probably propose products that do not meet this specification, or are not compatible with the specification.

Known problems/limitations: The specification is in draft stage and will not be available for full use before 1992.

Conformance testing: When a FIPS PUB is adopted, NIST plans to provide certification procedures and tests for demonstrating product conformance. No time schedule has been developed for these actions.

Future plans: The specification will be revised as needed to reflect the evolving national and international consensus.

Alternative specifications: AT&T System V Interface Definition (SVID), Open Software Foundation OSF/1, X/Open Portability Guide Issue 3 (XPG3)

Suggested reference: Commands and utilities offered as a result of the requirements of which this is a part shall conform to the requirements in NIST Planned FIPS on POSIX Command and Utility Application Interface for Computer Operating System Environments, defined in IEEE P1003.2.

2.1.3 System Management — NIST Planned FIPS PUB on Government Network Management Profile (GNMP)

Specification available from: Computer Systems Laboratory (CSL) National Institute of Standards and Technology (NIST)

Publication date: Draft dated March 8, 1991

Sponsoring organization: NIST

Rationale: Other arenas (including X/Open, OSF, and other organizations) recognize the need for a common framework for systems and network management. Our approach builds on the

established Open System Interconnection (OSI) network management framework for system management specifications. The recommended specification is not so much a recommendation for a system management specification as it is a recommendation for using specifications based on the common OSI management framework (i. e., any system or network administration specification chosen should integrate with the OSI framework).

Applicability: System management includes the capabilities of defining and managing user access, devices, file systems, administrative processes (job accounting), queues, machine/platform profiles, authentication (passwords), authorization of resource usage, and system backup on single platforms or in environments composed of heterogeneous networked platforms.

Level of consensus: This specification is in the process of being redefined to conform more to the OSI manner of specification. Previous drafts of the specification have been essentially abandoned and work has begun to define management objects that are comparable to OSI concepts. The draft referenced here is a "strawman" proposal developed by NIST.

Product availability: A few products implement parts and subsets of the specification, but full implementations will not be available until 1992.

Completeness: The current specification consists of informal notes and proposals that have not been discussed widely in the working group. The specification will be dealt with in three phases. Phase I will include OSI layers 1 and 2 for local area network (LAN) communication standardization. Communications protocols and system management parts of the specification are mostly complete. Phases II (OSI layers 3 through 7) and III (application services and operating system interface) will add functionality and include complete component descriptions.

Maturity: The specification is based on the OSI Reference Model and has a precedent in the OSI Network Management Framework which is an international standard (Basic Reference Model Part 4—Management Framework, ISO/IEC 7498-4:1989). There is, however, no consensus on what must be included in the dictionary of objects necessary to standardize system management. These objects must include options and parameters required to provide the functionality and services for connecting components within all of the OSI interconnection model layers.

Stability: Since previous versions of the draft have been discarded and work has basically started over, the specification will not be stable for the foreseeable future.

De facto usage: If users do not reference this specification in procurement documents, vendors will probably propose products that do not meet this specification, or are not compatible with the specification.

Known problems/limitations: The most pressing problem known is that of a lack of system object definitions and a method for managing the objects once they have been defined. The specification is still in a working draft stage.

Conformance testing: When a FIPS PUB is adopted, NIST plans to provide certification procedures and tests for demonstrating product conformance. No time schedule has been developed for these actions.

Future plans: NIST and other organizations are planning a review of the available technology and identifying promising areas for specification work. A specification is expected to be proposed as a FIPS PUB before 1992.

Alternative specifications: Common Management Information Protocol (CMIP) including Association Control Service Element (ACSE), and Remote Operations Service Element (ROSE); or Simple Network Management Protocol (SNMP). (These are generally immature specifications.)

Suggested reference: System management offered as a result of the requirements of which this is a part shall conform to the requirements in NIST planned FIPS PUB on Government Network Management Profile.

2.1.4 Operating System Security — Security Interface for the Portable Operating System Interface for Computer Environments (IEEE P1003.6 Draft 8)

Specification available from: IEEE

Publication date: November 5, 1990

Sponsoring organization: IEEE

Applicability: Security considerations are specified in terms of data encryption mechanisms, access control, reliability control, system logging, fault tolerance, and audit facilities. (The security interface does not specify a secure operating system; only its interface.)

Level of consensus: This specification is still in a draft stage and will probably be balloted in early 1991.

Product availability: Implementations exist with some, but not all, of the features.

Completeness: Major topics including key features have not yet been finalized.

Maturity: The basic technology is well-understood and the specification is based on several underlying standards/criteria.

Stability: All sections are subject to changes ranging from insignificant to major revisions.

De facto usage: If users do not reference this specification in procurement documents, vendors will probably propose products that do not meet this specification, or are not compatible with the specification.

Known problems/limitations: The specification is incomplete and will not be available for full use for several years. Projected completion is expected in late 1991, however, this projection could change due to modifications in the specification.

Conformance testing: A method of measuring conformance has not been defined. Until this has occurred, no determination of where or when testing will take place will be made.

Future plans: Security specifications will expand to integrate interfaces for other service area components.

Alternative specifications: National Computer Security Center "Rainbow" security standards for access control (NCSC-STD-020-A) and password management (NCSC-STD-002-85).

Suggested reference: System security offered as a result of the requirements of which this is a part shall conform to the requirements in Security Interface for the Portable Operating System Interface for Computer Environments, defined in IEEE P1003.6.

2.2 User Interface Services

	LOC	PAV	CMP	MAT	STB	DFU	PRL
X Window Sys.	●	○	○	●	●	●	
XVT		○	○				

LOC -- Level of consensus
PAV -- Product availability
CMP -- Completeness
MAT -- Maturity
STB -- Stability
DFU -- De facto usage
PRL -- Problems/limitations

Legend: ● -high evaluation ○ -average evaluation blank -low evaluation

The components of this service include the seven layers of the User Interface System Reference Model (UISRM) described in section 1.4.2. Layers 0, 1, and 2 are specified in FIPS PUB 158, which is based on the Massachusetts Institute of Technology's X Window System. Work is still in progress on layers 3, 4, 5, and 6. The work has not progressed enough to include specifications in this report on those layers. (A stop-gap measure that provides a stable platform in the form of an insulating layer is all that can be recommended at this time.)

2.2.1 Client-server Operations — User Interface Component of Applications Portability Profile, FIPS PUB 158 (MIT X Window System)

Specification available from: NTIS

Publication date: May 29, 1990

Sponsoring organization: Massachusetts Institute of Technology X Consortium

Applicability: Client-server operations define the relationships between client and server processes operating within a network, in particular, graphical user interface display processes. In this case, the program that controls each display unit is a server process, while independent user programs are client processes that request display services from the server. The services that constitute this service area are included in layers 0, 1, 2, and 3 of the UISRM as described in section 1.4.2.

Level of consensus: An X Protocol standard is being developed by X3K13.6; Xlib and the Xt Intrinsics are not standardized at this time. A FIPS PUB based on the X Consortium specifications described above has been approved as FIPS PUB 158.

Product availability: Virtually all major hardware vendors have produced implementations of the X Window System. A copy of the software is available from expo.lcs.mit.edu at Massachusetts Institute of Technology through the "ftp" command.

Completeness: The specification defines the primitives, intrinsic functions based on these primitives, and some of the lower level library specifications for user interface services. It does not specify any of the "look and feel" or style services that will be accessible at higher levels of abstraction. It does not contain a full complement of utilities and services required to allow application programmers to easily program user interfaces.

The X Window System defines a C language source code level interface to a network-based bit-mapped graphic display system. The computer program source code contained in Version 11, Release 3 is not part of the specification for the FIPS PUB. The specification for this FIPS PUB are the following documents from the X Consortium, X Window System, Version 11, Release 3:

- X Window System Protocol, X Version 11
- Xlib - C language X Interface
- X Toolkit Intrinsics - C Language Interface
- Bitmap Distribution Format 2.1.

Maturity: The X Window System has been in existence since 1983. It was one of the products to come out of Project Athena at MIT.

Stability: The Xlib and X Window System Protocol documents are fairly stable. X Toolkit Intrinsics will achieve a large measure of stability with the next release. Changes in these specifications are expected to include more-or-less tuning types of modifications rather than major deletions or additions. A revised FIPS PUB will probably be issued when appropriate as other national and international standards are approved.

De facto usage: Even if users do not reference this specification in procurement documents, vendors will likely propose products that meet the specification or are compatible with it.

Known problems/limitations: Most of the functionality is available at a very low level (i. e., too low for most application programming).

Conformance testing: The U. S. Government will provide third-party conformance testing services through the National Voluntary Laboratory Accreditation Program (NVLAP) when test suites and testing policy for FIPS PUB 158 become available.

Future plans: Revision of the FIPS PUB will be considered and made where appropriate as national and international standards are approved. IEEE Working Group P1201 is preparing two documents: IEEE P1201.1 focuses on a high level window application program interface toolkit; IEEE P1201.2 is concerned with drivability/usability of user interfaces (i. e., general location of on-screen

symbols, what happens when a mouse click is detected, etc.). The Inter-Client Communications Conventions Manual (ICCCM) from the MIT X Consortium, which defines how user application programs communicate with each other in a system, will be included in a future update of the FIPS PUB.

Alternative specifications: None

Suggested reference: User interface client-server operations offered as a result of the requirements of which this is a part shall implement FIPS PUB 158 User Interface Component of Applications Portability Profile, as well as any additional elements specified elsewhere in this requirements document, and shall require validation in accordance with provisions contained in FIPS PUB 158.

2.2.2 Presentation — Extensible Virtual Toolkit (XVT)

Specification available from: XVT Software, Inc., Boulder, Colorado

Publication date: January 1990

Sponsoring organization: XVT Software, Inc.

Rationale: XVT was chosen as a way to accommodate several dominant application programming interfaces (APIs) for user interface management systems and also several popular graphical interface implementations. XVT essentially buys time until a consensus in the standards arena has developed, or a clear statement from the marketplace appears. XVT in essence provides a logical platform to provide some, if not all, of the graphical and character user interface functionality needed in applications. It does not replace layers 4 and 5 of the user interface system reference model. It acts as a virtual application programming interface (VAPI) so that various implementations of layers 4 and 5 can be integrated without having to rewrite portions of the application.

Applicability: It is not cost effective to write applications that use only the Xt intrinsics and Xlib functions of the X Window System. Most developers would prefer to use higher level interfaces to window system functions. Ideally, the application program should be portable at the source code level among a wide variety of platforms, from high-end workstations, to minicomputers, and personal computers. When a standard high level interface emerges, it will probably be available on a wide range of vendors' products. Unfortunately, such a standard cannot be expected before 1992, possibly later. As an interim solution, NIST has identified a window system interface called the Extensible Virtual Toolkit (XVT) that allows C programs to be portable among a variety of workstations and personal computers. The interface allows applications to be ported to several window systems, including Motif, Open Look, MS-Windows, Macintosh, OS/2 Presentation Manager, and CTOS Presentation Manager. A subroutine library for character cell terminal support (i. e., character-oriented displays) is also available to support MS-DOS, UNIX, CTOS, and OS/2.

XVT is a thin insulating layer between application programs and native window systems. It does not provide window and graphics support; only a common interface that applications can use to access the features of the underlying window and graphics system. The application program makes calls to XVT. XVT, in turn, calls the underlying window system to invoke the functions needed

by the user. In terms of the reference model, it can be thought of as a layer at the application layer. An application using XVT would be organized as shown in figure 9. Note that a User Interface Management System is not included as part of XVT. UIMSs can sometimes make application development easier, but most window applications today are built without them. Figure 10 shows how XVT can be used with a variety of window systems.

Because XVT is essentially a mapping between the application and a window system, agencies should also select a window system that will meet their needs. To ensure portability and to reduce conversion costs, agencies are advised to acquire systems that support the X interfaces specified in FIPS PUB 158, as a future toolkit standard is expected to be based on these interfaces.

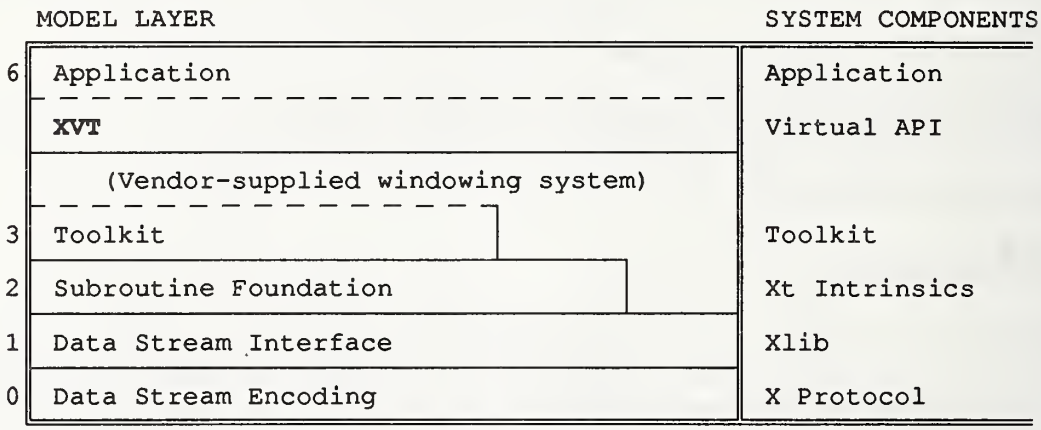


Figure 9. XVT Relationship to User Interface Reference Model.

XVT Software, Inc. has given permission to any organization or individual to use and disseminate the XVT specification for the creation of, documentation of, and distribution of a toolkit that implements user interfaces as part of the NIST Application Portability Profile.

Level of consensus: IEEE P1201 committee is debating whether or not to use the XVT specification (or one or more of several other specifications) as the basis for presentation and/or dialogue layers of the UISRM.

Product availability: Several major software/hardware vendors have chosen XVT as the VAPI used on UNIX platforms and make XVT available as part of implementations.

Completeness: XVT encompasses many dialogue and presentation services from the UISRM. Bindings for C and C++ programming languages are also available. It does not include the style capabilities that are inherent in vendor-supplied toolkits, presentation managers, and dialogue managers.

Maturity: XVT has been available since 1988.

Stability: Since the reference model is not firmly established, changes in XVT specifications can be expected.

De facto usage: If users do not reference this specification in procurement documents, vendors will probably propose products that do not meet this specification, or are not compatible with the specification.

Known problems/limitations: XVT requires a native windowing system (among which the X Window System may be chosen) designed for the platform on which it is to execute. The native windowing system provides the style guide (i. e., the look and feel) of specific types of application interactions. In the absence of a graphical user interface system, a character-based set of subroutines may be substituted (subroutines available in the XVT package).

Conformance testing: No conformance tests exist. The vendor ships test programs with the product for user verification of installation and demonstration of interactions with native windowing systems.

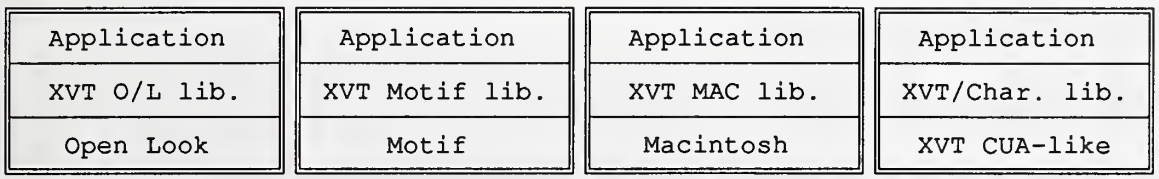


Figure 10. XVT Use in Window Systems.

Future plans: Additional platforms and windowing systems are being added as customer requirements are identified.

Alternative specifications: TAE+ (developed by National Aeronautics and Space Administration [NASA], available from COSMIC). Other popular toolkits available from vendors may be used directly in place of XVT. Users must evaluate the likelihood that these toolkits will become part of international standards.

Suggested reference: Object definition and management, window management, and dialogue support offered as a result of the requirements of which this is a part shall conform to the requirements in XVT Programmer's Manual from XVT Software, Inc., Boulder, Colorado. Any user interface toolkit proposed for use in procurements shall be based on FIPS PUB 158 components, and shall work in concert with and support XVT.

2.3 Programming Services

Programming languages and language bindings, and computer-aided software engineering (CASE) environments and tools are included as components of programming services. (Alternative specifications are not included for programming languages.)

	LOC	PAV	CMP	MAT	STB	DFU	PRL
Ada	●	●	●	●	○	○	○
C	●	●	●	●	○	●	●
COBOL	●	●	●	●	●	●	●
FORTRAN	●	●	●	●	●	●	●
PASCAL	●	○		●	●		
ECMA PCTE	○			○			
SCCS		○					

LOC -- Level of consensus
PAV -- Product availability
CMP -- Completeness
MAT -- Maturity
STB -- Stability
DFU -- De facto usage
PRL -- Problems/limitations

Legend: ● -high evaluation
○ -average evaluation
blank -low evaluation

2.3.1 Programming Languages and Bindings — Ada FIPS PUB 119

Specification available from: NTIS

Publication date: November 8, 1985

Sponsoring organization: Ada Joint Program Office

Applicability: Ada is a general purpose high-level programming language. In addition, it provides strong data-typing, concurrency, and significant code-structuring capabilities. It is particularly suited to embedded realtime systems, distributed systems, and highly reliable software development.

Level of consensus: Ada is a national standard (ANSI/MIL-STD-1815A-1983), an international standard (ISO 8652:1987), and a FIPS PUB. The Department of Defense has directed that Ada be used in all DoD systems development.

Product availability: Numerous FIPS-validated compilers and Ada environments are available commercially.

Completeness: Ada is complete for use as a general-purpose programming language.

Maturity: Ada was developed as a DoD-sponsored language and is based on well-defined predecessor languages such as Pascal.

Stability: Ada has the backing of the Department of Defense, the U. S. Government, and ISO.

De facto usage: If users do not reference this specification in procurement documents, vendors will probably propose products that do not meet this specification, or are not compatible with the specification.

Known problems/limitations: Unknown.

Conformance testing: Ada conformance and validation testing are carried out under the auspices of DoD's Ada Joint Program Office (AJPO). A list of validated compilers is published by NIST quarterly.

Future plans: A new revision of Ada (a.k.a. Ada-9X) is in the review process and is planned for release in 1992. Related standards are in the process of adding, or have added Ada bindings (e. g., SQL, POSIX, and MUMPS).

Suggested reference: Ada processors offered as a result of the requirements of which this is a part shall conform to the requirements of Ada (FIPS PUB 119) and shall require validation.

2.3.2 Programming Languages and Bindings — C FIPS PUB 160

Specification available from: NTIS

Publication date: March 13, 1991

Sponsoring organization: X3J11

Applicability: C is a general purpose high-level programming language designed for use in various levels of software including operating systems, system level software (e.g., special purpose processors), and business and scientific application software.

Level of consensus: ANSI has approved the C language standard (December 1989), and ISO is considering a draft international standard which should be identical to the ANSI standard. The ANSI standard has recently been accepted by the Department of Commerce as a Federal Information Processing Standard.

Product availability: Numerous C compilers, interpreters, and associated products are commercially available and supported.

Completeness: C includes facilities for every level of programming, from low level (hardware control) operations to high level abstract functions and procedures. Data structuring, reusable library support, and memory management are included.

Maturity: Development of C has progressed from a family tree of similar languages developed in academia to a well-defined, widely-supported language over a period of 15 years.

Stability: Since the standard is rather new, it is difficult to determine if all functionality, structure, syntax, and semantics are stable. Some changes may be necessary to fine-tune the standard as usage experience is gathered. No planned changes are foreseen, however.

De facto usage: Even if users do not reference this specification in procurement documents, vendors will likely propose products that meet the specification or are compatible with it.

Known problems/limitations: Problems with the standard have been ironed out as much as possible.

Conformance testing: Commercial conformance test suites are available. The U. S. Government is in the process of identifying a test suite for testing conformance to the FIPS PUB.

Future plans: X3 is considering development of a second language standard based upon the C language for object-oriented software development. The planned specification is called C++.

Suggested reference: C language processors offered as a result of the requirements of which this is a part shall conform to the requirements in FIPS PUB 160.

2.3.3 Programming Languages and Bindings — COBOL FIPS PUB 021-3

Specification available from: NTIS

Publication date: January 12, 1990

Sponsoring organization: X3J4

Applicability: COBOL is designed for use in programming self-documenting business oriented applications.

Level of consensus: The U. S. Government and international standards (ISO 1989:1985) are based on the ANSI standard, ANSI X3.23-1985 and Addendum X3.23A-1989.

Product availability: COBOL is the most widespread programming language. An estimated 60 to 80 percent of all Federal applications are written in COBOL. All major vendors offer FIPS COBOL.

Completeness: The current standard does not include realtime, operating system, and communications components. It is most complete in the areas of data manipulation, and business/financial applications.

Maturity: COBOL is one of the oldest standard general-purpose programming languages, having been established in the early 1960's by DoD initiative.

Stability: The X3J4 committee is in the process of adding new functionality for communications interfaces and screen management. Compatibility with previous versions of the standard will be maintained. This has historically been one of COBOL's stronger points.

De facto usage: Even if users do not reference this specification in procurement documents, vendors will likely propose products that meet the specification or are compatible with it.

Known problems/limitations: COBOL has always been very specialized toward the development of general purpose business and financial applications. It is very limited in other types of applications such as in realtime and communications, although this may change with functionality introduced by proposed revisions.

Conformance testing: Conformance test suites are available from commercial and federal sources. Testing services are available from NIST. NIST publishes an updated list of FIPS-validated compilers quarterly.

Future plans: The addition of new functionality will greatly expand the capabilities of COBOL to other application areas.

Suggested reference: COBOL processors offered as a result of the requirements of which this is a part shall conform to the requirements in COBOL (FIPS PUB 21-3) and shall implement all of the language elements of the level of COBOL specified elsewhere in this requirements document, and shall require validation, as well as any additional language elements specified elsewhere in this document.

2.3.4 Programming Languages and Bindings — Fortran FIPS PUB 069-1

Specification available from: NTIS

Publication date: December 24, 1985

Sponsoring organization: X3J3

Applicability: Fortran is a high-level programming language used largely in scientific and engineering applications where large amounts of data is analyzed and processed.

Level of consensus: The FIPS PUB and the international standard (ISO 1539:1980) are based on the national standard (ANSI X3.9-1978).

Product availability: Every major hardware vendor markets a Fortran compiler based on the standard. Additional compilers are available from a multitude of software vendors.

Completeness: It is a general purpose programming language with capabilities for performing virtually any type of application function. It was originally developed to assist in the development of scientific calculation applications, but it has since been extended to cover other types of applications.

Maturity: Fortran is one of the oldest programming languages, and was also the first one to be standardized.

Stability: Although it has undergone several major revisions over its lifespan, Fortran contains virtually all of the same capabilities that were available when it was new. In addition, it contains elements for assisting in the development of information systems, realtime and process control systems, structured programming constructs, etc.

De facto usage: Even if users do not reference this specification in procurement documents, vendors will likely propose products that meet the specification or are compatible with it.

Known problems/limitations: Due to loose data-typing and some idiosyncracies of various compilers, some debugging problems are very difficult to locate and fix.

Conformance testing: Conformance test suites are available from commercial and federal sources. Testing services are available from NIST. NIST publishes an updated list of FIPS-validated compilers quarterly.

Future plans: A revision to the standard is planned for 1991. An IEEE Working Group is defining a POSIX/Fortran binding.

Suggested reference: Fortran processors offered as a result of the requirements of which this is a part shall conform to the requirements in Fortran (FIPS PUB 69-1) and shall implement all of the language elements of the level of Fortran specified elsewhere in this requirements document, and shall require validation, as well as any additional language elements specified elsewhere in this document.

2.3.5 Programming Languages and Bindings — Pascal FIPS PUB 109

Specification available from: NTIS

Publication date: January 16, 1985

Sponsoring organization: Joint X3J9-IEEE Pascal Standards Committee

Applicability: Pascal is a high-level programming language used primarily in teaching environments for training computer science students in the concepts of programming. It is also used in general application areas such as business, science, etc.

Level of consensus: The FIPS PUB and international standard (ISO 7185:1983) are based on the national standard (ANSI/IEEE770X3.97-1983).

Product availability: Numerous hardware and software vendors market standard implementations of Pascal interpreters and compilers.

Completeness: Pascal is a general-purpose programming language. It does not presently have constructs for performing file input-output at other than the byte level.

Maturity: Pascal is a strongly-typed language based upon a model of program design that is well-understood, and has been used extensively for teaching programming in universities.

Stability: No major revisions of the standard have taken place. It is very stable.

De facto usage: If users do not reference this specification in procurement documents, vendors will probably propose products that do not meet this specification, or are not compatible with the specification.

Known problems/limitations: Pascal does not have well-developed intrinsic file input-output operations.

Conformance testing: Conformance test suites are available from commercial sources, and testing services are available from NIST. NIST publishes an updated list of FIPS-validated compilers quarterly.

Future plans: NIST is currently considering whether to discontinue validating Pascal compilers based on the low need of this language by Federal agencies and the limited resources that NIST has for validation efforts.

Suggested reference: Pascal processors offered as a result of the requirements of which this is a part shall conform to the requirements in Pascal (FIPS PUB 109) and shall implement all of the language elements of the level of Pascal specified elsewhere in this requirements document, and shall require validation, as well as any additional language elements specified elsewhere in this document.

2.3.6 Integrated Software Engineering Environments (ISEE) and Tools — European Computer Manufacturers Association (ECMA) Portable Common Tool Environment (PCTE)

Specification available from: ECMA

Publication date: December 1990

Sponsoring organization: ECMA

Rationale: NIST is working to develop a suite of standards in the ISEE area. A cornerstone of this standards effort is the development of a reference model for ISEE. NIST has adopted the ECMA reference model as the base definition for the framework for ISEE requirements, technical integration, and standards specification. The ECMA reference model is partially based on the ECMA PCTE specifications and is used to identify the needed set of standards that define a comprehensive interface for integrating software tools.

Applicability: Integrated software engineering environments (ISEE) and tools include systems and programs that assist in the automated development and maintenance of software. These include, but are not limited to, tools for requirements specification and analysis, for design work and analysis, for creating program code, for testing, for documenting, for prototyping, and for group communication. The interfaces among these tools include services for storing and retrieving information about systems and exchanging this information among the various system development environment components. An adjunct to these capabilities is the ability to manage and control the configuration of software components, test data, and libraries.

Level of consensus: European manufacturers have agreed to use the ECMA PCTE specification as the basis for any proposed ISO standards. (NIST has incorporated the ECMA reference model in

the NIST ISEE reference model. This will promote harmonization of federal and international standards in software engineering environments.)

Product availability: A very limited number of programming environments are based on ECMA PCTE and are now commercially available.

Completeness: While much work has been accomplished in defining the interfaces among various software engineering components, the interfaces are still incomplete and in a state of flux. For example, there is no consensus on the type of data model necessary to support the information structures for the environment. Standards for software engineering tools and environments are fairly new and in a state of rapid evolution.

Maturity: ECMA has been working on the programming environment reference model for the past three years. Antecedents of ECMA PCTE existed for several years prior to that. The technology and research necessary to define and implement the interfaces contained in such an environment have evolved rapidly and are continuing to expand and mature.

NIST has an ongoing project to develop guidelines on interface standards for ISEE and expects to identify and adopt open systems standards which will promote the development of integrated software engineering environments that support application software across the entire software lifecycle. A select working group is advising NIST on the merits of various ISEEs and has recommended the ECMA reference model as a basis for the NIST reference model. The ECMA PCTE and Ada Programming Support Environment (APSE) specifications can be mapped to the current ECMA reference model.

Stability: With possible changes emanating from ECMA members and possible modifications to be suggested by NIST, ECMA PCTE is still very much a draft specification. In European markets, ECMA PCTE is often cited as the preferred interface specification for software development and support environments, especially in entity-relationship models. In the U. S. and other markets, it is still viewed as new technology and not widely disseminated.

De facto usage: If users do not reference this specification in procurement documents, vendors will probably propose products that do not meet this specification, or are not compatible with the specification.

Known problems/limitations: Unknown at this time.

Conformance testing: None.

Future plans: NIST is developing criteria to evaluate proprietary and public domain specifications of tool interfaces for inclusion in a proposal for standardization. In addition, Ada and C application programming interfaces will be added.

Suggested reference: Integrated software engineering environments (ISEE) offered as a result of the requirements of which this is a part shall conform to the requirements defined in the European Computer Manufacturers Association (ECMA) Portable Common Tool Environment (PCTE).

2.3.7 Integrated Software Engineering Environments (ISEE) and Tools — Source Code Control System (SCCS)

Specification available from: X/Open

Publication date: 1989

Sponsoring organization: X/Open

Applicability: SCCS is used to manage and control various versions of machine readable text data including program source code, documentation, libraries, etc., throughout the lifecycle of each.

Level of consensus: SCCS file commands and utilities are contained in the X/Open Portability Guide Issue 3 (XPG3) in Volume 1, "XSI Commands and Utilities."

Product availability: Any implementation of UNIX based on the full functionality specified in the X/Open Portability Guide Issue 3 contains SCCS.

Completeness: SCCS contains virtually all capabilities necessary for managing multiple versions of files.

Maturity: SCCS and its antecedents have existed for 15 years.

Stability: Planned modifications have generally been announced well in advance of releases and have been upwardly compatible. No planned modifications are known at this time.

De facto usage: If users do not reference this specification in procurement documents, vendors will probably propose products that do not meet this specification, or are not compatible with the specification.

Known problems/limitations: SCCS implies use of an underlying UNIX operating system.

Conformance testing: None.

Future plans: One of the few programs considering configuration management is the ECMA PCTE project of the European Computer Manufacturers Association (ECMA). No specification has been developed.

Alternative specifications: None.

Suggested reference: Software configuration management processors offered as a result of the requirements of which this is a part shall conform to the requirements defined in Source Code Control System (SCCS) file commands and utilities, X/Open Portability Guide Issue 3 (XPG3), Volume 1, "XSI Commands and Utilities."

2.4 Data Management Services

Data management services include the data dictionary/directory component for accessing and modifying data about data (i. e., metadata), the database management system component for accessing and modifying structured data, and the distributed data component for accessing and modifying data from a remote database.

	LOC	PAV	CMP	MAT	STB	DFU	PRL
IRDS	○		○	●	○		
SQL	●	○	●	●	●	●	●
RDA			○		○		

LOC -- Level of consensus
PAV -- Product availability
CMP -- Completeness
MAT -- Maturity
STB -- Stability
DFU -- De facto usage
PRL -- Problems/limitations

Legend: ● -high rating, ○ -average rating, blank -low rating

2.4.1 Data Dictionary/Directory Component — Information Resource Dictionary System (IRDS) FIPS PUB 156

Specification available from: NTIS

Publication date: April 5, 1989

Sponsoring organization: X3H4

Applicability: Data dictionary/directory services consist of utilities and systems necessary to catalog, document, manage, and use metadata (information about data).

Level of consensus: The ANSI standard (ANSI X3.138-1988) and the FIPS are the same document. ISO is working on an IRDS specification that is significantly different in some respects from the ANSI standard.

Product availability: Commercial implementations have been developed, but their quality has not yet been determined. A prototype implementation is available from CSL which contains a large subset of IRDS functionality.

Completeness: The specification includes user interfaces only (i. e., there is currently no application programming interface).

Maturity: Antecedents of the IRDS have been in existence for fifteen years. The current specification has been in development during the major part of this time.

Stability: The next few years should see nominal changes in the current standard. Related standards efforts are specifying additional and upwardly-compatible functionality. Two on-going efforts will make the IRDS active (i. e., provide interfaces to other software). First, a draft proposed IRDS

Export/Import File Format is expected to become an ANSI standard in 1991; this will be appropriate for schema and metadata interchange among IRDS compliant databases, among IRDS and CASE tools with repositories or dictionaries, and between IRDS and application programs. Second, a draft proposed IRDS Services Interface is expected to become an ANSI standard in 1991 or 1992; this will be appropriate for metadata interchange with a database management system, and between IRDS and application programs.

De facto usage: If users do not reference this specification in procurement documents, vendors will probably propose products that do not meet this specification, or are not compatible with the specification.

Known problems/limitations: Virtually all procurements that specify a data dictionary/repository require it to be active. IRDS does not currently possess capabilities in this area. As stated above, standards work is being carried out to provide software interfaces to the IRDS to allow the development of active functionality.

Conformance testing: Conformance tests are currently under development.

Future plans: Related standards work will provide a software interface, enhanced functionality and capability to manage object-oriented data structures, and provide for communication of information between applications and other data management tools. A major revision to the standard is envisioned in about three years to include this new functionality.

Alternative specifications: None.

Suggested reference: Data dictionary/directory services provided as a result of the requirements of which this is a part shall conform to the requirements established in FIPS PUB 156 and shall implement all of the functions contained therein, as well as any additional programming requirements specified elsewhere in this document.

2.4.2 Database Management System Component — Database Language SQL FIPS PUB 127-1

Specification available from: NTIS

Publication date: February 2, 1990

Sponsoring organization: X3H2

Applicability: DBMS services include definition, management, query, and security of structured data stored in a relational database management system. The security interface for granting and revoking privileges does not specify a secure DBMS; only its interface.
Level of consensus: FIPS PUB 127-1 adopts both ANSI X3.135-1989 (SQL) and ANSI X3.168-1989 (Embedded SQL), and the first of these ANSI standards is identical to ISO 9075:1989. Embedded SQL is part of an emerging ISO revision. SQL has been adopted as the database management system component of X/Open, OSF, SQL-Access, and other vendor consortia.

Product availability: Numerous implementations of the original ANSI SQL exist on all classes and brands of platforms. Complete functionality as defined in FIPS PUB 127-1, along with optional features also specified in the FIPS PUB, are being implemented.

Completeness: The existing standard specifies data definition, data manipulation, access control, and limited integrity constraints. The FIPS requires ANSI X3.135-1989 Level 2 conformance to one or more FIPS programming languages and requires a FIPS Flagger to flag extensions in an implementation.

Maturity: The SQL data model is based on the relational model first published in 1969. The first commercial systems were available in 1979, and the first SQL standard was published in 1986.

Stability: The SQL language has firm mathematical foundations in first order predicate calculus. Standards groups and vendors are firmly committed to upward compatibility in revisions and future extensions to the standard. Existing features are expected to remain stable for the foreseeable future. An emerging enhanced specification under active development in both ANSI and ISO will include substantial additional features for schema manipulation, dynamic SQL, exception handling, enhanced integrity constraints, transaction management, and data administration.

De facto usage: Even if users do not reference this specification in procurement documents, vendors will likely propose products that meet the specification or are compatible with it.

Known problems/limitations: The existing standard is specified for stand-alone, single environment databases.

Conformance testing: Version 2.0 of the NIST SQL test suite has been publicly available since December 1989, and a formal SQL test service was instituted by NIST in April 1990. The SQL test suite measures conformance to both required and optional features of FIPS PUB 127-1. NIST publishes a quarterly list of FIPS-validated processors.

Future plans: An emerging SQL specification, with features identified in the Stability paragraph above, is expected to be adopted by ANSI and ISO in 1992. Further enhancements are expected in the 1995 time frame. Specifications for access to remote heterogeneous sites are under development in an emerging ISO Remote Database Access (RDA) specification (see sec. 2.4.3). Specifications for distributed database management are in very early stages of development. Tools for the support of object-oriented data management such as triggers, assertions, user-defined types, domain and table hierarchies, and stored procedures are under active consideration as follow-on enhancements to the SQL standard.

Alternative specifications: None.

Suggested reference: All relational database language interfaces offered as a result of the requirements of which this is a part shall conform to the requirements set forth in FIPS PUB 127-1, Database Language SQL, and shall be validated, and shall implement all of the language elements specified according to the guidance in FIPS PUB 127-1, as well as any additional language features as specified elsewhere in this document.

2.4.3 Distributed Data Component — Remote Database Access (RDA)

Specification available from: ISO/IEC JTC1/SC21 Secretariat

Publication date: Draft available.

Sponsoring organization: ISO/IEC JTC1

Applicability: RDA is used to establish a remote connection between an RDA client, acting on behalf of an application program, and an RDA server, interfacing to a process that controls data transfers to and from a database. The goal is to promote the interconnection of applications with database systems within heterogeneous environments, with emphasis on an SQL server interface.

Level of consensus: The ISO/IEC RDA specification is currently undergoing draft balloting in ISO/IEC Joint Technical Committee 1 (JTC1). The specification is in two parts: Part 1 -- Generic Model, Service, and Protocol, and Part 2 -- SQL Specialization. Final adoption is expected in 1992. Vendor consortia such as SQL-Access hope to have working prototypes operational in 1991 to demonstrate interoperability among different SQL servers. RDA is also a working task group of the NIST Open System Interconnection (OSI) Implementor's Workshop.

Product availability: There are no known RDA implementations, but many SQL vendors are planning to have conforming client and server products available before final adoption as a standard in the 1992 time frame.

Completeness: RDA services consist of dialogue management, association control, resource handling, and data language services between a single client and a single server. Association control includes making a connection to a specific database at the server site. SQL statements are sent as character strings with a separate list of input parameters, and resulting data or exception conditions are returned. Transaction management services are also included for both one-phase and two-phase commit protocols. Individual applications determine whether both one- and two-phase commits are available. The existing specification does not consider multiple simultaneous connections, so distributed database management is the concern of the client process. Extensions for true distributed database management among different SQL implementations are under consideration.

Maturity: Methods for establishing communications links between client and server sites are well known, but agreements on nonproprietary communications protocols are very new.

Stability: The client/server architecture is just one of several architectures used for implementing distributed systems and there is no final conclusion as to which is best. The stability of RDA depends on the stability of the client/server architecture.

De facto usage: If users do not reference this specification in procurement documents, vendors will probably propose products that do not meet this specification, or are not compatible with the specification.

Known problems/limitations: Although distributed extensions are under consideration, existing RDA only specifies the service and protocol between a single client and a single server. RDA does not currently specify distributed database access.

Conformance testing: RDA will likely become a future optional part of conformance testing for GOSIP. At the present time, RDA can be tested indirectly using the NIST SQL test suite, with application programs at the client site and data at the server site.

Future plans: Enhancement projects for distributed database and stored database procedures have already been proposed to ISO. Extensions to support new features in evolving SQL standards are under development. (Vendor agreements reached by SQL-Access, X/Open, and other vendor consortia are finding their way into the RDA standard.)

Alternative specifications: None.

Suggested reference: Remote database access services offered as a result of the requirements of which this is a part shall conform to the requirements defined in "Remote Database Access (RDA)," ISO/IEC DP 9759; "Part 1: Generic Model, Service, and Protocol," JTC1/SC21N4282; and in "Part 2: SQL Specialization," JTC1/SC21N4281.

2.5 Data Interchange Services

Data interchange services establish data formats for interchange of documents, graphics data, and product description data.

	LOC	PAV	CMP	MAT	STB	DFU	PRL
ODA/ODIF/ODL	●		●		○		○
SGML	●		○	●	●		●
CGM	●	●	●	●	●	●	●
IGES	●	●		○	●	●	●
STEP			●				

LOC -- Level of consensus
PAV -- Product
CMP -- Completeness
MAT -- Maturity
STB -- Stability
DFU -- De facto usage
PRL -- Problems/limitations

Legend: ● -high evaluation
○ -average evaluation
blank -low evaluation

2.5.1 Document Interchange — Open Document Architecture/Open Document Interchange Format/Open Document Language (ODA/ODIF/ODL) ISO 8613:1989

Specification available from: ANSI

Publication date: May 1989

Sponsoring organization: ISO/IEC JTC1, CCITT

Applicability: Interchange of documents — ODA, Open Document Architecture, is an architecture that enables users to interchange the logical structure, content, presentation style and layout structure (the physical appearance) of documents from one application to another, or from an application to various output devices. ODIF, Open Document Interchange Format, is an ASN.1 (Abstract Syntax Notation One--ISO 8824:1987 and ISO 8825:1987) encoding for documents suitable for interchange between applications. ODL, Open Document Language, is a generic SGML encoding for documents suitable for interchange between applications.

Level of consensus: The international standard (ISO 8613) was approved by two international standards bodies, ISO and CCITT. Additionally ODA has been adopted for use by the Department of Defense for inclusion in the Computer-Aided Acquisition and Logistics Support (CALS) initiative for encoding of raster images.

Product availability: A few vendors have implemented a major subset of ODA (Level 2 in the specification) using ODIF as an interchange format.

Completeness: ODA/ODIF/ODL covers all aspects of document interchange, including logical structure, layout structure, generic logical structure, generic layout structure, and presentation. Documents can be interchanged in either a processable, formatted, or a processable formatted form.

Maturity: The connection between document logical structure, layout, and content is still an active topic of research projects. Models are incomplete.

Stability: Minor revisions of the standard will be made as vendors develop implementations.

De facto usage: If users do not reference this specification in procurement documents, vendors will probably propose products that do not meet this specification, or are not compatible with the specification.

Known problems/limitations: Unknown.

Conformance testing: None.

Future plans: A FIPS PUB is planned within the next year.

Alternative specifications: Electronic Manuscript Preparation and Markup, standard ANSI/NISO Z39.59-1988, is an alternate national standard for representing the logical structure of books, articles, and serials. Several organizations have designed alternate nonproprietary architectures with SGML encodings. Those organizations are the Association of American Publishers (AAP), the Text Encoding Initiative (TEI), and the Department of Defense CALS program. Many vendors still recommend that organizations require unique nonstandard document architectures encoded in SGML.

Suggested reference: Document interchange services offered as a result of the requirements of which this is a part shall conform to the requirements in ISO 8613:1989, Office Document Architecture (ODA).

2.5.2 Document Interchange — Standard Generalized Markup Language (SGML) FIPS PUB 152

Specification available from: NTIS, ANSI, GCA

Publication date: September 26, 1988

Sponsoring organization: ISO/IEC JTC1

Applicability: Interchange of documents — SGML is intended to formally define the grammar of languages for document markup. It provides a means to specify what markup is allowed, what markup is required, and how markup is distinguished from text.

Level of consensus: The FIPS PUB is based on national and international standard ANSI/ISO 8879:1986.

Product availability: Several implementations that use SGML encodings to parse their input are available from vendors.

Completeness: A high percentage of SGML features are available in current implementations. SGML does not deal with the meaning of the markup. (Markup consists of the common sets of document formatting codes used in classes of document types. For example, technical manuals may use a different markup from management guideline documents due to the audience and content of the respective document types, and the types of publishing layouts that are commonly used for each.) Therefore SGML does not specify what to do after the document has been processed by an SGML-recognizing program. Additional specifications such as Electronic Manuscript Preparation and Markup (EMPM) or ODA are needed to determine the markup's meaning. Other organizations that have produced or are in the process of developing specifications include the DoD Computer-Aided Acquisition and Logistic Support (CALS) program, the American Association of Publishers (AAP), and the Text Encoding Initiative (TEI).

Maturity: The technology upon which SGML is based has existed for a long time. Precursors of SGML include Backus Naur Form, Regular, Context Free, LR(k), and Context Sensitive grammars. These are well understood and have a rich mathematical basis.

Stability: The position as a grammar representation standard makes SGML a very stable document. It is generalized to the extent that various other representations and models can be included and represented within the SGML framework. The market is having difficulty, however, adopting any of the many possible SGML-encoded markup architectures as a basis for interchange. See Known problems/limitations.

De facto usage: If users do not reference this specification in procurement documents, vendors will probably propose products that do not meet this specification, or are not compatible with the specification.

Known problems/limitations: While consensus on the SGML standard has been reached to some degree, there is still a great deal of disagreement on particular markup to be employed.

Conformance testing: The Graphics Communication Association (GCA) is discussing plans to produce a conformance test suite. A prototype test suite has been developed by CSL.

Future plans: None.

Alternative specifications: ASN.1

Suggested reference: SGML systems offered as a result of the requirements of which this is a part shall conform to the requirements established in FIPS PUB 152, Standard Generalized Markup Language (SGML), and shall implement all of the language elements of SGML, as well as any additional language features as specified elsewhere in this document, and shall require validation in accordance with the provisions of FIPS PUB 152.

2.5.3 Graphics Data Interchange — Computer Graphics Metafile (CGM) FIPS PUB 128

Specification available from: NTIS

Publication date: March 16, 1987

Sponsoring organization: X3H3

Applicability: Graphics data interchange is specified in terms of a file format that can be created independently of device requirements and translated into the formats needed by specific output devices, graphics systems, and computer systems.

Level of consensus: The FIPS PUB is based on national (ANSI X3.122-1986) and international (ISO 8632) standards for neutral (implementation and machine independent) graphics file formats. Vendors commonly use CGM as an exchange format for the storage, interchange, or output of a wide range of graphical pictures (from slides for presentation graphics or business charts to diagrams generated by scientific applications). Most CGM implementations conform to the CALS Application Profile which is the DoD effort to standardize technical documents and engineering drawings.

Product availability: Numerous CGM implementations exist for use in Federal procurements. Virtually all major microcomputer software products can generate and/or interpret CGM files.

Completeness: CGM contains capabilities to describe and format virtually any type of picture or drawing.

Maturity: CGM research and development has been performed for at least the past 10 years.

Stability: Three addenda are now being considered by the ANSI and ISO committees which are responsible for CGM. These changes will be upwardly compatible with existing versions of the specification.

De facto usage: Even if users do not reference this specification in procurement documents, vendors will likely propose products that meet the specification or are compatible with it.

Known problems/limitations: Unknown.

Conformance testing: Conformance test suites are available from commercial, federal, and other sources. A test service to test for conformance to the CALS Application Profile will begin in 1991. The U. S. Government publishes an updated list of FIPS-conforming implementations.

Future plans: Three upward-compatible addenda to CGM are being considered. These are additions to add a global symbol capability, add 3-dimensional geometry extensions, and add improved engineering drawing capabilities, such as better control over fine details of line drawings.

Alternative specifications: None.

Suggested reference: All computer graphics metafiles acquired to describe, store, and/or communicate graphical (pictorial) information in vector format among different devices, systems and installations must be in compliance with the requirements set forth in FIPS PUB 128, Computer Graphics Metafile (CGM).

2.5.4 Product Data Interchange — Planned FIPS PUB for Initial Graphic Exchange Specification (IGES)

Specification available from: PDES

Publication date: Draft available.

Sponsoring organization: IGES/PDES

Applicability: Product data interchange encompasses technical drawings, documentation, and other data required for product design and manufacturing, including geometric and nongeometric data such as form features, tolerances, material properties, and surfaces. The information typically associated with computer-aided design and manufacturing (CAD/CAM) can be described. IGES does not cover the complete life-cycle of manufactured products: it addresses only the specification of products; not the manufacturing process relationships.

Level of consensus: The specification was originally defined in National Bureau of Standards Interim Report (NBSIR) 88-3813. It has been defined as ANSI standard, ANSI Y14.26-1989, by the American Society of Mechanical Engineers (ASME) and is used in the DoD CALS initiative to specify transmission of technical documents and engineering drawings in a device independent manner.

Product availability: Numerous implementations of IGES are available in the marketplace.

Completeness: IGES defines the representation of engineering data, but does not include all interfaces for use, such as the interface between the data specification and numerically-controlled machining tools.

Maturity: The concepts of IGES have been in existence since before computers and programming were developed. The processes of machine numerical control have been defined and enhanced in direct relation to the requirements for defining and using exchange specifications for engineering data.

Stability: No substantial changes are foreseen in the near term. As the specification advances with newer versions, compatibility will be maintained.

De facto usage: Even if users do not reference this specification in procurement documents, vendors will likely propose products that meet the specification or are compatible with it.

Known problems/limitations: Not all interfaces between the data exchange specification and external components, such as user interface and machine interfaces have been defined.

Conformance testing: None.

Future plans: Version 5.0 is only being released as a NIST Interim Report (NISTIR) and does not contain B-rep solids. Version 5.1 will be released when B-rep solids are ready. Version 5.1 will be processed through ANSI/ASME to become an ANSI standard.

Alternative specifications: STEP (See sec. 2.5.5).

Suggested reference: Product description services offered as a result of the requirements of which this is a part shall conform to the requirements in ANSI Y14.26-1989, Initial Graphics Exchange Specification (IGES).

2.5.5 Product Data Interchange — Standard for the Exchange of Product Model Data (STEP) Draft Proposed ISO 10303

Specification available from: ISO TC184/SC4 Secretariat (NIST)

Publication date: Draft available.

Sponsoring organization: ISO

Applicability: STEP is used in total lifecycle descriptions of engineered products that can be implemented on advanced manufacturing systems. This includes specification of products from initial conception through process control of the manufacturing of the product.

Level of consensus: The specification is defined in ISO draft proposed International Standard 10303. (STEP was previously known as Product Data Exchange Specification [PDES], but the

name of the proposed standard was changed to differentiate it from PDES which is actually the initiative that is creating STEP and is now called Product Data Exchange using STEP.)

Product availability: Vendors are engaged in development of prototype implementations of small subsets of the specification.

Completeness: The standard defines a complete product lifecycle including all aspects of describing technical diagrams and documents in a neutral format for transmission over communications networks and processing by numerically-controlled machining and assembly tools.

Maturity: STEP was initially built on the concepts of IGES and was extended to include the full lifecycle of products from initial requirements and design through final production and installation.

Stability: STEP is still in a draft stage and may undergo revision at any time. Many of the component specifications have not been defined (i. e., early 1992 is projected as the goal for the majority of the component specifications to be ready).

De facto usage: If users do not reference this specification in procurement documents, vendors will probably propose products that do not meet this specification, or are not compatible with the specification.

Known problems/limitations: Many of the component specifications have not yet been defined.

Conformance testing: None.

Future plans: STEP will be proposed as an international standard when full agreement has been reached. Version 2.0 has been started to provide corrections and additions for known deficiencies in Version 1.0.

Alternative specifications: IGES

Suggested reference: Product description lifecycle services offered as a result of the requirements of which this is a part shall conform to the requirements in Draft Proposed ISO 10303 Standard for the Exchange of Product Model Data (STEP).

2.6 Graphics Services

Graphics services provide the interfaces for programming two- and three-dimensional graphics in a device-independent manner. The specifications included in this service area are the Graphical Kernel System (GKS) FIPS PUB 120, and the Programmer's Hierarchical Interactive Graphics System (PHIGS) FIPS PUB 153. They are targeted at different types of users and applications.

	LOC	PAV	CMP	MAT	STB	DFU	PRL
GKS	●	●	●	●	●	●	●
PHIGS	●	●	●	○	●	●	●

Legend: ● -high evaluation
○ -average evaluation
blank -low evaluation

LOC -- Level of consensus
PAV -- Product availability
CMP -- Completeness
MAT -- Maturity
STB -- Stability
DFU -- De facto usage
PRL -- Problems/limitations

2.6.1 Graphics Services --- Graphical Kernel System (GKS) FIPS PUB 120-1

Specification available from: NTIS

Publication date: January 8, 1991

Sponsoring organization: X3H3

Applicability: This specification fulfills the requirement for a language to program 2-dimensional graphical objects that will be displayed or plotted on appropriate devices (raster graphics and vector graphics devices).

Level of consensus: The GKS FIPS PUB is based on national (ANSI X3.124-1985) and international (ISO 7942:1985) standards. Bindings for Ada, Fortran, and Pascal have been defined and standardized.

Product availability: A full range of products and automated tools based on GKS has been available from various vendors for five or more years.

Completeness: The standard includes constructs and library calls for virtually any kind of two-dimensional graphic image.

Maturity: Initial work started on this specification in 1978 and has been developed substantially by international organizations in the ensuing years. It was founded on a graphics standards methodology developed in 1976.

Stability: No changes are foreseen.

De facto usage: Even if users do not reference this specification in procurement documents, vendors will likely propose products that meet the specification or are compatible with it.

Known problems/limitations: GKS is limited to two-dimensional graphics.

Conformance testing: NIST has licensed a conformance test suite for GKS. Using this test suite, NIST is currently operating a GKS Test Service to test implementations for conformance to the FIPS PUB.

Future plans: The GKS test suite will undergo revision as warranted.

Alternative specifications: PHIGS (see sec. 2.6.2)

Suggested reference: All two-dimensional graphics libraries/packages to be used as a programming interface to application programs offered as a result of a requirement in this requirements document shall comply with FIPS PUB 120-1, Graphical Kernel System (GKS).

2.6.2 Graphics Services — Programmer's Hierarchical Interactive Graphics System (PHIGS) FIPS PUB 153

Specification available from: NTIS

Publication date: October 14, 1988

Sponsoring organization: X3H3

Applicability: This specification fulfills the requirement for a language to program 2- and 3-dimensional graphical objects that will be displayed or plotted on appropriate devices in interactive, high-performance environments, and for managing hierarchical database structures containing graphics data.

Level of consensus: The FIPS PUB is based on national (ANSI X3.144-1988 and X3.144.1-1988) and international (ISO 9592:1988) standards.

Product availability: Numerous implementations are available for various hardware/software platforms.

Completeness: PHIGS is a full-functioned specification for the development of interactive two- and three-dimensional graphics applications which manage hierarchical database structures containing graphics data. Bindings for Fortran and Ada have been adopted.

Maturity: Many of the concepts for this standard were drawn from previous work. Chief among those works are the Association for Computing Machinery (ACM) Siggraph Graphics Planning Committee Core Graphics System and the American National Standard Graphical Kernel System (GKS) X3.124-1985.

Stability: No changes are planned in the next 1 to 3 years.

De facto usage: Even if users do not reference this specification in procurement documents, vendors will likely propose products that meet the specification or are compatible with it.

Known problems/limitations: Unknown

Conformance testing: NIST is currently developing a PHIGS Test Suite which tests implementations for conformance to the FIPS PUB. Version 1 of this test suite is currently available. The complete test suite is expected to be completed in 1991 at which time certificates will be issued to implementations passing the tests.

Future plans: Bindings for C and Pascal are under development. A new standard, called PHIGS Plus, is being developed which adds shading, lighting, and other advanced graphics programming capabilities that were not included in PHIGS. Conforming PHIGS programs will be able to execute under PHIGS Plus with no change.

Alternative specifications: None.

Suggested reference: All computer graphics software toolbox packages acquired to support very highly interactive graphics applications or needing rapid modification of both the graphics data and the relationships between the graphical data must be in compliance with the functional requirements as set forth in FIPS PUB 153, Part 1, the Programmer's Hierarchical Interactive Graphics System (PHIGS). In addition, such toolbox packages must support, at a minimum, one of the programming language bindings as specified in FIPS PUB 153, Part 2.

2.7 Network Services

This area of the APP includes data communications, transparent file access, personal/microcomputer support, and distributed computing support.

	LOC	PAV	CMP	MAT	STB	DFU	PRL
GOSIP II	●	○	●	●	○	○	○
TFA		○					
OSF/1 NCS/RPC							

LOC -- Level of consensus
PAV -- Product availability
CMP -- Completeness
MAT -- Maturity
STB -- Stability
DFU -- De facto usage
PRL -- Problems/limitations

Legend: ● -high evaluation ○ -average evaluation blank -low evaluation

2.7.1 Data Communications — Government Open System Interconnection Profile (GOSIP Version 2.0) FIPS PUB 146-1

Specification available from: NTIS

Publication date: April 1991

Sponsoring organization: OSI Implementor's Workshop

Applicability: GOSIP is applicable to all data communications environments where multi-vendor computing is anticipated. GOSIP is based on Open Systems Interconnection (OSI) standards, the world-wide consensus standards for multivendor data communications. The services identified in the OSI communications arena as used by the U. S. Government are based on GOSIP protocols. The GOSIP protocols provide interoperability among applications in a heterogeneous network, while the GOSIP APIs provide portability of applications across heterogeneous platforms in a network.

GOSIP mandates no service interface accessibility beyond that indicated in the Workshop Agreements; therefore, any additional service interface accessibility requirements must be clearly stated and mandated in procurement documentation. For example, GOSIP mandates no specific direct access to transport services. If the procuring authority requires direct access to transport services, such a requirement must be included in a solicitation. The issues involved in determining such a requirement are complex. Refer to the Government Open System Interconnection Profile (GOSIP) Users' Guide for a discussion of these issues.

Level of consensus: The GOSIP FIPS is based on the ISO and CCITT international standards, implementors agreements developed at the NIST OSI Implementors' Workshop, and U. S. Government requirements. (The OSI Implementors Agreements are specified in NIST Special Publication 500-183, Stable Implementation Agreements for Open Systems Interconnection Protocols Version 4 Edition 1 December 1990, and NISTIR 4448, Working Implementation Agreements for Open Systems Interconnection Protocols.) DoD has mandated GOSIP use in all computer communications procurements for all services. In addition, the Departments of Energy and Veterans Affairs, as well as other Federal agencies, have specified the use of GOSIP in all future procurements.

Product availability: Various products implementing specific protocol levels of the OSI layered model have been produced and conform to the FIPS PUB. Vendors are given an eighteen month period between promulgation of a new version of GOSIP and the date that it must be referenced in Federal procurements to ensure that new products will be available.

Completeness: GOSIP is essentially a family of protocols and representations. It provides a complete transparent, end-to-end data communications capability based on OSI transport class 4 (TP4) and connectionless network protocol (CLNP). Version 1.0 provides electronic mail and file transfer access and management applications. It operates over a variety of local- and wide-area network technologies. Version 2.0 adds remote logon and office document interchange applications, a new network addressing structure to support dynamic routing, provision to operate over an Integrated Services Digital Network (ISDN), and allows an optional connectionless transport service to support transparent file access (see sec. 2.7.2) and other applications.

Maturity: The OSI standards have been developed over the last 10 years and are based on a well-understood reference model, the Open Systems Interconnection (OSI) seven-layer communications model. As of August 15, 1990, GOSIP Version 1.0 is a mandatory requirement in Federal information technology procurements. GOSIP Version 2.0 will be required as of October 1992.

Stability: Significant changes are foreseen, mostly in the area of additional functionality. New versions of existing protocols will be upwardly compatible with current versions. Additionally, all of the APIs, except FTAM, are now in draft stages and will become standards within the next 10 to 12 months. FTAM has been defined functionally, but has no acceptable interface specification. A standard for FTAM is expected within the next 18 months.

Much of the work in the draft specifications was based on documents developed by X/Open, the Application Programming Interface Association (APIA), and other vendor consortia. Prototypes of the draft interfaces have been developed, and several full implementations are expected to be available within a year. Vendors are now producing proprietary implementations that contain much

of the functionality required, but may not exhibit APIs that are closely aligned with the draft specifications. Within 2 years, numerous implementations of standardized APIs should be available.

De facto usage: Even if users do not reference this specification in procurement documents, vendors will likely propose products that meet the specification or are compatible with it. (Notwithstanding the fact that TCP/IP is widely used in Government communications, especially DoD networks, GOSIP is mandated for all Federal communications procurements after August 1990.)

Known problems/limitations: The seventh and most abstract layer of the OSI model, the application layer, has not been fully described and implemented. There are still questions about how to best present application interfaces.

Conformance testing: NIST has developed a program for ensuring conformance to the FIPS PUB and for ensuring interoperability. Lists of GOSIP-conformant products will be published and updated periodically.

Future plans: Later versions of GOSIP will include substantial added functionality such as distributed directory services, transaction processing, remote database access, dynamic routing, network management, and network security. Key features may vary with specific combinations of vendor products and users. NIST is producing guidelines for users to evaluate which applications are best for individual requirements.

The APIs that will be recommended for use with GOSIP are tentatively: Interprocess Communication Over Networks (ICON) for allowing separate processes executing on the same or different platforms to communicate data between them (will be available in place of mechanisms such as sockets, XTI, TLI, etc. that are currently found in proprietary implementations); Directory Services for determining the locations of various components in a network (e. g., user locations, organization locations, file locations, etc.); Message Handling Service (MHS) for formatting and processing messages in a store-and-forward mode (e. g., electronic mail capabilities); and File Transfer Access and Management (FTAM) capabilities for communicating the contents of various types of files (e. g., text files, binary load modules, etc.). Work on these specifications is continuing, but final determination of APIs to be included in future releases of GOSIP has not been made.

Alternative specifications: None.

Suggested reference: All computer network and communications equipment, systems, and services offered as a result of this requirement must provide the protocols specified herein and defined in the standards contained in FIPS PUB 146-1, GOSIP: Government Open System Interconnection Profile. (Note: To determine applicability to FIPS PUB 146-1, refer to NIST Special Publication 500-163, Government Open System Interconnection Profile (GOSIP) Users' Guide. A complete specification of GOSIP requirements should include references to specific protocols and APIs.)

2.7.2 Transparent File Access (TFA) — IEEE P1003.8 Draft 4

Specification available from: IEEE

Publication date: Draft available.

Sponsoring organization: IEEE

Applicability: Transparent file access includes capabilities for managing files and transmitting data through heterogeneous networks in a manner that is transparent (i. e., does not require knowledge of file location or of certain access requirements) to the user. In a GOSIP environment, TFA should be based on the services provided by FTAM.

Level of consensus: A specification will be available in late 1991.

Product availability: Many functions of TFA are widely available in existing vendor implementations of network oriented file systems and have interfaces that closely resemble the TFA interface. (These implementations may or may not be based on FTAM services.)

Completeness: The specification is in draft form but is essentially complete. It is still subject to modification. It will eventually include all functionality found in current proprietary implementations that are available from commercial organizations.

Maturity: Much research on distributed file access and file systems has been performed and published over the last ten years. As a consequence, many of the problems of distributed files and file systems have become known and various solutions have been developed. There are still areas of distributed file access, such as global address resolution, concurrency, and security that will have profound effects on TFA functionality. In some cases, the capabilities of hardware and software that are available today cannot support the requirements of TFA in all cases. A minimal set of functionality has been identified.

Stability: Significant changes in the specification are unlikely, but numerous smaller changes are anticipated. This could have a cascading effect on other parts of the specification. Until consensus on the minor aspects of major features is reached, one must consider the specification in a state of flux.

De facto usage: If users do not reference this specification in procurement documents, vendors will probably propose products that do not meet this specification, or are not compatible with the specification.

Known problems/limitations: Currently, there is no specification for the file system semantics which result from most implementations of systems with TFA-like features. Such systems are usually referred to by the protocols which each implementation uses (e. g., NFS, RFS, AFS, NCS). The eventual TFA specification should overcome this limitation.

Conformance testing: None.

Future plans: A ballot on a draft standard will be undertaken by IEEE sometime near the end of 1991.

Alternative specifications: None.

Suggested reference: Transparent file access services offered as a result of the requirements of which this is a part shall conform to the requirements defined in Transparent File Access (TFA), IEEE 1003.8.

2.7.3 Distributed Computing Services — OSF/1 Network Computing Services (NCS) Remote Procedure Call (RPC)

Specification available from: Open Software Foundation (OSF)

Publication date: Draft available.

Sponsoring organization: OSF

Applicability: Distributed computing services include specifications for remote procedure calls and distributed realtime support in heterogeneous networks (as opposed to single node support as specified in operating system services). Distributed access services include functional support for submitting, starting, and stopping processes among processors in a heterogeneous network.

Level of consensus: OSF is developing a standard for use by its members.

Product availability: Vendor partial implementations are available and based on the OSF specification.

Completeness: No specifications exist that define a complete set of functions necessary to provide remote procedure communications for all types of application platforms (i. e., the language independent representation of remote procedure calls).

Maturity: In general, OSF specifications are based on object-oriented structures and relationships. The underlying services and data formats are well-established, but the objects to be managed are still evolving.

Stability: Other industry consortia are reviewing the possibility of adopting NCS/RPC. Other specifications are emerging as possible alternatives.

De facto usage: If users do not reference this specification in procurement documents, vendors will probably propose products that do not meet this specification, or are not compatible with the specification.

Known problems/limitations: The specification is incomplete and still in a draft state.

Conformance testing: None.

Future plans: The specification will continue to evolve as research and case histories become available for determining a predominant set of managed objects.

Alternative specifications: ONC RPC (Open Network Computing Remote Procedure Call)

Suggested reference: Remote procedure call (RPC) services offered as a result of the requirements of which this is a part shall conform to the requirements defined in OSF/1 "Network Computing Services (NCS)—Remote Procedure Call (RPC)," and shall work in concert with and support FIPS PUB 151-1 POSIX.

3. STRATEGIC EVALUATIONS

As part of the evaluation of APP specifications, users should take into account the strategic value of each specification. Table 1 summarizes NIST's views on the strategic value of each specification recommended in this report.

The valuations are made according to the following guidelines:

- Strategic now (STR)—In selecting these specifications, users would be reasonably safe in making substantial investment and long-term plans covering mission-critical systems and the infrastructure needed to support them. Changes are expected to be upwardly-compatible.

- Strategic in the future (FTR)—Specifications that are subject to change but appear to be headed for standardization fall into this category. Existing standards that may be subject to changes that are not entirely upwardly-compatible also fall into this category. There are some long-term risks involved, but the actions of the consensus-building process will tend to minimize them. Users should select these specifications where strategic specifications are unavailable and an investment must be made, but should plan for possible evolution in the future.

- Nonstrategic (GAP)—These specifications are stop-gap measures recommended with the warning that any user investment will be at significant risk. They are not appropriate for long-term planning. Users should, for these reasons, minimize their risk by minimizing investment.

Subsequent versions of this report may incorporate this dimension of evaluation in the overall evaluation criteria.

TABLE 1. Strategic Value of APP Specifications

	STR	FTR	GAP
OPERATING SYSTEM SERVICES			
Portable Operating System Interface for Computer Environments FIPS PUB 151-1	●		
NIST Planned FIPS PUB on POSIX Shell and Utility Application Interface for Computer Operating System Environments IEEE P1003.2 Draft 11		●	
NIST Planned FIPS PUB on Government Network Management Profile (GNMP)		●	
Security Interface for the Portable Operating System Interface for Computer Environments IEEE P1003.6 Draft 6		●	
USER INTERFACE SERVICES			
User Interface Component of Applications Portability Profile FIPS PUB 158	●		
Extensible Virtual Toolkit (XVT)			●
PROGRAMMING SERVICES			
Ada FIPS PUB 119	●		
C FIPS PUB 160	●		
COBOL FIPS PUB 021-3	●		
Fortran FIPS PUB 069-1	●		
Pascal FIPS PUB 109	●		
ECMA Portable Common Tools Environment (PCTE)		●	
Source Code Control System (SCCS)			●
DATA MANAGEMENT SERVICES			
Information Resource Dictionary System (IRDS) FIPS PUB 156		●	
Database Language SQL FIPS PUB 127-1	●		
Remote Database Access (RDA)		●	
DATA INTERCHANGE SERVICES			
Office Document Architecture/Office Document Interchange Format (ODA/ODIF) ISO 8613:1989	●		
Standard Generalized Markup Language (SGML) FIPS PUB 152	●		
Computer Graphics Metafile (CGM) FIPS PUB 128	●		
Planned FIPS Initial Graphic Exchange Specification (IGES)			●
Standard for the Exchange of Product Model Data (STEP) Draft Proposed ISO 10303		●	
GRAPHICS SERVICES			
Graphical Kernel System (GKS) FIPS PUB 120		●	
Programmer's Hierarchical Interactive Graphics System (PHIGS) FIPS PUB 153	●		
NETWORK SERVICES			
Government Open System Interconnection Profile (GOSIP) FIPS PUB 146-1	●		
Transparent File Access (TFA) IEEE P1003.8 Draft 4		●	
OSF/1 Network Computing Services Remote Procedure Call (NCS/RPC)			●

4. INFORMATION SOURCES

The following organizations are responsible for distributing standards for various standards-making organizations. Ordering and fee information for specific standards may be obtained directly from the addressees.

AAP
Association of American Publishers
EPSIG (Electronic Publishing Special Interest Group)
c/o OCLC
6565 Frantz Road
Dublin, OH 43017-0702
Phone: (614) 764-6000

ANSI
American National Standards Institute
1430 Broadway
New York, NY 10018
Phone: (212) 354-3300

ANSI International Publications
Information on standards from ISO and its member bodies (e. g., DIN, BSI, JISC), IEC, and CEN/CENELEC
Phone: (212) 642-4995

ANSI General Sales (National Standards)
Phone: (212) 642-4900

CCITT
International Telegraph and Telephone Consultative Committee
Place des Nations
CH-1211 Geneva 20
Switzerland

COSMIC
The University of Georgia
382 East Broad Street
Athens, GA 30602
Phone: (404) 542-3265
FAX: (404) 542-4807

ECMA
European Computer Manufacturers Association
Rue du Rhone 114
CH-1204 Geneva
Switzerland
Phone: 011-41-22-735-36-34

Federal Information Processing Standards (FIPS)
U. S. Department of Commerce
National Technical Information Service (NTIS)
Springfield, VA 22161
Phone: (703) 487-4650
FAX: (703) 321-8547
NIST publishes an index of FIPS that is available through NTIS. Request "NIST Publications List 58."

GCA
Graphic Communications Association
Suite 604
1730 North Lynn Street
Arlington, VA 22209-2085
Phone: (703) 841-8160

GPO
Government Printing Office
Superintendent of Documents
U. S. Government Printing Office
Washington, DC 20402
Phone: (202) 783-3238

IEC
International Electrotechnical Commission
3 Rue de Varembe
P. O. Box 131
CH-1211 Geneva 20
Switzerland
Phone: 011-41-22-34-01-50

IEEE (for accepted standards)
The Institute of Electrical and Electronics Engineers, Inc.
445 Hoes Lane
P. O. Box 1331
Piscataway, NJ 08855-1331
Phone: (201) 562-3800

IEEE (for draft standards)
1730 Massachusetts Avenue, N. W.
Washington, DC 20036-1903
Phone: (202) 371-0101

ISO
International Organization for Standardization
Central Secretariat
1 Rue de Varembe
P. O. Box 56
CH-1211 Geneva 20
Switzerland
Phone: 011-41-22-34-12-40

JTC1 TAG
Joint Technical Committee 1 Technical Advisory Group
311 First Street NW, Suite 500
Washington, DC 20001
Phone: (202) 737-8888 (Press 1 twice.)

National Technical Information Service (NTIS)
U. S. Department of Commerce
National Technical Information Service (NTIS)
Springfield, VA 22161
Phone: (703) 487-4650
FAX: (703) 321-8547

OSF
Open Software Foundation
11 Cambridge Center
Cambridge, MA 02142

SCCS/SVID
AT&T Customer Information Center (CIC)
Attn: Customer Service Representative
P. O. Box 19901
Indianapolis, IN 46219

SQL-Access
SQL Access Group
c/o Robert Crutchfield
Fransen and Associates, Inc.
2171 Campus Drive, Suite 260
Irvine, CA 92715
Phone: (714) 752-5942

UniForum
2901 Tasman Drive, #201
Santa Clara, CA 95054
Phone: (800) 255-5620 or (408) 986-8840
FAX: (408) 986-1645

UNIX International (UI)
Waterview Corporate Centre
20 Waterview Boulevard
Parsippany, NJ 07054
Phone: (800) 848-6495 or (201) 263-8400
FAX: (201) 263-8401

X3
Technical Committee X3 -- Information Processing Systems
Computer and Business Equipment Manufacturers Association (CBEMA)
Director, X3 Secretariat
311 First Street NW, Suite 500
Washington, DC 20001
Phone: (202) 737-8888 (Press 1 twice.)

X/OPEN — X/OPEN Portability Guide (XPG)
1750 Montgomery Street
San Francisco, CA 94111
Phone: (415) 323-7992

XVT - Extended Virtual Toolkit
XVT Software, Inc.
P. O. Box 17665
Boulder, CO 80308
Phone: (303) 443-4223

5. CONCLUSION

The long term goal of the program on which this report is based is the establishment of an open system environment for use in federal information systems support. In this open system environment, interoperability, portability, and scalability must be the driving forces for the development of standard interfaces, services, protocols, and formats to support the OSE. Eventually, we would like to see all of the OSE specifications take the form of FIPS. In the interim, we have reviewed many of the specifications that are now available and have made recommendations on those that we believe have a higher probability of becoming successful additions to the suite of OSE specifications.

The short term goals of federal information requirements require action now. In response to these goals, we have developed a suite of specifications that can be used in system development and procurements. There is, however, a measure of risk involved in using some of these specifications. Some of them are Federal standards, and others are national or international standards. These specifications are very stable and can be used with very little risk. A few are taken from standards work-in-progress and could change significantly before they become standards. Therefore, the risk is somewhat greater. A third group of specifications is based on nonopen specifications. This is a very risky group of specifications in that the federal user has very little control in what could happen to the specifications over time.

The tradeoffs amount to accepting less portability and interoperability in return for meeting current information requirements and not waiting until all open system specifications become available.

This state of affairs is possibly comparable to walking on a frozen river: in some places, it is safe to walk; in others, one must tread carefully. No clearly right or wrong decisions will be made in selecting specifications. Some decisions will be more right than others. Users can only hope with today's technology to ameliorate the effects of long-term changes. NIST will continue to perform evaluations and make recommendations. *Users must do the same.*

BIBLIOGRAPHY

An Analysis of Application Environments, Emerging Technologies Group, Inc., Dix Hills, NY, 1989, pp. 494.

Guide to the POSIX Open Systems Environment, Draft 11, IEEE P1003.0, March 1991.

Strategies for Open Systems — Stage Two — The Experience With Open Systems, DMR Group, Inc., Boston, 1990, pp. 196.

The Open Systems Directive, Draft 0.4, X/Open Company, Ltd., Berkshire, United Kingdom, 1990, pp. 346.

X/Open Portability Guide, Issue 3, Volumes 1-7, X/Open Company, Ltd., Prentice Hall, Inglewood Cliffs, NJ, 1988.

ACRONYMS

ANSI -- American National Standards Institute

API -- Application programming interface

APP -- Application Portability Profile

CALS -- Computer-Aided Acquisition and Logistic Support

CASE -- Computer aided software engineering (See ISEE)

CGM -- Computer Graphics Metafile

CSL -- Computer Systems Laboratory (part of NIST)

DBMS -- Database management system

ECMA PCTE -- European Computer Manufacturers Association Portable Common Tools Environment

FIPS -- Federal Information Processing Standard

GKS -- Graphical Kernel System

GOSIP -- Government Open System Interconnection Profile

IEEE -- Institute of Electrical and Electronic Engineers

IGES -- Initial Graphic Exchange Specification

IRDS -- Information Resource Dictionary System

ISO -- International Organization for Standardization

ISEE -- Integrated software engineering environment

NIST -- National Institute of Standards and Technology

NTIS -- National Technical Information Service

NVLAP -- National Voluntary Laboratory Accreditation Program (A NIST program)

ODA/ODIF -- Open Document Architecture/Open Document Interchange Format

ODL -- Open Document Language

OSE -- Open System Environment

OSF -- Open Software Foundation

OSI -- Open System Interconnection

PDES -- Product Data Exchange using STEP

PHIGS -- Programmer's Hierarchical Interactive Graphics System

POSIX -- Portable Operating System Interface for Computer Environments

RDA -- Remote Database Access

SGML -- Standard Generalized Markup Language

SQL -- Database Language SQL

STEP -- Standard for the Exchange of Product Model Data

TFA -- Transparent File Access

INDEX

NIST-114A
(REV. 3-90)

U.S. DEPARTMENT OF COMMERCE
NATIONAL INSTITUTE OF STANDARDS AND TECHNOLOGY

BIBLIOGRAPHIC DATA SHEET

1. PUBLICATION OR REPORT NUMBER
NIST/SP-500/187

2. PERFORMING ORGANIZATION REPORT NUMBER

3. PUBLICATION DATE
April 1991

4. TITLE AND SUBTITLE

Application Portability Profile - APP - The U.S. Government's Open System Environment Profile. OSE/1 Version 1.0

5. AUTHOR(S)

Gary Fisher

6. PERFORMING ORGANIZATION (IF JOINT OR OTHER THAN NIST, SEE INSTRUCTIONS)

U.S. DEPARTMENT OF COMMERCE
NATIONAL INSTITUTE OF STANDARDS AND TECHNOLOGY
GAITHERSBURG, MD 20899

7. CONTRACT/GRANT NUMBER

8. TYPE OF REPORT AND PERIOD COVERED
Final

9. SPONSORING ORGANIZATION NAME AND COMPLETE ADDRESS (STREET, CITY, STATE, ZIP)

Same as item #6

10. SUPPLEMENTARY NOTES

11. ABSTRACT (A 200-WORD OR LESS FACTUAL SUMMARY OF MOST SIGNIFICANT INFORMATION. IF DOCUMENT INCLUDES A SIGNIFICANT BIBLIOGRAPHY OR LITERATURE SURVEY, MENTION IT HERE.)

An Open System Environment (OSE) encompasses the functionality needed to provide interoperability, portability, and scalability of computerized applications across networks of heterogeneous hardware/software platforms. The OSE forms an extensible framework that allows interfaces, services, protocols, and supporting data formats to be defined in terms of nonproprietary specifications that evolve through open (public), consensus-based forums. A selected suite of specifications that define these interfaces, services, protocols, and data formats for a particular class or domain of applications is called a profile. The Application Portability Profile (APP) is the U. S. Government's OSE profile. It was developed to provide functionality across a broad range of Federal applications. This report describes the service areas and components included in the APP and provides evaluations of recommended specifications for the majority of the service area components. Organizations should use this report to assist in determining which specifications may be applicable to their particular environments.

12. KEY WORDS (6 TO 12 ENTRIES; ALPHABETICAL ORDER; CAPITALIZE ONLY PROPER NAMES; AND SEPARATE KEY WORDS BY SEMICOLONS)

APP, computer standards, FIPS, GOSIP, open system, operating system, OSE, portability, profile, user interface

13. AVAILABILITY

X	UNLIMITED
	FOR OFFICIAL DISTRIBUTION. DO NOT RELEASE TO NATIONAL TECHNICAL INFORMATION SERVICE (NTIS).
X	ORDER FROM SUPERINTENDENT OF DOCUMENTS, U.S. GOVERNMENT PRINTING OFFICE, WASHINGTON, DC 20402.
X	ORDER FROM NATIONAL TECHNICAL INFORMATION SERVICE (NTIS), SPRINGFIELD, VA 22161.

14. NUMBER OF PRINTED PAGES
74

15. PRICE

ELECTRONIC FORM

* U.S. G.P.O.:1991-281-557:40032

ANNOUNCEMENT OF NEW PUBLICATIONS ON COMPUTER SYSTEMS TECHNOLOGY

Superintendent of Documents
Government Printing Office
Washington, DC 20402

Dear Sir:

Please add my name to the announcement list of new publications to be issued in the series: National Institute of Standards and Technology Special Publication 500-.

Name _____

Company _____

Address _____

City _____ State _____ Zip Code _____

(Notification key N-503)

NIST Technical Publications

Periodical

Journal of Research of the National Institute of Standards and Technology—Reports NIST research and development in those disciplines of the physical and engineering sciences in which the Institute is active. These include physics, chemistry, engineering, mathematics, and computer sciences. Papers cover a broad range of subjects, with major emphasis on measurement methodology and the basic technology underlying standardization. Also included from time to time are survey articles on topics closely related to the Institute's technical and scientific programs. Issued six times a year.

Nonperiodicals

Monographs—Major contributions to the technical literature on various subjects related to the Institute's scientific and technical activities.

Handbooks—Recommended codes of engineering and industrial practice (including safety codes) developed in cooperation with interested industries, professional organizations, and regulatory bodies.

Special Publications—Include proceedings of conferences sponsored by NIST, NIST annual reports, and other special publications appropriate to this grouping such as wall charts, pocket cards, and bibliographies.

Applied Mathematics Series—Mathematical tables, manuals, and studies of special interest to physicists, engineers, chemists, biologists, mathematicians, computer programmers, and others engaged in scientific and technical work.

National Standard Reference Data Series—Provides quantitative data on the physical and chemical properties of materials, compiled from the world's literature and critically evaluated. Developed under a worldwide program coordinated by NIST under the authority of the National Standard Data Act (Public Law 90-396). NOTE: The Journal of Physical and Chemical Reference Data (JPCRD) is published quarterly for NIST by the American Chemical Society (ACS) and the American Institute of Physics (AIP). Subscriptions, reprints, and supplements are available from ACS, 1155 Sixteenth St., NW., Washington, DC 20056.

Building Science Series—Disseminates technical information developed at the Institute on building materials, components, systems, and whole structures. The series presents research results, test methods, and performance criteria related to the structural and environmental functions and the durability and safety characteristics of building elements and systems.

Technical Notes—Studies or reports which are complete in themselves but restrictive in their treatment of a subject. Analogous to monographs but not so comprehensive in scope or definitive in treatment of the subject area. Often serve as a vehicle for final reports of work performed at NIST under the sponsorship of other government agencies.

Voluntary Product Standards—Developed under procedures published by the Department of Commerce in Part 10, Title 15, of the Code of Federal Regulations. The standards establish nationally recognized requirements for products, and provide all concerned interests with a basis for common understanding of the characteristics of the products. NIST administers this program as a supplement to the activities of the private sector standardizing organizations.

Consumer Information Series—Practical information, based on NIST research and experience, covering areas of interest to the consumer. Easily understandable language and illustrations provide useful background knowledge for shopping in today's technological marketplace.

Order the **above** *NIST publications from: Superintendent of Documents, Government Printing Office, Washington, DC 20402.*

Order the **following** *NIST publications—FIPS and NISTIRs—from the National Technical Information Service, Springfield, VA 22161.*

Federal Information Processing Standards Publications (FIPS PUB)—Publications in this series collectively constitute the Federal Information Processing Standards Register. The Register serves as the official source of information in the Federal Government regarding standards issued by NIST pursuant to the Federal Property and Administrative Services Act of 1949 as amended, Public Law 89-306 (79 Stat. 1127), and as implemented by Executive Order 11717 (38 FR 12315, dated May 11, 1973) and Part 6 of Title 15 CFR (Code of Federal Regulations).

NIST Interagency Reports (NISTIR)—A special series of interim or final reports on work performed by NIST for outside sponsors (both government and non-government). In general, initial distribution is handled by the sponsor; public distribution is by the National Technical Information Service, Springfield, VA 22161, in paper copy or microfiche form.

U.S. Department of Commerce
National Institute of Standards and Technology
Gaithersburg, MD 20899