OPEN SYSTEM ENVIRONMENT FOR TRAINING SIMULATIONS

Benjamin D. Blood, Jr. and Philip W. Holden
Coleman Research Corporation
Orlando, Florida

ABSTRACT

A review of what constitutes an evolving open system environment for the U.S. Government as recommended by National Institute of Standards and Technology (NIST) in the Application Portability Profile (APP) and by Defense Information Systems Agency (DISA) in the Technical Reference Model for Information Management is presented. Its applicability as technical guidance to DoD components for the acquisition, development, and support of Department of Defense (DoD) training systems and simulations is described. The choices or recommendations made to implement some of these standards on an actual Army project is discussed. The project is concerned with the implementation of a simulation interface system for the Army War College.

ABOUT THE AUTHORS

Ben Blood is a Delivery Order Manager and Senior Systems Engineer at Coleman Research Corporation. His work has focused recently on providing engineering support services to the Army's Simulation, Training and Instrumentation Command (STRICOM) for concept formulation for the Closed Loop Artillery Simulation System (CLASS) and development program support for the Family of Battle Simulations (FAMSIM). In particular, studies and analyses have been performed for FAMSIM regarding an open system environment. He has a BSEE from Purdue University and an MS in Systems Engineering from the University of Arizona.

Phil Holden is a Senior Engineer at Coleman Research Corporation. His work has focused recently on being the lead engineer on a project to design, build and install a simulation and modeling software interface system for the U.S. Army War College (AWC). Key to this project has been his design choices regarding implementation of open systems standards and usage of their related off-the-shelf products. Prior to this project, he was part of the teams working on the CLASS and FAMSIM projects. He has a BS in Engineering from the U.S. Military Academy and an MBA from Oklahoma City University.

OPEN SYSTEM ENVIRONMENT FOR TRAINING SIMULATIONS

Benjamin D. Blood, Jr. and Philip W. Holden
Coleman Research Corporation
Orlando, Florida

INTRODUCTION

Software portability for DoD computer applications has been an elusive goal for many years. Portability implies interoperability, connectivity, and most importantly, vendor independence. Adoption of the Ada programming language was a step toward this goal; however, a common programming language alone cannot ensure portability. Other software components of a computer system must also be standardized. Such efforts will lead to systems that:

- Ensure hardware and software vendor independence,
- Improve user productivity,
- Improve software development efficiency,
- Ensure application portability across various hardware platforms,
- Improve application interoperability,
- Ensure application expandability to larger (faster) platforms,
- Reduce life cycle costs, and
- Improve security.

The term that expresses this goal and is beginning to drive the requirements in specifications and work statements for DoD computer applications is open systems.

An open system is defined by the Institute of Electrical and Electronic Engineers (IEEE) (1991) as "a system that implements sufficient open specifications for interfaces, services, and supporting formats to enable properly engineered applications software:

- to be ported with minimal change across a wide range of systems,
- to interoperate with other applications on local and remote systems, and
- to interact with users in a style that facilitates user portability."

The IEEE (1991) further defines an open specification as "a public specification that is maintained by an open, public consensus process to accommodate new technologies over time and that is consistent with international standards".

BACKGROUND

Various organizations have been in the process of defining a set of open specifications - usually referred to as a profile - in the last few years. These organizations involve industry, standards groups and the US Government. In industry most of the major computer manufacturers, such as DEC, IBM, SUN, have or are developing software that conforms with open standard service elements that have been defined. In fact, these companies participate in the various standards organizations that are defining these profiles. These may be industry consortia, such as the Open Software Foundation; or national/international standards organizations such as the IEEE, American National Standards Institute (ANSI), International Standardization Organization (ISO), or Comite Consultatif International de Telegraphie et Telephonie (CCITT). The resulting comprehensive set of interfaces, services, and supporting formats, including user aspects of interoperability or portability of applications, data or people, as specified by information technology standards and profiles, is called an open system environment.

OPEN SYSTEMS ENVIRONMENT

The US Government through the National Institute of Standards and Technology (NIST), has also established an open system environment (OSE) profile. This OSE profile is called the Application Portability Profile (APP) (NIST,1991). The APP is to be used by Federal agencies in the selection of appropriate OSE specifications for the acquisition of information systems and/or the development of their own OSE profiles. The APP describes an Open

Systems Environment Reference Model as shown in Figure 1.

Application Software
Application Program Interface (API)
Application Platform
External Environment Interface (EEI)
Platform External Environment

Figure 1. OSE Reference Model

The APP, in turn, has led to the issue of an OSE profile for the DoD by the Defense Information Systems Agency (DISA)(1992). This document, as are most of these profiles, is an evolutionary document. As the specifications that make up the profile mature or change, or as new ones are added where none now exist, the profiles will be updated. The current Technical Reference Model for Information Management is shown in Table 1. This set of standards is dated May 15, 1992. Future versions are to be published in the fall of 1992.

The introduction of this document states that it is intended "to provide technical guidance to DoD components for the acquisition, development and support of DoD information systems and associated infrastructure systems"(DISA,1992). DoD components are required to use this profile by selecting appropriate standards from it in order to develop new or update existing information systems.

In the world of training systems, particularly battle simulations used for training, there is often a difference in how standards are applied based on whether the training system is considered under weapon system acquisition procedures or information system procedures. For training systems in general, it can be argued that only weapon system procedures apply. However, in the area of battle simulations, the software is required to comply with information systems policy and standards. Accordingly, the use of the DISA Technical Reference Model for the acquisition of new or the evolution of existing training simulations is required. The benefits of using an OSE are of such potential significance that it is likely that all training systems will be required to use this model in the near future.

IMPLEMENTATION OF OPEN SYSTEM ENVIRONMENTS

General Discussion

Implementation of OSE standards varies depending on a number of factors. These factors include the acquisition phase of the system, status of the standards in the profile, and availability of standards compliant products across a representative set of platforms. Development of a new system, or a major redesign of an existing system, can be done in conformance with any existing standards of the applicable service areas only if compliant products are available. Conversion of an existing system is dependent on a careful approach which considers replacing various components based on a cost effective transition to a non-proprietary environment. Thus, planning for the implementation of OSE for systems will usually consider an incremental or phased approach inserting standards as they mature and become cost effective.

In the DISA Technical Reference Model in Table 1, there are more service area elements which are under development or non-existent than exist as current standards. However, those standards that do exist are in key areas. In many cases the developing standards are nearing completion of the standardization process and will soon be available for implementation by the software developers. Where choices must be made regarding the use of a developing standard from the profile or not, then a strong case can be made for using a mature draft standard. Use of a preliminary draft standard is not advisable due to the high probability of significant changes being made during the approval process, and any products developed from the draft standard would be obsolete and incompatible with the final version. Where a Technical Reference Model void exists, usually no national standard exists in this area. The use of defacto industry standards should be the next most desirable alternative, and the use of proprietary products or actual development of this component would be the least desirable choice.

Table 1 DISA Technical Reference Model for Information Management Open Systems Standards

DISA Service Area	Service Element	Standards: Reference	Standards: Status
Operating System Services			
Kernel Operations	POSIX.1	FIPS 151-1 (IEEE 1003.1-1988)	Now
Commands & Utilities	POSIX.2	IEEE P1003.2, Draft 11 (Proposed FIPS)	Future (Oct 92)
Operating System Security	POSIX.6	IEEE P1003.6, Draft 8	Future (Oct 92)
System Management	GNMP	Proposed FIPS supplemented by MIL-STD-2045-38000	Future (Oct 92)
Programming Services			
Programming Languages & Bindings	Ada, POSIX.5	FIPS 119 (MIL-STD-1815A)	Now
CASE Tools and Environments	PCTE	ECMA PCTE Specification 149	Future
Security			Void
User Interface Services	X Window System		
Data Stream Encoding	Layer 0, Ada Bindings	FIPS 158, STARS	Now
Data Stream Interface	Layer 1, Ada Bindings	FIPS 158, STARS	Now
Subroutine Foundation	Layer 2, Ada Bindings	FIPS 158, STARS	Now
Toolkit Components	Layer 3	IEEE P1201.X, FIPS 158 addition	Future (Oct 93)
Presentation	Layer 4	IEEE P1201.X, FIPS 158 addition	Future (Oct 93)
Dialogue	Layer 5	IEEE P1201.X, FIPS 158 addition	Future (Oct 93)
Human Computer Interface		DoD Human Computer Interface Style Guide	Now
Security			Void
Data Management Services			
Data Dictionary/Directory	IRDS	FIPS 156	Now
Data Management	SQL, Ada Bindings	FIPS 127-1, ANSI X3.168	Now
Distributed Data	RDA	ISO/IEC DP 9759 (draft)	Future (Oct 92)
Security			Void
Data Interchange Services			
Document Interchange	ODA/ODIF/ODL	ISO 8613:1989	Future (Oct 93)
	SGML	FIPS 152	Now
Graphics Data Interchange	CGM	FIPS 128	Now
Product Data Interchange	IGES	ANSI Y14.26-1989	Future (Oct 93)
	STEP	ISO 10303 (draft)	Future (Oct 95)
Electronic Data Interchange		FIPS 161	Now
Security			Void
Graphics Services			
Graphics	GKS, Ada Bindings	FIPS 120-1, ISO 8651-3/ANSI X3.124.3	Now
	PHIGS, Ada Bindings	FIPS 153, ISO 9593-3/ANSI 9593-3	Now
Security			Void
Network Services			
Data Communications	GOSIP Version 2.0	FIPS 146-1	Now
Transparent File Access	POSIX.8	IEEE P1003.8, Draft 4	Future (Oct 92)
Personal/Micro Computer Support			Void
Distributed Computing	NCS/RPC	OSF/1 NCS-RPC	Future
Security		ISO IS 7498-2	Now
		Various	Future

When designing or installing a computer based system, the availability of products becomes a deciding factor. Computer vendor statements of strategic direction, or announcements of products under development cannot be used to build a working system. The availability of compliant products varies across the levels of computers, and ranges from non-existent to plentiful depending on the platform. For example, there are various vendors providing GOSIP Version 1.0 compliant layered products for minicomputers, however there are no such products available for personal computers using MS-DOS.

Vendor implementations can lag the approved standard by years rather than months. Often implementations will only partially comply with the standard or will contain additional features which when used will make the application less portable. Again choices should be made based on the closest fit of vendor products to an approved standard with planned upgrade to a fully compliant system whenever possible. Software development must rely only on those services which are part of the standard and strictly avoid vendor enhancements to the standard.

Operating System Services

These services which provide the interface between the application software and the hardware are defined in the POSIX specifications (IEEE,1991). At the present time, only one of the standards, POSIX.1, has been approved. The others are in various draft stages with approval of some standards projected to require several years. One of the significant differences between the IEEE/OSI standards, such as the POSIX standard, and the DoD Ada language standard is the allowance of non-standard extensions. If the POSIX.1 standard were administered in the same fashion as the Ada language standard, there would be no operating system functions available in the implementation that were not specifically compliant with the POSIX.1 standard. This is not the case. The POSIX.1 standard allows added functionality so long as the core services are provided. This results in the burden of portability being placed on the programmer, who must use only those operating system services which are part of the POSIX.1 standard.

The POSIX.1 standard reflects a combination of the AT&T and University of California at Berkeley versions of the UNIX operating system. However, there are several NIST-certified POSIX operating systems which are proprietary operating systems with the POSIX.1 functions added. This has resulted in anomalies such as the certification of the DEC VMS operating system as one of the first POSIX-compliant systems. Figure 2 contains a list of the NIST-certified POSIX.1 operating systems.

Apple Computer Inc	A/UX ver 2.01	05/24/1991
Control Data	EP/IX ver 1.3.1	05/24/1991
Digital Equipment Company	ULTRIX ver 4.2 VMS ver x.x	04/26/1991 xx/xx/1991
Data General	DG/UX ver 5.4 ver 4.32	07/01/1991 05/24/1991
International Business Machines	AIX ver 3 rel 1	05/24/1991
Santa Cruz Operation	SCO UNIX V/386 ver 3.2	09/10/1991
UNISYS Corporation	CTOS II ver 3 rel 3	09/17/1991
SUN Computer Systems	SUN OS 4.1.2	02/xx/1992

Figure 2 - NIST POSIX.1 Validated Products

Programming Services

The NIST APP lists several IEEE standardized languages as acceptable alternatives for the programming services. These are COBOL, FORTRAN, Pascal, C, and Ada. The DISA Technical Reference Model, however, limits the choice of programming languages to Ada. This Ada only choice poses some problems in the near term, since there are few defined Ada bindings for the other service areas. The integration of the user interface is particularly difficult to implement at this time due to the lack of defined Ada interfaces to the User Interface Services. The standards for these services are defined in terms of the C language bindings and the TCP/UDP-IP network protocols.

User Interface Services

The User Interface Service standard is a seven-layer model similar to the OSI network model. The model is shown in Table 2.

This standard is not yet complete, with only the first three layers having been approved. These are the Data Stream Encoding, Data Stream Interface and Subroutine Foundation layers and are implementations of the Massachusetts Institute of Technology (MIT) X-Window system as defined in FIPS Publication 158. The fourth layer defines the toolkit standards which support the window management Application Programming Interface (API). The fifth and sixth layers define the look and feel of the Graphical User Interface (GUI). There is no consensus on the standard for these layers, but two competing implementations, the AT&T Open Look and the Open Software Foundation's MOTIF toolkits, are considered to be good examples of what the standard should encompass. DISA has published a draft style guide for the development of user interfaces which unfortunately only describes the two toolkits and does not mandate or recommend a preferred choice.

Table 2 - User Interface System Reference Model

Layer	Name	System Components
6	Application	Application Program
5	Dialogue	User Interface Definition Language (UIDL), User Interface Management System (UIMS)
4	Presentation	UIDL, UIMS
3	Toolkit	Toolkit components (widgets)
2	Subroutine Foundation	Functions for building widgets
1	Data Stream Interface	Function access to Data Stream
0	Data Stream Encoding	Definition of format and sequencing of byte streams

Data Management Services

The three components to this standard are the data dictionary/directory component for accessing and modifying information about data, 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. The first two components are already defined by the Information Resource Dictionary System (IRDS) and the Structured Query Language (SQL), contained in FIPS Publications 156 and 127-1. Remote Database Access (RDA) is only in draft status, and the existing RDA standard only specifies the service and protocol between a single client and a single server. The specification does not currently address distributed database access. There are a variety of certified SQL processors available for most hardware platforms, and Ada bindings are available.

Data Interchange Services

These services are divided into four segments - document, graphics data, product data, and electronic data interchange. The Open Document Architecture/Open Document Interchange Format/Open Document Language (ODA/ODIF/ODL) standards have been approved by two standards bodies, ISO and CCITT. A FIPS publication is planned. However, there are no certified implementations of this service. The Standard Generalized Markup Language (SGML) is defined in FIPS publication 152. A test suite is under development. Thus, there are no certified SGML products. The Computer Graphics Metafile (CGM) is defined in FIPS publication 128. Conformance test suites have been available for some time and conforming applications are readily available. The Product Data Interchange standard uses the Initial Graphic Exchange Specification (IGES) which was defined by the ANSI Y14.26 and has been used by the CALS Application Profile. A FIPS publication defining the standard is planned. Numerous implementations of the IGES are available. The Product Data Interchange standard uses the Standard for the Exchange of Product Model Data (STEP). STEP is still in draft status with many of the component definitions undefined. The Electronic Data Interchange standard is defined in FIPS publication 161.

Graphics Services

The Graphics Services standard is comprised of two components, the Graphical Kernel System (GKS) and the Programmer's Hierarchial Interactive Graphics System (PHIGS). The GKS standard is defined in FIPS publication 120-1. The standard also defines bindings for Ada, FORTRAN, and Pascal. Test suites and conforming implementations are available. The PHIGS standard is defined in FIPS publication 153. Bindings for FORTRAN and Ada have been defined, and conforming implementations are available. In many ways these are competing standards. Since a PHIGS extension to X-windows (PEX) will be included in FIPS 158-1, this may become the recommended standard.

Network Services

The Network Service standard is based on the OSI Reference Model, with the Government Open System Interconnection Profile (GOSIP) comprising a subset of the entire OSI model. GOSIP is essentially a complete end-to-end communications capability based on the OSI transport class 4 and connectionless network protocol (CLNP). GOSIP Version 1.0 has been a mandatory requirement since August 1990, and GOSIP Version 2.0 will be required after October 1992.

GOSIP Version 1.0 compliant, NIST-certified products are available on some minicomputer platforms. However, on the Apple Macintosh and Intel 80x86 platforms (IBM-PC compatible), they are not available above the physical layer. At the physical layer, at least one vendor has had their ethernet adapter cards certified as being IEEE 802.3 compliant. There are products which claim to provide some of the GOSIP services for the Intel platform; however, the degree to which they support the standard is questionable, and the services which are provided do not match the functionality required for most microcomputer-based networks. This functionality generally includes the ability to mount file systems from other computers as local disk drives or file systems, and the ability to redirect print streams from the workstation to a remote printer. An additional problem with the existing COTS implementations of GOSIP network protocols is that they require a substantial amount of the conventional (640 KB) memory on the Intel 80x86 MS-DOS microcomputers. This memory requirement precludes the simultaneous use of the network and the Commercial Off The Shelf (COTS) application software such as WordPerfect or Excel. There are currently no GOSIP Version 2.0 compliant products available.

U.S. Army War College Simulation and Modeling Interface System (USMIS)

The USMIS project encompasses building an environment to provide staff, faculty, and students with easy access to a variety of wargame simulations and models, and the ability to examine and analyze different courses of action. Some of the features required by the USMIS were the linking of different computer architectures, operating systems, and graphical user interfaces into one homogenous environment, and the extension of the environment to other wargaming centers including the Air War College, Naval War College, and the National Defense University.

Specific software development focused on building an intuitive, easy-to-learn graphical user interface to the tools and models used at the War College's Center for Strategic Wargaming (CSW). Complicating the development was the requirement to integrate the existing computer platforms associated with the different models and other application software into the new graphical environment. The existing equipment included DEC VAX 8800, 3000, and microVAX series computers, SUN SPARCstations, Apple

Macintoshes and Intel 80x86 microcomputers. The new system had to be able to operate with the GOSIP protocols and connect to the Defense Data Network's MILNET and DSNET1 subnetworks.

The development effort started with a market survey to determine the availability of products which complied with the NIST APP (the DISA Technical Reference Model had not yet been published) standards for appropriate services. These products needed to be available for each of the computers which would be used in the new system. Availability for the platforms is shown in Tables 3 through 6.

Table 3 - DEC VAX 8810, 3100, MicroVAX II

Service	Available Products	Comments
Operating System	VMS/ULTRIX/UNIX	VMS now POSIX compliant
User Interface	X-Windows; Open Look & MOTIF	
Programming	Ada	
Data Management	SQL-compliant	
Data Interchange	Various	Protocol tied to application programs
Graphics	Various	Protocol tied to application programs
Network	OSI & TCP/IP over Ethernet	

Table 4 - SUN SPARCstation 1+, 2, IPC, IPX

Service	Available Products	Comments
Operating System	SUN-OS Version 4.1.2	POSIX compliant
User Interface	X-Windows, Open Look & MOTIF	
Programming	Ada	
Data Management	SQL	
Data Interchange	Various	Protocol tied to applications programs
Graphics	Various	Protocol tied to applications programs
Network	OSI & TCP/IP over Ethernet	

Table 5 - Apple Macintosh

Service	Available Products	Comments
Operating System	System 7 & A/UX	A/UX is POSIX compliant
User Interface	Macintosh/X-Windows	
Programming	Ada	
Data Management	SQL	
Data Interchange	Various	Protocol tied to applications programs
Graphics	Various	Protocol tied to applications programs
Network	TCP/IP over Ethernet	

Table 6 - Intel 80x86

Service	Available Products	Comments
Operating System	SCO UNIX MS-DOS	Not compatible with AWC applications (Harvard Graphics, Excel, WordPerfect 5.1 Not POSIX compliant
User Interface	MS-Windows Presentation Manager X-Windows	Not DISA compliant Not DISA compliant Not available as a PC client application
Programming	Ada	Available
Data Management	SQL	Available
Data Interchange	Various	Protocol tied to applications programs
Graphics	Various	Protocol tied to applications programs
Network	GOSIP Version 1.0 GOSIP Version 2.0 TCP/IP over Ethernet	Not Available for MS-DOS Not Available Available

One of the driving functional requirements is the ability to interface the wargaming models with easy-to-use office automation tools, allowing the instructors and analysts to prepare After Action Review (AAR) materials in a real-time basis. Part of this requirement is to be able to display troop locations and other military map symbols on a map background, and to permit students to then take these map images and modify them for use in the next phase of the campaign.

The models used at the war college run on Intel 80x86 and Apple Macintosh microcomputers, SUN SPARCstations, and DEC VAX minicomputers. Analysis of the different systems, the standard office automation software tools currently used at the college, and the specific USMIS system requirements revealed the following:

Ethernet (802.3 compliant) would provide the necessary bandwidth for data transfers, was readily available for all of the computer systems, and provided the most cost effective network medium. The desired network protocol is GOSIP TP4; however, an exhaustive market study discovered no GOSIP-compatible products which would function with the planned microcomputers and their COTS applications. Additionally, there are no NIST-certified GOSIP network products for any microcomputers. GOSIP TP4 network protocols are available for the SUN workstations; and routing software is available to allow the SUN workstations to communicate using Internet Protocol (IP) with the microcomputers, and using TP4 to communicate with external agencies who will be using GOSIP in the future.

The operating systems would be driven by the system capabilities for the microcomputers. The POSIX implementations by SUN, AT&T, UNIX Labs, and SCO for the Intel 80x86 microcomputers require a substantial overhead cost in terms of disk storage, main memory (RAM) and processing power, and do not support the COTS software applications necessary for USMIS functionality. MS-DOS and System 7 were required for the Intel and Apple microcomputers in order to use the existing AWC application software. The SUN workstations will be upgraded to the POSIX compliant version of the SUN operating system. The VAX computer operating systems will be upgraded on the normal software maintenance schedule.

The user interface is required to be the same, or at least similar, across all of the platforms. The preferred method is to use the X-window system as the base protocol, and to build X-client applications which use the X-server protocol to display on the user's workstation. The SUN workstation operating system software includes a window manager and support for the X-client/server architecture. The USMIS uses several X based application, including one which provides the ability to access the Defense Mapping Agency terrain map products. By using X-server software on the SUN, Intel, and workstations, this and other applications are available across the network. The SUN Open Look toolkit is used with the SUN Open Windows window manager to provide the layer 2 through 5 functionality of the User Interface System Reference Model. On the microcomputers, clients and window managers are somewhat limited. For the Intel microcomputers, window managers which work with the MS-DOS based COTS applications are limited to IBM's Presentation Manager and Microsoft's MS-Windows. Of these, only MS-Windows has the native mode capability to work with the COTS applications and supports COTS X-server applications. To achieve the similarity across the various platforms, menu systems were built on each platform using the development toolkits appropriate to the environment for each workstation. The Graphical User Environment provides icons or buttons which activate applications either on that workstation, or cause the activation of X-client processes on other workstations, with the results displayed using the native mode window system for the displaying workstation.

Programming for the USMIS is limited to building menu systems and providing services not available from COTS software. In keeping to this guidance, programs are written to translate incompatible files which are the output of certain COTS applications or wargame models. These programs use the Ada language. The development of the menu systems and the graphical widgets and dialogue boxes are accomplished by using the toolkits associated with the workstation platform. The SUN systems use the SUN Development Graphical User Interface Development Environment (GUIDE). The MS-Windows microcomputers use the Windows toolkits.

Data management through the use of a database management system is a future issue for USMIS, and is not being exploited during the initial phase.

Data interchange is a crucial element of the USMIS system. Graphic images are generated by the terrain map display system and several other applications running on the SUN workstations. These graphic images must be viewed and manipulated using the workstations available to students and the faculty, which are the Intel and Macintosh computers. The standardization of color graphics formats in the COTS software community is almost non-existent. Although there are several file formats for the storage of bitmapped and vector images, each format has several variations or "flavors", which make an image created on one system (Adobe Illustrator) unusable on another (PageMaker). The recent addition of color to these formats has increased these incompatibilities. The most stable of the formats available for import and export at the present time is the Tagged Image File (TIF) format. While most of the COTS graphics applications offer a variety of file formats that they can import and export, images which are exported and then reimported into the same application suffer translation problems, sometimes so severely that the image is not viewable. The Encapsulated PostScript (EPS) format is relatively well standardized (since all applications need to print). However, few programs have the ability to import color EPS files. The preferred format, CGM, is not widely available, and the current implementations are not fully compliant with the standard. The USMIS workstations use the TIFF and Postscript formats to transfer image files between platforms, and use the windows (X-windows on the SUN and MS-Windows on the Pcs) clipboard for transfer of images between active applications.

USMIS Benefits

The use of an open system architecture provided the framework upon which to build the system. The selection of non-proprietary solutions for those areas where mature standards did not exist provides a layered solution into which future standards can be placed. The use of the TCP/IP network protocol provides for interconnection of numerous existing devices, and allows for an easy transition to a fully compliant GOSIP network when these implementations become available. Use of the Open Look toolkit greatly simplified the building of the application interfaces, and makes for easy maintenance of the system, as well as providing portability and vendor independence.

CONCLUSIONS

The U.S. Government has defined an OSE by a selected suite of specifications known as the APP. The APP provides guidance for Federal agencies to make informed choices regarding the selection and use of OSE specifications/standards. The DoD and the services support the concept of an OSE and in turn have adopted their on profile based on the OSE called the DISA Technical Reference Model. This profile will be required for use in the development/evolution of training simulations and eventually will be required as part of the development of new training systems in general. Implementation of these standards will be different dependent on whether a new system is being developed or an existing system is being upgraded. The partial status of many of the standards in the profile and the lack of availability of standards compliant products across a representative set of platforms will make implementation difficult. Such difficulties are demonstrated by the trade-offs made in the USMIS project.

Institute of Electrical and Electronic Engineers, Technical Committee on Operating Systems and Application Environments. (1991) Draft Guide to the POSIX Open Systems Environment (P1003.0/D14). New York: Author.

National Institute of Standards and Technology, Computer Systems Laboratory. (1991). Application Portability Profile (APP) The U.S. Government's Open System Environment Profile OSE/1 Version 1.0 (NIST Special Publication 500-187). Washington, DC: US Government Printing Office.

Defense Information Systems Agency, Center for Information Management. (1992). Technical Reference Model for Information Management (Version 1.2).