Using Standards to Facilitate Access and Reuse of Museum Information

Judi Moline[1]
National Institute of Standards and Technology
Technology Building Room B266
Gaithersburg, MD 20899
email: jmoline@nist.gov
Voice: (301) 975-3351
Fax: (301) 590-0932

Abstract

Standards for dealing with electronic records must be selected based on a careful analysis of user requirements. These requirements must take into account the long range goals of the larger community. For any particular museum this larger community could include archives and libraries, as well as other museums and the patrons of the museums. Consistent use of standards among members of the museum community would facilitate the interchange of information among systems. Further, standards allow the reuse of information. This reuse may be for the same or different purposes.

This paper presents a framework for determining the needed standards for museum records management. Although options are discussed based on the Open System Environment, these are not a predetermined set of standards. The goal of this paper is to help a particular user community select appropriate standards.

1 *Affiliated with the Office Systems Engineering Group, Systems and Software Technology Division, Computer Systems Laboratory and Ph.D. Candidate in Information Transfer from Syracuse University.*

1. Requirements of the Museum Community

A thorough understanding of user requirements is necessary before an efficient information system[2] can be designed. Design of the system includes the selection of hardware and software that meets the needs of the users for in-house information, as well as for data interchange requirements with other systems. This data interchange should be "application, implementation, system and vendor independent'[Com91, page 2].

Analysis of the tasks performed by museum professionals would provide a list of the resources and tools needed to perform each task. From this, a requirements model could be formulated.

1.1 Developing a Requirements Model

The objective of developing a model showing all the requirements of an information system for museums is to present a broad picture showing the information flow, how the information is reused, and the links among the components of the system. Some of the museum tasks are concerned with accession, documentation, access and storage, deaccessioning, lending, and preservation. These tasks need to be expressed as specific steps including input and output requirements.

For example, to accession an object, the object must be available along with a database and a set of measurement devices. Specifically, when an object is accessioned, the object is thoroughly described, usually by filling in fields in a database. These fields include descriptive data, as well as management data. Concurrently with the description the object is analyzed. The goal is a complete record of the object, placing it in its historical setting. Scientific analysis may be made of the material. Documentation includes photographs and drawings and any information available as to when and where the object was found and a record of where it has been since its discovery. Based on this analysis, the object is fit into the museum collection, both physically and conceptually. This "conceptual fitting" could cause major changes in historical thinking if the object presents new and/or unexpected evidence. Other information base records and collection catalogues might require revision and it would be important to publish the new material evidence to open the area for further research.

In developing the requirements model, the process for each task must be annotated in great detail with all the options cited. Further, terms must be carefully defined to assure that their use is completely understood. From this, data flow diagrams and other representations are generated to give the full picture of what information is needed by each component of the system and how to link the components so that the information is entered

2 *Information is used here as "information-as-thing"[Buc91].*

only once. With careful planning, the data entered on a template, such as that for the object record, triggers a series of links in the information bases of documents, images, sound tracks, etc. A map of these links can be viewed and modified.

Once the requirements model is determined, a model of the computerized information system must be generated. Some requirements of this system are that all information about each object held by the museum be included, that this information be in easily searched databases, that these databases can be accessed from terminals all over the world, that a variety of surrogates be allowed (text, photo/slide, drawing, video, etc.), and that each museum's system be tied in with those of other museums, universities, libraries, and collectors. Further, books, encyclopaedias, journals, catalogues, etc. should be made available via electronic libraries. At present such "libraries" exist on CD-ROM.

In addition to the more obvious tasks mentioned above, there are other issues which could be facilitated with computer assistance. Some of the administrative concerns that fit in this category are statistics, preservation schedules, and status. For example, the following statistical concerns could be programmed into the system: relative value of each object from an historical perspective, the frequency of "use" of any object whether locally or for loan purposes, and the relative cost of upkeep for any group or type of item. Preservation schedules could be put online to assure their completion on time (e.g., time line for maintenance and requirements for maintenance). Status issues include priorities for weeding or exchange and policies for loan, exhibition, and research. Besides the traditional database access, the researcher could be provided with a video sequence showing the object from various angles, an audio sequence made by the archaeologist who found the object, artificial intelligence in the form of expert systems, etc. In order to facilitate searching and to allow for browsing, a hypertext interface should be available.

2. The Open System Environment

The Open System Environment (OSE) is a conceptual framework[3] that provides a context for user requirements and standards specification. It is shown in Figure 1 in its simplest form [Fis91, page 4]. Within the Reference Model, standards address interfaces between entities and between services and supporting formats across those interfaces. The Reference Model is not a layered model; the application platform provides services to users across both platform interfaces [IEE91, page 20].

3 *The Open System Environment concept is being developed by the Systems and Software Technology Division of the Computer Systems Laboratory, NIST.*

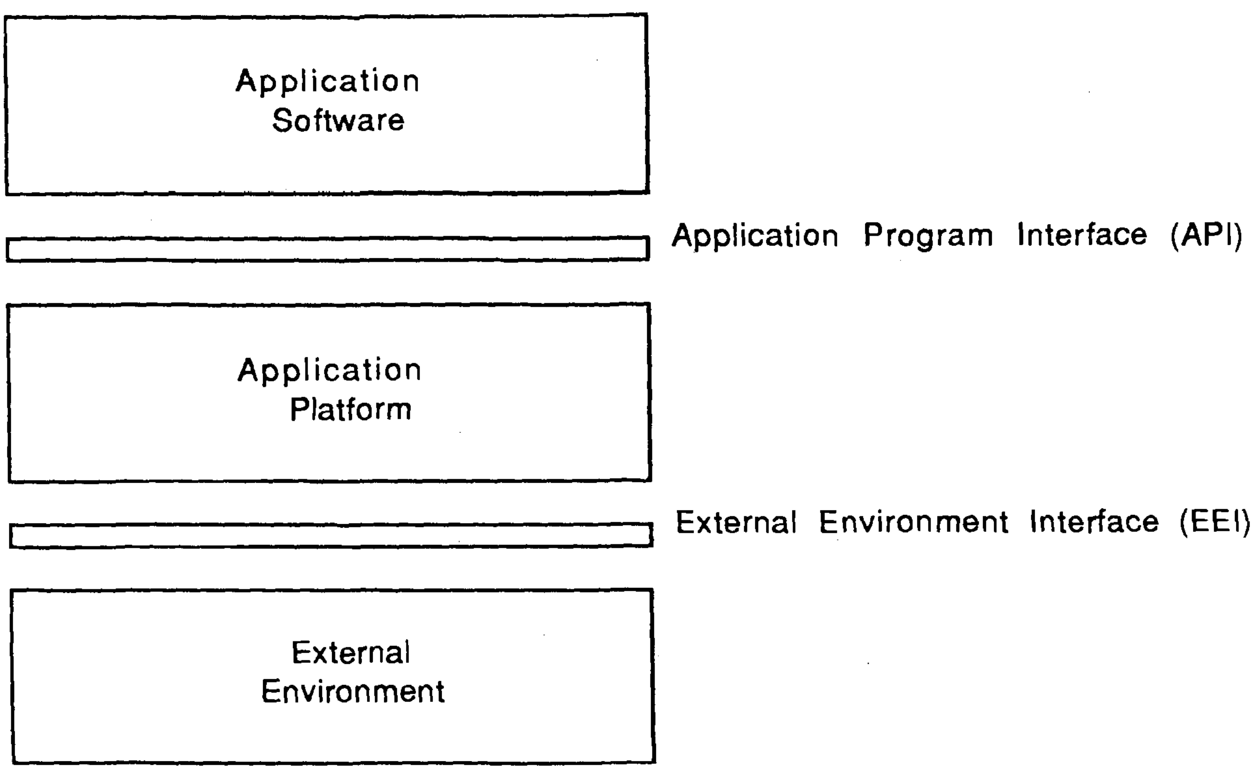
2.1 Application Program Interface

The Application Program Interface (API) contains a number of standards-based interfaces [IEE91, page 24]. The API allows for portability by providing a standard interface between the services of the application platform and the needs of the application software. Portability is the ability to use application software on heterogeneous hardware/software platforms.

2.2 External Environment Interface

The External Environment Interface (EEI) is the interface between the application platform and the external environment across which information is exchanged. The particular services comprise the human/computer interface, the information interface, and the communications interface [IEE91, page 22]. The EEI assures interoperability between devices and the hardware platform. Interoperability of application software operating on heterogeneous hardware/software platforms is necessary for performing some user functions and for data sharing. Specifically, the EEI supports information transfer between the application platform and the external environment.

THE OPEN SYSTEM ENVIRONMENT REFERENCE MODEL



3. An Open System Environment for Museums

A museum model based on the generic Open System Environment follows[4]. The examples in the discussion are given to facilitate the development of a concrete model for the reader. They are not intended to represent the design of an actual Open System Environment implementation.

3.1 Application Software

Once the user requirements or needs are specified, the application software can be selected. Sample categories of this software are included in Figure 2. Each application should be capable of communicating and synchronizing with other applications when necessary using communication mechanisms [IEE91, page 21]. It is important that the software selected uses standard interfaces.

Available standards for the services provided by the Application Platform include the following: POSIX for operating system services, X-Windows or higher level Graphical User Interfaces (GUIs) for user interface services, SQL for data management services, SGML, ODA, and CGM for data interchange services, PCTE for programming services, and GKS and PHIGS for graphics services. GOSIP is the Federal Government's profile of standard data communication protocols for network services. (See Appendix A for acronyms.)

The input and output devices for the system, whether CD-ROM, video disc, TV, etc., are determined by the user's choice of software. Figure 2 shows sample devices in the External Environment component of the OSE. Further, the human end-user is an important external entity with which the application platform exchanges information.

3.2 Application Platform

The Application Platform is well-defined in the Open System Environment Reference Model [Fis91, page 7] [MHW90, page 26]. The components are shown in Figure 2. Features needed for each particular application or system would be selected and implemented; not every application platform would provide all features [IEE91, page 22].

4. Using Standards

Standards may be used for many things: nomenclature, data representation, object description, graphics, supplemental documents, etc. The need for consistent use of stand-

4 *This concept is developed in [Mol91].*

ards is critical for data interchange among systems and even among components of a single system, and reuse of information.

Standards provide "consistency over time of a single application, uniformity throughout an institution, the ability to upgrade systems conveniently, the potential of communicating data among systems, and the possibility of making data available to a broad audience"[SP89, page1].

"The Standard Generalized Markup Language (SGML) and the Office Document Architecture/Office Document Interchange Format (ODA/ODIF) are the two most prominent standards for the markup and interchange of electronic documents" [NW91, page 1]. Although not equivalent standards, each is concerned with electronic document interchange. Because each of these standards was developed by a group with specific needs in mind, it is unlikely that they will merge. Therefore, the user must choose the standard more appropriate to his/her needs. Since SGML and ODA/ODIF are likely to be widely used, there will be translators to convert a document from one to the other[5]. However, since translation is an additional step, the user should choose the standard which offers the greatest benefits for the majority of the needs.

5. The Potential for Reuse of the Content Components

An area of concern to the museum community should be that of reusing electronic files. This reuse may be for the same or different purposes. One way to facilitate such goals is by using tags such as might be used in SGML. SGML is a general-purpose language for describing the structure of data. It is a metalanguage although the language created with it is commonly called a document type definition (DTD). A DTD is an ASCII text file describing what elements and in what order and frequency they may occur in a class of documents. A DTD may be thought of as a set of tags and how those tags may be used plus a grammar that constrains interrelationships. SGML documents are interpreted by a parser which first reads and interprets the related DTD [AW90]. Currently there are many products available which aid in SGML tagging. Further, there are well-developed DTDs available, for example that of the Association of American Publishers (AAP) [AAP88] and that of the Text Encoding Initiative (TEI) [SMB90] which is still under revision.

SGML may be used as generic markup with the DTD written to describe data. This markup identifies elements that have specific formats. Secondly, SGML may be structured markup. In this case the markup parallels the logical organization of a document; for example elements might be labeled "title" or "section." Third, SGML may be markup that facilitates database searching. In this case, elements are identified by name such as "vase

5 *Software has been written at NIST that demonstrates interoperability of ODA and SGML [NW91, page19].*

type I." "This type of markup is the most difficult to create, but where applicable (especially in tables, catalogs and databases) it offers the most potential for electronic processing, because the user can query the database based on its content" [AW90]. More realistically, SGML might be used for small tables embedded in a document or for interchange of information. Where data is in a relational database, SQL would be the appropriate query language.

6. Conclusion

With the rapid development of the computer industry and related standards, users must make decisions which will benefit their long term goals. Users to think about open systems today even though some standards are not yet fully mature.

> An open systems environment is something that must be migrated to; the more pieces already in place, the easier it will get. The trick for users is to choose carefully today so that a system can be flexible enough to change with the wanderings of open systems, and so that migration hassles and costs can be handled over time [Pat91, page 16].

The most important step for users is to make a careful analysis of their needs. Once the analysis is made, selection of software with standard interfaces is undertaken. The platform or hardware necessary to run the various software applications should be the next consideration. Again, care should be taken to select input/output devices which use standard interfaces.

A. List of Acronyms

The following acronyms were used in this paper and are provided here for reference.

API - Application Program Interface
CD-ROM - Compact Disc—Read-Only Memory
CGM - Computer Graphics Metafile
EEI - External Environment Interface
GKS - Graphical Kernel System
GOSIP - Government Open Systems Interconnection Profile
ODA - Office Document Architecture
OSE - Open System Environment
PCTE - Portable Common Tool Environment
PHIGS - Programmer's Hierarchical Interactive Graphics System
POSIX - Portable Operating System Interface for Computer Environments
RDA - Remote Database Access
SCSI - Small Computer System Interface
SGML - Standard Generalized Markup Language
SQL - Structured Query Language

References

[AAP88] AAP. *American National Standard for Electronic Manuscript Preparation and Markup: An SGML Application Conforming to International Standard ISO 8879 - Standard Generalized Markup Language, ANSI/NISO Z39.59-1988.* EPSIG and Association of American Publishers, 1988.

[AW90] George Alexander and Mark Walter. "A fresh look at SGML: The conventional wisdom changes." *The Seybold Report on Publishing Systems*, 20(7):1-14, Dec. 24, 1990.

[Buc91] Michael K. Buckland. "Information as thing." *JASIS*, 42(5):351-360, June 1991.

[Com91] CIMI Committee. "First CIMI committee meeting held." *CIMI News*, 1(1):1-4, January 1991.

[Fis91] Gary Fisher. "Application portability profile: The U.S. Government's Open System Environment Profile." NISTSP 500-187, National Institute of Standards and Technology, April 1991.

[IEE91] IEEE. Draft guide to the POSIX Open Systems Environment. Unapproved Draft P1003.0 / D11, Technical Committee on Operating Systems and Application Environments of the IEEE Computer Society, March 1991.

[MHW90] Judi Moline, Allen L. Hankinson, and Lawrence A. Welsch. Multimedia courseware in an Open Systems Environment: A Federal strategy. NISTIR 4484, National Institute of Standards and Technology, December 1990.

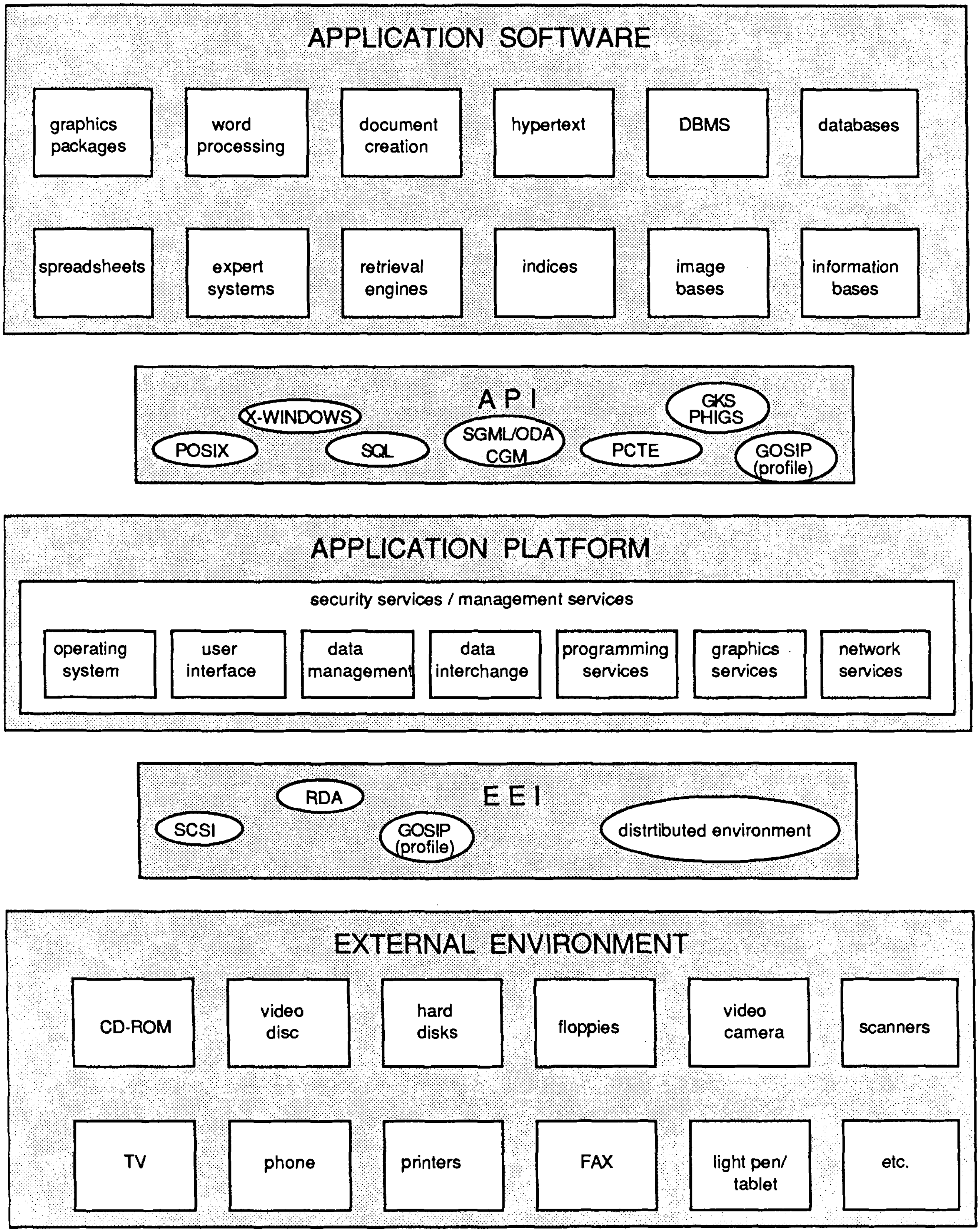
[Mol91] Judi Moline. "Designing multimedia systems for museum objects and their documentation." *Microcomputers for Information Management*, 8(2):69-86, June 1991. Paper presented at ASIS, Santa Clara, April 27, 1991.

[NW91] Charles K. Nicholas and Lawrence A. Welsch. On the interchangeability of SGML and ODA/ODIF. Draft of NISTIR, June, 1991.

[Pat91] Kim Patch. "Users assert control." *Open Systems Focus*, 1(1):8-16, Summer 1991.

[SMB90] C.M. Sperberg-McQueen and Lou Burnard, editors. *Guidelines for the Encoding and Interchange of Machine-Readable Texts*, July 1990. Draft: Version 1.0.

[SP89] Deirdre C. Stam and Ruth Palmquist. *SUART: A Marc-Based Information Structure and Data Dictionary for the Syracuse University Art Collection*. Syracuse, NY: Museum Computer Network, 1989.


APPLICATION SOFTWARE
graphics packages
word processing
document creation
hypertext
DBMS
databases
spreadsheets
expert systems
retrieval engines
indices
image bases
information bases
A P I
X-WINDOWS
GKS PHIGS
POSIX
SQL
SGML/ODA CGM
PCTE
GOSIP (profile)
APPLICATION PLATFORM
security services / management services
operating system
user interface
data management
data interchange
programming services
graphics services
network services
E E I
RDA
SCSI
GOSIP (profile)
distrtibuted environment
EXTERNAL ENVIRONMENT
CD-ROM
video disc
hard disks
floppies
video camera
scanners
TV
phone
printers
FAX
light pen/ tablet
etc.