# Quality of Service in Multimedia Digital Libraries*

**Elisa Bertino*** **Ahmed K. Elmagarmid†** **Mohand-Saïd Hacid†**

*Dipartimento di Scienze dell'Informazione
University of Milano
Via Comelico, 39/41 20135 Milano - Italy
*bertino@dsi.unimi.it*

†Department of Computer Sciences
Purdue University
West Lafayette, IN 47907 - USA
{*ake,mshacid*}*@cs.purdue.edu*

## Abstract

*There is currently considerable interest in developing multimedia digital libraries. However, it has become clear that existing architectures for management systems do not support the particular requirements of continuous media types. This is particularly the case in the important area of quality of service support. In this correspondence, we discuss quality of service issues within digital libraries and present a reference architecture able to support some quality aspects.*

**Keywords:** Quality of Service, Data Quality, Digital Libraries, Multimedia Data, Interoperability.

## 1 Introduction

Building digital libraries[1] is one of the most important Grand Challenge problems faced by information professionals. The size, the number, and the complexity of new advanced digital libraries is increasingly worldwide. New visual information in the form of images, graphics, animations, and videos is being published on the web at an incredible rate. However, cataloging and integrating this visual data within advanced applications is beyond the capabilities of current text-based systems. No tools or methodologies are currently available for systematically designing and developing advanced applications that integrate multimedia objects. This absence is particularly notable given the increasing number of applications that need to embed visual information.

Serious exploitation of this mass of data is nearly impossible without intelligent software systems. Next generation of systems will rely on the ability to store, access and reason about large volumes of multimedia objects (with a natural spatial distribution). Building a global multimedia repository will solve many troublesome current problems. One such problem is the *fundamental human need to obtain useful information as required.* A model of the process involved is desired. This enterprise is especially difficult; humans often have anomalous states of knowledge, which we sometimes interpret as needs or problems, but these are hard to characterize, depending in part on complex situations of uncertainty, ambiguity, partial knowledge, and missing information. All of the artificial intelligence problems of the knowledge representation, natural language understanding, planning, learning, etc. are in evidence here. This problem can be redefined in terms of computers carrying out the activities of expert librarians. A second problem is that existing information systems, and then digital libraries, have been built in isolation, using incompatible methods. A third problem is that of *scaling up* services to *handle large quantities of information*, *large information objects*, and *large numbers of users*. Finally, the information overload problem has a generalization in the problem of developing suitable interfaces to digital libraries. How can powerful multimedia systems be used with large amounts of incoming information? How can workstations coordinate the presentation of multimedia information to adapt to users' preferences, capabilities, challenges, styles, and tasks?

### 1.1 Services and Digital Libraries

It is recognized that digital libraries represent a set of significant social problems that require human and technological resources to solve. Digital libraries encompass two complementary ideas [4]:

- Digital libraries are a set of electronic resources and associated technical capabilities for creating, searching, and using information. In this sense they are an extension and enhancement of information storage and retrieval systems that manipulate digital data in any medium (text, images, sound, video) and exist in distributed networks. The content of digital libraries includes data, metadata that describe various aspects of the data, and metadata that consist of links or relationships to other data or metadata, whether internal or external to the digital library.
- Digital libraries are constructed, collected and organized, by a community of users, and their

*This work is supported by the National Science Foundation under grant 9972883-EIA.
[1]Or multimedia repositories

functional capabilities support the information needs and uses of that community. They are a component of communities in which individuals and groups interact with each other, using data, information, and knowledge resources and systems. In this sense they are an extension, enhancement, and integration of a variety of information institutions as physical places where resources are selected, collected, organized, preserved, and accessed in support of a user community. These information institutions include, among others, libraries, museums, archives, and schools, but digital libraries also extend and serve other community settings, including classrooms, offices, laboratories, homes, and public spaces.

The first idea emphasizes the fact that digital libraries are computer-based systems constructed for people to use and that they are extensions of information storage and retrieval systems. The second emphasizes the belief that digital libraries should be constructed in a way that accommodates the actual tasks and activities that people engage in when they create, seek, and use information resources; in this sense they are an extension of physical environments. Both assert that digital libraries are sets of information resources collected and organized on behalf of a community.

## 1.2 Digital Library and Decision Support Systems

During the past 25 years, great progress has been made in research and commercial applications of Decision Support Systems (DSS). Conceived originally as the application of computing technology to support decision making, DSS research focused on the implementation of tools from operations research. By allowing end users to state business problems in a higher level language and use software to translate requests, build suitable models, access required databases, integrate and execute models, and finally provide answers to the user, decisions can be made in a more effective manner.

While there are still many challenges and opportunities within this paradigm, significant changes have taken place in the environment surrounding DSS so that radically *new approaches* are *required*. These changes are based on end-user demand and technological changes. Instead of focusing on highly structured aspects of company operations that can be modeled using operations research tools, there is a need to analyze organization wide issues that incorporate a wide range of knowledge representations and data types.

The growth of distributed computing and the emergence of research on the "digital library" provide new insights to DSS research [1, 3, 5, 2]. Since mid-80's, electronic documents have become more powerful and more widely available each year. The term electronic document has come to encompass a wide variety of knowledge forms including text (reference volumes, books, journals, newspapers, etc.), illustrations, tables, mathematical equations, scientific data, scanned images, video, voice, hypertext links, and animation. In the mean time, digital networks and the number of users are growing exponentially. The massive information sources available on the network (i.e., the Internet) have formed the basic ingredient of a digital library. The digital library can be thought of as a repository of executable documents (e.g., equations, spreadsheets) that has different part scattered on different platforms across the network. The problem solving process can be tied to *identifying* parts of documents that can be executed. Each part is a model fragment that can be composed with others to form an executable compositional model. When users retrieve information from the library as to a particular query, an underlying system should be able to isolate the relevant model fragments, execute them, and return a sufficient answer to the user.

## 1.3 Service Quality Management

The concept of **Quality of Service** (QoS) was originally introduced in computer communications to mainly characterize data transmission performance [6]. In (distributed) digital libraries, considering Quality of Service only at the communication level is no more sufficient. Quality of Service is directly perceptible by the user who should have the possibility to express his requirements. Quality of Service management then becomes an essential end-to-end functionality that the entire (distributed) digital library should provide.

Quality of Service management on an End-to-End basis means that: (1) the user expresses her/his wishes on the quality she/he wants and the cost she/he is willing to pay, and (2) the system transparently operates in order to deliver the requested level of quality. This approach requires to consider Quality of Service not only at a specific layer, such as the network or the operating system layers, but at all the layers of the distributed digital library.

At least, three different activities are identified in QoS management: specification, mapping, negotiation:

- QoS specification deals with the definition of the requested QoS level. This definition is made according to different dimensions such as time, cost or quality. It should be offered through an adequate interface allowing the user to express how he perceives the quality of the provided service. As an example, a user could express her/his QoS level as the following statement: "I am interested with full-color videos that could be delivered for less than 10$ and in a 40 s delay". QoS specification can be provided by the user or by the application programmers for a set of applications. In both cases a specification language is required [7].

- QoS mapping occurs at the different layers of the digital library. Users perceived quality has to be mapped onto QoS parameters that will be supported by the different layers. As an example, the previous statement must be translated into parameters values for resolution, frame rate and throughput.

- QoS negotiation corresponds to setting up a contract between the different layers and components of the distributed digital library in order to satisfy the user's requirements. This function is in charge of finding a system configuration that should support the requested QoS and can be considered as similar to resource allocation in distributed systems. QoS negotiation leads to a commitment from the overall components concerning the quality level that will be offered.

The complexity of QoS processing, requires to systematically consider QoS in the different layers of distributed digital library.

### 1.4 Linking Digital Library and Data Quality

A possible link of the main components of a distributed digital library (DDL) reference architecture to a formal model of data quality [8], is shown in figure 1.
A closer examination of the quality factor hierarchy reveals several relationships between quality parameters and design/operational aspects of digital libraries.

- The simple digital library concept itself alleviates the problem of **accessibility**, by saving its users the effort of searching in large, poorly structured information space. However, the issue of delivering the information efficiently, is an important open problem in the light of its differences from traditional query processing. In a distributed digital library environment, there is an increased need for fast query processing.

- It remains difficult for DDL customers to **interpret** the data because the semantics of data description languages for digital library schemata is weak, does not take into account domain-specific aspects, and is usually not formally defined and therefore hardly computer-supported. It is necessary to ensure interpretability by investigating the syntax, semantics, and reasoning efficiency for *rich schema languages* which (a) give more structure to schemas, and (b) allow the integration of user defined needs (functions).

- The **Usefulness** of data is hampered because it is hard to adapt the contents of the distributed digital libraries to changing customer needs, and to offer a range of different policies for ensuring adequate timeliness of data at acceptable costs. What is required is the development of techniques and metrics for characterizing the usefulness of a given set of multimedia data in response to defined user needs.

- The **believability** of data is hampered because the DL customer often does not know the credibility of the source and the accuracy of the data. Moreover, schema languages (and then metadata) are too weak to ensure completeness and consistency testing. To ensure of individual DL contents, it is important to link rich metadata languages to techniques for efficient integrity checking for multimedia repositories. Moreover, recent techniques from meta modeling and distributed software engineering will help to identify and maintain inter-source relationships. This requires deep integration of AI, image processing, and database techniques, building on experiences in view integration and maintenance, and meta-level integration of heterogeneous databases.

- **Lineage**, simply put means the history of a data set. By history, we mean the recounting of the life cycle of a data set, from its collection or acquisition, through the many stages of compilations, corrections, conversions, and transformations to the generation of new interpreted products. Lineage then is that part of the data quality statement that contains information that describes the source observations or materials, data acquisition and compilation methods, conversions, transformations, analysis and derivations that the data has been subjected to.

## 2 A Reference Architecture

An important long term goal of the Digital Library is to afford a massively large number of heterogeneous classes of users (user groups), offering access to a massively large number of distributed autonomous resource repositories, in ways that are seamless, timely, and economic. Before describing our abstract architecture we make the following assumptions about two important entities: users and resources.

**Autonomous resource management**: The physical resource space would consist of a multitude of autonomous, geographically distributed Published Resource Repositories (PRR). For each PRR there would be an owning entity, managing the repository autonomously. Hence, for each PRR a management system called PRR subsystem is assumed. Also, for each PRR, a published access

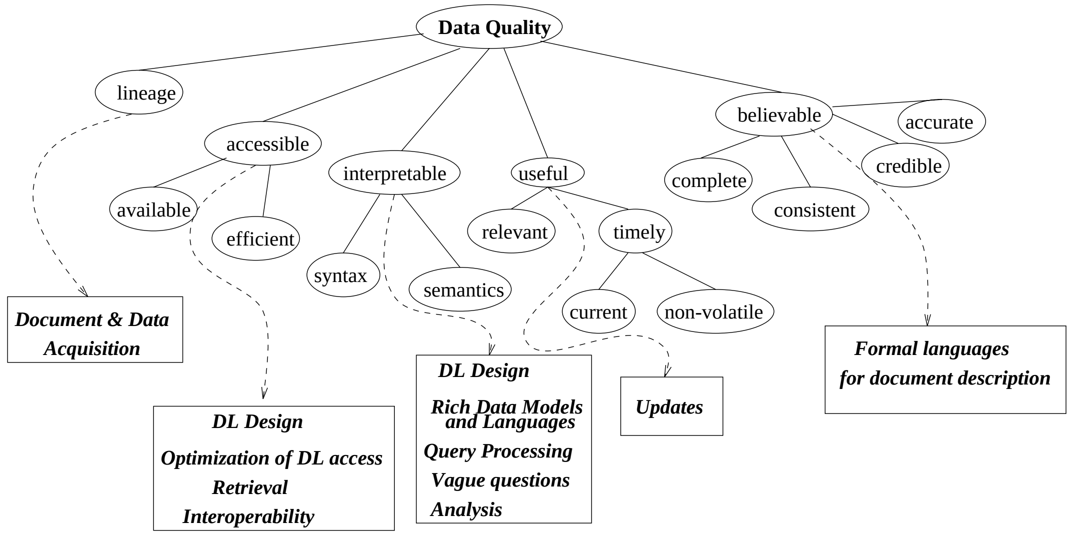


Figure 1: Data quality issues and digital libraries

schema (PAS) is defined in the framework of PRR subsystem, and is supported by the owning entity as the schema offered to potential users for accessing that PRR. For several reasons, it can be foreseen that entities offering to share their resources, would continue to exercise autonomy on their PRR. Owing to autonomy, several types of heterogeneities, in terms of resource representation, storage, and access schemas, could exist.

**Multiplicity of user interfaces**: Different classes of users, running different classes of applications would be concurrently accessing the underlying PRRs. The user-resource interface is a function of the class of users and the class of applications at hand, as well as the characteristics of the resources being accessed. Since the environment of the Digital Library is of such a scale, that the combinations of users, applications, and resources cannot be accurately identified a priori, it is simply impractical to have a single user interface accessing the underlying PRRs. Thus there is a need for supporting multiple customized (user, application, and resource dependent) user interfaces.

Notice that the necessity for multiple user interfaces expressed by the second assumption is orthogonal to the first assumption of PRR autonomy. Supporting uniform integrated access implies that the design of user interface(s) would not be affected by constraints of Published Access Schemas of different PRRs.

Our architecture (Figure 2) has three distinct layers : an Interoperability Layer (IL) managed by an interoperability protocol suite; a Resource Repository Layer (RRL) containing the different participating PRR subsystems; and a User Interface Layer (UIL).

**User Interface Layer (UIL)**: The top layer (UIL) may contain any number of User Interface Systems (UISs) each customized to the particular specification of a (user, application, resource) situation. A UIS is correct, i.e. supported in the UIL, if and only if it relies on a set of access primitives (APs) defined and supported by the Interoperability Layer. The set of APs forms the interface between UIL, and interoperability layer, and is designed to enable the UISs to view and access a *virtual integrated and structured resource space*. A given user may belong to any number of UISs. A user accessing resources by way of UISs is called a Digital Library User (DLU), since his access is managed by the digital library system. We distinguish a DLU from a user accessing a particular repository using the associated PAS, since access activity is directly managed by the corresponding PRR subsystem in this case.

**Interoperability Layer (IL)**: This layer is managed by an interoperability protocol suite. The major role of this protocol suite is to define and support an appropriate set of APs. It effectively performs a two-way mapping between the actual distributed physical resource space containing the separate repositories and the virtual space represented by the APs interface. Additional functions would be to bookkeep and manage events of join-in/walk-out, add/delete resources to a repository, administer accounting, and the like.

The interoperability protocol suite integrates a multitude of algorithms and tools, for organization, structure evolution, presentation and manipulation of both the virtual and actual resource spaces described above. Some examples are Hypermedia tools, Collaboration tools, Search algorithms, and so forth.

**Resource Repository Layer (RRL)**: This layer simply contains all PRRs and their associated PRR subsystems. Interoperability across heterogeneous PRRs in the RRL is accomplished by a contractual commitment of each PRR to support a set of resource repository primitives (RRPs). Specifically, the protocol suite of IL defines two sets of primitives at the interface between the IL and the RRL. The function of these primitives is to enable the IL protocol suite to view and uniformly access a distributed physical resource space.

### 2.0.1 Component Classes in Modular DL Toolkits

Even within a group of similar library applications (e.g., welfare case management, university geography department collections, digitized rare book collections), no two libraries have a sufficiently similar set of needs that a monolithic library software offering will satisfy many people. Individual users and individual custodians will have evolving views of what they want. We must define and design components which each enterprise, and to some extent, each user can select and combine for himself.

The software that creates DLs will include *at least* the following module classes. Here we say "module classes" because each tabulated item in the list may be represented by several implementations to create different look and feel, or to provide different data transformations, or for different hardware and operating system platforms.

- **A document storage subsystem** is a resource manager that creates a library abstraction implementing all the essential data storage, retrieval, protection, communication, and search functions as primitive operations which do not depend on the interpretation of the data handled. It is middleware that integrates more basic resource managers, such as database managers, video servers, cache managers, and file managers.

- **A source selector and fuser** is an application enabler which partitions a query among as many libraries as the query implies, calls on query managers to execute the queries, and assembles the results to hide those inter-library differences that the user considers irrelevant.

- A search engine is a resource manager which accepts a query and returns item descriptors, but not usually copies of the items themselves. There may be primitive search engines that operate only on certain kinds of database (e.g., SQL relations, inverted text indices, ...) and also more complex search engines which partition queries among simpler query engines and combine results using joins and selections which are beyond the simpler engines. Part of a search engine might be an external query optimizer intended to overcome the performance problems inherent in multiple, separated library catalogs.

- **A filter service** is a separately programmed bit-stream to bit-stream transformation that can be linked into a resource manager or an application enabler for functions like encryption, compression, and partial object access.

- **A preview/thumbnailer** is an application enabler which collects from one or more libraries a set of small data objects ("thumbnails"), or creates each such thumbnail from a library item, to present these to a user for selection. Each thumbnail remains bound to the associated item so that selecting it can be used to drive item retrieval or some other action.

- **A presenter or renderer** is an application enabler which prepares an object for manipulation in a client machine, possibly assembling item parts from one or more libraries.

- Indexing tools, document analyzers, and other tools to recognize patterns and structure are application enablers for creating search indices automatically.

# 3 Conclusion

Quality of Service (QoS) is defined as a set of perceivable attributes expressed in a user-friendly language with parameters that may be subjective or objective. Objective parameters are those related to a particular service and are measurable and verifiable. Subjective parameters are those based on the opinions of the end-users. We believe that quality of service should become an integral part of digital libraries and users should be able to query by requiring a quality of service from the system. In the framework of an NSF project, we are investigating formal specification languages for metadata and a prototype implementation based on the principles described in this paper.

# References


[1] D. Arnon, R. Beach, K. Mcisaac, and C. Waldspurger. CaminoReal: An Interactive Mathematical Notebook. In J. C. van Vliet, editor, *Proceedings of the International Conference on Electronic Publishing, Document Manipulation and Typography, Nice, France*. Cambridge University Press, April 1988.

[2] S. Ba, A. Hinkkanen, and A. B. Whinston. Digital Library as a Foundation for Decision Support Systems. In *Proceedings of the First Annual Conference on the Theory and Practice of Digital Libraries, College Station, Texas, USA*, June 1994.

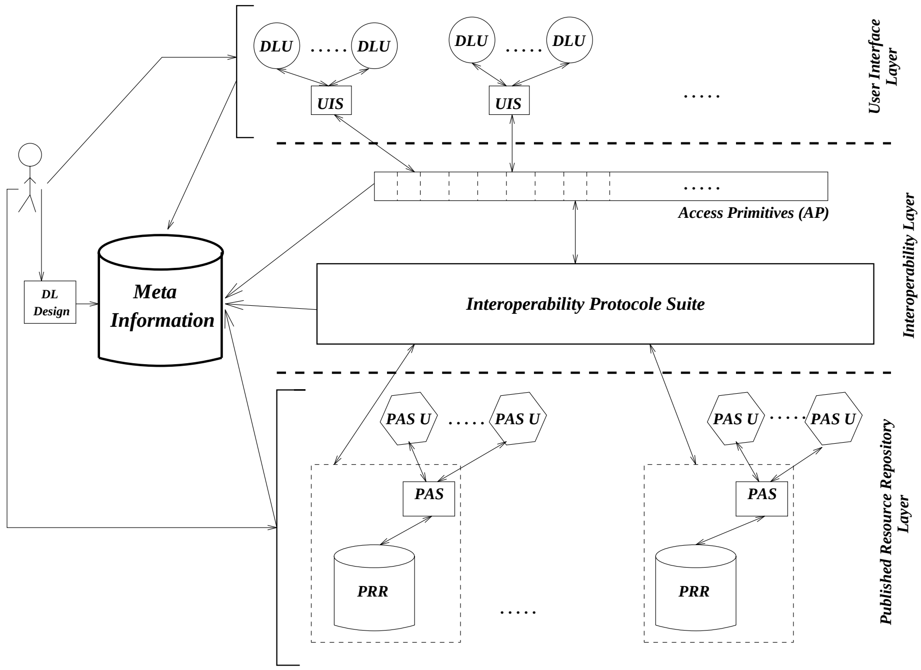


Figure 2: A reference architecture for digital libraries

[3] E. A. Bier and A. Goodisman. Documents as User Interfaces. In R. Furuta, editor, *Proceedings of the International Conference on Electronic Publishing, Document Manipulationn and Typography, Gaithersburg, Maryland, USA*. Cambridge University Press, September 1990.

[4] C. L. Borgman et al. Social Aspects of Digital Libraries. UCLA-NSF Social Aspects of Digital Libraries Workshop, November 1996.

[5] A. Hinkkanen, K. R. Lang, and A. B. Whinston. On the Usage of Qualitative Reasoning as Approach Towards Enterprise Modeling. *Annals of Operations Research*, 1993.

[6] D. Hutchinson, G. Coulson, A. Campbell, and G. Blair. Quality of Service Management in Distributed Systems. In M. Sloman, editor, *Network and Distributed Systems Management*, pages 273–303. Addison-Wesley, 1994.

[7] R. Staehli, J. Walpole, and D. Maier. A Quality-of-Service Specification for Multimedia Presentations. *ACM Multimedia Systems*, 3:238–263, November 1995.

[8] Y. Wang, M. P. Reddy, and H. B. Kon. Toward Quality Data: An Attribute-Based Approach. *Decision Support Systems*, 13:349–372, 1995.