CONSTRUCTING DIGITAL LIBRARIES WITH QUALITY

Ewerton M. Salvador, José Jorge L. D. Jr., Frederico B. Fernandes, Ed P. Bezerra
Departamento de Informática – Universidade Federal Paraíba (UFPB)
CEP: 58051-900, Paraíba – PB – Brasil

ABSTRACT

Digital Libraries can be considered islands of specialized collections on the Web, which have their own management policy to control publishing and access, offering an environment where the data are structured to allow information retrieving with quality (unlike the general Web). A pattern usually applied to describe metadata on Internet (and encouraged by W3 Consortium) is Dublin Core. This paper describes a methodology created to publish three Digital Libraries, which adopts Dublin Core and a video transmission policy, intending to increase even more the quality offered by these libraries. A Database was necessary to store these metadata. Finally, a retrieval mechanism was implemented, adopting the oriented object software pattern, called Model-View-Controller. Due to large size of videos from digital libraries, we adopted a video transmission policy distributed in Brazil.

KEYWORDS

Digital Libraries, Metadata, Video Transmission

1. INTRODUCTION

The creation of a Digital Library (DL) [Arms, 2000] involves three interrelated phases: (1) selection and digitalization of documents; (2) modeling of data and metadata; and finally (3) elaboration of the retrieval mechanism. At the first phase, the human factor is important (rare documents must be stored in a recoverable format). At the second phase, the documents are generally represented by metadata to facilitate its recuperation (metadata describes the document properties). The last phase is very important because directly affects the benefits of a DL (It must be user friendly).

One important aspect, because it determines the posterior quality of information retrieval, is whether the indexation of documents must be automatic, semi-automatic or handcraft. The indexing process is executed at phase 2 and it affects directly phase 3. The researchers in DL have investigated the construction of tools to created large multimedia collection. One of these tools is Greenstone [Witten and Bainbridge, 2003]. It consists of open code programs to construct and distribute digital collections. Other tool is Web-DL [Calado et al, 2003] that collects documents from Web, standardizes and disposes them in a DL. However, none of them treats data modeling. So their documents are recovered without criteria of usability. This paper describes a methodology created to implement the Polo of Production in Multimedia Digital Contents Polo, (project 551692/01-4 granted of CNPq-Brazil, http://www.polodigital.ufpb.br). The data and metadata modeling is strongly used, and serve as basis for recovering digital objects. The mechanism for video transmission allows, for instance, that a user reaches a video from a server geographically near of his location.

In Section 2 we describe a methodology to develop digital libraries. The Section 3 presents our conclusions about its use.

2. A METHODOLOGY OF DEVELOPMENT OF DIGITAL LIBRARIES

As with many computational systems, it is profitable to use methodologies and project patterns to obtain robust and maintainable applications. In this section we describe the process of our methodology:
digitalization; modeling of data and metadata; implementation of the retrieval mechanism; and finally the video transmission policy.

2.1 Digitalization

One of the first steps to construct a DL is the documents acquisition. According to [Cleveland, 1998], there are three acquisition methods:

- Digitalization – process to convert paper and others information media to digital format. One barrier of this method is its cost.
- Acquisition of original digital documents – acquired at academia, journalistic and literary environments.
- External acquisition – disposable at others DL or Web by means of pointers. Some tools and procedures are employed according to the type of processed material.

2.2 Metadata and Data Modeling

2.2.1 Modeling of Metadata

In order to developing an information retrieval system, a key point is how information will be retrieved. As such, a pattern must be used to describe efficiently the information in the Database (DB) to improve search and retrieval phases.

A pattern usually applied to describe metadata from documents published on Internet, and also encouraged by W3 Consortium [W3 Consortium] is the Dublin Core Metadata Project [Dublin Core]. This pattern has a group of elements, considered usual of all elements, used to describe the documents published on the Internet.

Our methodology uses the Dublin Core (DC) pattern, because it describes information in a simple and efficient way. However, many times, it is necessary to provide more elements to users regarding the particularities of the application. In Brazil, there is a standard named ABNT (Associação Brasileira de Normas Técnicas) [ABNT] that is more complete, although it is more complex than DC. In our methodology, we added elements from ABNT to DC to improve the digital object search and retrieval mechanism.

2.2.2 Modeling of Data

Database will store digital objects metadata and links from DL. We have adopted common approaches to Relational Database Design. Figure 1 shows the Entity-Relationship Diagram.

A brief discussion about each entity is presented bellow:

1. **Document** represents common attributes from all digital objects in a DL. The Dublin Core attributes were used here, because they can be applied for all published material.

2. **Type 1, Type 2, ..., Type N** represent specializing from each digital objects type (books, images, videos, etc). They have only specific attributes from specialized entity, where a lot of these attributes were used from ABNT pattern (for example, in type Article of Journal from Paulo Freire’s DL, we have used the elements section, notebook and pages).

3. **Subject** represents the keywords set used to determine the subject of each digital object. One field named glossary was added to store the meaning definition of each descriptor.

4. **Format** represents the information on the file formats that are in the DB.

5. **Type** represents the categories of the digital objects that are stored in the DB.

6. **Creator** represents the attributes of the authors registered in the DB.

7. **File** is the entity that stores information of the several files that compose the digital objects, as its links, size, name and other information that we judged to be pertinent to the context of the implemented DL.

A good policy for the metadata and link storing for digital objects in a DL is to use a Database Management System (DBMS).

We opt to the DBMS PostgreSQL, because, besides taking care of these requirements, it is open source, developed to different platforms (also of free distribution, as Linux) and is remotely accessible through tools for Windows.
2.3 Mechanism of Research and Retrieval

A good policy to implement oriented object systems is the use of design patterns; therefore they become a trustworthy application and with easy maintenance. One of these patterns is Model-View-Controller (MVC). The MVC divide the system responsibilities in three parts with the goal of to separate the business logic and data (Model) from user interface (View) and from the application flow (Controller).

**Model:** This part is responsible for the DL logic. We developed Java *classes* for each specific document type that inherit methods and properties from the main *class* of the methodology named *Document.* Some important *classes* are the following ones: Book (which represents specific data from registered books); Video (which represents data from registered videos); Format (which represents the file format from documents in DL) etc.

**View:** The data visualization part was implemented using JSP (Java Server Pages). JSP serve to create dynamic pages using the Java Language. For each presentation screen was created a JSP archive to visualize data. For example, to show the list of all books, we have created the listBook.jsp archive. Therefore, to show the details of a specific book, we have created BookDetail.jsp archive.

**Controller:** As the proper name already suggests, this is the architecture part that controls the user requests. Each requested operation is processed and the answers are sent for a presentation screen. We use the standard FrontController that it offers a centered controller to manage the request processing. In our case, the controller is implemented using Servlet technology. Servlet is a class written in Java that it can be connected in diverse servers types to expand its functionalities. Thus, we use only one *servlet* called *Control* to manage and process all user requests. In order to improve the architecture quality, we use another class, called DBAccess, that it will access the database and it will return the connection and data for Control. In this way, the controller does not need to know as the data persistence was implemented.

2.4 Video Transmission

The video distributed system provides video transmission on demand and also live. Regarding the immense size and amount of videos files, and also the bad transmission speed on server where they were initially stored, we adopted a policy that allowing user to view a video file from the server geographically nearer to him.

The idea here is spreading videos through several machines on the Web, called secondary servers. When user retrieves a video from DL, your request will be forwarded to a central machine, named manager, that it will be responsible to choose the nearer secondary server to user. The selected server will send video to user.
This process is illustrated in Figure 2. Firstly, user selects a video from DL. This request is passed through manager (part a), that it’s a machine responsible to choose a secondary server on the Web, nearer to user (part b), and re-pass this request to this server. Finally, the video recovered is transparently delivered to user (part c).

3. CONCLUSION

Best quality of the Web software is one of the main objectives in the software engineering area. According to Lynch [Lynch, 1997], Web is not created to support organized information publishing and recovering, differently of the DL.

We presented a methodology to develop DL which main steps are the following: digitalization; metadata and data modeling; implementation of the research and recovery mechanism; and a policy of videos transmission. The modeling phase combines the vantages of the Dublin Core pattern with the vantages of the Technical Rules Brazilian Association (ABNT). Metadata modeling is very important because, at the most times, unnecessary information is recovered at the Internet. So, the indexation treatment is essential. Respect to videos transmission, we verify that they are edited using formats different of the ideal format to realize their possible transmission. Also, the low Web transmission taxes have affected the satisfaction level of the Web users. We proposed a video transmission policy to resolve these problems.

REFERENCES

W3 Consortium, http://www.w3.org
Dublin Core, http://dublincore.org