dLibra — CONTENT MAINTENANCE FOR DIGITAL LIBRARIES

Paweł Gruszczyński
Cezary Mazurek
Stanisław Osiński
Andrzej Swędrzyński
Sebastian Szuber
Poznań Supercomputing and Networking Center,
ul. Noskowskiego 10, 61–704 Poznań, POLAND
phone: +48 61 858 20 30, fax: +48 61 852 59 54
e-mail: {grucha,mazurek,stachoo,kokosz,szuber}@man.poznan.pl

KEYWORDS

Digital library, content management, information society

ABSTRACT

In this paper issues of content management in digital libraries are addressed. We present three main factors that influence the quality of the digital library content organization and maintenance. We argue that, apart from sophisticated end-user tools, modern digital library systems must provide means for hierarchical content organization, document versioning and advanced access control. We also include a brief description of *dLibra* — a digital Library Framework developed by Poznan Supercomputing and Networking Center — with respect to the content management and maintenance facilities.

INTRODUCTION

With the development of the Internet, possibilities and needs for building digital libraries dramatically increased. One of the basic practical applications of such systems is using them as a building block of a global infractructure of services and applications for information society. National programme *Pionier: Polish Optical Internet* is an environment for building and making available those applications and services in Poland [1].

As the implementation work on digital library frameworks proceeds, end-users are delivered more and more sophisticated software tools for creating, viewing and searching electronic documents. Nevertheless, the constant growth of the number of available publications poses difficulties in several areas:

- content organization,
- duplicated, withdrawn and multiple document versions,
- access management and accounting,

Thus, apart from comprehensive end-user support, a digital library system must provide means for content management and maintenance. Such approach will greatly improve the quality and efficiency of the viewing and publication processes.

In the next three sections we discuss the above issues in general and in the last section we give an outline of the *dLibra* Digital Library Framework. To provide additional background information, we also briefly refer to similar functionality offered by the SCHOLNET Digital Library Testbed [2].

CONTENT MANAGEMENT

In authors' and readers' best interest is that documents are easy to find. However, in the world of digital publication a solution to this problem is not as easy to find. The difficulty obviously lies not only in the lack of good computer systems, which can assist readers in the process of searching, but also in our human nature. Authors are rarely concerned with supplying their work with the appropriate metadata description to make the publication easy to find. On the other hand, excessive medatata information (i.e. many irrelevant keywords) will much decrease the acuracy of searching.

The problem is obviously not solvable by means of a computer system. The only thing that can be done is to equip authors with a tool which makes inserting metadata to documents as painless as possible (e.g. automates the process to some extent), publishers with a tool which allows to easily control the authors' work and, of course, readers with a tool which allows to define queries easily and intuitively.

One solution is a very well-known and widely adopted hierachical catalogue, which has been in wide use long before the Internet was born. However, the question is: Are we using all the features which this structure offers in an electronic world? Consider the following issue. Should the author of fifty articles on neurobiology be allowed to publish a critical essay on Socrates philosophy? Probably not unless he can explain such a sudden change of his interests. This issue is discussed in more detail in section *Access Management* of this article.

The second problem can be best described by an example. If a reader is interested in one article about cat breeding then he is probably also interested in other works concerning this subject. Suppose that we have a branch in our catalogue (in *dLibra* such branch is called *directory* as a comparison to a computer filesystem) named "Cat breeding" and that we have only one article in it. A reader reads the article and then forgets about the library because he does not want to check every day if something new has appeared. We encounter a similar problem when a reader has found an article by running a search engine on "cat" & "breed" keywords.

The answer is a so-called *subscription service*. A reader can mark a directory as interesting to him, choose a form of notification and wait for information from the system if something new has appeared in this directory. Similarly, a reader can define a query which will be periodically performed by the system. If the query finds a new document, then the reader is automatically notified about the fact.

DOCUMENT VERSIONING

Documents and their Components

A document exists in many versions throughout its lifecycle. The first pre-print or draft version is replaced by its consecutive successors and eventually a final version is created. Some of the documents are under continous development, legislatives or software specifications being two of many examples. The problem of document versioning is not encountered in the world of well-known paper publications, because once printed on paper the document can never change. When the next version comes out, it has a different publication date, different ISBN number, probably different form and is in fact another document. This is, however, not true for electronic publications. An author or a publisher can switch to the next version available in electronic form (for example as an HTML web page) or remove the draft version completely until the next version is available without notice to anybody. A full coverage of this issue can be found in [3].

The answer to the problem is a system which assists an author and a publisher but also the readers in keeping track of the document's versions. This answer is not so obvious as it may seem, however. On the contrary to a paperwork, an electronic publication consists of many parts or *modules* [4], possibly independent to some extent. For example a web document can consist of many HTML files and a few graphic, sound, or video files as well. Each of these modules can be changed independently so they shall be tracked separately by the system.

A change made to a module does not however necessarily mean that the author wants to release a new version of the whole document. Consider an example document which is a book consisting of many long chapters. Each of the chapters is stored in a separate HTML file. It is fully understandable that during the process of preparation of the next version of this book, the author probably wants to improve more than one of the chapters. Only after changes have been incorporated into all of the chapters, does the author want to publish a new version of his work. Thus, there is a need for versioning on two levels. The first level is a module level when progress is tracked separately for every module (HTML file in our example) and on this level only the author can access all of the versions of modules. The second level is a document level when progress is tracked for the whole document. After a version of the document has been made available to public, it should not be taken away because it could create dead links and broken references from other documents. Consequently, when a new version of a document has been made available to the public, the old one is not removed, but is still in place so that the consistency with other documents is maintained.

Versions and Readers

A possibility of creating many versions of the same document is very attractive but it has its drawbacks. Let us consider the following scenario. A reader has found a very interesting article in a digital library. He has read it and made a research on his own, inspired by the article. Meanwhile, the authors of the article achieved new results and improved their article by adding new conclusions. These results and conclusions are probably of great interest to the original reader. The question is: how does he become aware that a new version of the article is available? He can of course check every day if there is a new version of a document but, given the number of documents available in electronic form nowadays, it may become very time consuming.

The answer is once again a subscription service. Using this service a reader can mark an interesting article to be tracked by the system. As soon as a new version of the document becomes available, the reader is notified of the fact by an e-mail or by different means.

ACCESS MANAGEMENT

A digital library or other electronic publishing system is a big infrastructure and needs proper access management. To achieve maximum flexibility, access restrictions should be applied on different levels of the library objects hierarchy. This section deals with three basic access management levels.

Let us consider the metadata by which authors describe their documents. The Dublin Core Metadata Element Set [5] is good for at least the majority of scientific publications but consider for example a set of fairy tales and a parent who wants to find a story which suits his needs. So he would like to search for a fairy tale designated for a child older than six and younger than ten. There is no possibility to prepare such a query using the Dublin Core Set mentioned above. There is an obvious need that a modern digital library system allows using more than one *metadata scheme*.

A question appears: who shall prepare such schemes in the system? Definitely not a developer or an administrator of the system because they lack the field knowledge and can only guess what is important and what is not for a given audience and type of documents. Leaving it to the authors is not a good choice either, because we would end up having one metadata scheme for each and every document in the library, which is even bigger mess, than not having metadata at all. The answer is that a person or a group of people should be chosen and given the rights to design and adjust metadata schemes for a given library. This is the first and most general level, which is the *library level* of right management.

Field knowledge is even more important as far as the hierarchical structure of library directories is concerned. Somebody has to make the subdirectories in the Biology directory and it apparently should not be an expert on atomic physics — the system should allow giving rights to modify the structure of each of the directories separately. The same applies to the permission to publish a document in a library directory. This is the *directory level* of access management.

Once a document is put in the library, the right to modify it is assigned to its creator. But there can be more than one author of the document. The system should then allow to grant rights to access and modify a document for more than one person. However, not everybody engaged in the document preparation process should be allowed to modify it. For example, people who are just reviewing the document and preparing comments or people who are accepting the document should not be allowed to modify it since they are not the authors. On the other hand, they should be allowed to access the document even before it was published because it is what their work is all about. All authors, reviewers and readers have their rights to the document but each right is different. This difference is meaningful only with regard to a specific document, so this access management level is the *document level*.

CONTENT MANAGEMENT IN *dLibra* DIGITAL LIBRARY FRAMEWORK

dLibra Digital Library Framework has been developed by Poznan Supercomputing Networking Center since 1999 [6][7]. *dLibra* facilitates all phases of a digital publishing process by supporting three basic groups of users: readers, writers and publishers.

Using a web-based interface the readers can easily browse the library and view selected publications. A search engine enables them to issue a query regarding various multilingual metadata attributes (e.g. using Dublin Core attribute scheme) such as the publication author, title, description, keywords, creation date and many more.

The editors are delivered intuitive GUI-based tools for placing new publications in the library and retrieving publications or some of their components for further editing. An advanced versioning system supports managing subsequent revisions of publication objects as well as branching.

The publishers receive tools for managing the whole library structure, in particular, putting out and hiding publications, managing access rights and other library resources.

Content organization

The whole library content is organized in a hierarchical structure of entities. A directory is an entity that groups any number of other items - subdirectories or publications. A publication is a unit of information (e.g. an article or a book) that consists of one or more basic objects of various types (e.g. HTML or image file). An example *dLibra* content structure is shown in figure 1.

The use of the hierarchical directory structure enables the publishers and library administrators to divide the whole library content into smaller areas, accordingly to e.g. the coverage or importance of the material. Additionally, with a comprehensive access management system, it is possible to assign readers and writers to particular parts of the library resources so that the search results are more accurate and new publications are placed only in appropriate directories. To support interdisciplinary documents, a system of links will be introduced that will make it possible to place a single publication in many directories of the library. The hierarchical structure of the publication itself enables the authors to logically group together modules of different types to make the digital material more attractive and comprehensive.

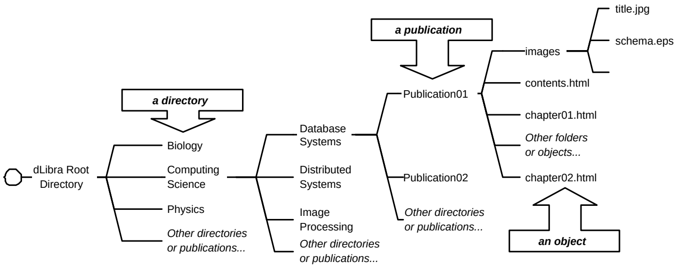

Figure 1: *dLibra* hierarchical directory structure

Every entity in the library — from the root directory down to a single publication object can be described by means of user-defined attribute schemes (e.g. Dublin Core). The values of attributes can be defined in several user-defined languages and are considered while searching the library.

Document versioning

dLibra provides support for both publication- and module-level versioning. A publication can be made available for public viewing by creating an edition, which is a set of certain versions of publication objects. Every publication can have an unlimited number of editions comprised of different versions of publication files or even different files. To explain the idea of object versioning and publication editions let us assume there is a publication consisting of only three files: `body.html`, `title.jpg` and `logo.gif`. Figure 2 shows how can subsequent versions of these files make publication editions.

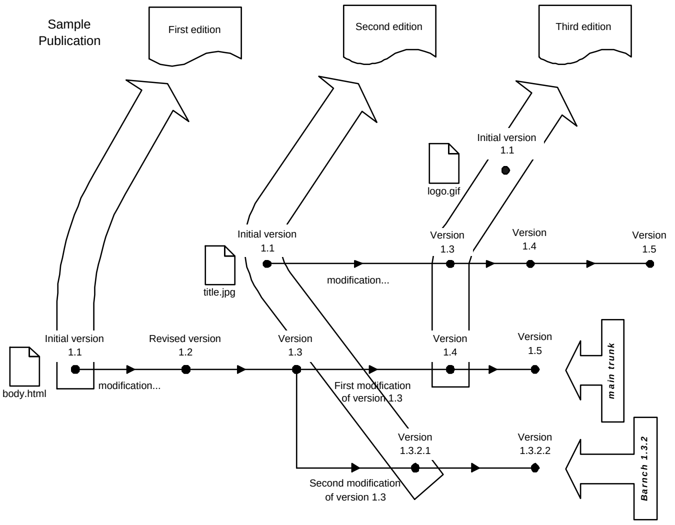

Figure 2: Object versioning and publication editions

In the figure, the sample publication starts from only one file — `body.html`. The first edition of the publication contains only this file, whereas the other files may or may not exist as well. Each publication object is versioned, which enables the editors to store subsequent revisions of the object and to place them in different publication editions. If there is a need for creating two or more object versions based on the same revision a branch can be created. The second edition of the sample publication, apart from newer reversion of `body.html` file, contains a new file: `title.jpg` in its initial version. The third edition comprises all the three files.

After creating an edition and making it accessible for public viewing, the author is unable to withdraw it from the library so that all references remain valid. It is, however, possible to create and publish another edition with modified content. Again, with the access management system, the roles of writers (content providers) and editors (reviewers) can be separated to increase the capabilities of the digital publication life cycle.

In the SHOLNET Digital Library Testbed a slighty different approach to publication structure and versioning is adopted. Apart from physical publication components, document view, which is a specific intellectual expression of a document instance, is used. This enables to present the document in a full text form, its abstract or e.g. metadata description. However, less emphasis is put on publication versioning (no branching support) and component reuse.

Access management

Access management in the *dLibra* library is based on a system of users and groups. A user can be made a member of any number of groups, which may be considered as an assignment of a specific role (the user inherits all the rights granted to the groups of which he is a member). Rights can be granted on a library, directory or publication basis. On each level, several access sublevels have been defined to enable a precise definition of user roles.

Library level access restrictions affect the functioning of the whole library:

- attribute scheme management
- file type hierarchy management
- user and group information management

Directory level access restrictions regard:

- directory visibility — some of the directories can be made invisible to particular users or groups
- permission to list a directory contents
- permission to read the contents of the publications contained in a directory
- permission to edit the directory structure (creating and removing empty subdirectories)
- permission to place publications in a directory
- right management for a directory and all its subdirectories

Publication level access restrictions regard:

- reading published editions a publication
- permission to edit the publication objects
- publication management (granting rights, branching, publishing)

In the SCHOLNET testbed, acces management tasks rest with system administrators. Every submission, withdrawal or replacement of a document involves a decision of an administrator of the appropriate part of the library. The advandate of such approach is its flexibility — every request can be handled individually. Nevertheless, the automated access control seems to be sufficient and less costly in most cases.

System model

dLibra Digital Library is implemented as a client-server system (figure 3). On the server side there are a number of independent modules connected via network intefaces. All modules are implemented using Java 2 techology, in particular RMI (Remote Method Invocation), JDBC (Java Database Connectivity) and JSP/Servlet technologies. Currently the data storage module utilizes the Oracle database system. Nonetheless, because of the use of the JDBC and SQL 92 standard *dLibra* can be easily ported to work with any other RDBMS. An event module built in the *dLibra* system provides a possibility of adding extension modules without modifying the already existing ones. On the client side GUI (JavaSwing) applications are provided that support publication creating and library management processes.

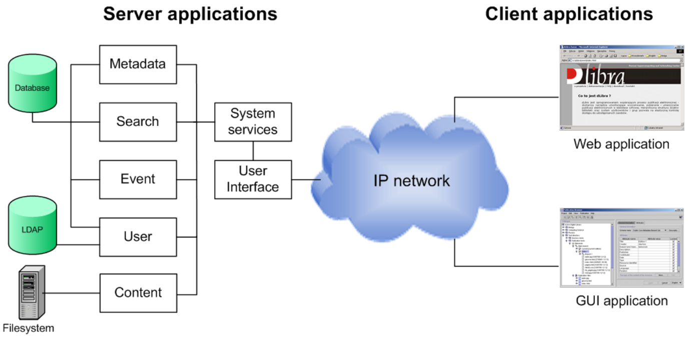

Figure 3: *dLibra* internal architecture

CONCLUSIONS AND FURTHER WORK

In this paper we have discussed some issues of content management in digital libraries. These observations are the result of work on system of digital libraries being conducted under the PIONIER programme. The system is one of the most important elements in general infrastructure of the programme. The system makes it possible to use achievements of the PIONIER programme like distance learning, groupwork applications etc. by information society.

We identified three crucial factors that contribute to the overall quality of publication and library maintenance processes: content organization, document versioning and access management. When attempting to build a large and well organized digital library, none of the elements can remain underestimated. During the design and implementation of the *dLibra* system we have put much emphasis on the library maintenance issues. The system proves to be an efficient tool for publishing and managing digital documents. Some aspects of the publishing process are currently investigated and have not been addressed in this article, e.g. groupwork and workflow management, document preparation process. Research and development will be continued to incorporate these ideas into the *dLibra* system.

Presently, *dLibra* is being put into practice in PSNC in order to facilitate publication, storage and access to company internal documents such as articles or reports. This will enable to evaluate and draw conclusions on the system's performance in an average workload environment.

References

[1] Jan Węglarz et al. *PIONIER — Optical Internet in Poland*. ISThmus, Poznań, April 2000.

[2] Donatella Castelli and Pasquale Pagano. *SCHOLNET — Global System Architecture Report*. CNR-IEI, Area di Ricerca di Pisa, 56124 Pisa, Italy, http://www.ercim.org/scholnet/del/D2.2.1-V2.pdf, January 2002.

[3] Joost G. Kircz. New practices for electronic publishing: how to maintain quality and guarantee integrity. In Dennis Shaw, editor, *Proceedings of the Second ICSU-UNESCO International Conference on Electronic Publishing in Science*. http://associnst.ox.ac.uk/∼icsuinfo/proc01fin.htm, 2001.

[4] Frédérique Harmsze. *A modular structure for scientific articles in an electronic environment*. PhD dissertation University of Amsterdam, http://www.science.uva.nl/projects/commphys/papers, 2000.

[5] Dublin core metadata element set, version 1.1: Reference description. Technical report.

[6] Cezary Mazurek and Sebastian Szuber. *Development of Digital Libraries at Poznań Supercomputing and Networking Center*. ISThmus 2000. Research and Development for the Information Society, Poznań (Poland), April 2000.

[7] Cezary Mazurek, Maciej Stroiński, and Sebastian Szuber. *Digital Library for Multimedia Content Management*. ERCIM 9th DELOS Workshop, Digital Libraries for Distance Learning, Brno, April 1999.

BIOGRAPHY

CEZARY MAZUREK, born in 1969, received his Master's Degree in Computer Science at the Poznan University of Technology in 1993. He worked for Poznan University of Technology between 1993 and 1994 and from 1993 for the Poznan Supercomputing and Networking Center (PSNC). He is currently the head of Network Services Department at PSNC. He is leading the development of services based on Internet technologies (e.g. Digital Library Framework: dLibra, Polish Educational Portal in co-operation with Interkl@sa, Multimedia City Guide in co-operation with Poznan City).