A CONCEPTUAL MODEL FOR DIGITAL LIBRARIES EVOLUTION

Andrea Baruzzo, Paolo Casoto, Antonina Dattolo and Carlo Tasso
Dept. of Computer Science, University of Udine, Via delle Scienze 206, Udine, Italy

Keywords: Digital libraries, Metadata, Service-oriented architecture, Multi-agent systems, Schema evolution.

Abstract: The evolution and preservation of digital libraries are not simply a matter of technological decisions, but they can be better understood if treated as the integration of three complementary dimensions (based on the informational, technological and social domains). These dimensions together form a conceptual framework suitable to characterize the whole digital library concept.
In this paper, starting from the experience and the lessons learned in the realization of the EU-India E-Dvara project, we propose such framework, providing motivational examples and discussing opportune solutions. More in particular, we discuss the issues concerned the technical infrastructure adaptation, the coordination of different user roles, and the data evolution in order to select the dimensions along which we base our framework.

1 INTRODUCTION

Many works coming from both the academia and the industry seem to suggest that preservation and evolution of digital libraries are firstly a matter of technological issues (Barkstrom et al., 2002). We recognize the need of data storage infrastructures, knowledge management systems (metadata and search mechanisms) or data transport and security facilities. However, the technology should be viewed ”simply” as a means to provide the services typically built around a digital archive. We recognize a deeper meaning in the evolution phenomena of digital libraries, taking into account also social aspects such as the diverse range of actor roles involved in the content production and exploitation processes. Thus, we contrast the “technology-centered” vision, characterizing the evolution of both the digital content and the services built upon it as the integration of three complementary dimensions (social, technological and informational). Such dimension form together a conceptual framework suitable to better formalize the digital library concept and its evolution issues over the time.

This paper is based on a three-years experimentation with the EU-India E-Dvara project[1]: a digital platform devoted to e-content management in Indian heritage and sciences (Challapalli et al., 2006; Baruzzo and Casoto, 2008; Baruzzo et al., 2008).

The main contribution of this work is the characterization of a digital library according to its evolution aspects; in particular, we:
1) introduce a conceptual framework to handle the evolution of digital archives along multiple dimensions (Section 3);
2) provide representative examples concerning evolution issues, weaknesses, and mistakes emerged during the evaluation of our current E-Dvara prototype (Section 3.1 - Section 3.3);
3) propose a new, distributed approach to handle evolution open problems (Section 4).

2 RELATED WORKS

In the last few years, several research projects have been proposed in order to cope with data preservation and organization (Bekaert et al., 2005; Lutzenkirchen, 2002; Candela and Pagano, 2007). For example, the storage of XML-based document, one of the core architectural properties of E-Dvara, has been previously proposed in Greenstone (Bainbridge et al., 2001; Witten et al., 2000), a digital library designed to provide librarians with the ability to create and publish heterogeneous collections of digital contents on the Web like text, images, videos and e-books. Each content in Greenstone can be described using *metadata*, either imported from standard schemas (e.g. Dublin

[1] http://edvara.uniud.it/india

Baruzzo A., Casoto P., Dattolo A. and Tasso C.
A CONCEPTUAL MODEL FOR DIGITAL LIBRARIES EVOLUTION.
DOI: 10.5220/0001835802990304
In *Proceedings of the Fifth International Conference on Web Information Systems and Technologies (WEBIST 2009)*, page
ISBN: 978-989-8111-81-4

Core[2]) or manually provided by librarians. However, Greenstone does not provide any policy or roles for the management of the content submission process. Moreover, it does not provide functionalities concerning the evolution management of both contents and collection templates.

D-Space (Tansley et al., 2003) is an author-oriented distributed digital library aimed at providing long-term preservation of heterogeneous contents, by improving some of the limitations affecting Greenstone. It provides long-term preservation facilities, by assigning a persistent identifier to each submitted resource and supporting software and hardware methodologies for data backup and content versioning. D-Space introduces also a multi-roles approach to content publishing, identifying the following actors: (1) authors and organizations, providing the contents, (2) librarians, performing content validation, and (3) users, interested in content retrieval. Content-based workflows can be customized in order to cope with the needs of specific organizations. Part of the policies defined in D-Space have been introduced also in E-Dvara to structure content and to delegate proper activities to different stakeholders.

Service-oriented architecture and data interoperability issues in digital libraries have been explored also by the Fedora Project (Lagoze et al., 2005), a distributed architecture for contents publishing, aggregation and retrieval. Composite information is obtained by means of aggregation of physical contents, viewed as bit-streams, located worldwide into the Fedora repositories. Preservation of each content is achieved by means of a naming service, which can be used to access the selected content. In addition to composition, Fedora provides users with the ability to define new contents by applying to existing physical objects custom components called disseminators (e.g.: a thumbnails generator applied to high-resolution pictures or videos). Both Fedora and E-Dvara allow content editors and archivists to define semantic connections between archived contents. In Fedora, however, connections are defined between two contents treated as set of physical contents. E-Dvara, vice versa, allows content writers to define relations implementing a specific template which enhances a closer *semantic validation* of the content.

The above mentioned systems are centred on contents, defined as *binary resources* enriched by metadata devoted to preservation, storage and retrieval purposes, but not intended for data structuring, as we do in E-Dvara. Thus, preservation and evolution of a data model is implemented as a low-level mechanism, where data is processed as bit-streams instead of as instances of well-defined structures (i.e. XML Schema). In E-Dvara we provide preservation facilities, managing both physical and *logical evolution* of the stored data. More specifically, E-Dvara is conceived to explicitly deal with evolution of a data model by means of preserving *information integrity*.

3 CONCEPTUAL FRAMEWORK AND EVOLUTION ISSUES

Our conceptual framework is based on the topology provided by Yates (Yates, 1989)[3], by incorporating the vocabulary suggested by Rowlands-Bawden (Rowlands and Bawden, 1999). The evolution of each concept (point) in the topology is described by considering the different directions from which it can be reached. The result, illustrated in Figure 1, highlights three specific domains:

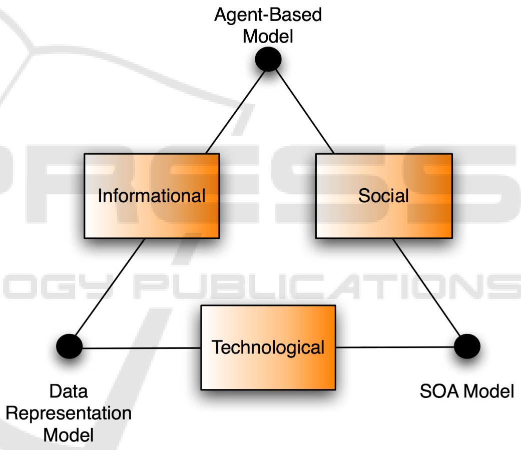
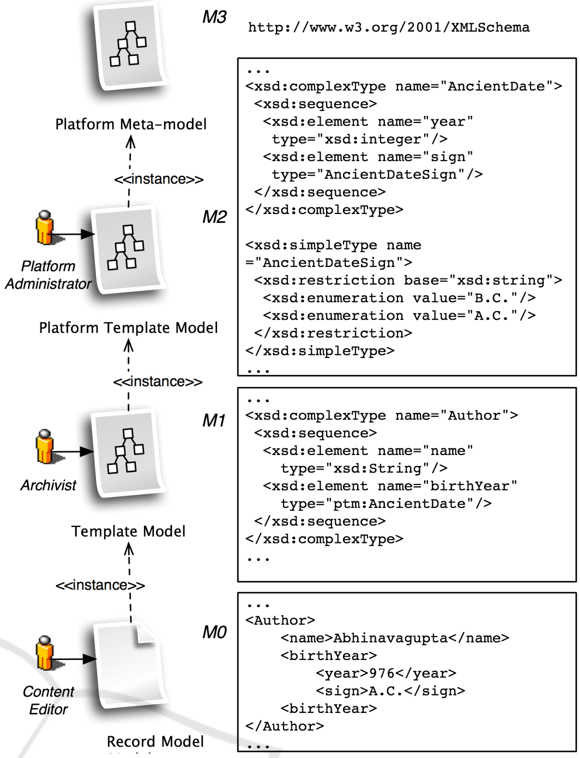


Figure 1: The evolution conceptual model.

1. the *Informational domain*, which describes knowledge organization and description (e.g. metadata).
2. the *Technological domain*, which describes knowledge organization and discovery (e.g. software agents), technical impacts on the information transfer chain, technology factors (e.g. human-computer interaction).
3. the *Social domain*, which describes human and organizational factors, information laws and policies, social impacts on the information transfer chain, and library management concerns.

[2] See for more details: http://dublincore.org/

[3] In the original Yates' model, these domains were called *documents*, *technology*, and *work*, respectively.



```
http://www.w3.org/2001/XMLSchema
```

```
...
<xsd:complexType name="AncientDate">
 <xsd:sequence>
  <xsd:element name="year"
   type="xsd:integer"/>
  <xsd:element name="sign"
   type="AncientDateSign"/>
 </xsd:sequence>
</xsd:complexType>

<xsd:simpleType name
="AncientDateSign">
 <xsd:restriction base="xsd:string">
  <xsd:enumeration value="B.C."/>
  <xsd:enumeration value="A.C."/>
 </xsd:restriction>
</xsd:simpleType>
...
```

```
...
<xsd:complexType name="Author">
 <xsd:sequence>
  <xsd:element name="name"
    type="xsd:String"/>
  <xsd:element name="birthYear"
    type="ptm:AncientDate"/>
 </xsd:sequence>
</xsd:complexType>
...
```

```
...
<Author>
    <name>Abhinavagupta</name>
    <birthYear>
        <year>976</year>
        <sign>A.C.</sign>
    <birthYear>
</Author>
...
```

Figure 2: XML Representation of Data Model.

The open issues faced during our experimentation with E-Dvara may be classified along three evolution dimensions:

1. *Informational-Technological* dimension, which identifies all *data evolution* problems due to changes in the underlying data model (data schema);
2. *Technological-Social* dimension, which identifies problems concerning the need to adapt the technical infrastructure of a digital library in order to fulfil new user requirements.
3. *Social-Informational* dimension, which concerns the different conceptual models needed to support the work of such different community of users, and their impact on the documents, by providing support to roles and workflows.

3.1 The Data Evolution Problem

The first prototype of E-Dvara provides users with a flexible way to define and update the metadata associated to each project representing a digital archive. In particular, users can define a set of *schemata* which supplies the structure adopted for storing documents. Each schema is expressed in terms of *fields*, *data types* and *constraints*. Metadata definition and update can take place every time during the digital collection life-cycle, leading to the problem of correctly handle the evolution of data defined by these *mutable templates*; in fact, each schema update should be properly spread to the previously validated archives, in order to automatically adapt the existing content to the new schema (or to provide modelers with the feedback necessary to manually fix the problem).

Examples of this process can be defined considering the data model illustrated in Figure 2; starting from the schema at level M1, user may insert more information into the Author element by *adding* new fields (`PlaceofBirth`, `Nationality`), *modifying* existing fields (splitting `Name` into `Suffix`, `FirstName`, `LastName`, `Prefix`) or *removing* fields which are considered no more useful. Moreover, user may also be interested in moving from the `AncientDate` format to a type representing dates in a modern way.

3.2 Technical Infrastructure Adaptation

One of the recurring issues we have faced during development of the first prototype of E-Dvara was the request for integrating new heterogeneous functional modules at the top of the digital library (e.g. virtual museums, meta-search engines, or applications for mobile devices). These requests lead to several issues such as ad-hoc business logic customization and service duplication; these issues clearly demanded for a *reusable integration layer*.
Moreover, we have learned also that integrating several applications in a common environment requires a substantial investment in understanding and implementing their *orchestration*, in order to handle incompatibilities between different business logics in a standard and transparent way.

3.3 Roles Coordination Issues

The integration of different applications, concerning different domains, may lead to new requirements involving user roles and the policies they were subjected to during information access. For example an external service, in according to its own data management policy, states *when* a particular workflow is required to organize the archived contents. In E-Dvara, a *workflow* expresses a set of roles, related activities and constraints that define together the structure of the information management process.

As an example of such a workflow, consider the curator of a digital museum which has to arrange a new gallery, composed by paintings, ancient books and movies hosted in three projects and owned by three different users. Consider now the case in which the curator wants to incorporate in the same gallery a set of features to search, organize and enrich the existing records, by adding new fields describing the position each item will have in the 3D rendering of the virtual museum. Moreover, final users may also be interested in improving the quality of the exhibition, by creating new relations between the existing content (e.g. opinions and links to a specific content in a typical Web 2.0 style).

These scenarios pose several issues that must be faced in order to provide flexibility in the way data management is achieved.

4 HANDLING EVOLUTION

This section proposes a distributed approach to handle the evolution problems discussed in Section 3.

4.1 Evolution Along the Informational - Technological Dimension

In order to handle the evolution problems concerning the changes in data format and schemata described in Section 3.1, we propose here a four-layer data representation model (Figure 2).

Records (level M0, Record Model) are aimed at representing the archived data; a record is an instance of a document stored in the digital platform. Every record must conform to a *document template* (level M1, Template Model), providing structural definitions (e.g. the document contains the `Title`, `Author`, and `Date` fields) and constraints (e.g. the `Data` field must conform to the `mm/dd/yy` format or the `Title` field is mandatory). Document templates are themselves conformed to a *platform template* (level M2, Platform Template Model) devoted to define both business rules and data types the archivists can use to build document templates (e.g. each record in every archive must contain the `CreationDate` and `Owner` fields). Finally, platform templates are instances of a more general layer, the *platform meta-model* (level M3, Platform Meta-Model), which defines a set of common low-level structures (e.g. primitive data types as `xsd : String`) and operations (e.g. data sequencing) available in order to define more complex data structures. This level is that of the W3C XML Schema specifications[4].

The overall data model involves the interaction with three different actors:

- *Content editor*, devoted to data entry, with respect to a specific document template but not allowed to perform any template change.
- *Archivist*, devoted to document templates definition.
- *Platform administrator*, devoted to the management of platform templates.

This hierarchical data model provides *automatic data validation policies*, which play a central role in our vision. Indeed, validation is applied both to the templates and (recursively) to all the records stored in the platform archives. Templates which do not respect the business rules defined in the platform template model should be manually updated by either archivists or content providers in order to become consistent. A detailed description of the proposed data model has been presented in (Baruzzo and Casoto, 2008; Baruzzo et al., 2008).

4.2 Evolution Along the Technological - Social Dimension

In order to handle the evolution issues concerning the adaptation to new requirements such as the integration of heterogeneous services described in Section 3.2 we base the second prototype of E-Dvara according to a Service-Oriented Architecture (SOA) model (Figure 3), characterized by:

- the introduction of an explicit *integration layer*, which unifies the interfaces of different subsystems into the same interoperable environment.
- the migration toward autonomous and composable *services*;
- the adoption of a common *peer-to-peer, message-based communication protocol* supported by the *Enterprise Service Bus* (ESB), devoted to service orchestration, which acts as a centralized authority to coordinate interaction between services and applications.

At the top of our SOA architecture we have placed applications such as administration interfaces to manage users and archives, publication interfaces to produce new content in the digital library, or virtual museums to exhibit a document archive in a “museum-like” setting. All these heterogeneous modules can exploit any reusable service available in the Integration layer, (e.g. to perform searches in the platform archives).

Archives are placed at the bottom of the architecture; they are managed by two modules: the *Archive Manager* which stores and retrieves documents, and the *Policy Manager* which manages users, accessing

[4] http://www.w3.org/XML/Schema

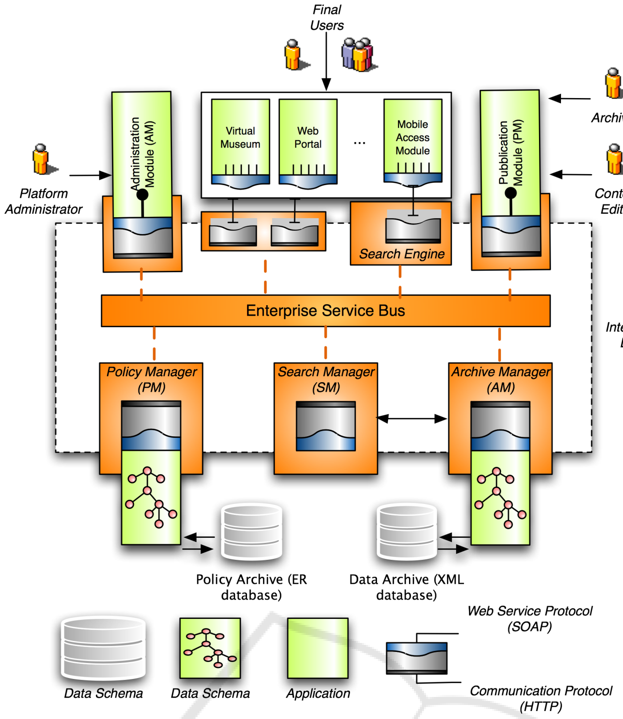


Figure 3: The architecture of the E-Dvara platform.

policies, and projects. The Archive Manager isolates the business logic needed to realize the data-model described in Section 4.1, whereas the Policy Manager implements the data validation rules, decoupling them from other architectural components.

4.3 Evolution Along the Social - Informational Dimension

The introduction of mutable templates in content representation leads to the challenge of evolution and re-evaluation of existing archives. In this section, we introduce a *multi-agent approach* to tackle the problem, aimed (when possible) to automatically resolve evolution issues.

The levels from M1 to M3, proposed in Figure 2, can be affected by updates during the digital library life-cycle. In particular, such updates can involve XML Schema definitions (level M3, with a low frequency), Platform Template Models (level M2, with a low-medium frequency) and Template Models (level M1, with a rather high frequency).
Each schema is connected by a dependency bond with: 1) the schemata on its top for validation purposes; 2) other schemata of the same project (e.g. template `Book` can be related with template `Author` by relation `WrittenBy`); 3) other schemata from different collections (e.g. template `GalleryRoom`, defined by the virtual museum application, can be related with templates `Book` and `Painting` defined in different collections). This propagation mechanism is achieved by means of a multi-agent system. Each agent is assigned to a specific schema, monitoring its evolution; an agent can interact with other agents assigned to depending schemata, send them messages and apply evolution to the instances of its schema.

A *coordinator agent* is assigned to each instance of the platform, in order to monitor the updates of the Platform Template Model and to activate the agents connected to each schema when required. The coordinator agent is also devoted to the creation of a new agent every time a new schema is defined.

A *schema agent* is devoted to the evolution of contents related to a specific template at level M1. It can perform a set of actions on the existing data, accordingly to the updates affecting related schemata.

Agents perform several evolutionary operation on data, in order to preserve data validity and, at the same time, to prevent archivists and content editors to spend a lot of time re-entering the whole set of existing contents. In (Guerrini et al., 2005; Guerrini et al., 2007) a complete taxonomy of updates, which can affect a generic XML schema, is described. A subset of the listed operations, covering the set of updates archivists can perform, has been implemented in E-Dvara, like the extraction of a vocabulary from the set of values assigned to a free-text String element. When archivist updates the type of the element Name from String to Vocabulary, the agent assigned to that schema should access each instance of the template and perform a change_item_type, verifying if the old values assigned to Name are still valid with respect to the new element type. When task is completed, the agent should notify the schema updates to the related agents (according to the dependency chain between schemata), in order to grant the consistency of any inter-dependent data.

5 CONCLUSIONS

In this paper we have extended an existing conceptual model for digital libraries, introducing the notion of evolution dimension and describing our proposal along three dimensions: Informational-Technological, Technological-Social, and Social-Informational. This characterization comes from the lessons learned during the experimentation with our E-Dvara platform. Now we are working to complete the second prototype which embodies the improvements described in this paper.

Our future plans include a validation of the overall prototype in different areas, concerning the exploitation of both information and services by means of mobile applications, virtual museums, and Web 2.0 environments.

REFERENCES

Bainbridge, D., Buchanan, G., Mcpherson, J., Jones, S., Mahoui, A., and Witten, I. (2001). Greenstone: A platform for distributed digital library applications. In *ECDL '01: European Digital Library Conference*, pages 137–148. Springer.

Barkstrom, B., Finch, M., Ferebee, M., and Mackey, C. (2002). Adapting digital libraries to continual evolution. In *JCDL '02: Proceedings of the 2nd ACM/IEEE-CS Joint Conference on Digital Libraries*, pages 242–243. ACM.

Baruzzo, A. and Casoto, P. (2008). A flexible service-oriented digital platform for e-content management in cultural heritage. In *IABC '08: Intelligenza Artificiale nei Beni Culturali Workshop*, pages 38–45.

Baruzzo, A., Casoto, P., Challapalli, P., and Dattolo, A. (2008). An intelligent service oriented approach for improving information access in cultural heritage. In *IACH '08: Information Access in Cultural Heritage (IACH) Workshop, European Conference on Digital Libraries*. Springer.

Bekaert, J., Liu, X., and Van de Sompel, H. (2005). adore: A modular and standards-based digital object repository at the los alamos national laboratory. In *JCDL '05: Joint Conference on Digital Library*, pages 367–367. ACM.

Candela, L., C. D. and Pagano, P. (2007). A reference architecture for digital library systems: Principles and applications. In *Digital Libraries: Research and Development, 1st International DELOS Conference*, pages 22–35.

Challapalli, S., Cignini, M., Coppola, P., and Omero, P. (2006). E-dvara: an xml based e-content platform. In *AICA*.

Guerrini, G., Mesiti, M., and Rossi, R. (2005). Impact of xml schema evolution on valid documents. In *WIDM '05: Proceedings of the 7th annual ACM International Workshop on Web Information and Data Management*, pages 39–44. ACM.

Guerrini, G., Mesiti, M., and Sorrenti, M. A. (2007). Xml schema evolution: Incremental validation and efficient document adaptation. In *Database and XML Technologies, 5th International XML Database Symposium*, pages 92–106.

Lagoze, C., Payette, S., Shin, E., and Wilper, C. (2005). Fedora: An architecture for complex objects and their relationships. *CoRR*.

Lutzenkirchen, F. (2002). Mycore - ein open-source-system zum aufbau digitaler bibliotheken. *Datenbank-Spektrum*, 4:23–27.

Rowlands, I. and Bawden, D. (1999). Digital libraries: A conceptual framework. *Libri: International Journal of Libraries and Information Services*, 49:192–202.

Tansley, R., Bass, M., Stuve, D., Branschofsky, M., Chudnov, D., McClellan, G., and Smith, M. (2003). The dspace institutional digital repository system: current functionality. In *JCDL '03: Joint Conference on Digital Libraries*, pages 87–97. IEEE.

Witten, I., McNab, R., Boddie, S., and Bainbridge, D. (2000). Greenstone: A comprehensive open-source digital library software system. In *ICDL '00: International Conference on Digital Libraries*. ACM.

Yates, J. (1989). *Control through communication*. The Johns Hopkins University Press.