Metadata Management in a Heterogeneous Digital Library

Bernhard HASLHOFER, Robert HECHT
Research Studios Austria – Studio Digital Memory Engineering,
Thurngasse 8/20, Vienna, 1090, Austria
Tel: +43 664 825 11 07, Fax: + 43 1 585 37 41
Email: bernhard.haslhofer@researchstudio.at, robert.hecht@researchstudio.at

Abstract: Many institutions operating in the cultural heritage domain have already made their content accessible via the Internet. However, when organizations want to realize projects that integrate content from various institutions, they face the problem of how to make this content accessible in a coherent way. Since metadata plays a key role in such scenarios, a Metadata Management system that can handle heterogeneous metadata descriptions and be easily integrated into new or existing digital library architectures would be useful. We propose a Metadata Management system that uses Semantic Web technology to handle the problem of heterogeneity and on Web Service technology to achieve easy system integration. Our solution facilitates the development of integrated digital library systems, and lifting the burden of heterogeneous metadata management from application developers.

1. Introduction

In recent years many cultural institutions have made their digitised collections publicly available using in-house solutions or commercial digital library systems. More and more they seek cooperation with other institutions in order to establish expert forums or business platforms that allow the exploitation of their digital artefacts. In addition to issues such as Digital Rights Management, one of the first obstacles such joint projects encounter is the problem of content integration from heterogeneous sources, which usually requires a high level of expertise and involves considerable costs.

Metadata repositories are a key component of such digital library systems. They provide appropriate storage for valuable information about content items and function as an entry point for search and discovery services. Many digital library systems (e.g. [1, 2]) were designed for a specific environment and follow a monolithic metadata management approach: in order to manage their content collections, they rely on a specific hard-wired metadata schema. Because such systems face the problem that they cannot be used easily in other environments, recent digital library systems rely on a heterogeneous metadata management approach (e.g. [3, 4]). The goal of these modern systems is to support arbitrary metadata schemas and to enable management of metadata that describe content in various, domain-specific ways. To ensure that metadata descriptions corresponding to various schemas are accessible in a coherent way, semantic interoperability between schemas is a key requirement. A first attempt, employing Semantic Web technology in order to deal with the problem of heterogeneous metadata, has been made in [5].

In this paper we present the Semantic Web based Metadata Management system we are developing for the BRICKS Integrated Project [6], which is part of the 6th EU Framework Program. BRICKS aims at establishing the organisational and technological foundations for a digital library network in order to share knowledge and resources in the cultural heritage domain. From a technical point of view, BRICKS will provide a distributed, component-

based, and service-oriented software infrastructure for the interoperability of heterogeneous content and services. Research institutions, industries and culture-based organisations can join the BRICKS network by making their content available or by building new application services, using the infrastructure components via their Web Service interface.

Because the member organizations of BRICKS use diverse metadata schemas to describe their content, the first requirement of the BRICKS Metadata Management system is to provide flexible metadata schema support. In the following sections we will describe how we apply Semantic web technology to provide this flexibility. Since we do not want our system to be decoupled from other digital libraries, we propose a solution which assures interoperability as well as easy integration with existing digital library systems.

2. Objectives and Key Requirements

2.1 Flexible Metadata Schema Support

In cultural heritage environments the term metadata is increasingly applied to value-added information created to arrange, track and otherwise enhance access to information objects. Metadata used to describe and identify information resources - also called descriptive metadata - always correspond to a certain metadata schema that expresses the semantics and valid structure of metadata descriptions. There exist standardized metadata schemas like Dublin Core [8], MARC [9] or MPEG-7 [10] as well as other institution-dependent proprietary schemas. Because we cannot assume that institutions are willing to transform their existing metadata to a common schema, our Metadata Management system must provide support for arbitrary metadata schemas and heterogeneous metadata descriptions.

The drawback of such a heterogeneous metadata management approach is that there is no single entry point for services making use of a metadata management system, such as search and discovery services. Generally, a metadata management system should be able to serve metadata requests not only for a given schema, but also for metadata conforming to other schemas. Therefore, the BRICKS Metadata Management system must deal with the problem of semantic interoperability between heterogeneous metadata and provide mechanisms to define mappings between proprietary and standardized schemas.

2.2 Interoperability with Existing Digital Libraries

Interoperability with other digital libraries is a key issue for the design of any digital library system wishing to disseminate its available content and metadata descriptions. In recent years, the necessity of establishing a common agreement on adoption and use of standards to facilitate efficient dissemination of metadata descriptions led to the establishment of the Open Archives Initiative and the development of the OAI Protocol for Metadata Harvesting (OAI-PMH) [11]. This protocol defines a mechanism for exchanging metadata descriptions between OAI-PMH compliant digital library systems over the HTTP-protocol.

As part of the BRICKS Metadata Management system we have implemented an OAI-PMH Harvester as well as an OAI-PMH Server to retrieve and expose metadata descriptions. However, because we store our metadata descriptions in RDF and the OAI-PMH exchanges metadata in XML, the system must insure that incoming and outgoing metadata records are transformed in a consistent way.

2.3 Easy System Integration

Within the BRICKS project, several cultural institutions should integrate the BRICKS Metadata Management system into their existing or future system landscapes. To achieve this objective we must consider two principal aspects of integration.

First, we should respect the fact that existing digital library applications were implemented in different programming languages and are now running on different hardware platforms. Adding new components such as the BRICKS Metadata Management system might be a very complicated process. To enable system integration requiring only minor modifications in existing systems, we follow the design principles of Service-Orientated Architectures (SOA) and expose our system as a language neutral and platform independent Web service component.

Second, we assume that unless the effort to import existing data is minimal, new institutions are not likely to use our system. As different digital library systems are built on top of different storage mechanisms (e.g. relational databases, XML-stores, etc.) it is impossible to define a common interface for all platforms to extract data from their storage layers. In order to solve that problem, we rely on the OAI-PMH for legacy data import. Hence, institutions that already have an OAI-PMH Server built on top of their existing systems can easily export their metadata descriptions to the BRICKS Metadata Management system without any implementation effort.

3. System Design and Technology

In this section we provide an overview of the main components of our Metadata Management system and will explain how we plan to fulfil the objectives and satisfy the requirements discussed in the previous section. We have implemented the system in Java and exposed it as a Web service using Apache Axis [13]. The Semantic-Web-related components of our Metadata Management system are built on top of the Jena Semantic Web Framework [14] in combination with the open-source Pellet OWL Reasoner [15].

3.1 Overall Architecture

As depicted in Figure 1, the BRICKS Metadata Management system is composed of three principal components: the System Core, Data Transformation Layer and Storage Back-End.

The System Core provides the necessary methods for the management and retrieval of schemas and metadata records. It encapsulates four subcomponents: the Schema Manager, which handles schemas and schema mappings defined in OWL, a Validator to detect possible inconsistencies in a schema, the Metadata Manager, which provides the storage and retrieval functionalities for metadata records and a Query Adapter which serves as a common entry point for search and discovery services. The System Core's functionality is exposed via a Web Service interface that is tailored to the digital libraries domain.

The Data Transformation Layer allows the import of metadata descriptions represented in RDF/XML. As we expect datasets to be delivered in XML format, conforming to a specific XML Schema, we require a mechanism to transform XML to RDF. The results of [5] demonstrate that transformation from XML to RDF/XML using XSLT is an adequate mechanism for syntax-to-syntax translation. Since we want to relieve cultural institutions of the burden of creating transformation style sheets, we include style sheets for standardized metadata schemas, such as Dublin Core, VRA, and MARC, as standard components of the BRICKS Metadata Management System.

The Storage Back-End is managed by the Jena Framework. Its built-in storage abstraction mechanism allows the Metadata Management system to operate on large-scale database servers like Oracle, MySQL, and PostgreSQL, or on a small-scale file-based storage system.

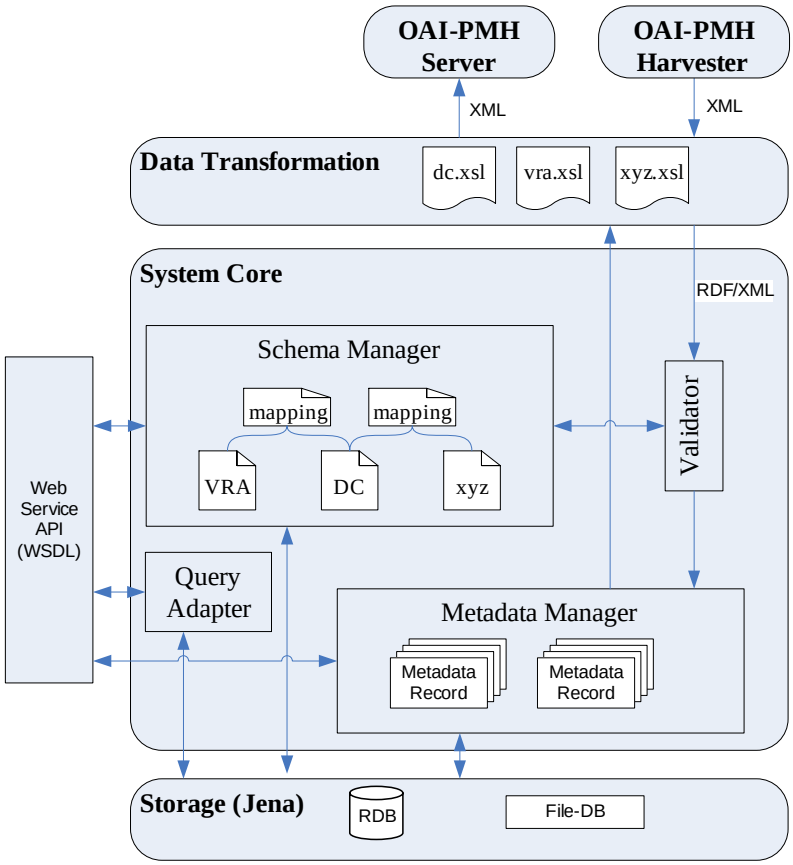

Figure 1. Overall Architecture.

3.2 Schema Management and Mapping

The BRICKS Metadata Management system accepts schema definitions in OWL-DL, which is a sublanguage of OWL. We restrict ourselves to this sublanguage of OWL because schema definitions within our system serve as input to a background inference process for which we must guarantee computational completeness. Furthermore, because we use OWL, which is a knowledge representation language, to define the structure of our metadata descriptions, we must introduce some additional restrictions, while at the same time conforming to OWL semantics. The most important restriction is the composite view of classes and their associated properties, which corresponds to the view of frame languages.

We use the mapping support provided by OWL (owl:equivalentClass and owl:equivalentProperty) and RDFS [16] (rdfs:subClassOf, rdfs:subPropertyOf) for crosswalks between standardized schemas as well as mappings from proprietary schemas. This information is exploited by the background inference process which serves as input for query resolution. In that way queries formulated according to a specific schema also return results spanning over a multitude of schemas, presumed that mappings were specified beforehand. For the definition of schemas we rely on external tools like Protégé [17], and especially the Protégé OWL Plug-in [18]. The internal Schema Manager keeps track of registered schemas and available schema mappings.

3.3 Metadata Management and Validation

We follow the Open Archives Initiative (OAI) methodology for metadata management and organize descriptive metadata as metadata records. A metadata record always describes a content item that is unambiguously identified by a Uniform Resource Identifier (URI). A

content item can have one or more metadata records, each corresponding to exactly one metadata schema. Within the BRICKS Metadata Management system, items and their metadata records are grouped into hierarchically structured repositories.

Whenever metadata records are imported into a repository, these records are validated against the corresponding schema to check restrictions imposed by the metadata schema. The metadata records, schema and the available schema mappings, also serve as input for a background inference process. This process exploits the knowledge represented in the schema definitions and in the mappings to derive additional information, which is stored in a parallel "inferred repository" and is used in turn by search and discovery services.

Since the BRICKS Metadata Management system will be accessed primarily by search and discovery services, we provide a Query Adapter to facilitate the search for content items. Its interface offers a simple object-based query language, which permits the selection of all content items within the repository that fulfil a given condition. A condition can consist of arbitrary nested conjunctions and disjunctions of selection predicates. All queries are translated into RDQL [19] and passed to the Jena Query Engine. Queries can optionally be executed on the "inferred repositories"; it is thereby possible to fully take into account the semantics of a schema definition as well as the mapping information.

The metadata-related operations provided at the Web service interface are similar to the operations dictated by the OAI-PMH standard (e.g. GetRecord, ListItems, ListRecords, ListRepositories, ListMetadataSchemas, etc.). In addition, we provide operations to create, update, and delete metadata records in the repositories.

3.4 OAI-PMH Server and Harvester

As mentioned earlier, the BRICKS Metadata Management system implements the OAI-PMH protocol to import and export metadata. The most important advantage of this approach is that the OAI-PMH client of our Metadata Management system can be configured to harvest metadata at regular intervals. This facilitates synchronization with existing content management systems. OAI-PMH also supports tracking changes since the last harvesting process – only data that have changed since then are imported. Additionally, this protocol allows the organization of metadata records into hierarchically structured "sets," which have their counterpart in the hierarchical structure of our metadata repository.

4. Case Study and Preliminary Results

We have implemented a prototypical digital library system that uses our Metadata Management system (and other components) to evaluate the applicability of our approach and to illustrate the benefits of the BRICKS architecture. The objective was to make image collections from several cultural organizations accessible via a common web interface. From a metadata management perspective, the focus of this prototype was to demonstrate that is possible to import metadata from heterogeneous sources into an RDF based metadata repository and that our component can easily be integrated with other system components.

In order to implement this scenario, we chose several collections from two cultural institutions that participate in the BRICKS project as content providers: the Austrian National Library's Pictures Archive (http://www.bildarchiv.at) and the Consorzio Forma's image collection from their project "La fortuna visiva di Pompeii" (http://pompei.sns.it/). Both institutions participating in the case study have existing content management systems and make use of metadata descriptions according to proprietary schemas. We have modelled their schemas in OWL-DL, imported them into the BRICKS Metadata Manager, and established mappings to the unqualified Dublin Core schema.

Because Consorzio Forma exposes their metadata descriptions via an OAI-PMH Server that was implemented for this case study, we were able to harvest their records in their

native format and store them in a repository. In the case of the Austrian National Library, we encountered the problem that no such mechanism was available at the time of this case study. However, the library was able to export XML documents from their relational database system underlying their image archive. After creating a suitable XSLT style sheet to transform those documents into RDF/XML, we successfully integrated their metadata descriptions into our Metadata Management system with minimal effort.

From this case study we have learned that the BRICKS methodology is suitable for developing new digital library applications within a short time by combining existing components created as part of the BRICKS infrastructure. We could easily integrate the BRICKS Metadata Management system into other newly built digital library systems following the methodology presented in previous sections.

The main results of this application scenario for the further development of the Metadata Management system are:

- Integrating legacy metadata via the OAI-PMH is suitable from a technical point of view and widely accepted approach among the content providers. The implementation of an OAI-PMH compliant server in cultural institutions not only requires minimal effort but is also useful for other reasons – e.g. to expose content to web-crawlers.
- Using RDF as the data model for repositories is suitable to tackle the problem of heterogeneous metadata. Triple stores like Jena work perfectly for datasets of small and medium-sized institutions. However, as soon as larger institutions are involved practical problems with performance, scalability, and query response time are to be expected.

5. Application Scenarios and Business Benefits

The described Metadata Management system will be made available as open-source software at the end of the BRICKS project in June 2007. BRICKS aims at building a distributed Digital Library in which each of the member institutions owns and is responsible for a node in the network. The benefits for the users of the system are:

- Small institutions that have no content / metadata management system in place can use the BRICKS system to manage their metadata (and content). Since the software infrastructure will be open-source, the costs of adopting BRICKS are minimised.
- Institutions with existing content / metadata management systems can easily integrate their systems with a node in the network. They only to implement an OAI server to make their metadata accessible. If there is already such a server in place, integrating with the BRICKS network involves zero costs from a technical point of view.
- Institutions can enhance their visibility by making their metadata (and content) accessible in a large network. The semantic interoperability between the metadata sets of different institutions will increase the probability that relevant data are found, which will also increase visibility. Institutions can also profit from synergies created by linking their metadata to those of other network members.
- Enhanced visibility can eventually lead to increased income; for example, because more people want to see the objects the institution owns, or to buy high-resolution images from the institution's repository.
- Scientists or organizers of exhibitions can use the system to find material relevant to their interests with a single query rather than having to contact and query all member institutions individually.

A common problem of many EU-funded projects is that support ends as soon as the funding terminates. BRICKS will actively avoid this by building a community of content providers and software developers. The decentralized structure of the project also contributes to its sustainability: because each member institution is responsible for its own node in the network and no central administration or maintenance is required, we hope that

the institutions will continue running the nodes beyond the life-time of the project. BRICKS nodes are designed to required low investment costs and even lower maintenance costs.

As a result of the service-orientated design approach of the BRICKS infrastructure, its components can also be used in other contexts. For example, the Metadata Management system could be easily re-used in Web Content Management systems or enterprise document management systems. Recent commercial attempts (e.g. [20, 21]) to use Semantic web technology for content management indicate that our approach is not only interesting from a scientific point of view but also from a business perspective.

6. Conclusions and Future Work

We have presented a Metadata Management system for Digital Libraries that provides the flexibility necessary to handle heterogeneous metadata descriptions. It is designed to integrate easily and inexpensively with existing content management systems. Additionally, we assure interoperability with other digital library systems by exposing all metadata records as proscribed by the OAI-PMH standard. As a proof-of-concept, we have implemented a prototype application, which demonstrates that our Metadata Management system can be easily deployed with other digital library components and that it can integrate metadata from heterogeneous sources with minimal effort.

However, experience indicates that it is unlikely that our users will provide bilateral mappings between all metadata schemas. For this reason, we will attempt to derive new mappings from existing ones using the "emergent semantics" approach [22]. As an alternative, we will investigate the use of CIDOC CRM [23] as a common semantic framework to which all proprietary metadata schemas can be mapped. Ideally, this approach would require just one mapping per metadata schema to achieve interoperability between all schemas.

For similar reasons, we must consider new, possibly semi-automatic methods to create metadata schemas in OWL-DL. Attempts to semi-automatically create such schemas from existing XML or relational schemas already exist [24, 25]. We have investigated these approaches and believe that providing such a mechanism as part of our Metadata Management system could further facilitate the integration process.

Concerning schema mapping, we have learned that the approach of using the mapping constructs provided by OWL has some shortcomings: the process of creating mappings between schemas is restricted to importing complete schemas and using the imported elements by direct reference. This is feasible for a small number of schemas but causes problems with a large quantity of complex schema definitions. In the future we intend to separate mappings from schema definitions and to design a query-rewriting mechanism that includes such mappings as input parameters.

Acknowledgments

This research was partly funded by the Austrian Federal Ministry of Economics and Labour and the European Union as part of the 6th Framework Program.

References

[1] R. Tansley, M. Bass, D. Stuve, M. Branschofsky, D. Chudnov, G. McClellan, and M. Smith, The DSpace institutional digital repository system: current functionality. In: Proceedings of the 3rd ACM/IEEE-CS joint conference on Digital Libraries, Houston, Texas, 2003, pp. 87-97

[2] T. Staples, R. Wayland, and S. Payette, The Fedora Project: An Open-source Digital Object Repository System. In: D-Lib Magazine, April 2003

[3] C. Lagoze, W. Arms, S. Gan, D. Hillmann, et. al., NSDL: Core services in the architecture of the national science digital library (NSDL). In: Proceedings of the 2nd ACM/IEEE-CS joint conference on Digital Libraries, Portland, Oregon, USA, 2002, pp. 201-209

[4] G. Amato, C. Gennaro, F. Rabitti and P. Savino, Milos: A Multimedia Content Management System for Digital Library Applications. In: Research and Advanced Technology for Digital Libraries: 8th European Conference, ECDL 2004, Bath, UK, LNCS 3232, Springer
[5] M. Butler, J. Gilbert, A. Seaborne and K. Smathers, Data conversion, extraction and linkage using XML in Project SIMILE. External technical report available at: http://www.hpl.hp.com/techreports/2004/HPL-2004-147.html
[6] BRICKS – Building Resources for Integrated Cultural Knowledge Services, EU FP6 project http://www.brickscommunity.org
[7] OWL Web Ontology Language Reference. 2004 W3C http://www.w3.org/TR/owl-ref/
[8] DCMI, "Dublin Core Metadata Element Set," DCMI Recommendation Version 1.1, Dublin Core Metadata Initiative (DCMI), July 1999, http://dublincore.org/
[9] Library of Congress – Network Developement and MARC Standards Office, http://www.loc.gov/marc/
[10] J.M. Martinez, R. Koenen, and F. Pereira, MPEG-7 – The Generic Multimedia Content Description Standard, Part 1. In: IEEE MultiMedia, vol 9, no. 2, 2002
[11] The Open Archives Initiative Protocol for Metadata Harvesting (OAI-PMH), v. 2.0. 2002-06-14, http://www.openarchives.org/OAI/openarchivesprotocol.html
[12] Web Service Description Working Group, The Web Service Description Language (WSDL) Version 2.0 Part 0: Primer, http://www.w3.org/TR/2004/WD-wsdl20-primer-20041221/
[13] Apache Axis, http://ws.apache.org/axis/
[14] J. Caroll, I. Dickson, C. Dollin, D. Reynolds, A. Seaborne, and K. Wilkinson, Jena: implementing the semantic web recommendations. In: Proceedings of the 13th World Wide Web conference on Alternate track papers & posters, New York, NY, USA, 2004, pp. 74-83
[15] Pellet OWL Reasoner, http://www.mindswap.org/2003/pellet/index.shtml
[16] RDF Vocabulary Description Language 1.0: RDF Schema, W3C Recommendation 10 February 2004, http://www.w3.org/TR/rdf-schema/
[17] Protégé Ontology Editor, http://protege.stanford.edu/
[18] H. Knublauch, R. W. Fergerson, N. F. Noy, and M. A. Musen, The Protégé OWL Plugin: An Open Developement Environment for Semantic Web Applications. In: Proceedings of the 3rd International Semantic Web Conference – ISWC 2004, Hiroshima, Japan, LNCS 3289, Springer
[19] RDQL – A Query Language for RDF, W3C Submission 9 January 2004, http://www.w3.org/Submission/2004/SUBM-RDQL-20040109/
[20] Aduna, http://aduna.biz/products/index.html
[21] IgentaConnect, http://www.ingentaconnect.com
[22] K. Aberer, P. Cudré-Maroux, and M. Hauswirth, The chatty web: emergent semantics through gossiping. In: Proceedings of the 12th international conference on World Wide Web, Budapest, Hungary, 2003, pp.197-206
[23] CIDOC Conceptual Reference Model, http://cidoc.ics.forth.gr/
[24] P. Lehti, and P. Frankhauser, XML data integration with OWL: experiences and challenges. In: Proceedings of the 2004 International Symposium on Applications and the Internet, 2004, pp. 160-167
[25] M. Ferdinand, C. Zirpins, and D. Trastour, Lifting XML Schema to OWL. In: Proceedings of the 4th International Conference on Web Engineering, ICWE 2004, Munich, Germany, pp. 354-358