Bridging repositories to form the MACE experience

Martin Wolpers
Fraunhofer FIT, Germany
martin.wolpers@fit.fraunhofer.de

Martin Memmel
DFKI GmbH, Germany
martin.memmel@dfki.de

Joris Klerkx
K.U.Leuven, Belgium
joris.klerkx@cs.kuleuven.be

Gonzalo Parra
K.U.Leuven, Belgium
gonzalo.parra@cs.kuleuven.be

Bram Vandeputte
K.U.Leuven, Belgium
bram.vandeputte@cs.kuleuven.be

Erik Duval
K.U.Leuven, Belgium
erik.duval@cs.kuleuven.be

Rafael Schirru
DFKI GmbH, Germany
rafael.schirru@dfki.de

Katja Niemann
Fraunhofer FIT, Germany
katja.niemann@fit.fraunhofer.de

September 21, 2009

Abstract

This paper reports on our experiences in bridging learning repositories on architecture within the MACE system. In order to make access to architectural learning resources simple and easy for students and teachers, we rely heavily on standards. We discuss the concepts behind the MACE system, describe related problems and how we solve them by employing and modifying standards from the computer science and architecture. By bringing together numerous heterogeneous repositories, combining and relating metadata of their learning resources and incorporating social community support, we enable open access while still respecting organizational and legislative issues of all participating repositories.

1 Introduction

Developed within the European project MACE[1] (Metadata for Architectural Contents in Europe), the MACE system and portal support open access to learning resources in the domain of architecture, specifically in higher education.

[1] http://www.mace-project.eu

The MACE system focuses on the need of architecture education to have simple and reliable access to vast amounts of information about architecture, used mainly for inspirational purposes [1, 2] and within courses.

In order to relieve students and teachers from the burden of browsing through numerous isolated repositories and dealing with redundant content, new forms of visually based discovery mechanisms for access to learning material [8] are needed. The MACE system therefore aims to extend common approaches for simple keyword search, navigation and result link presentation with advanced methods like image and location based search and classification browsing. Such advanced methods of access require rich information about the learning resources.

The MACE system bridges repositories that provide different facets and aspects of architecture learning resources in highly heterogeneous ways. Examples of such repositories are the Dynamo repository[2] providing information about architectural projects or the ICONDA[3] repository providing access to legislative documents important to building construction and design. These repositories are used to demonstrate the feasibility of the approach. As the MACE system is open, further repositories are continuously added. Details are found on the MACE homepage.

By automatically (using the real-world object approach described in 4.1) and manually (using collective intelligence and a group of architecture experts) linking related architecture learning resources of various non-related repositories with each other, we establish explicit relations among them within the MACE system and across repositories. The relations are then used to enable new, simple and unified access to architectural learning resources scattered throughout repositories world-wide. Consequently, users are provided with the ability to discover new learning resources that can serve as additional sources of inspiration.

Furthermore, the MACE system incorporates social software to draw on the capabilities of architecture communities within the MACE portal. Users are able to annotate learning resources with tags, comments and ratings, build up personal portfolios, and contribute new learning resources. In addition, user activities are analyzed to identify and explain relations between (implicitly) related learning resources.

This paper reports on our experiences in the development of the MACE system. Specifically for interoperability, we rely heavily on flexible standards as we discovered the need of their adaptation to our requirements. The following Section 2 briefly describes the resulting MACE portal. Section 3 outlines the conceptual architecture of MACE, while Section 4 describes the implementation of the MACE system, reflecting where appropriate on the problems we were facing and how we solved them using standards. Section 5 concludes the paper.

[2] http://dynamo.asro.kuleuven.be/dynamovi/
[3] http://www.iconda.org/

Figure 1: Example of facetted search.

2 Enabling the MACE portal services

This section briefly describes the MACE portal. The portal is used as a thematic access point to the various learning resources provided by the many participating repositories. As such, it features a number of search possibilities addressing the various needs of students and teachers of architecture. Apart from simple keyword search, we offer *browse by classification*, *search with the help of communities (Social Search)* and *facetted search*.

Simple Keyword Search: Using keywords and combinations thereof, the user is able to search through the learning resources within the MACE system. The keyword search uses the metadata that describes the learning resources, taking into account metadata provided by repositories as well as keywords provided by users.

Browse by Classification: Many of the learning resources included within the MACE system are classified using the MACE architectural classification system. A hyperbolic tree [6] visualization of the classification enables the user to click through the hierarchical classification until she has found the desired classification that then links to respective learning resources.

Facetted Search: Here, the user is able to qualify the keyword search with several additional facets: the repositories in which to search, the language of the results, the resource media type, the resource classification, and the associated competency. When a value is selected for a facet, the interface dynamically changes and provides the numbers of results for each facet that match the selected criteria. Figure 1 displays the faceted search for the Guggenheim Museum with respective results in MACE.

Social Search: The user is presented with the most popular tags contributed by MACE users, visualized by a tag cloud. A tag links to the respective learning resource(s). Furthermore, using the keyword search, the user can search through the contents that were already tagged by the MACE users.

The MACE portal is online and publicly accessible at `http://portal.mace-project.eu/`. The results of a search include a small overview for each result, containing information such as the resource title, a short description, a thumbnail of the resource and the repository where it was found. The user can decide to either immediately get to the result, or to view more metadata about the resource by calling up the respective MACE detail page as shown in Figure 1.

3 Conceptual architecture

This section describes the conceptual architecture. The MACE system builds on a distributed service oriented architecture with a three-tier structure. The front-end with its graphical user interfaces and widgets forms the client tier. The business logic that is responsible for the provision of functionalities is organized in the application-server tier while the metadata stores form the data-server or back-end tier.

Moreover, the MACE system incorporates both the ARIADNE[15] and the ALOE[4] systems. While the ARIADNE system is responsible for the storage of metadata and basic business logic, the ALOE system specifically deals with participation facilities for end users and the respective data that is generated. The overall architecture and functionality of the MACE system has already been described in [14] and [11], so we here focus on describing the conceptual setup rather than implementation details.

3.1 Integrating repositories

The MACE system aims to incorporate as many relevant repositories as possible. Instead of copying the content (and dealing with intellectual property rights and digital access management), we harvest the metadata from the repositories and store them in the central MACE store. Following [3], we ensure interoperability by using the OAI-PMH framework [13] harvesting standard and define the MACE application profile (MACE-AP) to describe the learning resources. The MACE-AP, based on the IEEE LTSC Learning Object Metadata (LOM) standard [10], is tailored specifically to the requirements of the architectural field. The MACE-AP is used to describe all learning resources made available through the MACE portal. Apart from ensuring a unified way of describing resources, it serves as basis for enriching incomplete metadata. For instance, if a resource is not classified in the MACE classification, an architect can use the MACE portal to classify the resource and therefore enrich the incomplete metadata. Some specific features of the MACE-AP are explained in more detail in 4.1.

Figure 2 illustrates the metadata flow from the content providers to the MACE central store. On the left side of this figure, content providers offer access to their non-enriched metadata using OAI-PMH interfaces. The ARIADNE harvester [15] pushes harvested metadata into a harvested metadata store that contains non-enriched metadata only. When metadata is enriched through the MACE portal and tools, it is pushed into the Enriched Metadata Store by using the Simple Publishing Interface (SPI). Metadata in both, the enriched and the harvested metadata stores is duplicated into the central MACE Metadata Store, so that is available to the MACE portal services, e.g. for querying. The MACE metadata infrastructure therefore preserves all harvested and enriched metadata separately, but still allows for a unified search on top of all metadata.

The MACE infrastructure is situated within the ARIADNE architecture [15], which is a standards-based architecture for managing learning objects in an open and scalable way. The architecture supports the integration of learning objects in multiple, distributed repository networks.

[4] `http://aloe-project.de`

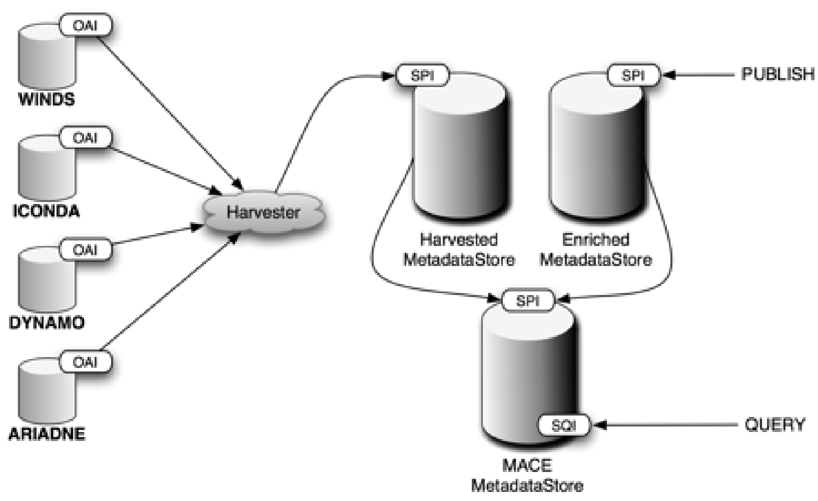

Figure 2: The MACE Metadata Store.

3.2 Social communities in MACE

MACE also aims to foster the building of a MACE community as a central part of the sustainability efforts to ensure that MACE will continue to be a lively and dynamic place that provides useful information for any user interested in architectural contents. Such communities provide different kinds of valuable information – such as tags, comments, and ratings, but also new resources as well as formal classifications for MACE resources. As such, MACE incorporates the ALOE platform to enable and provide community functionality.

The metadata and the information about the end users' relations to resources and other users allow for an improved provision of resources through social browsing and ranking methods that are based on end user contributions and usage metadata. For example, users can browse through the portfolios of other users, see which tags they already used, resources can be ordered by the number of times they have been bookmarked or viewed, etc.

The ALOE platform was developed as a white label[5] social sharing platform with the aim to offer a social-media-enabled single point of access for (heterogeneous) infrastructures with different kinds of resources and metadata. Thus, the focus was placed on the support of a maximum of resource types and metadata formats as well as the ease of integration with other applications in potentially arbitrary contexts. Furthermore, the ALOE Web Interface (which is not used in MACE) can easily be adapted to the needs of a specific scenario.

Within the MACE system, the ALOE platform is used to provide the opportunity to store user generated metadata such as tags, comments, ratings. More-

[5]Wikipedia defines a white label product or service as 'a product or service produced by one company (the producer) that other companies (the marketers) rebrand to make it appear as if they made it' [17]

over, ALOE is used to search in social metadata, to maintain profile information, to create portfolios of resources, and to maintain contact lists (”following” other users).

4 Implementing the MACE architecture

4.1 The MACE Application Profile

Section 3.1 briefly outlined why we developed the MACE application profile and how we deal with inconsistencies in learning resource descriptions by enriching them. Here, we discuss an extension to the LOM standard that enables us to describe real-world objects like buildings themselves as learning resources in MACE and how to classify them.

Representations of real-world objects (RWO) like buildings must be included in the MACE store so that we are able to interconnect related learning resources. Consequently, they are are non-information resources (following the W3C recommendation ([5]) so that an information resource is created to represent each RWO. This representation, identifying the RWO through its name and/or its location is used to bind learning resources about one specific RWO or physical manifestation together, e.g. grouping images, texts and plans about a building in one instance using the relation category, usually as part-of relation.

Representation of an RWO capture properties of the physical manifestation such as the creation date of a building (in the LOM lifecycle category) or its geographical location (in the geolocation element that we have used to extend the LOM technical category and has KML-XML as value space.) Where possible, we associate each RWO with its respective counterparts in DBPedia or Freebase to facilitate interoperability with existing and future repositories of architectural knowledge. In order to be able to distinguish between RWOs and non-RWOs, we introduce a new element into the MACE-AP.

Representations of RWOs as well as learning resources are classified with architectural classifications. Usually, one would rely on one classification but within MACE, users required the system to enable the use of a multitude of classifications. We therefore developed the MACE vocabulary as a unifying but expendable classification system that is based on a number of existing online and offline classifications used in the architectural domain, e.g. the Getty architectural thesaurus[6] and the Ci/SfB construction index[7]. Currently, the vocabulary includes more than 2.000 vocabulary terms in hierarchies with at most 5 levels with a tendency to continuously grow. Maintenance of the MACE vocabulary is carried out manually through an expert group with the help of the Protege system. Details about the management process are described in [9].

[6] $http : //www.getty.edu/research/conducting_{r}esearch/$

[7] $http : //www.ascinfo.co.uk/9/category/category13_{9}.html$

4.2 Harvesting learning resource descriptions

In order to connect content providers repositories, the harvesting infrastructure (see Section 3.1) solves a number of problems that are outlined in the following:

- **The use of proprietary metadata formats:** MACE provides support to content providers in creating a mapping from the provider metadata format to the IEEE LTSC LOM [10] standard based MACE-AP. In order to ensure that only MACE compliant metadata records arrive in the harvested metadata store, the ARIADNE metadata validation service enables the harvesting framework to check records against the MACE-AP. This service builds on XML Schema (XSD), Schematron rules and special purpose components to check for compliance [15].

  This is necessary because metadata instances that were harvested contained errors (e.g. not all mandatory fields available, empty fields, syntactic errors, etc.). Implementing the rules that are specified in the MACE-AP, the validation results are returned to the originating metadata repository to enable providers to correct the errors. Non-conforming instances are not published in the MACE repository as they can lead to unexpected behavior in the MACE portal. Furthermore, the MACE portal can be used for enriching the metadata of the architectural learning resources. Providers in turn are enabled to harvest back their enriched metadata for use in their own repository.

- **Non-interoperable, incompatible software architectures:** MACE provides a number of OAI-PMH target implementations that comply with the OAI-PMH 2.0 specification to content providers. MACE uses the ARIADNE harvester software for harvesting these OAI-PMH targets. The harvester stores the metadata records by publishing them with the Simple Publishing Interface (SPI) in the MACE Harvested Metadata Store.

- **Global unique identifiers:** MACE introduces a set of MACE unique identifiers (MUID) that are unique within the whole system. They are created for digital learning resources, users, locations, and metadata. The MUIDs use the OAI identifier schema, that consists of (i) an identifier that describes the source repository and (ii) the local identifier of the resource in the source repository. These identifiers are added to the metadata at harvesting time by the ARIADNE Harvester.

4.3 MACE community features

The MACE system has to offer community features to ensure the continuous update and use of learning resources, to add new resources, and to maintain them. The system to be chosen had to be open and easily integrateable into the MACE system. We therefore have chosen the ALOE platform that is designed as a server-based application where information is exchanged via HTTP. On the one hand, system functionalities are offered via a graphical user interface (the

AloeView) that can be accessed with any common web browser. On the other hand, a SOAP-based Web Service API is offered that allows to access the ALOE functionalities (the *AloeWebService*) and which we use to integrate ALOE in MACE.

In addition, we needed to ensure that the ALOE platform is scalable. Consequently, the ALOE architecture is based on standard technologies such as JAVA, JSP, Apache Tomcat, Axis2, MySQL, CSS and AJAX. The components ensure scalability as well as sustainability of the system.

4.4 Integrating Social Metadata in MACE.

Social Metadata is exchanged between ALOE and MACE through push/pull mechanisms in the MACE frontend and, in addition, through harvesting the ALOE store.

Data is pushed or pulled from the MACE portal and widgets. Given that a valid sessionID is provided and that the user has the right to access or contribute the data, all kinds of relevant metadata are pushed to or pulled from ALOE using the *AloeWebService* through the MACE portal. For tags, comments and ratings, the standard ALOE databases are queried. To store and access classifications by end users following the MACE classification standards, ALOE's generic classification scheme is used.

In addition, the MACE portal pushes social metadata also to the MACE usage metadata store. The usage metadata stores are integral part of the MACE search architecture. In this way, MACE offers searching over harvested, enriched and social metadata while, at the same time, keeping the types of metadata separate for reasons of data security. Furthermore, social metadata that was generated using ALOE can be harvested using the OAI-PMH target that is provided. It supports the MACE-AP and offers flow control by delivering incomplete lists and resumption tokens. Finally, the redundancy introduced by duplicating metadata hinders accidental data loss without the need to implement a complicated transaction mechanism.

4.5 Finding appropriate learning resources in MACE

4.5.1 Standards and Specifications

All domain and content metadata in the central MACE Metadata Store is stored in an XML database and can be queried through the Simple Query Interface (SQI) [12]. SQI (CWA 15454) provides interoperability between search applications and learning object repositories and is designed to support many types of search technologies. It is neutral in terms of query languages and result formats. PLQL [16] is a query language that has specifically been developed for repositories of learning objects. It consists of a number of levels where each level adds functionality. By supporting standards and specifications like SQI and PLQL, MACE can be queried in an interoperable way. For instance, MACE is searchable in other communities, such as the GLOBE consortium [4].

4.5.2 Search Performance

To increase performance of serving incoming queries, the MACE metadata store is enhanced with a Lucene index that complements the XML database. Lucene [7] has been widely recognized for its utility in the implementation of internet search engines and local, single-site searching. We chose Lucene for the following features:

- scalable, high-performance indexing with small RAM requirements.
- powerful, accurate and efficient search algorithms.

Lucene itself is just an indexing and search library and does not contain crawling and HTML parsing functionality. When metadata is inserted, the metadata is both stored in the XML database as well as parsed and transformed into a Lucene document. The Lucene index then enables direct PLQL [16] level 0 and PLQL level 1 queries so that the bottleneck of XQuery translation, which is needed to enable PLQL level 0 and level 1 on XML databases is avoided.

4.5.3 Faceted Search

Sometimes keyword searches can provide a vast amount of results which shows the need to include new ways of filtering these results in order to get more relevant content. Faceted search is a technique for accessing content by filtering available information. Some of the benefits of faceted search include:

- no hierarchy of options
- flexible classification and grouping of information

MACE provides this kind of functionality by integrating the Apache Solr [8] into the MACE infrastructure. Solr is a search server based on the Apache Lucene that extends its searching capabilities with faceted searching, hit highlighting, caching, replication and support for multiple output formats.

4.5.4 Multilingual Search

As MACE is a European project, multiple languages are used in the metadata instances. As part of the application profile, a number of multilingual vocabularies are available. MACE provides a unified, multilingual search on top of the multilingual metadata sets describing the learning resources.

4.6 User Management and Authentication

In general, the MACE system is open to all users. Nevertheless, user authentication and management is necessary to complicate/hinder misuse of the MACE system and to ensure that the community features of MACE are used by real humans in an appropriate way.

[8] http://lucene.apache.org/solr/project

When a user wants to register, an internal userID is generated by the MACE system and the ALOE system is asked to also register the user. The ALOE userID is then returned to the MACE system and stored within the user database.

The MACE portal offers users the ability to log into the system in order to be able to use the MACE community features. For this, the MACE login name and password must be entered correctly by the user. Once logged into the system, a login to ALOE is also performed, and an ALOE sessionID is returned. Using this sessionID, the user can immediately access the community features.

5 Conclusion

In this paper, we have outlined how the MACE architecture realizes flexible and open access to digital architectural learning resources provided by a large range of highly inhomogeneous repositories. In order to make this possible, the MACE architecture lays important foundations in both technical and conceptual areas, with a focus on metadata harvesting (OAI-PMH) and services for content enrichment.

In order to be able to combine learning resources across repositories, all learning resources are uniformly described with the MACE application profile of IEEE LOM. Naturally, the MACE application profile is specifically tailored to the requirements of architecture.

The integration of the ALOE social community platform enables users of the MACE system to actively discuss contents, provide metadata about contents like rating and comments and even contribute learning resources so that continuous participation is motivated and encouraged.

Moreover, the MACE portal complements the architecture with a visually based discovery system that enables teachers and students to find and hence, possibly reuse or repurpose interesting material.

At the time of writing, 100000+ relevant resources are already available in the MACE central repository: 17187 of WINDS15980 of ICONDA[9], 14824 of Dynamo[10], 9021 of Cumincad[11], 923 of Architecture.it[12] and 61928 or Archiplanet[13]. Connections that are currently being made to many other repositories like Nextroom[14], Columbus[15], Structurae[16], LightCampus[17] will bring in even more learning resources.

Through the use of standards and specifications, the MACE system is interlinked with the GLOBE consortium, an alliance of organizations that have

[9] http://www.iconda.org/
[10] http://dynamo.asro.kuleuven.be/dynamovi/
[11] http://cumincad.scix.net/
[12] http://www.architecture.it
[13] http://www.archiplanet.org/
[14] http://www.nextroom.at/
[15] http://www.columbus-portal.eu/
[16] www.structurae.de/
[17] http://lightcampus.iguzzini.com/

committed to provide ubiquitous access to quality education content. Therefore, all content providers of MACE are also able to deliver their resources at the global scale.

Finally, the overall sum of services offered by the MACE system contributes to the paradigm of open access. While the IPR rights on the contents remain with the owners, students and teachers are still able to discover learning resources, communicate and collaborate about them.

References

[1] John Beckmann. *Virtual Dimension: Architecture, Representation, and Crash Culture*. Princeton Architectural Press, 1998.

[2] M. Condotta and I. Del Ponte. Digipolazione architettonica, nuovi software convertiti. Master's thesis, Universita IUAV di Venezia, 2002.

[3] Erik Duval, Neil Smith, and Marc Van Coillie. Application profiles for learning. In *ICALT '06: Proceedings of the Sixth IEEE International Conference on Advanced Learning Technologies*, pages 242–246, Kerkrade, the Netherlands, 2006. IEEE Computer Society.

[4] Globe. The globe consortium. http://globe-info.org/globe/go, 2008. last retrieved: April 2008.

[5] I. Jacobs and N. Walsh. Architecture of the world wide web, volume one. w3c recommendation. online, 2004.

[6] Jonh Lamping and Ramana Rao. The hyperbolic browser: A focus+context technique for visualizing large hierarchies. *Journal of Visual Languages & Computing*, 7(1):33–55, 1996.

[7] Lucene. Lucene homepage. http://lucene.apache.org/, March 2008. last retrieved: April 2008.

[8] Gary Marchionini. Exploratory search: from finding to understanding. *Commun. ACM*, 49(4):41–46, April 2006.

[9] Katja Niemann and Martin Wolpers. Modeling vocabularies in the architectural domain. In *ICDIM*, pages 314–319. IEEE, 2008.

[10] Institute of Electrical and Electronics Engineers Learning Technology Standards Committee. Ieee standard for learning object metadata. ieee standard 1484.12.1, 2002.

[11] Christian Prause, Stefaan Ternier, Tim de Jong, Stefan Apelt, Marius Scholten, Martin Wolpers, Markus Eisenhauer, Bram Vandeputte, Marcus Specht, and Erik Duval. Unifying learning object repositories in mace. In David Massart, Jean-Noel Colin, and Frans Van Assche, editors, *LODE*, volume 311 of *CEUR Workshop Proceedings*. CEUR-WS.org, 2007.

[12] Bernd Simon, David Massart, Frans van Assche, Stefaan Ternier, Erik Duval, Stefan Brantner, Daniel Olmedilla, and Zoltan Miklos. A Simple Query Interface for Interoperable Learning Repositories. In *Proceedings of the 1st Workshop on Interoperability of Web-based Educational Systems*, pages 11–18, 2005.

[13] H. Van De Sompel, M. L. Nelson, C. Lagoze, and S. Warner. Resource harvesting within the oai-pmh framework. *D-Lib Magazine*, vol. 10(nb. 12), December 2004.

[14] Moritz Stefaner, Elisa Dalla Vecchia, Massimiliano Condotta, Martin Wolpers, Marcus Specht, Stefan Apelt, and Erik Duval. Mace - enriching architectural learning objects for experience multiplication. In Erik Duval, Ralf Klamma, and Martin Wolpers, editors, *EC-TEL*, volume 4753 of *Lecture Notes in Computer Science*, pages 322–336. Springer, 2007.

[15] S. Ternier, K. Verbert, G. Parra, B. Vandeputte, K. Klerkx, E. Duval, V. Ordonez, and X. Ochoa. The ariadne infrastructure for managing and storing metadata. *IEEE Internet Computing*, 13(4):pp. 18–25, July/August 2009.

[16] Stefaan Ternier, David Massart, Alessandro Campi, Sam Guinea, Stefano Ceri, and Erik Duval. Interoperability for searching learning object repositories, the prolearn query language. *D-Lib Magazine*, 14(1/2), January/February 2008.

[17] Wikipedia. White-label product — wikipedia, the free encyclopedia, 2009. Electronic document. Retrieved May 25, 2009, from `http://en.wikipedia.org/w/index.php?title=White-label_product&oldid=289642140`.