LUISA

Learning Content Management System Using Innovative Semantic Web Services Architecture

IST- FP6 - 027149

Deliverable D4.3

LOMR architectural prototype specification

Miguel-Angel Sicilia
Salvador Sanchez
Sinuhé Arroyo
Sergio Martín-Cantero

Due date of deliverable: 28/02/2007

Actual submission date: 28/02/2007

Start date of the project: 01/03/2006 Duration: 30 Months

Miguel-Angel Sicilia
University of Alcalá

Version 1.0, dated 20/01/2007

Change History

Version	Date	Status	Author (Partner)	Description
0.1	10.01.2007	Initial Draft	Miguel-Angel Sicilia	First draft
1.0	1.03.2007	First version	Miguel-Angel Sicilia	First version; after including reviewer's suggestions.

EXECUTIVE SUMMARY

This deliverable describes the specification of the architectural prototype of the LUISA LOMR, which represents a major milestone towards the final reference implementation. The main use cases to be addressed in the architectural prototype are identified, following the philosophy of exercising a part of each of the main functionalities of the final software product. The architecture of the prototype is sketched, as a result of the consideration of the main architectural issues identified so far in the project.

This document serves as the blueprint for a functional prototype that exercises key functionalities of the LUISA LOMR requirement, which serves as an early assessment of the adequacy of the technical solution for the LOMR.

Document Information

IST Project Number	FP6 – 027149	**Acronym**	LUISA
Full title	Learning Content Management System Using Innovative Semantic Web Services Architecture		
Project URL	http://www.luisa-project.eu /		
Document URL			
EU Project officer	Kypros Kyprianou		

Deliverable	**Number**	4.2	**Title**	LOMR architectural prototype specification	
Work package	**Number**	4	**Title**	Learning Object Metadata Repositories and LOMR Integration Development	

Date of delivery	**Contractual**	28.02.07	**Actual**	28.02.07
Status	Version 1.0, dated 20/01/2007		final ☑	
Nature	Report ☑ Demonstrator ☐ Other ☐			
Dissemination Level	Public ☑ Consortium ☐			
Abstract (for dissemination)	This deliverable describes the rationale, scope and main issues addressed in the LUISA LOMR architectural prototype.			
Keywords	Learning objects, learning object repositories, Semantic Web, Semantic Web Services, ontologies, WSMO, IRS-III			

Authors (Partner)	Miguel-Angel Sicilia (UAH), Salvador Sánchez-Alonso (UAH), Sinuhé Arroyo (UAH), Sergio Martín-Cantero (UAH)			
Responsible Author	Miguel Angel Sicilia		**Email**	msicilia@uah.es
	Partner	UAH	**Phone**	+34 91 886 6603

Project Consortium Information

Partner	Acronym	Contact
Atos Origin S.A.E. (Coordinator)	ATOS	Nuria de Lama Atos Origin S.A.E. c/ Albasanz 12 E-28037 Madrid, Spain Email: nuria.delama@atosorigin.com Tel.: +34 91 214 9321 Fax:+34 91 754 3252
University of Alcalá de Henares	UAH	Dr. Miguel-Angel Sicilia Information Research Unit University of Alcalá Ctra. De Barcelona, Km 33.6 E-28871Alcalá de Henares (Madrid), Spain Email: msicilia@uah.es Tel.: +34 91 886 6603 Fax: +34 91 885 6646
University of Uppsala	ULL	Dr. Ambjorn Naeve University of Uppsala Kyrkogårdsgatan 2 C Uppsala Email: amb@nada.kth.se Fax: +46 184-716-294
Open University	OU	Dr. John Domingue Knowledge Media Institute, The Open University, Walton Hall, Milton Keynes, MK7 6AA, United Kingdom Email: j.b.domingue@open.ac.uk Tel.: +44 1908 655014 Fax: +44 1908-653-169
University Henri Poincaré	UHP	Dr. Monique Grandbastien University Henri Poincaré Vandoeuvre les Nancy 54506, PO Box 239, France. Email: monique.grandbastien@loria.fr Fax: +33 383-278-319
Giunti Interactive Labs S.r.l.	GIUNTI	Dr. Fabrizio Giorgini Giunti Interactive Labs S.r.l. Abbazia dell'Annunziata Via Portobello Baia del Silenzio 16039 Sestri Levante (GE), Italy Tel.: +39.0185.42123 Fax: +39.0185.43347
EADS – Corporate Research Centre		Anne Monceaux EADS – Corporate Research Centre Avenue Didier Daurat - Centreda 1,Toulouse, 31700, France. Email: anne.monceaux@airbus.com Tel.: +33 561-184-725 Fax: +33 561-187-611

Table of Contents

1 INTRODUCTION

The mission of LUISA is that of exploiting the advantages of a Semantic Web Service Architecture to make richer and more flexible the processes of query and specification of learning needs in the context of Learning Management Systems and Learning Object Repositories.

This document describes the outcomes of the initial work on task 4.3 labelled "*Architectural prototype development*". The intention of the work reported here is that of devising an architectural prototype that exercises the main functionalities that are required in LUISA LOMR (Learning Object Metadata Repository) implementations. These include the storage of learning object metadata in semantic form, the provision of a service-oriented interface and the import of data in non-semantic form, among others. The main issues that have been identified are discussed here and a blueprint architecture is reported, with the required flexibility to evolve towards the final architecture of the LUISA LOMR reference implementation.

The document is structured as follows. Section 2 provides brief background information on the architecture of LUISA. Then, Section 3 states the objectives and functionality that will be addressed in the architectural prototype. Section 4 lists the major architectural issues discovered so far, along with potential solutions for them. Section 5 describes the architecture of the prototype and some advance on frameworks that will potentially be used. Finally, Section 6 provides an outlook to additional issues.

2 Background: Overall Architecture of LUISA and LUISA LOMR

This section provides a brief synthesis of the state of the overall LUISA architecture as the big picture of the whole project in which the LOMR architecture discussed plays a role.

2.1 Overall LUISA Architecture

The main components of the LUISA framework architecture are depicted in Figure 1.

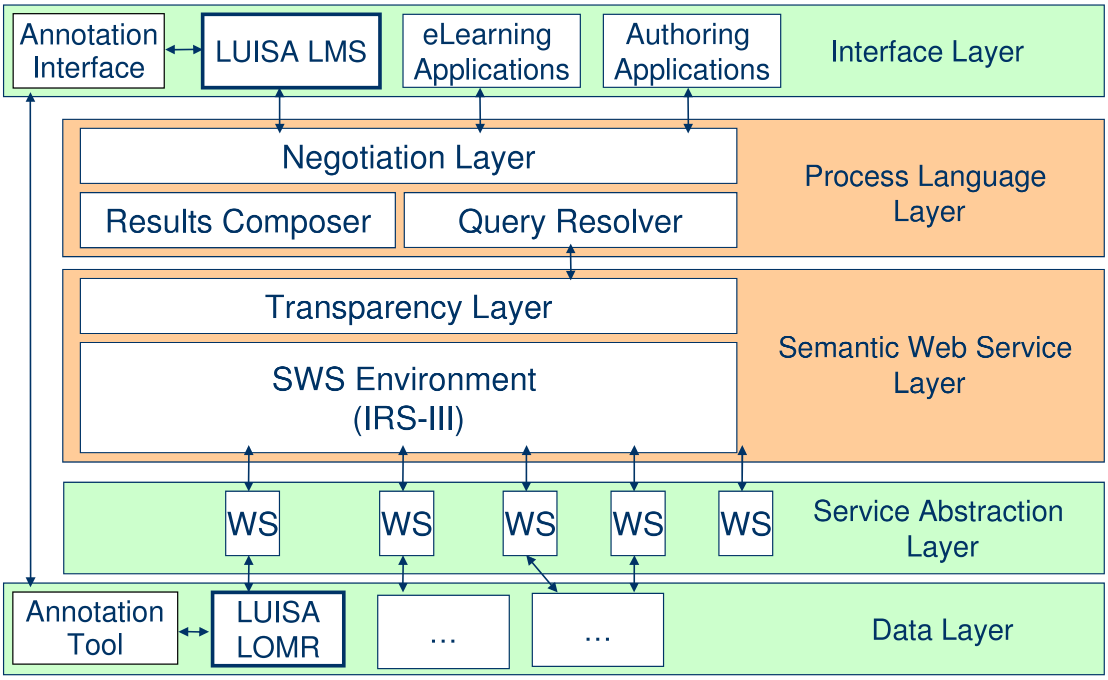

Figure 1. Main components of the LUISA Architecture.

The architectural prototype specification described here deals with the specific of LUISA LOMR repositories, labelled as *Data Layer* in the picture. LUISA LOMR instances provide a Web Service interface to the *Semantic Web Service Layer*. This enables the use of SWS technology to select the best repository for a given learning need that arises at the interface layer. It also enables that specialized components as custom *Query Resolver*s and *Result Composer*s benefit from the availability of different, heterogeneous LOMR instances. Further, LUISA LOMR instances can be accessed through the *Annotation Tool* for the edition of metadata in semantic form.

2.2 LUISA LOMR Architecture

Deliverable 4.1. "*LOMR overall architecture*" provided the following Architectural instantiation as the target for the LOMR reference implementation. Figure 2 depicts the initial architectural ideas that were reported in that document.

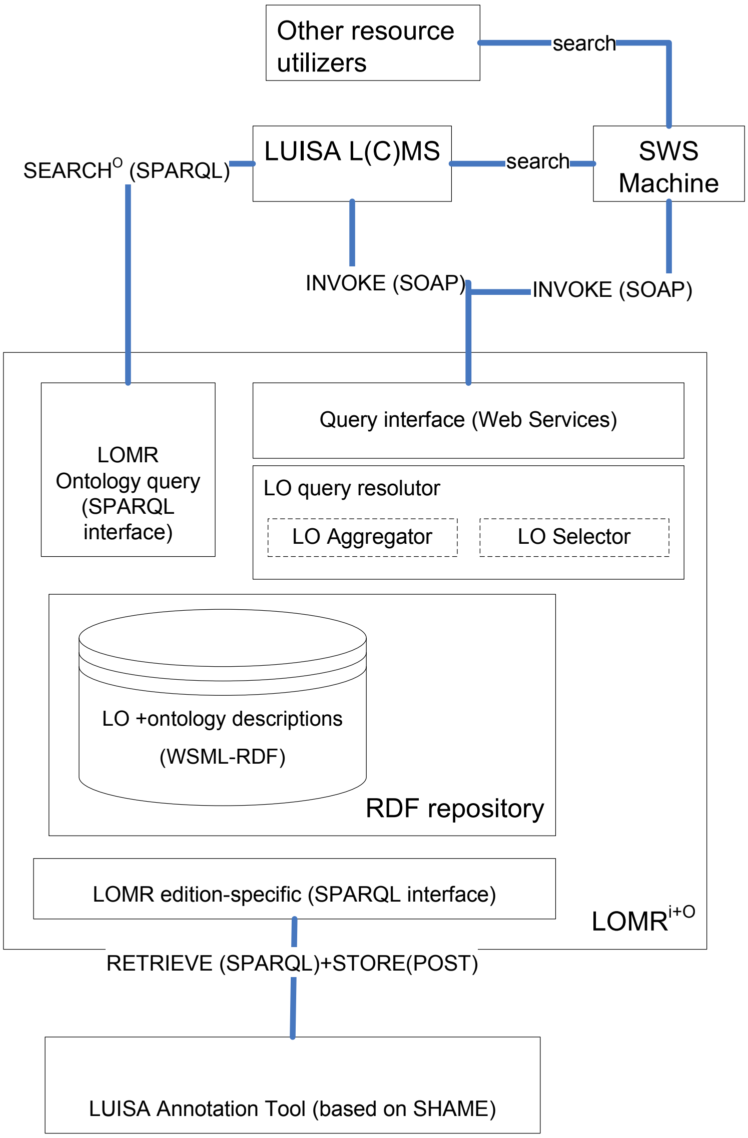

Figure 2. LOMR overall architecture (as in deliverable 4.1 version 1.0)

In Figure 2, several elements were identified that are detailed in this document:

- The RDF-based storage of ontology instances, notably including instances of learning object metadata records.
- Interfaces with the LUISA *Annotation Tool*.
- Query interfaces for the ontology in itself and specifics for the search of learning objects, the latter being realized in a service-oriented style.

The architectural prototype details that architecture into a more detailed layered view which considers specific libraries, frameworks and technologies. Some elements have been refined, especially according to the ongoing implementation of the architectural prototype. LUISA-compliant systems (as the LCMS in Figure 2) might use directly the services of LUISA LOMR instantiations or alternatively use the *Semantic Web Service Layer* (labelled *SWS Machine*) to take advantage of brokering capabilities, which could query the appropriate repositories for each given need, in a distributed setting. Such relations have been re-defined and explained in more detail in this deliverable and D2.1.

3 Rationale for the Architectural Prototype

This section describes the motivation for the architectural prototypes and the target functionality addressed.

3.1 Main objectives

Software architecture can be defined in broad terms as "*the software components, their external properties, and their relationships with one another*." The SWEBOK guide[1] adopts this definition, which considers architecture as a blueprint "*a description of the subsystems and components of a software system and the relationships between them*".

One of the most important deliverables of the process of Software Engineering is the executable architecture baseline (Jacobson, Booch and Rumbaugh, 2000). An approach to engineering such baseline is creating and early functional **architectural prototype.** The Open Process Framework (OPF)[2] defines architectural prototype as:

> *the application architecture work product that models a partial application that verifies the software architecture of an application*

The main objective of the architectural prototype is thus to **early shape the main architectural aspects of the LUISA LOMR reference implementation**.

Following the philosophy of OPF definitions, the concrete objectives of the LUISA LOMR Architectural prototype are:

1. To assess the application software architecture.
2. To provide an evolvable baseline that will end up in the software components of the reference implementation.
3. To signify completion of the Initiation phase.

The benefits of this approach are the following:

- Defects in the software architecture can be identified early in the development process when they can be fixed with less cost and less effort.
- The architecture prototype marks a major milestone in the evolution of the deliverable implementation.

The concrete architectural issues to be assessed in the prototype (that arose in LUISA technical meetings) are described in Section 4 of this document.

3.2 Use cases to be addressed

The LOMR is intended to be a framework rather than an end-used application for a particular domain. As a software framework, it will provide a number of

[1] http://www.swebok.org

[2] http://www.opfro.org/

services that would eventually be used in the implementation of learning object metadata repositories in different organizations and/or domains. In consequence, the use cases to be addressed are generic functionality, not tied to a particular domain.

This is the list of cases that will be addressed in the prototype:

I. Storage of learning object metadata in semantic form, using conventional data stores (e.g. relational DBMSs).

II. Provision of query services that allow for different query languages, from existing IEEE LOM based approaches to specific queries that exploit the ontologies stored as part of the semantic metadata.

III. Provision of a generic, service-oriented interface to the metadata, which will be used by other components of LUISA.

IV. Storage of composite learning objects, which may reference metadata in other semantic repositories.

V. Integration of existing metadata in other repositories.

VI. Integration with the architectures developed in the rest of LUISA work packages.

It should be noted that case V is described in depth in the separate deliverable "D4.2 *LOMR integration specification*".

4 Issues to be Addressed in the Prototype

This section describes the main architectural issues that are to be addressed in the architectural prototype, along with technology(-ies) and a tentative approach for each of them. This list comes from early coding/testing work that has already been done in WP4.

4.1 Persistence mechanism

The core element of the architectural prototype is the storage of semantic learning object metadata and the provision of retrieval and update interfaces.

4.1.1 Requirements and constraints

The core of LUISA technology is the exploitation of Semantic Web Services (SWS), and more concretely of the WSMO framework[3]. WSMO relies on WSML, a language that extends existing description logic languages in several directions. An obvious consequence is that there is a need to support WSML persistence.

Requirement #1 The LOMR should provide storage and retrieval facilities for WSML entities.

4.1.2 Proposed approach

WSML can be mapped to RDF through an existing specification (de Bruin, 2006). Further, the ORDI libraries[4] yet provide a persistence mechanism for WSML in combination with the WSML parsing and serializing libraries WSMO4J[5]. In consequence, using ORDI as the underlying persistence mechanism appears as the natural candidate.

The main problem on using ORDI is that it currently supports only Sesame[6] as the underlying triple store, while the repository used for the initial work on integrating with the annotation tool (WP3) used HP Jena[7]. However, ORDII features an extensible architecture that allows for using other triple repositories.

The approach for the prototype is that of coding a Jena wrapper for ORDI, enhancing additionally the storage options for the final implementation. The benefit of this approach is that of reusing existing investment in WSML persistence, and avoiding the manipulation of semantic metadata at the level of RDF triples.

[3] http://www.wsmo.org

[4] http://www.ontotext.com/ordi/index.html

[5] http://wsmo4j.sourceforge.net/

[6] http://www.openrdf.org/

[7] http://www.hpl.hp.com/semweb/

4.2 Integration with WP3 outcomes

The *Annotation Tool* based on ULL SHAME tool is a RDF-based flexible implementation of a generic editor. The Annotation Tool is capable of annotating all kinds of learning objects included in LUISA LOMR. It will be easily adaptable according to specific annotation profiles, and it will be embeddable into existing Learning Content Management Systems.

In consequence, there is a need to provide a RDF interface to support the requirements posed by the *Annotation Tool*.

Requirement #2 The LOMR should provide access at the triple level to deal with RDF processing.

4.2.1 Proposed approach

The use of ORDI with Jena is compatible with interfacing with the annotation editor. This has already been tested as part of the work of WP3. As part of the work in the prototype, the use of distribution frameworks for Jena repositories (e.g. Joseki) will be tested.

4.3 Service-oriented interface

As depicted in the overall LUISA architecture (Figure 1), the search for LO in LOMR must provide a service-oriented interface.

4.3.1 Requirements and constraints

The main requirement is the provision of a standard, Web Service interface to other parts of the LUISA architecture, at least for the querying functions.

Requirement #3 The LOMR should expose the LO querying capabilities through a Web Service interface.

4.3.2 Proposed approach

The early LUISA prototype (October 2006) yet included the implementation of a small number of Web Services that accessed WSML data stored in Jena. The architectural prototype will provide a re-factoring of those (or other similar) capabilities to adapt them to the interfaces and layers of the prototype architecture described below.

4.4 Query languages

One of the main principles of the overall design of the LOMR was that of providing a flexible architecture for different query styles (*query resolutors*)

4.4.1 Requirements and constraints

The LORI architecture[8] defines a framework and a simple protocol for interaction between learning object repositories. One of its main features is that

[8] This initiative is an open, collaborative effort, under the auspices of the CEN/ISSS Learning Technologies Workshop, to achieve interoperability between learning object repositories.

of separating the basic mechanism of query from the actual languages used for expressing the queries.

Requirement #4 The LOMR should provide room for different query styles.

4.4.2 Proposed approach

The architectural prototype will feature the design for different resolutors and at least a simple *query resolutor* based on IEEE LOM-compliant metadata. This includes the use of different kinds of queries, as prescribed by LORI. The intended approach is that of sketching a "query-by-example" approach.

4.5 Learning object metadata record identification

Even though IEEE LOM provides identification metadata for LO, the possibility of composing LO into aggregates raises the need for a LUISA-specific identification schema.

4.5.1 Requirements and constraints

Early work on the design of composition (aggregation) of LO in the LUISA architecture has raised the need for an identification schema with specific features.

Requirement #5 LUISA LOMR metadata record must feature globally unique identifiers, which encapsulate the information on the access point for querying about the metadata.

4.5.2 Proposed approach

The proposed approach is that of providing a naming mechanism following the schema:

```
luisa://<repository-URI>?=id=<local-unique-lo-metadata-record-id>
```

Once a reference of that kina is obtained, the application can access the URI of the repository to retrieve more metadata regarding the LO.

4.6 Integration with existing sources

The details of the integration are provided in the separate deliverable 4.2.

5 ARCHITECTURE OF THE PROTOTYPE

This section describes the resulting LOMR prototype architecture.

5.1 Overall architecture

Figure 3 depicts the main components of the LUISA LOMR architectural prototype. The components that will be (partially) implemented as part of WP4 are depicted with pointed lines. Additional LUISA software components are depicted in shaded (yellow) colour outside the main boxes, and with a reference to the WP(s) in which the bulk of the work is carried out.

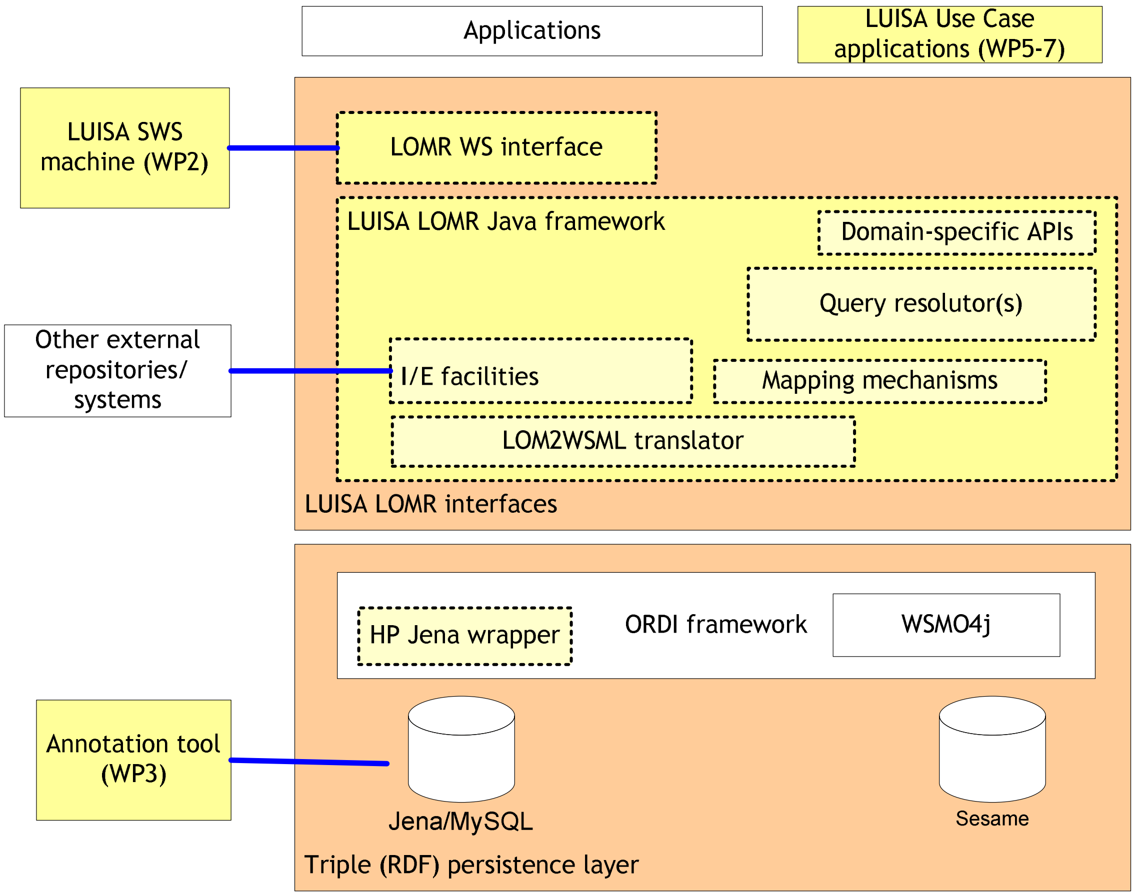

Figure 3. Main elements of the architectural prototype

The main architecture of the LOMR is divided in two layers:

- Triple (RDF) persistence layer.
- LUISA LOMR interfaces

The *Triple persistence layer* abstracts applications and LOMR components from the RDF storage of WSML data. Eventually, applications might access the data at the RDF level, either directly or through some kind of adapter. Work in that direction for the prototype is intended to include assessment of existing distributed triple querying interfaces.

The *LUISA LOMR interfaces* feature the interfaces to applications and the core components that will realize queries and composition. The following is a brief description of the elements that will be included in the prototype:

- The LOM2WSML translator libraries (described in a separate document) provide the required translation functionality from IEEE LOM-compliant metadata to the ontology representation in WSML. The early implementation of these libraries uses the *LOM for Java* libraries for the parsing of XML LOM records.
- Import facilities are based on the translation libraries. Specific import cases will include metadata from the ARIADNE[9] and CAREO[10] repositories.
- The mapping mechanisms feature the translation of non-IEEE LOM compliant metadata (which is covered by the LOM2WSML translator libraries) in the form expressed by domain-specific libraries to the WSML representation.
- Query resolutors (including eventually specific composition algorithms) are the central component for advanced retrieval capabilities. The design will allow for plug-ability of these resolutors, since these components encapsulate the multiple possible approaches to find the "best" learning objects for some given learning needs or for some given learning design assumptions (Sicilia, 2006). It should be noted that this component works within the scope of a single LOMR, while a similar component family in the overall LUISA architecture deals with several distributed heterogeneous sources. That is, composition in LUISA can be done *locally* (inside a single LOMR) or in a *distributed* way (carried out by components of the *Semantic Web Service Layer*). Both cases of composition are complementary, since they deal with different kinds of knowledge. In addition, they are transparent to each other.
- Domain-specific APIs represent the view for different domain ontologies or special-purpose ontologies that can be combined with the core learning object ontologies for the purpose of semantic search for a given application or domain. These include the competency ontology (described in a separate document) delivered as part of LUISA work.
- The LOMR WS interface is the decoupled interface to other parts of the LUISA selection and composition framework, and also to other applications or systems.

An important issue in the architecture is that the LOMR provides a Web Service interface for applications and the rest of the LUISA components. A part of this interface will be used by the *Semantic Web Service Layer*, in which the semantic annotations – according to the WSMO framework – will live. This interface is the core interfacing element in LUISA, but the implementation of the

[9] http:// www.**ariadne**-eu.org/

[10] http://**careo**.netera.ca/

LUISA LOMR will provide a supplementary Java API that could be used by some applications to get data from the repository. However, this other API are a convenience for future applications and evolution and not the core interfacing element of LUISA.

The next section provides additional details on the design of some of the components for the architectural prototype.

5.2 Selected architectural styles for the implementation

5.2.1 Division in specialized APIs

The LOMR features a number of core ontologies that define the basic structure, descriptions and relationships of learning objects. The prototype includes the LOM2WSML ontology as the most basic core ontology.

However, different applications or domains will require additional support for domain ontologies (e.g. a repository specialized in Art and Architecture) or special-purpose ontologies (e.g. one expressing learning needs and capabilities in terms of competencies). The LOMR stores these ontologies as the schema for the instances, even though the ontologies (but not the instances) are also available in the *Semantic Web Service Layer*. The rationale for storing the ontologies also is that the LOMR could be used as an independent entity with semantic capabilities, which require the use of the concept schemas in addition to the instances.

The principle of separation of concerns will be exercised in the design of the prototype by providing decoupled implementations for the core part and at least one specialized part – the competency ontology. In consequence, it could be possible to use the LOMR only to store competencies. This is intended to provide a flexible architecture for other future domains or specific usages.

5.2.2 Preparation for LORI compatibility

The accommodation of a LORI SQI[11] wrapper requires a number of elements to be taken into account, including session management and different query types. These will be reflected in the design of the prototype to plan for compatibility with LORI SQI implementations.

[11] http://www.prolearn-project.org/**lori**

6 OUTLOOK

The architecture described in this document will be realized in the LUISA LOMR architectural prototype, intended to become a first, evolvable version of the final reference architecture.

The prototype will be subject to a process of assessment with sample learning object metadata and specific cases that exercise the main aspects described in this document. The outcomes of the assessment will shape the final architecture.

REFERENCES

[1] de Bruin, J. (ed.), D32v0.1 *RDF Representation of WSML*. WSML Final Draft 20 December 2006, available at http://www.wsmo.org/TR/d32/v0.1/

[2] Jacobson, I, Booch, G. and Rumbaugh, J. (1999). *The Unified Software Development Process*, Addison-Wesley.

[3] Sicilia, M.A. (2006). Semantic learning designs: recording assumptions and guidelines. *British Journal of Educational Technology*, 37(3), pp. 331-350