On Integrating Semantic Learning Object Repositories in the Open Knowledge Initiative (OKI) Architecture

Salvador Sánchez-Alonso, Daniel Rodriguez, Alberto Abián, Sinuhé Arroyo, Miguel Ángel Sicilia

Information Engineering Research Unit, Computer Science Department University of Alcalá

Keywords: *e-learning, learning object repositories, semantic Web, OKI*

Abstract:

Learning Object Metadata Repositories (LOMRs) represent a key component in the architecture of e-learning systems, and open architectures as the Open Knowledge Initiative (OKI) already provide generic interfaces for them. Semantic LOMRs are a class of repositories in which metadata is expressed in reference to or as part of formal ontologies, aimed at providing advanced search capabilities. This paper describes an approach to adapting semantic LOMR interfaces to the generic OKI repository ones retaining compatibility with the standard but providing hooks to deal with advanced semantics if required. Since there are no standards specific to semantic LOMR, the adaptation of the interfaces of a concrete open source semantic LOMR are discussed as example.

1 Introduction

A learning object repository is a software system aimed at storing educational resources and/or metadata for those resources, which provides search interfaces either to humans, to other software systems or both. In most cases, learning object repositories are not repositories of learning objects in a strict sense but repositories of information on those learning objects. In fact, these systems often store metadata about the learning objects but not the resources themselves, thus becoming learning object metadata repositories (LOMRs). A specific form of learning object metadata repositories is called semantic learning object metadata repository (Soto, García & Sánchez-Alonso, 2007) (Rodriguez, Sicilia & Arroyo, 2006) (Ihsan et al., 2006), which can be defined as a LOMR that makes use of formal representations of knowledge, in the form of ontologies, to provide enhanced search and retrieval mechanisms to its users according to the kind of services envisioned in the so-called Semantic Web model (Berners Lee, Hendler & Lassila, 2001).

Learning technology is in a continuous process of increased standardization and different consortia push to produce specifications that enable higher levels of interoperability. In the area of component interoperability, the Open Knowledge Initiative[1] (OKI) provides a broad reference architecture and normalized interface definitions aimed at enhancing the plug-ability of learning technology systems. The OKI software architecture applies the concepts of separation, hiding, and layering towards the goal of interoperability and easy integration in order to pull the common elements out of a given problem, leaving the remaining portions more tractable. The OKI initiative has grown and evolved to become an important learning technology integration standard and it is being implemented on top of relevant systems such as Moodle and Sakai.

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

Learning object repositories in OKI are represented basically by two *Open Service Interface Definitions* (OSIDs) called `Repository` and `RepositoryManager`. These provide a high-level, generic interface abstracting the basic functionality of repositories and repository federations, respectively. These interfaces have been implemented on existing repositories but as of today they have not been used to adapt a semantic LOMR. The main problem with such adaptation is that there is a sort of "semantic impedance mismatch" since OSID definitions are general-purpose interfaces, and the nature of ontology-based applications is out of their current scope. Such mismatch is materialized in issues as how to deal with ontology-based types, and how to pass references to ontology terms and instances through these interfaces.

This paper reports on the preliminary design and a basic prototype of the implementation of a OKI `Repository` interface on top of a semantic LOMR, the rest of it having the following structure. Section 2 describes relevant background on `ont-space`, an ontology framework for semantic learning repositories. Section 3 describes implantation details and details the OI scenarios of interaction. Finally, section 4 concludes the paper and outlooks future work.

2 Background on ont-space

The `ont-space`[2] is a software framework for the deployment of semantic LOMRs. It is based on the LOMR developed as part of the LUISA project[3] which provides interfaces to any given, "real" learning object repository. LOMR instances allow developers to select the best repository implementation for a given application need, enabling specialized components, such as custom query resolvers and result composers, to benefit from the availability of different, heterogeneous LOMR instances. LOMR main features include the storage of learning object metadata in semantic format, the provision of a service-oriented interface and the import of metadata in non-semantic formats, among others. Other existing repositories provide learning object metadata in non semantic languages, as for example, the XML mapping of the IEEE LOM standard (LTSC 2002a). The idea of translation is that of enriching such kind of information to enable computational semantics. In LOMR, translation facilities to bridge the gap between the non-semantic and the ontology-based representations are provided. Essentially, translators convert metadata expressed in plain XML formats to representations in which learning objects are explicit instances of a *LearningObject* concept, and the descriptions thus become attributes – or in general terms – statements about the resource. The construction of translators with a degree of flexibility in their design enables adapting existing and future non-semantic repositories in which new non-semantic descriptions can be added incrementally to semantic representations inside the LOMR.

Translation libraries ease the task of importing metadata to the LOMR. Once such translation libraries are available, metadata can be harvested through standard mechanisms such as OAI-PHM[4], for the information in those metadata to be later included in the LOMR simply by passing metadata fragments to the libraries. This affects the integration of metadata from external sources; however, direct use of the LOMR is another area in which open specifications play a role.
UAH is involved in project SUMA[5], a large industry-lead project funded by the Spanish Ministry of Industry that aims at developing OKI wrappers for learning technology

[2] https://sourceforge.net/projects/ont-space

[3] http://www.luisa-project.eu/

[4] http://www.openarchives.org/OAI/openarchivesprotocol.html

[5] https://eduforge.org/projects/suma/

components that can be integrated in an organizational context. The results of LUISA are planned to be tested against the results of SUMA. This will serve as a demonstration on the integration of semantic sources into conventional, non-semantic but standardized technology.

3 Providing OKI interfaces to a LOMR

The Open Knowledge Initiative (OKI) develops and promotes specifications that describe how the components of a software environment communicate with each other and with other enterprise systems. To this end, OKI has developed and published the *Open Service Interface Definitions* (OSIDs). The OKI OSID interface specifications include two relevant entities of direct interest to our aims:

1. `OSID:Repository`
2. `OSID:RepositoryManager`

The first one is of interest to the LOMR to be able to be included in any OKI-compliant configuration. The `OSID:Repository` interface provides operation for different tasks including search but also creation and management of `Assets`. `Assets` are contents, metadata or combinations of both that can be given different types with different structures. The second one could be use to wrap distributed brokering for the practicalities of search in federated repositories.

3.1 General semantic search approach

The OKI specifications version 2.0, called OSID, provide a variety of interfaces which comply with functions of repository, federation of repositories and digital resources. In these interfaces, there are generic calls for resolving distributed searches of resources.

Possible extensions for wrapping semantic search with OKI can be classified into the following categories:

1. Integration of semantic repositories as LUISA repositories, enabling full compatibility with the architecture.
2. Integration of semantic federation criteria, using ontologies to select the most suitable repository among potential federated repositories for a particular need for learning, with full compatibility with LUISA standard architecture.
3. Extension of LUISA-OSID interface capabilities to comply with searching cases based on explicit semantic criteria.

Extensions (1) and (2) are basically adaptations of semantic elements to standard OSID architecture. Extension (3) implies the interface enlargement to include ontologies and semantic in them. This extension can not be implemented as OKI-compliant because it is needed consensus and acceptance by OKI initiative. Figure 1 summarizes key specification elements for extensions (1) and (2). Figure 1 shows that any element connected with OKI technology (*service consumer*) will access to semantic repositories using the standard interfaces `Repository` and `RepositoryManager`. Communication between semantic brokers and repositories can be done using OSID interfaces or any other kind of technology (e.g. WSMO-based software as used in LUISA). It is important to mention that an OKI repository may store only metadata, so search results can be metadata fragments that enable the location of resources, just as it is done in the LOMR.

Figure 1. Approximation to support semantic search using OSID.

3.2 Definition of scenarios

For the sake of illustration, we will describe a few concrete scenarios as examples of using OKI interfaces with semantic learning object repositories.

3.2.1 Scenario 1: Non-semantic search on semantic repositories

This scenario is used as a semantic repositories integration test since it causes they has the behaviour of a conventional repository. Figure 2 shows the interchange of information for the case. An OKI client accesses to the semantic repository by means of the respective OSID interface using `searchCriteria` as a keyword list. The semantic repository returns the `Asset`(s) corresponding to the resources which are compatible with the keywords.

Figure 2. Non-semantic search on semantic repositories

Figure 2 shows also how OKI client gets assets, which have only metadata, to obtain from them the URI to access the content. In this scenario, semantic repositories act as mere search mediators.

3.2.2 Scenario 2: Semantic query

Figure 3 shows essentials of a OKI-compliant semantic query interface. Available ontologies across the Web will be taken into account, and OKI clients can take elements such as concepts, instances or properties from them to compose searches. In a general case, these ontologies would be available either through a generic ontology repository or through a unique, stable URI.

Every possible combination will be defined as a `SearchType` according to OKI, which allows a great deal of flexibility. The delivery of these elements and a certain interpretation will allow the semantic repository to use different strategies of query, exploiting the knowledge represented in the ontologies. Results would be delivered according to scenario 1.

Figure 3. Semantic Query

3.2.3 Scenario 3: Semantic federation of repositories

The semantic natural extension of the `OSID:RepositoryManager` element is a semantic broker of repositories. This type of brokers substitutes the individual repositories in the semantic search scenario. They also are used to decide what repositories must be invoked according to their semantic descriptions. These descriptions may include information about supported standards, type of the stored resources, information about stored resource learning approach, etc.

Figure 4. Brokered semantic query

3.2.4 Scenario 4: Competency-guided searching

Searching by competencies is a specific case of semantic searching. However, it is an important scenario to take into account since a good number of learning resources are formulated as competencies or competency elements. Since positions and academic degrees can be expressed in terms of competencies, they can be used as a shared framework to integrate learning needs at semantic level. This in turn enables the development of different applications related to human resources and training scheduling.

4 Main implementation issues

As stated previously, the OSID (the OKI specifications version 2.0) provides a variety of interfaces which comply with functions of repository, federation of repositories and digital resources. These interfaces provide generic calls for resolving distributed searches of resources. Table 1 summarises important elements of those interfaces providing semantic search capabilities.

Interface	Operation	Additional requirements
`RepositoryManager`	**`createRepository`**`( String displayName, String description, Type repositoryType)`	- Types can include semantic repositories
	`getAssetsBySearch``( Repository[] repositories, Serializable searchCriteria, Type searchType, Properties searchProperties)`	- Semantic search can be included as a special type of search. - The query distribution in a set of repositories can follow a semantic criterion.
`Repository`	**`getAssetsBySearch`**`( Serializable searchCriteria, Type searchType, Properties searchProperties)`	- As in `RepositoryManager` but locally.
`Asset`	**`addAsset`**`(Id assetId)`	- It allows the composition of digital resources.
`RecordStructure`	**`getFormat`**`()` **`getSchema`**`()`	- Formats and schemes can be used to define semantic correspondences of educational resources descriptions.

Table 1. Relevant OSID 2.0 elements for searching learning resources from repositories

An important additional element of the OKI approach is that the repository information must be stored as an Asset of a special type. As stated by OSID. In fact, there may be information about a Repository or its contents that is not suitable for representation through the description. One strategy is to make that information into an `Asset` of a specific `AssetType`. This information can then be easy retrieved and presented. Some examples of summary data are the number of `Assets` in the Repository or the usage rules. Another example is that in place of inspecting the Repository for the `RecordStructures` contained, that information could be maintained in this special `Asset`.

4.1 Searching languages in OSID repository

The searching interface in OSID repositories is performed with the following operation:

```
getAssetsBySearch(Serializable searchCriteria,
```

```
org.osid.shared.Type searchType,
org.osid.shared.Properties searchProperties)
```

The only requirement of the searching criteria (`searchCriteria`) is to be `Serializable`, i.e., this parameter can contain any type of information as Java objects. For example, in the OSID implementation of the P2P repository *lionshare*[6] ver. 1.2, `searchCriteria` is a `String` to be passed to a `QueryRequest` (as part of *gnutella*[7] API). The search is based only on *keywords*.

The implementation criteria can be summarised as follows. The `searchCriteria` object will include one or several shared ontologies with their fully qualified URL (referencing a concrete RDF element). The information about the meaning of those elements is determined by the `searchType` parameter, which in turn, will be identifiable as an ontology element of searching types. The parameters will be used according to the specifications of the such ontology of types of searching. An example of this type of implementation will be composed of the following elements:

- A list of URIs to ontology elements, for example, to concepts in an ontology about human diseases.
- `searchType` identified as `[ns]#NonSpecificSemanticSearch`, where `[ns]` will be the name space of the ontology with the types of query.
- Properties will be empty.

In the case of searching for competencies we would have the following:

- A list of URIs that refer to definition instances of competencies and/or their components.
- `searchType` identified as `[ns]#CompetencyBasedSearch.`
- In the properties, there will be references to types of searching modes. For example, when searching for competencies it is possible to specify queries with or without compensation criteria. In the former case, if there are no resources with a given competency, a similar one could be considered.

4.2 Design pattern: Shared searching types

According to OSID 2.0 specification, search types (`searchType`) are specific to each repository, but it is no specified how to specify that a type of search is shared by more than a repository.

To increase the level of interoperability, it is possible to declare search types as shared OWL[8] resources on the Web, which include the required `searchProperties`, as well as possible relationships with other types of search[9].

The ontology of types of queries will include *machine-understandable* information for the processing of the results or for sending search elements.

[6] http://lionshare.psu.edu/

[7] http://www.gnutella.com/

[8] Other formats could be used but OWL is more flexible in the definitions.

[9] A Basic ontology will be maintained at:
http://www.cc.uah.es/ie/ont/repository-search-types/current/repository-search-types.owl

4.3 Semantic Metadata Recovery using the `Asset` Interface

Often applications require metadata for some of the learning resources, either to be shown to the users or for further processing. The structure of OSID interfaces is composed of the association of one or several `RecordStructure` to each `AssetType`. This allows retrieving different types of *asset*s as the result of semantic searches. The following design criterion allow us to recover arbitrary metadata from a `Asset` which represents the metadata of a resource in a semantic repository.

The asset types that semantic repositories will return must support a `RecordStructure` to recover subject-predicate-object triples, the foundation of both RDF and ontology languages. In this way, there will be two part structures, predicate and object, the former a property in a shared ontology and the later a URI (which will be in turn a reference to a property in a shared ontology) or a literal. The mechanism described is actually recovering URIs from a shared ontology (with the exception of *datatype* properties, connected to literals). Clients can return to shared ontologies to continue with the processing, thus exploring the representations contained in those ontologies.

5 Conclusions and future work

Learning object metadata repositories (LOMRs) provide learning resource management capabilities that are central to the architecture of e-learning systems. OKI interfaces can be used as an interoperable framework to plug any existing repository with other learning technology components; however some issues that deserve special attention arise when the search and metadata management is in semantic form.
This paper has discussed the high-level design of an OKI interface for a semantic LOMR and identified main elements. The approach respects OKI OSIDs but adds some conventions that can be included in different implementations of semantic repository adapters.
Future work will address the full implementation of the OKI interface devised herein and will report on the tests carried out as part of the SUMA project, were the OSID interfaces for LOMR will be used in conjunction with other OKI components. As reported, SUMA work package 4.3 will be crucial to assess on the integration of semantic sources into conventional, non-semantic but standardized technology.

Acknowledgements

The research reported herein is part of the activities of the EU-funded project Organic.Edunet (ECP-2006-EDU-410012), from which it receives partial funding, and also by the Spanish Ministerio de Industria, Comercio y Turismo as part of the activities of SUMA: elearning multimodal y adaptativo (TSI-020301-2008-9). This work reuses outcomes of LUISA, an EU-funded project (IST- FP6 - 027149).

References

[1] Berners-Lee T., Hendler J., and Lassila O. (2001). The Semantic Web. *Scientific American*, 284(5):34-43, 2001.
[2] Ihsan, I., Mobin-uddin, A., Mohib-ur, R., Abdul Qadir, M. and Iftikhar, N. (2006) Semantically Meaningful Unit – SMU; An Openly Reusable Learning Object for UREKA

Learning-Object Taxonomy & Repository Architecture – ULTRA. In Proceedings of the 4th ACS/IEEE International Conference on Computer Systems and Applications.
[3] Learning Technology Standards Committee of the IEEE (2002a), Draft Standard for Learning Technology—Extensible Markup Language (XML) Schema Definition Language Binding for Learning Object Metadata, IEEE P1484.12.3, Draft 8 http://ltsc.ieee.org/wg12/files/IEEE_1484_12_03_d8_submitted.pdf
[4] Learning Technology Standards Committee of the IEEE (2002b), Draft Standard for Learning Objects Metadata IEEE P1484.12.1/D6.4 12. June 2002, http://ltsc.ieee.org/wg12/20020612-Final-LOM-Draft.html
[5] Rodriguez, E., Sicilia, M.A. and Arroyo, S. (2006) Bridging the semantic gap in standards-based learning object repositories. In *Proceedings of the Workshop on Learning Object Repositories as Digital Libraries: Current challenges*, part of the 10th European Conference on Digital Libraries, Alicante, Spain.
[6] Soto, J., García, E. and Sánchez-Alonso, S. (2007). Semantic learning object repositories. *International Journal of Continuing Engineering Education and Life-Long Learning*, 17(6), pp. 432-446.

Author(s):

Dr. Salvador Sánchez-Alonso
University of Alcala, Computer science department, Information Engineering research unit
Edificio Politecnico, Ctra. de Barcelona Km. 33,6 – 28871, Alcala de Henares (Spain)
salvador.sanchez@uah.es

Dr. Daniel Rodriguez
University of Alcala, Computer science department, Information Engineering research unit
Edificio Politecnico, Ctra. de Barcelona Km. 33,6 – 28871, Alcala de Henares (Spain)
daniel.rodriguezg@uah.es

Alberto Abián
University of Alcala, Computer science department, Information Engineering research unit
Edificio Politecnico, Ctra. de Barcelona Km. 33,6 – 28871, Alcala de Henares (Spain)
alberto.abian@uah.es

Sinuhe Arroyo
University of Alcala, Computer science department, Information Engineering research unit
Edificio Politecnico, Ctra. de Barcelona Km. 33,6 – 28871, Alcala de Henares (Spain)
sinuhe.arroyo@alu.uah.es

Dr. Miguel A. Sicilia
University of Alcala, Computer science department, Information Engineering research unit
Edificio Politecnico, Ctra. de Barcelona Km. 33,6 – 28871, Alcala de Henares (Spain)
msicilia@uah.es