Closing the Interoperability Gap: Connecting Open Service Interfaces with Digital Repository Interoperability

Marek Hatala[1], Griff Richards[1], Scott Thorne[2], Jeff Merriman[2]
[1]School of Interactive Arts and Technology
Simon Fraser University
Surrey, BC, Canada, V3T 2W1
{mhatala, griff}@sfu.ca

[2]Open Knowledge Initiative
Massachusetts Institute of Technology
Boston, MA
{thorne, merriman}@mit.edu

Abstract: Interoperability between e-learning systems and repositories is one of the hottest topics in e-learning community. With an availability of standards and specification for the single learning objects, courses and related learning artifacts the technical focus of the e-learning community has shifted towards the interoperability of between different learning systems and learning systems and other sources of digital objects such as digital libraries. In this paper we start with a review of the interoperability initiatives. Next, we describe eduSource's ECL and OKI's OSIDs: two approaches to interoperability and highlight their strengths and how they complement each other. Finally, we describe our present effort in merging the two approaches together with first results and observations from the implementation of the ECL/OKI connector within the scope of LionShare project.

Introduction

The e-learning community has seen fruitful initiatives in the standardization of learning object metadata by IEEE and the emergence of specifications towards the standardization of other aspects of learning objects and learning processes by organizations such as IMS and ADL. More recently, the e-learning community has been focusing on the ability to connect and use resources located in distributed and heterogeneous repositories. Interoperability between e-learning systems and repositories is one of the hottest topics in e-learning community. With an availability of standards and specification for the single learning objects, courses and related learning artifacts the technical focus of the e-learning community has shifted towards the interoperability of between different learning systems and learning systems and other sources of digital objects such as digital libraries.

Interoperability is not easy. There are many choices, including service definitions, data vocabularies, network protocols and programming language bindings that must all come together to provide various levels of interoperability. There does not exist, and likely never will, a single technology that offers the holy-grail of interoperability in all of its dimensions.

These are statements that many information technology leaders do not wish to hear. We would rather be wooed by the promise of some magic solution that will once and for all solve the entire interoperability problem in all of its various complexities. This hope is partly fueled by the marketing arms of companies whose business it is to sell software or systems and find profit in playing upon the misguided dream of magical solutions.

But more so, it is fueled by confusion around what interoperability is. We need to better define what we mean when we talk about interoperability. For instance, information exchange at the inter-enterprise level brings with it a set of requirements that differ from those associated with integration between systems within an enterprise, and differ greatly from those associated with tool portability and modularity required of applications within a system (the kind of interoperability that the Open Knowledge Initiative is most directly addressing). In some cases, network protocols provide the appropriate answers, in others API definitions are required, and some dimensions of interoperability simply require that a common data description is identified.

Web Services and its associated technologies, SOAP, WSDL, UDDI, provide a new suite of tools for helping to solve important interoperability problems. These new tools bring a certain level of hopefulness to the landscape due to the fact that most, if not all, of the vendors in the information technology marketplace have agreed to support them.

Unfortunately, however, there is significant confusion about the appropriate roles of network protocols such as the Web Services protocols, and implementation independent Application Programming Interfaces (APIs), such as those that the Open Knowledge Initiative is defining. As a community of specifications and standards designers it is all too easy to enter into black and white, either-or debate around these two interoperability elements. Such debate only serves to distract us from solving the problems at hand. To move beyond these issues we must begin to realize that both interfaces and protocols are critical elements of complex systems and when used together they have the potential of solving the basic problems posed by our various interoperability goals.

This paper presents an important stepping stone towards a more complete interoperability solution. In the next section we provide an overview of major initiatives and projects that aim to address the interoperability issues. Next we describe two approaches developed by teams of the authors: a. OKI’s Open Service Interface Definition, and b. the eduSource Communication Layer (ECL). The complimentary nature of these two approaches resulted in the joint effort to connect these two approaches that we describe in some detail, after which we conclude with a brief description of future work.

Major Interoperability Initiatives in E-Learning

OAI. Although not specifically oriented on education, the Open Archive Initiative [7] develops and promotes interoperability standards for content dissemination. The Protocol for Metadata Harvesting (PMH) developed by OAI provides an application-independent interoperability framework for metadata harvesting. The protocol enables repositories (called harvesters) to selectively harvest metadata from other sources (providers) and create cumulative and/or specialized collections of metadata. In addition to the protocol, OAI provides guidelines and community support. The protocol is used widely used by other initiatives to support harvesting functionality.

NSDL. The National Science Digital Library project (www.nsdl.org) is a major project funded by the National Science Foundation with the goal of building a digital library for education in science, mathematics, engineering and technology. The potential collections for inclusion in NSDL have a wide variety of data types, metadata standards, protocols, authentication schemes, and business models [1]. The aim of the NSDL interoperability is to build coherent services for users from technically different components. NSDL aims to support three levels of interoperability:

- **federation** implements the strong standards approach with libraries agreeing to use specific standards.
- **harvesting** allows higher autonomy. The only requirement is to enable a limited set of services via a simple exchange mechanism. NSDL is using Protocol for Metadata Harvesting (PMH) developed by the Open Archive Initiative. Harvesting is supported on the repository side by implementing a relatively simple wrapper communicated via PMH and providing metadata based on Dublin Core.
- **gathering** uses the web crawler technique to collect information from organizations that do not formally participate in the NSDL program.

NSDL has selected eight preferred metadata element sets for metadata storage. Preferably, libraries should store the metadata in their original format but they also have to be able to serve the metadata in Dublin Core (www.dublincore.org) format. Effectively this solution establishes Dublin Core as the lowest common denominator for the NSDL.

IMS DRI. The IMS Digital Repository Interoperability Group, in its specifications for the digital repository interoperability [4], provided a functional architecture and reference model for repository interoperability. Aiming at very broad application of the specification the DRI document makes recommendations only to a certain level and leaves the resolution of more operational issues to the system implementers. Five basic functions defined by IMS DRI are: search/expose, gather/expose, submit/store, request/deliver, and alert/expose. For the search function, the specification recommends using XQuery (www.w3c.org/XML/Query) with SOAP protocol or Z39.50. For the gather function, the OAI's harvesting protocol is recommended. No recommendation is made for other functions in the current version of the specification. The current version of IMS DRI envisions but does not explicitly deal with heterogeneity of the repositories and it is up to the implementers to ensure format compatibility. The DRI Group recommends development of "search intermediaries" that will deal with multiple formats.

POOL. The POOL project ran from 1999 to 2002. One of its major goals was to build an infrastructure for connecting heterogeneous repositories into one network [2]. The infrastructure used a peer-to-peer model in which nodes could be either individual repositories (called SPLASH) or community or enterprise repositories (PONDs). PONDs were connected to the POOL network using a specialized peer performing the functions of both a gateway and wrapper. The POOL network used the JXTA peering protocol (www.jxta.org) and followed the CanCore/IMS metadata profile/specification (www.cancore.org) to exchange metadata. Connected PONDs communicated using wrappers either via HTTP and CGI or XML-RPC protocol. The wrapper also performs the metadata schema translation functions that are needed. The network supported a high autonomy for the repositories, but this required creating a specialized wrapper translating between the metadata schemas and communication protocols.

ELENA/Edutella. This collaborative European project is creating Smart Spaces for Learning [6]. Smart learning spaces are defined as educational service mediators, which allow the consumption of heterogeneous learning services via assessment tools, learning management systems, educational (meta) repositories and live delivery systems such as video conferencing systems. ELENA builds a dynamic learner profile which is used as a basis for offering the learner with the choice among a variety of knowledge sources. ELENA forms a layer on top of a learning management network built on Edutella [8]. Edutella is an RDF based peer-to-peer (P2P) infrastructure that aims to connect highly heterogeneous educational peers with different types of repositories, query languages and different kinds of metadata schemata.

eduSourceCanada. The eduSource project (www.edusource.ca) brought together major Canadian LOR players to create an open infrastructure for linking interoperable LORs. The infrastructure will support a wide range of services and promises both ease of connecting and ease of using new and existing systems. For example, a repository using PMH protocol and Dublin Core metadata can either communicate with the eduSource network as a whole via a gateway mechanism or can become a participant with access to wider range of services via the "semantic cobblestone" interoperability connector.

OKI. The Open Knowledge Initiative [5] builds an open and extensible architecture that specifies how the components of an educational software environment communicate with each other and with other enterprise systems. The OKI provides service specific APIs called Open Service Interface Definitions (OSIDs) that fosters effective application development for higher education by providing definitions for educational services as well as services that are common across application domains. OSIDs cover a wide range of learning services from generic ones such as Authentication and Digital Repository to services specific for education such as Course Management and Grading. Currently OSIDs have a few test implementations but has promising support from both the academic and industrial communities.

EduSource Communication Layer (ECL)

EduSource is a broad network as it aims at the wide spectrum of services it wants to support. One of the major goals of eduSource is to create an open network for users, organizations and service provides. EduSource identified three major pieces to support openness of the network: i) ready to use tools, repositories, and services, ii) clearly defined protocol, and iii) connecting middleware for existing systems. Another important requirement for eduSource to become an open network is that it has to build its protocol on existing standards and recommendations. EduSource defines its eduSource Communication Layer (ECL) as an implementation of the IMS DRI specification. However, IMS DRI recommendation is not specific enough for direct implementation and the current penetration of recommended technologies is not as widespread as assumed in the IMS DRI specification.

Table 1 eduSource services

ECL Service	Description
Expose	When asynchronous messaging is required, this service will be called by service providers to return the responses for search, gather, and alert.
Gather	Repositories wanting to provide gather service must implement gather service handler.
Gateway Gather	A type of gather service. It allows EduSource members to make gather requests to repositories external to eduSource. By default, the results are converted to IEEE LOM format or alternatively they remain in the format used in external repository Gateway gather service providers must specify a repository type and protocol (e.g. OAI).
Search	As recommended by IMS DRI, ECL protocol uses XQuery. To enable connection of the repositories that do not support (full) XQuery a set of XQuery templates is used. The repositories register their Search Service with an indication of supported templates or full XQuery search capability.
Alert	IMS DRI recommends Alert for push gather.Whenever repository has new metadata matching subscribe parameters, it sends an alert message to the subscribers.
Submit	It is a function for moving an object (metadata and learning object) to a repository.
Store	When asynchronous messaging is required, this service is called by service providers to return the results of submit function.
Request	It provides a function to ask to deliver objects to a client. The transfer protocol could be a successive SOAP request to download the object or FTP transfer protocol
Deliver	When asynchronous messaging is required, this service will be called by service providers to return the chunked of results.

The criteria for the protocol and its development process that affected our approach included:

- eduSource is a heterogeneous network consisting of existing and future institution repositories, peer-to-peer network, individual small repositories, and application interfaces.
- ECL will be evolving over time of the project which makes all the parallel activities vulnerable to changes in the protocol.
- ECL supports many new services non-existing in the current systems. Some of these services require asynchronous communication, such as search through a peer-to-peer network or alert.
- ECL is a complex protocol. To achieve its fast and easy adoption it has to be supported with pre-configured middleware.
- A solution for connection between eduSource and other initiatives has to be easy to maintain and easy update if there is a change in the protocol used by the other initiative.

ECL closely follows IMS DRI specification and uses SOAP as a communication layer. IMS DRI core functions are defined and implemented as eduSource services (Table 1). Repositories or tools connected to the eduSource network can implement some of these services and register them in an eduSource maintained registry (such as UDDI). Registration is a preferred way for discoverability of permanent services. However, in many cases user tools connected to the network will not register any service[1]. For more details on ECL design and implementation see [3].

ECL Connector and ECL Gateway

ECL is a complex protocol with communication patterns that may be challenging to implement. To lower the technical barriers for service providers to join the network it was essential to have a solution that made the ECL easy to implement.

Since the complexity of the ECL protocol might be detrimental to its adoption, we provide an eduSource Connector (Figure 1) which implements the ECL protocol. The connector provides a standard API to connect an existing repository to the eduSource network. The ECL protocol requires institutional repositories or tools to implement connector handlers only for those services they want to expose to others - this is a far simpler than implementing and deploying every service in every institution. The connector also facilitates version synchronization during the protocol evolution. Changes in the protocol itself rarely propagate to the API level. In most cases, repositories do not have to worry about the change in the protocol, they only need to update the connector with a never version.

[1] For example, a search application does not provide any services on its own but needs to implement 'deliver' service for asynchronous search results.

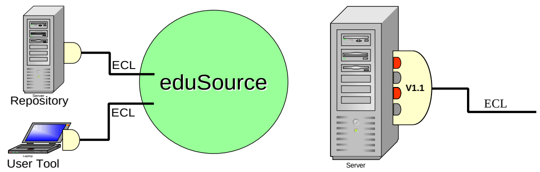

Figure 1. EduSource Connector

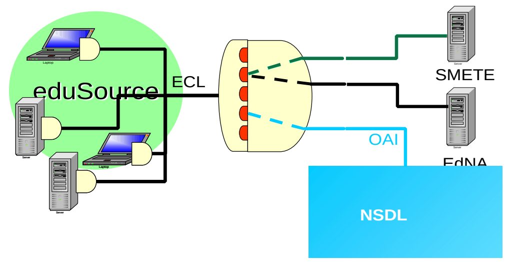

Figure 2. EduSource Gateway

Although the eduSource internal protocol provides a flexible and efficient solution it is unlikely that well established repositories and initiatives will invest resources and convert their protocols to the ECL. However, an ability of the eduSource project to connect to other established protocols and major initiatives is of the utmost importance to the project participants. EduSource addresses the problem of outside interoperability by providing a second type of mediator simply called the eduSource gateway. The eduSource gateway is modeled after the design pattern of an adapter functioning at the network level. The main function of the gateway is to mediate between ECL and communication protocols used by the outside systems.

Figure 2 shows a schema of the eduSource gateway. One side of the gateway is formed by the ECL connector. The other side of the gateway provides a framework for defining a chain of handlers that perform a conversion between ECL protocol and the protocol of the external network. The gateway framework enables us to define the mapping between protocols at four levels:

L1. Communication protocol (HTTP, SOAP, XML-RPC, Peer-to-peer, etc.)
L2. Communication language (ECL, OAI, POOL, etc)
L3. Metadata (IMS, CanCore, Dublin core)
L4. Ontologies (vocabularies for metadata)

The eduSource Gateway typically runs on a dedicated computer and provides services for all participants in the eduSource network. The main benefit of placing the mapping functionality for an outside network onto a gateway instead of with each participant is that it can be easily updated if the change in the outside network protocol occurs. In such a case, a chain of mapping handlers is updated at the one place and all eduSource participants can continue to communicate with the gateway using ECL protocol without any change necessary at the individual sites. There can be several gateways for the same outside network if the traffic between the two networks is high. One gateway can provide services for several other networks.

EduSource Network Development Status

The eduSource network currently connects repositories of the eduSource project partners: Explora repository at TéléUniversité in Montreal, CAREO repository in Calgary, Pond repository at SFU and LO repository at Athabasca University. EdNA, SMETE and OAI based repositories are connected to the eduSource network via gateways. Several tools provide access to the eduSource network: Splash is a P2P repository that also provides eduSource federated search and submit functionality, a web-based federated search is available from www.edusplash.net, and Explora tools can be used by Explora's user community. To illustrate the development effort required to connect a tool/repository to the network, we organized a workshop for eduSource partners and found that implementation of the search functionality on each repository site required approximately 4 hours of work and implementing the submit functionality approximately 1 developer-day of work.

OKI Open Service Interface Definition (OSID)

The Open Knowledge Initiative [5] builds an open and extensible architecture that specifies how the components of an educational software environment communicate with each other and with other enterprise systems. The OKI provides a service specific API called Open Service Interface Definition (OSIDs) that fosters an effective application development for higher education by providing definitions for data and common services. OSIDs cover whole range

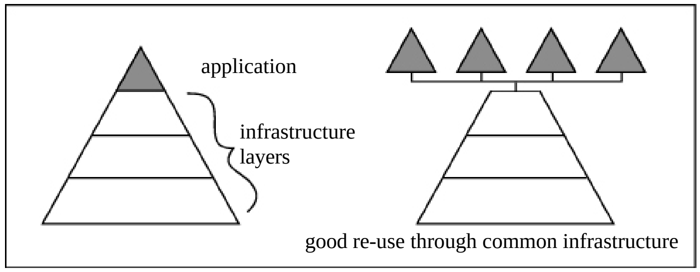

Figure 3 Interoperability through shared infrastructure

of services from generic ones such as Authentication and Digital Repository to services specific for the education such as Course Management and Grading. Currently several institutions are building OSID test implementations and the specification has promising support both from the academic and industrial community.

The OSIDs, taken together, establishes a comprehensive array of services upon which application can be built. It employs an 'iceberg' approach to application development where good reuse is enabled by writing OSID implementations once and re-used and distributed across many projects (Figure 3).

OKI supports interoperability between clients and servers by using a plug-in architecture (Figure 4). The plug-in implements the standard OSID that enables the application to communicate with the server application either on the local system or with a remote system via a specific protocol. To enable an application to use a new service a plug-in implementing the OSID needs to be developed and deployed at the client using one of the standard methods for updating client software. As an API is not specific about the format of the data passed between the plug-in and the application, different plug-ins need to be developed for different clients or clients need to be modified to be able to consume data passed by the plug-in.

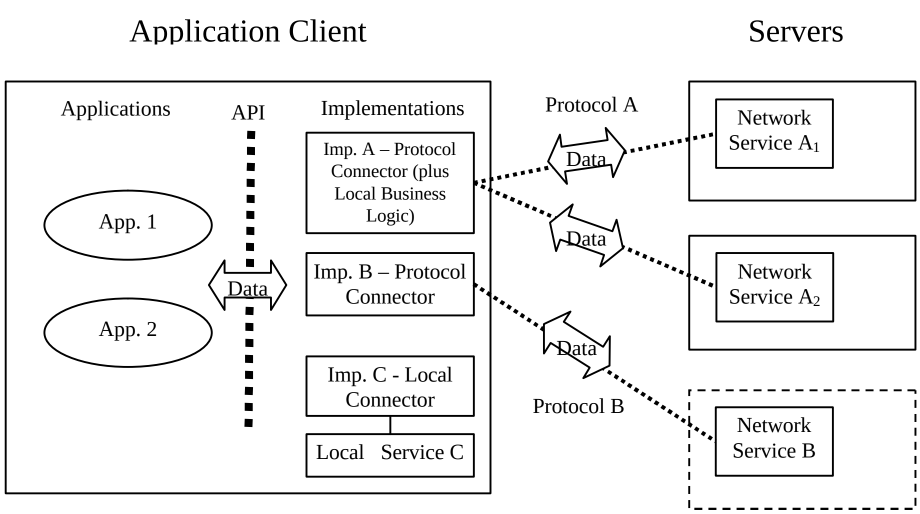

Figure 4 OKI plug-in architecture using OSID

From the perspective of a programmer connecting two systems, the OKI provides the following benefits: it provides a clear specification, removes architectural detail, isolates the implementation, minimizes coupling between modules, maximizes cohesion, and encourages writing for reuse.

OKI/ECL Plugin

The two approaches presented above address interoperability at different levels. While OKI addresses internal

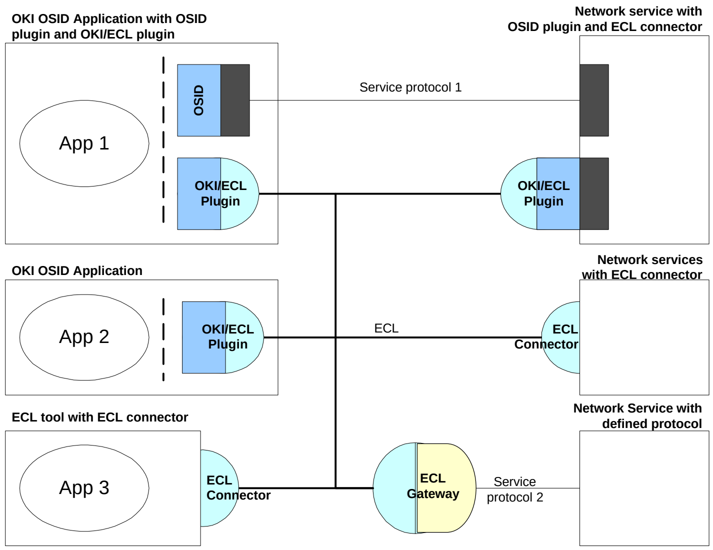

Figure 5 Interoperability supported by OSID and ECL with the use of ECL connector and OKI/ECL plug-in

interoperability of the learning system by specifying OSIDs, the ECL defines the protocol and provides a middleware supporting main interoperability functions between repositories. Merging these two approaches is a natural step that brings new qualities into the interoperability solution. The OKI/ECL connector is a middleware component currently being developed by Simon Fraser University Surrey team and MIT's OKI group within the LionShare project.

Effective implementation of the OKI/ECL plug-in will be required to resolve following issues:

1. Mapping between DR OSID functions and ECL API functions,
2. Expressing DR OSID searchCriteria in the form of the XQuery for the ECL Search function,
3. Specifying how metadata records retrieved via ECL protocol map to the DR OSID Asset objects.

The power of clearly specified API's in OSID and the ECL connector was demonstrated in the speed of the implementation of the first prototype of OKI/ECL plug-in. Two developers working independently for most of the time were able to demonstrate a working demo of DRQuickSearch from the OKI test implementation after only 3 hours of work. DRQuickSearch, a search tool that utilizes the DR OSID API, was originally searching Fedora repository using the Fedora plug-in and now is able to show both the results from the Fedora and multiple ECL repositories.

The OKI/ECL plug-in brings interoperability to systems originally designed with very different perspectives and expands the available interoperability solutions for the following types of systems (Figure 5):

1. Applications implementing the digital repository OSID API can now use the OKI/ECL plug-in to communicate using ECL protocol. This provides them with an access to both ECL enabled repositories and networks and repositories connected through ECL gateway mechanism.
2. Repositories implementing the digital repository OSID API can use the OKI/ECL plug-in to make themselves available to the eduSource tools and repositories.
3. Existing repositories that were not designed with interoperability in mind can implement the ECL connector directly.
4. For existing repositories with a well-established remote protocol, an ECL gateway can be configured to connect them to the eduSource network via the ECL gateway mechanism.
5. New E-learning tools now have an option of implementing OSID layer abstraction and using the OKI/ECL plug-in or implementing an ECL connector directly.

Future Work

There are several issues that need to be addressed to complete our implementation of the OKI/ECL plug-in. We are particularly looking at the ways of representing specific metadata and content object on the OSID structures and hope to finalize the development of the OKI/ECL plug-in in January 2004. This will provides us with a deeper understanding of the issues when connecting two approaches to interoperability and we will be able to formulate guidelines for future adopters. OKI specification and test implementations are available from http://sourceforge.net/projects/okiproject and ECL protocol specification, documentation and ECL connector code are available from http://www.edusplash.net/technical/ecl/index.html.

The work described here complements our earlier work in eduSource and OKI initiatives and is one of the basic building blocks for the LionShare project (lionshare.its.psu.edu). LionShare project aims at development of authenticated peer-to-peer network and tries to integrate state-of-the-art technologies in interoperability and security such as OKI, ECL and Shibboleth (shibboleth.internet2.edu).

Conclusions

The work presented in this paper describes another step into achieving interoperability in e-learning systems. We claim that tools, protocols, interfaces and specifications currently available to developers and architects of e-learning systems provide them with the reasonable toolkit to develop systems that are interoperable with existing frameworks.

Acknowledgements

The work at Simon Fraser University Surrey is partially funded under eduSource Canada, a Canarie eLearning Program Project, and through the LORNet NSERC Research Network. The Open Knowledge Initiative started at MIT in 2001 with funding from the Andrew W. Mellon Foundation. LionShare is managed by Mike Halm at PennState and is funded by the Andrew W. Mellon Foundation.

References

[1] Arms, W. Y., Hillmann, D., Lagoze, C., Krafft, D. , Marisa, R., Saylor, J. , Terrizzi, C. and Van de Sompel, H. (2002) A Spectrum of Interoperability: The Site for Science Prototype for the NSDL, *D-Lib Magazine*, 8 (1), January. http://www.dlib.org/dlib/january02/arms/01arms.html

[2] Hatala, M. and Richards, M. Global vs. Community Metadata Standards: Empowering Users for Knowledge Exchange, in: I. Horrocks and J. Hendler (Eds.): *The Semantic Web – ISWC 2002*, Springer, LNCS 2342, pp. 292-306, 2002.

[3] Hatala, M., Richards, G., Eap, T., Willms, J.: The eduSource Communication Language: Implementing an Open Network for Learning Object Repositories and Services, Will appear in Special Track on Engineering e-Learning Systems held at ACM Symposium on Applied Computing (SAC) 2004, Nicosia, Cyprus, March 2004

[4] IMS DRI (2003) IMS Digital Repositories Interoperability - Core Functions Information Model, Version 1, Available: http://www.imsglobal.org/digitalrepositories/index.cfm

[5] Open Knowledge Initiative, http://web.mit.edu/oki/

[6] Simon, B., Miklos, Z., Nejdl, W., Sintek, M., Salvachua, J. Smart Space for Learning: A Mediation Infrastructure for Learning Services. In Proceedings of the Twelfth International Conference on World Wide Web, Budapest, Hungary, 20-24 May 2003.

[7] Van de Sompel, and H. Lagoze, C. Notes from the Interoperability Front: A Progress Report on the Open Archives Initiative. ECDL 2002, Rome, Italy, Springer, LNCS 2458, , pp.144-157

[8] W. Nejdl, B. Wolf, C. Qu, S. Decker, M. Sintek, A. Naeve, M. Nilsson, M. Palmr, and T. Risch. Edutella: A p2p networking infrastructure based on rdf. In *Proceedings of the Eleventh International World Wide Web Conferenc*e, Honolulu, Hawaii, May 2002.