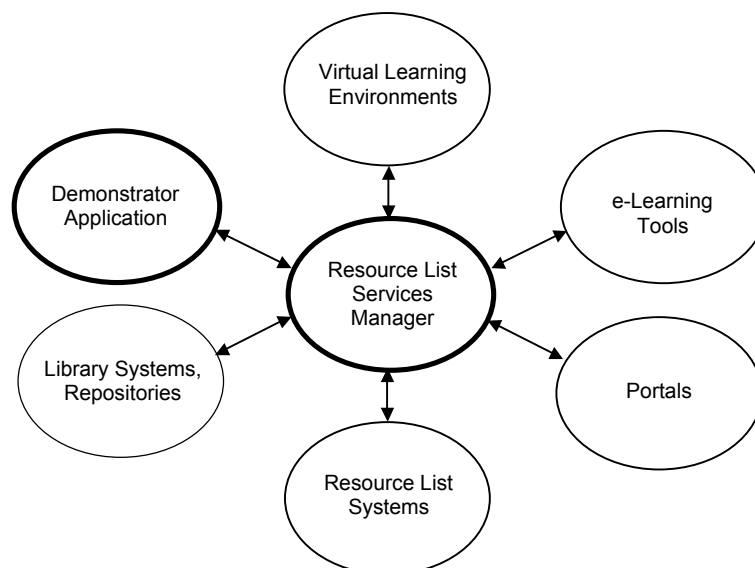


Resource List Services-oriented Management

Tender for JISC E-Learning Programme Technical Development Projects
Version 1.1, Boon Low and John MacColl, 3rd February 2005

Description of Services and Developments

1. The tender describes a project to develop a software development toolkit (SDK) to facilitate one of the learning services domains of the JISC e-Learning Framework – Resource List (Wilson S *et al*, 2004).
2. Resource List is a generic collection of data such as bibliographic citation information, web site links, course reserves and reading materials. Reading (materials) lists are a typical type of resource list. Resource lists are being used in various educational system scenarios ranging from tutor handouts to bookmarks of web resources created and stored in virtual learning environments. Library systems also provide other application contexts, e.g. electronic and printed course reserve lists. The use of dedicated resource list systems has also become prevalent within the various JISC communities.
3. A common characteristic of resource lists in many different scenarios is their lack of reusability. They are often created and stored in proprietary formats, e.g. tutor handouts saved in Word or PDF formats. It is also problematical to export lists which are intrinsically stored in some system repositories. Hence, reusing resource lists often requires mediation and laborious manual tasks such as reproducing the same lists for other system use scenarios.
4. The primary aim of the SDK is to mediate the reuse of resource lists created and stored in distributed and heterogeneous systems ranging from e-learning tools, VLEs, portals, dedicated resource list systems and repositories (see the following diagram). Our proposed SDK consists of two components: a services-oriented manager and a demonstrator application - shown in bold in the diagram.



5. **Resource List Services Manager.** This is a typical application framework which can be installed on a server as a hub to facilitate a range of web services for managing pre-existing resource lists. The services correspond to the established Create/Read/Update/Delete (CRUD) model, and enable resource lists to be exchanged and managed in scenarios involving heterogeneous systems. For example, an instructor can export a reading list from a dedicated system via one of the SDK services to extrinsic e-learning tools and repositories for editing and archiving respectively. Learners can view or import the resource lists stored elsewhere within a course environment in a VLE. The services manager consists of the following two components:
 - *Core web services framework* exists at the core of the manager to facilitate the resource lists management operations by means of SOAP messaging services. The framework is essentially a reference implementation of the IMS Resource List Interoperability (RLI)

Specification (IMS 2004a). It consists of a service **consumer** (RLI source) and **provider** (RLI target) for issuing and responding to web services requests respectively.

- *System interfaces (software bridges)* adapt the above framework to work with the proprietary interfaces of heterogeneous systems, for example, translating a services operation into native system functions such as create and import list as native data.
6. **Demonstrator Application.** Whilst facilitating the exchange of existing resource lists, the services manager is not a resource list creation tool *per se*. There is a need to enable potential implementers of the SDK to create sample lists easily and use them to test the services manager without acquiring a dedicated resource list system. The web-based demonstrator provides for this need as it enables resource lists to be created from federated searching of library and third party licensed databases. We propose a simplistic implementation based on the IMS RLI Specification Use Case 1 (IMS 2004a, 6.1) and reusing the outputs from previous JISC projects – see paragraphs 18-19.
 7. The demonstrator is also a services consumer prototype which sends out the CRUD services requests to, and receives results from the services manager.
 8. A large number of interoperability scenarios are feasible because of the heterogeneities inherent in system types (VLE, e-learning tools, repositories etc.) and implementations (per type, e.g. WebCT, Blackboard, SAKAI). Enabling interoperability for each of these scenarios would require the development of specific system interfaces, bridging the core web services framework of the services manager with the functionalities of the systems. Due to the short time scale, this project will focus only on the development of the core web services framework, i.e. providing a reference implementation of the IMS RLI specification, while piloting a limited set of system interfaces sufficient for basic demonstration and testing.
 9. A set of system interfaces will be developed for demonstrating the following basic interoperability scenarios (mapped to the IMS RLI Specification services and use cases):
 - **Export lists** as web pages, PDF and standard file formats based on the IMS Resource List and Content Packaging Specifications (Create service, Use Case 1, 5).
 - **Manage lists** from within the demonstrator, targeting the following systems (Create/Read/Update/Delete – CRUD services, Use Case 1, 3, 4, 5, 11):
 - VLEs - SAKAI, Blackboard, WebCT
 - A test repository
 - **Retrieve lists** archived in a test repository from (Read service, Use Case 1, 4):
 - The demonstrator
 - Courses in VLEs – SAKAI, Blackboard, WebCT
 10. We anticipate the SDK to be further developed for additional scenarios, particularly the realisation of all IMS RLI use cases within different institutional contexts. For example, implementers can adapt the SDK to work with their dedicated resource list products and VLEs, to facilitate learning objects interoperability in the X4L (JISC initiative) context such as annotating resource lists in the RELOAD editor and then sending the lists to the Jorum repository for archiving. The SDK would therefore underpin future ELF demonstration developments.

Relevant Standards and Interfaces Usage

11. This project will implement the IMS Resource List Interoperability (RLI) Specification. It aims to uncover implementation issues including data and web services (WSDL) bindings validation. We anticipate that the outcomes of the project will contribute towards the implementation recommendations section of the “Best Practice and Implementation” document of the specification (IMS 2004a, Section 4).
12. Development of the services manager will be based on the behavioral model described in the RLI specification, underpinning service operations include lists creating, reading, replacing, deleting, (user groups) assigning and deassigning (IMS 2004b, Section 3). Existing WSDL files from the IMS will be adopted for the core web services framework development. (IMS 2004c, Section 3).

13. The default data schema for this project will be based on the IMS RLI Data Model (IMS 2004b, Section 4). The schema will be utilised as a means of exchanging resource lists among the test systems. In addition, we will investigate the use of the IMS Content Packaging Specification and the IEEE Learning Object Metadata (LOM, which is the IMS Metadata Specification) as optional data schemas, evaluating their relationships with respect to the RLI data model.
14. We will also reuse the metadata mapping work undertaken by previous JISC projects, particularly the ELF d+ mapping of native metadata schemas to LOM and RLI metadata.
15. The OpenURL protocol will be deployed as a means for retrieving and embedding resource lists in VLEs and other environments.

Technology and environments

16. Java technologies provide the underlying environment in which the SDK is developed and deployed. The core web services framework will be deployed on the Apache Tomcat Application Server with JSP and Servlets technologies. AXIS (Apache eXensible Interaction Systems) will be used for web services (SOAP) messaging and to generate Java code from WSDL files.
17. The project aims to explore the use of presentation frameworks for the demonstrator application, i.e. JavaServer Faces with JSTL (JavaServer Pages Standard Tag Library).
18. Existing middleware and open source products will be utilised where appropriate to ensure speedy implementation of the services manager and the demonstrator. We will reuse the outputs from the JISC funded toolkit project, d+ to provide for repository access and integration requirements, namely federated searching and archiving resource lists in DSpace (JISC 2004a).
19. The DEVIL (Dynamically Enhancing VLE with Information from the Library) project piloted nascent application that enables existing metadata to be repurposed and annotated as resource lists in VLEs (<http://devil.lib.ed.ac.uk:8080/resources/>). The application will be reengineered and released as an open source list creation tool as part of the demonstrator. We will incorporate features from a relevant application developed by the MDC (Multiple Distributed Cognition) project team (JISC 2004b).
20. We propose the use of DSpace (<http://www.dspace.org>) as a test repository. DSpace has been tested successfully in the DEVIL project as a preservation repository for resource lists. Work has also been undertaken in the d+ project to enable resources stored in DSpace to be accessible in a services-oriented context. Due to the reuse of previous project outputs, no additional resources are sought for repository developments in this project.

Risk Analysis

Risk	Probability	Impact	Resolution
Staff changes	low	high	We can resource staff from the EUL Information System team and the project partners if need be.
Changes to IMS RLI Specification	low	low	We will implement according to the specification and rely on automatic code generation tool (AXIS) where possible
Proprietary systems integration	definite	moderate	Systems have publicly accessible services and APIs (e.g. DSpace, SAKAI). VLE partners have committed to provide proprietary interface. If integration proves difficult, alternative target systems will be used.
Failure to implement the Resource List Services Manager	medium	high	Previous experience in d+ reduces this risk. Consult OCLC, NeSC, IMS developers whom have experience implementing web services framework. Prioritise on a reduced services scope, i.e. focus only on "createListService".

Relevant Experience

21. Delivering high quality e-learning has been one of the prime objectives of the University of Edinburgh. The University recognises the goals of the JISC e-Learning framework as it seeks to evaluate and develop a standardised e-learning infrastructure for the diverse pedagogic-driven requirements of existing e-learning practices (Ritchie 2003, p.13).
22. The project builds on previous JISC-funded initiatives involving Edinburgh University Library (EUL), particularly the DEVIL project which piloted a resource list application (see paragraph 19). In addition, the EUL was lead partner in the *Theses Alive!* project, and a major partner in *SHERPA*, both funded within the FAIR programme and both deploying DSpace repository solutions. The EUL is also the lead partner of the JISC ELF d+ project which developed a toolkit for accessing heterogeneous repository systems. This project will use the d+ toolkit as a benchmark for implementing the proposed SDK.
23. University of Wales Swansea (UWS) is the lead RSC institution in Wales and the first university to offer support for FE colleges which was later developed by JISC to become the national RSC service. UWS run the Welsh Video Network Support Centre, a contract with UKERNA to provide video support for all colleges of HE and FF in Wales, and the support centre for the Welsh Assembly Life Long Learning network, connecting all 22 local authorities in Wales. Blackboard was first introduced at UWS in September 2001. It was established as the University's VLE in 2003, to provide a mission-critical interface that supports teaching and learning initiatives across a wide range of disciplines and more than 600 academic modules. UWS will provide VLE (Blackboard) integration expertise to this project.
24. WebCT, Inc is a leading provider of VLE solutions and has been an active participant in the development of various learning technologies interoperability standards, including IMS specifications.

Appendix A: Reference

- IMS Global Learning Consortium (2004a), IMS Resource List Interoperability Specification, Best Practice and Implementation Guide, IMS <http://www.imsglobal.org/rli>
- IMS Global Learning Consortium (2004a), IMS Resource List Interoperability Specification, Information Model, IMS <http://www.imsglobal.org/rli>
- IMS Global Learning Consortium (2004c), IMS Resource List Interoperability Specification, XML/WSDL Binding, IMS <http://www.imsglobal.org/rli>
- JavaServer Faces, <http://java.sun.com/j2ee/javaserverfaces/>
- JISC (2003), Dynamically Enhancing VLEs with Information from the Library (DEVIL), <http://srv1.mvm.ed.ac.uk/devilweb/index.asp>
- JISC (2004a), Discovery+ (d+) *Brokerage for Deep and Distributed e-Learning Resources Discovery*, <http://devil.lib.ed.ac.uk>
- JISC (2004b), Multiple Distributed Cognition application, <http://www.jafer.org/mdc.html>
- JISC (2005), E-Learning Programme – Technical Framework and Tools Strand, second call for Technical Development Projects, JISC.
- Ritchie J (2003), e-Learning Development at the University of Edinburgh, report for research commissioned by EUCS
- Wilson S, Olivier B, Jeyes S, Powell A, Franklin T (2004a), A Technical Framework to Support e-Learning. http://www.jisc.ac.uk/index.cfm?name=elearning_framework