Project Document

Monitor Technical Frameworks and Infrastructures

Synopsis:
A report for the JORUM Research and Development Project

Authors: Adrian Stevenson, Tish Dockerty, Andrew Baxter,
(with comments from JISC OSS Watch and UKOLN)

Date: September 2005

Version: Final

Status: Final

Authorised: Signed off by JISC January 2006

REVISION	DATE	CHANGE	DETAILS
1.0	20/04/2005	Adrian Stevenson	Basic document structure and initial content added
2.0	26/04/2005	Tish Dockerty	
2.1		Tish Dockerty	
2.2		Tish Dockerty	
2.3	07/05/2005	Adrian Stevenson	Comments supplied to Tish Dockerty, Andrew Baxter and Jackie Carter in separate document
Final	29/09/2005	Andrew Baxter	Restructuring, additional material and incorporation of comments from Andy Powell (UKOLN) and Randy Metcalfe (JISC OSS Watch).
Final	10/01/2006	Andrew Baxter	Incorporated comments from SE.

Related Documents

RELATED DOCUMENT	DETAILS

Glossary of Acronyms

API	Application Programming Interface
BECTA	British Educational Communications and Technology Agency
CETIS	Centre for Educational & Interoperability Standards
CIE	Common Information Environment
CORDRA	Content Object Repository Discovery and Registration/Resolution
CRM	Customer Relationship Management
CSS	Cascading Style Sheets
DELTA	Distributed e-Learning Tool and Resource Architecture
DEST	Department of Education, Science and Training (Australia)
DfES	Department for Education and Skills
DRM	Digital Rights Management
ebXML	Electronic Business using eXtensible Markup Language
eLAWS	eLearning Annotation Web Service
ELF	E-Learning Framework, now known as e-Framework for Education & Research
FAQ	Frequently Asked Questions
HEFCE	Higher Education Funding Council for England
HTTP	HyperText Transfer Protocol
ICT	Information and Communications Technology
IEEE	Institute of Electrical and Electronic Engineers
JANET	Joint Academic Network
JBI	Java Business Integration
JISC	Joint Information Systems Committee
JMX	Java Management Extensions
JORUM	JISC Online Repository for learning Materials
JPEG	Joint Photographic Experts Group
JSP	Java Server Pages
JSR	Java Specification Request
MDC	Middleware for Distributed Cognition
MERLOT	Multimedia Online Resource for Learning and Online Teaching
MLE	Managed Learning Environment
NMR	Normalized Message Router
OASIS	Organization for the Advancement of Structured Information Standards
OKI	Open Knowledge Initiative
OSID	OpenService Interface Definition
OWL	Web Ontology Language
P3P	Platform for Privacy Preferences
PDA	Personal Digital Assistant
PHP	Personal Home Page
RDF	Resource Description Framework
RSC	Regional Support Centre
RSS	Variously Rich Site Summary, RDF Site Summary or Really Simple Syndication
SCORM	Sharable Content Object Reference Model
SOA	Service Oriented Architecture
SOAP	Simple Object Access Protocol
UDDI	Universal Description, Discovery and Integration
UML	Unified Modelling Language
WCKER	Wizard Construction Kit Extension for Reload
WSDL	Web Services Description Language
WSRP	Web Services for Remote Portlets
XHTML	XML HyperText Markup Language
XML	eXtensible Markup Language
XSLT	eXtensible Style Language Transformations

Table of Contents

Summary and Recommendations

The evolution of Service Orientated Architecture and Web Services has been to the fore in commercial systems development for a number of years. The education sector could gain much by adopting the SOA approach, benefiting particularly from the flexibility and sustainability which it promotes.

Developments in Open Source, both from JISC funded programmes and other international initiatives, are of great significance to the education sector. The JISC's Frameworks and Tools Strand and the Distributed e-Learning Strand toolkits will be of interest when exploring the integration of Open Source systems with proprietary systems.

The growing recognition and pertinence of the JISC e-Framework for Education and Research as a pattern for the future of e-learning reinforces the importance of associating every JISC funded activity to a specific service within the framework.

JORUM may offer elements of its functionality to the community as contributions to the e-Framework for Education and Research.

By keeping abreast of developments in the JISC Common Information Environment, JORUM improves its chances of securing the broadest user base for the services that it offers.

It would be prudent to ensure that any new functionality provided via web services is conformant with the IMS General Web Services specification.

The approach of seeking out existing technologies and adapting or repurposing them is something which should not be despised or feared. We must take heed of the users' needs, whether they are teachers or learners, and ensure that pedagogical demands drive development rather than the technology – but that the technology underpins these needs where appropriate.

1. Overview

1.1 Introduction

Lately, the subject of repositories seems to have dominated e-learning conference and seminar agendas around the world. The role of web services in facilitating the provision of platform and language-independent functionality to client applications has assumed far greater importance. At a recent JISC conference, the consensus appeared to be that the technological difficulties inherent in implementing a repository service no longer represented the main challenge. Issues associated with Digital Rights Management, culture change and usability are now perceived as the principal hurdles that must be overcome. However, this is not to say that the technological issues are no longer significant. Technical developments continue apace, and it is vital for the longevity of the JORUM service that it should maintain pace with developments on this front and evaluate their level of applicability to JORUM.

1.2 Objectives

The aim of this report is to provide a survey of developments in the technical sphere. Specifically, the report has the following four aims:

- To continue JISC IE watch[1], especially e-learning technical architectures.
- To consider the applicability of web services technology to the requirements of the e-Framework.
- To consider the way in which JORUM fits in with local infrastructures.

1 Started under JORUM Scoping and Technical Appraisal Study (Jan 2004), in Volume VIII – "JORUM and the JISC Information Environment". http://www.jorum.ac.uk/research/archive/docs/vol8_Fin.pdf

- To consider developments in related areas of the JISC e-learning programme: e-learning and pedagogy, e-learning and innovation and distributed e-learning.

1.3 Scope of the Report

This report is not intended to be an exhaustive investigation into all of the activity in this sphere. Rather, it represents an assessment of current thinking in technical architectures – and specifically the extent to which the JISC e-learning programme in general, and JORUM in particular, might be affected.

2. Background and Context

2.1 DfES 'Harnessing Technology'

Transforming Learning and Children's Learning – March 2005

The March 2005 Department for Education and Skills (DfES) white paper on e-learning has a broad scope. It includes many observations upon the current state of e-learning, the one of greatest relevance to this report being recognition that growth in e-learning technology has been too haphazard. This has resulted in a large number of systems, few of which are capable of consuming, presenting or acting upon the data produced by the others.

In the words of the report, "*We have allowed for each institution or organisation the freedom to buy their own system and support services. The result is that they are often more expensive than they need be and there are too few economies of scale.*".[2]

The strategy recognises for the first time that a more integrated and systematic approach in our application of technology could have a measurable impact on reaching a much wider set of goals.

'The infrastructure must support these goals. So we plan an integrated teaching, research and administrative network for education. We want common systems for electronic learning, administration and business. We need common open standards to communicate with each other easily and safely. And we will enable all organisations to benefit from a collaborative approach to purchasing ICT equipment and services.'[3]

The suggestion is that a more planned approach to the future development of ICT is required. DfES believes that it is possible to build an open accessible system within a common infrastructure. The intention is for all learners to have a personal identifier and for every institution to offer a personalised online learning environment. For this to be realised, there is a need for a greater number of digital resources of good quality, incorporated into more flexible learning packages. Understandably, the utilisation of existing resources exerts a strong appeal. Previous investment in content can be maximised if teaching staff are able to create, adapt, re-use and share resources using flexible software tools.

This thinking is outlined in priority 6: "Build a common digital infrastructure to support transformation and reform", which has a number of key points:

- Develop best value approaches to ICT infrastructure and services.
- Develop a common systems framework, making it possible to plan the development and adoption of data, technical and interoperability standards for products and services.
- Contribute to the development of common open standards and specifications for interoperability, accessibility, quality of service and safety.

2 DfES 'Harnessing Technology' – Transforming Learning and Children's Learning p.5
3 DfES 'Harnessing Technology' – Transforming Learning and Children's Learning p.7

2.2 HEFCE 'Strategy for e-Learning' – March 2005

The strategy sets out the Higher Education Funding Council for England (HEFCE) vision and implementation plan for supporting e-learning in higher education institutions. HEFCE is clearly committed to working with partners including JISC as the main protagonist to fully embed e-learning in a sustainable way within the next 10 years.

'*…we therefore aim to support the HE sector as it moves towards embedding e-learning appropriately, using technology to transform higher education into a more student-focused and flexible system, as part of lifelong learning for all who can benefit*'.[4]

A clear message throughout the strategy is that Universities are still struggling to 'normalise' e-learning as part of higher education processes. The reason for this is not stated, although there is recognition that e-learning has been criticised for being too technology led but has relatively recently become more focused on the learner and enabling students and other users. This change matches the developments in pedagogy and the increasing need to support diversity and flexibility in higher education.

A key area that is relevant to this report,

'We will seek to support institutions in the strategic planning, change management and process development that are necessary to underpin their development and embedding of e-learning. *This will include strategic approaches to sustainable funding, infrastructure investment, and the development of teaching quality and technological standards. In particular we will support the establishment within institutions of processes and structures that are appropriate for the development and delivery of high quality education underpinned by technology….We will encourage and help institutions to design effective processes and structures, for both people and technology, to align strategies and to develop and deliver quality in services.'*[5]

More specifically, the role of JISC's Information Environment in co-ordinating the joint implementation of embedding e-learning (in HE) is described as follows:

1.4: JISC and the HE Academy[6] to co-ordinate initiatives and align priorities to provide a national support infrastructure comprising e-learning tools, pedagogical and technical advice, and examples of innovative practice in the design and provision of e-learning opportunities.

2.2: JISC and the HE Academy to take forward development and implementation of frameworks and tools for the development and management (acquiring, adding, finding and delivering) of shareable resources and networked learning.

Both the HEFCE and the DfES strategy documents clearly point to the requirement for a frameworks approach to the delivery of e-learning for sound technical, pedagogical and economic reasons.

4 HEFCE Strategy for e-learning – March 2005 http://www.hefce.ac.uk/pubs/hefce/2005/05_12/05_12.pdf
5 HEFCE Strategy for e-learning - p.6
6 http://www.heacademy.ac.uk/

2.3 The Role of JISC

The vision underlying the JISC Strategy 2004-06[7] is one of 'ubiquitous and reliable access to an information and communication environment, so that users are able to enjoy world class technologies in support of their work and study.' In reality, the achievement of this vision requires the development of sophisticated tools, complex management mechanisms, and services to support users and the collaboration of activities across different communities. JISC works with further and higher education by providing strategic guidance, advice and opportunities to use ICT to support teaching, learning, research and administration. With funding from the UK further and higher education funding councils, JISC provides a centralised and coordinated direction for the development of the infrastructure and activities, in line with its strategy. JISC provides:

- New environments for learning, teaching and research
- Access to electronic resources
- A world-class network - JANET
- Guidance on institutional change
- Advisory and consultancy services
- Regional support for FE colleges (RSCs).

JISC is cited as a key partner in delivering HEFCE and DfES's e-Learning Strategies.

2.4 JISC Frameworks Programme

Technology is becoming increasingly embedded in the systems and processes of our educational institutions. Although technology has the potential to extend and improve educational activities, this potential can only be fully realised if the activities are built upon in stable and coherent technical infrastructure. JISC's role is to enable the development and exploitation of a common infrastructure for finding, accessing, delivering and using Internet based resources tailored to and seamless across, education and research communities.

To develop this common infrastructure JISC is already supplying a range of services (including JANET) and is running a number of development activities and initiatives, specifically:

- Information Environment (IE) architectural developments.[8]
- Demonstrators of a Common Information Environment (CIE) that crosses library, research, learning and teaching, museum and archive and health infrastructures.[9]
- Development of Managed Learning Environment (MLE) architectures.[10]
- Explorations into a common technical framework to support e-Learning.[11]
- Infrastructure developments to support e-Research and e-Science.[12]

The aim of the Frameworks programme is to develop and evaluate a framework to facilitate interoperability across learning, teaching, research and their supporting systems.

2.5 A Common Information Environment

A wealth of broadly educational content exists today in digital form. However, the manner in which online content is acquired, described, stored and disseminated throws up obstacles to its retrieval and use; obstacles that we need to overcome if we are to realise the full potential of the content.

The Common Information Environment (CIE) provides a vision in which artificial barriers to the creation, discovery and use of content across domains can be removed. Google and the

7 http://www.jisc.ac.uk/index.cfm?name=strategy_jisc_04_06
8 http://www.ukoln.ac.uk/distributed-systems/jisc-ie/arch/
9 http://www.common-info.org.uk/
10 http://www.jisc.ac.uk/index.cfm?name=pub_mlesg
11 http://www.jisc.ac.uk/index.cfm?name=programme_frameworks
12 http://www.jisc.ac.uk/index.cfm?name=event_report_eresearch

other search engines do not give us full access to the structured databases and repositories of high quality, authoritative content that universities, museums, libraries and other publicly funded initiatives create. In the CIE vision, an individual with a desire to find answers is not required to discover and visit a large number of these in turn, each with a different interface and each home to a body of compelling, high quality content that is essentially invisible to current Internet search engines.

2.6 JISC Information Environment

The JISC Information Environment[13] provides an architecture within which a range of services, tools and mechanisms can be developed to allow colleges and universities to exploit fully the value of online resources and services. The architecture is three-layered: a provision layer, a fusion layer and a presentation layer. It specifies a set of open standards that allow service providers within the JISC IE to work together to provide services more seamlessly than was previously possible. In general, the JISC IE enables the discovery, delivery and use of online resources in ways tailored to support individual and institutional requirements in learning, teaching and research. JISC is one of the organisations sponsoring the Common Information Environment[14], and the JISC IE will contribute to the CIE.

It has been recognised that the way ahead is to align JISC IE architecture more closely with architectures based on Web Services. Such an alignment will set the JISC IE firmly within the more general architectural frameworks being developed to support commercial portal activities and other commercial services.

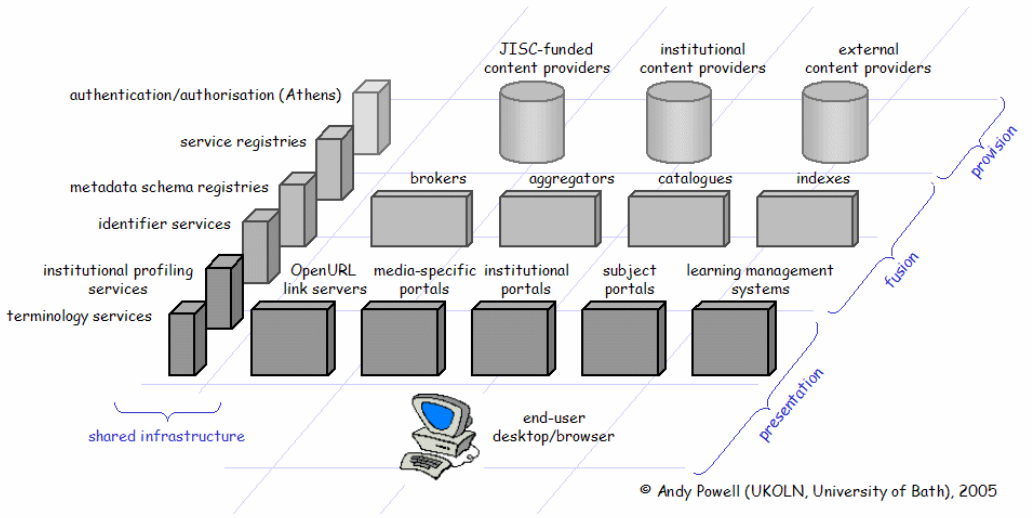

Figure 1 - The JISC Information Environment Architecture[15]

13 http://www.jisc.ac.uk/index.cfm?name=ie_home
14 http://www.common-info.org.uk/organisations.shtml
15 http://www.ukoln.ac.uk/distributed-systems/jisc-ie/arch/

3. The JISC e-Learning Programme

JISC has made a considerable contribution to the development of the MLE (Managed Learning Environment) concept and is supporting the development of interoperability standards. However, there is still a need to produce a coherent vision of how to integrate systems to support institutional processes such as effective e-learning. Research in current MLE development indicates that there has been a rapid expansion but that there is still a lack of pedagogical flexibility and innovation in the design of e-learning tools, environments and architectures. The JISC e-Learning Programme intends to address this issue and will focus on four themes:

- e-Learning and pedagogy.
- A technical framework and tools for e-learning.
- Innovation.
- Distributed e-learning.

The overall aim is to identify how e-learning approaches might be used to facilitate learning and to advise on how these approaches might be effectively implemented by:

- Providing the FE and HE community with accurate, up-to-date and research- or evidence-based information about effective practice in the use of e-learning applications.
- Promoting the application and development of e-learning tools and standards to better support effective practice.
- Developing and evaluating a framework to facilitate interoperability across learning, teaching, research and their supporting systems.
- Considering and promoting the use of innovative technologies and models to support e-learning.

3.1 e-Learning and Pedagogy

The e-learning and pedagogy strand aims to ensure that e-learning, as practised in UK HE and FE, should be 'pedagogically sound, learner-focused and accessible.'[16] The general background for this programme is the ongoing need to support practitioners in realising this aim:

- The widespread implementation and increasing use of Virtual Learning Environments in UK HE and FE has led to demand from practitioners for more effective guidance on good pedagogical practice. There has been a specific call for help in designing e-learning activities in these environments.[16]
- There is evidence that neither learning object metadata, learning environments, nor existing practitioner-based vocabularies for describing learning and teaching are in themselves effective in supporting the development and transfer of effective e-pedagogical practice.

In January 2004 a number of projects were funded under the broad theme 'Designing for Learning', looking at how teachers and practitioners design learning activities. The work focused on four areas:

- **Theme One**: The focus is on developing ways of linking theory to practice.
- **Theme Two**: The study aimed to define effective tools, resources, institutional and national support services.
- **Theme Three:** This area looked at the learning design concept and tools.

16 JISC Invitation to Tender - E-Learning Models Desk Study
http://www.jisc.ac.uk/uploaded_documents/elearning_models-ITT-v2.pdf

- **Theme Four**: A set of case studies has been developed to illustrate examples of effective practice. The Effective Practice with e-Learning guide summarises the work of this theme and is a useful introduction to some of the more theoretical work.

There may be many useful outcomes from which the JORUM project might benefit. In a recent interview, programme manager Sarah Knight said, 'Hopefully the results of these evaluations will feed back to those people developing tools and toolkits in the Distributed e-Learning and Technical Frameworks strands.'[17] The JORUM project needs to develop the real benefits to the end user of the repository, as part of its ongoing service evaluation and promotional literature.

3.2 Frameworks and Tools

Research into e-learning systems and the UK community's active involvement in the development of international learning technology standards and specifications (primarily through JISC and CETIS) have highlighted the potential of technical frameworks to provide a common basis for designing e-learning architectures.

The Frameworks and Tools strand of the e-Learning Programme is exploring the possibility of developing a more flexible approach to technical infrastructures for e-learning. This is based on the concept of a move from the use of large 'monolithic' systems to a more flexible 'service oriented approach'.

The first phase of these projects (10/05/04 to 29/10/04) was to create toolkits for developers (from commercial as well as open-source communities), allowing them to create applications that provide and/or consume services defined within the e-Framework. The second phase (01/04/05 to 30/09/05) funds eight projects to consider the development of web services that address local issues.

Projects funded from the last round which would be worth looking at more closely include:

- **MDC** (Middleware for Distributed Cognition) toolkit project. This project developed a toolkit to enable developers to simply integrate Search and Discover functionality into e-learning applications.
- **D+** Brokerage for Deep and Distributed e-Learning Resources Discovery. This project developed a software toolkit that mediates the discovery of resources in distributed and heterogeneous repositories. D+ particularly focuses on translating queries into formats for searching different repositories.

3.3 Distributed e-Learning

HEFCE has made IT infrastructure funds available to the JISC to develop technologies to underpin the Funding Council's political and strategic agendas and to work with regional and subject communities to use the technologies to support learning and teaching.[18] The aim of defining the Distributed e-learning architecture for teachers is to provide guidance on how to access, plan and use e-learning resources within appropriate e-learning systems. This will include guidance on locating and using existing learning resources and advice on how to share teaching materials with others. The Distributed e-Learning programme will offer benefits to institutions by enabling links between schools, colleges and universities that can be used to encourage progression into higher education for some institutions. The programme will make available some open source e-learning tools that will complement commercially-provided resources.

17 Interview with Sarah Knight - http://www.elearning.ac.uk/features/knightint
18 Strategic Overview: HEFCE-funded Distributed E-Learning Programme 2004-5

There are currently 22 projects funded under this programme. For more information on individual projects see http://www.elearning.ac.uk/del/etools/spws
Projects funded from the last round which would be worth looking at closely include:

- **DELTA** (Distributed e-Learning Tool and Resource Architecture).[19] The DELTA project developed a new range of tools using open standards and specifications that will allow practitioners and learners to share e-learning resources. The project uses some of the techniques from the emerging 'Semantic web' i.e. rather than just searching for resources using simple keywords, such as in Yahoo and Google, the system will be able to reason about these resources and provide much more sophisticated search and retrieval mechanisms. Ultimately, users should be able to search for any type of e-learning resource by either following a structured question/answer dialogue or by browsing a 'graph' for the domain that they are searching within.
- **eLAWS** (e-Learning Annotation Web Service)[20]. The eLAWS tool allows learners to add comments, links and images into the body of existing web pages and share these additions with other learners. It does this without the need to rewrite the original document.
- **JSmirk and SmirkBoard**[21]. JSmirk is an easy to use tool for authoring accessible audio-visual slideshows. It is an open source java version of a tool called Smirk, already developed at the University of Hertfordshire. The final presentations will be like a cross between a discussion board and an online lecture. Each slide will show comments associated with it. Also, the authoring software is so easy you don't need technician support to produce online multimedia: the audio is rendered as MP3 and the graphics as JPEGs. The whole presentation can then be bundled as an IMS content package capable of being viewed in VLEs.
- **WCKER** (Wizard Construction Kit Extension for Reload)[22]. WCKER is a wizard which simplifies the creation of content packages using the Reload tool. Reload is an open source tool which assists the creation and editing of IMS content packages. These IMS content packages can be imported into a number of IMS compliant VLEs while maintaining course structures.

3.4 e-Learning and Innovation

Dealing with pedagogical and technological issues, this strand will identify areas where innovative technologies can be harnessed to support learning, in particular the use of wireless and mobile technologies such as games and 'virtual world' simulation software, voting devices, multi-media PDAs, 3G phones and stronger wireless networks. The benefits offered are increased use of mobile devices and wireless networks that will require new models of learning and teaching. Early indications suggest these technologies will benefit learners by supporting individualised learning, as devices become smaller, quicker, cheaper and easier to use.

Projects are underway with key research labs to identify new technologies that are mature enough to pilot in institutions and to identify the organisational, technological and pedagogical issues that arise from using these technologies. For more information on this programme see http://www.jisc.ac.uk/index.cfm?name=elearning_innovation

19 http://www.essex.ac.uk/chimera/delta/
20 http://eel.hud.ac.uk/
21 http://smirk.herts.ac.uk/
22 http://wcker.conted.ox.ac.uk/cgi-bin/trac.cgi

4. Technology Watch

This section of the report attempts to collate recent technological developments and associated standards, grouping them under broad headings according to their principal area of applicability. Inevitably, there is some overlap between the categories - and some technologies do not fit comfortably into any one pigeonhole - but the headings act as a reasonable guide.

4.1 Communicating the Data

4.1.1 Introductory Remarks on Web Services

A web service may be described as an item of application functionality which can be invoked over the Internet using some standard XML messaging protocol, and which carries a description of the service it provides. The service should be discoverable by other applications wishing to invoke it.

In the context of this document, the messaging protocol is taken to be SOAP (Simple Object Access Protocol), although alternatives do exist. The transport protocol via which the SOAP messages are transmitted is assumed to be HTTP, which is almost invariably the case. The description of the web service is assumed to be expressed in WSDL (Web Services Description Language). No assumption is made regarding the means by which a web service might be discovered by a client application, but a number of competing standards exist. Foremost amongst these are:

- UDDI (Universal Description, Discovery and Integration)[23].
- ebXML (Electronic Business using eXtensible Markup Language)[24].

A web service's description (in the form of WSDL) is typically published to a public registry of services which conforms to one or both of these standards in order to enable its discovery. The JISC IESR (Information Environment Service Registry)[25] does not support UDDI or ebXML at present, but the provision of an UDDI view is being investigated as part of the third phase of the project (now underway, and scheduled to continue until April 2006).

4.1.2 IMS General Web Services

The General Web Services Base Profile document[26] seeks to provide a basic structure for the definition of interoperable web services. It addresses the issue of interoperability at the application level, and recommends the use of:

- HTTP 1.1 as the protocol binding used for SOAP messages.
- SOAP 1.1 as the messaging protocol.
- WSDL 1.1 as the means of defining the service interface.

Whilst web services are based upon standards, these are not currently sufficient to ensure that different implementations of XML parsers on different platforms can communicate. The General Web Services specification seeks to address the most common interoperability problems encountered when implementing web services (although it does not guarantee complete interoperability). To this end, it defines numerous data types in terms of which the parameters supplied to a web service may be specified unambiguously.

Unlike most IMS specifications to date, the documentation package does not include an XML schema definition (".xsd" file) describing a precise data model. Rather, it contains a UML (Unified Modelling Language) model, in the form of a ".xmi" file which may be imported into a popular open-source modelling tool called Poseidon from Gentleware[27]. This software is available in a free "community edition", but one may purchase versions with more

23 http://www.uddi.org/
24 http://www.ebxml.org/
25 http://iesr.ac.uk/
26 http://www.imsglobal.org/gws/gwsv1p0pd/imsgws_profilesv1p0pd.html
27 http://www.gentleware.com/index.php

comprehensive capabilities (such as the ability to "reverse engineer" Java code and create UML diagrams directly from the class structure). IMS has chosen not to make interoperability with other modelling tools (such as Rational Rose) a priority, since models exported from one tool can seldom be imported into another without difficulty. From a UML model representation of the protocol for a web service, a developer can then apply a XSLT style sheet in order to generate a WSDL binding automatically.

Since the specification is likely to be widely adopted in the education sector, it would be prudent to ensure that any new functionality provided via web services is conformant.

4.2 The Presentation Layer

This section concentrates largely on portals and related standards. Since portals form a major component of the JISC Information Environment architecture's presentation layer, standards related to interoperability in this area are of no small significance to JORUM. The two standards discussed below are now implemented by several portal products, and seem set to enjoy widespread adoption.

4.2.1 Portals, Portlets and JSR168

Portals

Descriptions of what constitutes a portal vary widely - depending upon whether one consults a portal server vendor, a developer or a technical manager. However, Sun Microsystems' JSR168 specification provides a good working definition:

"A portal is a web based application that –commonly- provides personalization, single sign on, content aggregation from different sources and hosts the presentation layer of Information Systems. Aggregation is the action of integrating content from different sources within a web page. A portal may have sophisticated personalization features to provide customized content to users. Portal pages may have different set of portlets creating content for different users."[28]

JISC's Portals FAQ[29] provides a more detailed definition :

"Technically, a portal is a network service that brings together content from diverse distributed resources using technologies such as cross searching, harvesting, and alerting, and collates this into an amalgamated form for presentation to the user. This presentation is usually via a web browser, though other means are also possible. For users, a portal is a, possibly personalised, common point of access where searching can be carried out across one or more than one resource and the amalgamated results viewed. Information may also be presented via other means, for example, alerting services and conference listings or links to e-prints and learning materials."

Portals are not dissimilar to conventional web applications, and are to a large extent built upon the same technologies. The content presented to the user by a portal may be provided by one or more portlets (discussed at greater length below), each of which will be responsible for rendering part of the web page presented to the user.

Portal servers typically provide features such as:

- Content Management. Most portals allow a degree of content management (e.g. versioning, access control).
- Collaboration tools. These may include discussion forums, whiteboards, application sharing etc.

28 The Java Portlet Specification 1.0, October 27, 2003.
29 http://www.jisc.ac.uk/index.cfm?name=ie_portalsfaq

- Workflow tools may be provided, assisting users in accomplishing tasks that involve several stages and/or require data from several sources.
- Syndication feeds. A portal server may periodically check (for example) an RSS feed for updates, maintaining the information for use by portlets.

This is by no means an exhaustive list, and there is much variation in the features available in different portal implementations.

Portlets

A portlet may be regarded as a web-based user interface component which provides a view of an information system. Several of these may be aggregated by a portal server into a single page, possibly with some customisation for the person viewing the information.
The JISC Portal FAQ defines a portlet as follows:

"In the context of personalisation and embedding, portals can achieve this through creating distinct building blocks of functionality, e.g., cross-search, alerting, listing, and each one offering a visible component to the user. Each building block is known as a portlet. These can be joined together to create a portal environment, within which various degrees of personalisation can be incorporated, or embedded within a separate environment as required. Portlets feature heavily in many of the current portal building frameworks such as the Apache Jetspeed[30] *project, IBM's WebSphere Portal Server*[31] *and Oracle's Application Server Portal*[32]*."*

Interaction with portlets, including management of their lifecycle, is handled by a portlet container. This is similar to a servlet container (and indeed is implemented on top of one), but provides some additional capabilities along with one or two restrictions. For example, a portlet has access to more information about the user than a typical servlet, and may store information about the user's session in two different scopes – one for information applicable to the whole application of which it forms a part, and one for the private use of the portlet.

JSR168 – The Java Portlet Specification

JSR168 attempts to present a "lowest common denominator" view of what constitutes a portlet, and to specify the minimal level of support that a portal server must provide for the portlets whose content it aggregates. Amongst other topics, the specification addresses details of a portlet's lifecycle, the modes it may occupy, how it handles requests, and how it accesses user information. Some of these issues are examined in more detail below.

Request Handling

A portlet can handle two kinds of request. One of these is an instruction to render its content, and the other is a request for some form of action which changes the state of the portlet. When a portlet receives an action request, it processes this and updates its state before issuing a render request to all of the portlets on the same page (since they may need to alter their appearance as a result of the state change).

Portlet Modes

A portlet's appearance at any given time will depend upon its mode, (i.e. its current functional state). Three standard modes are defined in JSR168. Portlets are not obliged to support any beyond the first of these, but may implement additional ones:

- VIEW – This is the "normal" mode in which the portlet displays its data to the user.
- EDIT – This mode allows the user to configure properties of the portlet.
- HELP – In this mode, the portlet displays help text.

Window States

30 http://portals.apache.org/jetspeed-2/
31 http://www-306.ibm.com/software/genservers/portal/
32 http://www.oracle.com/solutions/enterprise_portals/index.html

The window state determines how much screen space is allocated to the portlet. JSR168 specifies three states:

- NORMAL – The portlet will probably be sharing the screen with other portlets, so should try not to hog too much of the available area.
- MINIMIZED – The portlet should occupy little or no space.
- MAXIMIZED – The portlet should occupy the whole of the available space.

Portlet Preferences

Preferences are the mechanism whereby a portlet is able to perform customisation of its appearance for individual users. Each instance of a portlet on screen has an associated preferences object, and these are stored in the form of a "dictionary" containing strings. Each entry in the dictionary may have several associated values. If the portlet requires a value which is anything other than a string, then some form of conversion will be necessary.

Presentation

JSR168 specifies a set of Cascading Style Sheets (CSS) that portlets should use in presenting their content. These contain the same styles defined by WSRP (the OASIS Web Services for Remote Portlets specification, discussed later in this document).

User Information

A portlet may access user information via a set of attributes, which coincide with the W3C Platform for Privacy Preferences (P3P) and OASIS WSRP standards. These permit the retrieval of such information as contact details, home web page and birth date when these data are available to the portal.

Deployment Information

JSR168 specifies the format of an XML deployment descriptor containing information about the portlet and the services it uses. This includes:

- The name of the portlet.
- Details of user attributes accessed by the portlet.
- Details of any "custom" modes implemented by the portlet.
- Details of security constraints employed by the portlet.

Tag Library

This section of the specification describes the manner in which JSP authors may reference portal components from within their pages, enabling the construction of action and render requests for the current portlet.

Implementations

Several JSR168-compliant portal servers are currently available, including the following, all of which are available as open source products:[33]

- Apache Jetspeed2 (http://portals.apache.org/jetspeed-2/). This is intended to supplant the popular Jetspeed portal server, offering greater scalability and numerous architectural improvements over its predecessor in addition to JSR168 compliance. It is based around Apache's Pluto portlet container.
- Liferay Enterprise Portal (http://www.liferay.com/cms/servlet/PRODUCTS-PORTAL). This product claims both JSR168 and WSRP compliance, and is available bundled with a suitable application server. It comes equipped with a selection of ready-made portlets offering (amongst other options) blog, wiki, RSS feed, calendar and message board functionality.
- eXo Portal (http://www.exoplatform.com/portal/faces/public/exo). This portal is based on Java Server Faces – a relatively recent Sun innovation which (amongst other features)

33 This section draws upon Ch1 of "Professional Portal Development with Open Source Tools", W. Richardson et al, Wrox Press, 2004.

permits a cleaner separation of presentation issues from presentation-related application code than has hitherto been possible.

The uPortal server (http://mis105.mis.udel.edu/ja-sig/uportal/index.html), developed by and for HE institutions, is now fully JSR168-compliant in its latest release (2.5.0 as at 07/09/2005), the portlets being managed by Apache's Pluto container. However, there are currently several serious unresolved issues with the software, details of which may be found at http://jasigch.princeton.edu:9000/display/UPC/2.5.0 .

4.2.2 Web Services for Remote Portlets (WSRP)

The focus of WSRP differs somewhat from that of JSR168, but is complementary. It concentrates on the manner in which a portlet may make its data available to a remote portal server, for aggregation with other portlets. To this end, it specifies standard interfaces via which a portal may invoke the functionality of a remotely hosted portlet, or make its own portlets available for consumption. It also specifies CSS styles (the same ones employed in JSR168), which should be used in rendering the portlet markup.

From the developer's perspective, WSRP is relatively simple – no modifications are required in the portlet code, since a WSRP-compliant portlet container handles all of the details of the interaction. So long as the portlet output conforms to JSR168 (and hence also to WSRP), the most that might be required is a configuration change to the server.

The WSRP specification[34] defines some terms which will be relevant to the scenario which follows:

- Producer – This is a portlet container which makes the output of its contained portlets available via WSRP interfaces. The producer must provide the following web service interfaces:
  - Self-Description: This allows remote portal servers to discover the capabilities of the container and the portlets it hosts, and the information required to interact with it.
  - Markup: This is used to request a portlet's output, or to interact with it.

  The producer may optionally provide additional interfaces:
  - Registration: This is used to establish a relationship between producer and consumer (e.g. for security purposes).
  - Portlet Management: This interface allows control of the portlet's lifecycle, or access to its state data.
- Consumer – This is a portal or application which communicates with producers, gathering the data and presenting the aggregation to users.

A typical interaction between producer and consumer could take the following form:

1. The consumer consults a registry (e.g. UDDI or ebXML), and finds the producer. It may also be able to find out about registration requirements and some detail about the available portlets.
2. The consumer and producer exchange information about security requirements and other capabilities, which may involve a registration stage.
3. The consumer may seek authentication from the user in order to determine whether it can fulfil the producer's security requirements.
4. The consumer requests the output from the remote portlet, and presents the results to the user.
5. The user interacts with the portlet, possibly resulting in additional requests for content.
6. At some point, the user's session ends and the associated resources are reclaimed.

WSRP also defines web service operations to handle complex portlet interactions which may require persistence of the portlet's state.

34 Web Services for Remote Portlets Specification 1.0, http://www.oasis-open.org/committees/download.php/3343/oasis-200304-wsrp-specification-1.0.pdf

4.3 Application and Service Integration

4.3.1 Service Oriented Architectures

Foster, Kesselman and Tuecke define a service-oriented architecture as follows:

"A service is an entity that provides some capability to its clients by exchanging messages. A service is defined by identifying sequences of specific message exchanges that cause the service to perform some operation. By thus defining these operations only in terms of message exchange, we achieve great flexibility in how services are implemented and where they may be located. A service-oriented architecture is one in which all entities are services, and thus any operation visible to the architecture is the result of message exchange."[35]

The service-oriented approach is not new, but is gaining in popularity owing to the degree of flexibility it promises in the construction of application software. It permits operations to be grouped together as an interface, and such interfaces may then be aggregated into a service with the required characteristics. A user (or client application) invoking a service may do so without regard to the manner in which that service is implemented. So long as the service conforms to the interfaces (specified by using some interface definition language such as WSDL, the Web Services Definition Language), clients are insulated from such concerns.

By employing a service-oriented approach, the process of integrating and configuring systems is greatly simplified. So long as the system as a whole does not depend upon functionality outside that specified by the service interfaces, individual services may be regarded as replaceable components owing to the deliberately loose coupling between them. This is in marked contrast to more conventional "monolithic" applications in which the functionality of disparate systems is frequently brought together by "glue" code which is tightly bound to the internal workings of those systems. In practice, such applications prove fragile in the face of changes to any of the component systems, requiring a "patch" to be issued in order for the whole ensemble to continue to operate as normal.

Note that the composition of services to form a system happens at the application level, rather than at the presentation level (e.g. by a portal server). This means that further processing of information from the component services may be performed at this level (by another service!). This discourages the practice of placing application logic in the presentation layer, which usually leads to fragmentation of the code and poor maintainability.

This vision of systems built from interchangeable components depends upon conformance to standards to ensure interoperability, particularly with regard to the way in which the definitions of service interfaces are expressed. It also requires a service infrastructure to enable such operations as:

- Service discovery (via some form of registry holding interface definitions).
- Service monitoring (e.g. examination of state or metadata) .
- Client authentication and other security negotiations.
- Service lifecycle management (instantiation and disposal).
- Fault handling.

SOA and Web Services

Another driver behind the increasing interest in the service-oriented approach is the ubiquity of web services, most often based upon the following standards:

- SOAP - Simple Object Access Protocol, the protocol which specifies the format of message exchange for web services.
- WSDL – Web Services Description Language. This is a format for expressing the operations available from a web service, along with the data required by each operation in order to function. The WSDL also expresses the way in which the abstract definition of the service interface is bound to a particular messaging protocol (e.g. SOAP), but keeps this separate from the interface definition. In this way, the service may support several bindings – perhaps using SOAP for distributed

35 I. Foster, C. Kesselman and S. Tuecke in "The Grid 2" Ch 17, Elsevier Inc., 2004

communications but using a more optimised protocol for interactions with processes running on the same machine.
- UDDI – Universal Description, Discovery and Integration. This is a specification for building a distributed registry of web services, publishing to that registry and discovering services described therein.

These standards present the minimal requirements for usable web services, but do not address higher-level concerns such as security, transaction processing, expression of policy or negotiation of service level agreements. Much effort is currently being directed toward addressing these issues by the provision of additional standards. Of particular relevance from JORUM's perspective are the Open Service Interface Definitions (OSIDs) being developed and promoted by the Massachusetts Institute of Technology's Open Knowledge Initiative (OKI). These are discussed at greater length later in this document.

Further into the future, the Semantic Web offers the possibility of services which carry richer descriptions of their behaviour, which may be discovered by intelligent software agents and automatically integrated into an application to fulfil some desired function.

Benefits of SOA

Amongst the benefits of SOA are the following:
- It allows communication between applications hosted either remotely or locally, the distinction being transparent to the system in all respects other than those relating to performance or reliability.
- It permits greater freedom in the choice (and potentially the re-use) of component systems, since the only functionality on which it depends is that specified in the interface. The interface may be regarded as a contract for communication between the component system and the service infrastructure.
- Applications are scalable, limited only by the capabilities of the service infrastructure.

It would probably be fair to say that the uptake of the SOA approach has been slower in the HE and FE sectors than in others. The appeal of a ready-made system from a major vendor - with the attendant guarantees of service levels and maintenance - should not be underestimated. However, SOA may offer institutions the option of avoiding vendor lock-in whilst remaining responsive to technological developments, without compromising the quality of their information systems or committing too many resources to support.

4.3.2 Java Business Integration (JBI) and JSR208

On 11/06/05, Sun Microsystems announced Java Business Integration 1.0, which specifies an infrastructure for a service-oriented architecture. Along with this came a reference implementation of the standard (which currently provides all of the functionality described in the specification with the exception of service monitoring), and a Technology Compatibility Kit which assists in determining whether or not an implementation conforms to the standard. Taken together, this package represents a realisation of the JSR208 specification[36].

The relevance of JBI to JORUM is hard to gauge at present, owing to the relative youth of the specification and the attendant uncertainty about the likely level of uptake. However, it could well provide a mechanism for integrating disparate information systems and making them available to JORUM, as well as exposing JORUM functionality for consumption by external services. This would depend on suitable JBI interfaces for these systems being implemented.

JBI defines an architecture in which plug-in components communicate via mediated message exchange. This mediation is handled by the JBI environment, an infrastructure which also deals with installing and managing the lifecycle of the plug-ins, and deploying any additional resources (such as configuration information) that they might require. The management of plug-ins is based upon Java Management Extensions (JMX).

36 Java Business Integration (JSR) 1.0, Final Release, 24th May 2005

The plug-ins describe their interfaces to the JBI environment by means of WSDL, and may be of two types:

- Service Engines. These may act as consumers or providers of services (or both), and are either Java-based or possess a Java API. For example, a service engine might encapsulate an XSLT (eXtensible Style Language Transformations) processor, which client services could use to transform their XML output (e.g. for presentation as XHTML). In this case, the JBI environment handles deployment of style sheets to the service, according to instructions contained in an XML deployment descriptor.
- Binding Components. These enable the integration of services external to the JBI environment, which may be based upon non-Java technology or use some communication protocol which is not directly available in Java. For example, a binding component may provide the ability to communicate with remote services via SOAP over HTTP.

The primary function of the JBI environment is routing messages between the plug-in components via its Normalized (sic) Message Router, or NMR. Messages from external services, arriving via Binding Components, must be converted to a normalised form before being communicated to other components. Similarly, outgoing messages must be converted into a form appropriate for the receiving service. Service Engines, being internal to the JBI environment, need not incur this overhead. The router permits different qualities of service to be employed in delivering messages, according to the needs of the binding components and service engines. These are:

- Best effort. Messages may be lost in transit, or may be delivered more than once.
- At least once. The message is guaranteed to be delivered, but may still arrive more than once.
- Once and only once.

It may seem odd to specify an unreliable service, but sometimes this is needed. For instance, the message may have a finite period of validity (e.g. a timestamp), after which there's no point in delivering it at all.

The normalised message has three main parts:

- The bare message – XML conforming to the WSDL-defined message type.
- Metadata – This may include information about transaction state or security information.
- Attachments – This may consist of non-XML data referenced by the message.

It is worth noting some issues that the specification does not address:

- Messages are regarded as short-lived entities, and no recommendation is made as to how they might be placed into relatively persistent storage (e.g. to allow for recovery following system failure)
- As described in the specification, JBI is not distributable (i.e. an instance of a JBI environment can't be hosted across multiple servers, and migration of services from one server to another in the event of failure cannot take place).

The specification indicates that these topics may be addressed in a future release, but makes no firm promises.

4.3.3 The Open Knowledge Initiative (OKI) and Open Service Interface Definitions (OSIDs)

OKI[37] (based at the Massachusetts Institute of Technology) develops and promotes interface definitions (the OSIDs) that describe how the components of an educational information environment should interact. As may be inferred from the name, the OSIDs represent a service-oriented approach to integrating applications. At the 2005 Alt-i Lab in Sheffield,

37 http://www.okiproject.org/index.html

representatives of OKI were able to demonstrate interoperability of their Sakai[38] learning environment with Harvest Road's Hive repository[39] via the OSID interfaces.

Until 02/06/05 the OSIDs were only available in the form of Java interfaces, but on that date OKI released the XOSIDs, a language-neutral expression of the interfaces. Work is underway to port the OSIDs to PHP, Objective-C and C#.

The OSIDs are open source, and are available from SourceForge[40]. A wide range of functional areas is addressed by the interfaces, the most significant from JORUM's perspective being the repository OSID (which OKI regard as one of the more mature definitions), and possibly the workflow and content packaging OSIDs (for the latter, an implementation is available).

4.3.4 Content Object Repository Discovery and Registration/Resolution (CORDRA)

CORDRA is a model of how to enable the next step in the evolution of e-learning, namely, how to solve the problem of finding and reusing learning content.

Anticipating the desire of the global e-learning community to take advantage of the economy of re-use, the CORDRA project was conceived by a small group of individuals who had been involved with SCORM and digital object management. They envisioned an infrastructure scheme that was based on registering an organization's content in a way that would facilitate broad re-use of the sharable learning content beyond the original organization.
As the amount of learning content that conformed to the SCORM guidelines has grown, attention has shifted towards solving the problems of discovering, locating and retrieving that content. Without such facilities, the investment in producing conformant content is lost. An infrastructure to support search and retrieval of content does not currently exist.

CORDRA's goal is to develop a model of how to enable the next step in the evolution of e-learning, namely, how to solve the problem of seamless discovery and access to learning content. They approach this problem through the creation of interoperable registries of content and content repositories, i.e. establishing collections of repository federations, all conforming to a set of agreed standards. Building upon existing technology from the worlds of learning content management and delivery, content repositories and digital libraries, this model aims to identify and specify (not develop) appropriate technologies and existing interoperability standards that can be combined into a reference model that will enable learning content to be found, retrieved and re-used.

The CORDRA system, like the Google search engine, will rely on a simple search of a catalogue of "metadata" to find specific content the instructional designer or other user has requested. Google's search area is limited to what can be found on the Internet: Google does not find content that resides in repositories, nor can it provide content specifically authored for learning. Its index/search functions do not generally rely on formal, authored metadata (such as IEEE Learning Object Metadata). The CORDRA system will provide a way to search for learning content via the catalogues of authored metadata in the registries. CORDRA believe that federating the metadata for content from several repositories via a registration process and then running the search on the combined metadata will avoid problems encountered by processes that attempt a direct search of multiple content repositories.

CORDRA will be based on existing standards for learning content, repositories and digital libraries. The CORDRA project's proponents believe that sufficient standards and technologies exist; what is needed is to utilize them in combination to achieve interoperability. Thus, while the "blueprint" for the CORDRA model will combine and refine other standards documents as necessary, the CORDRA model should not be regarded as a completely new specification.

38 www.sakaiproject.org
39 www.harvestroad.com
40 http://sourceforge.net/projects/okiproject

The CORDRA approach raises concerns about control and accountability. MERLOT's[41] Martin Koning-Bastiaan has some concerns about the fact that MERLOT holds metadata about learning resources, but does not hold the resources themselves. Just giving away all the metadata to a higher level CORDRA instance would be problematic, since that would mean giving away the whole value-add of MERLOT. Similarly, Marek Hatala of Simon Frasier University has concerns about what kinds of control a repository owner would have to give up in practice when joining a CORDRA instance. The Splash[42] and LionShare[43] personal, peer-to-peer repositories rely heavily on the fact that their owners retain control over what gets exposed to whom.

Whilst JORUM is well placed to comply with CORDRA, there is as yet no firm position with regard to its adoption, pending further investigation as part of the JORUM Service in Development phase.

41 http://www.merlot.org/Home.po
42 http://www.edusplash.net/
43 http://lionshare.its.psu.edu/main/

5. A Technical Framework to Support e-Learning

The aim of a framework is to support the development of technical environments in individual institutions, enabling them to develop their own architectures specific to their own needs and existing technologies. The framework ought not to be considered as imposing a blueprint as such, but more as a requirement for compliance to a set of standards and the provision of a library of functionality that can meet the diverse needs of HE and FE in their learning goals. A useful analogy is, 'the framework provides a vocabulary and grammar - it is up to individual institutions to write the stories' [44]

A framework helps by providing communities with a common reference set of service definitions, a toolkit to assist developers and a way of coordinating efforts related to standards and shared services.

JISC has found it useful to adopt a framework approach since it supports multiple institutions which conduct R&D on its behalf. It is the intention that applications developed using the framework can be easily re-used and integrated by and with other organisations. JISC has formalised this approach by developing the e-Framework for Education and Research, formerly known as ELF. This exercise was initiated by JISC in collaboration with DEST[45] (Australia), and Industry Canada (details of which can be found in 'An e-Learning Framework – A Summary'. http://www.jisc.ac.uk/uploaded_documents/Altilab04-ELF.pdf).

44 S. Wilson, B.Oliver, S.Jeyes, A.Powell, T.Franklin, A Technical Framework to Support e-Learning.
45 http://www.dest.gov.au/

5.1 The JISC e-Framework for Education and Research

The e-Framework (formerly ELF) is an attempt to catalogue the functions that are most common within an e-learning system, a 'palette' of possibilities, as Scott Wilson describes it,

'*The idea is that anyone wanting to develop an e-learning application can look at this palette, select services of interest, download toolkits provided by other developers and help integrate the service, and then incorporate them into their application*'[46]

Below is a diagram showing the services. The upper blue set of boxes identifies applications (largely focused on presentation-tier technologies) via which users may interact with the other services provided by the framework. The second green layer shows services specific to the domain of e-learning. The lowest layer identifies services which are common to multiple application domains.

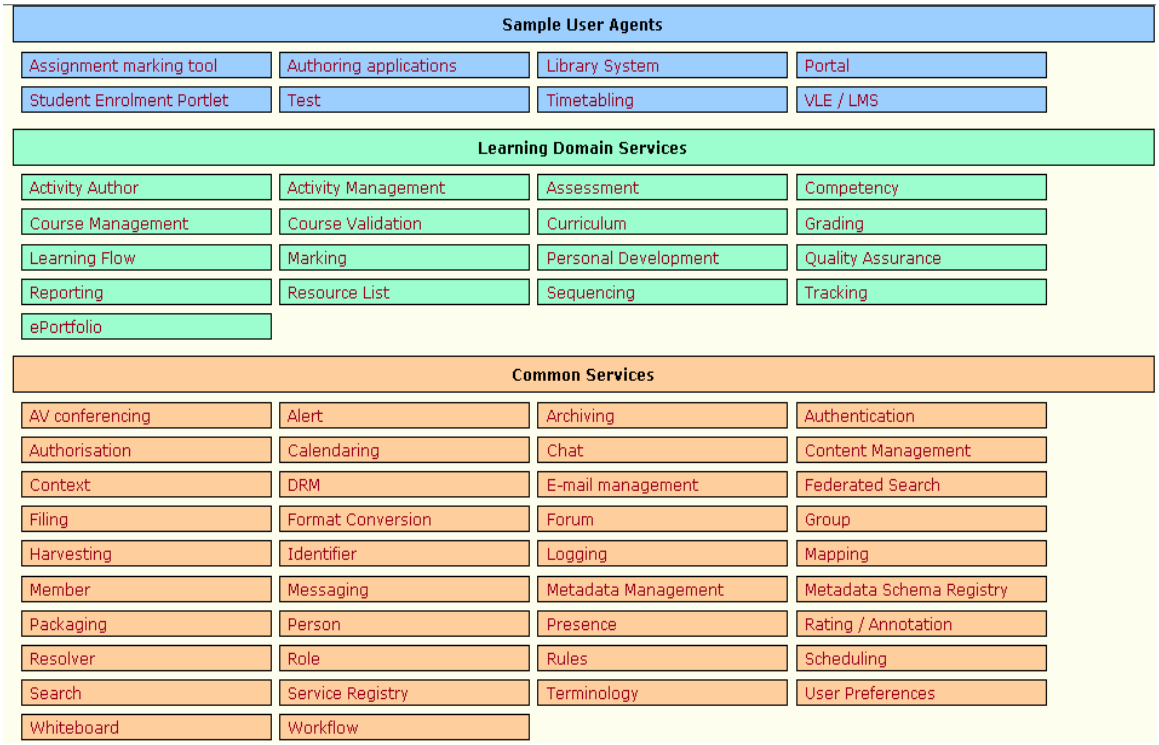

Figure 2 - The Components of the JISC e-Framework for Education and Research[47]

The purpose of this categorisation is to aid a wider community in deciding what components might be relevant for their own use.

Scott Wilson stated at a recent conference, he was "not sure what ELF means anymore, it's just ELF now", signifying a recognition that it makes little sense to limit its applicability solely to e-learning. The framework's functionality could equally well be of use in areas such as research and administration.

Applications developed using the framework can potentially be re-used and integrated by and with other organisations. JISC is currently funding a range of projects aimed at exploring areas of functionality in the framework, specifically by developing toolkits, libraries and exploring open standards.

46 'Can web service technology really help enable 'coherent diversity' in e-learning?' Scott Wilson www.elearning.ac.uk/features/pedagandws
47 As at 29/08/05. The current framework is available at http://www.elframework.org/framework/

The current priority is to disseminate the e-Framework to the wider development community in order to keep it alive and evolving. JISC is currently funding the implementation of some of the web services, but it will probably take another two years before the construction of web service based e-Framework applications becomes a reality.

The significance of the e-Framework for JORUM is considerable. The ability to compose a new application from a catalogue of re-usable components (or to contribute such components for use by others) has the potential to change the whole structure of the repository application. JORUM could also offer elements of its functionality (including ancillary services) as a contribution to the e-Framework. These may in turn depend upon other services from the framework.

Examples of e-Framework components offering services which might be used by JORUM (or equally, areas in which JORUM might contribute services) include the following elements of the “Common Services” stratum in the diagram above:

- Authentication
- Authorisation
- DRM
- Federated Search
- Format Conversion
- Forum

JORUM could also provide or consume services from elements of the “Learning Domain Services” and “Sample User Agents” stratum, perhaps including:

- Authoring Applications
- Portal
- Assessment
- Learning Flow
- Sequencing

6. The Service-Oriented Approach

This section complements the technical information about Service-Oriented Architectures which was presented in section 4.4.1, concentrating more upon the business-related implications surrounding its adoption.

6.1 Business Drivers

Technology is becoming inseparable from the majority of educational activities (whether pedagogical or administrative), and it is necessary to seek more efficient means of integrating existing systems and augmenting their functionality. For example, departments are naturally loath to lose their investment in an information management system because of an institutional commitment to the adoption of a new VLE. A better solution would be to find some means of integrating the existing system, which might also help to minimize the changes in working practices required of staff.

The arrival of web services in the arena promises to simplify the integration of disparate external services, particularly if these conform to open standards. Therefore whether it's an e-CRM (Customer Relationship Management) system or a virtual campus, there is a need to look beyond the 'turn-key' or 'shrink-wrapped solutions' to a more flexible, open, modular structure.

Amongst the many business benefits of such a Service-Oriented Architecture, the following considerations make a particularly compelling case for its adoption:

- The functionality provided by the system is more likely to be driven by business requirements, rather than by technical developments or limitations. The creation of services arises from a genuine need, rather than being dictated by a vendor's view of what functionality ought to be provided.
- The system is more robust in the face of changing business processes or organizational change, since components need not be closely dependent upon each other and are not inextricably bound into a monolithic application. Owing to the system's loosely coupled character, individual components can be added or modified in order to integrate legacy systems or to provide new functionality;
- The resulting system is much more closely tailored to an institution's needs.
- The return on investment is likely to be better owing to the system's flexibility, since upgrades do not necessarily involve the installation and configuration of a complete system.
- Modules may be selected upon the basis of fitness for purpose, rather than the choice being forced by vendor lock-in to a particular product.
- The cost of entry is potentially smaller, since toolkits and modules are likely to be freely available under open-source licences and to comply with open standards.
- In principle, a SOA-based system should be less expensive to maintain. It is likely to require less effort on the part of the developer tasked with integrating the component systems. (However, it should be noted that there exists considerable disagreement as to whether or not the "total cost of ownership" of a system based upon open source software is really so small. The JORUM R&D Report on Open Source Learning Object Repositories (to be published) discusses this in greater detail.)

However, it must be borne in mind that of the plethora of specifications surrounding web services (dealing with reliable messaging, encryption, expression of policy, transaction processing, message attachments etc.), most are very much "works in progress". Many specifications have stabilised to a usable (and often widely used) form, but others are more mutable. Ease and cost-effectiveness of development must be weighed up against a lack of richness of functionality at present. That said, the potential gains in system flexibility and the freedom from being bound to particular vendors and technologies should not be understated, and there is no shortage of effort being directed towards remedying the current shortcomings of web services.

In addition, it would also be relevant at this stage to consider Scott Wilson's suggestion that a service-orientated approach could lead to greater pedagogic diversity[48]. He contends that a web service interface, being neutral with regard to platform and programming language, permits a freer choice by users of which application they use to connect to that particular service. He also says:

'Using this "composite application" approach, it should be easier for an e-learning developer to combine features in such a way that they support the educational objectives they seek to attain, but without having to develop everything from scratch.'

However, SOA ought not to be regarded as the only approach to systems integration. Very effective solutions are currently based upon standards such as OAI-PMH and SRW (as supported by the JISC IE), so the full machinery associated with SOA may not be necessary in many cases.

6.2 Is it too late?

As Derek Morrison states in his article 'e-Learning Flexible Frameworks and Tools: Is it too late?'[49]

'The problem is that some institutions, at least, have been doing more than gathering experience and have made a full-blown strategic commitment to products which represent only one way of offering e-learning.'

Morrison worries over the burgeoning monolithic e-learning culture that is developing within HE and FE institutions, where strategic decisions that cost thousands have been made that tie that institution tightly to one vendor's system. Given the virtues of the e-Framework one might hope that it would be ubiquitous in HE/FE, but the reality is somewhat different.

6.3 To be Service Oriented or not?

Whilst the SOA approach has much in its favour, it should not be presented as a panacea (any more than should a major vendor's VLE). There are several reasons why an institution might decide that it does not seem right for them.

It may be an issue for a school or college that have little in the way of computerized systems and as yet hasn't contracted for a VLE from one of the big vendors. Add to this a general wariness and lack of understanding about how e-learning is going to work within their institution and how it will benefit their existing educational output, then the prospect of deciding what business processes they are going to have to support is a daunting one (although this neglects the fact that organization like JISC and Becta[50] are able to assist in advising as to how to go about the activity). But as Sarah Holyfield states,

'One of the advantages of purchasing a large commercial system is that it defines how many of the business processes take place, whether these are concerned with stock control, or how invoices are handled and so on.' [51]

However, this could equally be interpreted as a disadvantage – your business processes must fit the system, not the other way around!

The SOA approach is definitely gaining many adherents, but if it is to attain its considerable potential within the next two to three years then a good deal of investment in training of in-house IT personnel will be required. Ability will be required in a broader set of skills than has hitherto been the case – possibly in data modelling, application design, application

48 S. Wilson – "Can web services technology really help enable 'coherent diversity' in e-learning? http://www.elearning.ac.uk/features/pedagandws/view

49 http://www.bath.ac.uk/dacs/cdntl/pMachine/morriblog_comments.php?id=P164_0_4_0

50 http://www.becta.org.uk/

51 Sarah Holyfield March 2005 A non-technical guide to technical frameworks – Part One: http://www.elearning.ac.uk/features/nontechguide1

integration, testing and project management. Ultimately organisations will need to change the way they think about applications and develop new skill sets and tools, as well as spending money to create infrastructure supporting the thousands of loosely coupled components replacing hundreds of more monolithic applications.

Doing what everyone else does is cheaper than designing and implementing your own pedagogic vision - although the web services approach certainly makes it considerably more affordable than it used to be. The benefits still have to be sold to management, and a suitably experienced software architect has to be employed to write the code that knits all the services together and present the kind of experience that the designer envisaged.

From JORUM's perspective the SOA approach is appealing for several reasons. Although Web Services are by no means the only route to offering services to remote systems, the ubiquity of HTTP and the ready availability of development tools are likely to lower the cost of entry for those wishing to consume JORUM-provided Web Services or to provide localised instances of JORUM. Also, the cost to JORUM of integrating remotely hosted services is likely to be significantly reduced. The loosely-coupled nature of the SOA architecture would in principle enable JORUM to respond more rapidly to changing requirements, maximise the extensibility of the service as a whole and minimise dependence upon particular implementations of functionality.

6.4 Sustainability

The ability to self-service a system that has been designed in-house is an important aspect that would require attention to staff training and retention. At a minimum, the following areas would need to be considered:

- A strict policy on the maintenance of documentation (both user guides and source code comments) would certainly be required, since having knowledge of vital elements of the system residing solely in the heads of one or two key individuals is a dangerous situation.
- Configuration, deployment and maintenance of application servers, web applications and associated infrastructure is likely to occupy a large slice of someone's time.
- Maintenance of application source may well constitute more than one full-time job, (especially if developers spend some of their time answering queries from other institutions using packages which have been developed in-house and released under an open source licence). For this to be sustainable, there ought to be adequate provision of issue tracking software (for logging bug fixes, requests for additional features etc.) and a version control system (which will enable previous versions of code to be recovered in the event of a failure, and will assist in merging code changes if several developers are working on the same module). For open source projects, sites such as SourceForge[52] offer these facilities free of charge.
- Staff and/or student training in the use of systems may well be necessary.

The level of investment required to enter this arena may prove too taxing for some institutions, especially smaller organisations within sectors like FE. A service contract with WebCT[53] or Blackboard[54] could look more attractive for these organisations.

The JORUM R&D Report on Open Source Learning Object Repositories (to be published) contains further detail on this topic.

52 http://sourceforge.net/
53 http://www.webct.com/
54 http://www.blackboard.com/uki/

6.5 The Disjunction of Pedagogy and Technology

It is pertinent to ask whether or not the e-Framework really contributes to learning goals and outcomes. The range of components could be employed in many IT systems outside the scope of e-learning. The range of web-accessible technologies and services which are particularly useful to education can be classified broadly into three categories.[55]

1. Existing web services that are designed specifically for education;
2. Web services that are not specifically designed for the purpose but are essential for enabling it;
3. More widely deployed web-services that could be adapted for learning, education and training;

There is a growing emphasis within support services such as JISC on a wider understanding of e-learning which is not just focused on the current technological concerns. If teachers are to embrace ICT in learning there is a real need to examine issues concerning accurate, up-to-date, evidence and research-based information about effective practice in the use of e-learning tools. In her article 'Why Focus on Pedagogy?', Christina Smart cites a recent survey by JISC of FE and HE institutions which suggests that while there may be greater learner access to online resources, 'pedagogical issues…are of less concern'.[56]

6.6 ELF (e-Framework for Education and Research) Conference, Oxford, November 2004

The Conference covered many issues and the key points that were relevant to repositories are as follows:

- Identifiers – there is still ongoing R&D into Identifiers to ensure that they are robust enough to deal with shifting data needs;
- Digital Rights Management – need to tackle rights and responsibility issues;
- Metadata – the current model of manual metadata creation is unrealistic and unsustainable, a distributed model is likely to be more effective;
- There needs to be an attempt to identify and classify different types of repositories;
- The industry shouldn't underestimate the importance of Open Source technology;
- More emphasis is required on prototyping and usability than technical issues.

There seemed to be agreement that the area of repositories was one of the more mature parts of the framework (much work already having taken place in the digital libraries sector), and that the associated technology was relatively robust. Key technical issues included the questions of identifiers and rights management (where the discussion centred upon the interface between the technical, legal and social dimensions). Other issues raised included the need for institutions to become more open, the need for common vocabularies, the question of sustainability and the need to find out what the community wanted.

55 Blinco, Mason, McLean, Wilson 'Trends and Issues in E-learning Infrastructure Development'
56 http://www.elearning.ac.uk/features/whyped/view

7. Conclusion

The evolution of Service Orientated Architecture and Web Services has been to the fore in commercial systems development for a number of years. The education sector could gain much by adopting the SOA approach, benefiting particularly from the flexibility and sustainability which it promotes.

Developments in Open Source, both from JISC funded programmes and other international initiatives, are of great significance to the education sector. The JISC's Frameworks and Tools Strand and the Distributed e-Learning Strand toolkits will be of interest when exploring the integration of Open Source systems with proprietary systems.

The growing recognition and pertinence of the JISC e-Framework for Education and Research as a pattern for the future of e-learning reinforces the importance of associating every JISC funded activity to a specific service within the framework. The ultimate aim is that all of the e-Framework services will conform to internationally agreed standards and specifications, and there will exist a library of open-source modules whose functionality is exposed by means of web services. Using these components, developers will be able to pick and choose the functionality they require in constructing their e-Learning solutions. The framework promises to offer a rich range of functionality which might be exploited by JORUM – either in providing additional services or even in providing core services as part of a hybrid open source/proprietary solution.

JORUM may also offer elements of its functionality to the community as contributions to the e-Framework. Interfaces to these services could be made discoverable through publication in registries such as the JISC IESR[57] (including interfaces based upon Web Services as soon as the IESR supports these). Examples of areas in which JORUM could contribute or use services from the e-Framework are detailed in section 5.1.

The prospect of a Common Information Environment is not unrealistic. The creation of registries and repositories of content which all conform to a set of agreed standards is of major significance, even though this alone is not sufficient ensure interoperability. By keeping abreast of developments in this area, JORUM improves its chances of securing the broadest user base for the services that it offers.

The approach of seeking out existing technologies and adapting or repurposing them is something which should not be despised or feared. We must take heed of the users' needs, whether they are teachers or learners, and ensure that pedagogical demands drive development rather than the technology – but that the technology underpins these needs where appropriate.

57 http://iesr.ac.uk/