

## The e-Learning Framework – Summary

This paper provides a quick overview of the services identified within the e-Learning Framework activity. To see the current state of the e-Learning Framework, and look at the details of the service definitions, go to <http://www.cetis.ac.uk:8080/frameworks>

### ***What is the e-Learning Framework?***

The e-Learning Framework is a service-oriented factoring of the core services required to support e-Learning applications, portals and other user agents. Each service defined by the Framework is envisaged as being provided as a networked service within an organisation, typically using either Web Services or a REST-style HTTP protocol.

The ultimate aim of the Framework is, for each identified service, to be able to reference an open specification or standard that can be used to implement the service, and also to be able to provide open-source implementation toolkits such as Java and C# code libraries to assist developers.

The intention is not to provide a blueprint for an open-source solution, but rather to facilitate the integration of commercial, home-grown, and open source components and applications within institutions and regional federations, by agreeing common service definitions, data models, and protocols.

The Framework began life within JISC as a way of making sense of its funded development activities within the learning and teaching space, and to focus future efforts.

### ***Overview of services***

The model on the following page shows the set of services defined within the framework. The upper sets of boxes identify services specifically within the domain of e-learning; the lower set identify services that may be common across multiple domains. The services are clustered into logical groups to aid readability, however there are no dependencies or explicit associations between service definitions<sup>1</sup>.

---

<sup>1</sup> In practice, if several services with similar capabilities are exposed in an environment, the service interfaces may be realized using a shared implementation; for example, Presence, Context, Chat, and AV Conferencing services could all be managed using a single Jabber server.

Sequencing	Assessment	Course Management	Personal Development	Course Validation
Activity Management	Marking	Resource List	ePortfolio	Quality Assurance
Learning Flow	Grading			Reporting
Tracking	Competency			Curriculum
Activity Authoring				
Authentication	Resolver	Mapping	Whiteboard	Calendar
Authorisation	Metadata Schema Registry	Format Conversion	Chat	Scheduling
DRM	Metadata Management	Filing	AV Conferencing	Group
Role	Harvesting	Logging	Context	Member
Rules	Search	Workflow	Presence	Person
	Federated Search	Service Registry	E-mail Management	User Preferences
	Archiving	Identifier	Messaging	
	Content Management	Packaging	Forum	
	Rating / Annotation		Alert	
	Terminology			

## ***Service Descriptions***

Below are brief descriptions of the services shown in the diagram:

### **Application Services**

#### **Sequencing**

A sequencing service provides support for the use of sequenced learning objects, primarily of use within VLEs and similar environments for the delivery of complex learning packages.

#### **Activity Management**

Supports the management of learning activities.

#### **Course Management**

Supports management of courses, modules and other units of learning

#### **Resource List**

Supports the creation, access and management of reading lists and other lists of resources.

#### **Assessment**

Supports the use of automated assessments.

#### **Grading**

Supports submitting grades against courses, modules, and other units of learning.

#### **Competency**

Supports the management of competency frameworks, and the mapping of units of learning, assessments, and activities against specific competencies.

#### **ePortfolio**

Supports the management of artefacts created by learners, such as essays and projects.

#### **Learning Flow**

Supports the management and use of complex learning scenarios.

#### **Activity Author**

Supports the process of creating learning activities.

#### **Marking**

Marking service provides automated marking services. It could also support a dropbox method for traditional artifact-based assessment.

#### **Curriculum**

Supports curriculum development and access to curriculum structures and properties.

#### **Course Validation**

Supports the lifecycle of course proposal, approval, and validation.

**Quality Assurance**

Supports the recording and retrieval of quality assurance statements, such as second marking of assessments and approval of content for use in courses.

**Personal Development**

Supports the management of a learner's personal development plans and personal development records.

**Reporting**

Provides standardised institutional (performance) data to outside agencies.

**Tracking**

Supports recording and retrieval of performance information; typically this is as a result of students undertaking learning activities or interacting with materials.

**Common Services****Messaging**

Allows broadcast of messages to users and groups using appropriate communication technology.

**Authentication**

Allows the identity of agents to be established.

**Authorisation**

Supports the management of access to resources by agents.

**Resolver**

Provides services based on the use of OpenURL (or similar) metadata, including redirection to document delivery services, redirection to online bookshops, to local library services, and other discovery activities.

**DRM**

Supports the allocation and application of rights policies against resources, consuming data in a digital rights expression language (DREL) to determine access. Typically works through Authorization services, and is generally intended to be called by Authorization implementations as the result of a request to use a resource.

**Metadata Schema Registry**

Supports the registration of metadata schemas, including obtaining definitions of elements.

**Logging**

Generic logging service for applications.

**Identifier**

Supports the creation, registering and deregistering of identifiers for objects.

**Filing**

Supports access to remote storage facilities, such as a SAN or central RAID, for storage and retrieval of arbitrary data sets. For long-term preservation of materials, an Archiving service should be used instead.

**Workflow**

Supports generic workflow management.

**Search**

Supports the finding of information resources including learning objects, assets, e-reserves, learning opportunities, funding sources and so on. Search will typically target a single search type (such as SRW for library resources) although a Search service can support multiple search types that can be selected from by consumers. For searches that aggregate results from across multiple types of search, a Federated Search service should be used instead.

**Service Registry**

Supports discovery of available services.

**Mapping**

Supports cross-mapping of values in different namespaces, such as crosswalks between metadata schema.

**Presence**

Provides information about a user's on-line status, much like a chat application.

**Rules**

Supports access to, creation, and management of rules and policies, such as may be used within access management or workflow processing. Rules may be machine readable or human readable.

**Harvesting**

Allows application services to support resource discovery by harvesting copies of some or all metadata records.

**E-mail management**

Support for email management (as provided by IMAP, POP and SMTP)

**Scheduling**

Supports the management of allocation of resources against time, such as rooms, people and equipment.

**Content Management**

Supports the publishing, retrieval, description, and organisation of information resources, including their lifecycle management.

**Packaging**

Supports the assembly of packages of information resources by aggregation and disaggregation, and their preparation for transport and delivery.

**Archiving**

Supports the long-term preservation (and managed destruction, if appropriate) of materials.

**Rating / Annotation**

Provides support for the creation, management and use of secondary metadata (user ratings and text annotations).

**Terminology**

Provides automated, machine-readable mappings between terms, either within a particular thesaurus or across multiple thesauri or classification schemes.

**User Preferences**

Provides machine-readable information about users' personal preferences.

**Chat**

Manages one on one messaging as well as multiple users.

**Federated Search**

Supports the processing of searches that target multiple types of repository, such as a combined search using SRW, XQuery and Z39.50 protocols against repositories supporting a range of different metadata formats the results from which are then aggregated for presentation to the consumer. Contrast with Search service, which is aimed at supporting single-type searches.

**Group**

Supports access to information about groups, including courses, modules, activities, seminar groups, teams, and departments.

**Person**

Provides a means to manage basic information about people, such as names, contact information, and perhaps some types of demographic and system management information, such as would typically be stored in a directory.

**Role**

Supports the management of role types, whether for access management (security roles) or organisational management (job roles) purposes.

**Member**

Supports the management of membership of persons in groups, such as student enrolment on modules.

**Calendaring**

Supports the sharing of calendars, such as personal calendars and course timetables.

**Metadata Management**

Supports the management of metadata for objects, including creation, validation and retrieval.

**Format Conversion**

Supports transformation of information from one format to another; for example, from IMS Content Packaging to METS.

**Alert**

Allows access to news, updates, and announcements.

**AV conferencing**

One of the collaborative services. It includes functions like audio and videoconferencing.

**Context**

Supports accessing information about what activity a user is currently engaging in, such as the module or activity that a student is accessing within a learning system. Also supports registering and deregistering of context by applications.

**Whiteboard**

Supports the use of collaborative tools that focus on shared editing and idea sharing, such as electronic whiteboards.

**Forum**

Supports the use of asynchronous collaborative messaging, as offered by fora and message boards.

***Gap Analysis***

Looking over the set of services, it is possible to identify where there are open standards that can support service implementation. However, for the most part the specifications available only cover part of the solution; for example, while there is a Sequencing data specification, there is no specification for a sequencing service.

Where services have some form of open standard, these are highlighted in light grey in the diagram on the next page. Where a complete set of open standards – including both data models and service definitions – is available, the service is highlighted in dark grey with white text.

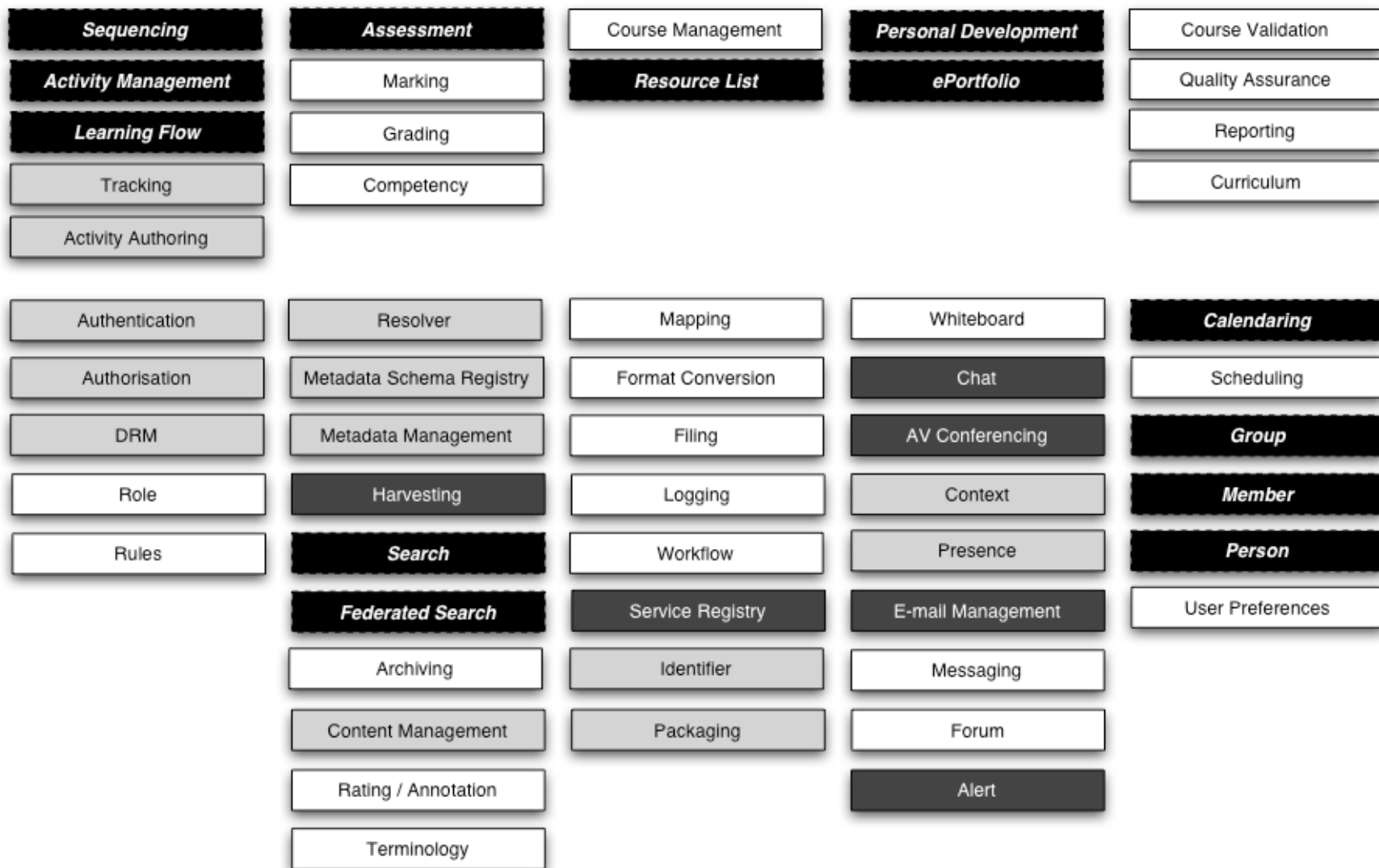
Note that some services that do not yet have any standards for data models are the subject of OKI OSIDs – Grading, Filing, Workflow, etc. However we have not shaded these services in this analysis as the OSIDs have not yet been submitted to an open standards process.

Sequencing	Assessment	Course Management	Personal Development	Course Validation
Activity Management	Marking	Resource List	ePortfolio	Quality Assurance
Learning Flow	Grading			Reporting
Tracking	Competency			Curriculum
Activity Authoring				
Authentication	Resolver	Mapping	Whiteboard	Calendaring
Authorisation	Metadata Schema Registry	Format Conversion	Chat	Scheduling
DRM	Metadata Management	Filing	AV Conferencing	Group
Role	Harvesting	Logging	Context	Member
Rules	Search	Workflow	Presence	Person
	Federated Search	Service Registry	E-mail Management	User Preferences
	Archiving	Identifier	Messaging	
	Content Management	Packaging	Forum	
	Rating / Annotation		Alert	
	Terminology			

## ***Current work by JISC***

JISC is currently funding a range of projects aimed at exploring areas of functionality defined in the framework, specifically by developing toolkits and libraries, and exploring the use of open standards. The diagram on the following page highlights the areas covered by the current set of active projects, which will be completed in October. There will be at least two subsequent waves of development projects building on this initial work and expanding the range of services addressed.

Services addressed by current projects have black backgrounds, white italicized text, and a dashed border.



## ***Definitions of individual services***

Each service in the Framework consists of:

- A simple description of the functions the service supports
- A scope and definition statement which provides more detailed requirements
- A specification matrix, which addresses:
  - o Data Representation
  - o Interface
  - o Service Definition
- Files, such as guides on best practice and patterns for implementation
- Links to projects that investigate use of this service
- A discussion forum

## ***Example: Group service***

### **Functional definition**

The key functions for a Group management service are:

- Creating one or more groups
- Reading one or more groups
- Updating one or more groups
- Deleting one or more groups
- Querying for groups of a certain type or criteria

Additional functions

- Querying the types of Group supported by the service

### **Abstract Model**

The models for the Group service are those provided by the IMS Enterprise Services specification Information Model document for Group Management and Groups Management [reference]. This document provides abstract models in UML for both behaviour and data.

### **XML Data Representation**

The Group and Groups UML static structure models from the IMS Enterprise Services specification Information Model document for Group Management and Groups Management [reference] should be used to represent Group instances, and Groups as collections, respectively. Where such data is exchanged using XML, the XML schema binding for these classes defined by IMS should be used.

### **Interface Specification**

The Group Management and Groups Management UML Interface models from the IMS Enterprise Services specification Information Model document for Group Management and Groups Management [reference] should be used as the basis for defining any APIs to support a Group service.

## **Web Service Definition**

The IMS Enterprise Services Web Service bindings for Group Management and Groups Management [reference] should be used to implement a Group service as a Web Service using WSDL and SOAP.