IMS Abstract Framework: A Review

**Colin Smythe, IMS
(csmythe@imsglobal.org)**

Since its incorporation the IMS has released a unique set of interoperability specifications within the eLearning technology community. During that period the various eLearning technology development activities have begun a slow convergence that has been the result of extensive work behind the scenes by IMS and other organizations. It is now apparent that the current set of IMS activities and processes need to evolve to tackle the next series of technical issues. As a consequence, IMS has produced its Abstract Framework to be the context within which:

- The next set of IMS specifications are developed;
- The migration from the current to the new specifications can be defined and managed;
- The relationship between IMS and non-IMS specifications can be explained and clearly demonstrated.

The adoption of the IMS Abstract Framework will be based upon the creation of suitable Domain Profiles. Domain profiling is the process that is undertaken to define which specifications and the detailed usage of the data objects within each specification are to be adopted to provide a particular solution.

Content

- **The context**
  - **What**
  - **Why**
  - **How**
- **The framework**
  - **Layers**
  - **Services**
  - **Components**
- **What next**

The IAF is a 'living' document i.e. it is not archival in nature. Our ideas for various parts of the IAF are constantly being developed and so the information contained herein should always be considered in that context. The IAF white paper is one of a set of closely related documents, the others being:

- The IMS Abstract Framework: Glossary – the definitions of the key terms used throughout the IAF and the associated specifications;

- The IMS Abstract Framework: Applications, Services & Components – the identification of the set of applications and services and their corresponding implementation components which can be used to support eLearning system interoperability (the separation of the detailed descriptions of the applications, services and components allows the details of this white paper to focus on the abstract representation itself);

- The IMS Use-case Portfolio – the collection and collation of the core set of use-cases that reflect the interoperability needs within eLearning systems (work on the collection of these use-cases is underway);

- The IMS Specification Development Methods & Best Practices – the identification of the methods and best practices that must be used when developing and documenting IMS specifications.

What is the IAF ?

- **It is an abstract representation of the services that comprise e-learning systems in the broadest sense**
- **It describes how a designated set of data models can be manipulated by a designated set of behaviours**
- **It consists of:**
  - **Glossary**
  - **White paper overview**
  - **Application, Service & Component Descriptions**
- **It supports a range of implementations !!!**

The IMS Abstract Framework (IAF) is a device to enable the IMS to describe the context within which it will continue to develop its eLearning technology specifications. This framework is not an attempt to define the IMS architecture, rather it is mechanism to define the set of interfaces for which interoperability specifications will be needed by some application domain. These specifications may come from IMS or from other courses. Prioritization of their development will depend on both technical feasibility and market need. In the cases where IMS does not produce a particular specification, IMS will attempt to identify and recommend suitable specifications provided by other organizations. The IMS Abstract Framework is so named because:

- It is an abstract representation of the services and their interfaces that are used to construct an eLearning system in its broadest sense;

- It is focused on the support of distributed electronic learning systems;

- It is a framework that covers the possible range of eLearning architectures that could be constructed from the set of defined services and interfaces.

It is the intention of IMS that this Abstract Framework and the associated IMS specifications produced to realise the exchange of information between the identified services will be adopted in a manner suitable for a particular system requirement. This process is called the Application Profiling of the framework to define an architecture or reference model.

What Does the IAF Provide ?

- **A common language for the development of IMS specifications**
- **An explicit system context for individual IMS specifications**
- **A context for the joint development of specifications**
- **A model of how how it all fits together**

The IAF is not an attempt to define or specify THE definitive and complete set of services or implementation architectures for e-learning. It is intended to document the technical assumptions about general model, terminology, and set of behaviors and data structures which guide the development or recommendations of specifications by the IMS community.

How is the IAF to be used ?

- **Identify the services from the set that need to be specified**
- **Identify the services that do not need to be specified**
- **Identify all the components required to provide end-to-end support for a service**

5

The IAF provides the context for the development of the service based IMS specifications. The IMS specification development process uses the IAF to:

- Identify some of the eLearning components that are to be specified. Once these components have been specified then any appropriate amendments to the services identified in the IAF will also be made;

- Identify the components that will be adopted from other specification and standards activities. These components may be tailored to enhance their adoption for eLearning;

- Demonstrate the process by which the UML representations of the information model descriptions of the components are converted to their XML binding equivalents;

- Demonstrate the ways in which the 'sea of components' is combined and profiled to support the specification of an eLearning architecture.

As with all of the IMS specifications there is a need to develop best practice for each of the activities summarised above. The IAF will be updated on a regular basis, typically once every six months, and as apart of other activities e.g. development of the content framework. While these updates will include important new information, the underlying principles of interoperable, component-based service definitions within a layered framework will not be changed.

Underlying Principles

- **Interoperability**
- **Service-oriented**
- **Component-based**
- **Layering**
- **Both behaviours & data models**
- **Binding neutral (XML, WSDL, Java, etc.)**
- **Borrow first; build last**

- Interoperability – the specifications are focussed on the exchange of information between systems. The specifications make no assumptions on how the data is managed within the communicating systems;

- Service-oriented – the exchange between the systems is to be defined in terms of the services being supplied by the collaboration of the systems. This service collaboration could take many forms as well as being based upon peer-to-peer and client-server techniques;

- Component-based – the set of services will be supplied as a 'sea of components' that can be mixed and matched to form a particular service. A single component may provide all or a sub-set of a service;

- Layering – the total set of services required to make an eLearning system will be modelled as a set of layers with each layer providing a clearly defined set of services;

- Behaviours and Data Models – a service will be defined in terms of its behaviours and data model. The behaviours will cause changes in the state of the data model and the state of the data model will only be altered as a result of a clearly defined behaviour;

- Multiple bindings – the information model for a specification (behaviour and data) can be supported through multiple bindings e.g. as Java, XML, web services, etc;

- Adoption – new specifications will only be created as required. Whenever possible, appropriate specifications will be adopted from wherever.
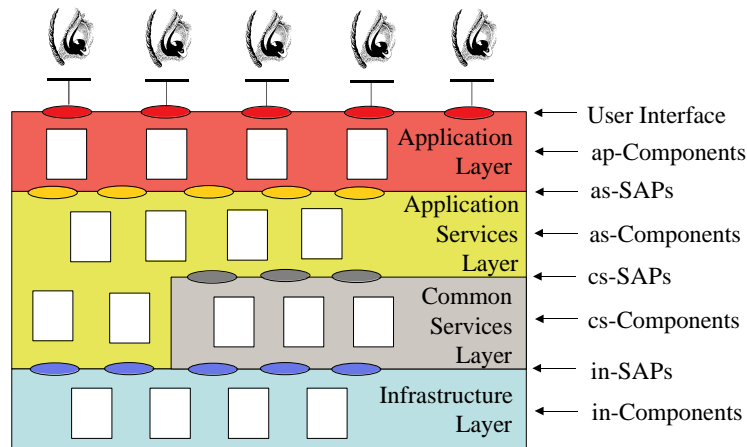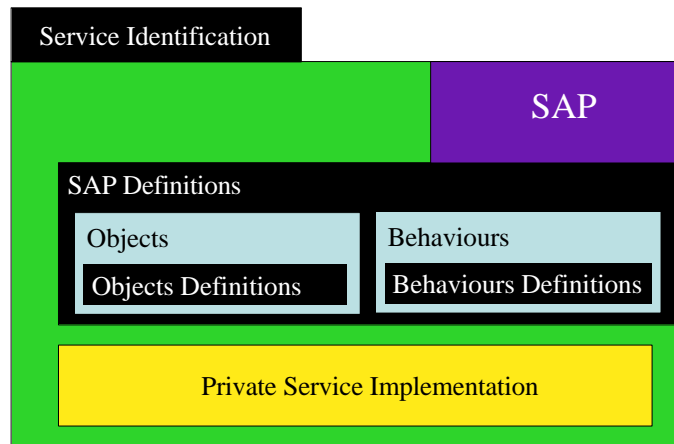
The abstract framework can be represented as a layered model, as shown above, consisting of four layers:

- Application layer – this is a tool, system, agent, etc. that presents the appropriate application services to the user i.e. an application manages the user interface. The application may use one or more application services but whenever possible the system composition should be hidden from the user;

- Application Services layer – a set of services that provide the required eLearning functionality to the applications. An application service may make use of one or more common services. Distributed application services communicate using via the Infrastructure Layer;

- Common Services layer – a set of services that are available to the application services. Common services may use other common services. Therefore, a common service is available to any other service;

- Infrastructure layer – this provides the end-to-end transaction and communications services for the application and common services.

Access to a service is through the appropriate Service Access Point (SAP). Each service has a single SAP. A Component may support one or more SAPs (in an object oriented representation, a SAP could be supported by one or more operators where the class is itself the definition of the service).

IAF Service Representation

Service Identification

SAP

SAP Definitions

Objects

Objects Definitions

Behaviours

Behaviours Definitions

Private Service Implementation

8

One of the design principles for the IAF is the adoption of service abstraction to describe the appropriate e-learning functionality. The service is hidden behind a SAP and can only be accessed by using the SAP (an API could be one way in which the SAP is actually implemented). However, even though the service provision is hidden behind the SAP, an implementation requires that the behavioural capabilities of the service be defined. In the diagram we have a schematic representation of a service, namely:

- The service has a clearly defined service access point. Each service has only one SAP. The SAP is now defined in terms of its constituent objects and behaviours;

- The SAP may consist of one or more objects and each object will, in general, will have more than one operation. Each object is defined using a class definition and consists of a group of attributes and operations. The operation describe how the state of the attributes may be changed. The set of behaviours permitted for each class must also be defined;

- The 'Private Service Implementation' is beyond the scope of the IAF. The only requirements on this are that it provides all of the appropriate behavioural characteristics of the service and nothing more.

This approach means that every service (application and common) must be defined using this form of abstraction. In many cases the services interact with each other e.g. an application service will use a common service. This interaction is reflected by the service invoking the SAP of the required service.
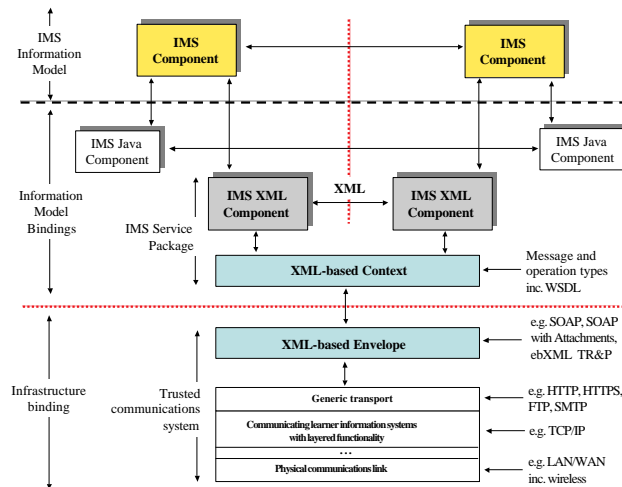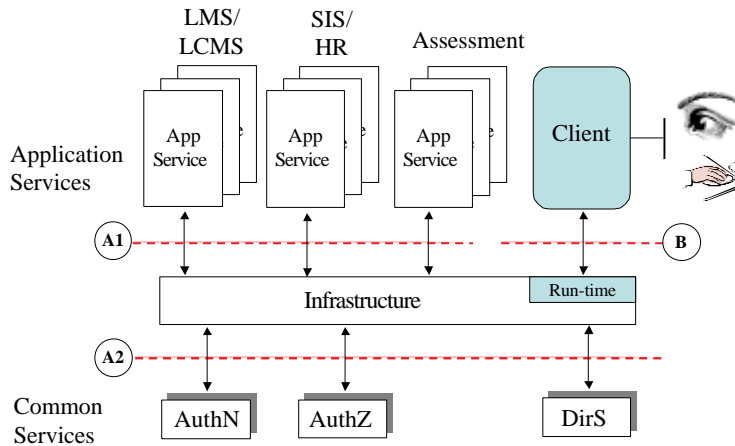
8

The IMS specifications are focused on data exchange interoperability. To this end they define a data model of the information to be exchanged and a behavioural model that encapsulates the data model and constrains the way in which the data can be manipulated. An IMS information model is the manifestation of this behavioural and data description and it will consist of one or more IMS Components. These components can then be realised in a variety of ways; the defined IMS method is as an XML-binding. As such, this XML-binding describes the exchange in the form of XML structures:

- IMS XML Components – the application and common services components that are combined to create the e-learning system required. These components exchange information in the form of XML instances;

- XML-based Context – the system components and their specific XML messages are mapped onto a common XML messaging infrastructure that is designed to support the required end-to-end services;

- XML-based Envelope – the common XML messaging system can be supported using several types of XML envelope encapsulation e.g. SOAP, SOAP with attachments, eb-XML, etc.

- Generic transport – the envelope is then transported across the network using an appropriate end-to-end protocol e.g. SMTP, FTP, HTTP, etc;

- Communications network – this is the actual data network that is used to physically transport the data from one system to another. This will almost certainly be based upon the ubiquitous TCP/IP.

Service Components

LMS/
LCMS   SIS/
HR   Assessment

Application
Services

App
Service   App
Service   App
Service   Client

A1 -------- B

Infrastructure   Run-time

A2

Common
Services   AuthN   AuthZ   DirS

In a distributed implementation of this layered framework, as typified in a webs services environment, the interaction between the services would as shown above. In this interaction framework there are three categories of interface that must be supported by the Infrastructure layer namely:

- The Application Services interface (A1) – this interface is used to provide interoperability between common application services e.g. Enterprise-to-Enterprise systems;

- The Common Services interface (A2) – this interface is used to provide interoperability with the set of common services that are made available to the application specific services e.g. authentication and authorization;

- The run-time interface (B) – this interface is used to interconnect the client's run-time application with the remote service provider.

There are two types of interaction behaviour that need to be specified to ensure interoperability, namely:

- Message passing – where information is exchanged between systems using some form or another of message exchange. The content and sequence of the messages define the expected behaviour;

- Run-time – where the end systems have to reliably operate on information using some predefined algorithm. The data will determine the outcomes of the algorithms but the behaviour is well defined for all possible outcomes.

What Next ?

- **Agree complementary activities with OKI, SIF, ADL, etc.**
- **Refine the Common Services using available definitions e.g. OSIDs**
- **Incorporate learner, learning & content models**
- **Further specification work in the Infrastructure Layer e.g. WSDL**
- **Extend the Glossary**

It is not the objective of IMS to continually refine the IAF. The IAF is a means to an end and not the 'end' itself. However, there are several key activities to be undertaken in the near future:

- Complementary and convergent activities need to be undertaken with other e-learning specification and standardization organizations;
- The IAF Common Services need to be more completely defined so that these can be uniformly used across the IMS and non-IMS specifications;
- The IAF must be extended to incorporate the learner, learning activities and models, and to define an appropriate content model;
- Web service support must be defined as one form in which the infrastructure layer can be implemented;
- The glossary needs to be extended as the set of IMS specifications grows and as convergence between the different specification and standard development activities converge.

Summary

- **Now available on Public web-site**
  - **White paper**
  - **Applications, Services & Components**
  - **Glossary**
  - **http://www.imsglobal.org/af/index.cfm**
- **The IAF is not an 'architecture'**
  - **It is a means to the end, not the end**
  - **There is no reference implementation**

The IMS Abstract Framework is available through the IMS Public web-site. All organizations interested and active in the domain of e-learning are encouraged to download and review the IAF document set.

All comments and questions concerning the IAF should be forwarded to:

Colin Smythe

IMS Chief Specification Strategist

Email: csmythe@imsglobal.org