IMS Enterprise Services Specification

Version 1.0 Final Specification

IPR and Distribution Notices

Recipients of this document are requested to submit, with their comments, notification of any relevant patent claims or other intellectual property rights of which they may be aware that might be infringed by any implementation of the specification set forth in this document, and to provide supporting documentation.

IMS takes no position regarding the validity or scope of any intellectual property or other rights that might be claimed to pertain to the implementation or use of the technology described in this document or the extent to which any license under such rights might or might not be available; neither does it represent that it has made any effort to identify any such rights. Information on IMS's procedures with respect to rights in IMS specifications can be found at the IMS Intellectual Property Rights web page: http://www.imsglobal.org/ipr/imsipr_policyFinal.pdf.

Copyright © 2004 IMS Global Learning Consortium. All Rights Reserved.

If you wish to copy or distribute this document, you must complete a valid Registered User license registration with IMS and receive an email from IMS granting the license to distribute the specification. To register, follow the instructions on the IMS website: http://www.imsglobal.org/specificationdownload.cfm.

This document may be copied and furnished to others by Registered Users who have registered on the IMS website provided that the above copyright notice and this paragraph are included on all such copies. However, this document itself may not be modified in any way, such as by removing the copyright notice or references to IMS, except as needed for the purpose of developing IMS specifications, under the auspices of a chartered IMS project group.

Use of this specification to develop products or services is governed by the license with IMS found on the IMS website: http://www.imsglobal.org/license.html.

The limited permissions granted above are perpetual and will not be revoked by IMS or its successors or assigns.

THIS SPECIFICATION IS BEING OFFERED WITHOUT ANY WARRANTY WHATSOEVER, AND IN PARTICULAR, ANY WARRANTY OF NONINFRINGEMENT IS EXPRESSLY DISCLAIMED. ANY USE OF THIS SPECIFICATION SHALL BE MADE ENTIRELY AT THE IMPLEMENTER'S OWN RISK, AND NEITHER THE CONSORTIUM, NOR ANY OF ITS MEMBERS OR SUBMITTERS, SHALL HAVE ANY LIABILITY WHATSOEVER TO ANY IMPLEMENTER OR THIRD PARTY FOR ANY DAMAGES OF ANY NATURE WHATSOEVER, DIRECTLY OR INDIRECTLY, ARISING FROM THE USE OF THIS SPECIFICATION.

Copyright © 2004 by IMS Global Learning Consortium, Inc.

All Rights Reserved.

The IMS Logo is a registered trademark of IMS Global Learning Consortium, Inc.

Document Name: IMS Enterprise Services Specification

Date: 11 June 2004

Table of Contents

1.   Introduction

1.1   Enterprise Services Overview

The Enterprise Services specification is the definition of how systems manage the exchange of information that describes people, groups and memberships within the context of learning. The Enterprise Services specification is constructed following the recommendations documented in the IMS Abstract Framework (IAF) [AbsGloss, 03], [AbsASC, 03], [AbsWhite, 03]. This means that this specification is based upon the concepts of:

*   Interoperability – Enterprise Services focuses on the exchange of information between Enterprise systems. There are no assumptions in the specification on how the data is managed within the Enterprise systems;

*   Service-oriented – Enterprise Services defines the exchange of information in terms of the services being supplied by the collaboration of the systems. This takes the form of Person Management Services, Group Management Services, and Membership Management Services;

*   Component-based – the set of services will be supplied such that they can be combined to form a range of services. The Person Management Services, Group Management Services, and Membership Management Services can be combined to provide other services and the Enterprise Service will have other services added to it in later releases;

*   Layering – the Enterprise Service and its constituent services (Person, Group, and Membership) are part of the Application Services layer;

*   Behaviors and Data Models – the Enterprise Services are defined in terms of their behaviors and data models. The behaviors 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 behavior;

*   Multiple Bindings – the Enterprise Services information model is to be defined using the Unified Modelling Language (UML). This enables reliable mapping of the information model into a range of different bindings. The bindings of immediate importance are to the Web Services Description Language (WSDL);

*   Adoption – the Enterprise Services are based upon the original Enterprise specification data model. While there are significant changes the underlying data model has been maintained and the core Person, Group, and Membership structures remain.

1.2   Scope and Context

This document is the IMS Enterprise Services Specification v1.0 and as such it is used as the basis for the development of the following documents:

   a)   IMS Enterprise Services Conformance Specification v1.0 [EntServices, 04b] – the definition of the conformance criteria that must be followed by systems that wish to claim compliance to the Enterprise Services Information model;

   b)   IMS Enterprise Services with WSDL Binding Best Practices & Implementation Guide v1.0 [EntServices, 03c] – the recommended best practices to be adopted when implementing the Enterprise Services using a webs services binding;

The core uses cases for the Enterprise Services are described in [EntServices, 04a]. The Enterprise Services specification supersedes the original Enterprise specifications:

   a)   IMS Enterprise Information Model Final Specification v1.1 [Enterprise, 02a].

   b)   IMS Enterprise XML Binding Final Specification v1.1 [Enterprise, 02b];

   c)   IMS Enterprise Services Best Practice and Implementation Guide Final Specification v1.0 [Enterprise, 02c].

This specification is based upon the aggregation of the Person Management, Group Management and Membership Management Services specifications, namely:

   a)   IMS Person Management Services Information Model v1.0 [PersonServices, 04a];

b)   IMS Person Management Services WSDL Binding v1.0 [PersonServices, 04b];

c)   IMS Group Management Services Information Model v1.0 [GroupServices, 04a];

d)   IMS Group Management Services WSDL Binding v1.0 [GroupServices, 04b];

e)   IMS Membership Management Services Information Model v1.0 [MemberServices, 04a];

f)   IMS Membership Management Services WSDL Binding v1.0 [MemberServices, 04b].

This information model defines the Enterprise Service Abstract Application Programming Interface (a-API). The original Enterprise specification was based upon the description of the data model for the information to be exchanged between communicating enterprise systems. The Enterprise Services specification extends this work by adding a series of behavioral models that define how the data models are to be manipulated. These behavioral models are described using UML and the syntax adopted for the description of the UML is given in the IMS Specifications, Methods and Best Practices document [SpecDev, 03].

The Enterprise Services Specification v1.0 is to be implemented using a Web Services infrastructure based upon a SOAP/HTTP transport mechanism. These web service bindings are detailed in the corresponding service binding documents [PersonServices, 04b], [GroupServices, 04b], [MemberServices, 04b].

1.3   Nomenclature

API	Application Programming Interface
a-API	Abstract Application Programming Interface
CRUD	Create, Read, Update and Delete
HR	Human Resources
HRMS	Human Resources Management System
http	HyperText Transfer Protocol
IAF	IMS Abstract Framework
LMS	Learning Management System
LibMS	Library Management System
MIS	Management Information System
SOAP	Simple Object Access Protocol
SIF	Schools Interoperability Framework
SIS	Student Information System
TMS	Training Management System
UML	Unified Modelling Language
URL	Universal Resource Locator
VLE	Virtual Learning Environment
W3C	World Wide Web Consortium
WSDL	Web Services Description Language
XML	Extensible Mark-up Language

1.4   References

[AbsASCs, 03]   *IMS Abstract Framework: Applications, Services & Components v1.0*, C.Smythe, <u>IMS Global Learning Consortium, Inc.</u>, July 2003.

[AbsGloss, 03]	*IMS Abstract Framework: Glossary v1.0*, C.Smythe, IMS Global Learning Consortium, Inc., July 2003.
[AbsWhite, 03]	*IMS Abstract Framework: White Paper v1.0*, C.Smythe, IMS Global Learning Consortium, Inc., July 2003.
[Cockburn, 01]	*Writing Effective Use-case*, A.Cockburn, Addison-Wesley, 2001, ISBN 0-201-70225-8.
[Enterprise, 02a]	*IMS Enterprise Information Model v1.1*, G.Collier, C.Etesse, W.Veres and C.Smythe, IMS Global Learning Consortium, Inc., July 2002.
[Enterprise, 02b]	*IMS Enterprise XML Binding v1.1*, G.Collier, C.Etesse, W.Veres and C.Smythe, IMS Global Learning Consortium, Inc., July 2002.
[Enterprise, 02c]	*IMS Enterprise Best Practice & Implementation Guide v1.1*, G.Collier, C.Etesse, W.Veres and C.Smythe, IMS Global Learning Consortium, Inc., July 2002.
[EntServices, 04a]	*IMS Enterprise Services Core Use Cases Description v1.0*, C.Smythe and C.Vento, IMS Global Learning Consortium, Inc., June 2004.
[EntServices, 04b]	*IMS Enterprise Services Conformance Specification v1.0*, C.Smythe and C.Vento, IMS Global Learning Consortium, Inc., June 2004.
[EntServices, 04c]	*IMS Enterprise Services with WSDL Binding Best Practices and Implementation Guide v1.0*, C.Smythe and C.Vento, IMS Global Learning Consortium, Inc., June 2004.
[GroupServices, 04a]	*IMS Group Management Services Information Model v1.0*, C.Smythe and C.Vento, IMS Global Learning Consortium, Inc., June 2004.
[GroupServices, 04b]	*IMS Group Management Services WSDL Binding v1.0*, C.Smythe and C.Vento, IMS Global Learning Consortium, Inc., June 2004.
[MemberServices, 04a]	*IMS Member Management Services Information Model v1.0*, C.Smythe and C.Vento, IMS Global Learning Consortium, Inc., June 2004.
[MemberServices, 04b]	*IMS Member Management Services WSDL Binding v1.0*, C.Smythe and C.Vento, IMS Global Learning Consortium, Inc., June 2004.
[PersonServices, 04a]	*IMS Person Management Services Information Model v1.0*, C.Smythe and C.Vento, IMS Global Learning Consortium, Inc., June 2004.
[PersonServices, 04b]	*IMS Person Management Services WSDL Binding v1.0*, C.Smythe and C.Vento, IMS Global Learning Consortium, Inc., June 2004.
[SpecDev, 03]	*IMS Specification Development Methods and Best Practices Draft 5.0*, C.Smythe, IMS Global Learning Consortium, Inc., August 2003.

2.   Core Use Case Descriptions

The following use cases[1] are the core subset adopted to represent key requirements for this version of the Enterprise Services (the full description for each of these use cases is given in [EntServices, 04a]):

- Person management service
  - Create person (Person01/01)
  - Read person (Person01/02)
  - Update person (Person01/03)
  - Delete person (Person01/04);
- Group management service
  - Create group (Group01/01)
  - Read group (Group01/02)
  - Update group (Group01/03)
  - Delete group (Group01/04)
  - Delete group relationship (Group01/05);
- Membership management service
  - Create membership (Membership01/01)
  - Read membership (Membership01/02)
  - Update membership (Membership01/03)
  - Delete membership (Membership01/04)
  - Read a person's memberships (Membership01/05)
  - Read a group's memberships (Membership01/06).

The Enterprise Services core use case description document [EntServices, 04a] contains many use cases that are not supported by the v1.0. Support for those use cases will be addressed in later versions.

---

1.   Note that each use-case will be supported by one or more behaviors in the corresponding specification. For example, the 'Read Person' use case in the Person Management Service is supported by the 'readPerson()', 'readPersons()' and readPersonsForGroup()' behaviors.

3.   Enterprise Service Description

3.1   An Abstract Representation

It is important to remember that this document is describing the underlying information model in terms of the abstract API. The manner in which this abstract representation is visualized is not intended to dictate the implementation form of an Enterprise System. The breakdown of the service into its component services and the further breakdown of each service into the interface classes used is a convenient way to document the set of behaviors. The internal organization of an implementation of the full abstract API is beyond the scope of this specification. The only constraint is that the external behavior of the abstract API complies with this specification. This means that a .NET, Java, etc. physical implementation of this abstract API does not have to represent the functionality using the same breakdown of operations/methods. This physical implementation is not subject to the conformance specification.

It is important to note that the UML representation of the interfaces is used to help develop and document the Enterprise Services information Model. It is not a requirement for an implementation to implement this interface as defined, i.e., to use the same parameters, etc. Conformance against this specification will be confirmed by inspecting the appropriate binding of the information model and ensuring that the relevant information is present and that different sequences of activity result in the predicted and mandated behavior. It is essential that the behaviors described by each of the operations are fully supported and it is also essential that the behaviors described by different sequences are also maintained.

3.2   Enterprise Service Architecture Model

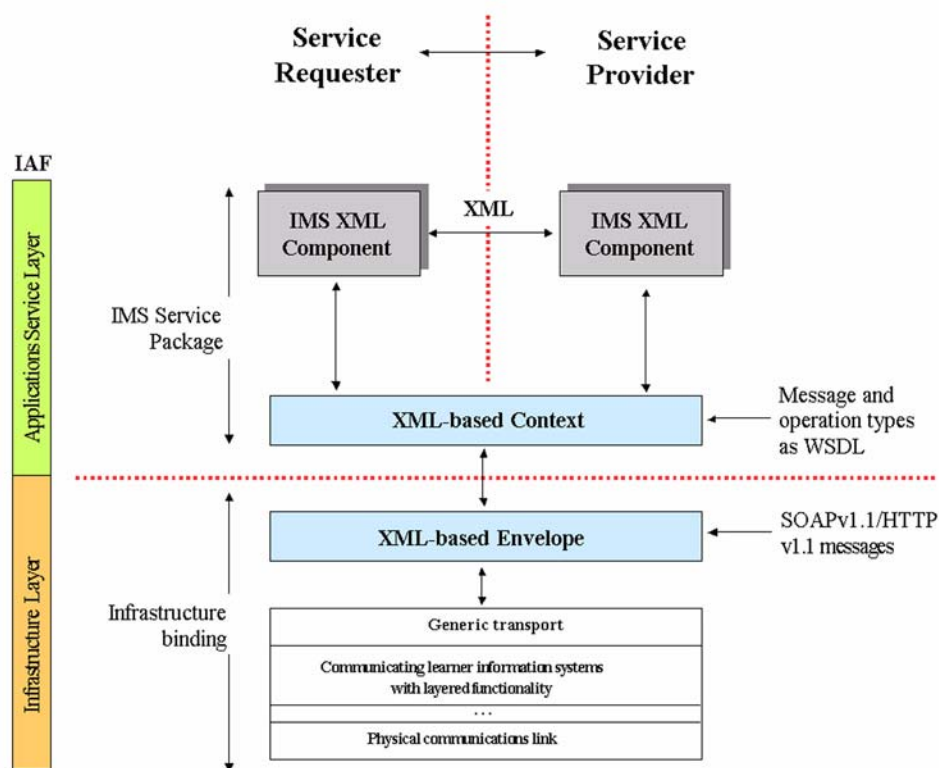The basic abstract architectural model for the Enterprise Services specification is shown in Figure 3.1.

Figure 3.1 Enterprise Services architecture model.

In this architecture the scope of the IMS enterprise services specification is shown as the dotted line. The scope of the interoperability is the data and behavioural models of the objects being exchanged. Figure 3.1 is an abstract model showing the interfaces that are under specification (see the IMS Abstract Framework [AbsWhite, 03]; it is not a representation of how an Enterprise System should be constructed. IMS Service Package is the definition of the IMS Enterprise Services resulting in a WSDL binding. The WSDL binding is then mapped onto a SOAP/http Infrastructure Layer.

The IMS Enterprise Services specification contains a very simple abstract messaging model that is assumed to underlie the exchange of the data between the communicating Enterprise systems. The basic data exchange mechanism is shown in Figure 3.2 in which the 'source' and 'target' Enterprise systems exchange an IMS Enterprise Services 'message' using a 'Request/Response' transaction. This abstract messaging representation is adopted because the Enterprise System architecture is based upon loosely coupled system and the primary IMS binding of this information model is based upon the exchange of XML documents/messages.

It is important to remember that the structure of the exchanged information has NO bearing on how the same information is contained within the 'source' and 'target' Enterprise systems. It is simply an exchange interoperability representation of the data (the information that crosses the dotted line in Figure 3.2).
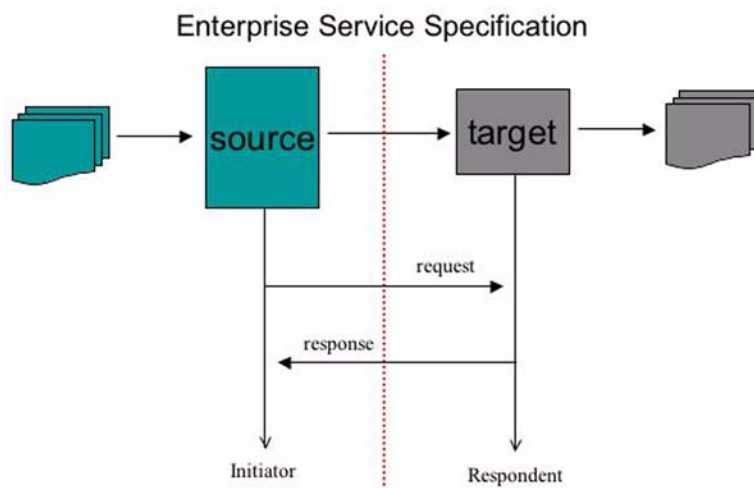

Figure 3.2 Enterprise Services abstract information exchange model.

The behavioral descriptions will describe:

- The data that is to be exchanged between the communicating Enterprise Systems;
- The operations that can be invoked to support the exchange of the data between the communicating Enterprise Systems;
- The permitted sequence of operations that can be used to support the exchange of data between the communicating Enterprise Systems.

3.3   Enterprise Services Domain Model

A schematic representation of the core objects exchanged using the IMS Enterprise Services specification is given in Figure 3.3. Only those services that are shown in grey are defined in this information model. The core enterprise services are:

- Person management service – the management of information about individuals who are undertaking some form of study and/or group related activity e.g., they are members of a particular course. The person record is designed to be a data model for all of the personal information to be exchanged about an individual;

- Group management service – the management of information about a collection of objects related to learning activities or individuals. The group is a generic container used to define any set of related activities e.g., a tutor group, a class, a curriculum, etc. There is no restriction on how the Group and sub-group structures can be used with respect to containing other groups, persons, etc.;

- Membership management service – the management of information about the membership structure is used to define the members of a Group. A member can be a Person or another Group. A Group or Person can be a member of any number of groups. The Group and the member are identified using their sourcedids.

Two other Enterprise Services have been identified: Gradebook management service and Course catalog management service. This will not be defined as part of the V1.0 information model but may be included in later releases.

The underlying set of services used by the core enterprise services composition application services (as defined in the Abstract Framework):

- IAF common services – provision of services such as authentication authorization, etc. In the case of the Enterprise Services the core common services include the 'StatusInfo and 'Identifier';

- IAF infrastructure – provision of the end-to-end communications services.
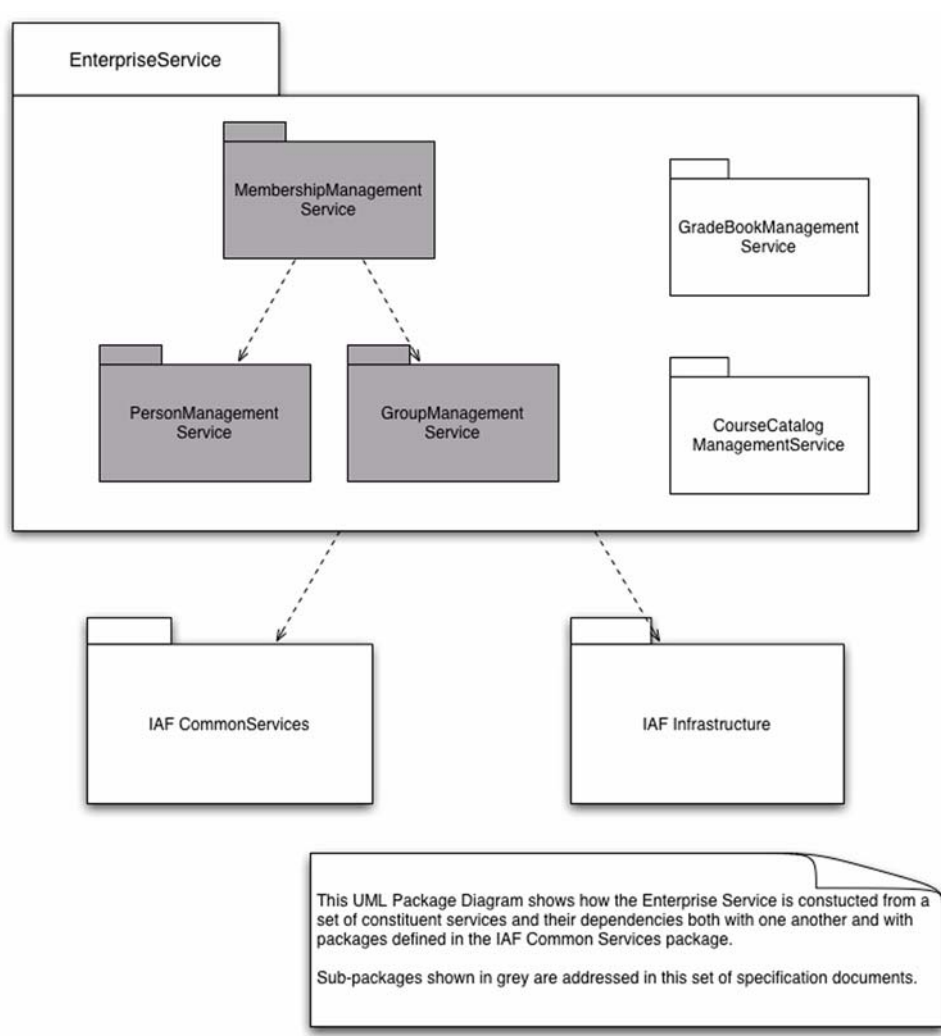

Figure 3.3 Enterprise Services domain model.

An underlying assumption in a message-based system is that the sequence of messages will, in general, refer to the same objects at different moments in time e.g., the 'creation' of a Person, the 'updating' of a person and eventually the 'deletion' of the Person record. This means that the objects must have a unique identifier and that this address/label is

unique between the communicating systems (an object may have more that one identifier even between the same two systems). The IMS Enterprise specification calls these identifiers the 'sourcedId' of the object and it consists of a 'source' (globally unique across all the Enterprise systems) and an 'id' (unique within the source Enterprise system).
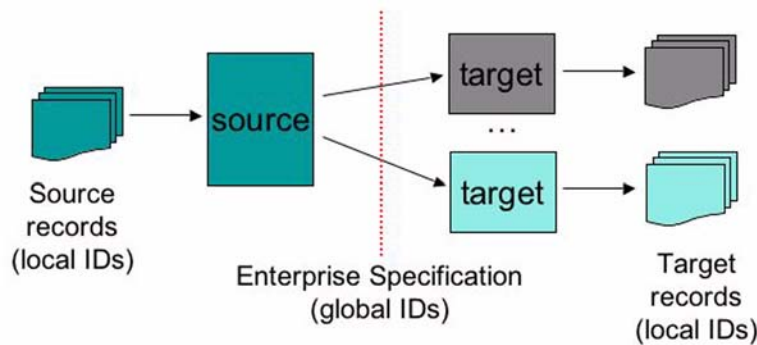

Figure 3.4 Enterprise Services simple object addressing schema.

The usage of the 'sourcedid' for labelling the objects being exchanged gives rise to the scenario shown in Figure 3.4. The consequence of this approach is that any object will have multiple identifiers depending on which part of the system is being considered, i.e.:

- The 'sourcedid' is the exchange identifier and will be unique to a particular source system;

- The local identifier within the source system. This could be a database record number, etc. The source system must maintain the local identifier mapping table between the local identifier and the 'sourcedid';

- The local identifier within the target systems. In general the local identifier in each target system will be different. It is the responsibility of the target systems to maintain the local mapping table between the 'sourcedid' and their local identifier.

Within the Person management, Group Management and Membership management services the 'sourcedid' is defined as a flat string Identifier. For compatibility with the Enterprise v1.1 and earlier specifications the 'sourcedid' will be mapped onto the flat string as: 'source:id,' i.e., the colon (:) is used to delimit the 'source' and the 'id' within the flat string.

4.   Person Management Service

The PersonManagementService class is used to model the service responsible for manipulating information about people. The PersonManagementService package is shown in Figure 4.1.
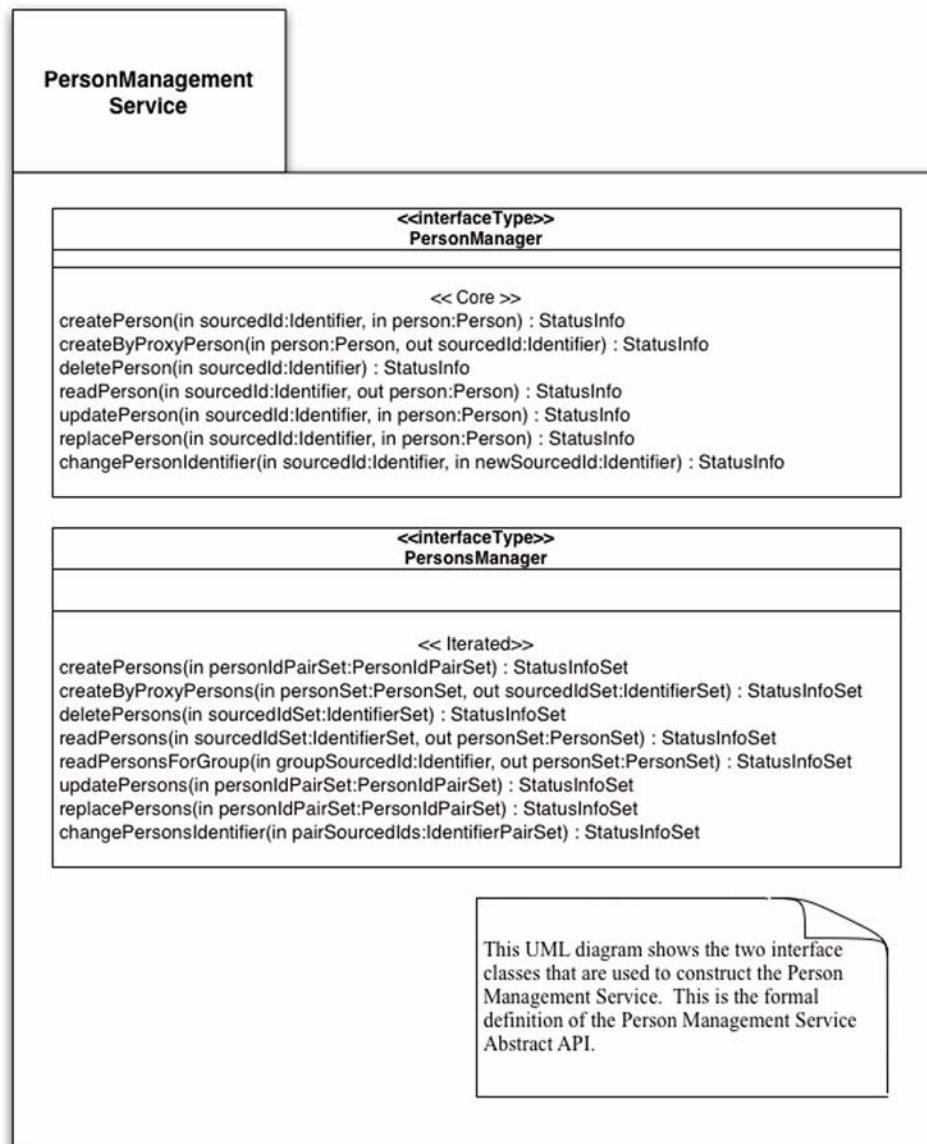

Figure 4.1 Person management service definition.

The PersonManagementService is split into two interface classes: PersonManager that supports the manipulation of a single Person object; PersonsManager that supports the manipulation of two or more Person objects bound in a single transaction. The manipulation of multiple Person objects can also be supported by the iteration over the PersonManager however this will result in multiple independent transactions. The data-types used to support the PersonsManager class are derived as sets of the equivalent data-types used for the PersonManager class. The service operations are summarized in Table 4.1.

Table 4.1 Summary of PersonManagerService operations.

Operation	Description
createPerson	To request the creation of a populated 'person' record on the target system and the source is responsible for the allocation of the unique identifier.
createByProxyPerson	To request the creation of a populated 'person' record on the target system and the target is responsible for the allocation of the unique identifier.
deletePerson	To request the deletion of a 'person' record. The 'person' record is deleted and all of its associated relationships.
readPerson	To read the full contents of the identified 'person' record. The target must return all of the data it has for the identified 'person' record.
updatePerson	To write new content into the identified 'person' record. The target must write the new data into the 'person' record. This is an additive operation.
replacePerson	To replace the content of the identified 'person' record. The target must write the new data into the 'person' record. This is a destructive write-over of all of the original information.
changePersonIdentifier	To change the sourcedId of the 'person' record. The completion of this operation will result in later actions using the original sourcedId reporting an unknown identifier status.
createPersons	To request the creation of a set of populated 'person' records on the target system. The source is responsible for the allocation of the unique identifiers.
createByProxyPersons	To request the creation of a set of populated 'person' records on the target system. The target is responsible for the allocation of the unique identifiers.
deletePersons	To request the deletion of a set of 'person' record. The 'person' records and all their associated relationships are deleted.
readPersons	To read the full contents of the set of identified 'person' records. The target must return all of the data it has for each of the identified 'person' records.
readPersonsForGroup	To retrieve the 'person' records for a particular Group. This returns the person record for every person that is a member of the Group i.e., for whom a membership record in the Group exists.
updatePersons	To write new content into the set of identified 'person' records. The target must write the new data into each of the 'person' records. These are additive operations.
replacePersons	To replace the content of a set of identified 'person' records. The target must write the new data into the 'person' records. These are destructive write-overs of all of the original information.
changePersonsIdentifier	To change the sourcedId of the set of 'person' records. The completion of this operation will result in later actions using the original sourcedId reporting an unknown identifier status.

5.   Group Management Service

The GroupManagementService class is used to model the service responsible for manipulating information about groups. The GroupManagementService class diagram is shown in Figure 5.1.
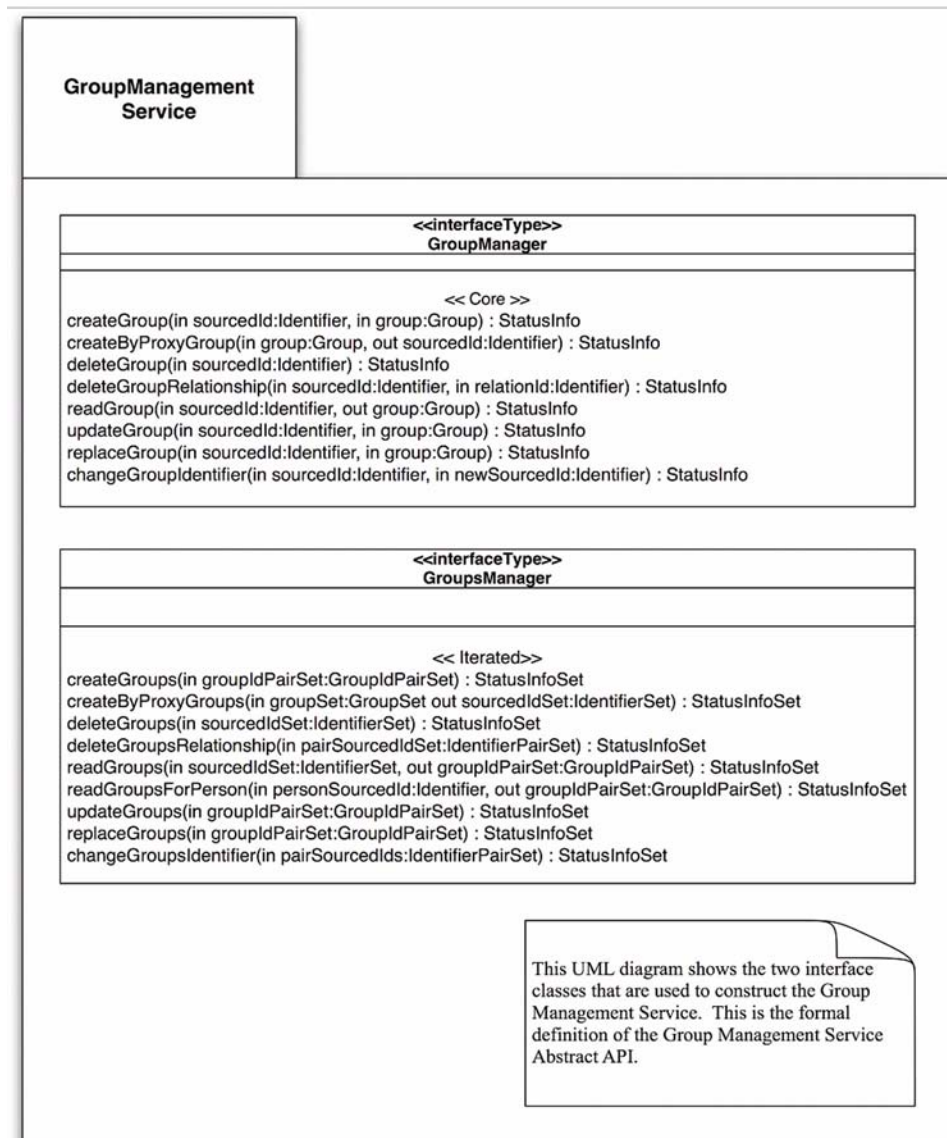

Figure 5.1 Group management service definition.

The Group Management Service is split into two interface classes: GroupManager that supports the manipulation of a single Group object; GroupsManager that supports the manipulation of two or more Group objects bound in a single transaction. The manipulation of multiple Group objects can also be supported by the iteration over the GroupManager however this will result in multiple independent transactions. The data-types used to support the GroupsManager class are derived as sets of the equivalent data-types used for the GroupManager class. The operations are summarized in Table 5.1.

Table 5.1 Summary of GroupManagerService operations.

Operation	Description
createGroup	To request the creation of a populated 'group' record on the target system and the source is responsible for the allocation of the unique identifier.
createByProxyGroup	To request the creation of a populated 'group' record on the target system and the target is responsible for the allocation of the unique identifier.
deleteGroup	To request the deletion of a 'group' record. The 'group' record is deleted and all of its associated relationships.
deleteGroupRelationship	To request the deletion of a relationship within a 'group' record. This does not delete the associated 'group' records.
readGroup	To read the full contents of the identified 'group' record. The target must return all of the data it has for the identified 'group' record.
updateGroup	To write new content into the identified 'group' record. The target must write the new data into the 'group' record. This is an additive operation.
replaceGroup	To replace the content of the identified 'group' record. The target must write the new data into the 'group' record. This is a destructive write-over of all of the original information.
changeGroupIdentifier	To change the sourcedId of the 'group' record. The completion of this operation will result in later actions using the original sourcedId reporting an unknown identifier status.
createGroups	To request the creation of a set of populated 'group' records on the target system and the source is responsible for the allocation of each of the unique identifiers.
createByProxyGroups	To request the creation of a set of populated 'group' records on the target system. The target is responsible for the allocation of the unique identifiers.
deleteGroups	To request the deletion of a set of 'group' record. The 'group' records and all their associated relationships are deleted.
deleteGroupsRelationship	To request the deletion of a set of relationships within the 'group' records. This does not delete the associated 'group' records.
readGroups	To read the full contents of the set of identified 'group' records. The target must return all of the data it has for each of the identified 'group' records.
readGroupsForPerson	To retrieve the 'group' records for a particular Person. This returns the set of group records of which the person i.e., for whom a membership record in the Group exists.
updateGroups	To write new content into the set of identified 'group' records. The target must write the new data into each of the 'group' records. These are additive operations.
replaceGroups	To replace the content of a set of identified 'group' records. The target must write the new data into the 'group' records. These are destructive write-overs of all of the original information.
changeGroupsIdentifier	To change the sourcedId of the 'group' record. The completion of this operation will result in later actions using the original sourcedId reporting an unknown identifier status.

6. Membership Management Service

The MembershipManagementService class is used to model the service responsible for manipulating information about the membership of people or groups in groups. The MembershipManagementService class diagram is shown in Figure 6.1.
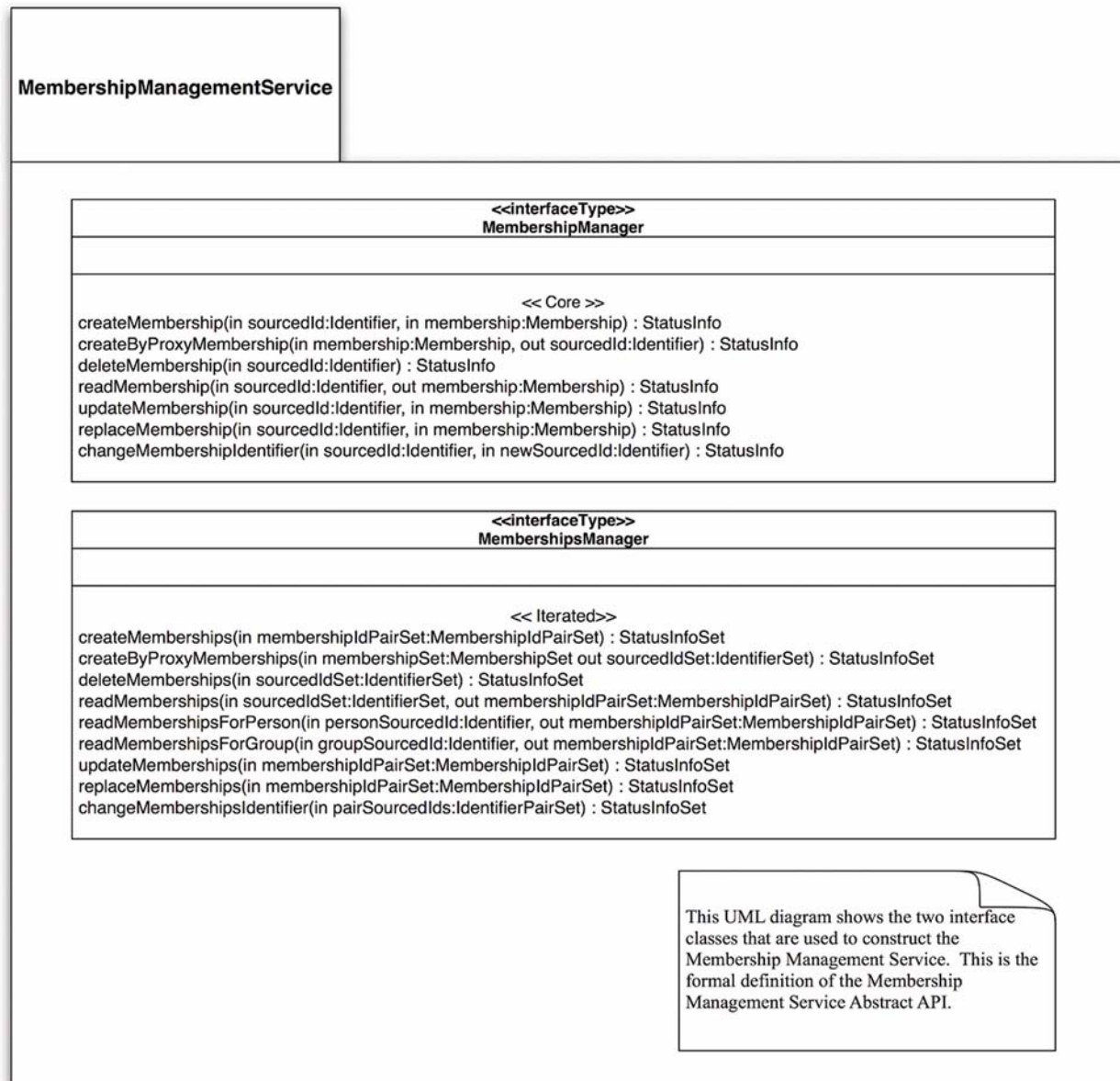

Figure 6.1 Membership management service definition.

The Membership Management Service is split into two interface classes: MembershipManager that supports the manipulation of a single Membership object; MembershipsManager that supports the manipulation of two or more Membership objects bound in a single transaction. The manipulation of multiple Membership objects can also be supported by the iteration over the Membership Manager however this will result in multiple independent transactions. The data-types used to support the MembershipsManager class are derived as sets of the equivalent data-types used for the Membership Manager class. The operations are summarized in Table 6.1.

Table 6.1 Summary of MembershipManagerService operations.

Operation	Description
createMembership	To request the creation of a populated 'membership' record on the target system and the source is responsible for the allocation of the unique identifier.
createByProxyMembership	To request the creation of a populated 'membership' record on the target system and the target is responsible for the allocation of the unique identifier.
deleteMembership	To request the deletion of a 'membership' record. The 'membership' record is deleted and all of its associated relationships.
readMembership	To read the full contents of the identified 'membership' record. The target must return all of the data it has for the identified 'membership' record.
updateMembership	To write new content into the identified 'membership' record. The target must write the new data into the 'person' record. This is an additive operation.
replaceMembership	To replace the content of the identified 'membership' record. The target must write the new data into the 'person' record. This is a destructive write-over of all of the original information.
changeMembershipIdentifier	To change the sourcedId of the 'membership' record. The completion of this operation will result in later actions using the original sourcedId reporting an unknown identifier status.
createMemberships	To request the creation of a set of populated 'membership' records on the target system. The source is responsible for the allocation of the unique identifiers.
createByProxyMemberships	To request the creation of a set of populated 'membership' records on the target system and the target is responsible for the allocation of each of the unique identifiers.
deleteMemberships	To request the deletion of a set of 'membership' record. The 'membership' records and all their associated relationships are deleted.
readMemberships	To read the full contents of the set of identified 'membership' records. The target must return all of the data it has for each of the identified 'membership' records.
readMembershipsForPerson	To retrieve all the 'membership' records for a particular Person. This returns every 'membership' record for the identified 'person' record.
readMembershipsForGroup	To retrieve all the 'membership' records for a particular Group. This returns every 'membership' record for the identified 'group' record.
updateMemberships	To write new content into the set of identified 'membership' records. The target must write the new data into each of the 'membership' records. These are additive operations.
replaceMemberships	To replace the content of a set of identified 'membership' records. The target must write the new data into the 'membership' records. These are destructive write-overs of all of the original information.
changeMembershipsIdentifier	To change the sourcedId of the set of 'membership' records. The completion of this operation will result in later actions using the original sourcedId reporting an unknown identifier status.

7.   Extending the Enterprise Service

7.1   Proprietary Extensions

The proprietary extensions of the specification are based upon two approaches:

    a)   The extension of the data models being manipulated by the current set of operations;

    b)   The inclusion of new operations to support new proprietary functionality.

It is NOT permitted to change the behavior of the current set of operations. Such changes MUST be supported by the creation of new operations.

Proprietary extensions must make use of the extensions facilities in the respectively components services. Refer to the appropriate specification documentation for the full details.

7.2   Further Work

The Enterprise Services Specification and is based on the composition of other clearly defined and functionally discrete services. This means that the addition of new services and the extension of the established services can be supported without compatibility problems. The areas of work that will be addressed in new versions of the Enterprise Services are:

*   Inclusion of the 'queryObject()' operation for each of the current and new services. This will enable each service to be queried to supply details of the objects that conform to the set of search criteria;

*   Inclusion of operations that support direct manipulation of some of the sub-objects within the main services e.g., the ability to add/delete/read/change a name without recourse to the manipulation of the full Person object. The corresponding operations for all aggregated Enterprise Services will be considered;

*   Inclusion of operations that support higher-level Enterprise Service business model requirements will be considered. Such requirements may require the definition of a set of Enterprise Services explicit because they require coordinated manipulation of more than one of the aggregated services, e.g., 'copyCourse' that creates a duplicate of the course and all its memberships;

*   Inclusion of the Grade-book Management Service. This will enable Enterprise Services to management grade books. This work will look at the Grade-book specification available from SIF;

*   Inclusion of the Course Catalog Management Service. This will enable Enterprise Services to management course catalogs for learning.

About This Document

Title	IMS Enterprise Services Specification
Editor	Colin Smythe (IMS)
Team Co-Lead	Chris Vento (WebCT Inc.)
Version	1.0
Version Date	11 June 2004
Status	**Final Specification**
Summary	This document presents the IMS Enterprise Services Specification. The original Enterprise specification was based upon the description of the data model for the information to be exchanged between communicating enterprise systems. The Enterprise Services specification extends this work by adding a series of behavioral models that define how the data models are to be manipulated. These behaviors are defined in a set of independent and aggregate services: Person Management Service, Group Management Services and Membership Management Services. This information is the definition of the abstract application-programming interface for the Enterprise Service. This version supersedes the IMS Enterprise v1.1 specifications.
Revision Information	11 June 2004
Purpose	This document has been approved by the IMS Technical Board and is made available for adoption.
Document Location	http://www.imsglobal.org/es/esv1p0/imses_specv1p0.html

To register any comments or questions about this specification, please visit:
http://www.imsglobal.org/developers/ims/imsforum/categories.cfm?catid=20

List of Contributors

The following individuals contributed to the development of this document:

Name	Organization	Name	Organization
Scott Baker	Oracle Inc.	Les Smith	SCT Inc.
Fred Beshears	UC Berkeley, USA	Colin Smythe	Dunelm Services Ltd.
Kerry Blinco	IMS Australia	Chris Vento	WebCT Inc.
Chris Etesse	Blackboard Inc.	Kimberley Voltero	WebCT Inc.
John Hallet	WebCT Inc.	Scott Wilson	JISC (CETIS), UK
Cathy Schroeder	Microsoft Inc.	Nathaniel Zinn	Blackboard Inc.

Revision History

Version No.	Release Date	Comments
Base Document 1.0	21 July 2003	The final approved Base Document for the IMS Enterprise Services Information Model.
Public Draft 1.0	12 January 2004	The final approved Public Draft Document for the IMS Enterprise Services Specification.
Final Specification 1.0	11 June 2004	This is the formal Final release of the IMS Enterprise Services specification.

Index