

internet2-courseID-eduCourse-200505.html

17 May 2005

Tom Barton, editor

The University of Chicago

Comments to: mace-courseid-comments at internet2 dot edu

eduCourse Data Model

1. Introduction

One of the goals of the MACE-courseID working group is to produce schema and recommended practices for identifying objects related to courses. This document presents the eduCourse data model, an abstract UML model that provides a frame of reference for the objectclasses under consideration by the working group and identifies their primary interrelationships and some of their leading attributes. Two schematic derivatives of the abstract model are also presented.

The eduCourse data model does not fully characterize all attributes of the objectclasses being identified. However, it should suffice to clarify the types of instances to which identifiers must be attached, and about which the working group will recommend good practices in separate documents.

A UML static structure diagram [3] graphically represents the eduCourse data model (figure 1). A narrative description of the depicted objectclasses, their leading attributes, and their associations is given in section 2. To avoid confusion with the subject matter under discussion, these are referred to in this document as “objectclasses” rather than using the proper UML terminology of “classes”.

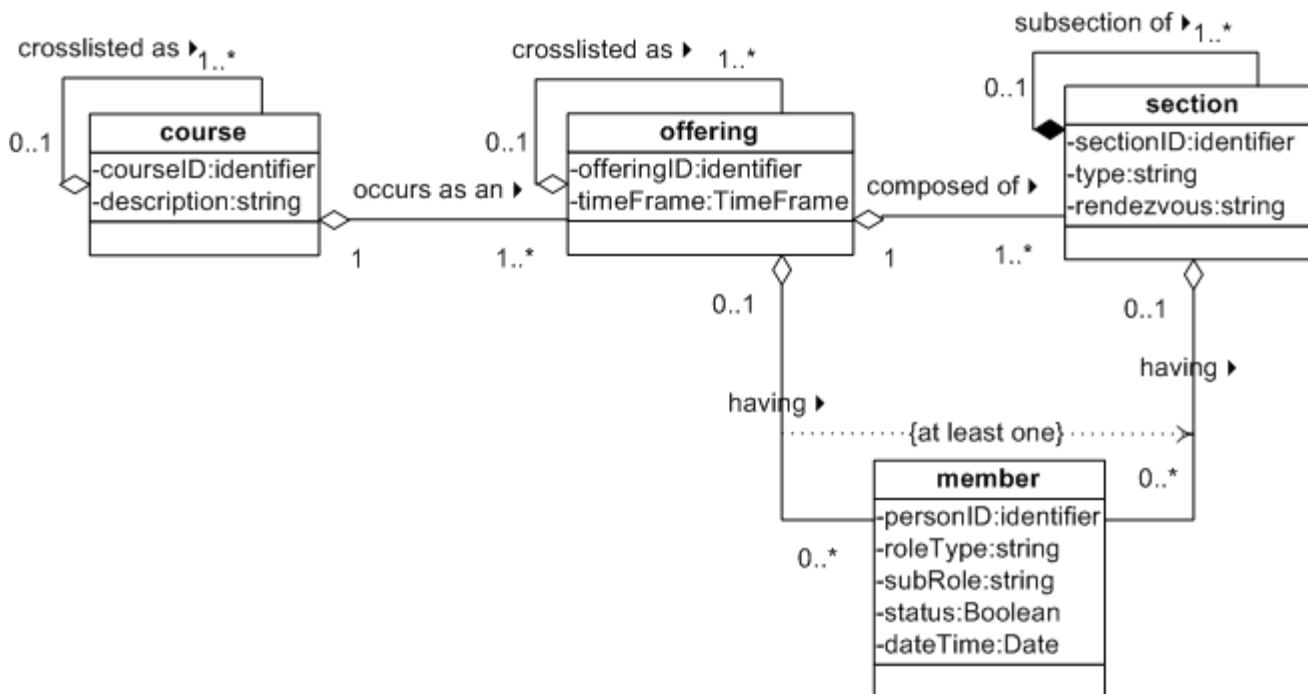


Figure 1: eduCourse data model

Several attribute value types in the tables in section 2 below are marked “(IMS)”. These are defined in the IMS Common Data Definitions Information Model [1].

2. Description of objectclasses, attributes, and associations

2.1. Course

The *course* objectclass models an abstract course, for example, one that appears in a catalog of courses. *Course* instances are not temporal objects (although the catalog of which they are a part might be).

Attribute Name	Type	Multiplicity	Description
courseID	identifier	1	Site-defined identifier for the course. Example: Physics 101.
description	string	1	Description of the course.

The “*crosslisted as*” association enables a *course* instance to be associated with one or more other *course* instances for the purpose of expressing a crosslisting relationship among them. The target of a *crosslisted as* association is crosslisted as each of the sources of the association. Each source is considered to be a presentation of the target. The relationship need not be symmetric. A *course* instance which is a source for a *crosslisted as* association may also be a source for additional *crosslisted as* associations, i.e., one course can appear as a crosslisted presentation of any number of other courses. Not every *course* instance need be the target of a *crosslisted as* association.

2.2. Offering

A temporal instance of a *course* is represented as an *offering*. For example, Physics 12100 as offered during the 2004 Autumn Quarter is an *offering* instance.

Attribute Name	Type	Multiplicity	Description
offeringID	Identifier	1	Site-defined identifier for the course offering.
timeFrame	TimeFrame (IMS)	0..1	Period of time during which the offering occurs.

The “*occurs as an*” association allows each *offering* instance to be the instantiation of one or more *course* instances. This facilitates a temporally constrained form of crosslisting. Each *offering* instance must be a source for at least one *occurs as an* association.

The “*crosslisted as*” association enables an *offering* object to be associated with one or more other *offering* objects for the purpose of expressing a crosslisting relationship among them. The target of a *crosslisted as* association is crosslisted as each of the sources of the association. Each source is considered to be a presentation of the target. The relationship need not be symmetric. An *offering* instance which is a source for a *crosslisted as* association may also be a source for additional *crosslisted as* associations, i.e., one course offering can appear as a crosslisted presentation of any number of other course offerings. Not every *offering* instance need be the target of a *crosslisted as* association.

2.3. Section

Multiple instances of a course offering may be required, either to scale the offering to meet course enrollment levels, or to reflect and manage different types of colocated activities necessary for the course. These are the *sections* of an *offering*. For example, students might separately register for lab, lecture, and recitation sections of a course.

Attribute Name	Type	Multiplicity	Description
sectionID	identifier	1	Site-defined identifier for the section.
type	string	1	Declares the type of section this is (e.g., lab, lecture, recitation, ...).
rendezvous	string	1	A specification of where or how the section's activities are undertaken. Could be a list of times and a location (traditional brick 'n mortar classroom), a URL, or other form to be discussed.

The “*composed of*” association enables one *offering* to be composed of multiple *sections*, and also enables each *section* to be an element of one or more *offerings*. The latter capability is the fourth way that this data model supports a form of crosslisting – in this case, one in which only certain sections of a course offering serve a crosslisted function

(perhaps because not all instructors of *sections* of one *offering* hold a credential necessary to meet a requirement of the crosslisted *course*). Each *section* instance must be a source for at least one *occurs as an* association.

The “*subsection of*” association enables a *section* instance to contain any number of other *section* instances. This enables the eduCourse data model to express components of sections, for example, to group members at a more granular level than the section that instantiates a course offering, or to represent changes in *roleType* persons may have with a section over time. This version of the model does not incorporate any notion of content, but if it did, subsections could also be used to delineate the parts of a section by their content. A *section* instance that is a source for a *subsection of* association is exclusively “owned” by its target *section* instance – it may not be a source for any other *subsection of* association nor any *composed of* associations.

2.4. Member

The *member* objectclass is similar to the *Role* objectclass in the IMS Membership Management Services Information Model [2]. It represents a person connected with a *section* in one of several capacities (such as instructor, learner, teaching assistant, etc.). Attributes of the *member* objectclass that are named the same as IMS *Role* objectclass attributes have the same meaning as in the IMS specification.

Attribute Name	Type	Multiplicity	Description
personID	identifier	1	Externally supplied identifier for the person holding this role with respect to the associated <i>offering</i> or <i>section</i> .
roleType	string	1	The <i>member</i> 's function within an <i>offering</i> or <i>section</i> . Allowed values are determined by the IMS Membership Management Services Information Model [2]. These are currently the case-sensitive strings 'Learner', 'Instructor', 'ContentDeveloper', 'Member', 'Manager', 'Mentor', 'Administrator', and 'TeachingAssistant'.
subRole	string	0..1	As in the IMS Membership Management Services Information Model.
status	Boolean (IMS)	1	Indicates if the <i>member</i> is active or inactive in the associated <i>offering</i> or <i>section</i> .
dateTime	Date (IMS)	0..1	Date the current membership status was established.

Several IMS *Role* attributes are absent from this list. Of particular note is “timeFrame”, which in this model is located with the *offering* objectclass. *Members* inherit a timeFrame from the *offering* to which they are associated either directly or indirectly via a *section* object. This model posits a persisted “personID” as opposed to the “userid” used in the IMS *Role* object to avoid the requirement that this identifier be meaningful to learning management applications.

The “*having*” association connects a *member* instance with the *section* or *offering* instance of which it is a member. Each *member* instance can be associated with at most one *section* instance and with at most one *offering* instance. Both types of association are permitted, enabling memberships to be rolled-up to the offering level or to be expressed only at the offering level in contexts in which that is appropriate. Each *member* instance must be associated with at least one *offering* instance or at least one *section* instance.

3. Bindings of the eduCourse data model

The term “eduCourse” will be used as a prefix or component of names of course-related schematic artifacts like attributes of person or course-oriented schema and names of XML schema or LDAP objectclasses the MACE-

CourseID working group may decide to create, and as a qualifier for names of related UML packages, WSDL interfaces, or the like. Items named with or qualified by "eduCourse" should have a corresponding appearance in the evolving eduCourse data model (figure 1). Example uses:

“eduCourseOffering” - the name of an attribute (defined below).

“eduCourse offering” - qualifies that the "offering" refers to an instance of the *offering* objectclass in the eduCourse data model.

In this section are gathered the initial bindings of elements of the eduCourse data model.

3.1. **eduCourseOffering**

This is an attribute whose values are identifiers of eduCourse offering or section objects. Values are URIs.

When eduCourseOffering appears as an attribute of a person object, it is multivalued and represents a set of eduCourse offerings or sections associated with eduCourse member objects in which the person is identified by the personID attribute of the eduCourse member objects.

The MACE CourseID working group may recommend particular means of mapping a local system of eduCourse offering or section identifiers into a corresponding system of URIs. Example values:

eduCourseOffering: urn:mace:uchicago.edu:classes:autumn2004:phys12100.003

eduCourseOffering: http://wisc.edu/course/offering/2004fall/physics1101

3.2. **eduCourseMember**

This is an attribute whose values represent the roleTypes of eduCourse member objects associated with eduCourse offering or section objects. The value syntax is <role>@<eduCourseOffering value>, where <role> is either the value of the roleType attribute of an eduCourse member object that is associated with either an eduCourse offering or an eduCourse section object identified by <eduCourseOffering value>, or <role> is a URI.

When eduCourseMember appears as an attribute of a person object, it is multivalued and represents a set of associations between eduCourse offerings or sections and eduCourse member objects in which the person is identified by the personID.

Example values:

eduCourseMember: Learner@urn:mace:uchicago.edu:classes:autumn2004:phys12100.003

eduCourseMember: Instructor@http://wisc.edu/course/offering/2004fall/physics1101

4. **References**

[1] “IMS Common Data Definitions Information Model”,

http://www.imsglobal.org/es/esv1p0/imscommon_infov1p0.html

[2] “IMS Membership Management Services Information Model”,

http://www.imsglobal.org/es/esv1p0/imsmembership_infov1p0.html

[3] “OMG Unified Modeling Language Specification, version 1.5”, March 2003

<http://www.omg.org/cgi-bin/apps/doc?formal/03-03-01.pdf>

5. **Changelog**

5.1. **Draft working group document version 03**

- Updated IMS reference URLs to location of accepted form (no longer Public Discussion) of referenced documents.

- Removed obligation to register `eduCourseOffering` and `eduCourseMember` in the `urn:mace:dir:attribute-def` namespace.
- Corrected case of example `eduCourseMember` values.

5.2. Draft working group document version 02

- Added self-referential association to the `section` objectclass and added corresponding text.
- Tightened up use of UML modelling elements, multiplicities mostly, after closely re-reading portions of [3]. Tightened up corresponding sections of explanatory text.
- Clarified the case-sensitivity of `roleType`.

5.3. Draft working group document version 01

- Changed terminology to use “objectclass” instead of “object” to refer to abstract classes, and to use “object” and “instance” synonymously to refer to instances of abstract objectclasses.
- Updated introduction a bit and made consistent use of “eduCourse data model” as the name of the URL static structure.
- Inserted section 3.
- Changed title, filename, and converted from an individual submission to a draft working group document.

5.4. Individual submission version 03

- Embedded the UML static structure diagram in an actual document (this).
- Modified the “objStructure-v2” UML static structure diagram substantially:
 - Replaced the `Session` class with the “timeFrame” attribute of the `offering` class.
 - Tightened up the multiplicities, end adornments, and names of all associations to better reflect UML practice. They used to be more representative of relationships between tables in an RDBMS, which is an incorrect metaphor in the UML static structure context.
 - Tightened up attributes for each class to (1) declare their types, (2) omit ones named “...” that used to signal the fact that the working group hasn’t yet attempted to fully schematize these objects (this is now acknowledged in the text), and (3) adopt as much of the IMS stuff as seems reasonable given the differences between our task and the circumstances that IMS is governing.

5.5. Individual submission version 02

This was a .gif-only version, “objStructure-v2.gif”, the first attempt to express desired object structure using a UML static structure diagram.