



**Contribution to the conference:**

**“Developing an EML authoring and content management environment”.**

**DRAFT**

*The case of the Open Virtual Education Space.*

**TS Vreeland**  
Architect and Chief Technologist  
OpenVES



## Table of contents.

Introduction. ....	3
The Open Virtual Education Space (OpenVES.org) Project.....	3
Instructional Design and Content Management approach.....	5
Use Cases.....	7
Use Cases authoring. ....	10
Use Cases content management.....	14
Use Case reference runtime system.....	20
Class Diagrams. ....	23
Expectations re the conference. ....	28

# Introduction.

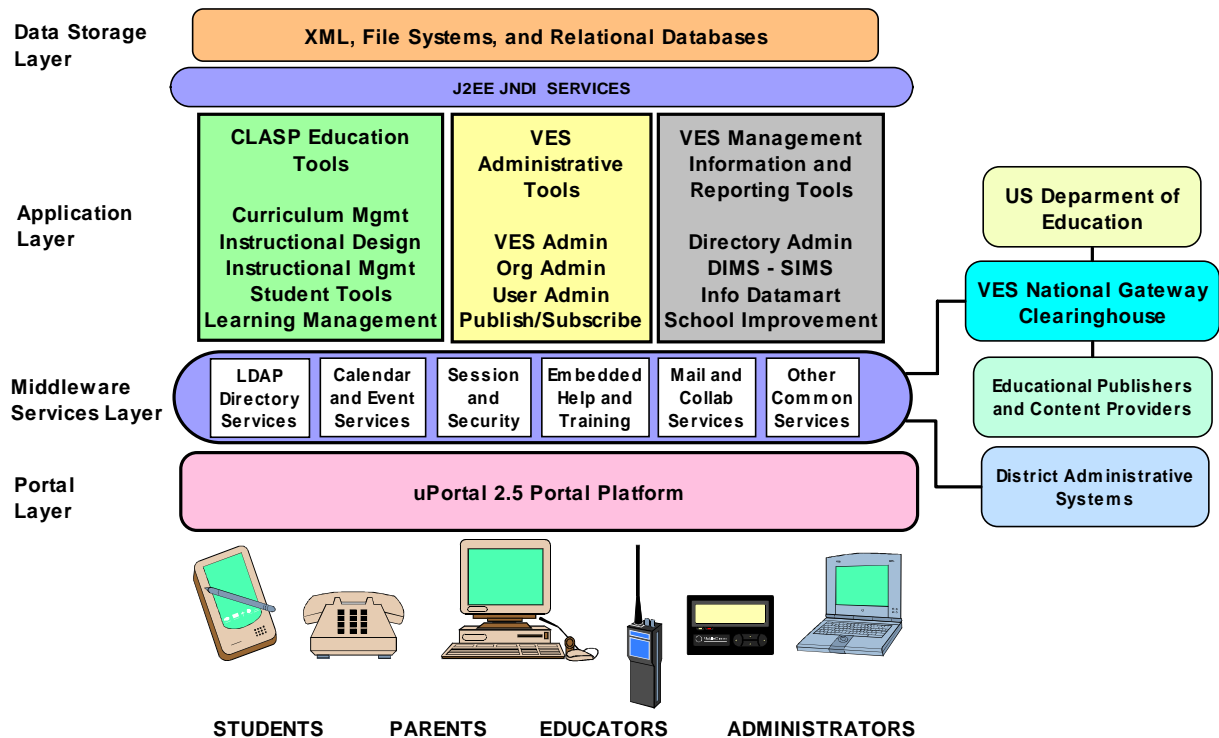
**NOTE:** It is very difficult to normalize the use cases and class model from an implementation not build within the EML framework. Given the short time available to do this I have done the best time would permit. I hope to take away from the conference a better idea of how our work can be informed by the EML work.

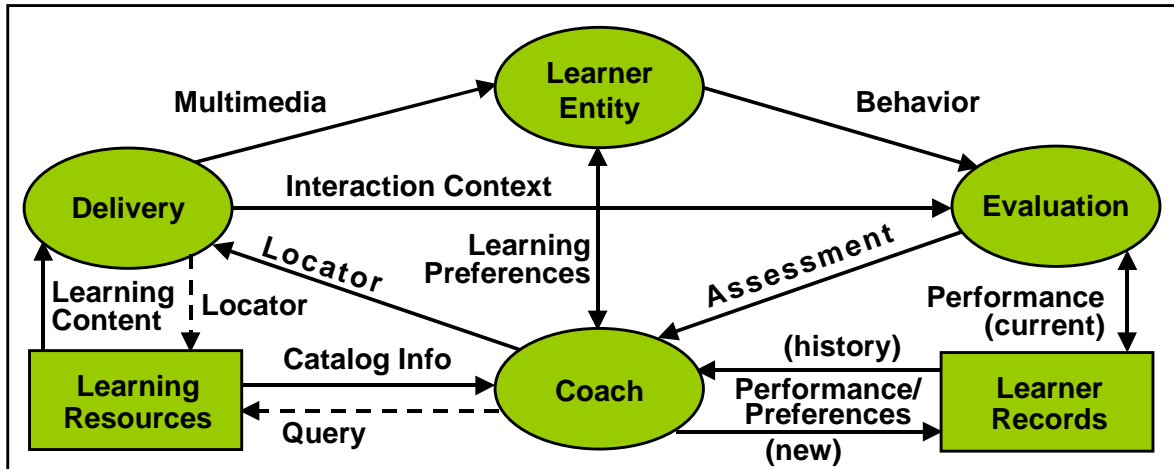
## The Open Virtual Education Space (OpenVES.org) Project.

**Open Virtual Education Space (OpenVES)** is a free, public, k12 eLearning environment that empowers educators, students, and parents with resources designed to help all students achieve high standards. VES is not intended to replace the face-to- face classroom, but to support, enhance, and extend it in time and space. VES enables schools to create their own online learning communities with powerful personalized tools and collaborative workspaces, which have the potential to transform teaching and learning in our schools, one classroom at a time.

It is the OpenVES mission to specify and implement a massively scalable enterprise architecture, capable of supporting millions of students and hundreds of thousands of educators. VES is an existing reference implementation of that open architecture, standards based, enterprise scale eLearning platform, a middleware services layer providing directory, security, collaboration, workflow and communication services, and a set of integrated educational applications, tools and resources. VES is also a rich web service, and pervasive XML based, application development environment.

### Virtual Education Space (VES) National Model

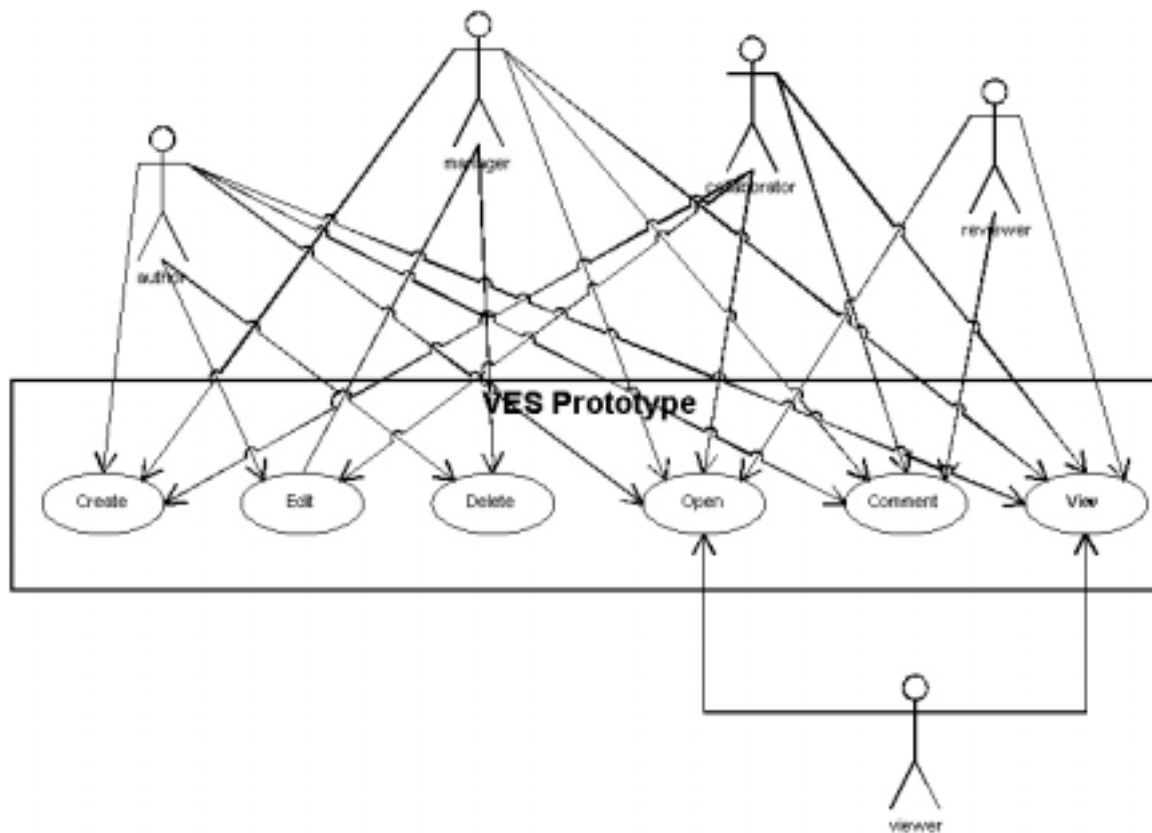




We have attempted to incorporate the best ideas in educational technology, cognitive learning theory, pedagogical principles and best practice into the OpenVES educational tools. Based on the IEEE 1484 Learning Technology Standards Architecture, the OpenVES Architecture models best practice in Educational Technology and in Enterprise Web Design. Recognizing that each state and school district would want to localize VES to brand it and adapt it to local needs, we made it highly customizable. Recognizing that each student, parent and educator would want a personalized interface we created affinity groups and options for deep personalization. Recognizing that tools and resources needed change as roles change we provide role customization. Recognizing that users have different privacy preferences and needs for access, we created accessibility and privacy personalization options. Recognizing that students and teachers need a safe and secure environment on the web we have created an authentication and security system to assure that safety.

## ***Instructional Design and Content Management approach.***

At the highest level, three key VES subsystems are relevant to the work of the conference.



*Figure 1: Overview of Users and Roles.*

### ***1. Instructional Design system (Authoring system).***

The Instructional Design System is a pervasively collaborative environment for design, documentation and evaluation of RLOs and RIOs. These objects can be completely new, or they can be adaptations of existing RLOs and RIOs. All communication to and from the Instructional system is in the form of RLOs, RIOs or Instructional Objects (Units of Study<sup>1</sup>). Although the Instructional Design system is represented as a single system here, it is actually comprised of various components. The Instructional Design system is a web based tool and its content repository is a web based repository.

### ***2. Content Repository (Content Management System).***

This is a central web based file store where all Reusable Objects (ROs) are stored and managed. Reusable Objects are Reusable Learning Objects (RLOs) or Reusable Information Objects (RIOs). The objects exist as XML documents and they are stored in a SoftwareAG Tamino native XML database. Users in all participating organizations have access to the content repository. At any point in time the objects in the content repository are in various states along a developmental continuum, and within the collaborative process. Objects are only visible to learners when completed and deployed.

<sup>1</sup> A Unit of Study is an RLO that contains all elements needed to provide a full-fledged learning experience.

### ***3. Personalized Virtual Education Space (Virtual Learning Environment).***

The Virtual Education Space (a Virtual Learning Environment (VLE)) is used to present a set of personalized and highly customized views (WorkSpaces) to learners and instructors within VES organizations. Instructional objects are available to learners in their Classroom or Course WorkSpaces. In these WorkSpaces the Learning objects can be rendered by the Learning Management System.

## Use Cases.

Each subsystem comprises a number of Use Cases describing the activities to be carried out by the various actors with the system.

Use Case name	#	Diagram #	Summary description (15 words max.)
Create Instructional Object	1	figure 2	Author creates Instructional Object in an Instructional context
Edit Instructional Object	2	figure 2	Manager (Author) or Collaborator edits Instructional Object in an Instructional Design context
Sequence Instructional Object	3	figure 2	Manager (Author) or Collaborator sequences Instructional Object in an Instructional Design context
Delete Instructional Object	4	figure 2	Manager (Author) deletes Instructional Object in an Instructional Design context
Comment on Instructional Object	5	figure 2	Manager (Author), Reviewer, or Collaborator comments on Instructional Object in an Instructional Design context
View Instructional Object	6	figure 2	Permissioned User views Instructional Object in an Instructional Design context
View Analysis Instructional Object	7	figure 3	Manager (Author), Reviewer, or Collaborator views analysis on Instructional Object in an Instructional Design context
View Details Instructional Object	8	figure 4	Manager (Author), Reviewer, or Collaborator views details on Instructional Object in an Instructional Design context
View Comments Instructional Object	9	figure 5	Manager (Author), Reviewer, or Collaborator views comments on Instructional Object in an Instructional Design context
View History Instructional Object	10	figure 6	Manager (Author), Reviewer, or Collaborator views history on Instructional Object in an Instructional Design context
Instructional Object Strategy	11	figure 7	Manager (Author), Reviewer, or Collaborator sets strategy on Instructional Object in an Instructional Design context
Schedule Instructional Object	12	figure 8	Manager (Author), Reviewer, or Collaborator schedules an Instructional Object in an Instructional Delivery context
Align Instructional Object	13	figure 9	Manager (Author), Reviewer, or Collaborator aligns Instructional Object to learning standards in an Instructional Design context
View Usage of Instructional Object	14	figure 10	Manager (Author), Reviewer, or Collaborator views actual usage on Instructional Object in an Instructional context
Collaborate or Share Instructional Object	15	figure 11	Manager (Author) changes scope of sharing or collaboration on an Instructional Object in an Instructional Design context
Learner View Instructional	16	Figure 12	See Use Case 6 Permissioned User views Instructional Object in an

Object			Instructional Delivery context
--------	--	--	--------------------------------

*Table 1: Use Cases listing.*

Learning Objects are visible with differing levels of interaction to users who possess one of the following four roles:

<b>Actor name</b>	<b>Use Case #</b>	<b>Role description</b>
Manager	1	This is the person who is responsible for the object. The manager has overall control of the object's state and status (within a customized process workflow), as well as the roles other users will have with respect to the object. It is expected that the manager will guide the development of an object through the states of the organizational workflow.
Collaborator	2	These are the users who are working in concert with the manager to instantiate, design, and evaluate the object and its child objects. They can execute all actions the manager can, except deletion of the object and modifications to the object's workflow state.
Reviewer	3	Can view and make comments on an object.
Viewer (Learner)	4	These are users who can look at an object's settings and properties, but not make any modifications; also, they cannot make comments about the object. They can view (or play) an object in a learning context.

*Table 2: User Roles listing.*

## ***Use Cases authoring.***

In the authoring system we distinguish four actors. It is possible that one person may perform the role of various actors at different times with respect to different learning objects.

The system employs an object paradigm that allows a user access to objects based on a specified set of *roles* with respect to each object. These abilities of these roles depend on the object *state* and *status*.

Roles specify *permissions* to act on an object, and are used (in concert with the object state and status) to determine what set of actions a specific user can execute on a specific object. Roles can be assigned both to users and to groups: when assigned to a group, all members of that group are automatically assigned that role. Figure 3 gives the primary actions users with various roles can execute on an instructional object.

The object status takes on one of 3 values, *private*, *shared*, or *public*, which determine the visibility level of the object. When an object is private, it is visible only to its creator, the user who initially entered the object into the system. This is the default status an object is in, and the only role applicable to an object in this status is the *creator* role. Following the creation of the object, there is a period in which the creator modifies and develops the instructional object alone.

An object in private status can be shared. In the shared status, there is collaboration involving a number of users in the ongoing development of the instructional object. While in the shared status, the object may exist over time in a number of different states. These states, in the shared status, represent the workflow a particular state or district designs for its users. An example of a three state workflow model is the default:

1. School District Draft
2. Pending District Approval
3. Approved - Awaiting Publication and Use

During the time an object is in a shared status, the object is visible with differing levels of interaction to users who possess one of the following four roles:

- a. **Manager:** this is the person who is in charge of the object. The manager has overall control of the object's state and status, as well as the roles other users will have with respect to the object. It is expected that the manager will guide the development of an object in the states of the organizational workflow.

The system defines a *workflow* that specifies how the district approves instruction; this approval is given when the instruction adheres to all the district curriculum guidelines and objectives for the selection of grade (span) and discipline. Within this workflow a set of managers are designated as approvers; these are users who are authorized to approve an object. These reviewers can execute all actions collaborators can, but can also give the object organizational approval or endorsement.

- b. **Collaborators:** these are the users who are working in concert with the manager to design the object and its child objects. They can execute all actions the manager can, except deletion of the object and modifications to the object's state.
- c. **Reviewers (Commentators):** Can view and make comments on an object.
- d. **Viewers:** these are users who can look at an object's settings and properties, but not make any modifications; also, they cannot make comments about the object.

ACTIONS	Instructional Objects						Administrative Objects				
	Course	Unit	Lesson	Activity	Assessme	Resource	State	District	School	Classroom	User
Open	x	x	x	x	x	x	x	x	x	x	x
Create	x	x	x	x	x	x	x	x	x	x	x
Edit	x	x	x	x	x	x	x	x	x	x	x
Sequence		x	x	x							
Copy To	x	x	x	x	x	x					
Move To								x	x	x	x
Delete	x	x	x	x	x	x					
Upload					x	x					
Download					x	x					
Comment/Evaluate	x	x	x	x	x	x	x	x	x	x	x

Figure 2: Mapping of Use Cases to Instructional and Administrative Objects

Authors and their Collaborators will utilize a structured tool to create, edit, and document learning objects and associate them with the Instructional hierarchy. An important part of the VES Instructional Design process is the alignment of learning objects with the subject taxonomies and the learning standards and their metadata. It is expected that Authors will use a wide array of Media Creation tools to create RIOs and other resources to be used in the interactive rendering of the RLOs. It is the “alignment” of RLOs and RIO components to the learning standards and subject taxonomies that make them discoverable to other authors

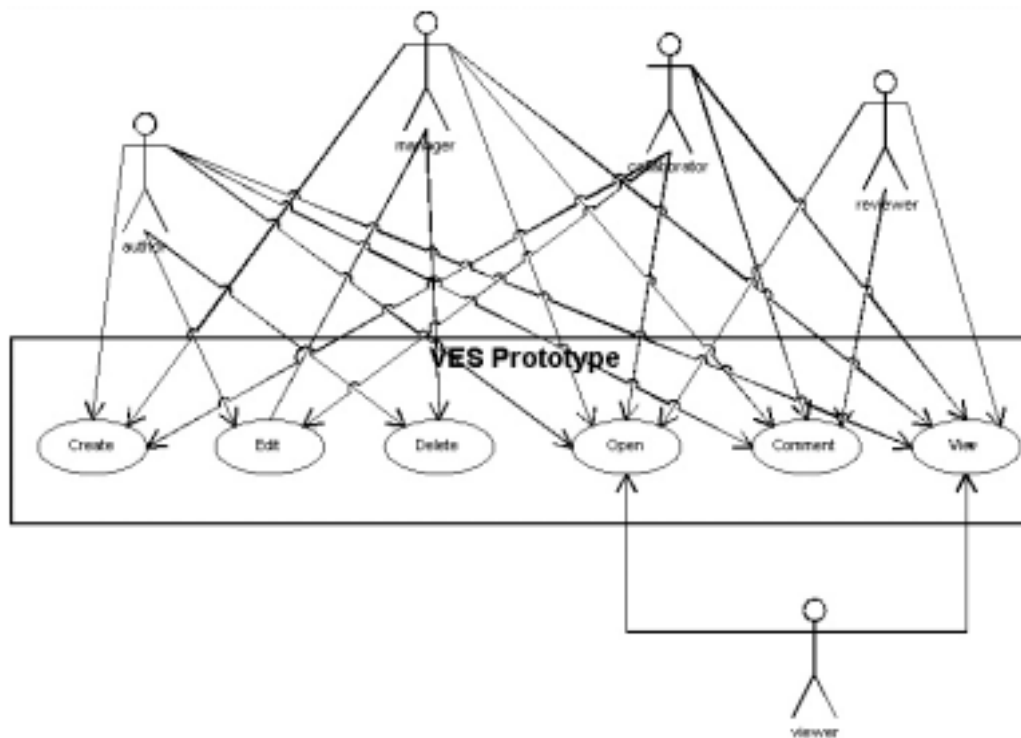


Figure 2: Overview of Authoring Use Cases.

Use Case #: 1	
Use Case name	<b>Create Instructional Object</b>
Description (max. 200 char.)	Author creates Instructional Object in an Instructional context
Actors	Author (Manager)
Prerequisites	
Outcome	Instructional Object (RLOs or RIOs).
Main success scenario (optional)	
Exceptions (optional)	
Remarks (optional)	

Use Case #: 2	
Use Case name	<b>Edit Instructional Object</b>
Description (max. 200 char.)	Manager (Author) or Collaborator edits Instructional Object in an Instructional Design context
Actors (5 max.)	Manager (Author) or Collaborator
Prerequisites	Instructional Object
Outcome	Modified Instructional Object
Main success scenario (optional)	
Exceptions (optional)	
Remarks (optional)	

Use Case #: 3	
Use Case name	<b>Sequence Instructional Object</b>
Description (max. 200 char.)	Manager (Author) or Collaborator sequences Instructional Object in an Instructional Design context
Actors (5 max.)	Manager (Author) or Collaborator
Prerequisites	Instructional Object
Outcome	Sequenced Instructional Objects
Main success scenario (optional)	
Exceptions (optional)	
Remarks (optional)	

Use Case #: 4	
Use Case name	<b>Delete Instructional Object</b>
Description (max. 200 char.)	Manager (Author) deletes Instructional Object in an Instructional Design context
Actors (5 max.)	Manager (Author)
Prerequisites	Instructional Object

Outcome	
Main success scenario (optional)	
Exceptions (optional)	
Remarks (optional)	

Use Case #: 5	
Use Case name	<b>Comment on Instructional Object</b>
Description (max. 200 char.)	Manager (Author), Reviewer, or Collaborator comments on Instructional Object in an Instructional Design context
Actors (5 max.)	Manager (Author), Reviewer, or Collaborator
Prerequisites	Instructional Object
Outcome	Commented Instructional Object
Main success scenario (optional)	
Exceptions (optional)	
Remarks (optional)	

Use Case #: 6	
Use Case name	<b>View Instructional Object</b>
Description (max. 200 char.)	Permissioned User views Instructional Object in an Instructional Design context
Actors (5 max.)	Permissioned User
Prerequisites	Instructional Object
Outcome	Instructional Object
Main success scenario (optional)	
Exceptions (optional)	
Remarks (optional)	

The user interface of the authoring system can vary for each actor. Each actor should be presented with a view which fits his role and the way he uses the system. Within the system this should all be translated into standard-EML, preferably automatically. For example the educationalist can use an activity diagram (like in UML), or the metaphor of the TeleTop system where the timetable may be presented as a play. Defining the right (or at least operational) user interfaces for the various actors in the authoring process is an activity requiring special attention. In addition, the information entered by these actors needs to be integrated into one or more RLOs or RIOs before it can be delivered to the content management system.

## Use Cases content management

The actors for the content management system are the content manager, the authoring system and institutional systems. Within the institutions, additional sub-roles may be defined.

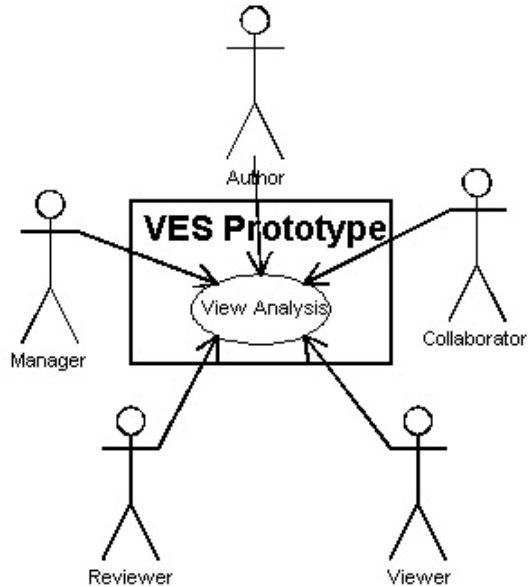


Figure 3: View Analysis for Instructional Object.

Use Case #: 7	
Use Case name	<b>View Analysis for Instructional Object</b>
Description (max. 200 char.)	Manager (Author), Reviewer, or Collaborator views analysis on Instructional Object in an Instructional Design context
Actors (5 max.)	Manager (Author), Reviewer, or Collaborator
Prerequisites	Instructional Object
Outcome	Instructional Object
Main success scenario (optional)	
Exceptions (optional)	
Remarks (optional)	

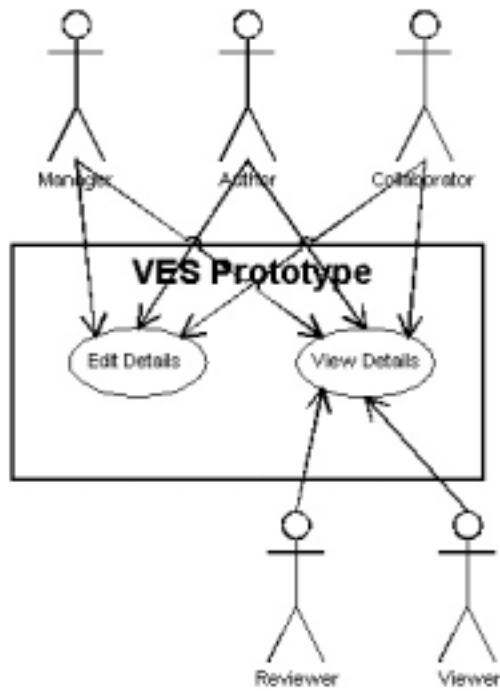


Figure 4: View or Edit Details for Instructional Object.

Use Case #: 8	
Use Case name	<b>View or Edit Details for Instructional Object</b>
Description (max. 200 char.)	Manager (Author), Reviewer, or Collaborator views details on Instructional Object in an Instructional Design context
Actors (5 max.)	Manager (Author), Reviewer, or Collaborator
Prerequisites	Instructional Object
Outcome	Modified Instructional Object
Main success scenario (optional)	
Exceptions (optional)	
Remarks (optional)	

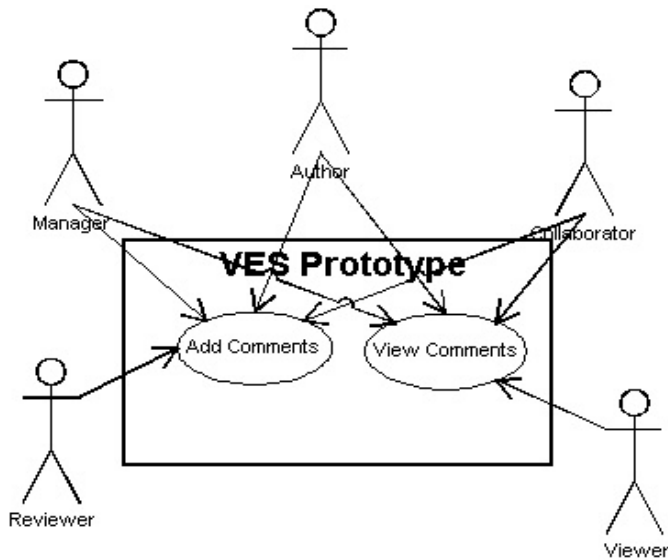


Figure 5: View or Edit Comments for Instructional Object.

Use Case #: 9	
Use Case name	<b>View or Edit Comments for Instructional Object</b>
Description (max. 200 char.)	Manager (Author), Reviewer, or Collaborator views comments on Instructional Object in an Instructional Design context
Actors (5 max.)	Manager (Author), Reviewer, or Collaborator
Prerequisites	Instructional Object
Outcome	Modified Instructional Object
Main success scenario (optional)	
Exceptions (optional)	
Remarks (optional)	

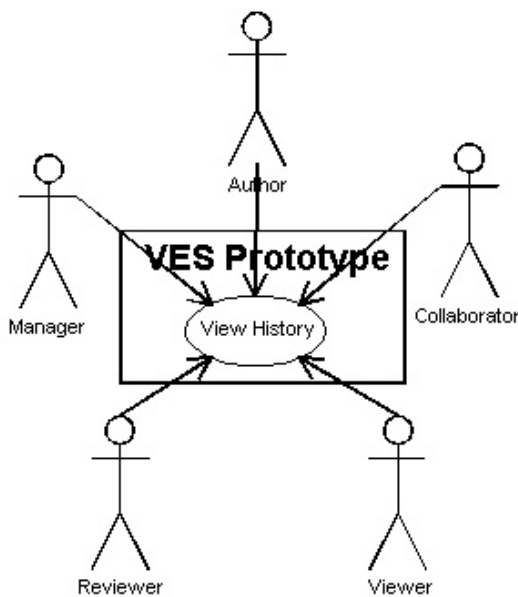


Figure 6: View History for Instructional Object.

Use Case #: 10	
Use Case name	<b>View History for Instructional Object</b>
Description (max. 200 char.)	Manager (Author), Reviewer, or Collaborator views history on Instructional Object in an Instructional Design context
Actors (5 max.)	Manager (Author), Reviewer, or Collaborator
Prerequisites	Instructional Object
Outcome	Instructional Object
Main success scenario (optional)	
Exceptions (optional)	
Remarks (optional)	

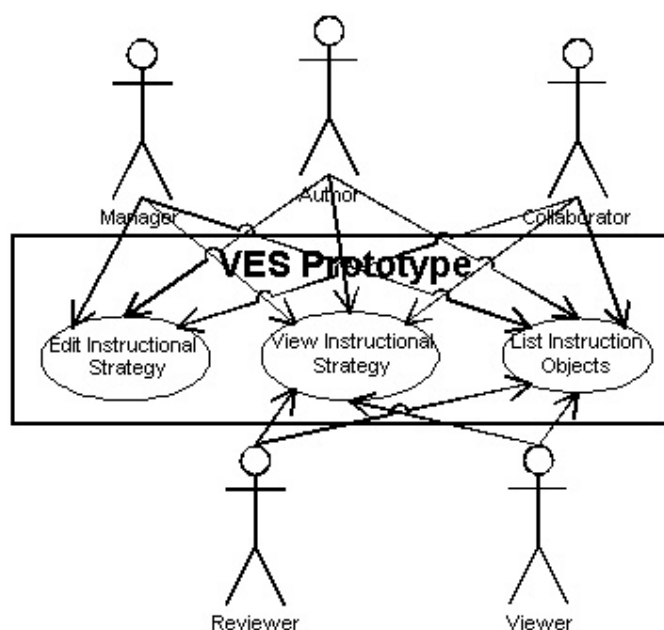


Figure 7: View – Edit Instructional Object Strategy.

Use Case #: 11	
Use Case name	<b>View – Edit Instructional Object Strategy</b>
Description (max. 200 char.)	Manager (Author), Reviewer, or Collaborator views and sets strategy on Instructional Object in an Instructional Design context
Actors (5 max.)	Manager (Author), Reviewer, or Collaborator
Prerequisites	Instructional Object
Outcome	Modified Instructional Object
Main success scenario (optional)	
Exceptions (optional)	
Remarks (optional)	

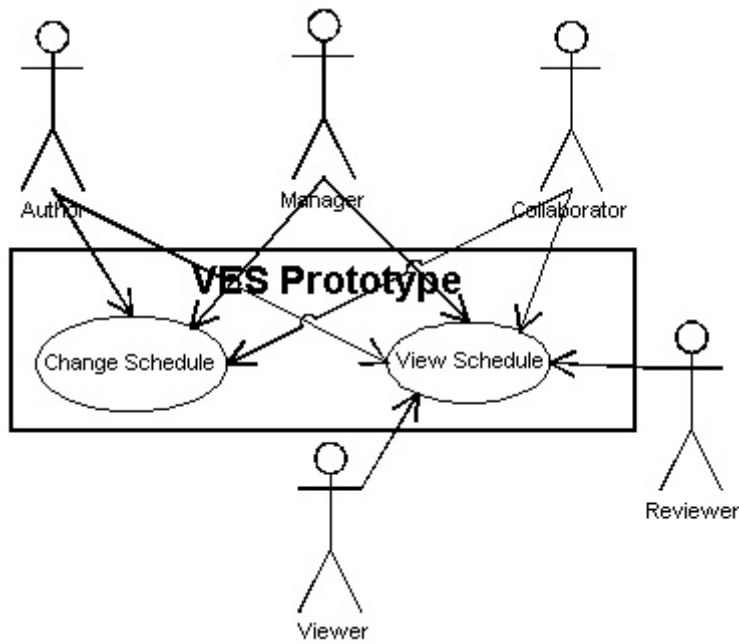


Figure 8: View – Schedule Instructional Object.

Use Case #: 12	
Use Case name	<b>Schedule Instructional Object</b>
Description (max. 200 char.)	Manager (Author), Reviewer, or Collaborator schedules an Instructional Object in an Instructional Delivery context
Actors (5 max.)	Manager (Author), Reviewer, or Collaborator
Prerequisites	Instructional Object
Outcome	Modified Instructional Object
Main success scenario (optional)	
Exceptions (optional)	
Remarks (optional)	

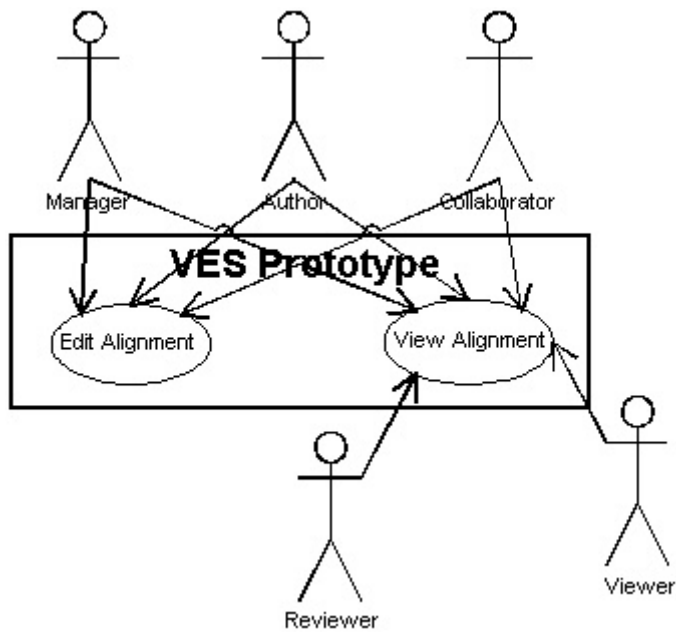


Figure 9: Align Instructional Object.

Use Case #: 13	
Use Case name	<b>Align Instructional Object</b>
Description (max. 200 char.)	Manager (Author), Reviewer, or Collaborator aligns Instructional Object to learning standards in an Instructional Design context
Actors (5 max.)	Manager (Author), Reviewer, or Collaborator
Prerequisites	Instructional Object
Outcome	Modified Instructional Object
Main success scenario (optional)	
Exceptions (optional)	
Remarks (optional)	

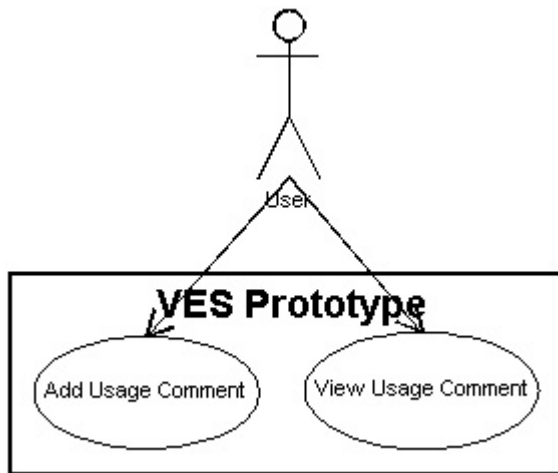


Figure 10: View – Edit Usage on Instructional Object.

Use Case #: 14	
Use Case name	<b>Usage for Instructional Object</b>
Description (max. 200 char.)	Manager (Author), Reviewer, or Collaborator views actual usage on Instructional Object in an Instructional context
Actors (5 max.)	Manager (Author), Reviewer, or Collaborator
Prerequisites	Instructional Object
Outcome	Modified Instructional Object
Main success scenario (optional)	
Exceptions (optional)	
Remarks (optional)	Diagram is incorrect re: users

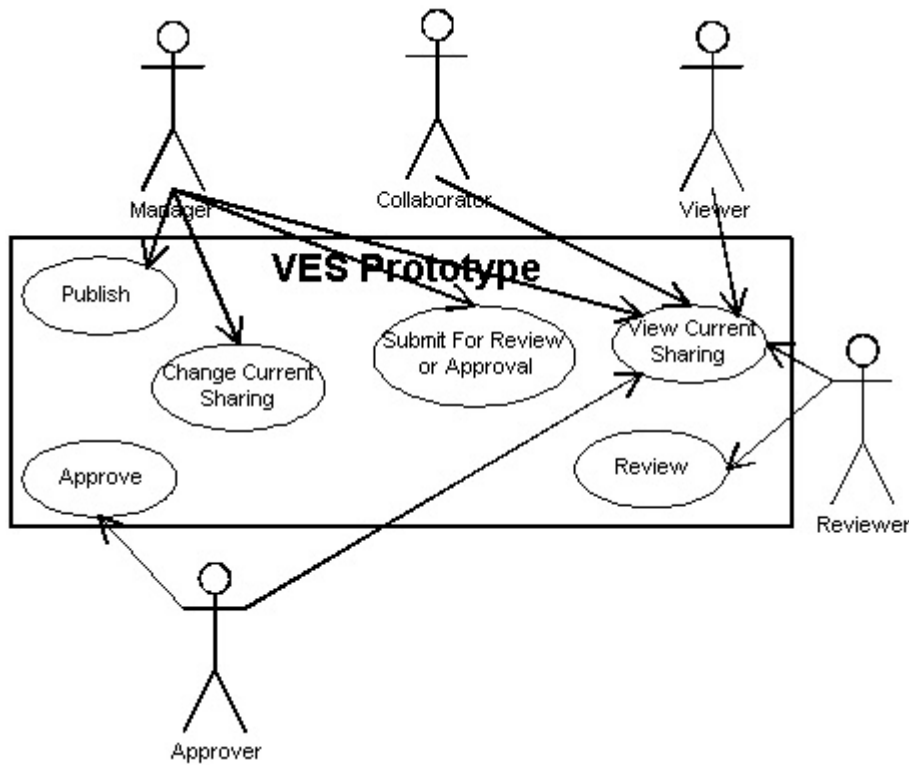


Figure 11: View – Edit Sharing and State on Instructional Object.

Use Case #: 15	
Use Case name	<b>View – Edit Collaboration, Sharing, and State on Instructional Object</b>
Description (max. 200 char.)	Manager (Author) changes scope of sharing or collaboration on an Instructional Object in an Instructional Design context
Actors (5 max.)	Manager (Author)
Prerequisites	Instructional Object
Outcome	Instructional Object
Main success scenario (optional)	
Exceptions (optional)	
Remarks (optional)	

## Use Case Virtual Learning Environment

The VES VLE only knows one actor and one Use Case, which is playing an Instructional Object (Unit of Study). This Use Case however contains many options for playing.

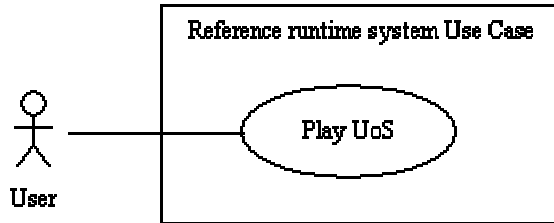


Figure 12: View – Edit Sharing and State on Instructional Object.

Use Case #: 16	
Use Case name	<b>Render Instructional Object (Play UoS).</b>
Description (max. 200 char.)	Permissioned User views Instructional Object in an Instructional Delivery context
Actors (5 max.)	Learner (User).
Prerequisites	Instructional Object
Outcome	Instructional Object
Main success scenario (optional)	
Exceptions (optional)	
Remarks (optional)	

## Class Diagrams.

VES is an eLearning environment designed to enable elementary, middle school and high school teachers within the Massachusetts educational system to design and deploy instructional programs which are aligned to the a state curriculum frameworks and standards.

The underlying basis for a standards based educational system is a series of national standards in education that outline what a student at each level of the K-12 educational system is supposed to be able to know and be able to do. Each state studies these *national standards* and then builds their own *state framework* that outlines the goals of how the standards are to be implemented within the state across spans of grades (one or more grades at a time). In Massachusetts the standards are outlines within each *strand* (effectively sub-field or concentration) of the 7 *disciplines* (Mathematics, English Language Arts, Foreign Language, the Arts, History and Social Science, Science and Technology, and Health). Within the state, each district takes these standards and outlines the *district curriculum guidelines* as a set of *objectives* that indicate specifically what students should know and be able to do at each grade level in the appropriate grade span.

There may be on the order of 100 applicable standards and objectives within each grade for any one discipline, and ensuring that all of these are covered by some instructional element is a problem that VES is helping teachers to solve. This process, called *alignment*, is discussed further below.

As such, the primary goal of the application is to provide a design and documentation tool that allows educators to record information about the structure of their instructional programs, record the alignments to the appropriate state standards and district objectives, and collaborate with other educators in this design process. A number of other functions also emerge in concert with these aims, and these are outlined below.

For the purposes of this application, instructional objects are organized into a particular hierarchy. The hierarchy consists of 6 objects, with the hierarchy as outlined in Figure 13 below.

The objects are:

- Course, which contains one or more Units, and zero or more Assessments and Resources.
- Unit, which is contained within a Course, and contains one or more Lessons, and zero or more Assessments and Resources.
- Lesson, which is contained within a Unit, and contains one or more Activities, and zero or more Assessments and Resources.
- Activity, which is contained within a Lesson, and contains zero or more Assessments and Resources.
- Assessment, which is contained within a Course, Unit, Lesson, Activity, or Assessment, and contains zero or more Assessments and Resources. Assessments may only nest one level deep (i.e. if an Assessment is the child of another Assessment, it may not itself contain Assessments).
- Resource, which is contained within a Course, Unit, Lesson, Activity, Assessment or Resource, and contains zero or more Resources. Resources may only nest one level deep (i.e. if a Resource is the child of another Resource, it may not itself contain Resources).

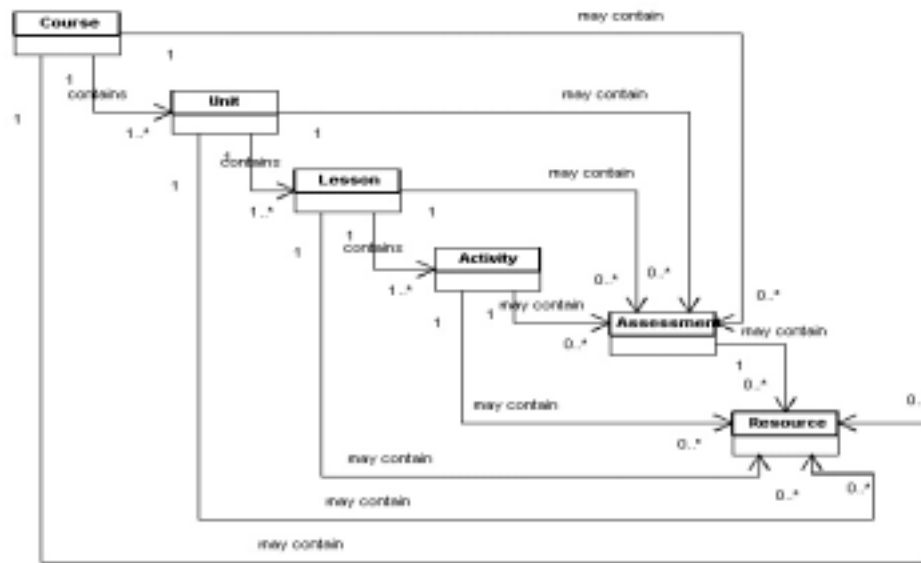


Figure 13: Class Diagrams for the Instructional Objects.

Class name	Instructional Object (Reusable Object (RO).)
Diagram #	13
Summary description	A RO is either a RIO or a RLO. It is used as a container concept for all objects managed from a reusability perspective. All ROs contain meta-data, but not all objects containing meta-data are ROs. Many objects requiring meta-data according to IMS-EML, are not reusable, e.g. the element 'learning objective'. Such elements are relevant for a specific piece of instruction, but are irrelevant from the perspective of reusability. As such they are no ROs.
Attribute(s) description	- meta data (compulsory)
Operation(s) description	- create - edit - delete
Remarks (optional)	

Class name	Resource Object - Reusable Information Object (RIO)
Diagram #	13
Summary description	A RIO is a reusable object containing meta-data, without any prerequisites or learning objectives linked to it. It is thus 'neutral' in the sense that it can be used in a variety of learning situations. An example of a RIO is a videoclip of a speech by Martin Luther King. It can be used in material about race discrimination as well as in a context about making video productions.
Attribute(s)	- meta data: domain classification

description	
Operation(s) description	- see RO
Remarks (optional)	

Class name	Resource Object - Reusable Learning Object (RLO)
Diagram #	13
Summary description	A RLO is a reusable piece of educational material containing prerequisites and learning objectives. It is necessary to distinguish RIO and RLO because of the way they are used. Without learning objectives and prerequisites it is no learning material. In an educational setting the RLOs are the things that really matter. While RLOs are developed and used specifically in an educational setting, RIOs form a diffuse category to be used in various settings.
Attribute(s) description	- meta data - prerequisites - learning objectives
Operation(s) description	- see RO
Remarks (optional)	

Class name	Activity Object - Unit of Study (UoS)
Diagram #	13
Summary description	A UoS is a RLO composed of a coherent set of learning tasks and related learning environments. According to the IMS-specific EML interpretation a UoS is a piece of educational material containing meta-data, learning objectives, prerequisites, content (can be referenced) and a play (scenario); special cases of a UoS are a module, course, or even a curriculum as an aggregate of UoS's.
Attribute(s) description	- meta data - prerequisites - learning objectives - roles - content - play/method
Operation(s) description	- see RO - submit to reference runtime system - submit to institution-specific VLE
Remarks (optional)	

Class name	Course, Unit, Lesson Object
Diagram #	13
Summary description	These are all containers for Activity Objects (UoS's) within specific educational organizations, settings or methods. They cannot be defined as such in IMS-EML, but can of course be specified in the meta-data. It

	is possible in IMS-EML to nest UoS' (e.g. modules in courses in curricula) by referencing to other UoS's in a UoS' play.
Attribute(s) description	- see UoS
Operation(s) description	- see UoS
Remarks (optional)	

Class name	Assessment Object
Diagram #	13
Summary description	These are types of RIOs defined in IMS-Q&TI.
Attribute(s) description	
Operation(s) description	
Remarks (optional)	

Metadata: Describes the content of a RO, but is not part of the content itself. Dublin Core and IEEE/IMS LOM are relevant metadata specifications.

There are also a number of administrative classes represented by administrative objects. The administrative object inheritance hierarchy is shown in Figure 14 below.

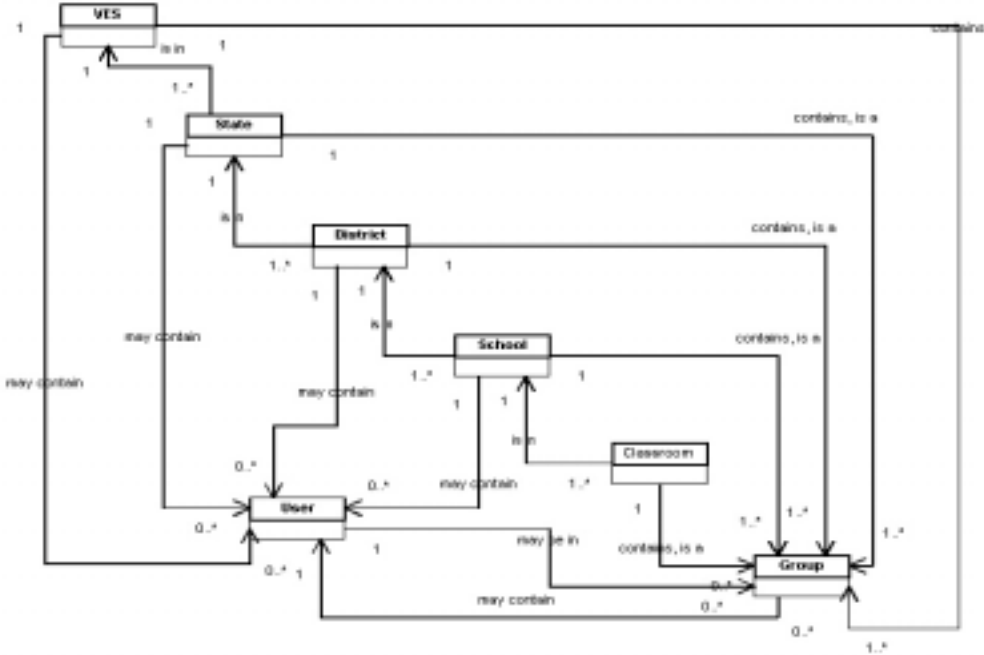


Figure 14: Class Diagrams for the Administrative Objects.

The Administrative classes are:

- VES: this object represents the application as a whole, and contains 1 or more States, 1 or more Groups (discussed below), and 0 or more Users (discussed below).
- A State, which is a special case of a Group, and represents a State within the system. It contains 1 or more Districts, 1 or more Group, and 0 or more Users.
- A District, which is a special case of a Group, and represents a School District within the system. It contains 1 or more Schools, 1 or more Group, and 0 or more Users.
- A School, which is a special case of a Group, and represents a School within the system. It contains 1 or more Classrooms, 1 or more Group, and 0 or more Users.
- A Classroom, which is a special case of a Group, and represents a Classroom within the system. It contains 1 or more Groups and 0 or more Users.
- A Group, which is a collection of Users and other Groups within the system. These can nest infinitely deep, and contain 0 or more Groups and 0 or more Users.
- A User, which represents a unique user (specified by a unique username) and with a set of roles and permissions within the system.

## **Expectations re the conference.**

I have a number of important expectations I hope to meet at the conference:

- To better understand the status and philosophy of the Education Modelling Language and the technology being built using it.
- To reconcile the approach we have taken with the EML Tools approach
- To work toward a better understanding of the ways the EML approach can add value to our efforts in the states to build a SCORM-like set of reference models for k12 eLearning platform standardization and deployment.