Modeling units of study from a pedagogical perspective
the pedagogical meta-model behind EML

Rob Koper
Educational Technology Expertise Centre
Open University of the Netherlands
First Draft, version 2
June, 2001

This text is a short summary of the work on pedagogical analysis carried out when EML (Educational Modelling Language) was being developed. Because we address pedagogical meta-models the consequence is that I must justify the underlying pedagogical models it describes. I have included a (far from complete) list with literature used in the pedagogical analysis. I am sorry for its length, but for every pedagogical meta-model it is crucial to define the space of models where it is ‘meta’ to.

As an aid to comprehension, I will use UML diagrams to express static and dynamic relationships when appropriate. All diagrams are drawn from a conceptual perspective and not from an implementation perspective.

This paper is provided as input for the IMS Learning Design group.

Please mail comments, questions and any other reactions to: rob.koper@ou.nl

With thanks to Adrian Rawlings for reviewing this version.

# Table of Content

1. INTRODUCTION ................................................................................................................... 3
2. LEARNING DESIGN IS MODELING 'UNITS OF STUDY' ......................................................... 3
3. THE LEARNING OBJECTS MODEL ..........................................................................................4
4. REQUIREMENTS FOR UNITS OF STUDY AS A RESULT OF LEARNING DESIGN ............. 6
5. ARCHITECTURAL REFERENCE MODEL ..................................................................................7
6. THE PEDAGOGICAL META-MODEL ..........................................................................................8
   6.1. PART 1: THE LEARNING MODEL ....................................................................................... 10
   6.2. PART 2: THE UNIT OF STUDY MODEL .............................................................................11
   6.3. PART 3: THE DOMAIN MODEL .......................................................................................12
   6.4. PART 4: THEORIES OF LEARNING AND INSTRUCTION ..................................................12
   6.5. AN INTEGRATED PICTURE OF THE META-MODEL ..........................................................14
7. TYPES OF LEARNING OBJECTS ..............................................................................................15
8. THE IMPLEMENTATION OF THE META-MODEL IN AN XML SCHEMA ............................ 17
   8.1. XML SCHEMAS ..................................................................................................................17
   8.2. THE BASIC STRUCTURE OF EML ..................................................................................17
   8.3. THE COMPONENTS OF THE UNIT OF STUDY .................................................................18
   8.4. EXAMPLES OF EML CODED UNITS OF STUDY ..........................................................20
       8.4.1. Initial example: Problem based learning (PBL) ............................................................ 20
       8.4.2. Reflection on the PBL example ...................................................................................22
       8.4.3. Example: lesson plan 'A Bottle Project for Learning About Graphs' ....................... 22
       8.4.4. Reflection on the lesson plan example .......................................................................25
       8.4.5. More examples ...........................................................................................................25
   8.5. HOW TO PUBLISH EML FILES .......................................................................................26
9. CONCLUSION ........................................................................................................................ 27
LITERATURE USED IN THE ANALYSIS ..................................................................................28

© Rob Koper, June 2001  
Open University of the Netherlands  
Educational Technology Expertise Center  
Valkenburgerweg 177 PO 2960  
6401DL Heerlen  
The Netherlands  
E-mail: rob.koper@ou.nl  
Telephone: +31 455762657
1. Introduction

The title of this article could have been: Where is the learning in e-learning? The promise of e-learning, and the enabling learning technologies, is to make learning experiences in all types of settings more effective, efficient, attractive and accessible to the learners. In e-learning the Internet is used as the core medium for the delivery of information and the support of communication. Most people also think that the Internet, itself, as the key factor in the success of e-learning. However, a vast amount of research provides evidence for the proposition that it is not the medium (Internet), itself, which is accountable for the accomplishment of these promises, but the pedagogical design used in conjunction with the features of the medium (I refer to the classical medium discussions started by Clark, 1983, 1986, 1990, 1999). The message is that we should concentrate on the quality of the pedagogical design and its relationship to the possibilities of the Internet if we want to accomplish the promises of e-learning.

Another common belief is that learning is the same as knowledge transfer. The idea which comes with it, is that it is enough to make knowledge available to learners according to some pedagogical structure. However, providing adequate knowledge is not enough: it has to be learned. It is this learning process that is the process we are putting at the center when we discuss instructional design or learning design, and not the knowledge it works on. Ask yourself: ‘where is the learning’ in e-learning? On top of that, a lot of learning does not come from knowledge resources at all, but stems from the activities of learners solving problems, interacting with real devices, interacting in their social and work situation. A lot of research about learning processes provides evidence for this stance that learning doesn’t come from the provision of knowledge solely, but that it is the activities of the learners into the learning environment which are accountable for the learning.

This is not to say that knowledge objects are not of importance in learning situations, but I say that they are not the key thing in effective learning processes.

In this article I will address the topic of the pedagogical design of learning experiences. Learning experiences are offered mostly in chunks, like courses. These chunks (in the next paragraph we abstract them to the concept of ‘units of study’) are the major delivery units for e-learning. From a design perspective, the course is the aggregate containing all the necessary features to make learning successful. It is at this level that educational modelling (or instructional design/learning design) takes place; it is at this level that the pedagogical models are implemented; it is this level of aggregation that is accountable for the quality of learning.

I will specifically address the analysis of pedagogical models we did in order to provide a meta-model from which we could build a notation for units of study.

2. Learning design is modeling ‘units of study’

In 1998, we started a research project aimed at building a semantic notation for complete units of study to be used in e-learning. The concept of ‘unit of study’ is central to this case. It is the smallest unit providing learning events for learners, satisfying one or more interrelated learning objectives. This means that a unit of study can not be broken down to its component parts without loosing its semantic and pragmatic meaning and its effectiveness towards the attainment of learning objectives. The unit of study could be considered as a gestalt. In practice you see units of study in all types, sorts and sizes: a course; a study program; a workshop; a practical; a lesson could all be considered to be a unit of study.
A unit of study could be delivered through what is called:
- online learning (completely through the web).
- blended learning (mix of online and face-to-face)
- hybrid learning (mix of different media: paper, web, e-books, etc.).

We called the notation of units of study an “Educational Modelling Language\(^1\)”.

Note that our work is twofold:
1. Pragmatic from a user point of view: we think that a complete, integrated framework for
   the notation of units of study is a necessity for providers of education and training (just
   as we are, ourselves).
2. Academic in the sense that we want to search for notations which meet the
   requirements.

3. The learning objects model

In practice, as well as in the literature, the concept of ‘learning objects’ is heavily used but
not strictly defined. The IEEE LTSC (2000) has made a proposal for a standard definition
which is extraordinarily broad. A learning object is any entity, digital or non-digital, that can
be used, re-used, or referenced during technology-supported learning. More restricted
examples of definitions – which are all within the scope of the IEEE LTSC definition – are
also found in literature (Wiley, 2000). Examples of learning objects are: printed materials,
study tasks, exercises, study texts, cases, media assets, courses, study programmes and
also persons. A fundamental idea is that a learning object can stand on its own and may be
re-used. In practice this means that learning objects are mostly smaller objects – smaller
than courses - which can be re-used in different courses. One of the underlying ideas is that
courses in themselves can hardly be made re-usable, because of all sorts of local factors
(see e.g. Downes, 2000). Only some institutions are really successful in course exchange,
but most institutes share learning objects such as text books or geographical maps.

![Diagram of a common view of learning objects and its metadata.](image)

Figure 1. A common view of learning objects and its metadata.

\(^1\) The terms: instructional design, learning design and educational modeling are used
interchangeable. Terms in the XML vocabulary of EML are in UK English.
There are several ways of viewing learning objects. The most common, but often implicit, idea is that of figure 1. This view is in conformance with the IEEE definition.

Learning objects are entities that may be referred to with metadata. The metadata itself are separate from the object it refers to. The metadata, and sometimes the learning objects itself, may be stored in databases. The metadata specification is described in the IEEE LOM (draft) standard specification. IMS provides a binding in XML. In principle learning objects have content (attributes and other learning objects) and descriptions of the behaviour of the learning object (operations). It is clear that the idea of a learning object model conforms to the principles of objects in the theories of object-orientation. This also implies that principles of encapsulation, abstraction and inheritance may be present. Content packaging specifications organize and transfer series of learning objects.

The major question from a perspective of use in real educational practice is: does this model of learning objects and packages provide us sufficient means to build complete, flexible and valid units of study to be delivered through learning management systems?

The answer is clearly 'no'. From an educational perspective it is not enough to have learning objects and metadata as such. Different types of learning objects have different functions in the context of real education. A study task and a study text have both a different function in a unit of study. This is also true for tests and (e.g.) communication facilities. Also, there are different constraints in the relations between different types of learning objects. A study task (a type of learning object), for example, almost always refers to resources (other types of learning objects) needed to perform the task. So there is a structural relationship between tasks and resources within the context of a unit of study.

In our analysis, the major problem with the learning objects model as it is applied until now, is that learning objects are not typed to their usage in the context of a unit of study. To put it in another way; there is a lack of a containing framework. The learning object model expresses a common overall structure of objects within the context of a unit of study, but does not provide a model to express the semantic relationship between the different types of objects in the context of use in an educational setting. As a result, the learning object model also fails to provide for a model of the structure of the content of the different objects. The typing of objects also varies according to different pedagogical stances, so there is a need for a meta-model to describe the relationships. The basic idea, we have elaborated, is to:

1. classify, or type, the learning objects in a semantic network, derived from a pedagogical meta-model,
2. build a containing framework expressing the relationships between the typed learning objects and
3. define the structure for the content and behaviour of the different types of learning objects.

This approach has a lot of advantages, such as the following:

- It supports developers in building valid and high quality units of study, using and re-using smaller components;
- It supports builders of authoring and delivery tools by providing a common framework for valid units of study;
- Learners and teachers can identify and search learning objects, knowing their function within the framework of the course;
- It provides a semantic expression for the content of learning objects, supporting re-use, interoperability and assembly of the components of units of study into different units of study.
4. Requirements for units of study as a result of learning design

Actors in the learning process, dealing with units of study are:

- Learners
- Staff
- Developers of units of study, or the components it refers to like study materials

Besides these direct users of the system there are lots of other different actors in e-learning, specifically all types of managers (system managers, HRM managers, etc.), vendors and publishers.

Also the different roles mentioned can be spit down further to a lot of sub-roles. The role developer may (e.g.) be split by: author, interaction designer, graphical designer, etc.

In our use case analysis of the actor requirements (including the once not mentioned here), they all want four different types of outcomes from e-learning. They want more effectiveness, more efficiency, more attractiveness and higher accessibility. All stakeholders fill these aspects from their own perspective. A learner wants more effective, efficient, attractive and accessible learning; a tutor wants to tutor in a more effective, efficient, attractive and accessible way, and so forth. The translation in general categories of requirements are as follows:

An Educational Modelling Language, which describes a unit of study, must meet the following general requirements:

1. The notational system must describe units of study in a formal way, so that automatic processing is possible (formalisation).
2. The notational system must be able to describe units of study that are based on different theories and models of learning and instruction (pedagogical flexibility).
3. The notational system must explicitly express the semantic meaning of the different learning objects within the context of a unit of study. It must provide for a semantic structure of the content or functionality of the typed learning objects within a unit of study, alongside a reference possibility (explicitly typed learning objects).
4. The notational system must be able to fully describe a unit of study, including all the typed learning objects, the relationship between the objects and the activities and the workflow of all students and staff members with the learning objects (completeness). And regardless of whether these aspects are represented digital or non-digital.
5. The notational system must describe the units of study so that repeated execution is possible (reproducibility).
6. The notational system must be able to describe personalization aspects within units of study, so that the content and activities within units of study can be adapted based on the preferences, prior knowledge, educational needs and situational circumstances of users. In addition, control must be able to be given, as desired, to the student, a staff member, the computer or the designer (personalization).
7. The notation of content components, where possible, must be medium neutral, so that it can be used in different publication formats, like the web, paper, e-books, mobile, etc. (medium neutrality).
8. When possible, a ‘wall’ should be placed between the standards that are used for notating units of study and the technique used to interpret the notation of the units of study. Through this, investments in educational development will become resistant to technical changes and conversion problems (interoperability and sustainability).
9. The notational system must fit in with available standards and specifications (compatibility).
10. The notational system must make it possible to identify, isolate, decontextualize and exchange useful learning objects, and to re-use these in other contexts (reusability).
11. The notational system must make it possible to produce, mutate, preserve, distribute and archive units of study and all of its containing learning objects (life cycle).

5. Architectural reference model

Besides pedagogical perspectives there are other perspectives that are of importance when designing an educational modeling language. Especially the architectural reference model, which provides a conceptual view on the position of the EML within an e-learning architecture. Figure 2 provides a summarized view of the architectural reference model we used.

![Architectural reference model diagram]

Developers work with the development environment. The development environment may consist of design tools, authoring tools and a content management system.

Learners and staff work with one or more delivery platforms (LMS, paper, CD’s, e-books, etc.).

The publisher (or ‘player’ as we sometimes call it) is in essence the converter of EML code to the code which could be used in the delivery platform. It must be able to do two things:

**Dynamic publishing**: Interpret the EML code, to make instances for every person bound to a role in a dynamic way (dependent on the personal dossier and the state of the processes within the unit of study). The format to which the EML code is converted is dependent on the interactive delivery platform (e.g. XHTML and javascript).

**Static publishing**: Make a static interpretation (mostly of parts of the unit of study) for every person bound to a role in the unit of study. This is meant for delivery through static (non-interactive) media like print, LMSs or e-books which can only deal with static HTML content. It enables off line work in general. This content can not be personalized by the user on-the-fly but is pre-published.
The delivery platform for units of study can consist of one or more different media. In real education there is in most cases a mix of different media, like an LMS in combination with printed materials, face to face communication, etc. The publisher must be able to publish different parts of the unit of study to different media on demand.

At this moment in time most LMSs and other publication media are static in their representation of content. We built a piece of middleware (called Edubox) to test the dynamic publishing of EML.

Of importance are the different dependencies among the packages within the diagram. Because of principles of loosely coupled systems, it is a good principle to define the public interfaces between them. Here I will shortly mention a few interfaces which are of importance.

The public interface between development and publishing is a critical one. When there is a common public interface here, you will meet a lot of the requirements (re-use, interoperability, etc.). That’s one of the places where we position the EML specification. Not only because of its semantic expression, but because of the interface capabilities of XML vocabularies in general. EML also is used for the interface to developers in terms of helping them produce valid units of study (through the use of tools). Here the semantic aspects are more predominant.

The interface between publisher and delivery platform is dictated by the delivery platform (e.g. websites want HTML, book publishers want pdf, etc.).

The interfaces between dossier and publisher/delivery platform/user is an extra concern, partly addressed by the learner information specification, enterprise specification and the competency definitions. This work is however not harmonized yet to a general interface specification to learner dossiers.

6. The pedagogical meta-model

What is a pedagogical meta-model? In our view it is a model which models pedagogical models. This means that pedagogical models could be described (or derived) in terms of the meta-model. This is of importance when you want to express semantic relationships between pedagogical entities and want to be pedagogical neutral. Compare this for instance with a text editor like MS-Word. MS-word is neutral to the type of text you can edit with it. It is possible to write memo’s, letters, poetry and literature. That’s flexibility. However, MS-word has no real means of validating whether the text you have edited really is a poem or a letter, given all the varieties of poems and letters in practice. You can imagine what great help it would be when these types of tools are aware of the type of content you are editing. You could expect a lot more support in the writing process than you get now. Text writing has so much varieties in practice that a real semantic framework for texts in general is not available at the moment, and it maybe never will be. However, education is a more restricted domain with a lot of commonalities in its instances. This is mainly due to the hard work done in research into learning and instruction. There are still a lot of different stances when answering questions about learning, but there are also a lot of commonalities. These commonalities are the focus of a meta-model, the differences are made by parameterization of the meta-model. This idea has led us to the work on the meta-model behind EML.

The main topics of the static structure of the pedagogical meta-model is expressed in UML diagrams here. The pedagogical semantics of EML are designed according to this model. The model is based on educational research, specifically in the field of learning psychology and instructional design (see literature). Most of these models in literature are expressed in natural language and ad hoc schemas. I have drawn the UML model just for this example so as to focus on the important aspects. Most of the classes have more elaboration.
Like all models this model abstracts reality. It must not be confused with the reality itself and it is not the only model possible describing learning from instruction. This is also true for learning design in general. Course designs are something different from what actually happens when courses are instantiated and used in real practice. It is not the intention of course designs to abstract all the details of the course, but its major points. Also, what I have drawn in the UML diagrams are expressions of the pedagogical models underlying units of study. It highlights the important points. In its details of implementation the models have more complexity.

First I will draw the major packages of the pedagogical meta-model (figure 3).

![Figure 3. Packages in the metamodel.](image)

There are four packages:
1. The learning model, which describes how learners learn based on commonalities (consensus) in learning theories.
2. The unit of study model, which describes how units of studies which are applicable in real practice are modelled, given the learning model and given the instruction model.
3. The domain model, which describes the type of content and the organization of that content. For example, the domain of economics, law, biology, etc.
4. Theories of learning & instruction, which describe the theories, principles and models of instruction as they are described in literature or as they are conceived in the head of practitioners.
6.1. Part 1: The learning model

Figure 4 provides a summary of the learning model.

![Diagram of the learning model]

Figure 4. The learning model.

The learning model is based on the following axioms:
1. A person learns by (inter-)acting in/with the external world.
2. The real world could be considered to be composed of social and personal situations, which provide the context for actions.
3. A situation is composed of a collection of things and living beings in a specific interrelationship.
4. One part of situations are communities of practice and – more specifically – learning communities.
5. There are different types of learning, the one of interest to us is learning invoked by instructional measures.
6. Learning can be considered to be a change in the cognitive or metacognitive state. However, changes in the connotation and affection can also be considered as the result of learning. When a person has learned he or she can a) carry out new interactions or carry out interactions better or faster in similar situations, or b) carry out the same actions in another situations (transfer).
7. A person can be urged to carry out specific interactions, if:
   - a person is willing to do so or stimulated to do so (conation / motivation factor);
   - a person is able to do so (cognition factor);
   - a person is in the mood to do so (affection / emotional factor);
   - a person is in the right situation to do so (situational factor);
8. What has been set out here regarding an individual is also valid for a group of people or an organization, even though this does not have to be reducible to individuals.
The essence here is that no value judgment is made in these axioms about the following questions:

1. What does a person or group learn (knowledge, competencies, skills, insight, attitudes, intentional behavior) and in which domain?
2. What kinds of activities must be carried out to learn? For example: observing, describing, analyzing, experiencing, studying, problem solving, experimenting, predicting, practicing, exploring and answering questions.
3. How should a learning situation be arranged (context, which people, which objects) and what relationship does the situation have to the teaching-learning process?
4. To what extent are the components of the situation present externally and to what extent are they represented cognitively-externally?
5. How, precisely, do the learning and transfer processes occur?
6. How is motivation stimulated?
7. How is the learning result captured?
8. How should activities be stimulated?

The answers to precisely these questions determine the educational philosophy, the instructional model and the more practical design of the units of study. The meta-model provides the semantic framework for the units of study's notational system, alongside the structure of learning environments that was dealt with earlier.

A citation from Duffy & Cunningham (1996, p. 171) in this area: 'As the quote from Skinner suggests, everyone agrees that learning involves activity and a context, including the availability of information in some content domain. Traditionally, in instruction, we have focused on the information presented or available for learning and have seen the activity of the learner as a vehicle for moving that information into the head. Hence, the activity is a matter of processing the information. The constructivists, however, view the learning as the activity in context. The situation as a whole must be examined and understood in order to understand the learning. Rather than the content domain sitting as central, with activity and the 'rest' of the context serving a supporting role, the entire gestalt is integral to what is learned.'

6.2. Part 2: The unit of study model

Figure 5. describes the unit of study model.
A model for a unit of study is the result of a learning design process in which a real product (the unit of study) is the result. It must take into account issues such as:
- the roles of staff and learners in the learning process
- the learning objectives and target group
- the prerequisites of the learners
- other learner characteristics (learning styles, preferences, situational circumstances, etc.).
- the domain of learning (e.g. mathematics is different from cultural sciences)
- the context of learning (distance learning, blended learning, support structure available, library, etc.)
- the assessment of learning

In this model terms like ‘activity’ and ‘environment’ are used as counterparts for ‘action’ and ‘situation’ in the learning model. However, in the unit of study model, they refer to planned activities and environments. In essence this is the difference between the two models: the unit of study model deals with the design of learning processes and the learning model deals with the way learning takes place in real.

In EML we must take care that all these different information categories could be described in meaningful semantic terms and not restricted to one of the views of teaching and learning models.

For the vocabulary of the EML the meta-model is used to model the vocabulary.

### 6.3. Part 3: The domain model

Every pedagogical model must take into account the characteristics of the content domain. Content domains are e.g. mathematics, cultural science, economics, psychology, electrical engineering, law, etc. Every content domain has its own structuring of knowledge, skills and competencies. There are different cultures and communities of practice. Often there are also specifically designed pedagogical models for the domain. For instance in mathematics teaching.

### 6.4. Part 4: Theories of learning and instruction

Figure 6. provides a model of the generalization relationships between instruction models.

![Theories of learning & instruction](image)

Figure 6. Theories of learning & instruction
In educational technology, there are different streams in which the characteristics appear to have what Thomas Kuhn (1962) describes as scientific paradigms. Greeno, Collins & Resnick (1996) make – in a meta-analysis – a distinction between three major streams of instructional theories:

1. empiricist (behaviorist)
2. rationalist (cognitivist and constructivist)
3. pragmatist-sociohistoric (situationalist)

All stances have different views on topics such as: knowledge, learning, transfer and motivation. I will shortly address some of the differences:

According to the empirical approach, as typified by Locke and Thorndike, all reliable knowledge is based on experience. Locke says: 'There is nothing in the mind that was not in the senses.' The assumption is that behaviour is predictable, given he specific environmental conditions, and that processes can be analysed in isolation. The idea is that learning can influenced outside of its context and without knowledge of the internal learning processes.

In the rationalist approach, as typified by Descartes and Piaget, thinking is considered the only reliable source of knowledge. In this case, it is supposed that cognition mediates the relationship between a person and the environment. As there is the possibility of large individual differences in cognitive processing, for example, because of differences in prior knowledge (Dochy, 1992), meta-cognition (Flavell, 1979; Brown, 1980), motivation (Malone, 1981) and learning styles (Vermunt, 1996), the assumption of predictable behaviour falls away, and those involved must work with more open, authentic environments in which students themselves can build knowledge. The student is given a central, self-managing role in the educational process (Shuell, 1988; Schunk & Zimmerman, 1994).

The third approach is called the pragmatic and cultural-historic approach, as typified respectively by James, Dewey and Vygotsky, Leont'ev, or in educational theory as social constructivism (Simons, 1999). In this approach, the situation and the cultural-historical context that a learner is in are given primary attention (Lave & Wenger, 1991; Cole & Engestrom, 1993). Knowledge is distributed among individuals, tools and communities, such as those of professional practitioners. The assumption is that there is collective as well as individual knowledge. Learning is considered as the adaptation of behaviour to the rules of the community. An important instrument for adapting and acquiring common views is discussion and cooperation in the communities.

According to most scholars and practitionars, these streams, or stances, are supplementary and offer different perspectives on the same themes (see also: De Boer, 1986; Molenda, 1991; Greeno, Collins & Resnick, 1996; Sfard, 1998; Jonassen, 1999; Roblyer & Edwards, 2000). Just as psychology, economy and biology look at human behaviour in different ways.

Based on these stances there are – in literature - descriptions of hundreds of more theoretical or practical theories and models of learning and instruction. To name but a few: competency based learning, project based learning, mastery learning, problem based learning, case based learning, experiential learning, action learning, etc. (see literature like Reigeluth, Merrill, Jonassen, Kearsly). Also lots of more informal teaching plans are available (see e.g. Eric’s lesson plans at: http://ericir.syr.edu/Virtual/Lessons/). Another approach is based in human resource management, mostly referred to as performance improvement (sometimes human performance technology, see Stolovitch & Keeps for an overview).

We studied and analysed most of these models. We mapped the commonalities and listed the differences in order to derive the meta-model.
We have also added a fourth type of model: the eclectic model. These are instructional design models using principles from different stances, just for the practical occasion. These models can be explicitly formulated, but mostly they are implicit.

6.5. An integrated picture of the meta-model

The integrated picture of the meta-model could be drawn as in figure 7. Most of the specializations are kept out of the model. The focus in the model is also on the learner and not on the role of staff. It is drawn here to trace the dependencies within the model. A more exact picture is provided in the package diagram and the elaboration of the different packages.

---

**Implied elements**

Not all pedagogical models address all elements in an explicit way. In our view these elements are kept *implicit*. For instance: there are learning management systems which doesn’t provide activities to learners and/or staff. This can mean two things:

- The activities are implied, the students have to find them out themselves. Mostly this is the case with classical forms of education with a lot of standard, quite evident, taks, such as: “read this book”, “solve this problem”, “answer the questions”.
- The activities are not implied, but they are not part of the course offered through an LMS. The idea is that teachers will set the activities for students. This is the case in classroom situations. The LMS only serves some environment functions like communication facilities and learning resources. In this case the LMS cannot support units of study, only parts of it to be integrated by the teachers. The LMS isn’t really a platform for all e-learning situations.
7. Types of learning objects

Given the analysis above we typed different learning objects in several categories (see table I).

Table I: Types of learning objects in the context of a unit of study.

**Legend**
- In the first column the learning object type is mentioned. The second column provides the standard abbreviation we use. The third column gives the multiplicity in UML notation, that means the possible occurrences within the context of a unit of study (0..* means: zero or more).

<table>
<thead>
<tr>
<th>Learning object type</th>
<th>Abbr.</th>
<th>Multiplicity</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>Unit of study</td>
<td>uos</td>
<td>1</td>
<td>The containing framework (the ‘root’)</td>
</tr>
<tr>
<td>Learning objective</td>
<td>lo</td>
<td>1..*</td>
<td>The intended output of learning the uos</td>
</tr>
<tr>
<td>Prerequisite</td>
<td>pr</td>
<td>0..*</td>
<td>Prior knowledge, situational circumstances, needed to study the uos</td>
</tr>
<tr>
<td>Role Learner</td>
<td>rl</td>
<td>1..*</td>
<td>The persons intended to study the uos have the role learner. This role could be subdivided in sub roles, as is needed in some pedagogical models like instructional games and simulations</td>
</tr>
<tr>
<td>Role Staff</td>
<td>rs</td>
<td>0..*</td>
<td>The persons interacting with learners in order to support their learning process. This role could be subdivided in sub roles, e.g. tutors, assessors, trainers, quality assurance. In self study packages there is no tutoring staff role, but evaluation and monitoring staff roles may be present.</td>
</tr>
<tr>
<td>Property object</td>
<td>prop</td>
<td>0..*</td>
<td>Properties contain the information stored in the dossiers of persons and roles in order to be able to monitor progress, enable personalization (e.g. adapt on learning styles), store preferences, etc. Property objects may be contained in property-groups.</td>
</tr>
<tr>
<td>Activity</td>
<td>act</td>
<td>1..*</td>
<td>The prescription of the actions to be carried out by the different roles. Different subtypes: learning activity, support activity and instrumental activities. Activities have different states, such as ‘completed’). Activities may contain learning objectives, prerequisites and environments.</td>
</tr>
<tr>
<td>Activity structure</td>
<td>str</td>
<td>0..*</td>
<td>When there is more than one activity for a role, a structure of sequences or selections of activities could apply. The structure also identifies which environment goes with which (set of) activities.</td>
</tr>
<tr>
<td>Environment</td>
<td>env</td>
<td>1.*</td>
<td>An activity is carried out in an environment, which is a collection of resources with some sort of ecological relationship. Environments may contain sub environments (building-room).</td>
</tr>
<tr>
<td>Knowledge object</td>
<td>ko</td>
<td>0..*</td>
<td>Contained by env. All objects with coded information in it, having a knowledge transfer function in the context of the activities. Examples: study books, fact sheets, manuals, dictionaries, encyclopedia, etc.</td>
</tr>
<tr>
<td>Section object</td>
<td>sect</td>
<td>0..*</td>
<td>Section objects contain typed content and may be contained by all other objects which have content. It is the smallest re-usable component in a uos. For instance. An activity has a description. This description may be build from paragraphs of text or sections of paragraphs, like: introduction, what to do, when to do, etc. Sections may contain sections.</td>
</tr>
<tr>
<td>Special object</td>
<td>spec</td>
<td>0..*</td>
<td>All content may contain special objects. These objects identify content types which may be shown or hidden for certain users or under certain conditions.</td>
</tr>
<tr>
<td>----------------</td>
<td>------</td>
<td>------</td>
<td>----------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Communication object</td>
<td>co</td>
<td>0..*</td>
<td>Contained by env. All facilities, synchronous or asynchronous which are used to contact other persons or roles within the uos. Example: email, telephone, video/audio conferences, discussion rooms, etc.</td>
</tr>
<tr>
<td>Tool object</td>
<td>to</td>
<td>0..*</td>
<td>Contained by env. These objects are used in order to master them, or to support certain functions in the learning or support process. Examples: computers, trees, laboratory inventory, like test tubes, turning lathe.</td>
</tr>
<tr>
<td>Index object</td>
<td>io</td>
<td>0..*</td>
<td>Contained by env. Provides an index of the objects contained in the environment, contained in a uos or contained. There are different types of index (free text, table of content, terminology index, etc).</td>
</tr>
<tr>
<td>Search object</td>
<td>so</td>
<td>0..*</td>
<td>Contained by env. Related to a index. Within the scope of an index queries are possible which return a result list.</td>
</tr>
<tr>
<td>Role-information object</td>
<td>roli</td>
<td>0..*</td>
<td>Contained by env. Provides information of other persons and other roles within the uos. E.g. for tutors who want to monitor the progression of groups or want to access uploaded work from learners.</td>
</tr>
<tr>
<td>Questionnaire object</td>
<td>qo</td>
<td>0..*</td>
<td>Contained by env. This object contains the test items and logic for tests and questionnaires. It contains an item model like the IMS QTI (not only with reproductive but also productive items) and the score model. Items (mc questions, short answer questions, etc. are separate objects not mentioned here further).</td>
</tr>
<tr>
<td>Announcement object</td>
<td>ao</td>
<td></td>
<td>Contained by env. Messages providing actual information for learners or staff within the context of a uos or a certain activity.</td>
</tr>
<tr>
<td>Play object</td>
<td>po</td>
<td>1..*</td>
<td>An object containing the script (dynamics) in the uos: which role performs which activity, activity structure or sub-uos? Its model is comparable to the model of activity diagrams within swim lanes (representing the roles) in UML.</td>
</tr>
<tr>
<td>Condition object</td>
<td>cond</td>
<td>0..*</td>
<td>Series of condition statements (if, then, else) used to hide and show certain content types or activities. The If part refers to the state in the uos or of the person/role as it is reflected in the dossier. Example: when a role wants to hide technical examples (content type), this is handled in the conditions.</td>
</tr>
</tbody>
</table>

**Subtypes to implement pedagogical models**

All these typed objects may be subtyped to the terminology used into a certain pedagogical model. For instance, An activity may have the type of: introduction, problem analysis, search problem, monitor, etc. It is in this subtyping that the pedagogical models are implemented in the meta-model of EML. This typing is free for a designer to choose.

**Content structures**

Besides the above mentioned learning object types, we identified two types of content structures:

1. Extra P. Series of entities like paragraphs, special content types, audio-visual components, tables, formulas, etc. Needed for the structuring of content.
2. Intra P. Series of entities contained within a paragraph, like emphasis, inline pictures/formulas, etc.

This part of EML is not pedagogical in nature. The structure is derived from both HTML and the DOCBOOK content structures. The CALS table structure is for instance part of it.
Identifying re-usability

Different objects have different abilities for reuse in different context. Activities, knowledge objects and even units of study can be designed in such a way that they are re-usable in other contexts. In some situations, learning objects are bound to the context they are used in and are not re-usable. Some authors even hold the position that components must not be written re-usable at all. They find it artificial: real and good study text have a lot of internal references. For this reason we thought it is necessary to provide a mechanism to identify which learning objects are re-usable and which are not. This also means that re-usability starts with the design: which objects do you want to design to be re-usable? The others do not matter and should not receive any further attention, since they are identified as being 'not re-usable'.

8. The implementation of the meta-model in an XML schema

8.1. XML schemas

We implemented the idea's about educational modelling in an XML schema. There are two possibilities for doing this:

1. Include relationship information only in the learning object metadata. The LOM provides facilities for this.
2. Implement the whole semantic logic in the schema, naming the learning objects to its type.

We tried both options in the beginning, but soon it came clear to us that only the second option provided enough strength to express the whole model of relations. In the first option you have to make an additional schema language that will fit into the LOM specification. There are no tools available for editing a validation, and you don’t use the strength of XML to express these types of relationships. On the other hand, the schema type we used (dtd’s) also have their weaknesses in expressing some kinds of constraints. The new XSD recommendation from the W3C does a better job. DTD’s have the advantage of being rather simple and can be used better for communication purposes.

8.2. The basic structure of EML

The basic structure of EML is shown in a tree view in figure 8. I have shown, for clarity, only a selection of the elements and relationships and no attributes. The complete schemas can be downloaded from [http://eml.ou.nl](http://eml.ou.nl). What do you see:

- The tree structure of the containing framework of learning objects.
- Every learning object is mentioned after its type name (every learning object also has a ‘type’ attribute to identify its role in the pedagogical model used).
- The tree is made up of components which could be exist (and edited) on their own. For instance: metadata, activities, knowledge objects, test items (not shown here).
- Parameter entities have been deleted to improve the readability.
- The relationships between learning object is expressed in the framework within the possibilities provided by XML: ? means optional; * means zero or more; + means one or more.

Notation: The \[\square\] fork represents a sequential list of elements, and the \[\land\] fork represents a selection of one of the elements.
8.3. The components of the unit of study

Figure 8 shows the framework, that is the relationship of all types of learning objects within the framework of a unit of study. However, every component mentioned in the typed learning object table can stand on its own. This means: edited, stored and published on its own, without the containing framework it fits in. For example the knowledge object or activities may be edited, stored and published as separate entities. But when you do so it does not mentioned education but, for instance, an activity list or a series of knowledge objects (a series of chapters in a book for instance). An important point here is that series of knowledge objects or activities is not enough for learning. A valid unit of study contains more!

Figure 9 provides the structure of a knowledge object.
Figure 9. Structure of a knowledge object in EML

There are two possible situations when dealing with content:
1. The source of the knowledge object is outside EML (mostly at the Internet) and only a reference is provided by means of an URI.
2. The source of the knowledge object is also in EML (this has a lot of pros and cons, but this discussion is out of scope here).

EML supports both situations, as was mentioned earlier. When the type of learning object is known, one is able to define a specific content model for that object. We did so for all the learning objects in EML. At the same time, EML supports the model when the content is kept completely separate from the containing framework. In our practice, most people use the EML content model when authoring new materials and use references to materials when there are existing learning objects. In the first model the accessibility is guaranteed, in the second situation there are lots of accessibility problems.

For us we prefer the source to be in EML, because it provides us with a way to:
1. Fully manage a complete unit of study, with all of its content in a medium-neutral way;
2. Have personalization and other dynamics work on the source of knowledge objects.

In order to be flexible in the editing tools, we built converters from RTF to EML text.
8.4. Examples of EML coded units of study

I will provide some short examples here, focusing on the topic of implementing pedagogical models in EML. I will summarize the models and its coding to its essence. Texts are very short and most objects are referenced to make the examples shorter.

8.4.1. Initial example: Problem based learning (PBL)

Problem based learning (PBL) has been described by Barrows and Tamblyn (1980). It is applied in a lot of sciences, but the major followers are in the medical disciplines. Some schools and universities adopted the model throughout their curriculum. In PBL, students are actively involved with problems coming from real practice. For instance cases of patients with medical problems to be solved. Learning of facts, like anatomy and pathology is always done in the context of these problems. The learner works within a small group of students. There are no lectures. There is no tutor, but a coach. The coach could be a more senior fellow student. The coach takes care of the process when analyzing problems and learning from them. A fixed procedure is followed for all types of problems. Some institutes follow the next 7 steps:

Given a description of a problem situation:
1. Identify concepts and parts of the problem that needs clarification
2. Define the problem
3. Analyze the problem, brainstorm about solutions or causes
4. Structure solutions or causes
5. State learning objectives
6. Self study directed towards learning objectives
7. Report things learned and application to the problem

Different forms of assessment can be applied in the framework of PBL. In one case, assessment is not done on a problem for problem basis, but on the competencies which should be acquired to show mastery in the field. The idea is to test students several times a year on the same test. When they score positively on the test, they pass the examination. In most cases however, only the progress from year to year is measured.

An EML implementation of PBL can have the following structure:

```
<Unit-of-study Id="c1234" Type="PBL problem">
  <Metadata><Title>Patient with pain in one leg</Title></Metadata>
  <Roles><Learner Id="student" Max-number="8"/> <Staff Id="coach"/></Roles>
  <Content>
    <Activity Id="coach-instruction" Type="PBL support activity"><!-- activity for coach -->
      <What>Guide students through the steps. Don’t explain, only react on process.</What>
      <Completed><Unrestricted/></Completed>
    </Activity>
    <Activity Id="step1" Type="PBL Identify">
      <What>Read the problem description and identify concepts you don’t know. Communicate in the discussion group.</What>
      <Completed><User-choice/></Completed>
    </Activity>
    <Activity Id="step2" Type="PBL Define">
      <What>Define the scope and type of the problem.</What>
      <Completed><User-choice/></Completed>
    </Activity>
    <Activity Id="step3" Type="PBL Analyse">
      <What>Analyze the problem and brainstorm with peers about solutions or causes.</What>
      <Completed><User-choice/></Completed>
    </Activity>
    <Activity Id="step4" Type="PBL Structuring">
      <What>Structure the solutions or the causes.</What>
      <Completed><User-choice/></Completed>
    </Activity>
  </Content>
</Unit-of-study>
```
The major containers of EML are colored in order to navigate more easily through the code. There are several ways to read EML code. I find the easiest way to look at the method section first: it summarizes the pedagogical model, without having content in it. Within the method you must start with the <play> structure. Backward tracking you can find all other
elements. For instance this play says: The role “student” gets an activity structure called “str-student”. At the same time (otherwise it would have been separated with a continue statement), the role “coach” gets an activity structure called “str-coach”. You can look for the structure of “str-student” somewhat higher: it consists of an environment (env-students) and a sequence of eight activities (step1..7 and a assessment). To look at the content of these activities look in the <content> section. To see the resources needed for the completion of the activities, look at the <environment> section within content.

8.4.2. Reflection on the PBL example

From this example it is clear that most of the pedagogical structure of a unit of study can be seen from the method section. In EML, it is not necessary to include the content itself (it is optional in a unit of study), it is enough to include a reference to the activity and environment objects. Figure 10. provides the minimum structure of EML, including obligatory objects only. Theoretically, this is the real kernel of EML: the smallest valid unit of study. However, the optional extensions are almost always needed when coding real units of study, to be used in real practice.

8.4.3. Example: lesson plan ‘A Bottle Project for Learning About Graphs’

This example comes from Neiss, (1995) and is included in a database of lesson plans from Roblyer & Edwards (2000). I have selected it randomly from the database in order to get a kind of regular example. It is first stated in text and than mapped into EML.

---

**Smallest valid EML model of a unit of study**

```xml
<Unit-of-study>
  <Metadata>
    <Title>Course on X</Title>
  </Metadata>
  <Roles>
    <Learner Id="learner"/>
  </Roles>
  <Method>
    <Play>
      <Role-ref Id-ref="learner"/> <Activity-ref Worldwide-unique-id-ref="default-student"/>
    </Play>
  </Method>
</Unit-of-study>
```

**Smallest valid EML instance of a unit of study**

```xml
<Unit-of-study>
  <Metadata><Title>Course on X</Title></Metadata>
  <Roles><Learner Id="learner"/></Roles>
  <Method>
    <Play>
      <Role-ref Id-ref="learner"/> <Activity-ref Worldwide-unique-id-ref="default-student"/>
    </Play>
  </Method>
</Unit-of-study>
```
Lesson description
Objective: Learn to use graphs to communicate and interpret numerical information.
Target group: grade level: Middle school;
Grouping: whole class
Theory of learning and instruction: constructivist
Content domain: mathematics
Topic area: measurement
Technologies: spreadsheets

Lesson direction
In this activity, students do a hands-on experiment with bottles to learn how to use graphs to communicate numerical information.

1. Preparation - Obtain the required materials: some clear glass bottles of different sizes and shapes, a graduated cylinder for measuring volume in milliliters, rule for measuring height in millimeters, and some water. Each student or small group of students gets one of the bottles. Have them use the Draw mode of an integrated software package such as ClarisWorks to draw a picture of their bottle (see Figure 11-1). The teacher prepares a spreadsheet to display the data. (See example in Figure 11-2).

![Figure 11-1. Pictures of student-drawn bottles.](image)

![Figure 11-2: Spreadsheet Showing Heights of Bottles](image)
2. Hands-on experiment - Have students pour 10 ml of the water into their bottle and measure the height of the water in the bottle. The students enter the data in a spreadsheet that the teacher has prepared. Have them continue adding water in 10 ml increments, measuring the height and recording the data as they go.

3. Graphing and analyzing the data - Have students use the spreadsheet’s charting mode to graph the data, and use cut and paste to transfer the spreadsheet data and the graph to a word processing file. The teacher begins a discussion of how the shape of the bottle relates to the graph. For example, if a bottle has straight sides, the graph is a straight line. Have them summarize what they have learned about the relationship between a bottle and the shape of the graph.

4. Analyzing “mystery bottles” - Give the students sets of data already entered in the spreadsheet and ask them to determine the shape of the bottle from the graph. Encourage students to use the term “slope” between the sets of points in their discussions. Finally, ask students to predict and sketch graphs for various shapes of bottles that have not yet been filled with water.

Required resources: Integrated software package with drawing and spreadsheet software; clear glass bottles of various sizes/shapes; graduated cylinder for measuring volume in ml; ruler for measuring height in mm; and water.

The EML mapping can be done as follows:

```
<Unit-of-study Id="c435" Type="Lesson plan">
  <Metadata>
    <Title>A Bottle Project for Learning About Graphs</Title>
    <Meta Base="Roblyer&Edwards" Description="grade level">middle school</Meta>
    <Meta Base="Roblyer&Edwards" Description="theory of learning">constructivist</Meta>
    <Meta Base="Roblyer&Edwards" Description="grouping">whole class</Meta>
    <Meta Base="Roblyer&Edwards" Description="content domain">mathematics</Meta>
    <Meta Base="Roblyer&Edwards" Description="topic area">measurement</Meta>
    <Meta Base="Roblyer&Edwards" Description="technologies">spreadsheets</Meta>
  </Metadata>
  <Roles>
    <Learner Id="student"/>
    <Staff Id="teacher"/>
  </Roles>
  <Learning-objectives>
    In this activity, students do a hands-on experiment with bottles to learn how to use graphs to communicate numerical information.
  </Learning-objectives>
  <Method Type="lesson plans conform Roblyer & Edwards">
    <Play>
      <Role-ref Id-ref="teacher"/>
      <Activity-ref Id-ref="Prepare"/>
      <Role-ref Id-ref="student"/>
      <Activity-ref Id-ref="Draw-bottles"/>
      <Continue><When-competed/></Continue>
      <Role-ref Id-ref="student"/>
      <Activity-ref Id-ref="Hands-on-experiment"/>
      <Continue><When-competed/></Continue>
      <Role-ref Id-ref="student"/>
      <Activity-ref Id-ref="draw-graph"/>
      <Role-ref Id-ref="teacher"/>
      <Activity-ref Id-ref="ask-questions-about-graphs"/>
      <Continue><When-competed/></Continue>
      <Role-ref Id-ref="student"/>
      <Activity-ref Id-ref="summarize-learning-experiences"/>
      <Continue><When-competed/></Continue>
      <Role-ref Id-ref="teacher"/>
      <Activity-ref Id-ref="ask-questions-about-bottle-shapes"/>
      <Role-ref Id-ref="teacher"/>
      <Activity-ref Id-ref="ask-questions-about-graphs"/>
      <Role-ref Id-ref="student"/>
      <Activity-ref Id-ref="predict-shapes"/>
    </Play>
  </Method>
</Unit-of-study>
```

The lesson plan database contains its own metadata. In EML we thought it would be wise to integrate an extension mechanism for all sorts of structured and unstructured metadata; given that there are still a lot of different existing metadata schema’s around and that the metadata specifications like the IEEE LOM are/where still changing.

The <content> section with the texts and the resources has been left out; they are rather straightforward. The environment has references to Tool-objects like bottles and drawing tools. These tools are not electronically available, but physical. Furthermore there is a communication-object, synchronous with the teacher and the students as participants. The activity descriptions are worked out in text like the descriptions given before.
Again: read the method section. This one is different from the previous one. There is only one play synchronizing activities of the two roles over time. It says: teacher and student start each with an activity: prepare and draw-bottles. When they are finished (the continue statement), the next series of activities will be active. In this case only one, the hands-on experiment performed by students. When they are done, two activities are active simultaneously: the student draws graphs and the teacher asks questions to the students about the graphs. Then the students are asked to summarize the learning experience. And do on.

8.4.4. Reflection on the lesson plan example

There are thousands of lesson plans like this one around. Several databases are maintained (like the Eric clearinghouse lesson plan database at http://ericir.syr.edu/Virtual/Lessons/). The example has the typical design seen most often in the context of classroom teaching. These designs are not very easy to translate to delivery via e-learning. Think what happens when the EML file gets published. It will work, but it won’t be an optimal design for the situation. I didn’t optimize it either; I translated the example rather literally. However, when you look at the example carefully, you see that the example provides only a task for teachers. Taking the text literally will not result in a valid unit of study, because a learner role is obligatory in a valid unit of study. The student activities are implied in the description and I have distilled them in order to be able to format the unit of study.

When you want to optimize this example for e-learning, several design questions have to be answered:
- Is the role of the teacher as defined really needed?
- Can the designer/computer can take over activities like: ask questions about graphs?
- The student can prepare the experiment themselves (get bottles and ruler).
- The spreadsheet template could have been developed in advance and distributed and shared as an object in the environment.
- Do students collaborate or work in isolation with the teacher. Implicit in the example is that peers see the work of others and hear the reactions of the teachers to their work.

In this example you see that:
1. It is possible to map the pedagogy of the a classroom lesson plan to EML;
2. This creates valid units of study, which can be delivered through a LMS;
3. There are, however, different alternatives in the design of the unit of study, the one more optimal than the other. In these questions EML doesn’t help. You need a theory of learning and instruction to decide in these situations.

Don't think that 'a theory' is always a heavy thing, that is never used in practice. What I refer to is – besides the formal described theories from educational research – all kinds of ideas and principles a designer uses as it comes from his or her experience.

8.4.5. More examples

In order to illustrate all the aspects of EML, a lot of different examples can be given. We tested it for instance on topics like:
- personalization
- competency based learning
- portfolio’s and personal development plans
- collaborative learning and learning communities
- individual study plans
- advanced workflows in distance teaching
- new ways of testing and assessment
- prior knowledge tests and intake assessment
- integration of learning with work

((I will work out some more examples in the next version of this text.)).
8.5. How to publish EML files

This topic is a little bit out of scope here, but I think most of you will ask this question now. There are several ways of doing this. Recall the architectural reference model I discussed before. The publisher converts the EML files to the publication format needed. There is a distinction between static and dynamic publishing. The latter converts only the requested parts at runtime, given the state of the unit of study and the information in the dossier of the user. I will sum up some possibilities of publication formats:

1. Publish it to a mail or groupware environment as the delivery environment. In this case every person in a role of a certain instance (run), get e-mails according to the design with the resources of the environment attached to the e-mails. This also works the other way around: people get mail forms to input information into e.g. the dossier or to steer the workflow.
2. An alternative way would be to publish it to a workflow management system. Works about the same as above.
3. Publish to a web-browser or a web portal. In this case facilities for mail and conferencing has to be available and linked. This publishing could be static or dynamic.
4. Import (and export) EML files into a Learning Management System. Learning Management Systems provide easy, but mostly rather restricted ways of editing, storing and the deployment of content. Study progression and preferences are stored in a dossier. The best way to integrate is through import and export filters. Only the functionality of EML is used which is supported by the platform. This is mostly a subset, but it still provides user friendly tools and the possibility to export the whole logic and content of courses in EML and use it into another platform.
5. Publish EML to postscript (and then PDF) or another format needed for publishing on demand. Use in print or e-books. Personalized or not. There are several ways of doing this. Our R&D work the coming period is focused at this problem. Every role gets his or her owns printed materials with activities, resources, forms and procedural descriptions.

We developed EML in conjunction with the development of a dynamic publisher in order to be sure that the EML code can be processed automatically. A lot of functionality and features of EML came from these tests in practice. The most advanced system at the moment converts EML to XHTML, declares database formats and properties and writes Jscript to handle the database access. It also integrates the communication facilities and media specific files. This works as a kind of middleware which can be integrated in every web browser call. I will only show a screenshot here of one of our English demonstration courses about Jazz (figure 12).
9. Conclusion

In this article the pedagogical meta-model behind EML is presented. In our analysis the current thinking about 'learning objects' has some shortcomings. These were addressed and a containing framework for typed learning objects was provided. This framework ensures that the structure of the units of study used in e-learning is valid. However, this is at itself not the same as an effective, efficient and attractive pedagogical design; whether a design conforms to these criteria comes for a large part from the theories and principles of learning and instruction. These theories form the basis for the design of the meta-model behind EML. On the other hand, the designs themselves are not enough to guarantee high quality designs. They tend to be defined at too abstract a level, not providing enough details for the real structuring work that must be done when developing real units of study.

EML makes the use of pedagogical models explicit. This is one of the factors needed to enhance the quality of a pedagogical design. So the combination of good design and good structuring of the design in a notation will bring us the quality of learning we are searching for. EML provides the framework to notate and communicate the designs in a complete form, validate them on completeness in structure, makes it possible to identify the functionality of learning objects within the context of a unit of study and provides means for real interoperability and re-usability. Moreover, we think that EML can make the building of learning management systems easier (because the requirements are explicit) and can make learning management systems more effective, because the design of the systems can take advantage of the huge body of knowledge available in educational research, based on theories, empirical findings and the experience from practitioners.
Literature used in the analysis


Dunn, R.S. & Dunn, K.J. (1979). Learning styles/teaching styles: should they, can they, be matched? Educational leadership, 36, 238-244.


Gass, R.S. & Dunn, K.J. (1979). Learning styles/teaching styles: should they, can they, be matched? Educational leadership, 36, 238-244.


