CEN/ISSS WS/LT
Learning Technologies Workshop

"Survey of Educational Modelling Languages (EMLs)"

Version 1
September 19st 2002

Authors:
Adrian Rawlings, Peter van Rosmalen, Rob Koper (OUNL)
Miguel Rodríguez-Artacho (UNED)
Paul Lefrere (UKOU)

This report has been produced under the auspices of the CEN/ISSS Learning Technologies Workshop. It has been based upon input from Workshop members.

The report has been agreed upon by its authors. Giving that this document is not a CWA, the Workshop members have not been asked to approve of its contents.

This report has been produced by the EML project team, sponsored by the CEN/ISSS Workshop on Learning Technology (CEN/ISSS WS-LT)

EML working group

Prof. Rob Koper, Open University of the Netherlands (OUNL), Chair

Dr. Miguel Rodríguez Artacho, Universidad Nacional de Educación a Distancia (UNED)

Dr. Paul Lefrere, The Open University in UK (UKOU)

Dr. Adrian Rawlings/ Drs. Peter van Rosmalen, Open University of the Netherlands (OUNL)

Contributors

Simon Price (University of Bristol)

Prof. Dr. Gunnar Teege (Universität der Bundeswehr München)

Christian Süß (University of Passau, Germany)

Dr. Eddy Forte (Swiss Federal Institute of Technology)

Drs. Jocelyn Manderveld (Open University of the Netherlands)

Contents

Introduction

This survey and analysis of EMLs is a project of the CEN/ISSS Workshop on Learning Technologies (WS-LT). The purpose is to arrive at a CEN/ISSS Workshop Agreement (CWA) for EML that could eventually be passed on for consideration as part of the regular standardization work.

The kick-off meeting took place in Torino in October 2001, at which there were 49 participants. A number of EMLs and other systems were presented, and a draft framework for the analysis was drawn up, and this was used as the basis for a questionnaire that was completed by participants at the meeting.

Responses to this questionnaire were brought together in a draft report (version b), which was presented at a meeting in Berlin on 30th November 2001. The purpose of the meeting to get a consensus as to what should be considered for inclusion in the CWA. The findings from the questionnaire and this meeting were incorporated in version h of the report. An overview and the conclusion of version h was presented and discussed at a meeting in Brussels on 2nd July 2002. This version, the final one, incorporates the comments on this version and some minor editorial changes.

Structure of this report

This report is in two parts:

1. Inventory of EMLs
    - Introduction
    - Inventory of EMLs
    - Information models
    - Use cases
    - Comparison of the different EMLs

2. Annex
    - Participants in the EML Survey
    - Questionnaire text
    - for each EML studied:
        - DTD (with diagram)
        - Sample of course material showing tags and screen dump
        - Supplementary information

Inventory of EMLs

Following extensive desk research, six potential EMLs were found for inclusion in the survey. Inclusion or exclusion was decided on the basis of whether the available published data promised to comply with the definition of an EML.

The working definition of an EML that is used throughout this project is:

An EML is a semantic information model and binding, describing the content and process within a 'unit of learning' from a pedagogical perspective in order to support reuse and interoperability

To state it differently: an EML is a semantic notation for units of learning to be used in e-learning to support the reuse of pedagogical entities like learning designs, learning objectives, learning activities, etc.

The concept of 'unit of learning' (also called 'unit of study' or 'unit of instruction') is central to this case. A unit of learning describes the learning design, the resources and the services needed in order to achieve one or more interrelated learning objectives. This means that a unit of study cannot be broken down to its component parts without loosing its semantic and pragmatic meaning and its effectiveness towards the attainment of the learning objectives. Common names for units of learning are: course, module, lesson or curriculum.

The search criterion for candidate EMLs was whether the information model and binding of a candidate is aimed at the semantic modelling of 'units of learning', or parts of it. It is the depth and scope of the underlying semantic description of the teaching-learning environment that is of interest here. The eventual aim is to analyse the different information models, compare them and bring them into a joint information model. The bindings are just used as another view on the information model, but will not be worked out or compared as such. A future CWA will concentrate on a joint information model only. Out of scope are runtime systems and real world implementations.

Besides the six candidates there are numerous of proprietary formats that are not open to the public and could not be included here.

We also didn't include IMS Content Packaging (and its derivates like ADL SCORM). IMS Content Packaging is able to aggregate a collection of resources towards e.g. a unit of learning, but it doesn't include any pedagogical semantics. These specifications can possibly be used as technical frameworks where the semantics can be included into.

To summarize. The underlying *raison d'être* of this project is to:

1. Identify possible EMLs or parts of an EML, given the EML definition.
2. Investigate the underlying information models and compare and position them.
3. Create a preliminary joint information model (in terms of a UML class diagram).

Six EML systems have been studied in this survey. These are:

CDF	Swiss Federal Institute of Technology (EPFL)
EML	Open University of the Netherlands (OUNL)
LMML	University of Passau, Germany (UP)
PALO	UNED University, Spain
Targeteam	Universität der Bundeswehr, München (UB)
TML / Netquest	ILRT, University of Bristol, UK (ILRT)

Note: EML is used in this document as a *generic concept* and as the *Educational Modelling Language of the OUNL.*. To distinguish between the two, the educational modelling language of the OUNL will be referred to as OUNL-EML.

Overview of EMLs

CDF

A completed CDF (an XML text file) can easily be used by an LMS (Learning Management System) to generate the actual online course, if the LMS is provided with operational access to the electronic pedagogical contents needed for the course - and referenced in the CDF. This is the case, for instance, of the Ariadne combination of tools: Curriculum Editor / LMS / KPS

Other features that were sought are: simplicity, pedagogical neutrality and generality. Without these properties, a model pedagogical process (the other important facet of the A-CDF) has an almost nil probability of being adopted by anybody else but its authors.

ARIADNE is suited to all pedagogical domains, but especially higher education.

OUNL-EML

OUNL-EML has been developed by the Open University of the Netherlands for use in e-learning. The version 1.0 information model and XML binding has been released in December 2000. Version 1.1 is in beta. OUNL-EML has been selected as the base for the IMS Learning Design specification, where it is integrated with IMS Content Packaging and IMS Simple Sequencing. It has been used and deployed in several runtime systems and a large number of courses has been modelled and delivered with it.

Current learning technology specifications allow only for some simple ordering and sequencing of resources used in e-learning (e.g. SCORM, IMS Content Packaging, IMS Simple Sequencing). OUNL-EML adds to this the ability to integrate learning designs (‘instructional designs’) to enable more advanced e-learning applications, e.g. to model competency based education, portfolio's, collaborative learning.

OUNL-EML is a semantic specification, based on a pedagogical meta-model, which describes the structure and processes in a ‘units of learning’. It aggregates learning objects with learning objectives, prerequisites, learning activities, teaching activities and learning

services in a workflow (or better 'learning flow'), which itself is modelled according to a certain learning design. The learning design is a concrete instance of a pedagogical model, which at its turn is an instance of the pedagogical meta-model. The meta-model doesn't force users to use a certain pedagogical model, but allows to create and describe their own models in an expressive way. The meta-model is derived from an analysis of current existing models, based on (social) constructivist's approaches, behaviourist's approaches or cognitive approaches.

OUNL-EML provides a semantic information model and several bindings (in SGML and XML).

LMML

The Learning Material Markup Language (LMML) is based on a meta modelling architecture for knowledge management (see section Information Model). It is an implementation of the XML binding of the teachware-specific meta-model of that architecture. It is a flexibly adaptable and extensible family of XML markup languages for learning and teaching material (teachware). LMML provides sub-languages for various educational fields. LMML is used in university education, further education as well as company training.

There is a short introduction in the LMML tutorial at the project homepage www.lmml.de. Here are some key features:

Model-based XML framework for the modelling and markup of teachware:

- Adaptable to different domains of application, e.g. computer science, operations research, financial planning etc.
- Extensible by elements of other XML languages e.g. MathML, SMIL etc.

Fragmentation of a domain of application into modules of arbitrary granularity which can be:

- Flexibly reused
- Adapted to different learning situations and to individual learners
- Published to different target media (Cross-Media-Publishing)

Simple yet powerful development and usage of multimedia teachware:

- Incremental, document-oriented, top-down or bottom-up development
- Easy migration of legacy documents in different formats
- Support by standard XML authoring tools

LMML as a framework is oriented to describe primarily the content of units of learning in arbitrary domains of application. LMML provides basic pedagogical markup but is non-specific to a certain pedagogical model (see later in this analysis). For example, concerning tests, it incorporates a simple exercise model which can be extended to more complex questions & answer models. Prerequisites can be expressed using corresponding associations between modules, i.e. using <referencesLink> with corresponding type. LMML also could be used to model the contents used in a more pedagogical-centric EML. Furthermore, tasks, objectives, prerequisites could be incorporated using appropriate meta-data. Actually, using subtyping, a pedagogical model can be integrated into LMML (as described in Franz Weitl, Christian Süß, Rudolf Kammerl, Burkhard Freitag: *Presenting Complex e-Learning Content on the Web: A Didactical Reference Model*. In: *Proceedings of e-learn 2002 world*

conference on E-Learning in Corporate, Government, Healthcare, & Higher Education, 2002).

PALO

PALO Language is an Educational Modelling Language developed at the Department of Languages and Computer Systems of UNED University, in exploitation since 1998. PALO defines a cognitive approach of an EML that describe courses structured in modules. Each module includes a declaration of the structure, the activities students and tutors undertake, and the scheduling of activities and content. Module and task sequencing is scheduled by mean of attributes of the language, providing deadlines and dependences between modules and tasks based on different types of prerequisites.

PALO defines learning scenarios by mean of *instructional templates*. An instructional template defines a *type* of learning scenario with certain pedagogical properties. Pedagogical domain is defined by mean of elements of the language that provides different functionality.

Instructional templates, however, do not define different "languages" by themselves, but a subset of the element of the language that provides a certain type of pedagogical functionality. A group of consistent elements of PALO that provide an instructional or pedagogical purpose constitute a template.

PALO is intended to be pedagogically neutral. Elements of the language provide, however, the capability of configuring learning scenarios based on both behaviourist and constructivist approaches

Targeteam

Targeteam is a system for supporting the preparation, use, and reuse of teaching materials. It is centred around the XML based language TeachML. It supports representing, structuring, and managing content used in all kinds of learning situations. It also supports the *production* of these materials, their flexible *reuse*, and the generation of different delivery formats which are used in the learning process. Targeteam does not actively support the learning process itself. It is applicable to all kinds of pedagogical domains, especially higher education.

TML/Netquest

NetQuest (http://www.ilrt.bris.ac.uk/netquest/about/general) is a project building upon the TML language to explore the creation of searchable question banks for online delivery of tutorials and assessment. TML is an interchange framework designed to separate the semantic content of a question from its screen layout or formatting. The language is designed to support several different types of question within the same content model.

Tutorial Markup Language (TML) an interchange format designed to separate the semantic content of a question from its screen layout or formatting. The language is designed to support several different types of question within the same content model. TML 4.0 is essentially a super-set of HTML, with new elements added to describe question information. TML version 4.0 has been specified using SGML, an ISO standard language for formally describing document types.

Work on TML 5.0 is supported in part by the JISC JTAP (http://www.jtap.ac.uk/) programme. Documentation is currently very sketchy. We are participating in standards-track work within the World Wide Web Consortium (W3C) and the Instructional Management Systems project (IMS). In September 1998 the NetQuest project at ILRT co-hosted with the UK IMS Office a workshop on interchange formats for question/quiz content. We hope TML 5.0 will contribute to the ongoing work of this community.

Information models

CDF

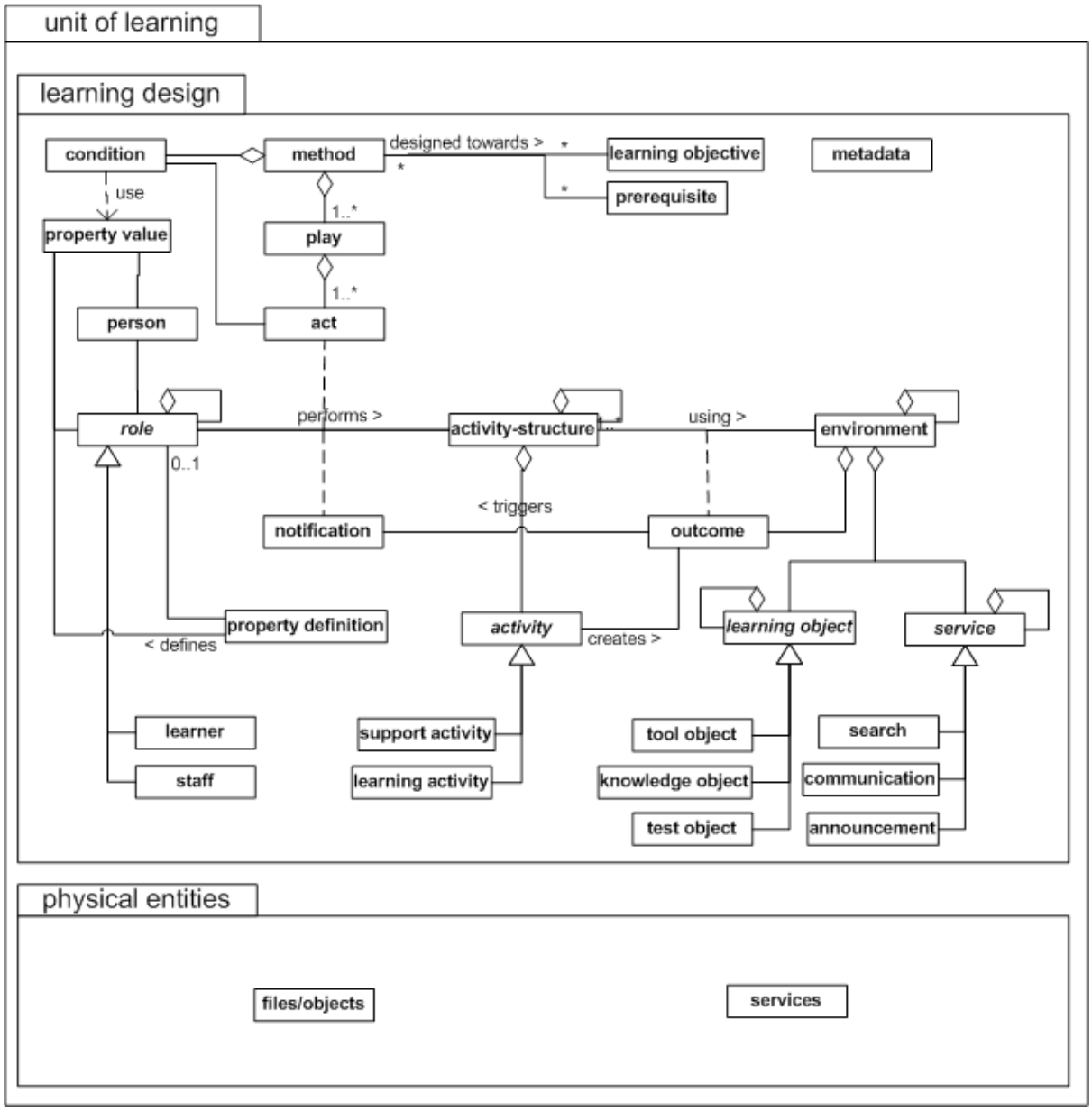
The Ariadne Course Description Format (A-CDF) is used to make courses. A completed CDF takes the form of an XML text file and is used in conjunction with an LMS (Learning Management System) to generate online courses.

OUNL-EML

The major requirements for the development of the OUNL-EML information model were:

1. OUNL-EML must describe units of learning in a formal way, so that automatic processing is possible (*formalisation*).
2. OUNL-EML must be able to describe units of learning that are based on different theories and models of learning and instruction (*pedagogical flexibility*).
3. OUNL-EML must explicitly express the semantic meaning of the different learning objects within the context of a unit of learning. It must provide for a semantic structure of the content or functionality of the typed learning objects within a unit of learning, alongside a reference possibility (*explicitly typed learning objects*).
4. OUNL-EML must be able to fully describe a unit of learning, 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. OUNL-EML must describe the units of learning so that repeated execution is possible (*reproducibility*).
6. OUNL-EML must be able to describe personalization aspects within units of learning, so that the content and activities within units of learning 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. OUNL-EML describes content resources in a medium neutral way, so that it can be used in different publication formats, like the web, paper, e-books, mobile, etc. and also in different settings like distance teaching, online learning, blended learning, hybrid learning, … (*medium and setting neutrality*).
8. OUNL-EML files must be interoperable between different learning (content) management systems (*interoperability and sustainability).*
9. The notational system must fit in with available standards and specifications (*compatibility).*
10. OUNL-EML must make it possible to identify, isolate, de-contextualize and exchange useful learning objects, and to re-use these in other contexts (*reusability*).
11. OUNL-EML must make it possible to produce, mutate, preserve, distribute and archive units of learning and all of its containing learning objects (*life cycle).*

OUNL-EML has currently two versions (1.0 and 1.1 beta). There is a XSLT converter available automatically converting 1.0 to 1.1. The information model of version 1.1 is depicted in the diagram below.



The Information model was developed by analysing all kinds of pedagogical models. In its most general form, it can describe a *role* (student or staff) who perform(s) a series of *activities* in an *environment* that consists of learning objects and learning services.

The information model is implemented in the OUNL-EML 1.0 and 1.1 beta binding (DTD).

LMML

LMML is based on a meta modelling architecture (see Christian Süss, A Meta-Modelling Adaptive Knowledge Management: Approach and its Binding to XML (2000)). This architecture is the conceptual foundation for the adaptation of LMML to different domains of application, the flexible reuse of teachware and the efficient storage of LMML modules in relational databases.

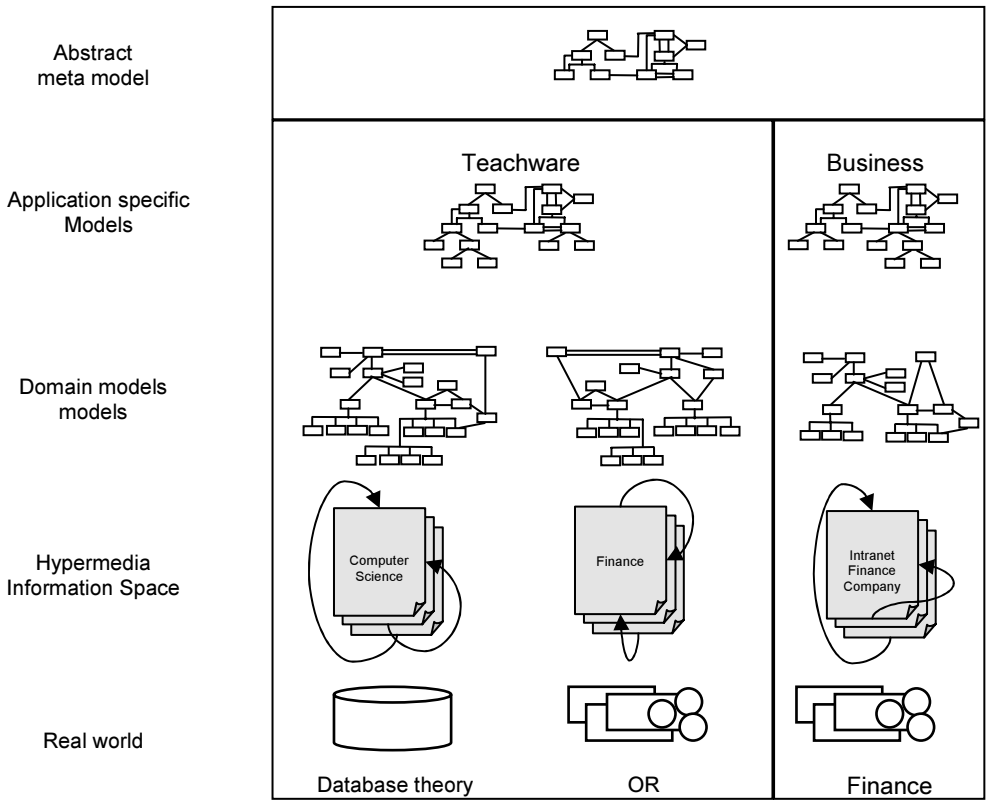
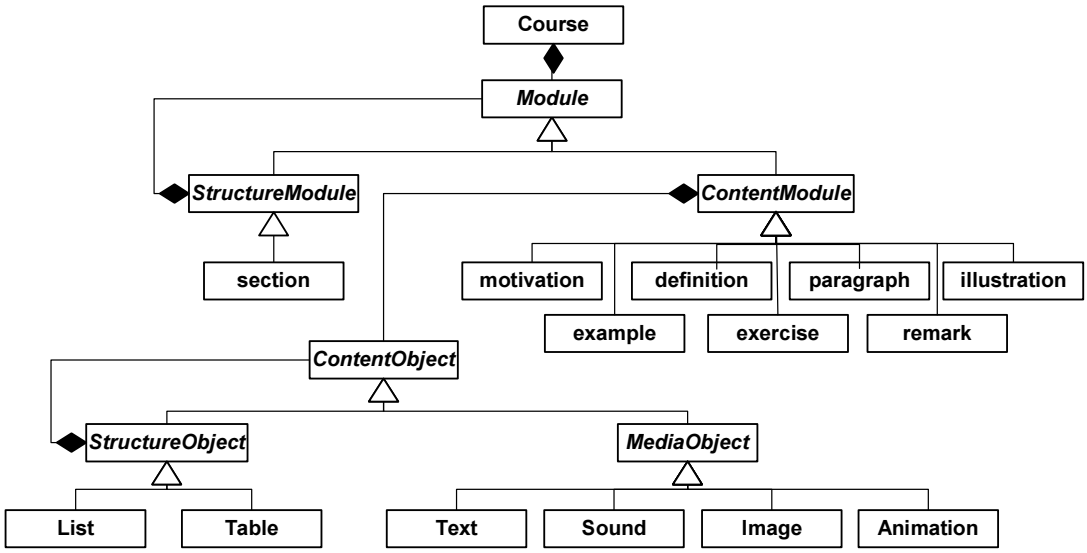


Figure 1: Meta-Modelling Architecture

A simplified version of the teachware model can be found in this UML diagram:



The LMML framework is an XML-binding of that model. According to this, learning material is organized modularly. It consists of various modules which in return may contain further modules. The structure of LMML-documents, and therefore the form of the resulting learning material, is defined by DTD modules as mentioned above. The basic units of information in this module-hierarchy considered relevant by the provider of educational content are called ContentModules, e.g. definition or motivation (see UML diagram above). They contain Media Objects, for instance pictures, animations or text that can be structured as lists or tables. Because of this, learning material reveals a hierarchical structure. At the same time this also indicates a possible way to organize the development of learning material: The authors of MediaObjects create multimedia content such as pictures or animations. In return, the producers of ContentModules apply them e.g. in illustrations, motivations etc. Finally, the designers of the courses arrange them to develop curriculums.

All relevant publications are available from the project homepage at www.lmml.de

PALO

PALO learning environments are supported by an information model that manages learning components of the system as well as the activities carried out within the system itself. Information models are defined in such a way that the same PALO file can be linked to several different groups of users. Some of the components of this information model are related to the components of the learning system as defined in the PALO file.

The information model is defined by the language is based in a decomposition of learning components based in five levels of abstraction.

Content level: Describes learning content included in the course. This information can be inserted “as is” in a variety of formats (HTML, LaTeX, SMILES and plain text). Additionally PALO allows to insert instructional queries to a knowledge domain of the content based on a learning ontology.. These *knowledge* has to be created previously and is externally stored in a DB.

Further explanations of these mechanisms can be found at PALO website.

Task level: Describes the activities included within the learning environment. Activities configure one or more tasks that are intended to be developed both individual or collaboratively. Tasks are simple activities that have a description and an *outcome*. Outcome can be produced by a community using one or more tools. **Structure level:** This elements define the structure of the learning environment. Depending on the final delivery model structure elements can be presented as the navigational model (HTML, CD-ROM binding) or an index (any hard copy, like PDF). Examples of structure elements is to divide a learning environment in *modules*, *theme*, *parts*, *subparts* and *sections*.

Management Layer	LMS interoperability info
Scheduling Layer	Scheduling, prerequisites, dependences
Activity Layer	Navigational model, Table of contents
Structure Layer	Activity, community, Roles, resources, tools
Content Layer	RIO's, RLO's Knowledge Domains Assets

PALO Description Layers

Schedule level: Defines the schedule and dependencies between elements of the learning scenario for a specific use. Scheduling includes:

- Dependencies between *modules*
- Deadlines of *modules*

In PALO, scheduling properties are assigned to modules. Themes have no scheduling properties in order to provide different functionality.

It is also possible to define prerequisites between modules, thus given a certain module, some attributes can be set to check for a positive assessment in the previous module, a simple validation, or a deadline.

Management level: Describes management aspects related to the production process of the learning environment from the PALO description in . Some of the description is related to the location of the final environment (directory) and location of the learning objects repository, among others.

Targeteam

Targeteam is a system for supporting the preparation, use, and reuse of teaching materials. It is centred around the XML based language TeachML .Targeteam supports representing, structuring, and managing content used in all kinds of learning situations. Targeteam supports the *production* of these materials, their flexible *reuse*, and the generation of different delivery formats which are used in the learning process. Targeteam does not actively support the learning process itself.

The information model consists of a homogeneous hierarchy of neutral paragraph-sized *issues* containing text or media objects such as images, animations or applets. Additional metadata can be attached on all levels to identify the educational purpose of the issue, such as illustration, exercise, explanation, motivation etc. It is planned to massively extend metadata in future versions, including standards such as LOM.

TML & NetQuest

NetQuest is a project building upon the TML language to explore the creation of searchable *question banks* for online delivery of tutorials and assessment. TML is an interchange framework designed to separate the semantic content of a question from its screen layout or formatting. The language is designed to support several different types of question within the same content model.

Tutorial Markup Language (TML) an interchange format designed to separate the semantic content of a question from its screen layout or formatting. The language is designed to support several different types of question within the same content model. TML 4.0 is essentially a super-set of HTML, with new elements added to describe question information. TML version 4.0 has been specified using SGML, an ISO standard language for formally describing document types.

Use Cases

CDF

Not available

OUNL-EML

The following are examples of using OUNL-EML from an (educational) designer's point of view.

Use case 1: To Define Staff roles

To define the different forms of guidance that have to be offered by the staff. For course X we would like to distinguish the roles of the administrator, the tutor and the evaluator. Persons can fulfil more than one role but never at the same time, since activities for a certain role are carried out according to a specific sequence. Every role therefore most be executed within a specific activity structure.

In course X the administrator has the sole responsibility to authorize and manage new (versions of) sources and adapt them, besides this he is only allowed to look at the student activities. The tutor is the sole role that is responsible for the assessment of student activities and the final grading of the reports that have to be submitted. Finally, the evaluator can only look at the student activities (e.g. for research purposes), but cannot influence the learning environment directly.

Use case 2: To Monitor Study progress students

To monitor the progress of the individual students or student groups the tutor must have access to an actualised overview at all times. In this overview he must find: the assessments on the assignments, the gradings of the reports, and the amount of time the students have spent on individual assignments. The amount of time spent on an assignment might influence the assessment on the assignment, the assessments on the assignments might influence the grading of the reports. Some reports have to graded as sufficient first, before students gain access to subsequent assignments. The conditions are described in the (different types of) activity structures for (different types of) students.

Apart from the overviews for groups of students for the tutor, every individual student has to be offered an overview of his personal study progress. In this personal dossier, the individual assessments and gradings (with comments from the tutor) will also be kept.

Use case 3: To Define a standard task form

In order to assess tasks consequently and honestly for every student, a standardized argumentation form with criteria will be used. All the constituting criteria for a task will be dealt with in little sub-reports on the different assignments a student can submit before submitting the final report. The form presents an overview of all these sub-reports that were written by an individual student. The benefit of using such a form is that it serves as a checklist for the tutor (no sub-reports / criteria are left out), and makes it possible to bring all information back together again, with the preliminary or final grading. When the student receives an insufficient grading, he can decide to rewrite the sub-reports or the final report. A

student who wrote good sub-reports, but fails on the final report, might be given a sufficient overall grading on the task as a whole after all. Of course, a student might decide to skip these sub-reports and start with the final report immediately.

Use case 4: To Manage source materials

Files to be used for learning activities will have to be updated. Managing the subsequent versions of new and old sources is expected to be an issue of concern. Therefore the learning environment has to provide actualised overviews of the sources for the different tasks, that are to be managed by the administrator. In these overviews also the (latest version of) criteria for each task are kept.

Sources might be text files, but also multimedia animations, or references to outside the course materials, like on the Study-net website. Therefore, for every source the address must be kept.

Use case 5: To design In line Question-Answer interactions

For every learning activity the questions and answers for the student self have to be defined. The answers to the questions should only be shown on student's request. The questions have to consist of text and (sometimes moving) images. Sometimes reference is made to other multimedia sources that have to be shown.

Use case 6: To Define a standard grading form

The grading of the task is executed in a standardized way. In the grading form students will find their report (and constituting sub-reports, see standard task form) together with the grade and tutor's comments. Also available for some tasks that won't be grade personally, are worked out examples.

Use case 7: To Constitute individual learning environments

In order to execute an activity various tools, sources and materials are accessed through the environment. Several of these environments have to be defined and developed. Depending on the activity, some environments may be accessed while others remain closed. So, environments differ from one activity to another.

Use case 8: To Define individual activity structures

Tasks have to be executed in a predefined order. The order of the tasks is fixed, but the order of the constituting assignments within each task is facultative. Each assignment (=activity) has its own place in the activity structure (or activity tree). The accessibility of the various environments and sources depends on the task or even assignment the student is executing at that moment, on his place on the learning path also. In order to achieve this in advance, a rather complex number of conditions has to be defined for various student- and staff roles, and every activity has to be linked to these environments and resources (including all kinds of references, links and buttons that are either shown or hidden). The accessibility also has to depend on the role of the student or staff member; so there must be different activity structures for different roles.

Use case 9: To Define objectives and prerequisites

For every learning activity there has to be at least one objective or prerequisite defined for the student. The type of objective should also be provided.

LMML

Reuse and adaptation of eLearning content:

- Simple yet powerful development of eLearning content.
- Easy migration of legacy documents in different formats.
- Flexible reuse of eLearning content at arbitrary levels of granularity.
- Adaptation of eLearning content to different learning situations and to individual learners.
- Publication of eLearning content to different target media (Cross-Media-Publishing).

Reuse and adaptation of educational modelling:

- Adaptation of LMML to different domains of application, e.g. computer science, operations research, financial planning etc.
- Integration of elements of other XML languages e.g. MathML, SMIL etc.
- Use of standards like LOM, IMS etc.

PALO

PALO-based courses are on exploitation since 1998.

Current instances are:

- B.D. on Computer Science, A Program verification Course with 3 environments for a population of ~1000 students
- Open Courses: 4 open courses with ~150 students per course
- Industrial Engineering, Chemical virtual laboratory with 6 different environments

In all cases PALO descriptions (An SGML file) are turned into a learning scenario including both teacher and student environments.

Cases of use include:

a) Users Management

Student groups and teachers can be organized to carry out different courses written in PALO. When a PALO course is compiled, both the student and teacher environments are created and set up with their corresponding login accounts.

b) Individual Tasks

Activities can be defined individually. Configuration of an activity includes

- Type of outcome: Quiz, formatted text and uploaded file

- Assessment: Assessed tasks appear in the corresponding teacher environment and produce an interactive dialogue between student and teacher
- Simulations: Software assets based on Java applets can be added to a task to provide content specific information

c) Module dependences

Module sequencing can be configured by mean of a set of attributes. Types of sequencing considered by PALO are:

- Deadlines: Modules cannot be coursed after a given date
- Assessment prerequisites: Modules cannot be coursed until a list of other modules have been passed

Targeteam

Not available

TML / Netquest

Not available

Comparison of the different EMLs

To complete the survey in this chapter the EMLs are compared. The comparison fulfils two purposes. It shows the relation between the EMLs e.g. how much do they have in common or extend each other. Secondly, it reveals the current state of each of the EMLs in relation to the definition established. Before starting to read this chapter it is important to note that the comparison has been prepared upon the material available in this document. In other words updates of an EML or new and additional background material may and likely will impose changes to this comparison.

Basis for the comparison of the EMLs is a) the definition and b) a preliminary information model as was discussed and agreed upon in the meeting in Berlin and c) a comparison based on the questionnaire.

The questionnaire consisted of three parts:

> **Analysis framework**. The analysis framework provided the survey with in depth background information of the concerning EML. Including a.o. with name, scope, information model and schema/DTD and documentation.
>
> **Quantitative factors**. Quantitative factors include how longs the EML exists and upon its operational use and constraints.
>
> **Qualitative factors**. They include the tools available and the easy of use and re-usability of the material produced.

In the next sections first the results of the comparison are shown. The comparisons are deliberately brief and only give a quick overview in a matrix form. The first comparison displays in which way and to what extend each EML fulfils the definition. The second comparison shows how the EMLs fit into the preliminary information model and how their elements relate. The third comparison summarizes the results of the quantitative and qualitative factors of the questionnaire. Following the comparisons, in separate sections the overall outcome is discussed and a number of future actions proposed.

I. Comparison matrix based on the EML definition

The definition used for an EML is the following:

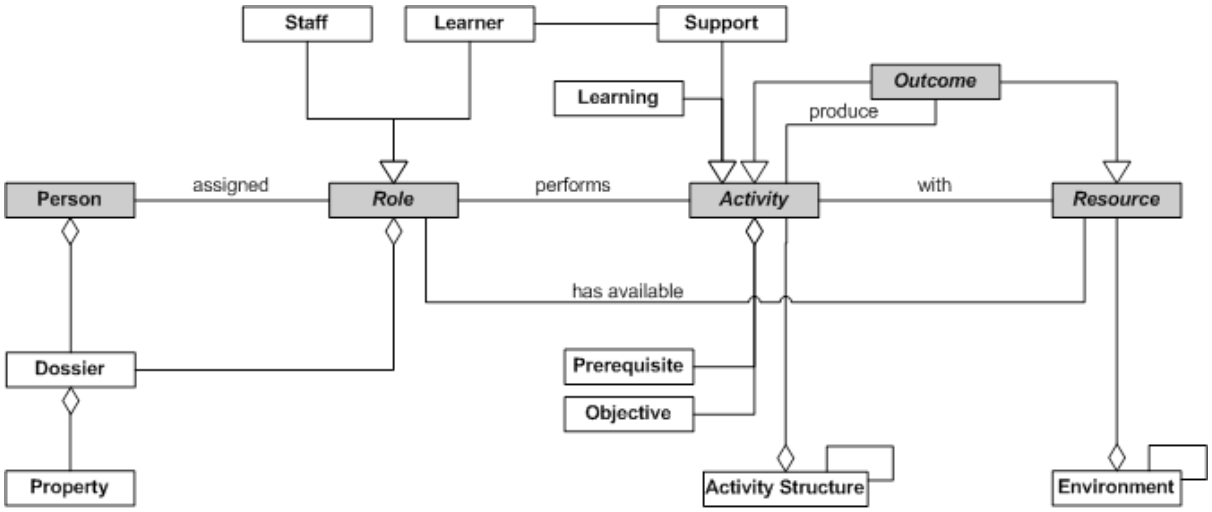
An EML is a semantic information model and binding, describing the content and process within a 'unit of learning' from a pedagogical perspective in order to support reuse and interoperability.

	CDF	**OUNL-EML**	**LMML**	**PALO**	**Targeteam**	**TML**
semantic information model available	Y	Y	Y	Y	N	Y
models a 'unit of learning'	Y (called 'course')	Y (called 'unit of study')	Y (called 'course')	Y (called 'module')	Y (called 'issue')	Not applicable
binding available	unknown	Y	Y	Y	Y	Y
language(s) of binding	XML	XML, SGML	XML	XML, SGML	XML	XML, SGML
one generic model/binding?	Y	Y	Y/N One generic framework binding which can be adopted to different domains	N For each instructional template its own schema defining a subset of PALO elements for a certain type of pedagogical functionality.	Y	Y
Includes complete content description	Y	Y	Y	Y	Y	Y
Includes complete process description	N	Y	N	Y (So far limited to individual tasks)	N	N
Pedagogical perspective *a) ignorant of any model;* *b)restricted set of models* *c) allows expression of own models*	Ignorant Note: It is not possible to actually express a pedagogical model.	Expressive Pedagogical meta language allowing to express all types of pedagogical models.	Ignorant/restricted Note: It is possible to actually express specific pedagogical models using subtyping (see [WSKF02] at www.lmml.de).	Restricted/ Expressive Constructivists and objectivists learning scenarios written in PALO are documented in the PALO home site.	Ignorant Note: It is not possible to actually express a pedagogical model.	Restricted to tutorial models and assessment
Remarks	1. focus on content and content aggregation. Not on expressing a learning flow based on a pedagogical model (expectations towards roles, activities and activity structures) 2. values are given at initial creation time. There is no instantiation process	1. Basis for the IMS Learning Design specification	1. c.f. CDF 1. 2 Framework based on a meta modelling architecture for knowledge management. Can easily be extended to suite different domains of application.	PALO is open in the sense that it can easily be extended each time there is a need for a certain pedagogical functionality. However, that the scope of the current version is restricted to a small selection	1. c.f. CDF 1. 2. Based on TeachML (XML based)	1. Based on Tutorial Mark-up Language a superset of HTML using SGML.
Re-usability	Y	Y	Y	Y	Y	Y

Use/integration of other Standards	LOM	LOM, IMS Content Packaging, IMS Simple Sequencing, WfMC, XHTML	Framework extension mechanism allows for use of LOM metadata etc. Media object mechanism allows for integration of content markup such as XHTML, LaTeX, MathML, SMIL etc.	Dublin Core HTML LaTeX	-	HTML

Table 1. Comparison matrix based on the EML definition

II. Comparison on preliminary information model



CEN/ISSS WSLT Basic Model of an Educational Modelling Language

	CDF	**OUNL-EML**	**LMML**	**PALO**	**Targeteam**	**TML**
Main scope of the entities in the information model	Limited to content and the required resources (staff, physical and software)	Content and process	Limited to content and domain specific.	Content and process. Five layers of abstraction: - Content - Task - Structure - Sequence and Schedule - Management	Limited to content.	Limited to questions: separates content from formatting.
main semantic entities in the information model: overview	Course metadata scheme: - general - target learner - sessions type - sessions - communication resources - teachers - locations	UoL : - learning objectives - prerequisites - roles learner, staff - activities learning/support - environment knowledge obj. test object tool object index/search communication announcement - Learning method process description - Properties, Conditions & Notifications	Course: - module - structure model - content module - structure object - media object	Course - Module: - activities - lo - embedded learning content	Module: - Hierarchy of paragraph sized issues	- Question types - Choices - Hints - Responses - Score

Table 2. Comparison matrix based on the preliminary information model

	CDF	**OUNL-EML**	**LMML**	**PALO**	**Targeteam**	**TML**
Unit of Learning	Course	Unit of study	Course	Course, Module	Module, Issue	*The scope of TML is so much different that it is left out of this mapping*
Role	Fixed: Learner; teacher	Any: at least one of type learner	Fixed: Learner	Fixed: Learner; teacher	Fixed: Learner	
Activity	SessionList	Activity	StructureObject	Tasks	-	
Resource	Location; Communication resources	Environment: knowledge obj. test object tool object index/search communication announcement	Media object	Knowledge domains of semantically linked material	-	

Table 3. A first attempt to 'map' key elements of the CEN/ISSS WSLT Basic Model between the EMLs.

III. Comparison on the basis of the quantitative and qualitative part of the questionnaire

Quantitative factors

	CDF	OUNL-EML	LMML	PALO	Targeteam	TML
How long has the EML been in development	6 years	3 years	3 years	4 years	3 year	8 year
How long has the EML been used operationally?	5 years	Official release 15-12-2000	Since April 2000	4 years	3 year	8 year
How many courses have been produced using the EML	Several Dozens	> 20	> 20	>10	> 10	7-8
Do academics use the system operationally	Y	Y	Y [also industrial partners]	Y	Y	Not any more
Do students use material generated by the system	Y	Y	Y	Y	Y	Y
Which pedagogical models does the EML system support *	ignorant	expressive	Ignorant/restricted Note: Different didactical models can be integrated.	Restricted / Expressive c.f. comparison matrix of EML-def.	Ignorant	Restricted: Self-study, formative and summative assessment
Which pedagogical models does the EML explicitly not support?	None	None	None	None	None	See above
How many person-years have been spent in developing the system?	>10	Unknown	4	3	1.5	< 1
Is the EML developed exclusively in-house, or has it been developed within a partnership?	Partnership	In-house, but with close interaction with companies & institutes	In-house some instances together with others	In-house	Started in-house. Now partnership	Open source
Is the EML a purely academic development, purely commercial, or a blend of the two	Initially academic. Now supported by a non-profit association	Public domain. (Subject to copyright restrictions)	Academic framework free for academic use; different academic	Academic	Academic	Academic

			and commercial extensions.			

* The answers are 'translated' to be in-line with the pedagogical perspective used in table 1, i.e. a) ignorant of any model; b) restricted set of models c) allows expression of own models.

Table 4. Comparison matrix Quantitative factors

Qualitative factors

	CDF	**OUNL-EML**	**LMML**	**PALO**	**Targeteam**	**TML**
What are the authoring tools like? Are they user-friendly or low-level	Any authoring tool can be used	Customizations for Framemaker + SGML, Xmetal, XMLspy. In principle any other XML editor Plan for dedicated new authoring tools.	Standard XML tools, e.g. XML Spy or XMetaL and some specialized tools In principle any other XML editor Plan for dedicated new authoring tools.	Standard SGML and XML editors XEmacs	Standard XML editors and some system specific tools	Text-based mark up
How difficult is it to update content	Easy	Easy	Easy	Easy	Easy	Easy
To what extent is the course material produced by the system re-useable	Completely	Completely	Completely	Completely	Completely	Completely
To what extent is the course material produced by the system media-independent	Course material is pure text	Media independent	Media independent	Media independent (HTML, CDROM)	Independent with exception of specific media formats	High degree

Table 5. Comparison matrix Qualitative factors

Conclusions

The outcome of this comparison makes clear that the scope of the six EMLs (c.f. table 1) differs very much. There appear to be two groups. The first group consisting of CDF, LMML, Targeteam and TML restrict themselves – be it very sophisticated- to modelling learning content and structure, in the case of TML even a specific type of content i.e. questions. They seem to be 'ignorant' in expressing pedagogical models. They can be used within any pedagogical model (with the exception of TML), but they cannot express executable knowledge about the model. (Note: by subtyping LMML can integrate specific pedagogical models). The second group consists of PALO and OUNL-EML. They live up to the definition the survey started from '*An EML is a semantic rich information model and binding, describing the content and process within 'units of learning' from a pedagogical perspective.'* From the two, OUNL-EML fully lives up to the definition, PALO e.g. is limited to the definition of individual tasks. Nevertheless at this stage it is already clear that the expressive power of the current version of OUNL-EML - by simply studying and looking at the number of defined elements and attributes - exceeds the expressive power of PALO.

Moreover looking at the table 1, a number of general remarks can be made. Practice shows there is a general agreement on the use of XML and/or SGML and on the importance of having a formal binding. Each of the approaches share a strong interest in supporting re-usability. However, looking into more detail (c.f. the preliminary information model and table 2 and 3) it is clear that the element/attributes that are in use in each of the EML are proprietary. Interoperability, at least at this stage, between the EMLs cannot be achieved. How much they overlap and subsequent a mapping can be made requires more study. In this context it is important to note that currently OUNL-EML (version 1.1) is the only one of the EMLs discussed which both is compatible with various international standards *and* follows the process and procedures to be accepted as a standard.

To last two tables, i.e. Table 4 and 5, show some general characteristics. All initiatives started from an academic background and with exception of CDF started by one institute. All EMLs did prove their value in real applications with serious amounts of students. In each of the EMLs a serious concern has been to assure re-usability and a high degree of media independence. Finally, its is important to notice -though not mentioned explicitly- that each of the EMLs relies on a professional process of designing and developing Units of Learning giving the dependence of standard XML/SGML tools in the authoring process.

Future actions

This survey contains an overview and an analysis of six EMLs. It has been carried out as a project of the CEN/ISSS Workshop on Learning Technologies (WS-LT). The purpose is to arrive at a CEN/ISSS Workshop Agreement (CWA) for EML that could eventually be passed

on for consideration as part of the regular standardization work. To arrive at this point the first step to be taken is to have this text discussed by a wider audience to get feedback.

Therefore the first action proposed is to discuss this text in the Prometeus community in the Paris conference September 29/30, 2002, get feedback and if required start-up additional actions to clarify whatever necessary.

Additionally, we have to discuss the next steps to be taken. Before taking the next steps it is important to look back what we have achieved. So far we identified an answer (or part of an answer) to the five mayor question posed in the Turin WS-LT:

What is the need for an EML standard? Where and how can we use such a standard?

> The current communities of practice of EML all have as a high priority re-usability. They have many similarities in both approach and objectives to achieve. One standard would be of benefit to all.

What are the differences and commonalities between the different EML initiatives?

> Going into detail the comparison shows that there are clear differences both in scope and the implementations.

What is the relationship with existing learning technology specifications?

> The awareness of standards is clear, but so far only OUNL-EML actively adapts and is used as a basis of a standardization processes.

Can we identify modules in this framework for splitting up work?

> At this stage it is not possible.

How do we take care that the EML is both pedagogically rich and yet does not restrict any pedagogical modes.

> In the survey PALO and OUNL-EML showed how this can be achieved.

Looking at these answers we propose -for discussion- the following two possible ways to contribute to standardization:

- The first one is to further elaborate on the CEN-ISSS WSLT information model for EML and based on this analysis prepare an in depth analysis of each EML, how they fit and how they together can contribute to this EML information model. However, giving the current variety in scope and implementation this will be difficult. Moreover, this approach will compete with existing standardization work in IMS LD.
- The second option is in line with the CWA '**Internationalization of the IEEE Learning Object Metadata"**. It is to comment upon a standard (in this case a standard-under-preparation, i.e. IMS LD as soon as it comes available). The experience and richness of the EMLs discussed in this document can be used to assure the validity of this EML in a multicultural and multilingual environment, from a European Perspective.

Annex

Participants in the survey and analysis

ARIADNE Dr. Eddy Forte
EPFL (Swiss Federal Institute of Technology)
CH-1015 Lausanne, Switzerland
+41.21.693 4755/6662
Eddy.Forte@epfl.ch
http://www.ariadne-eu.org

EML Prof. dr. Rob Koper
OTEC
OU NL
Valkenburgerweg 177
Postbus 2960
6401 DL Heerlen, The Netherlands
+31 45 576 2657
rob.koper@ou.nl
www.ou.nl

LMML Christian Süß
University of Passau
D-94030, Passau, Germany
+49 (0)851 509 3075
suess@fmi.uni-passau.de
http://daisy.fmi.uni-passau.de/db/literatur.php3?key=S00

PALO Dr. Miguel Rodríguez Artacho
Assistant Professor -- Profesor Titular de EscuelaUniversitaria
UNED University , Madrid
+ 34 91 398 7924
miguel@lsi.uned.es
http://sensei.lsi.uned.es/palo

Targeteam Prof. Dr. Gunnar Teege
Fakultät für Informatik
Universität der Bundeswehr München
D-85577 Neubiberg, Germany
+49 89 6004 3353
Gunnar.Teege@UniBw-Muenchen.de
htttp://www11.in.tum.de/forschung/projekte/targeteam/

TML Grainne Conole, Dan Brickley, Simon Price
Institute for Learning and Research Technology
University of Bristol
8-10 Berkeley Square
Bristol, BS8 1HH, United Kingdom
T +44 (0)7071 226 720
Daniel.Brickley@ilrt.bris.ac.uk
http://www.ilrt.bristol.ac.uk

Questionnaire text

Respondents provided their input through the questionnaire shown below.

Analysis framework

- Name of framework
- Scope, e.g., What is the pedagogical domain? Which pedagogical entities are included?
- Type of semantic information model
- Summary
- Instances of this framework
- Schema / DTD, samples, documentation
- Information model
- Tutorial
- Sample courses
- Experiences of users
- Interoperability
- Further publications
- Contact

Quantitative factors

- How long has the EML been in development?
- How long has the EML been used operationally?
- How many courses have been produced using the EML?
- How many students have used the EML?
- Do academics use the system operationally?
- Do students use material generated by the system?
- Which pedagogical models does the EML system support?
- Which pedagogical models does the EML explicitly not support?
- How many person-years have been spent in developing the system?
- Is the EML developed exclusively in-house, or has it been developed within a partnership?
- Is the EML a purely academic development, purely commercial, or a blend of the two?

Qualitative factors

- What are the authoring tools like? Are they user-friendly or low-level?
- How difficult is it to update content?
- To what extent is the course material produced by the system re-useable?
- To what extent is the course material produced by the system media-independent?

Supplementary information

In this section, information is presented for each EML in the following format:

- DTD
- Sample of course material
- Sample of actual EML code in use, showing tags , etc.
- Quantitative factors
- Qualitative factors

CDF

DTD

The ARIADNE Curriculum (or course) description format (CDF) was first developed, in a simpler version, during the ARIADNE European Projects (1996-2000). The present version incorporates, amongst other, many pieces of information stemming from the *Pedagogical Scenario* document, also developed by these EU Projects.

Implemented as an editor tool - or even as a simple paper template -, the CDF provides a powerful *design framework* for pedagogical engineers and ODL trainers, for many kind of technology-supported learning processes.

A completed CDF (an XML textfile) can easily be used by an LMS (Learning Management System) to generate the actual online course, if the LMS is provided with operational access to the electronic pedagogical contents needed for the course - and referenced in the CDF. This is the case, for instance, of the Ariadne combination of tools: Curriculum Editor / LMS / KPS.

ARIADNE intends to submit this specification to both the IEEE Learning Technologies Standardization Committee, as input for a possible international standard, and to the ADL Project's Technical Board (editors of the SCORM specification), for comments and possible re-use in the framework of future versions of the SCORM itself

This CDF specification is based on the same *structured metadata* approach that was successfully used for producing the original ARIADNE Educational Metadata Scheme in 1996, and the IEEE-LTSC's LOM specification (of which the current ARIADNE Educational Metadata Set V.3.1 is an application profile). In fact, the A-CDF can be viewed as a *Course Metadata Scheme.* Note that the tokens and numbering used in the present document are still under discussion.

Other features that were sought are: *simplicity, pedagogical neutrality* and *generality*. Without these properties, a *model pedagogical process* (the other important facet of the A-CDF) has an almost nil probability of being adopted by anybody else but its authors.

The complete methodology- and rationale - behind the A-CDF approach is described in a longer document, yet to be published. The comments included in this more compact document may, however, help the reader to judge for himself of the applicability, advantages and shortcomings of the proposed structure.

A CDF comprises seven categories of elements:

sequence of	*Comments*
1. General	General characteristics of the curriculum *at design and ordering time*
2. Learner	General Information on the *target learner population*
3. SessionTypeList	List of *session types* defined for and used in this course.
4. SessionList	List of *actual sessions* defined for this course.
5. CResourceList	List of *communication resources* needed for this course
6. TeacherList	List of *teaching staff members* involved in this course
7. LocationList	List of *generic or actual physical locations* needed for all kinds of learning activities included in this course

General

The *General* category comprises 31 data elements that must be sought and provided at the preliminary course *design and ordering stage:*

sequence of	*Comments*
course-title	Approved by the *Executive Person ordering the course*
course-descriptor-file-name	(May be needed when the CDF is used by an LMS)
cdf-id	(Needed when the CDF is used by an LMS or stored as an LO in a database. Auto-generated)
exec-resp-person	The *executive* person (in the framework of the *Organization*) ordering the course to take place and allocating human and other resources to this end.
exec-resp-institution	The *organization under whose responsibility the course is being constructed*
executive-summary	A (short) *textual description* provide by - or approved by- the Executive Person
pedag-resp-person	The person *responsible for actually setting-up the course*
pedag-resp-institution	This person's Organization (usually the same as 1.5)
course-level	
course-background	Whether this is a *new course* or if it is *based on an existing course*
teaching-language	

course-remark	
discipline-name	
sub-discipline	
leading-concept	
pedag-duration	The estimated study time (*in hours*) for completing the course
course-timespan	The estimated timespan (*in weeks*) over which the course is supposed to extend.
course-start-date	
course-end-date	
course-student-load	Whether this course is supposed to be a *full-time* or *part-time* activity for the learners
actor-learner-number	An estimate of the number of learners *simultaneously following the course*. If there are several consecutive course editions, only one must be considered.
actor-teacher-number	An estimate of the number of teachers needed
actor-assistant-number	An estimate of the number of teaching assistants needed
actor-other-number	An estimate of the number of other staff members needed (such as course admin, secretary etc.)
training-environment	Whether this course is intended for *academic, professional, continuous,…*, formation
course-access	Whether the course is *mandatory or optional*
fee	Whether there is a *course tuition or fee* to be paid by the learner.
learner-assessment-kind	Whether the learners are *assessed, initially, continuously, terminally or not at all*.
certification	Whether the course leads to a certificate or not.
certification-title	The certificate's title, if any

Learner

The *Learner* category comprises 16 elements of information relative to the target learner population, together making-up a socio-geographic profile of this population. This data must be sought or estimated at the preliminary course *design and ordering stage:*

sequence of

learner-country

learner-kind This attribute can take one of two values: CAPTIVE and FREE. *Captive learners* have no choice but to do what they are told to do (e.g. school pupils, employees being told to train in X..., army staff, etc.).*Free learners* follow a course by their own choice.

learner-pop-description

learner-pop-distribution Whether the expected learner population is *local* (in a radius of less than 10 km from the course centre), *regional* (in a radius of more than 10 and less than 100 km from the course centre) or *remote* ((in a radius of more than 100 km from the course centre)

learner-prerequisites

learner-remark

learning-locations Whether the learning is expected to take place *at home*, *on the workplace*, in a *resource centre*, in a *lab* or in *more than one location*

learning-timeframe Whether the learning is expected to happen during the learner's *work-time*, *free-time* or as *night-studies* or otherwise.

available-learntime-monday An estimate (in hours) of the time available for learning on that week-day for the target learner

available-learntime-tuesday

• • •

available- learntime-sunday

tot-avail-course-learntime An estimate (in hours) of the *total time available for learning* (add 2.10 to 2.16 and multiply by 1.17...) during the expected course timespan for the target learners.

SessionTypeList

The *SessionTypeList* category regroups all defined SessionTypes; a *SessionType* is characterized by 14 attributes :

sequence of **SessionType**	*Comments*
template-title	The name given to the session type (or session profile)
template-description	A textual description, if appropriate, of the type.
temporal-type	This attribute may take one of two values: FIXED and FUZZY (or FLOATING). *Fixed* means that actual sessions derived from this profile are fixed both in time and space. This is the case of a session presented in a classroom. *Fuzzy (*or *Floating*) means that actual sessions derived from this profile may be spread over more than one day and possibly accessible from more than one location. This is usually the case for home-based sessions.
pedagogic-duration	The typical learning time (in hours) needed for working through such a session
recurrent	May take one of two values: RECURRENT or NON-RECURRENT. A session derived from a *recurrent* session profile, will be scheduled regularly on the same week-day, every week, during the course's timespan. A session derived from a *non-recurrent* profile must be given a specific date.
fuzzy-rec-start-day	The starting weekday of recurrence (for *fuzzy* recurrent profiles only)
fuzzy-rec-end-day	The ending weekday of recurrence (for *fuzzy* recurrent profiles only)
fixed-rec-day	The weekday of recurrence (for *fixed* recurrent profiles only)
start-time	The start-time (for fixed session profiles only)
end-time	The end-time (for fixed sessions only)
active-period	The preferred percentage of the session's learning time to be devoted to *active documents* (simulations, quizzes, etc.)
expositive-period	The preferred percentage of the session's learning time to be devoted to *expositive documents* (text, hypertext, videos, etc.)
interactive-period	The preferred percentage of the session's learning time to be devoted to direct *human interaction* (with teachers, assistants, other learners...)
location-type	The kind of location needed for that session type

SessionList

The *SessionList* category regroups all actual Sessions; an actual *Session* is characterized by up to 18 attributes, most of which can be inherited from an appropriate *SessionType* (if defined).
Two important elements are only defined at the Session level: the session's *ConceptList and PeddocList* :

sequence of **Session**	**Comments**		
session-name			
session-description			
temporal-type			
pedagogic-duration			
start-date			
start-time			
end-date			
end-time			
ref-template			
location-type			
location-name			
active-period			
expositive-period			
interactive-period			
ConceptList	sequence of	*the list of topics taught in this session*	
	Concept	sequence of	
		concept-language	The language used to express the concept
		concept	One of the topics taught in that session
PeddocList	sequence of	*the list of pedagogical documents (LOs) used in this session*	

	Peddoc	sequence of	
		peddoc-LKP-ID	The unique identifier of the doc. in the KPS
		peddoc-title	
		peddoc-label	An optional textual label
		peddoc-duration	
		peddoc-URL	An optional URL for locating the doc.

CResourceList

The *CResourceList* category regroups all CResource's *(Communication Resources)* to be used in the course.
An actual *CResource* is characterized by 4 attributes.

sequence of

CResource

name

URL The URL of the resource

icon-URL The URL where an appropriate icon can be found.

description

TeacherList

The *TeachersList* category regroups all Teachers (teaching staff members) intervening in the course.
An actual *Teacher* is characterized by 5 attributes.

sequence of	
Teacher	A designated teaching staff member
teacher-name	
teacher-email	
teacher-phone	
teacher-remark	
qualifier	The actual role of that staff member (*teacher, assistant, technician, etc*.)

LocationList

The *LocationList* category regroups all Locations *(actual or virtual learning locations)* used for the course.
An actual *Location* is characterized by 4 attributes.

sequence of

Location

location-name	The name (or address) of the location
location-type	The kind of this location (see the list in the **General** category)
location-description	
location-capacity	The maximum number of learners for that location
location-computer-number	
os-type	
network-type	
specific-peripheral	A description of other peripherals or facilities (blackboard, projectors..)
location-remark	

Sample of course material

Not available

see http://www.ariadne-eu.org/5_RD/5.1_AFRefDocs/CDFV300.htm

A number of courses have been developed using the Ariadne system, and at a number of institutions across Europe. The Ariadne website shows the following courses: (http://www.ariadne-eu.org/2_AS/2.3_courses/main.html)

Concepts & Languages Orientés-Objets

Cours PASCAL

Cours PASCAL pour les ingénieurs physiciens 1ère année EPFL

Hydrologie Générale

Introduction à Java

SEPHYR tool training course

VBV AFA: Bes. Branchenk. Vermögens - u. Sachversicherungen

Maitriser sa force

Travaux pratiques d'automatique pour l'orientation IFE en GM

VBV AFA: Marketing und Verkauf

Ariadne servers are located at:

CAFIM/DSU/UFJ Grenoble, France

IRIT/CICT Toulouse, France

KUL/CS Leuven, Belgium

EPFL Lausanne, Switzerland

UG/DIBE Genova, Italy

UNIL Lausanne, Switzerland

Sample of EML code

Not available

Quantitative factors

How long has the EML been in development?

6 years

How long has the EML been used operationally?

5 years

How many courses have been produced using the EML?

several dozens

How many students have used the EML?

several thousands

Do academics use the system operationally?

yes

Do students use material generated by the system?

yes

Which pedagogical models does the EML system support?

neutral

Which pedagogical models does the EML explicitly not support?

none

How many person-years have been spent in developing the system?

more than 10

Is the EML developed exclusively in-house, or has it been developed within a partnership?

partnership

Is the EML a purely academic development, purely commercial, or a blend of the two?

Neither. Initially an academic venture, It is now supported by a non-profit Association.

Qualitative factors

What are the authoring tools like? Are they user-friendly or low-level?

Any authoring tool can be used.

How difficult is it to update content?

As easy as replacing the old content and referencing to in the CDF i.e. very easy.

To what extent is the course material produced by the system re-useable?

Course content is completely reusable and independent of the system

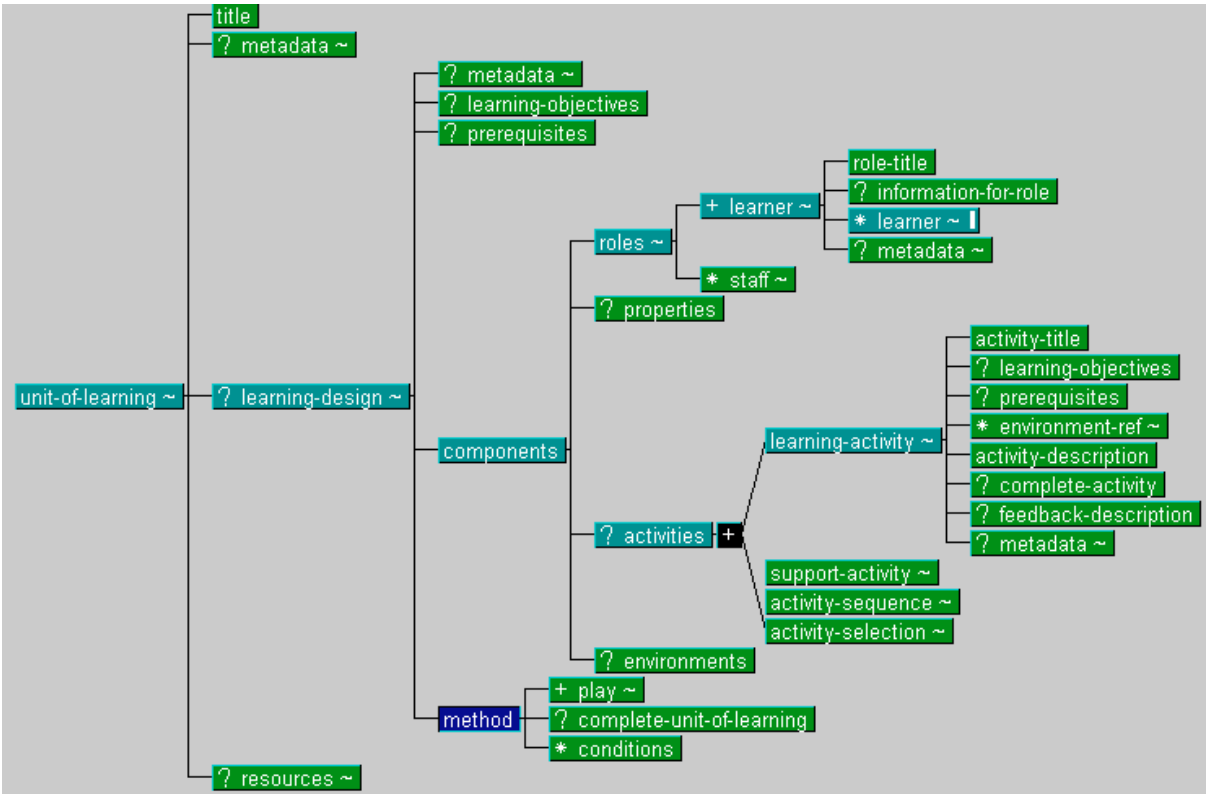
To what extent is the course material produced by the system media-independent?

The system doesn't produce the content. The course 'material' produced by the system is pure text, i.e. media-independent. The question remains for the actual content, incorporated in courses created with the system.

OUNL-EML

DTD

The OUNL-EML Binding (DTD), reference manual, and articles about OUNL-EML can be downloaded form: http://eml.ou.nl. Also specific sample courses are accessible at this website (Jazz course). The figure below gives the tree structure for the new version.



Sample of OUNL-EML document instance

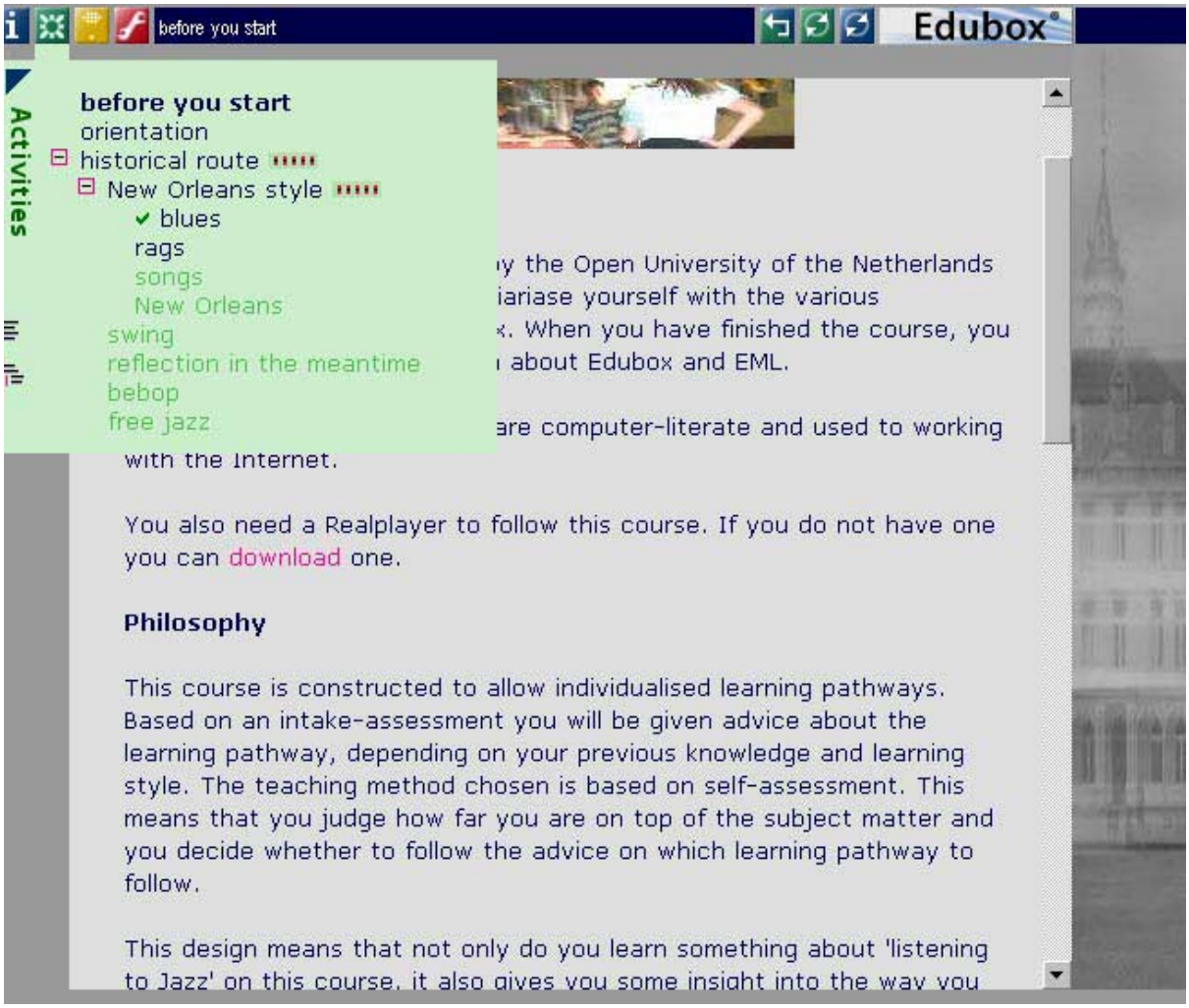
```
<unit-of-learning>
  <title>Introduction into EML</title>
  <metadata><<LOM schema included>></metadata>
  <learning-design>
    <components>
      <roles>
        <learner identifier="student"><title>student</title></learner>
        <staff identifier="tutor"><title>begeleider</title></staff>
      </roles>
      <properties/>
   <activities>
    <learning-activity identifier="id-F8" isvisible="true">
      <title>introduction</title>
      <activity-description><item identifierref="resource-act1" /></activity-description>
      <complete-activity><user-choice /></complete-activity>
    </learning-activity>
    <learning-activity identifier="id-87" isvisible="true">
      <title>about elements</title>
```

```
        <learning-objectives><item identifierref="resource-lo1"></item>
        </learning-objectives>
      <activity-description><item identifierref="resource-act2" /></activity-description>
      <complete-activity><user-choice /></complete-activity>
    </learning-activity>
    <activity-selection identifier="AS-intro" number-to-select="1">
      <title>Overview</title>
      <information><item identifierref="resource-info1" /></information>
      <environment-ref ref="id-69" />
      <learning-activity-ref ref="id-F8" />
      <learning-activity-ref ref="id-87" />
    </activity-selection>
    <environments>
      <environment identifier="id-69">
        <title>In general</title>
        <knowledge-object identifier="id-B66">
          <title>de knoppen in Edubox</title>
          <item identifierref="res-1.3.1.1.2.1" />
        </knowledge-object>
      </environment>
    </environments>
    </components>
   <method>
     <play>
       <act>
        <role-part><role-ref ref="student"/><activity-selection-ref ref="AS-intro" /></role-part>
       </act>
     </play>
     <conditions/>
          </method>
  </learning-design>
  <resources>
  <resource identifier="resource-lo1" type="emlcontent" href="resource-121111.xml">
  <file href="resource-121111.xml" /></resource>
  <<ETC>>
  </resources>
</unit-of-learning>
```

Content is any format, however more advanced content (e.g. using the properties of the unit-of-learning file to personalize content) uses XHTML with a specific set of global elements mixed into this using namespaces.

Sample of course material

A screen dump from the Jazz course is shown below. Note that this is a screen dump of a course presented using Edubox, a runtime system which interprets course material described using OUNL-EML. The screen dump shows the user having clicked on the Activities button and the corresponding drop-down menu.



There are many instances of OUNL-EML, some of these are:

- Four fully online 200 hour courses in Business and Public Administration for OUNL
- Courses in Psychology, Geography, Law, Geographical Information Systems, Economics, etc for OUNL
- 12 dual mode courses (the whole curriculum) for the Hotel Management School, institute for higher vocational education
- UNISA (Open University of South-Africa) has developed several courses in OUNL-EML
- Courses for Medienzentrum Innsbruck in Austria
- Digital University of the Netherlands (consortium of 9 universities) develops law courses in OUNL-EML.
- Training modules for training departments of different companies

There have been build several runtime systems during the development of OUNL-EML (Elon system, Edubox prototypes, etc.). Perot Systems is currently building a new scalable Java based runtime for OUNL-EML, including hosting services (of which some service levels are free).

Some examples of instance are available in the article: Modelling units of study from pedagogical perspective: The pedagogical metamodel behind OUNL-EML. This article can be downloaded form http://eml.ou.nl.

A demonstration course is available, *The Jazz Course.* It is a demonstrator of OUNL-EML and the reference system (Edubox), which plays OUNL-EML files. This course can be made publicly available. Accounts for these course are also available. Please get in touch with Jocelyn.manderveld@ou.nl

Quantitative factors

How long has the EML been in development?
3 years

How long has the EML been used operationally?
about 2 years, OUNL-EML was officially released in December 2000. So officially OUNL-EML is used in operation for one and a half year now. Before that time we also developed a lot of courses in OUNL-EML.

How many courses have been produced using the EML?
more than 50 courses of more than 120 study hours each.

Do academics use the system operationally?
Yes, across the whole institution and within other institutions

Do students use material generated by the system?
Yes

Which pedagogical models does the EML system support?
Nearly every known model can be represented by OUNL-EML

Which pedagogical models does the EML explicitly not support?
As far as we know (and we tested it heavily) all pedagogical models are supported. We do not exclude any model.

How many person-years have been spent in developing the system?
A team of about 30 specialists has worked on it during the development time. The team consisted of educational specialists, researchers, designers, ICT specialists, etc.

Is the EML developed exclusively in-house, or has it been developed within a partnership?
OUNL-EML was developed in-house in our learning technology development laboratory, but we used a lot of companies and other institutions for testing of the concepts and models. For the production of runtime system different companies where involved in the past (Cap Gemini and CMG), the new version of the commercial runtime will be build offered by Perot Systems (from September 2002 onwards).

Is the EML a purely academic development, purely commercial, or a blend of the two?
People, institutions, companies can download and use OUNL-EML for free. The runtime system will be available from Perot Systems. OUNL-EML is completely in the public domain, so other vendors are free to build their own runtimes or authoring environments. Several initiatives in this area are around worldwide (see eml.ou.nl for more information). The aim of the OUNL is to innovate education and that the community can have the benefits of this innovation.

Qualitative factors

What are the authoring tools like? Are they user-friendly or low-level?

At present, the main authoring system uses FrameMaker+SGML. This is not very user friendly, but it works. We want to develop a new authoring tools.

How difficult is it to update content?

Updating is key to the concept of producing re-usable course material. Where material has be re-used, specific instances of the material can be altered without affecting other courses where it might be employed.

To what extent is the course material produced by the system re-useable?

The course material specified in OUNL-EML is fully re-usable, within the constraints of the media employed.

To what extent is the course material produced by the system media-independent?

Since the material takes the form of an XML description, it is media independent.

LMML

DTD

Available at project web site, http://daisy.fmi.uni-passau.de/pakmas/lmml/

A sample (`LMML-CS.dtd`) of the specification is given below to illustrate the information available at the website:

```
<!-- LMML-CS-Content DTD..................................... -->
<!-- file: LMML-CS.dtd.............................. -->
<!--
    This is LMML-CS, a language for specifying teachware content from
    the domain of application of teaching and learning computer science as
    described in Christian Suess, Burkhard Freitag, Peter Broessler,
    "Metamodeling for Web-Based Teachware Management". In: P.P Chen,
    D.W. Embley, J. Kouloumdijan, S.W. Liddle and J.F. Roddick, Proc. Intl. ER'99
    Workshop on the World-Wide Web and Conceptual Modeling, Nov. 15-18 1999,
    Paris, France. LNCS 1727, Springer Verlag,
    http://daisy.fmi.uni-passau.de/db/literatur.php3?key=SFB99

    It is an implementation of the XML binding of the teachware-specific meta-model
    described in Christian Suess, Adaptive Knowledge Management: A Meta-Modeling
    Approach and its Binding to XML". In: Proceedings 11. Workshop Grundlagen von
Datenbanken,
    Arbeitskreis Grundlagen von Informationssystemen im GI-Fachausschuss 2.5,
Ploen, Germany, 2000,
    http://daisy.fmi.uni-passau.de/db/literatur.php3?key=S00

    This is a driver file corresponding to the modularisation mechanism as
    defined in "Building XHTML Modules",  W3C Working Draft 5 January 2000 ,
    http://www.w3.org/TR/2000/WD-xhtml-building-20000105

    Please use this formal public identifier to identify it:
          "-//DE.UNI-PASSAU.DAISY//DTD LMML-CS 1.1//EN"
    Please use this formal system identifier to identify it:
          "LMML-CS.dtd"

    Author:           Christian Suess <suess@fmi.uni-passau.de>
    Date:              Wednesday 12-Apr-2000
    Last Changes:     Wednesday 12-Oct-2000

    This is SUBJECT TO CHANGE, pending final approval.
    Please send your questions, bug reports, comments, or
    suggestions for changes to suess@fmi.uni-passau.de
->
<!--
 ____________________________________________________________________________
|
|   Copyright 1999-2000 Christian Suess, University of Passau, Germany
|
|   All rights reserved.
|
|   Permission to use, copy, modify, and distribute this document for
|   any purposes and without fee is hereby granted in perpetuity,
|   provided that the above copyright notice and this permission notice
|   appear in all copies of this document.
|
|   WE PROVIDE THIS DOCUMENT TO USER "AS IS."  WE DISCLAIM ALL
|   WARRANTIES, EXPRESS OR IMPLIED, WITH RESPECT TO THIS DOCUMENT,
|   INCLUDING THE WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A
|   PARTICULAR PURPOSE.  IN NO EVENT SHALL WE BE LIABLE FOR ANY
```

```
|   DIRECT, INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES,
|   OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA, OR
|   PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE, OR OTHER
|   TORTUOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OF
|   THIS DOCUMENT.
|
|   If you modify the LMML modules and drivers in any way, label your DTDs as a
|   variant of LMML.
|_____________________________________________________________________
->
<!ENTITY % LMML.version "-//DE.UNI-PASSAU.DAISY//DTD LMML-CS 1.1//EN">
<!ENTITY % LMML.ns "lmml">
<!-- ::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::: -->
<!-- Computer science-specific content models ..................... -->
<!ENTITY % Contentobject.extra "| proposition | theorem | proof | algorithm |
formula">
<!ENTITY % Incontentobject.extra "| math | code | LMMLcode">
<!ENTITY % Inline.extra "| math | code | LMMLcode">
<!-- ::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::: -->
<!-- Text extension entities ................................ -->
<!ENTITY % Multimedia.extra "| LMMLtext">
<!ENTITY % Bibitem.content "(LMMLtext)">
<!ENTITY % Inline.nolink.class "| emphasized | quoted | annotated | defined |
formatted %Inline.extra;">
<!-- ::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::: -->
<!-- Modular Framework Module  ................................. -->
<!ENTITY % lmml-framework.mod SYSTEM "LMML11-framework-10.mod">
%lmml-framework.mod;
<!-- ::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::: -->
<!-- Text extension ........................... -->
<!ENTITY % lmml-text.extenstion "INCLUDE">
<![%lmml-text.extenstion;[
         <!ENTITY % lmml-text.ext
           PUBLIC "-//DE.UNI-PASSAU.DAISY//ENTITIES LMML 1.1 Text Extension
1.0//EN"
                  "LMML11-text-10.ext" >
         %lmml-text.ext;]]>
<!-- ::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::: -->
<!-- Computer science-specific modules ......................... -->
<!ENTITY % lmml-cs-spec.mod SYSTEM "LMML11-CS-spec-10.mod">
%lmml-cs-spec.mod;
<!-- end of driver LMML-CS.dtd
```

There is also a ZIP archive at:

http://daisy.fmi.uni-passau.de/pakmas/lmml/11/LMML11-CS-dtds.zip

that contains all the LMML core DTD modules together with the LMML-CS specific DTD modules and the LMML-CS driver file. The LMML XML schema is modularized in a manner similar to XHTML. The archive contains a module LMML11-framework-10.mod which references all DTD modules required for the LMML framework / LMML core. The following modules are integrated using external parameter entities: cmodel, metadata, datatypes, multimedia, table, list, link, and charent. The framework module itself is integrated in the driver file. In case of LMML-CS this is the file LMML-CS.dtd. This driver file also integrates the CS specific module CS-spec and the LMML text extension for inline text. In addition to LMML-CS, there is a DTD LMML-FP for teachware in the field of Financial Planning. Also a LMML-OR for Operation Research is in development. In the near future, there will be a modularization of LMML using XSD.

Sample of course material

Learning Material Markup Language for Computer Science (LMML-CS), a language for specifying teachware from the domain of application of teaching and learning computer

science as described in Christian Süss, Burkhard Freitag, Peter Brössler, Metamodeling for Web-Based Teachware Management (1999). See:
(http://daisy.fmi.uni-passau.de/db/literatur.php3?key=SFB99)

Learning Material Markup Language for Financial Planning (LMML-FP)

Learning Material Markup Language for Operations Research (LMML-OR)

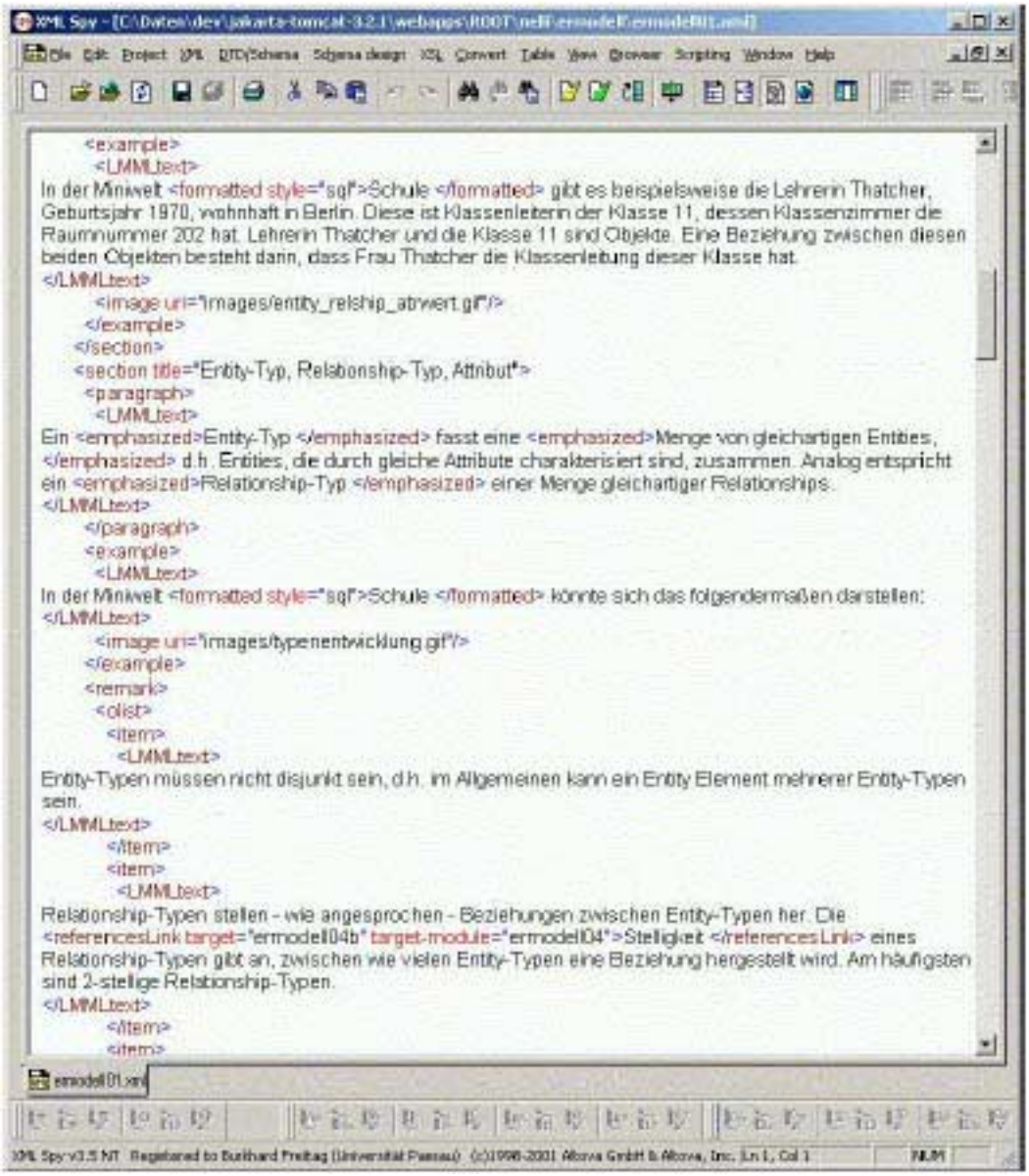
There is a tutorial available which itself is written in LMML-CS:

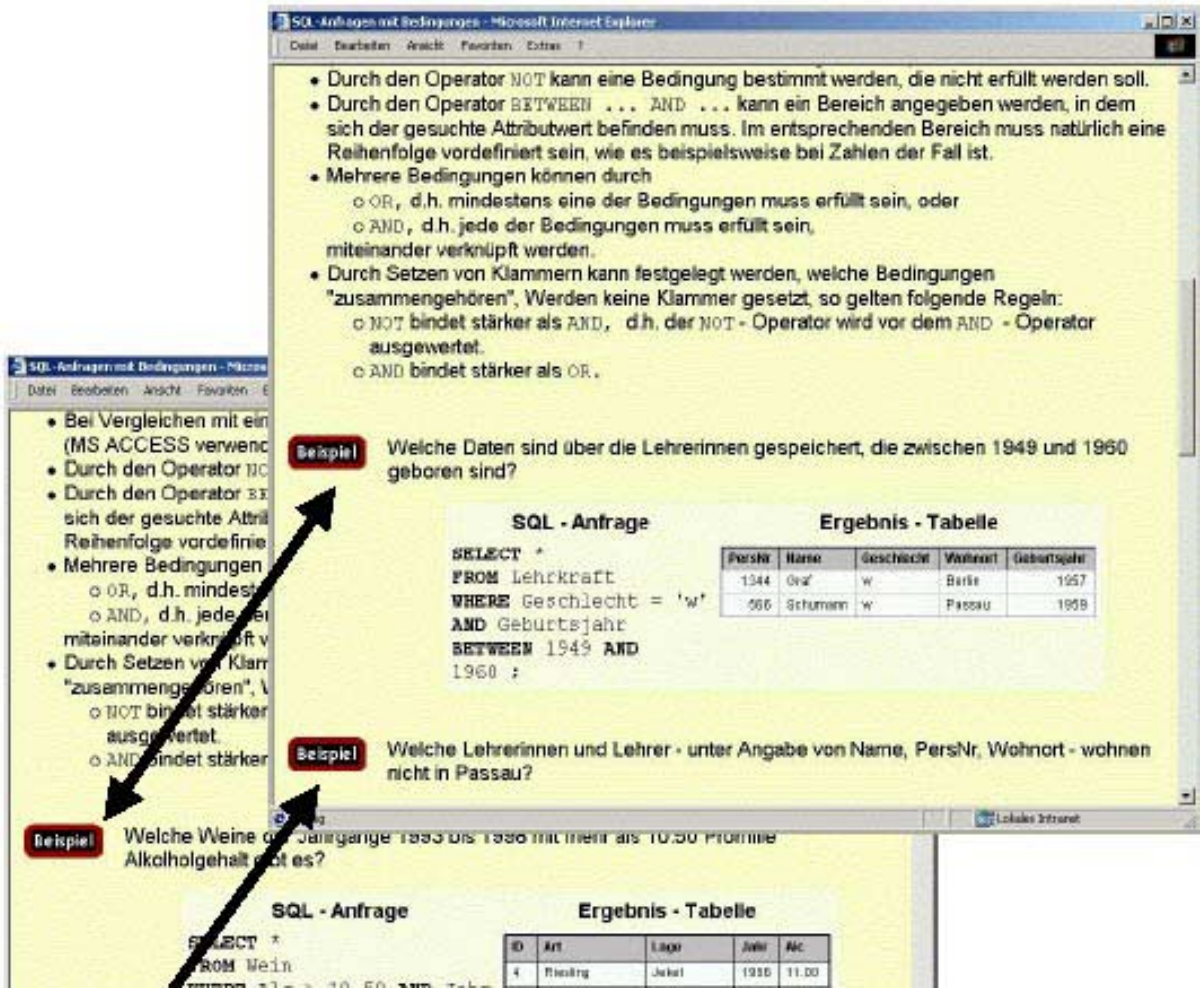
http://daisy.fmi.uni-passau.de/pakmas/lmml/11/doc/de/html/

Sample courses may be seen at :

http://daisy.fmi.uni-passau.de/projekte/nelli/homepages/nelli/passau/demo_db/index.htm

Sample of EML code





Quantitative factors

How long has the EML been in development?

> Development of the XML binding started in 5/1999. LMML version 1.0 was released in 4/2000. Current version LMML 1.1 was released in 12/2000.

How long has the EML been used operationally?

> LMML has been used operationally since its first release in 4/2000. Initially it was used in the projects Nelli (http://www.nelli-bayern.de) and LAMP (http://www.lamp-bayern.de). See also the co-operations listed on the project homepage.

How many courses have been produced using the EML?

> It should be noted that there are instantiations of the LMML framework, i.e. there are different LMML markup languages, e.g.
>
> - Learning Material Markup Language for Computer Science (LMML-CS)
> - Learning Material Markup Language for Financial Planning (LMML-FP)
> - Learning Material Markup Language for Operations Research (LMML-OR)
>
> Nevertheless, there are many courses which have been developed or which are developed using these LMML markup languages:
>
> Already produced:

- Four courses in the project Nelli using LMML-CS: two at the university of Passau, one at Munich and 1 at Erlangen
- Three courses in the project LAMP using LMML-CS: two at the university of Passau and one at Augsburg
- Some financial planning courses at the Gesellschaft für Financial Planning, GFP, Passau using LMML-FP

Currently in production:

At the universities Bern, Lausanne and Zürich a course having 3 variants using LMML-CS

At the universities of Paderborn, Delft, Helsinki and Lappeenranta in the OR-WORLD project some courses in the field of OR/MS using LMML-OR. Many more will be produced in the follow-up project VORMS at eight German universities.

At the university of Passau 7 modules in the domain of foundation of electrical circuits, real time scheduling, real time protocols and softcomputing. How many students have used the EML?

- The Nelli courses are used by about 70 students.
- The material produced in the LAMP project will be used at four universities
- The material produced in the OR-WORLD project will be used by about 300 students at the university of Paderborn as well as many other students at the OR-WORLD and VORMS partner universities.
- The material produced in at Bern, Lausanne and Zürich will be used by about 240 students.

Do academics use the system operationally?

LMML is operationally used mainly by academics but also by industrial partners.

Do students use material generated by the system?

See above.

Many Students use LMML to produce material for the cooperations mentioned above. Students also have used LMML to create term papers.

Which pedagogical models does the EML system support? Which pedagogical models does the EML explicitly not support?

There are hundreds of different pedagogical models, e.g. concerning specific scopes like multiple choice questions. LMML is non-specific in this regard. It incorporates a simple exercise model which can be extended to more complex questions & answer models. LMML also can be used to implement multiple teaching strategies like behaviouristic, cognitivistic or constructivist strategies. See http://daisy.fmi.uni-passau.de/db/literatur?key=SKF00 for more details.

How many person-years have been spent in developing the system?

About 1,5 PY have been spent on LMML and additional 2,5 PY on the development of tools and the support of users.

Is the EML developed exclusively in-house, or has it been developed within a partnership?

The LMML framework has been developed and is being developed completely in-house. The same holds for the LMML-CS instance. The LMML-FP instance was developed with the aid of GFP. The LMML-OR was developed together with a partner from ORWORLD.

Version LMML 2.0 will be developed in coordination with G. Teege's TeachML 2.0 (see above).

Is the EML a purely academic development, purely commercial, or a blend of the two?

LMML was initially a purely academic development. Since spring 2001, however, commercial applications of LMML and commercial variants of LMML are emerging.

Qualitative factors

What are the authoring tools like? Are they user-friendly or low-level?

Authoring of LMML teachware is supported by standard XML tools. LMML module, new DTDs for new LMML instances and XSLT style sheets for LMML transformations can be authored using XML editors and IDEs like the XML Spy (http://www.xmlspy.com/) or XMetaL (http://www.softquad.com/). Authoring with these tools is very user-friendly as e.g. XmetaL provides a WYSIWYG development.

The resulting LMML web courses can be distributed using standard web technologies like the Apache web server. They can be viewed using standard web browsers like the Netscape Navigator or MS Internet Explorer.

There are also some specialized tools being developed. For example, the Passau Knowledge Management System integrated tool set provides a java based ContentBrowser, as well as CourseComposer to (re)arrange LMML modules via drag & drop. In the OR-World project a special LMML-OR editor is being developed which facilitates authoring OR case studies using a forms. Other partners are developing tools for migration from Microsoft Word documents to LMML.

How difficult is it to update content?

LMML content can be updated very easily like any usual XML document. At the moment, there is no versioning in LMML. Versioning has to be done by an appropriate course deployment system.

To what extent is the course material produced by the system re-useable?

The modular structure of LMML Teachware allows for the fragmentation of a domain of application at arbitrary levels of granularity. Thus, material developed in LMML is reusable at all levels of granularity. I.e. authors can reuse media objects like images, sound or animation as well as content modules like motivations and definitions up to course sections, chapters and whole courses. The same course can be reused as a course or even as a subsection of a new course. The same way, a subsection can be used as a course of its own.

To what extent is the course material produced by the system media-independent?

The course material is completely media-independent. LMML only describes the content of teachware together with basic pedagogical information. Content and visualization are strictly separated, the latter being specified by XSL style sheets. Courses developed in LMML (LMML-CS, LMML-OR etc.) can be published e.g. as personal PaKMaS web course as HTML web course as PDF text book via XSL-FO or

latex, as PS text book via latex to mention just a few target media. If one or the other target media is not able to reproduce media objects of certain types, e.g. flash movies in a printed text book, there can be used alternative media objects if provided by the author e.g. a screenshot of that flash movie. Thus, reuse of LMML modules as well as adaptation of content, structure and navigation is made easier.

PALO

DTD

(Samples and documentation can be reached at http://sensei.lsi.uned.es/palo)

The PALO schema is defined by the DTD.

Each template has its own schema, but all of them share approximately the same organisation. Each PALO document defines:

Name of the learning environment

Management information (Domain models, physical location)

Metadata (Dublin Core based binding)

One or more *modules*.

Each module has a hierarchy of structure and each one of the parts can contain any of the following components:

Activities

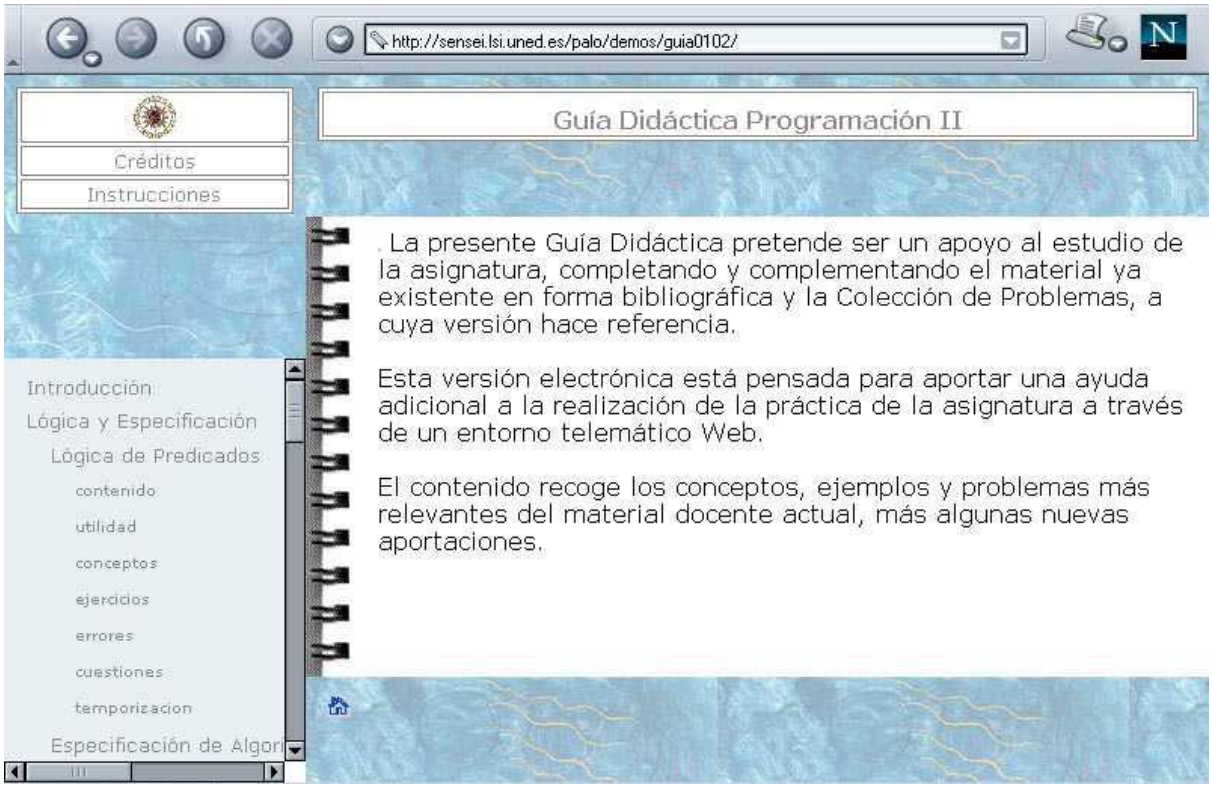
Learning objects

Embedded learning content

Sample of course material

The following example of PALO was downloaded from the website. Note that the example given is a guide, the code for which runs to over sixty pages of A4. For the sake of brevity, only the code that corresponds to the screen dump is given.

The following is a screen dump from the PALO website for a *didactical guide*.



The following DTD defines experimental environments in PALO. (Taken from http://sensei.lsi.uned.es/palo/xml/guia.dtd)

```
 <?xml version="1.0" encoding="ISO-8859-1"?>
<!-- edited with XML Spy v4.1 (http://www.xmlspy.com) by HOLA (HOLA) -->
<!--
    PALO Language Template
    STEED Project   http://sensei.lsi.uned.es/palo

    LSI Dept. UNED University
    Miguel R. Artacho

    Version 3.7  Nov 2001
-->
<!--
    ####################################
    PALO Language
    Course Template
    ####################################
-->
<!ELEMENT course (itemize*, as_is*, management, directory?, module+)>
<!ATTLIST course
    id ID #IMPLIED
    name NMTOKEN #REQUIRED
    use-diectory NMTOKEN #REQUIRED
    assessment-diectory NMTOKEN #REQUIRED
    trace (yes | no) #IMPLIED
>
<!ELEMENT directory (objectives?, credits?, instructions?, requisites?)*>
<!ELEMENT objectives (#PCDATA | itemize | as_is)*>
<!ATTLIST objectives
    id ID #IMPLIED
    trace (yes | no) #IMPLIED
>
<!ELEMENT credits (#PCDATA | itemize | as_is)*>
<!ATTLIST credits
    id ID #IMPLIED
    trace (yes | no) #IMPLIED
```

```
>
<!ELEMENT instructions (#PCDATA | itemize | as_is)*>
<!ATTLIST instructions
    id ID #IMPLIED
    trace (yes | no) #IMPLIED
>
<!ELEMENT requisites (#PCDATA | itemize | as_is)*>
<!ATTLIST requisites
    id ID #IMPLIED
    trace (yes | no) #IMPLIED
>
<!ELEMENT as_is (#PCDATA)>
<!ATTLIST as_is
    id ID #IMPLIED
    type (html | latex | daylight | jme) #REQUIRED
>
<!ELEMENT module (metadata?, itemize*, as_is*, explanation*, glossary*, bd_object*,
bd_relation*, part+)>
<!ATTLIST module
    id ID #IMPLIED
    name NMTOKEN #REQUIRED
    trace (yes | no) #IMPLIED
    label NMTOKEN #IMPLIED
    date NMTOKEN #IMPLIED
    module-prerrequisite NMTOKEN #IMPLIED
    condicion-prerrequisite (aprobado | corregido | ninguno) #IMPLIED
>
<!ELEMENT task (as_is*, explanation*, bd_object*, simulation*, qualifier*)>
<!ATTLIST task
    id ID #IMPLIED
    name NMTOKEN #REQUIRED
    type (texto | latex | test | jme | daylight | fichero | simulation) #REQUIRED
    trace (yes | no) #IMPLIED
    label NMTOKEN #IMPLIED
    assessable (si | no) #IMPLIED
    prerrequisite NMTOKEN #IMPLIED
    mensaje NMTOKEN #IMPLIED
>
<!ELEMENT simulation (#PCDATA | as_is | bd_object | tool)*>
<!ATTLIST simulation
    id ID #IMPLIED
    name NMTOKEN #REQUIRED
>
<!ELEMENT tool (#PCDATA | as_is)*>
<!ATTLIST tool
    id ID #IMPLIED
    type (logica | fisica) #REQUIRED
    url NMTOKEN #REQUIRED
    trace (yes | no) #REQUIRED
>
<!ELEMENT qualifier (#PCDATA | itemize)*>
<!ATTLIST qualifier
    id ID #IMPLIED
    peso NMTOKEN #REQUIRED
>
<!ELEMENT questionaire (#PCDATA | itemize | as_is | task)*>
<!ATTLIST questionaire
    id ID #IMPLIED
    name NMTOKEN #REQUIRED
    label NMTOKEN #IMPLIED
    trace (yes | no) #IMPLIED
>
<!ELEMENT essay (#PCDATA | itemize | as_is | explanation | bd_object | bd_relation
| glossary | task)*>
<!ATTLIST essay
    id ID #IMPLIED
    name NMTOKEN #REQUIRED
    label NMTOKEN #IMPLIED
```

```
    trace (yes | no) #IMPLIED
>
<!ELEMENT glossary (#PCDATA | reference)*>
<!ATTLIST glossary
    id ID #IMPLIED
    category NMTOKEN #REQUIRED
    domain NMTOKEN #REQUIRED
    label-attrib NMTOKEN #REQUIRED
    content-attrib NMTOKEN #REQUIRED
    trace (si | no) #IMPLIED
    order (alfabetico | secuencial) #IMPLIED
>
<!ELEMENT reference (#PCDATA)>
<!ATTLIST reference
    id ID #IMPLIED
    name NMTOKEN #REQUIRED
    domain NMTOKEN #REQUIRED
    category NMTOKEN #IMPLIED
    content-attrib NMTOKEN #REQUIRED
    position (antecedent | consequent) #IMPLIED
>
<!ELEMENT explanation (#PCDATA | content-explanation)*>
<!ATTLIST explanation
    id ID #IMPLIED
    name NMTOKEN #IMPLIED
    trace (yes | no) #IMPLIED
>
<!ELEMENT content-explanation (#PCDATA | as_is | explanation)*>
<!ELEMENT part (#PCDATA | itemize | as_is | explanation | questionaire | bd_object
| bd_relation | task | essay | glossary | subpart)*>
<!ATTLIST part
    id ID #IMPLIED
    name NMTOKEN #REQUIRED
    trace (yes | no) #IMPLIED
>
<!ELEMENT subpart (#PCDATA | itemize | as_is | bd_object | bd_relation | task |
glossary | explanation | essay)*>
<!ATTLIST subpart
    id ID #IMPLIED
    name NMTOKEN #REQUIRED
    ref NMTOKEN #IMPLIED
    trace (yes | no) #IMPLIED
>
<!ELEMENT itemize (#PCDATA | item)*>
<!ELEMENT item (#PCDATA | itemize | as_is | bd_object | bd_relation | glossary |
explanation)*>
<!ELEMENT bd_object (#PCDATA | links)*>
<!ATTLIST bd_object
    id ID #IMPLIED
    domain NMTOKEN #REQUIRED
    name NMTOKEN #REQUIRED
    category NMTOKEN #REQUIRED
    label-attrib NMTOKEN #IMPLIED
    content-attrib NMTOKEN #IMPLIED
    trace (yes | no) #IMPLIED
    faq (yes | no) #IMPLIED
>
<!ELEMENT bd_relation (#PCDATA)>
<!ATTLIST bd_relation
    id ID #IMPLIED
    name NMTOKEN #REQUIRED
    domain NMTOKEN #REQUIRED
    subject NMTOKEN #REQUIRED
    trace (yes | no) #IMPLIED
    atrib NMTOKEN #IMPLIED
    category NMTOKEN #IMPLIED
>
<!ELEMENT links (#PCDATA | bd_object | bd_relation | glossary)*>
```

```
<!ATTLIST links
    id ID #IMPLIED
>
<!ELEMENT management (objectsDB, tasksDB, metadata?)>
<!ELEMENT objectsDB (#PCDATA)>
<!ATTLIST objectsDB
    id ID #IMPLIED
    type (trial | exploitation) #REQUIRED
    sgdb (mSQL | Oracle) #REQUIRED
    location NMTOKEN #IMPLIED
>
<!ELEMENT tasksDB (#PCDATA)>
<!ATTLIST tasksDB
    id ID #IMPLIED
    type (trial | exploitation) #REQUIRED
    sgdb (mSQL | Oracle) #REQUIRED
    location NMTOKEN #IMPLIED
>
<!ELEMENT metadata (content, copyright, instance)>
<!ATTLIST metadata
    id ID #IMPLIED
    type (dc | ims | ieee) #REQUIRED
    cod (rfc2731) #REQUIRED
>
<!ELEMENT content (title, matter, description, origin, language, relation, scope)>
<!ELEMENT copyright (author, editor, colaborador, rights)>
<!ELEMENT instance (date, type, format, identifier)>
<!ELEMENT title (#PCDATA)>
<!ELEMENT matter (#PCDATA)>
<!ELEMENT description (#PCDATA)>
<!ELEMENT origin (#PCDATA)>
<!ELEMENT language (#PCDATA)>
<!ELEMENT relation (#PCDATA)>
<!ELEMENT scope (#PCDATA)>
<!ELEMENT author (#PCDATA)>
<!ELEMENT editor (#PCDATA)>
<!ELEMENT colaborador (#PCDATA)>
<!ELEMENT rights (#PCDATA)>
<!ELEMENT date (#PCDATA)>
<!ELEMENT type (#PCDATA)>
<!ELEMENT format (#PCDATA)>
<!ELEMENT identifier (#PCDATA)>
<!ENTITY lt "menorque">
<!ENTITY gt "mayorque">
<!ENTITY amp "ampersand">
```

The following was taken from: http://sensei.lsi.uned.es/palo/xml/Program_Verification_Guide.xml

The code below corresponds to the screen dump above.

```
<?xml version="1.0"?>
<!--
PALO Language EML System
File:Program_Verification_Guide.xml

This file describes an educational environment written in PALO
The contents, elements and information contained in this file are copyrighted
by the Dept. of Languages and Computer Systems at UNED University

More information:  http://sensei.lsi.uned.es/palo
->
<GUIA
NOMBRE="GuÃa Didáctica Programación II"
DIR="guiap29900"
TRAZA="si"
```

```
>

<GESTION
>  <BDOBJETOS
TIPO="pruebas"
SGDB="mSQL"
>ProgII_Faq</BDOBJETOS>
  <BDOBJETOS
TIPO="explotacion"
SGDB="mSQL"
>ProgII_Faq</BDOBJETOS>
  <BDTAREAS
TIPO="pruebas"
SGDB="mSQL"
>pruebas_progII</BDTAREAS>
  <BDTAREAS
TIPO="explotacion"
SGDB="mSQL"
>PROG2_9900</BDTAREAS>

  <METAINFORMACION
TIPO="dc"
COD="rfc2731"
>
    <CONTENIDO
>      <AMBITO
>UNED Curso 9900</AMBITO>
      <FUENTE
>Asignatura de Programación II</FUENTE>
      <DESCRIPCION
>Guía Didactica de Programación II</DESCRIPCION>
      <LENGUAJE
>Español</LENGUAJE>
      <MATERIA
>ProgramacióOn</MATERIA>
      <RELACIONADO
>Asigntatura de Programación II</RELACIONADO>
      <TITULO
>Guía Didáctica de ProgramaciOn II</TITULO>
    </CONTENIDO>
    <COPYRIGHT
>      <AUTOR
>Miguel Rodriguez Artacho</AUTOR>
      <COLABORADOR
>Yolanda Calero Caro</COLABORADOR>
      <DERECHOS
>UNED. Proyecto STEED</DERECHOS>
      <EDITOR
>UNED</EDITOR>
    </COPYRIGHT>
    <INSTANCIA
>      <FECHA
>7-4-2000</FECHA>
      <FORMATO
>PALO v3.0</FORMATO>
      <IDENTIFICADOR
></IDENTIFICADOR>
      <TIPO
></TIPO>
    </INSTANCIA>
  </METAINFORMACION>

</GESTION>



  La presente Guía Didáctica pretende ser un apoyo al estudio de la
  asignatura, completando y complementando el material ya existente en
```

```
forma bibliográfica y la Colección de Problemas, a cuya versión
hace  referencia.


Esta versiOn electrónica estó pensada para aportar una ayuda
adicional a la realizaciOn de la prÁctica de la asignatura a través
de un entorno telemático Web.

El contenido recoge los conceptos, ejemplos y problemas mÁs
relevantes del material docente actual, mÁs algunas nuevas
aportaciones.

INSTRUCCIONES

La GuÃa puede recorrerse en cualquier orden y tiene la intenciOn de
servir de ayuda para la comprensiOn de los conceptos bÁsicos de los
temas de la asignatura. Los temas estÁn estructurados de la misma
forma que lo estÁn en la versiOn escrita. Además, se han añadido
ejemplos  y problemas nuevos para que sea posible ilustrar todos los
conceptos del temario.
```

Quantitative factors

How long has the EML been in development?

4 years

How long has the EML been used operationally?

4 years

How many courses have been produced using the EML?

A total of 11 courses structured as follows:

- 3 courses for a regular matter of B.S. degree on CS at UNED
- 4 Open Courses for non-regular students (UNED Open Courses)
- 6 virtual laboratories for Industrial Engineering degree at UNED

Do academics use the system operationally?

Yes

Do students use material generated by the system?

Yes

Which pedagogical models does the EML system support?

Actually both behaviourist and constructivist features can be added to a PALO defined learning environment

Which pedagogical models does the EML explicitly not support?

First versions (1.0 to 3.7) were designed to provide only individual learning. Cooperative activities can be defined in the 4.0 version, to appear in Jan 2002.

How many person-years have been spent in developing the system?

PALO development team is composed by 2 researchers and 1 developer. Man power needed to develop the software was 3/year.

Is the EML developed exclusively in-house, or has it been developed within a partnership?

In-house.

Is the EML a purely academic development, purely commercial, or a blend of the two?

At its first stages was conceived as an academic development and provided with a GPL copyright license.

Actually one of our intentions is oriented to provide PALO compiler freely available as a Linux-Debian package by Jun 2002.

Qualitative factors

What are the authoring tools like? Are they user-friendly or low-level?

No authoring tools have been developed by the moment. The mail objective was to develop the language. Authoring tools are actually SGML and XML editors, like Emacs.

How difficult is it to update content?

Stages of an upgrade of a learning environment are:

- Edit and update the PALO file
- Compile the PALO file in *test mode*
- Check consistency and confirm the changes with the resulting environment
- Compile the PALO file in *production mode*

A PALO compilation takes between 1-3 minutes.

To what extent is the course material produced by the system re-useable?

Totally reusable. Any PALO file can be inserted in another PALO file according to the DTD.

To what extent is the course material produced by the system media-independent?

Everything but *as-is* learning objects.

PALO language incorporates a tag that allow to embed directly a chunk of content in a given format (HTML, LaTeX, Smiles, …). These capability is as useful and as disturbing as pointers in any programming language.

Supplementary information

Name of framework

PALO Language

http://www.palolanguage.com

http://sensei.lsi.uned.es/palo

Scope, e.g., What is the pedagogical domain? Which pedagogical entities are included?

PALO defines learning scenarios by mean of instructional templates. An instructional template defines a type of learning scenario with certain pedagogical properties. Pedagogical domain is defined by mean of elements of the language that provides different functionality.

Instructional templates, however, do not define different "languages" by themselves, but a subset of the element of the language that provides a certain type of pedagogical functionality. A group of consistent elements of PALO that provide an instructional or pedagogical purpose constitute a template.

PALO is intended to be pedagogically neutral. Elements of the language provide, however, the capability of configuring learning scenarios based on both behaviourist and constructivist approaches.

Sample courses

Fully functional courses are available at the DEMO page of PALO.

Experiences of users

Experiences have been taken during the following years:

- **1998**: Regular matter of B.S. Degree on Computer Science. Three learning environments to provide support to a population of ~7000 students.
- **1999-2002:** Also offering open courses developed in PALO to a population of ~700 students.

Academic staff other than the staff directly involved in PALO development has also created PALO content.

Interoperability

PALO incorporate DC Metadata specifications

At the moment, a study is taken place to create an import/export capability to Tec-Infor learning platform (http://www.uned.es/iued).

Further publications

Papers and articles are available at PALO website: http://sensei.lsi.uned.es/palo

Targeteam

DTD

```
TeachML documents may have a document type declaration. If it is present it must
use the public id:
"-//TU-Muenchen//DTD TeachML 1.2 Modularization//EN"
A TeachML Document with a Document Type Declaration
<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE teachml
PUBLIC "-//TU-Muenchen//DTD TeachML 1.2
Modularization//EN"
"TeachML-Mod.dtd">
Download from : <teachml xmlns="http://www11.in.tum.de/XMLspec/TeachML" >
<?xml version="1.0" encoding="ISO-8859-1"?>

<!-- ................................... -->
<!-- TeachML 1.2 Integrated Materials Language -->
<!-- file: TeachML-Int.dtd

   This is the TeachML language for complete integrated
   TeachML materials. It consists of all TeachML language
   modules but TeachML Integration.

   Note, that this is not the real DTD, as it is used in the
   Targeteam system. The real DTD is modularized into several files,
   according to DTD modularization in XHTML, and it makes heavy use of
   parameter entities. This file contains all DTD modules and
   most parameter entities have been eliminated for better
   readability. This file is provided for getting a first quick
   impression of the Targeteam DTD.

   More information about the elements, their semantics, their use and
   examples can be found in the Targeteam documentation, available at
   the Targeteam homepage:
     http://www11.in.tum.de/forschung/projekte/targeteam/

   ..................................... -->

<!-- The document element is teachml. -->

<!-- TeachML document element ..................... -->
<!ELEMENT teachml ( module ) >

<!-- Next, all sublanguages for content are integrated. Each
     sublanguage XXX contributes the elements in ContentXXX.mix
     to the elements which may be used anywhere in content. -->

<!-- Language Core -->
<!-- content contribution elements: -->
<!ENTITY % ContentCore.mix
  "note | definition | code | defined | emph | quote | ref | whatsit | xor"
 >

<!-- complete element set: -->
<!ENTITY % ElementsCore.mix
  "%ContentCore.mix; | header | intro | kernel
```

```
   | details | illustration | exercises | metaissues | summary
   | module | issue " >

<!-- Sublanguage Box -->
<!-- complete element set: -->
<!ENTITY % ElementsBox.mix "vbox | hbox | cbox | ivbox | ihbox" >

<!-- content contribution elements: all -->
<!ENTITY % ContentBox.mix "%ElementsBox.mix;" >

<!-- Sublanguage Tup -->
<!-- complete element set: -->
<!ENTITY % ElementsTup.mix "tuples | metatuple | tuple | ten" >

<!-- content contribution element: tuples -->
<!ENTITY % ContentTup.mix "tuples" >

<!-- Sublanguage Astep -->
<!-- complete element set: -->
<!ENTITY % ElementsAstep.mix "atom-stepping | atoms | step | nosteps" >

<!-- content contribution elements: all -->
<!ENTITY % ContentAstep.mix "%ElementsAstep.mix;" >

<!-- Additionally, content elements which are only present
     after integration (this is the atom element) -->
<!-- complete element set: -->
<!ENTITY % ElementsIntegrated.mix "atom" >

<!-- content contribution elements: all -->
<!ENTITY % ContentIntegrated.mix
  "%ElementsIntegrated.mix;"
 >

<!ENTITY % Content.mix
  "| %ContentCore.mix; | %ContentBox.mix; | %ContentTup.mix; |
   %ContentAstep.mix; | %ContentIntegrated.mix;" >

<!ENTITY % Elements.mix
  "| %ElementsCore.mix; | %ElementsBox.mix; | %ElementsTup.mix; |
   %ElementsAstep.mix; | %ElementsIntegrated.mix;" >

<!-- Now we define the language modules. They only
     interact via the Content.mix parameter. -->

<!-- TeachML core language ..................... -->
<!-- Parameter defaults: -->
<!ENTITY % Inissue.class
  "header | intro | kernel
   | details | illustration | exercises | metaissues | summary"
 >

<!-- The Root Element: module.  -->

<!ELEMENT module ( issue )+ >
<!ATTLIST module
    id ID #IMPLIED >

<!-- ******************************** -->
<!-- The sublanguage for structuring   -->
```

```
<!ELEMENT issue ( %Inissue.class; )* >
<!ATTLIST issue
     id ID #IMPLIED
     kind CDATA #IMPLIED
     author CDATA #IMPLIED
     affiliation CDATA #IMPLIED >

<!ELEMENT header ( #PCDATA %Content.mix; )* >

<!ELEMENT intro ( #PCDATA %Content.mix; )*" >
<!ATTLIST intro id ID #IMPLIED>

<!ELEMENT kernel ( #PCDATA %Content.mix; )* >
<!ATTLIST kernel id ID #IMPLIED>

<!ELEMENT summary ( #PCDATA %Content.mix;  )* >
<!ATTLIST summary id ID #IMPLIED>

<!ELEMENT xor ( alt  )* >

<!ELEMENT alt ( #PCDATA | issue %Content.mix;  )* >
<!ATTLIST alt types NMTOKENS "standard">

<!-- Issue groups: -->

<!ENTITY % Issuegroup.attrib "
  id ID #IMPLIED
  header CDATA #IMPLIED
  labels (numbers | letters | capletters | dots) #IMPLIED
  first-label CDATA #IMPLIED
  label-pre CDATA #IMPLIED
  label-post CDATA #IMPLIED
" >

<!ELEMENT details ( issue | xor )* >
<!ATTLIST details %Issuegroup.attrib; >

<!ELEMENT illustration ( issue | xor )* >
<!ATTLIST illustration %Issuegroup.attrib; >

<!ELEMENT metaissues ( issue | xor )* >
<!ATTLIST metaissues %Issuegroup.attrib; >

<!ELEMENT exercises ( issue | xor )* >
<!ATTLIST exercises %Issuegroup.attrib; >

<!-- ********************************* -->
<!-- The sublanguage for textual content -->

<!ELEMENT definition ( #PCDATA %Content.mix;  )* >
<!ATTLIST definition id ID #IMPLIED>

<!ELEMENT note ( #PCDATA %Content.mix;  )* >
<!ATTLIST note id ID #IMPLIED>

<!ELEMENT code ( #PCDATA %Content.mix;  )* >

<!ELEMENT defined ( #PCDATA %Content.mix;  )* >

<!ELEMENT emph ( #PCDATA %Content.mix;  )* >
```

```
<!ELEMENT quote ( #PCDATA %Content.mix;  )* >

<!ELEMENT whatsit ( #PCDATA %Content.mix;  )* >
<!ATTLIST whatsit kind CDATA #REQUIRED>

<!-- ********************************** -->
<!-- The sublanguage for cross references -->

<!ELEMENT ref ( #PCDATA %Content.mix;  )* >
<!ATTLIST ref target-module NMTOKEN #IMPLIED
              target-atom NMTOKEN #IMPLIED
              target NMTOKEN #IMPLIED
              url CDATA #IMPLIED>

<!-- ********************************** -->
<!-- The sublanguage for simple includes  -->

<!ELEMENT include EMPTY >
<!ATTLIST include select CDATA #REQUIRED>

<!-- TeachML Box .................. ..................... -->

<!ELEMENT vbox ( %ElementsBox.mix;  )+ >
<!ATTLIST vbox
    width CDATA #IMPLIED
    id ID #IMPLIED >

<!ELEMENT hbox ( %ElementsBox.mix;  )+ >
<!ATTLIST hbox
    width CDATA #IMPLIED
    id ID #IMPLIED >

<!ELEMENT cbox ( #PCDATA %Content.mix;  )* >
<!ATTLIST cbox
    width CDATA #IMPLIED
    id ID #IMPLIED >

<!ELEMENT ivbox ( %ElementsBox.mix;  )+ >
<!ATTLIST ivbox
    width CDATA #IMPLIED
    id ID #IMPLIED >

<!ELEMENT ihbox ( %ElementsBox.mix;  )+ >
<!ATTLIST ihbox
    width CDATA #IMPLIED
    id ID #IMPLIED >

<!-- TeachML Tup .................. ..................... -->

<!ELEMENT tuples ( metatuple?, tuple+ ) >
<!ATTLIST tuples
    arity CDATA #IMPLIED
    id ID #IMPLIED >

<!ELEMENT metatuple ( ten+ ) >

<!ELEMENT tuple ( ten+ ) >

<!-- "ten" is the abbreviation of Tuple ENtry -->
<!ELEMENT ten ( #PCDATA %Content.mix;  )* >
```

```
<!-- TeachML Astep ................. ..................... -->

<!ELEMENT atom-stepping ( #PCDATA %Content.mix;  )* >
<!ATTLIST atom-stepping
    id ID #IMPLIED  >

<!ELEMENT atoms ( atom )+ >

<!ELEMENT step ( #PCDATA %Content.mix;  )* >
<!ATTLIST step
    atom CDATA #REQUIRED
    initial (true|false) "false" >

<!ELEMENT nosteps EMPTY >
<!ATTLIST nosteps
    atom CDATA #REQUIRED >

<!-- TeachML Integration extensions  ................... -->
<!ELEMENT atom ( alternative )+ >
<!ATTLIST atom
    id ID #IMPLIED >

<!ELEMENT alternative EMPTY >
<!ATTLIST alternative
    name         CDATA #REQUIRED
    extension    CDATA #REQUIRED
    type         CDATA #REQUIRED
    derived-from CDATA #IMPLIED
    derived-by   CDATA #IMPLIED>
```

DTD, samples, documentation (=sample) available at the Targeteam homepage :
htttp://www11.in.tum.de/forschung/projekte/targeteam/

Sample of course material

Currently, about 10 half-year-courses in Informatics at Technische Universität München, Universität der Bundeswehr München and Virtuelle Hochschule Bayern.

Tutorial material is available as part of the documentation.

Sample courses may be seen at the Targeteam homepage :
htttp://www11.in.tum.de/forschung/projekte/targeteam/

Sample of EML code

```
Every Targeteam module is a separate XML document.
The following example consists of two modules. The first module
includes content from the second module.

---------------------- module os_main ----------------------
<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE teachml PUBLIC "-//TU-Muenchen//DTD TeachML 1.2 Modularization//EN"
"TeachML-Mod.dtd">
<teachml
    xmlns="http://www11.in.tum.de/XMLspec/TeachML"
    xmlns:t="http://www11.in.tum.de/XMLspec/TeachML"
    xmlns:xsl="http://www.w3.org/1999/XSL/Transform">
    <module>
        <issue>
            <header>Introduction to Operating Systems</header>
            <kernel>
    <xor><alt types="short interactive">
```

```
                    Operating system: abstracting hardware for application
programs.
                </alt><alt types="full">
                    An operating system is a software level which provides an
                    abstraction of the computer hardware. This abstraction is
                    used as an <quote>interface</quote> by application programs.
                </alt></xor>
                <atom>
                    <alternative name="oslevel.swf" type="application/x-shockwave-
flash"/>
                    <alternative name="oslevel.eps" type="application/postscript"/>
                </atom>
            </kernel>
            <illustration>
    <issue>
        <header>Examples of well-known Operating Systems</header>
        <kernel>
            Overview of the history of unix and windows systems.
        </kernel>
        <details>
            <!-- include the 'history' subsections from modules
                                'unix' and 'windows' with modifications applied.
-->
            <xsl:apply-templates
select="t:module[@id='unix']/t:issue/t:details/t:issue[@id='history']"/>
            <xsl:apply-templates
select="t:module[@id='windows']/t:issue/t:details/t:issue[@id='history']"/>
        </details>
    </issue>
    <issue>
          ....
                </issue>
            </illustration>
        </issue>
    </module>

    <!-- Modifications to be applied to included modules -->
    <!-- Substitute headers 'history' issues -->
    <xsl:template

    match="t:module[@id='unix']/t:issue/t:details/t:issue[@id='history']/t:header">
        <header>Unix Systems</header>
    </xsl:template>
    <xsl:template

    match="t:module[@id='windows']/t:issue/t:details/t:issue[@id='history']/t:heade
r">
        <header>Windows Systems</header>
    </xsl:template>
</teachml>
-----------------------------------------------------------------

-------------------- module unix -------------------------------
<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE teachml PUBLIC "-//TU-Muenchen//DTD TeachML 1.2 Modularization//EN"
"TeachML-Mod.dtd">
<teachml xmlns="http://www11.in.tum.de/XMLspec/TeachML">
    <module>
        <issue>
            <header>Case Study: Unix and Linux</header>
            <kernel>
    Here we look at how the operating system concepts are used in
    the Unix and Linux systems.
            </kernel>
            <details>
    <issue id="history"><header>History of Unix</header>
        <kernel>
            This is an overview over the predecessors and variants of
```

```
            the Unix operating system.
        </kernel>
        <details>
            <issue><header>Predecessors</header>
                <kernel>....     </kernel>
                <details>
        <issue><header>CTSS</header>
            <kernel> .... </kernel>
        </issue>
        <issue>
            <header>MULTICS</header>
            <kernel> .... </kernel>
        </issue>
        <issue>
            <header>UNICS</header>
            <kernel> .... </kernel>
        </issue>
                </details>
            </issue>
            <issue>
                <header>AT&amp;amp;T Unix</header>
                            ....
            </issue>
                        ....
        </details>
    </issue>
            </details>
        </issue>
    </module>
</teachml>
-----------------------------------------------------------------
```

Quantitative factors

How long has the EML been in development?

Since March 1999.

How long has the EML been used operationally?

Since May 1999.

How many courses have been produced using the EML?

About 10 half-year courses, some of them repeated in several versions.

Do academics use the system operationally?

Yes

Do students use material generated by the system?

Yes

Which pedagogical models does the EML system support?

Neutral

Which pedagogical models does the EML explicitly not support?

None

How many person-years have been spent in developing the system?

About 1 ½.

Is the EML developed exclusively in-house, or has it been developed within a partnership?

Started as in-house, now as partnership.

Is the EML a purely academic development, purely commercial, or a blend of the two?

purely academic.

Qualitative factors

What are the authoring tools like? Are they user-friendly or low-level?

Mixture of standard tools (any XML editor can be used) and system-specific tools (still low-level, graphical user interface under development).

How difficult is it to update content?

Ranges from easy (rewrite and publish again) to complex (specify transformation to adapt foreign content to own context).

To what extent is the course material produced by the system re-useable?

Fully re-usable, this is the main goal of the system.

To what extent is the course material produced by the system media-independent?

Original content is in XML, delivery formats are HTML, PDF, Postscript. However, specific media formats can be included (currently gif, png, jpeg, flash animation, java applets) which are only partly media independent.

Supplementary information

Scope, e.g., What is the pedagogical domain? Which pedagogical entities are included?

All kinds, especially higher education .

Instances of this framework

Currently, about 10 half-year-courses in Informatics at Technische Universität München, Universität der Bundeswehr München and Virtuelle Hochschule Bayern.

Schema / DTD, samples, documentation

DTD, samples, documentation (=sample) available at the Targeteam homepage :
htttp://www11.in.tum.de/forschung/projekte/targeteam/

Tutorial

Available as part of the documentation.

Sample courses

Links are available at the Targeteam homepage :
htttp://www11.in.tum.de/forschung/projekte/targeteam/

Experiences of users

Successful use since 1999 in daily work by 3 authors. For details contact the project leader Gunnar Teege.

Interoperability

Delivery formats are standard Web formats such as HTML, PDF, Flash.

Using legacy content from other XML DTDs or proprietary formats (MS Word, PowerPoint) is possible but needs manual assistance.

Further publications

In preparation.

A publication (in German) about the use of Targeteam in context is:

Gunnar Teege, Jürgen Koch, Pamela Tröndle, Wolfgang Wörndl, Johann Schlichter

ModuVille: Komponenten für virtuelle WWW-basierte Lehrveranstaltungen
PIK - Praxis der Informationsverarbeitung und Kommunikation, pp. 148-155, 2000

TML / Netquest

DTD

The Scheme may be downloaded from:

http://www.ilrt.bris.ac.uk/netquest/liveserver/TML_INSTALL/lib/ETS/dtd/TML_4.0

This document runs to seventeen pages of definitions; it hasn't been included in this report for reasons of brevity.

Sample of course material

Demonstration tutorials are available at the Liverpool NT Port :
http://www.dfm.livjm.ac.uk/cgi-tml/tml_nt_demo.pl/tml/example/demo.tmlo?Index=%2F

An example of how the above MCQ looks in action can be found at the following link. Some screen dumps from this website are shown below.

http://www.chm.bris.ac.uk/cgi-bin/tml_v4.2_anon/tml/questions/demo_v4.2/demo.tmlo

Sample of text produced by the TML/Netquest system:

Next question. Introduction. List questions.

Question 1 of 7. How to answer this type of question.
Score for this question : 0 from 3. This is your first attempt.

This is an example of a simple multiple choice text-based question.

Which of the following is a **noble gas** ?

a) Magnesium
b) HCl
c) Xenon
d) Nitrogen
e) HCN

Hints are available.

When incorrect answer is selected, the following text was displayed:

Next question. Introduction. List questions.

Question 1 of 7. How to answer this type of question.
Score for this question : 0 from 3. You have had one attempt.

This is an example of a simple multiple choice text-based question.

Which of the following is a **noble gas** ?

You have answered : e) HCN
Sorry, that is not the correct answer. Maybe you should ask for a hint?

a) Magnesium
b) HCl
c) Xenon
d) Nitrogen
e) HCN

Hints are available.

A Hint was requested, and this resulted in the following:

Next question. Introduction. List questions.

Question 1 of 7. How to answer this type of question.
Score for this question : 0 from 3. You have had one attempt.

This is an example of a simple multiple choice text-based question.

Which of the following is a **noble gas** ?

a) Magnesium
b) HCl
c) Xenon
d) Nitrogen
e) HCN

Hints are available.
1) A noble gas is very **unreactive**

Sample of EML code

TML is written using a series of mark-up tags similar to HTML. It is compatible with SGML and DTD. The questions are written as separate files, named q001.tml, q002.tml,..., along with a header and footer file. These are then compiled into a version that can be executed by the cgi script, but *not* directly read by a browser (security measure).

Header File

This contains the start tags for the document, as well as any introductory information you may like to give the students before they begin the MCQ test. This introductory info can be a single web page, or many linked pages, or even an entire lecture course!

The entire MCQ test is enclosed between two tags <TML> and </TML>, and in the header there is a tag called <TUTORIAL> which contains the path to the cgi program that will execute the code.

```
<!DOCTYPE TML PUBLIC "-//ETS//DTD TML 4.0//EN//" [ ] >
<TML>
<HEAD>
<TITLE>This is an example MCQ test for Chemistry</TITLE>
</HEAD>
<BODY>
Introductory info goes here
</BODY>
<TUTORIAL ACTION="/TML/cgi-bin/tutorial">
```

Footer File

This file simply closes all the open tags at the end of the document.

```
</TUTORIAL>
</TML>
```

Example Question File

Each separate question is enclosed with the <QUESTION>, </QUESTION> tags, and other tags define Choices, Score, Hints and Responses.

```
<QUESTION TYPE=Multiple-Choice ATTEMPTS=2>

<I>Which of the following is a noble gas? </I>

<CHOICES>
       <CHOICE OPTION=a> Magnesium
       <CHOICE OPTION=b> HCl
       <CHOICE CORRECT OPTION=c> Xenon
       <CHOICE OPTION=d> Nitrogen
       <CHOICE OPTION=e> HCN
</CHOICES>

<SCORE>
       <GAIN CORRECT VALUE=3 ATTEMPT=1>
       <GAIN CORRECT VALUE=1 ATTEMPT=2>
       <LOSE HINT VALUE=1>
</SCORE>

<HINTS>
       <HINT>A noble gas is very <B>unreactive</B>
       <HINT>It's <B>not</B> likely to be something corrosive,
```

```
        poisonous or solid...
        <HINT>You have lost 3 points already asking for hints so far...
        <HINT>Asking for too many hints can leave you with a negative
        score !
</HINTS>

<RESPONSES>
        <WHEN CORRECT><B>Congratulations, that is correct !</B>
        <WHEN OPTION=d>No, although nitrogen is inert, it's not
        classified as a noble gas.
        <WHEN INCORRECT>Sorry, that is not the correct answer. Maybe
        you should ask for a hint?
        <WHEN COMPLETE>A question completion comment
</RESPONSES>

</QUESTION>
```

The header, questions, and footer are put into an appropriate directory in the web tree, and compile to form one executable file called `filename.tmlo`. This directory is protected so that only the cgi-bin program can read its contents. The MCQ test is accessed using the path given in the original `<TUTORIAL ACTION="/TML/cgi-bin/tutorial">` tag.

Quantitative factors

How long has the EML been in development?

Developed in 1994. Recently revisited by revision is work in progress.

How long has the EML been used operationally?

Since 1994 but has effort switched to supporting IMS and feeding into that.

How many courses have been produced using the EML?

7-8

Do academics use the system operationally?

Not any more although content deployed is still in use.

Do students use material generated by the system?

Yes

Which pedagogical models does the EML system support?

Self-study and formative assessment and formal summative assessment

How many person-years have been spent in developing the system?

Less than 1.

Is the EML developed exclusively in-house, or has it been developed within a partnership?

Open source. Has been adapted from Unix to NT by other universities.

Is the EML a purely academic development, purely commercial, or a blend of the two?

Purely academic.

Qualitative factors

What are the authoring tools like? Are they user-friendly or low-level?

Text-based mark up. No tools

How difficult is it to update content?

No management tools, so text files must be edited.

To what extent is the course material produced by the system re-useable?

Highly. Content could easily be ported to IMS and will be ported to RDF as part of TML5

To what extent is the course material produced by the system media-independent?

High degree.