

Die Strukturierung von Lehreinheiten mit XML-basierten Modellierungssprachen

Diplom-Forstwirt Univ. Sebastian Paar

Fachgebiet für Biometrie und Angewandte Informatik Department für Ökosystem- und Landschaftsmanagement am Wissenschaftszentrum Weihenstephan für Ernährung, Landnutzung und Umwelt der Technischen Universität München

Zusammenfassung

Die Extensible Markup Language (XML) bietet als Metasprache die Möglichkeit eigene Sprachen über verschiedene Document Type Definition's (DTD's) zu definieren.

Educational Modelling Language (EML) stellt eine dieser Sprachen dar. Sie besitzt eine eigene erweiterbare DTD. Das besondere an EML liegt darin begründet, dass nicht bloß Lerninhalte, sondern auch Rollen, Beziehungen, Interaktionen und Aktivitäten von Lernenden und Lehrenden integriert werden. Dabei können verschiedene Lernmodelle (z. B. problemorientiertes Lernen, Selbststudium) umgesetzt werden.

Eine weitere XML-basierte Seitenauszeichnungssprache für Lehreinheiten stellt die Learning Material Markup Language (LMML) dar. Sie ist konzipiert als Seitenauszeichnungssprache für Lehreinheiten. Die strikte Trennung von Inhalt und dem äußeren Erscheinungsbild ermöglicht durch Extensible Stylesheet Language (XSL) eine spätere Transformation in verschiedene Ausgabeformate. Der strukturelle Aufbau eines LMML-Dokumentes wird durch seine erweiterbare DTD fixiert.

Das Lernmaterial in einem LMML-Dokument besteht aus einzelnen Einheiten, die selbst wieder andere Einheiten enthalten können. Die kleinsten Inhaltseinheiten sind sogenannte Inhaltsmodule (z.B. Definition oder Motivation). Sie können selbst wieder in Form von Aufzählungen oder Tabellen strukturiert sein und enthalten z. B. Texte, Bilder oder Animationen.

Der Ansatz den pädagogischen Überbau von EML und die stärker inhaltlich geprägte Ausrichtung von LMML zu kombinieren wirkt vielversprechend (Schnittstelle Knowledge-object in EML).

Abstract

The extensible markup language (XML) is a meta language that offers the possibility to define one's own languages with the help of different document type definitions (DTD).

Educational Modelling Language (EML) is one of these languages. It contains an extensible DTD on its own. The special aspect of EML is that not only learning material may be integrated but also roles, relations, interactions and activities of students and instructors. Thus various learning models can be realized (e.g. problem based learning, self study).

A further XML based markup language for units of study is the Learning material markup language. It is designed as a markup language for units of study. The strict separation of contents and appearance allows a future transformation into various output types on the basis of the extensible stylesheet language (XSL). The structure of an LMML document is defined by its extensible DTD.

The learning material of a LMML document consists of several units which may in turn contain further units. The smallest units are the so-called content modules (e.g. definition or motivation). These modules may have the structure of lists or tables and contain texts, images, animation etc.

The idea to combine the pedagogical possibilities of EML with the more content orientated aspect of LMML is worth to be elaborated in more detail (interface: knowledge-object of EML).

1. Extensible Markup Language (XML)

1.1. XML Überblick

Die Extensible Markup Language (XML) ist eine Untermenge des ISO-Standards Standard Generalized Markup Language SGML (ISO 8879, 1986). XML wurde 1996 vom W3C Konsortium verabschiedet und liegt derzeit in der Second Edition als Recommendation vom 6 Oktober 2000 vor (W3C 2001 a). Sie ist eine textbasierte Auszeichnungssprache, die die Beschreibung, den Austausch, die Darstellung und die Manipulation von strukturierten Daten erlaubt.

Im folgenden sind XML Charakteristika aufgeführt:

- W3C-Standard, firmen- und plattformunabhängig
- Datenaustauschformat (reiner Text)
- Metasprache (eigene Sprachen definierbar)
- Extensible (eigene Elemente)
- Markup Language (Markups $\langle \rangle$)
- Selbstbeschreibend („sprechende Elementnamen“)
- Strukturierte Daten (Baumstruktur, Wohlgeformtheit, Gültigkeit)
- Grundkonzept der Trennung des Inhalts (XML Datei) von dem Layout (CSS oder XSL Datei, siehe 1.3), Transformation (XSL Datei) und Grammatik (DTD oder Schema Datei, siehe 1.2)

XML ist eine Metasprache, die es erlaubt eigene Sprachen mit Hilfe einer Document Type Definition (DTD, siehe 1.2) oder Schema zu definieren. Diese Sprachen werden als XML Anwendungen bezeichnet. Extensible Hypertext Markup Language (XHTML), WML (Wireless Markup Language), EML (Educational Modelling Language) und LMML (Learning Material Markup Language) sind Beispiele dafür.

Partl (2002) schreibt: „Unter XML-Anwendung oder XML-Applikation versteht man die Festlegung (Normierung) von XML-Befehlen für eine Klasse von XML-Dokumenten gleicher Struktur, also für einen bestimmten Zweck. Das Format und die Struktur der XML-Files sowie die Eigenschaften und die Schachtelung der darin vorkommenden Elemente (XML-Befehle, Tags, Entities) werden für eine XML-Anwendung mit einer DTD oder einem Schema definiert ...“. Einen Einblick in die Vielfalt der vorhandenen XML Anwendungen bietet OASIS (2001).

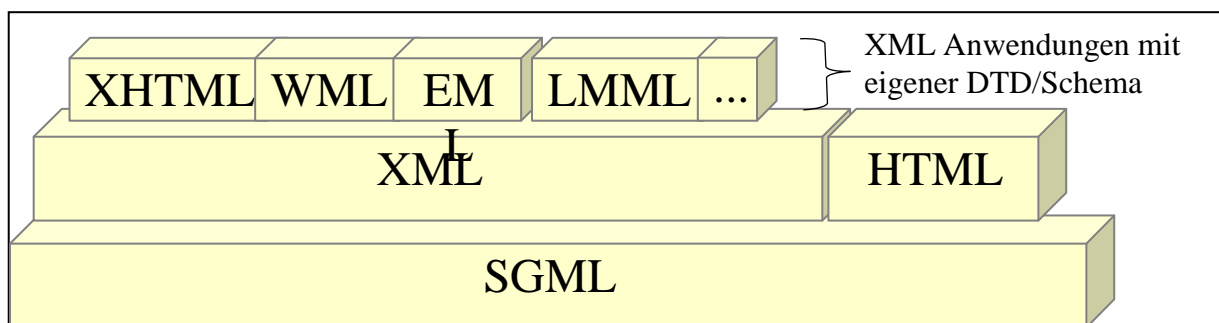


Abbildung 1: XML Anwendungen

Die Daten werden in XML in einer Baumstruktur abgelegt. XML Dokumente müssen dabei wohlgeformt sein (W3C 2001 a). Hält sich die Struktur einer XML Datei an die in einer DTD vorgegebenen Syntax, wird sie zusätzlich als gültig bezeichnet (W3C 2001 a). Abbildung 2 zeigt ein Beispiel einer wohlgeformten und gültigen XML Datei. Die Gültigkeit ist gegenüber der LMML-CS DTD geprüft.

```
<?xml version="1.0" encoding="iso-8859-1"?>
<!DOCTYPE lmml SYSTEM "LMML-CS.dtd">
<lmml>
  <section>
    <paragraph>
      <LMMLtext>
        LMML (Learning Material Markup Language)
      </LMMLtext>
    </paragraph>
  </section>
</lmml>
```

Abbildung 2: Beispiel einer wohlgeformten und gültigen XML Datei

1.2. XML Document Type Definition (DTD)

Eine DTD beschreibt den strukturellen Aufbau und die logischen Elemente einer XML Datei. Sie stellt eine formale Grammatik dar, in der die Bezeichnungen der in Dokumenteninstanzen zulässigen Elemente und deren Verschachtelung definiert sind. Unter Instanzen versteht man Dokumente, deren Elemente und Struktur den Vorschriften einer bestimmten DTD entsprechen.

Neben DTD's werden zunehmend XML Schemas (W3C 2002) verwandt, um XML Dokumente in ihrer Struktur und mit ihren Elementen zu beschreiben. Im Gegensatz zu DTD's (geschrieben in der Extended Backus Naur Syntax) sind XML Schemas XML Dateien und können damit von beliebiger XML Software be- und verarbeitet werden. Ein weiterer Vorteil eines Schemas besteht darin, daß eine Überprüfung auf gültige Datentypen und Wertebereiche möglich ist.

Folgende Abbildung 3 zeigt einen Ausschnitt aus einem Modul der LMML-CS (Learning Material Markup Language-Computer Science) DTD. Es zeigt die Stelle an der das Element lmml eingeführt wird (LMML siehe 3).

```
...
<!--===== document element =====>
<!ENTITY % LMML.content "%Structuremodule.class; %Contentobject.class;">
<!ELEMENT lmml (%LMML.content);>
<!ENTITY % LMML.version.attrib "version CDATA #FIXED '%LMML.version;'">
<!ATTLIST lmml
  %LMML.version.attrib;
>
<!-- end of LMML11-cmodel-10.mod -->
```

Abbildung 3: Ausschnitt aus einem Modul der LMML-CS DTD (SÜSS 2001).

1.3. Extensible Stylesheet Language (XSL)

Die derzeitige Extensible Stylesheet Language XSL (W3C 2001 c) zerfällt in drei Komponenten:

- XSL Transformation XSLT (W3C 2001 d). XSLT erlaubt die strukturelle Transformation von einem XML Format in andere Formate.
- XSL Formatting Objects (W3C 2001 c). XSL-FO ermöglicht die Formatierung mit Hilfe der formatting objects.
- XPath (W3C 2001 e). XPath bietet eine Syntax an auf bestimmte Knoten in einem XML Dokument zuzugreifen. XPath wird zusätzlich in der XML Linking (W3C 2001 b) Spezifikation eingesetzt.

XML Dateien enthalten nur logische Auszeichnungen. Eine Auszeichnung mit Elementen sagt nichts darüber aus, wie sie darzustellen sind. Die so bezeichneten Daten sind unabhängig vom Ausgabemedium (z. B. Bildschirm, Display, Lautsprecher, Drucker) und enthalten keinerlei Angaben zur Formatierung (z. B. Schriftart, Schriftgröße).

Für das Layout stehen derzeit hauptsächlich zwei Formatierungssprachen zur Verfügung: CSS und das weniger häufig eingesetzte XSL-FO. CSS wird zur Formatierung von HTML bereits eingesetzt. Sie kann aber auch zur Darstellung von Elementen einer XML Datei in einem Browser eingesetzt werden. XSL kann dagegen wesentlich mehr. Mit Hilfe von XSLT können XML-Daten in HTML, WML, SVG, usw. transformiert werden.

Folgende Abbildung zeigt den Ablauf einer Transformation und Formatierung eines XML Dokumentes mit XSL.

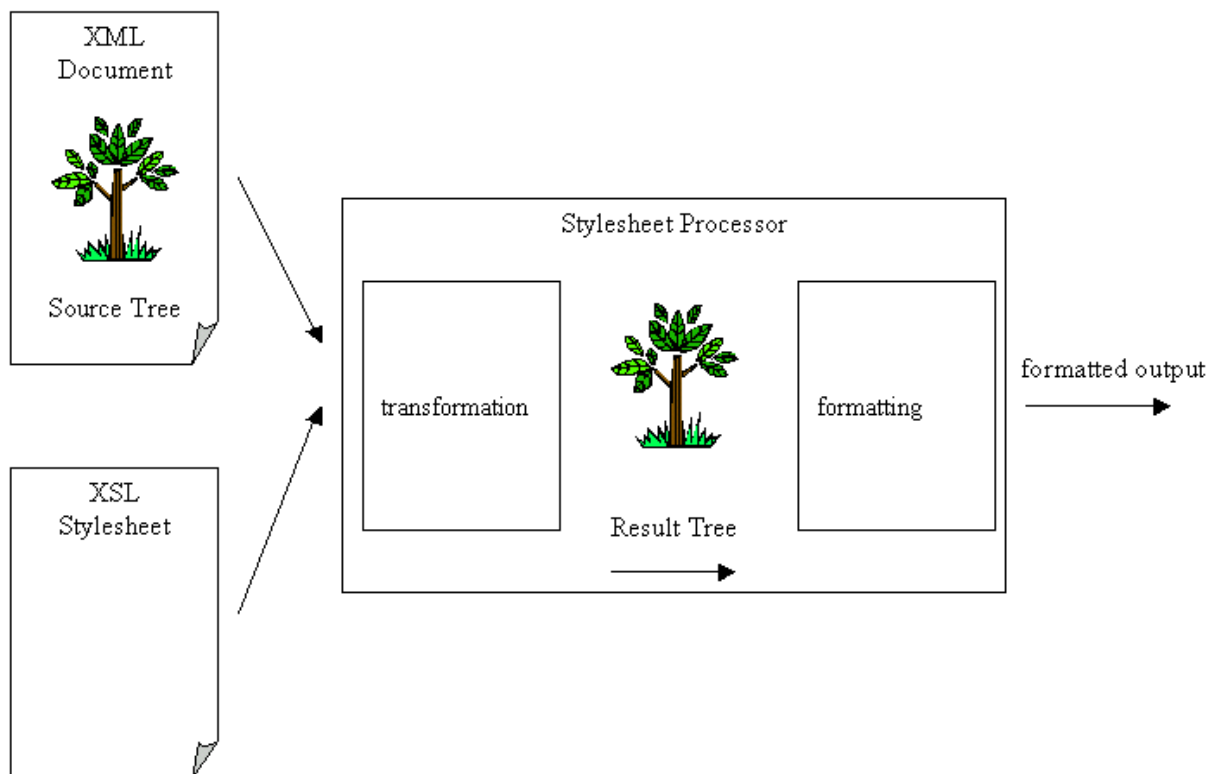


Abbildung 4: Ablauf einer Transformation und Formatierung eines XML Dokumentes

XSL besitzt im Gegensatz zu CSS keine eigene von XML abweichende Syntax. Sie ist selbst eine mit Hilfe von XML definierte Auszeichnungssprache. Eine Style-Sheet-Datei (XSL Datei) ist damit XML-konform. Anwendungen, die für XML entwickelt wurden, sind somit auch für XSL anwendbar.

Folgendes Beispiel zeigt einen Auszug aus einer XSL Datei. Sie wird von einer XML Datei aus aufgerufen und transformiert den XML Inhalt in HTML.

```
<?xml version="1.0" encoding="iso-8859-1"?>
<xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
xmlns="http://www.w3.org/TR/xhtml1/strict">
  <xsl:output method="html" indent="yes" encoding="iso-8859-1"/>
  <xsl:template match="/">
    <html>
      <head>
        <title>
          <xsl:value-of select="@title"/>
        </title>
      </head>
      <body>
        <xsl:apply-templates/>
      </body>
    </html>
  </xsl:template>
```

Abbildung 5: Ausschnitt einer XSL Datei zur Transformation von XML in HTML

2. Educational Modelling Language (EML)

2.1. EML Überblick

EML ist eine Entwicklung des Educational Technology Expertise Center der Open University of Netherlands (OUNL). Laut KOPER (2000) existiert kein vergleichbares Regelwerk (Metamodell), das es erlaubt Lerneinheiten in einer integrierten Art und Weise darzustellen.

Das besondere an EML liegt darin begründet, dass dabei nicht bloß Lerninhalte, sondern auch Rollen, Beziehungen, Interaktionen und Aktivitäten von Lernenden und Lehrenden integriert werden. Dabei können verschiedene Lernmodelle berücksichtigt werden (z.B. problemorientiertes lernen, Selbststudium, Frontalunterricht). Implementiert wird EML in der Extensible Markup Language (XML).

Aktuell liegt der CEN/ISSS (Comité Européen de Normalisation/Information Society Standardization System) ein Antrag vor EML standardisieren zu lassen.

2.2. unit of study model

Der Begründer von EML Prof. Koper (2000, S. 4) definiert EML wie folgt: „We called the notation of units of study an ‘Educational Modelling Language’“. Eine unit of study ist dabei „... the smallest unit providing learning events for learners, satisfying one or more interrelated learning objectives.“ (KOPER 2000, S. 3). Beispiele sind ein Studiengang, Kurs, Workshop, Praktikum oder Unterrichtseinheit.

In Abbildung 6 wird ein Ausschnitt aus der Struktur einer Lerneinheit mit ihren Lernobjekten gezeigt.

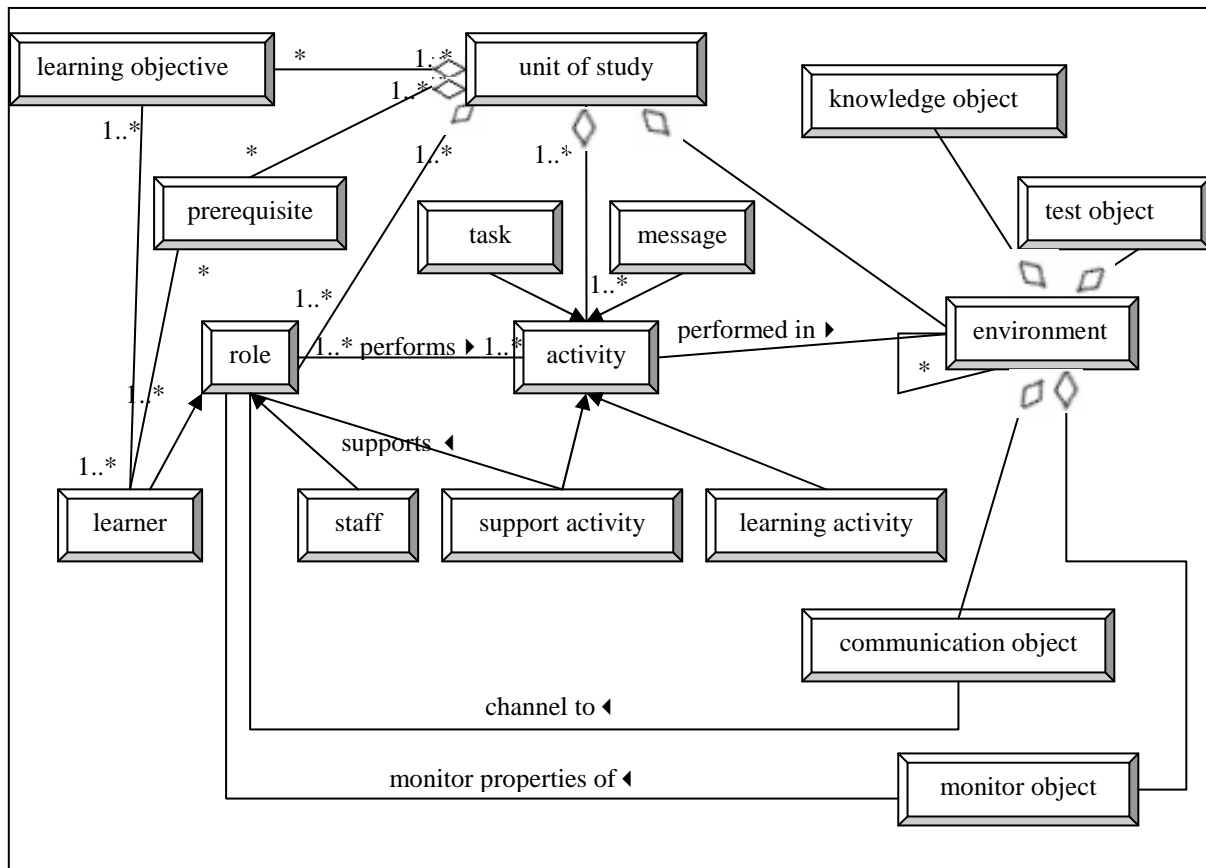


Abbildung 6: Auszug aus dem unit of study model (KOPER 2000, S. 11)

2.3. EML Beispiel

EML ist implementiert in XML. Die DTD der OUNL dient dabei als Grundstruktur für die Erstellung von Lerneinheiten. Es handelt sich damit bei EML um eine XML Anwendung.

Abbildung 7 zeigt einen Auszug aus der Darstellung der Basisstruktur des Rahmenwerks für Lernobjekte implementiert in der DTD von EML.

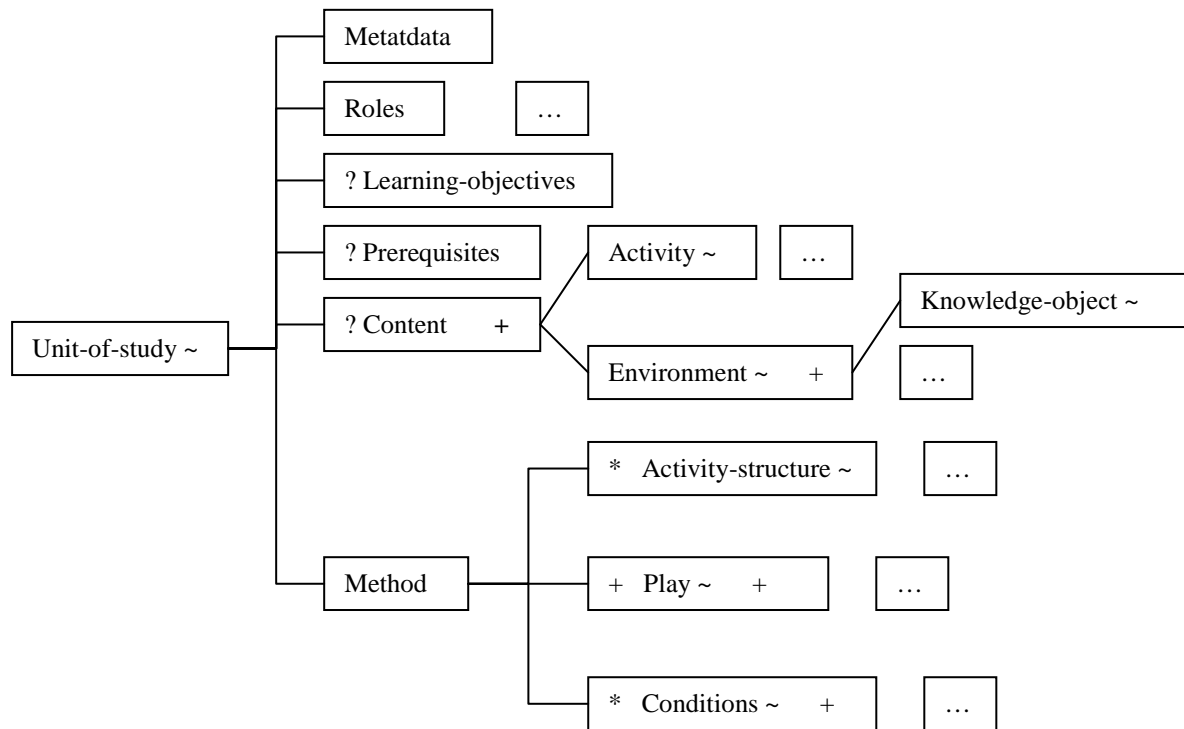
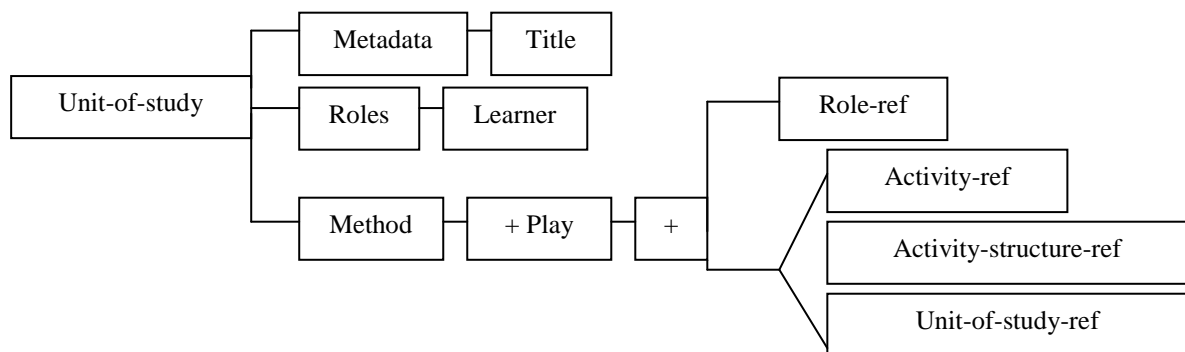


Abbildung 7: Einbindung von Lernobjekten in die DTD von EML (KOPER 2000, S. 18)

Die vollständige DTD ist im Internet unter <http://eml.ou.nl/> zu finden.

Abbildung 8 zeigt das kleinste gültige EML Modell einer unit of study und einen möglichen XML Quellcode (EML Datei).

Smallest valid EML model of a unit of study



Smallest valid EML instance of a unit of study

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

Abbildung 8: Kleinste EML Modell of a unit of study und ein mögliches Beispieldokument in XML (KOPER 2001, S. 22)

2.4. Erstellung von Lerneinheiten (unit of study)

Lerneinheiten werden arbeitsteilig in einem Team (optimale Lösung) mit folgenden Spezialisten erstellt (PLEGER 2001):

- Inhalts-Expertin
- Educational Designer (Pädagoge)
- EML-Spezialist
- MultiMedia-Produktions-Assistent

Inhalts-Experte und Educational Designer erarbeiten gemeinsam den Entwurf einer Lerneinheit. Nach SCHLUSMANS (2001) werden u.a. folgende inhaltliche Aspekte des didaktischen Entwurfs beschrieben:

- Lernziele
- Voraussetzungen
- Prüfungsplan
- Personalisierung/Differenzierung
- Rollen
- Lernaktivitäten
- Lernumgebung

Diese Aspekte werden in einem sogenannten didaktischen Szenario zusammengefasst. Das didaktische Szenario dient u.a. zur Aufteilung der weiteren Arbeiten und bietet eine Gesamtübersicht über die Lerneinheit. Der Inhalts-Experte erstellt die zentralen Fachinhalte, die Educational Designer arbeitet die Interaktionen zwischen den festgelegten Rollen aus.

Aus diesen Vorgaben erstellt der EML-Spezialist eine EML-Datei in XML auf der Grundlage der DTD von EML. In diese Datei werden von dem Multimedia-Produktions-Assistenten erstellte Multimedia-Elemente eingebunden.

Ausgehend von der EML-Datei im XML-Format ist es möglich z.B. mit Hilfe der Edubox (siehe unten) automatisiert die eigentlichen Lernmaterialien zu erstellen. Verschiedene Ausgabemedien sind dabei realisierbar. So kann aus einer EML-Datei z.B. die Web-Version und die Print Version der Lerneinheit erzeugt werden.

2.5. Edubox

Die OUNL entwickelte eine flexible, rechnergestützte, elektronische Lernumgebung auf der Grundlage von EML (Edubox).

Diese Edubox (OUNL 2002) ist die spezifische Implementierung einer EML/XML Verbindung, die auf einer Document Type Definition (DTD) beruht. Mit ihr ist ein Player programmiert worden, der EML Dateien interpretiert und eine konkrete Lernumgebung entwirft, indem er Lernobjekte (learning objects) mit Personen, Ressourcen, Daten und andere verschiedene Medien verbindet. Dazu wird die EML Datei im XML Format an die Edubox übergeben.

Sie kann verschiedene Releases aus einer Lehreinheit erzeugen. Releases heben dabei auf das Herstellungsdatum ab. Somit kann sichergestellt werden, dass ein Student des Jahres 1999 auch wirklich über den zu diesem Zeitpunkt gelehrt Stoff geprüft wird. Innerhalb eines Releases gibt es in der Edubox weitere Publikationsvarianten, die ein bestimmtes Layout, eine bestimmte Sprache oder Ausgabemedium bedienen können.

Die erste Edubox außerhalb der Niederlande befindet sich nach Herrn Pleger (2001) in Innsbruck. Die weitere softwaretechnische Entwicklung wird von der Firma Perot Systems übernommen. Ca. gegen Ende 2002 soll ein lauffähiges System für Lehranstalten zur Verfügung stehen.

3. Learning Material Markup Language (LMML)

3.1. LMML Überblick

Nach SÜSS (2000) ist LMML ein XML basiertes Modell für Lerninhalte und kann als eine Familie von Auszeichnungssprachen bezeichnet werden. Es wurde im Rahmen des Forschungsprojekts PaKMaS (Passauer Knowledge Management System) an der Universität Passau entwickelt (SÜSS & FREITAG 2002). Eine Instanz des LMML-Framework stellt die LMML-CS (Computer Science) mit ihrer eigenen DTD dar. Sie dient zur Beschreibung von Lehrinhalten im Fachgebiet Informatik. Durch die Schaffung weiterer spezifischer DTD's, die in das Framework eingebunden werden, können neue LMML Instanzen für verschiedene Fachgebiete geschaffen werden. Das LMML Framework mit der LMML-CS DTD ist im Internet unter <http://www.lmml.de/> zu finden.

3.2. LMML Modell

Lerninhalte können in Module gefasst werden (SÜSS 2000), die wiederum Module in sich aufnehmen können. Diese Bausteine werden in LMML umgesetzt. Die für das Fachgebiet spezifische DTD (z. B. LMML-CS) legt dabei die Struktur der LMML Dokumente fest. Strukturmodule (Structure Module) in LMML ermöglichen - ähnlich der Kapitelangaben in einem Buch - die Strukturierung von Inhaltsmodulen (ContentModules). Die Inhaltsmodule setzen sich zusammen aus z.B. Motivation, Example, Definition. Informatik-spezifisch sind aber auch z. B. Formel, Algorithmus möglich. Die Inhaltsmodule können selbst wiederum in Listen oder Tabellen strukturiert sein und beinhalten letztlich die Medienobjekte (MediaObjects), wie Text, Ton, Bild und Animation (siehe Abbildung 9).

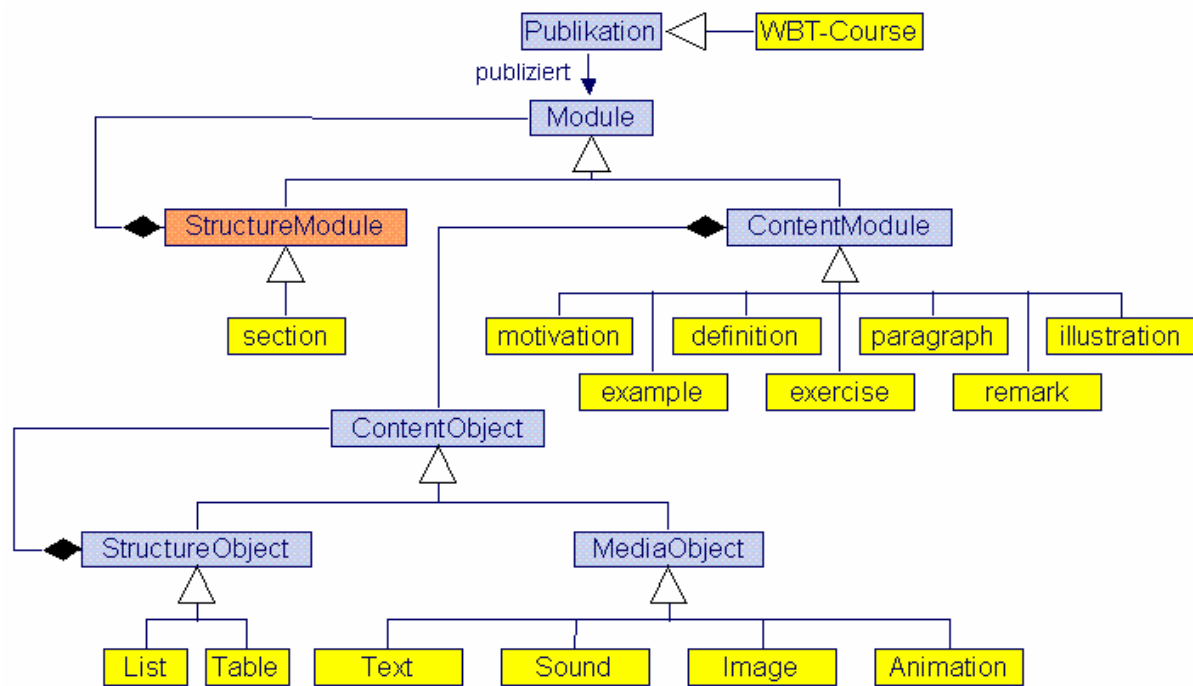


Abbildung 9: Modulare Struktur von Lehrinhalten (SÜSS & FREITAG 2001 b)

Ein Fachgebiet kann damit in Module beliebiger Granularität unterteilt werden. Jeder einzelne Baustein, von Strukturmodulen (z. B. ganze Kurse oder Kursabschnitte) über einzelne Inhaltsmodule bis hinunter zu den Medienobjekten, kann dabei in anderen Lerneinheiten wiederverwandt werden.

3.3. Passauer Knowledge Management System (PaKMaS)

PaKMaS ist nach SÜSS & FREITAG (2001 b): „... ein plattformunabhängiges, modulares Softwaresystem, das die Vorzüge von datenbank-basiertem XML-Contentmanagement und adaptiven Hypermedia-Systemen vereint.“ Es kann als Autoren- und Lernumgebung eingesetzt werden. PaKMaS ist offen für die unterschiedlichsten Formate des vorliegenden Lehrmaterials. So können z. B. HTML-Seiten, PDF-Dokumente, MS Word-Dokumente oder Microsoft Powerpoint-Präsentationen eingebunden werden. Als Ausgabeformate sind gedruckte Skripte im PDF- oder HTML-Format oder ein personalisierbarer Kurs in PaKMaS möglich. Die Möglichkeiten der Personalisierung werden dabei durch PaKMaS fest vorgegeben.

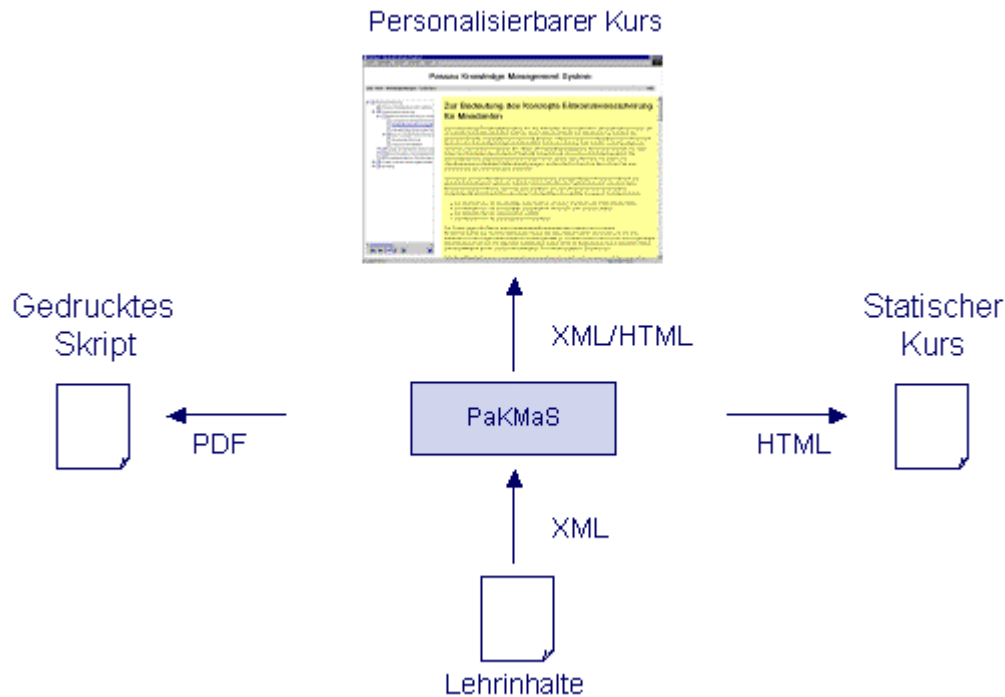


Abbildung 10: PaKMas (SÜSS & FREITAG 2001 b)

Um die Personalisierung und mehrdimensionale Anfragen geeignet unterstützen zu können, wurde laut SÜSS & FREITAG (2001 b) für PaKMaS eine relationale Speicherung von XML-Dokumenten implementiert. Damit ist es möglich XML-basierte LMML Lerninhalte in einer relationalen Datenbank (RDBMS) zu speichern. Lernende können damit als Ergebnis in einer auf sie zugeschnittenen Lernumgebung die Auswahl von z. B. Inhalten, Beispielen, Übungen vornehmen.

Die Autoren erstellen die Lerninhalte mit einem Standard XML Editor. Im LAMP (Lehrverbund Augsburg München Passau) Projekt wurden dazu nach SÜSS & FREITAG (2001 b) XML-basierte Publikationsdokumente benutzt, die den Inhalt, das Layout, die Navigation, die Personalisierung und die möglichen Anfragen der Lernenden festlegen.

3.4. LMML Beispiel

Die folgende Abbildung stellt den XML Quellcode einer LMML Datei auszugsweise dar. Als DTD wurde die LMML-CS.dtd genutzt. Zur einfachen Darstellung kann (wie bei EML) ein XML-fähiger Browser mit XSL-Unterstützung dienen (z. B. MS Internet Explorer 6 mit MS XML Parser 3). Mit Hilfe einer XSL-Datei wird die Transformation in eine HTML-Datei durchgeführt. Soll die XSL-Transformation serverseitig durchgeführt werden, übernehmen Softwareprodukte wie z. B. Apache Cocoon (APACHE COCOON 2001) oder andere XML-Server diese Aufgabe. Dies hat den Vorteil, daß der Client-Rechner mit reinem HTML bedient werden kann. Er muss nicht in der Lage sein, XML/XSL-Dateien zu verarbeiten.

```
<?xml version="1.0" encoding="iso-8859-1"?>
<?xml-stylesheet type="text/xsl" href="layout.xsl"?>
<!DOCTYPE lmml SYSTEM "LMML-CS.dtd">
<lmml>
  <section title="Der Schutzwald in den Bayerischen Alpen">
    <section title="Vorwort">
```

```

    <paragraph>
      <LMMLtext>
        Die bayerischen
        ...
      </LMMLtext>
    </paragraph>
  </section>
  <section title="Der Naturraum Bayerische Alpen">
    <section title="Landschaft">
      <paragraph>
        <LMMLtext>
          ...

```

Abbildung 11: Auszug aus dem XML Quellcode einer LMML Datei

4. Diskussion und Ausblick

Vielversprechend wirkt der Ansatz beide XML basierte Auszeichnungssprachen für Lerneinheiten zu kombinieren. Der pädagogische Überbau wird dabei von EML gestellt. LMML wird zur Aufbereitung der Lerninhalte eingesetzt. Als Schnittstelle kann das Knowledge-object in EML dienen.

Sinnvoll erscheint die Entwicklung einer eigenen EML/LMML Umgebung, die es den Autoren von Lerninhalten erlaubt vereinfacht EML/LMML Dateien im XML Format zu erstellen. Der XML Editor XML Spy bietet mit dem Document Editor dazu einen Ansatz.

Die Umsetzung der EML Dateien in die eigentliche Lerneinheit wird derzeit bereits von der Software Edubox realisiert. Die Firma Perot Systems will Ende 2002 eine Edubox der Version 3 vorstellen. Inwieweit eine Einbindung von LMML Inhalten in Form von XML Dateien möglich ist, bleibt abzuwarten.

Des weiteren ist auch ein Einbinden von EML Elementen in PaKMaS vorstellbar.

Literaturverzeichnis

ANDERSON, R., BIRBECK, M., DIDIER, M., KAY, M., LIVINGSTONE, S., LOESGEN, B., MOHR, S., OZU, N., PEAT, B., PINNOCK, J., STARK, P., WILLIAMS, K. (2000): XML professionell (1. Auflage). Bonn: MITP-Verlag. 957 S.

APACHE COCOON (2001): The Apache XML Projekt: Apache Cocoon, <http://xml.apache.org/cocoon/>, Abruf am 2002-03-20

HERMANS, H, KOPER, R., LOEFFEN, A., MANDERFELD, J., RUSMAN, M. (2000): Edubox-EML Reference Manual (Beta Version). Educational Technology Expertise Center, Open University of Netherlands, <http://eml.ou.nl/eml/>, Abruf am 2001-12-09

ISO 8879: ISO (International Organization for Standardization). ISO 8879:1986(E). Information processing -- Text and Office Systems -- Standard Generalized Markup Language (SGML). First edition -- 1986-10-15. [Geneva]: International Organization for Standardization, 1986.

KOPER, R. (2000): From change to renewal: Educational technology foundations of electronic learning Environments, Educational Technology Expertise Center, Open University of Netherlands 2000, <http://eml.ou.nl/introduction/articles.htm> , Abruf am 2001-11-21

KOPER, R. (2001): Modeling units of study from a pedagogical perspective, Open University of Netherlands, <http://eml.ou.nl/introduction/articles.htm>, Abruf am 20.01.2002.

OASIS (2001): Oasis and Robin Cover: The SGML/XML Web Page. SGML and XML News. <http://www.oasis-open.org/cover/sgmlnew.html>, Abruf am 2001-12-09

OUNL (2001): Open Universiteit Nederland: Educational Modelling Language, <http://eml.ou.nl/>, Abruf am 2001-11-21.

OUNL (2002): Open Universiteit Nederland: Educational Modelling Language, Edubox <http://eml.ou.nl/forum/sept/item6.htm>, Abruf am 2002-01-21.

PARTL (2002): Universität für Bodenkultur: Zentraler Informatik Dienst ZID, <http://www.boku.ac.at/html/inf/xmlkurz.html#applikationen> , Abruf am 2002-01-21.

PLEGER, G (2001): mündliche Auskunft, Treffen vom 11.10.2001, Medienzentrum des Landes Tirol, Innsbruck.

SCHLUSMANS, K. (2001): Het inhoudelijk-didactisch ontwerp von onderwijs in EML. Heerlen: OUNL.

SÜSS, C. (2000): Adaptive Knowledge Management: A Meta-Modelling Approach and its Binding to XML. In: H.-J. Klein (Ed.), 12.GI-Workshop Grundlagen von Datenbanken, Plön, TR 2005, Christian-Albrechts-Universität Kiel, Germany, 2000

SÜSS, C. (2001): Document Type Definitions (DTDs). <http://daisy.fmi.uni-passau.de/pakmas/LM2L/> , Abruf am 2001-11-21.

SÜSS, C., FREITAG, B. (2001 a): Learning Material Markup Language –LMML. IFIS-Report 2001/03, IFIS - Institut für Informationssysteme und Softwaretechnik. Universität Passau 2001

SÜSS, C., FREITAG, B. (2001 b): Das LAMP Projekt: Lehrverbund Augsburg München Passau, http://daisy.fmi.uni-passau.de/lamp/tp2_werkzeuge.html , Abruf am 2002-03-02.

SÜSS, C., FREITAG, B. (2002): Datenbanken und Informationssysteme: Passauer Knowledge Management System (PaKMaS), <http://daisy.fmi.uni-passau.de/pakmas/pakmas.html>, Abruf am 2002-03-20.

IFIS (2002): Institut für Informationssysteme und Softwaretechnik Universität Passau: <http://www.lmml.de/>, Abruf am 2002-03-20.

W3C (2001 a): World Wide Web Consortium: Extensible Markup Language (XML) 1.0 (Second Edition). <http://www.w3.org/TR/2000/WD-xml-2e-2000-08-14> , 2000-08-14, Abruf am 2001-10-03.

W3C (2001 b): World Wide Web Consortium: XML Linking Language (XLink), Version 1.0. <http://www.w3.org/TR/2000/REC-xlink-20010627/> , 2001-06-27. Abruf am 2001-12-19.

W3C (2001 c): World Wide Web Consortium: Extensible Stylesheet Language (XSL), Version 1.0. <http://www.w3.org/TR/xsl/> , 2001-10-15, Abruf am 2002-03-20.

W3C (2001 d): World Wide Web Consortium: XSL Transformations (XSLT), Version 1.0. <http://www.w3.org/TR/1999/REC-xslt-19991116.html> , 1999-11-16, Abruf am 2002-02-11.

W3C (2001 e): World Wide Web Consortium: XML Path Language (XPath) Version 1.0. <http://www.w3.org/TR/xpath> , W3C Recommendation, 1999-11-16, Abruf am 2002-02-11.

W3C (2002): World Wide Web Consortium: XML Schema W3C Recommendation, 2 May 2001, <http://www.w3.org/XML/Schema>, Abruf am 2002-03-20

YOUNG, J. (2000): XML Schritt für Schritt. Unterschleißheim: Microsoft Press Deutschland