

# Die Dokumentenbeschreibungssprache der Wissenswerkstatt Rechensysteme

Ulrike Lucke, Djamshid Tavangarian, Denny Voigt  
Universität Rostock, Fachbereich Informatik, Lehrstuhl für Rechnerarchitektur  
ul | tav | dvoigt @informatik.uni-rostock.de

**Abstract:** Der Beitrag motiviert anhand der Defizite aktueller Entwicklungen zum Einsatz von XML in der multimedialen Lehre das innovative Konzept des Projekts „Wissenswerkstatt Rechensysteme“. Schwerpunkt der Darstellung ist die entwickelte Beschreibungssprache für Lehr- und Lernmodule, die eine Skalierung des Materials in drei Dimensionen erlaubt, multimediale und interaktive Elemente beinhaltet sowie eine sachlogischen und didaktische Strukturierung der Module bietet. Ausgewählte Beispiele erläutern interessante Aspekte der Sprachdefinition.

## 1 Einleitung

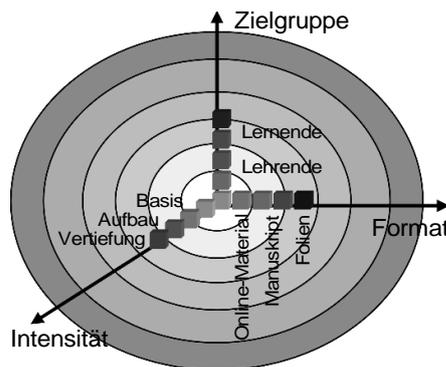
Die Extensible Markup Language (XML) hat neben ihrem Einsatz als plattformunabhängiges Format zum Datenaustausch oder für dynamisch veränderbare Web-Inhalte zunehmend auch in der multimedialen Lehre an Bedeutung gewonnen. Zahlreiche Entwicklungen bezeugen, dass die durch die mit XML verbundene Trennung von Struktur, Präsentation und eigentlichem Inhalt von Daten erreichte Flexibilität insbesondere auch zur Gestaltung von anspruchsvollen Lehr- und Lernumgebungen genutzt werden kann [1]. Die Arbeiten erstrecken sich beispielsweise auf die Beschreibung von Metadaten, von formalen (z. B. didaktischen) Sachverhalten oder von Lehrmaterial selbst. Allerdings erfolgt noch keine adäquate Unterstützung durch entsprechende Werkzeuge. Es stehen vor allem reine XML-Editoren ohne Fokussierung auf bestimmte Datenformate oder Einsatzgebiete zur Verfügung. Ein weiteres Problem stellt die auch durch einen Wechsel auf das Datenformat XML nicht gestiegene Wiederverwendung bereits vorhandener Materialien dar. Dies liegt zumeist an ihrer fehlenden Anpaßbarkeit an den jeweiligen Kontext durch den Dozenten (weniger in thematischer Hinsicht, sondern vor allem bzgl. des Umfangs und der äußeren Gestaltung). Diesen Nachteilen soll u.a. im Rahmen des Vorhabens „Wissenswerkstatt Rechensysteme“ begegnet werden. Ein in mehreren Dimensionen skalierbares Dokumentenformat erlaubt die freie Anpassung eines Moduls an konkrete Anforderungen und Lernziele von Lehrenden und Lernenden.

## 2 Konzept der Wissenswerkstatt Rechensysteme

Wesentliches Ziel des Verbundvorhabens „Wissenswerkstatt Rechensysteme“ (WWR) ist die Erstellung von multimedialen Lehr- und Lernmodulen im Fachgebiet Technische Informatik [2]. Grundlegende Innovationen sind die Mehrdimensionalität der Module, die zusammen mit der adaptiven, zeitorientierten Granularität zu einer flexiblen Wiederverwendung durch den Dozenten führt, sowie die konsequente XML-Nutzung bei der Modulimplementierung.

Das Lehrmaterial besteht aus Vorlesungsstoff, praktischen Beispielen, Übungsaufgaben und Lösungen, Literaturverweisen sowie parallel zur Verfügung stehenden Praktikumsumgebungen, die zu fachlich und didaktisch sinnvollen Einheiten zusammengefaßt werden. Der umzusetzende Lehrstoff wird in Module von vergleichbarer Größe unterteilt, die jeweils Material für ca. vier Wochen Lehrveranstaltung und Übung bzw. Praktikum enthalten. Jedes Modul ist skalierbar bzgl. folgender Eigenschaften:

- Intensität bzw. Umfang  
Aus einer abstrakten Modulbeschreibung können Basis-, Aufbau- oder Vertiefungsmodule erzeugt werden.
- Zielgruppe  
Ein Modul enthält spezielle Elemente für Lernende und Lehrende.
- Ausgabeformat  
Ein Modul kann online, als Foliensatz oder druckbares Manuskript publiziert werden.



Es ergibt sich ein dreidimensionales Modell des zu implementierenden Inhalts, das in Abbildung 1 dargestellt ist. Dies führt zu insgesamt 18 potentiell verschiedenen Ausgabevarianten jedes Moduls. Um aus der abstrakten XML-Beschreibung eine konkrete Variante zu erzeugen, muß für jede Dimension der jeweilige Wert spezifiziert werden.

**Abbildung 1.** Dimensionen der WWR-Modul

Um den Austausch mit herkömmlichen Datenformaten (insbesondere bei der Modulerstellung) zu erleichtern, wurden Importfilter für die Formate Word, Powerpoint, LaTeX und FrameMaker erstellt. Die hierbei gesammelten Erfahrungen sind für die Gestaltung entsprechender Exportfilter nutzbar.

Aus den Modulen können Lehrende individuelle Kurse zusammensetzen. Ein Kurs ist definiert als Liste von Modulen in einer bestimmten Intensität (Basis, Aufbau oder Vertiefung). Aus der Kursbeschreibung werden verschiedene Dokumentarten erzeugt.

Darüber hinaus können aus den Modul- und Kursbeschreibungen ergänzende Informationssammlungen generiert werden, wie z. B. Inhaltsverzeichnis, Stichwortverzeichnis, Glossar, Abkürzungsverzeichnis oder eine Liste der zitierten Personen. Dazu werden entsprechend markierte Elemente der betreffenden Module ausgewertet und in den jeweiligen Index sortiert. Da der Dozent eines Kurses die thematisch passenden Module anhand der ihm zur Verfügung stehenden Zeit in Umfang bzw. Intensität anpassen kann, wird hier eine größtmögliche Flexibilität erreicht, die ein starrer Ansatz mit themenorientiertem Modulumfang nicht bieten kann.

### 3 Die WWR-Beschreibungssprache

Zunächst wurden existierende XML-Sprachen für die multimediale Lehre untersucht. Hierzu zählten u. a. das Universal Learning Format (ULF) [4], das Sharable Content Object Reference Model (SCORM) [5], die Educational Modelling Language (EML) [6] und die Learning Material Markup Language (LMML) [7]. Keine der Sprachen war in der

Lage, die im Konzept der WWR spezifizierten Dimensionen umzusetzen. Daher wurden die Basiskonzepte der Sprachen aufgegriffen, in ein innovatives Format integriert und um die Charakteristika der WWR erweitert. Die daraus resultierende Sprachdefinition wurde in einem Satz von DTDs bzw. Schemata beschrieben und ist theoretisch (unter Verlust der WWR-spezifischen Anteile) wieder in die Ausgangsformate transformierbar.

### 3.1 Sachlogische Strukturierung

Der Inhalt eines Moduls wird durch den Autor einer sachlogischen Strukturierung unterzogen. Diese muß den Anforderungen eines druckbaren Manuskripts sowie einer elektronischen Präsentation genügen. Daher sind sowohl Kapitel und Abschnitte als auch (Bildschirm-) Seiten zu definieren. Ein Modul besteht demnach aus beliebig ineinander verschachtelten Gliederungsebenen, die ihrerseits einzelne Präsentationseinheiten beinhalten. Diese werden aus Inhaltsblöcken gebildet, die aus didaktischer Sicht als Algorithmus, Aufgabe und Lösung, Beispiel, Bemerkung, beschreibender Text, Definition, Hinweis, Satz und Beweis oder Zitat ausgezeichnet werden und auf unterster Ebene aus einfachem Text, Multimedia-Komponenten o. ä. bestehen. Abbildung 2 veranschaulicht diese Struktur. Diese Vorgehensweise entspricht einer Kombination der aus anderen Sprachkonzepten bekannten Definition von Gliederungsebenen und Präsentationsseiten, die hier kombiniert werden und eine wichtige Grundlage der in Abschnitt 3.7 beschriebenen didaktischen Struktur darstellen.

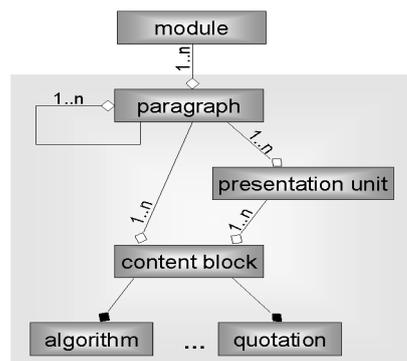


Abbildung 2. Sachlogische Strukturierung eines Moduls.

### 3.2 Multimediale Komponenten

Die WWR-Dokumentensprache erlaubt die Integration verschiedenster Multimedia-Objekte, angefangen von Vektor- und Rastergraphiken (z.B. SVG, GIF, JPG, BMP) über Sound (z.B. MP3, WAV) und Video (AVI, MPG) bis hin zu interaktiv steuerbaren Animationen (wieder SVG, Flash), 3d-Szenarios (VRML) und Applets. Dieser impliziten Klassifizierung der Multimedia-Objekte entspricht auch deren Umsetzung in der Dokumentensprache durch die Elemente `image`, `sound`, `video`, `animation` und `scene3d`. Diese Elemente sind von Grund auf gleich aufgebaut, kennzeichnen jedoch jeweils genau eine der vorgestellten Klassen. Sie alle besitzen die obligatorischen Attribute `uri` und `type`, von denen das Erstere Ort und Name der Ressourcendatei festlegt und durch das Zweite der Medientyp bekanntgemacht wird, wie z.B.:

```
<image uri="beispiel1.gif" type="gif"/>
```

Analog dazu erfolgt das Markup der anderen Multimedia-Objekte innerhalb der zu erstellenden Lehr- und Lernmodule. Der Vorteil dieser Art der Auszeichnung liegt darin, dass die einzelnen Multimedia-Elemente einfach um zusätzliche Typen ergänzt werden

können, indem der zulässige Wertebereich des `type`-Attributes entsprechend erweitert wird. Eine derartige Anpassung kann z.B. erforderlich sein, wenn sich neue Multimedia-Formate etablieren bzw. die benötigten Plug-Ins für die Darstellung in Browsern verfügbar sind.

Die bereits angesprochene Skalierbarkeit des Lernstoffs erfordert in bestimmten Fällen die Verwendung alternativer Informationen anstelle der standardmäßig zu präsentierenden Multimedia-Objekte. Das trifft z.B. dann zu, wenn ein Video in der Skriptversion eines Moduls erscheinen soll. Zu diesem Zweck besteht in der WWR-Dokumentensprache die Möglichkeit, zu einem Multimedia-Objekt eine textuelle Beschreibung mitsamt einer alternativen repräsentativen Bildsequenz zu definieren, die immer dann, wenn die Darstellung des originalen Objekts nicht möglich sein sollte, an dessen Stelle angezeigt wird. Hiermit kann einem eventuellen Informationsverlust während der Ausgabetransformation vorgebeugt werden. Das nachfolgende Beispiel verdeutlicht diese Vorgehensweise anhand einer Videoeinbindung mit textueller Beschreibung und Bildfolge:

```
<video uri="videos/experiment01.avi" type="avi">
  <description>
    <text>Dieses Video zeigt ... </text>
  </description>
  <imageseq>
    <image uri="images/experiment01a.jpg" type="jpg"/>
    <image uri="images/experiment01b.jpg" type="jpg"/>
    <image uri="images/experiment01c.jpg" type="jpg"/>
  </imageseq>
</video>
```

Zu den mächtigsten Typen der in der Wissenswerkstatt unterstützten Multimedia-Objekte zählen zweifelsohne neben den in beliebiger Form integrierbaren Applets die interaktiv steuerbaren Animationen, die sich z.B. mittels Flash oder durch geskriptete SVG-Graphiken (selbst wieder ein XML-Format!) realisieren lassen. Mit ihnen ist den Entwicklern das erforderliche Werkzeug und die notwendige Freiheit zur Realisierung abwechslungsreicher sowie multimedial anspruchsvoller Lehr- und Lerninhalte gegeben.

Weiterhin wurde eine Praktiums- und Simulationsumgebung in Form eines Frameworks entwickelt, um von einem Modul aus, auf extern ablaufende Experimente und Simulationen, die z.B. spezielle Hardwareanforderungen benötigen oder aus anderen Gründen nicht überall verfügbar sind, zugreifen zu können [8].

Trotz aller Möglichkeiten, die die WWR-Dokumentensprache zur Integration von Multimedia-Objekten bereitstellt, sollte stets deren didaktischer Sinn im Vordergrund stehen und eine multimediale Überladung vermieden werden [9].

### 3.3 Übungsaufgaben

Übungsaufgaben sind ein wichtiges Hilfsmittel für den Lernenden, um den vermittelten Lernstoff aufzuarbeiten und zu vertiefen, und spielen daher in der Wissenswerkstatt, wie in jeder anderen Lehr- und Lernumgebung auch, eine wesentliche Rolle. Die WWR-Dokumentensprache bildet deshalb die grundlegendsten Aufgabetypen [10] auf die nachfolgenden Elemente ab:

`multipleChoice` Realisierung einer  $m$ -aus- $n$ -Auswahl ( $0 < m < n$ ).

yesNo	Frage, die entweder mit „ja“ oder mit „nein“ zu beantworten ist.
fillBlanc	Ein Lückentext, dessen freie Felder beliebige Zeichenketten oder Zahlen beinhalten können.
essay	Frei zu formulierender Text zur Beantwortung einer Frage.
dragAndDrop	Zuordnungsproblem, d.h., bei zwei gleich großen Mengen von Objekten müssen die korrespondierenden Objekte beider Mengen richtig erkannt werden.
imageChoice	Analog zu multipleChoice, nur muß in diesem Fall innerhalb einer Grafik eine gewisse Auswahl getroffen werden.

In Ergänzung zu diesen vordefinierten Typen ist es dem Autor gestattet eigene spezielle Aufgabentypen mit Hilfe von Applets oder interaktiv steuerbaren Animationen bereitzustellen.

Gemein haben alle hier betrachteten Elemente die Unterelemente `question`, `consult` und `hint` zur Beschreibung der Fragestellung sowie zur optionalen Angabe von korrespondierenden Inhaltsabschnitten und Lösungshinweisen. Des Weiteren ist es möglich, einer Fragestellung ein gewisses Antwortverhalten des Systems zuzuordnen. Dieses Feature muß, sofern es verwendet wird, an verschiedenen, vom Aufgabentyp abhängigen Stellen in die Aufgabenbeschreibung eingebunden werden. Der Aufbau einer solchen *Response* auf die Beantwortung einer Frage ist jedoch stets derselbe: Die Reaktion auf die richtige Beantwortung der Frage wird im `right`-Element festgelegt, während das `wrong`-Element die Antwort des Systems für den Fall einer fehlerhaften Beantwortung beinhaltet. Beide Elemente, `right` als auch `wrong`, kapseln ausschließlich einfache Inhaltselemente, wie Texte, Formeln oder Multimedia-Objekte, und können entweder einzeln oder zusammen angegeben werden. Anhand der Implementierung einer einfachen Mehrfach-Auswahl ohne Antwortverhalten soll die Auszeichnung von Fragestellungen innerhalb der WWR demonstriert werden:

```
<multipleChoice>
  <question>
    <text>Welche Übertragungsraten bietet Ethernet?</text>
  </question>
  <hint>
    <text>Beachten sie, dass ...</text>
  </hint>
  <consult ref="lessonX.xml#PAR0001">
    <text>Hier finden sie ...</text>
  </consult>
  <choice right="no"><text>1 MByte/s</text></choice>
  <choice right="yes"><text>10 MBit/s</text></choice>
  <choice right="no"><text>155 MBit/s</text></choice>
  <choice right="no"><text>20 MByte/s</text></choice>
  <choice right="yes"><text>1 GBit/s</text></choice>
</multipleChoice>
```

Die einzelnen `choice`-Elemente kapseln hierbei die anzuzeigenden Auswahlmöglichkeiten. Ob eine Auswahlmöglichkeit Teil der Lösungsmenge ist, wird durch das Attribut `right` angezeigt.

Die in XML implementierten Aufgaben können per Stylesheet-Transformation in eine Vielzahl von Ausgabeformaten gebracht werden. Zur Zeit laufen Arbeiten, die Übungsaufgaben für die Online-Präsentation neben HTML auch nach SVG umzusetzen, sowie die Konzeption einer Architektur für die serverseitige Auswertung.

### 3.4 Skalierbarkeit der Zielgruppe

Zur Eingrenzung des Geltungsbereichs auf eine Zielgruppe werden in der WWR-Dokumentensprache die zu skalierenden Elemente durch das Attribut `target` erweitert, das entweder die Werte `student`, `teacher` oder `inherit` annehmen darf. `inherit` ist dabei die Standardbelegung und bedeutet, dass die Skalierung vom übergeordneten Element, dem Väterelement, übernommen wird. `student` und `teacher` bedeuten hingegen eine Eingrenzung auf die Zielgruppe Lernender bzw. Lehrender. Somit darf die Skalierung in Richtung feingranularer Inhaltselemente bzw. niedriger Hierarchiestufen des Moduls ausschließlich eingeschränkt und nicht wieder erweitert werden (Restriktionsprinzip). Folgender, einen Hinweis beschreibender Inhaltsblock ist z.B. nur für Lehrende angedacht:

```
<remark target="teacher">...</remark>
```

Alle Unterelemente sind damit implizit ebenfalls auf die Zielgruppe `teacher` festgelegt. Eine Verletzung des Restriktionsprinzips, wie etwa

```
<remark target="teacher">
  <text>Sie vermitteln den Stoff am Besten, indem ... </text>
  <image target="student" uri="skizze.jpg" type="jpg"/>
  ...
</remark>
```

würde während der Validierung durch einen speziell auf die WWR-Dokumentensprache zugeschnittenen Parser als Fehler erkannt werden.

Die Eingrenzung bestimmter Inhaltsabschnitte auf bestimmte Zielgruppen ist insbesondere für Komponenten, wie Musterlösungen, Prüfungsfragen, Randbemerkungen, Selbsttests und Praktikas sinnvoll.

### 3.5 Skalierbarkeit der Intensität

Die Skalierung der Intensität stellt eine wesentliche Grundlage für die flexible Nutzung des Moduls in beliebigem Kontext dar. Die prinzipielle Vorgehensweise ist hier analog zu der in Abschnitt 3.4 beschriebenen: Die zu skalierenden Elemente werden durch das Attribut `intensity` erweitert, das die Werte `basic`, `advanced` oder `expert` bzw. eine Kombination Zweier annehmen darf. Auch hier gilt wieder das Restriktionsprinzip und die Vererbung des Attributwertes vom Vater- an ein Sohnelement, wenn des Letzteren `intensity`-Attribut den Wert `inherit` aufweist oder überhaupt nicht vorhanden ist.

Das nachfolgende Beispiel zeigt die Skalierung der Intensität innerhalb eines Inhaltsblocks, der einige Ausführungen zum Thema *Ethernet* macht:

```

<description title="Ethernet">
  <text>Grundlagen...</text>
  <image intensity="advanced expert" uri="detail.gif" ... />
  <text intensity="expert">CSMA/CD...</text>
</description>

```

Befindet sich im Beispiel das `description`-Element in einem Kontext, in dem noch keine Einschränkung bzgl. der Intensität erfolgte, so gilt das erste `text`-Element für alle Intensitätsstufen, das `image`-Element nur für die Modi `advanced` und `expert`, und der letzte Textabschnitt ausschließlich für den Experten.

Bei der Implementierung eines Moduls erfolgt die Codierung der unterschiedlichen Intensitäten i. A. wie folgt: Zunächst wird die vollständige Vertiefungsversion konzipiert, und diese wird dann schrittweise auf die Aufbau- und die Basisversion reduziert. Dies erlaubt eine gute Abschätzung der für die Bearbeitung der einzelnen Stufen nötigen Zeit.

### 3.6 Skalierbarkeit des Ausgabeformats

Das WWR-Dokumentenformat erlaubt die Generierung von konsistentem Material für verschiedene Nutzungsformen aus einer einzigen Modulbeschreibung. Modifikationen an der Modulbeschreibung wirken sich auf diese Weise direkt auf alle Ausgabeformate aus, was zu konsistentem Lehr- und Lernmaterial führt.

Während die Online-Version eines Moduls sowohl die kompletten Texte, vollständige multimediale Elemente, interaktive Anteile als auch Navigationsmöglichkeiten beinhaltet, umfasst ein druckbares Manuskript lediglich die Texte und, falls vorhanden, statische Versionen der dynamischen Multimedia-Elemente. Diese erhält man, wenn, wie in Abschnitt 3.2 beschrieben wurde, ergänzende textuelle Beschreibungen sowie repräsentative Bildfolgen zu dynamischen Multimedia-Elementen angegeben werden. Bei der Folienausgabe liegt der Fokus auf essentielle Multimedia-Objekte und kurzen erläuternden Texten.

Zur Auszeichnung der Skalierung des Ausgabeformats stellt die WWR-Dokumentsprache das Attribut `device` mit den möglichen Werten `online`, `script`, `slide` und `inherit` bereit. Wie bei den anderen Skalierungsformen gilt hier wieder das Restriktionsprinzip und die implizite Vererbung der Skalierungswerte.

### 3.7 Didaktisches Konzept

Die begleitende Entwicklung eines didaktischen Konzepts ist eine wichtige Voraussetzung für die Qualität und Wiederverwendbarkeit eines Moduls. Dem trägt die parallel zur sachlogischen Struktur eines Moduls definierte didaktische Strukturierung Rechnung. Sie unterteilt das Modul in Lerneinheiten, die aus einer Einleitung, beliebig vielen Lernschritten, einer Zusammenfassung und einem praktischen Übungsteil bestehen. Diese Komponenten beinhalten die in der sachlogischen Strukturierung (vgl. Abschnitt 3.1) definierten Präsentationseinheiten. Die Präsentationseinheiten können in einer neuen Reihenfolge arrangiert oder komplett neu definiert werden. Daneben sind auch die vorab erstellten Definitionen nutzbar. Dies erlaubt eine Anpassung des Materials an unterschiedliche didaktische Konzepte und eröffnet damit breitere Nutzungsmöglichkeiten.

## 4 Zusammenfassung und Ausblick

Im Rahmen des Verbundvorhabens „Wissenswerkstatt Rechensysteme“ entstehen derzeit multimediale Lehr- und Lernmaterialien auf Basis eines innovativen, in drei Dimensionen skalierbaren XML-Dokumentenformats. Jedes Modul kann bzgl. der Intensität, der Zielgruppe und des Ausgabeformats vom Nutzer an den jeweiligen Einsatzzweck angepaßt werden. Die Module enthalten didaktisch motivierte multimediale und interaktive Komponenten zur Effektivierung des Lernerfolgs. Ein flexibles Didaktik-Konzept erlaubt die einfache Modifikation eines Moduls durch den Dozenten. Das Dokumentenformat wurde unter Beachtung der Konzepte existierender Beschreibungssprachen entwickelt, um eine Transformation zu ermöglichen.

Aktuelle Arbeitsschwerpunkte sind neben der Implementierung von Modulinhalten und deren Integration in laufende Curricula vor allem die transparente Anbindung von externen Tools sowie die Optimierung der Werkzeuge zur Erstellung, Verwaltung und Nutzung der Module. Zukünftige Arbeitsbereiche umfassen die Einbindung von externen Partnern bzw. Materialien in die WWR-Umgebung sowie die Evaluation des WWR-Einsatzes aus didaktischer Sicht. Langfristig wird eine Ausdehnung des Vorhabens auf andere Fachgebiete, andere Ausbildungsbereiche (Abitur, Berufsausbildung, Weiterbildung) und andere Sprachen angestrebt.

## 5 Literatur

- [1] U. Lucke, A. Wiesner, H. Schmeck: „XML: Nur ein neues Schlagwort? Zum Nutzen von XML in Lehr- und Lernsystemen“, it+ti Sonderausgabe „Internet und neue Medien in der Aus- und Weiterbildung“, Oldenbourg Verlag, Juni 2002.
- [2] U. Lucke, D. Tavangarian: „Turning a Current Trend into a Valuable Instrument: Multidimensional Educational Multimedia based on XML“, Ed-Media 2002, Denver/Colorado/USA, 24.-29. Juni 2002.
- [3] IEEE Learning Technology Standards Committee: „Learning Object Metadata“. <http://ltsc.ieee.org/wg12>
- [4] Saba Software, Inc.: „Universal Learning Format“. <http://www.saba.com/standards/ulf>
- [5] Advanced Distributed Learning: „Sharable Content Object Reference Model“. <http://www.adlnet.org/>
- [6] Open Universiteit Nederland: „Educational Modelling Language“. <http://eml.ou.nl/>
- [7] Universität Passau: „Learning Material Markup Language“. <http://daisy.fmi.uni-passau.de/pakmas/lmml/>
- [8] W. Kalfa, R. Kroeger: „Integrating Simulators and Real-Life Experiments into an XML-based Teaching and Learning Platform“, Ed-Media 2002, Denver/Colorado/USA, 24.-29. Juni 2002.
- [9] A. Holzinger: „Basiswissen Multimedia“, Band 2, Vogel Verlag, Würzburg 2001.
- [10] A. Schreiber: „CBT-Anwendungen professionell entwickeln“, Springer-Verlag, Berlin 1998.