

Mit konventionellen Autorensystemen zum E-Learning Portal

Lutz Finsterle * , Martin Rotard †

Universität Stuttgart Fakultät Elektrotechnik, Fakultät Informatik
E-Mail: finsterle@ind.uni-stuttgart.de,
rotard@informatik.uni-stuttgart.de

Kurzfassung

Die Landschaft heutiger web-basierter Lehr- und Lernsysteme ist vielfältig. Dieser Vielzahl von Lehr- und Lernsystemen steht jedoch eine fast ebenso große Anzahl an bereits in Verwendung befindlicher Autorensysteme gegenüber. Soll ein Lernsystem erfolgreich nutzbar sein, so muss diesem Umstand Rechnung getragen werden. Soll ein Lernsystem den Anforderungen moderner Pädagogik gerecht werden, so sind viele Anforderungen zu erfüllen, zu denen nicht zuletzt die Akzeptanz durch Autoren, Lehrende und Lernende gehört. Dies erfordert, dass ein Lernsystem als ein an die Nutzer anpassbares Portal ausgeführt werden muss, welches im besonderen die Autoren nicht in der Wahl ihrer vielfältigen Autorensysteme einschränkt, aber ebenfalls die Aufbereitung der Lehr- und Lernmaterialien in verschiedenster Art und Weise ermöglicht. Diese Anforderung wird jedoch besonders im Autorenbereich in den wenigsten Fällen unterstützt, so dass sich die Autoren in ein neues Autorensystem einarbeiten müssen und zudem noch die große Menge an bereits existierenden Materialien händisch umsetzen und im schlimmsten Fall neu schreiben müssen. Hier wird eine Lösung vorgestellt, die dieses Problem minimiert und den ersten Schritt zu einem flexiblen Lernportal darstellt. Der gewählte Ansatz ist die Umsetzung der Materialien in ein autorensystemunabhängiges Zwischenformat. Der Aufwand für die Autoren vermindert sich ganz erheblich, indem es ihnen ermöglicht wird, ihre gewohnten Autorensysteme weiter zu verwenden und die bereits existierenden Materialien mit vertretbarem

Aufwand in das System einzupflegen.

Die Vorteile der beschriebenen Methode enden jedoch nicht an dieser Stelle. Darüber hinaus bietet sich die Möglichkeit der systemübergreifenden Wiederverwendung der Materialien durch automatische Generierung von Metadaten und Strukturinformationen.

1 Einleitung und Problemstellung

Der größte Feind des Wissens ist nicht der Irrtum, sondern die Trägheit.

Henry Thomas Buckle (1821 - 1862).

Am einfachsten kann ein E-Learning Portal eingeführt werden, indem man sich viele interessierte Partner mit ins Projekt holt bei denen das Wissen über den zu vermittelnden Lehrstoff bereits vorhanden ist. Dort existiert das „Wissen“ auf hunderten von Vortragsfolien und in Skripten. Durch Präferenzen bei der Verwendung unterschiedlicher Werkzeuge entstehen diverse Basisformate, die man idealerweise, für den Autor, direkt im Portal verwenden kann. Hier entstehen dann aber eine große Anzahl von Problemen bezüglich der Navigation durch die Materialien und der Wiederverwendbarkeit (Corporate Identity und Suchmechanismen sind hier zu nennen). Eine große Herausforderung ist es somit eine Methode zu entwickeln, damit jeder Autor sein präferiertes Werkzeug weiter verwenden kann und trotzdem ein einheitlich gestaltetes und strukturiertes E-Learning Portal entsteht. Dieser Beitrag stellt eine solche Methode vor.

Kapitel 2 beschreibt zunächst das Rahmenprojekt in dem die Methode entwickelt wird. In Kapitel 3 werden die Anforderungen an ein Lernportal, die Inhalte sowie dessen grundlegende Architektur dargestellt. Anschließend zeigt Kapitel 4 auf, warum

*Lutz Finsterle, Institut für Nachrichtenvermittlung und Datenverarbeitung, Pfaffenwaldring 47, 70569 Stuttgart

†Martin Rotard, Institut für Informatik, Abteilung Visualisierung und Interaktive Systeme, Breitwiesenstraße 20, 70565 Stuttgart

die Wiederverwendung von Inhalten für einen Autor wichtig ist. In Kapitel 5 wird ein Weg vorgestellt, um Basisdokumente für die Transformation in das autorensystemunabhängige Format aufzubereiten. Kapitel 6 beschreibt dann die Grundlagen und Randbedingungen für die Entwicklung eines zentralen Dokumentenformats. Darauf aufbauend wird in Kapitel 7 die Transformation der nativen Formate erläutert. Abschließend beschreibt Kapitel 8 die Erzeugung des Ausgabeformats.

2 ITO Rahmenprojekt

Im Verbundprojekt Information Technology Online (ITO) [2] des bmb+f-Rahmenprogramms „Neue Medien in der Bildung“ soll mit Hilfe von Multimedia- und Internettechnologien ein vielfältiges Lehrangebot im Umfeld internationaler Studiengänge der Fachrichtungen Elektrotechnik und Informationstechnik sowie Informatik aufgebaut werden. Dabei orientiert sich das Projekt an dem Master-Studiengang Information Technology der Universität Stuttgart, welcher sich auf die Themengebiete „Communications Engineering and Media Technology“ sowie „Embedded Systems Engineering“ konzentriert. Vergleichbare Inhalte werden auch an den Partneruniversitäten Dresden, Hamburg-Harburg und München gelehrt. Gerade in diesen Fachgebieten ist ein immer schnellerer Wandel des Fachwissens zu beobachten, so dass die Aktualität des Lehrstoffs sowohl bei der Ausbildung als auch bei der Weiterbildung ein immer wichtigerer Faktor wird. Zudem kann der Lehrstoff aufgrund der hohen Komplexität mit den konventionellen Lehrmedien nur unzureichend dargestellt werden. Das Projekt verfolgt daher das Ziel die Inhalte der von den Projektpartnern durchgeführten Vorlesungen multimedial aufzubereiten, so dass diese sowohl in Präsenzveranstaltungen als auch im Rahmen Web-basierter Lernszenarien eingesetzt werden können.

Um die Inhalte innerhalb verschiedener Lehr- bzw. Lernszenarien einsetzen zu können, ist ein modulares Konzept erforderlich. Daher erstellen die Projektpartner multimediale Lehrmodule zu ihren Vorlesungsinhalten, welche auf einfache Weise in andere Lehrveranstaltungen integrierbar sind. Aus den erstellten Modulen werden dann im Rahmen des Projektes multimediale englischsprachige Vorlesungen für den Studiengang Information Technology realisiert und im regulären Vorlesungsbetrieb einge-

führt.

3 Anforderungen und Architektur des Lernportals und der Inhalte

Bei der Konzeption eines Lernsystems welches die oben beschriebenen Anforderungen erfüllen soll müssen sowohl für das System selbst sowie für dessen Inhalte die im folgenden beschriebenen Randbedingungen erfüllt werden.

Bereits 2001 beschreibt Rob Koper in [1] die grundlegenden Anforderungen an Inhalte und deren Formate für deren erfolgreichen Einsatz in Lernsystemen. Er hat die folgenden elf Punkte herausgearbeitet:

- Formalisierung
- Pädagogische Flexibilität
- Explizit typisierte Lernobjekte
- Vollständigkeit der Beschreibung
- Reproduzierbarkeit von Ergebnissen
- Personalisierbarkeit
- Ausgabemedien Neutralität
- Austauschbarkeit und Nachhaltigkeit
- Standard Konformität
- Wiederverwendbarkeit von Inhalten
- Versionierung und Lebenszyklus

Diese Randbedingungen bilden die Basis für das hier beschriebene Dokumentenformat. Auf die einzelnen Punkte soll hier nicht weiter eingegangen werden, da ihre Auswirkungen und Umsetzung in den Kapiteln 6 bis 8 eingehend beschrieben werden.

An das Portal stellen sich zum Teil deckungsgleiche Anforderungen. Rolf Schulmeister beschreibt diese in [23]. Soll ein Lehr- und Lernsystem erstellt werden, welches als Erweiterung oder gar Ersatz des heutigen Universitätsbetriebs genutzt werden kann, so muss es die existierenden erfolgreichen Strukturen abbilden und erweitern, muss also sowohl die instruktivistischen als auch die konstruktivistischen Anteile der heutigen Universitätslehre implementieren. Um dieses zu leisten und um eine allgemeine Akzeptanz sicherzustellen müssen die Anforderungen

- Personalisierung
- Kommunikativität und
- Interaktivität

durch das System erfüllt werden. Bild 1 zeigt die wichtigsten Nutzerrollen und die Bausteine eines generischen Lernportalsystems. Hierbei stellt die Rolle des Nutzers den globalen Teil der Personali-

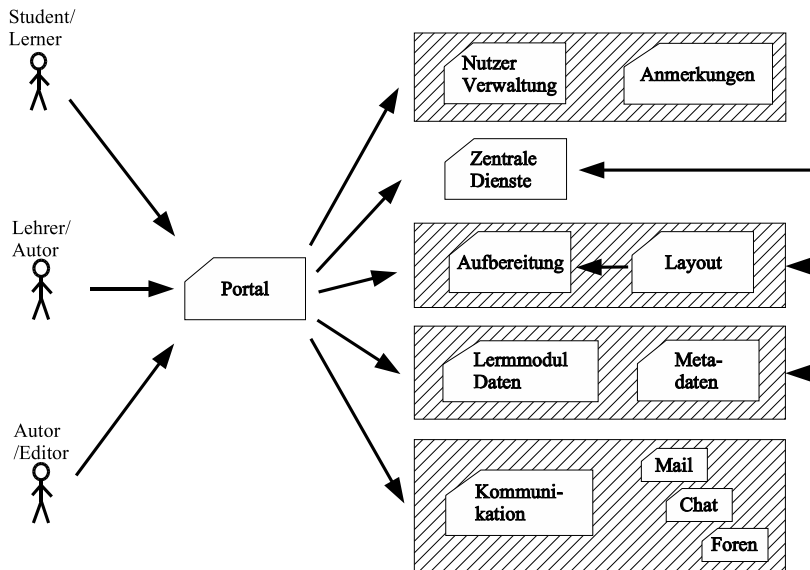


Abbildung 1: Nutzerrollen und Elemente des Portalsystems

sierung dar und muss mit weiteren Daten im Profil des jeweiligen Nutzers ergänzt werden. Diese weitergehenden Daten umfassen den Ausbildungsstand, Informationen über sensorische Behinderungen und die Zugehörigkeit zu einer Institution. Zu den Nutzerdaten und der Nutzerverwaltung kommen noch weitere Daten hinzu, welche dem Aspekt der Interaktivität des Systems Rechnung tragen, die Anmerkungen, welche jeder Nutzer zu beliebigen Inhalten für sich hinterlegen kann. Interaktivität innerhalb der Lernmodule wie Simulationen müssen vom System unterstützt (Aufbereitung) jedoch in den Inhalten modelliert werden.

Der Aspekt der Kommunikation wird durch in das Portal eingebundene Kommunikationsmodule realisiert, wobei hier ein Kontextbezug zu den Inhalten wichtig scheint und ebenfalls realisiert werden muss. In den folgenden Kapiteln wird ausgehend von den dargelegten Randbedingungen der erste aber wichtigste Schritt zu diesem System vorgestellt.

4 Integration und Wiederverwendung von Inhalten

Bei der Erstellung der Lehrmodule ist die Wiederverwendung ein sehr wichtiges Kriterium. Die Standardformate im Internet sind einem ständigen Wandel unterworfen [3], so lösen beispielsweise XML-basierte Dokumentenformate gerade HTML ab [4][5]. Lehrmodule müssen daher stets an die neuen Randbedingungen anpassbar sein. Betrachtet man die Entwicklung von Lehrmodulen in herkömmlicher Form (HTML, Powerpoint, etc.), so ist der Aufwand für den Autor die notwendigen Anpassungen vorzunehmen sehr hoch. Eine zum Beispiel in HTML direkt geschriebene Webseite lässt sich nur sehr aufwändig an neue Gegebenheiten anpassen. Eine nachhaltige Lösung dieses Problems ist, das aktuell benötigte Ausgabeformat ad hoc zu generieren. Diese Aufbereitung umfasst nicht nur die Darstellung des eigentlichen Inhaltes sondern im Besonderen auch die Erstellung von sinnfälligen Navigationsmöglichkeiten. Es folgt daraus, dass

der Lehrstoff strukturiert und unabhängig vom Ausgabeformat gespeichert werden muss und erst beim Abruf des Lehrmodules in Abhängigkeit der Anforderungen aufbereitet wird. Dazu ist es notwendig, den Inhalt getrennt von Layout, Struktur und den Metadaten zu speichern.

Besonders wertvoll ist die Speicherung der inhaltlichen Struktur der Absätze eines Textes im Sinne von Inhaltsdomänen. Innerhalb dieser semantischen Struktur gliedert sich der Inhalt weiter anhand von „pädagogischen“ Merkmalen. Wenn also ein Absatz innerhalb einer Inhaltsdomäne ein Attribut „Definition“ hätte, dann wäre es möglich den Absatz anhand dieses Attributes zu strukturieren. Das spätere Aussehen auf der Webseite wird dann durch dieses Attribut und eine verwendungsspezifische Layoutdefinition bestimmt. Dadurch erhalten alle Absätze des gesamten Lehrinhalts mit dem Attribut „Definition“ ein konsistentes Aussehen. Ferner ist es so möglich, sowohl Struktur- als auch Metadaten für die Inhalte automatisiert zu erstellen um so Navigationselemente generieren und Suchfunktionalitäten bieten zu können.

Ein Weg solche Attribute innerhalb eines konventionellen Autorensystems zuzuweisen, wird im nächsten Abschnitt beschrieben. Für den Autor bedeutet der dort beschriebene Weg eine enorme Aufwandsersparnis. Eine Gruppe von Autoren profitiert zusätzlich von der konsistenten Auszeichnung, ohne umfangreiche Richtlinien beachten zu müssen. Wenn sich dann die Techniken von HTML, CSS [6], etc. ändern, müssen nicht die Lehrmodule selbst geändert werden, sondern nur die Komponente, die für das Ausgabeformat verantwortlich ist.

5 Strukturierung der Basisdokumente mit Formatvorlagen

Um die oben angedeuteten Attribute den Absätzen in konventionellen Autorensystemen zuweisen zu können, werden Formatvorlagen verwendet. Diese müssen vorher zentral wohldefiniert werden, damit eine einheitliche und präzise Verwendung gewährleistet ist. Hierbei dürfen sich die Bezeichnungen und die Bedeutung der Absatzformate nicht mehr auf das Layout der Absätze beziehen. Das entwickelte Schema beschreibt vielmehr die „Semantik“ der Inhalte und die Struktur des Dokumentes. Beispiele für Formatvorlagen für Absätze sind:

- Introduction

- Motivation
- Definition
- Example
- Knowledge
- Exercise
- Solution, etc.

Wichtig ist dabei, nur so viele Formatvorlagen wie nötig zu definieren. Wenn ein Autor beim Schreiben oder Strukturieren eines Lehrtextes mehrere Formatvorlagen einem Absatz zuweisen könnte und jede zu einer anderen Formatierung führt, geht dabei der Sinn der Separation des Inhalts vom Layout verloren. Die Layout-Attribute der Formatvorlagen werden bei der anschließenden Transformation ignoriert. Dort ist nur der Name der Formatvorlage, das heißt dessen semantische Bedeutung relevant. Damit der Autor jedoch bereits bei der Bearbeitung des Lehrtextes eine Vorstellung von der späteren Formatierung erhält, bietet es sich an, das vordefinierbare Format der Vorlagen, im Sinne des WYSIWYG, entsprechend anzupassen.

Neben den Absatz-Formatvorlagen ist es ebenfalls notwendig Zeichen-Formatvorlagen mit semantischen Bedeutungen zu definieren. Dabei lassen sich zusammenhängende Zeichenketten innerhalb von Absätzen einheitlich mit semantischen Bedeutungen verknüpfen. Beispiele hierfür sind:

- Keyword
- Reference
- Index, etc.

In späteren Verarbeitungsschritten lassen sich durch diese Zeichen-Formatvorlagen automatisch Verzeichnisse generieren. Beispielsweise lässt sich jedes „Keyword“ in einem „Definition“ Absatz automatisch in den Glossar einfügen oder im Sinne einer automatischen Generierung von Metadaten, z.B. für Schlüsselworte, verwenden. Die Auswertung der „Reference“-Zeichenfolgen ermöglicht die Generierung eines Literaturverzeichnisses, wobei alle Referenzen in externe Dokumente als Literaturverweise und Referenzen in andere Lehrtexte des Systems als direkte Links realisiert werden können.

Die Tragfähigkeit des beschriebenen Ansatzes wurde anhand der Textverarbeitungssysteme OpenOffice 1.0 / StarOffice 6.0, Adobe Framemaker und Microsoft Word 97/2000 getestet. Es wurden Formatvorlagen erstellt und deren Austausch untereinander getestet sowie die Umsetzung der Dokumente in das autorensystemunabhängige Format proto-

typisch nach dem in Kapitel 7 beschriebenen Verfahren durchgeführt. Für den Austausch unter Autorentensystemen bietet sich weiterhin das Rich-Text-Format (RTF) an, welches von allen getesteten Produkten unterstützt wird. Bei dem Weg über dieses Format bleiben alle Formatvorlagen erhalten und lassen sich von dort aus in die jeweiligen anderen Systeme importieren, jedoch gibt es bei Grafiken, Formeln und anderen komplexen Objekten Kompatibilitätsprobleme.

6 Grundlagen und Randbedingungen für die Entwicklung eines autoren-systemunabhängigen zentralen Dokumentenformates

Die Problematik liegt also wie beschrieben in der Vielfältigkeit der bereits vorliegenden Dokumente und der Notwendigkeit auch in Zukunft die Autoren in einer Form zu unterstützen, die eine breite Akzeptanz des Systems ermöglicht. Zusätzlich sind die in Kapitel 3 aufgelisteten Randbedingungen zu beachten.

Speziell sind hierbei die folgenden Punkte von Relevanz:

- leichte Anpassbarkeit der Inhalte an die Bedürfnisse des Nutzers (Personalisierung)
- einfache Aufbereitung der Inhalte nach pädagogischen Anforderungen und Designkriterien (Ausgabemedien Neutralität)
- Wiederverwendbarkeit der Lernmodule
- Vollständige Beschreibung der Lernmodule für z.B. Suchfunktionalität
- Genaue Beschreibung der semantischen Zusammenhänge für die Navigation und die pädagogisch sinnvolle Aufbereitung

Die hier beschriebenen Kriterien in Verbindung mit den restlichen Punkten aus Kapitel 3 legen die im folgenden aufgezeigte Basis nahe:

- Das Format sollte XML basiert sein, da dieses eine fast uneingeschränkte Bearbeitung und Transformation der Materialien ermöglicht.
- Es sollten ausschließlich offene Standards verwendet werden, da dies die Anzahl der Möglichkeiten maximiert bereits existente Tools einzusetzen. Der Einsatz offener Standards ist auch für die Nachhaltigkeit der erzeugten Materialien von essenzieller Bedeutung.
- Die Strukturierung der Inhalte sollte aus-

schließlich nach Sinnzusammenhang (Domänenmodell) der Lehrmaterialien und pädagogischen Kriterien und nicht nach Layout Gesichtspunkten erfolgen, da nur so eine sinnvolle Modularisierung erreicht werden kann.

- Die Speicherung von Inhalts-, Struktur-, Layout- und Metadaten sollte getrennt erfolgen.
- Die Möglichkeit eine geschlossene Version zu erzeugen, in dem alle Informationen eines Kurses enthalten sind, soll gegeben sein. Dies dient unter anderen auch dazu, den Kurs an einem Stück in einem entsprechenden Autorentool zu erstellen.

Einen prinzipiell gleichen Ansatz verfolgt auch das Projekt TargeTeam mit dem innerhalb des Projektrahmens entwickelten Formates TeachML [7][8]. Das hier im folgenden vorgestellte und anhand der oben genannten Kriterien entwickelte Format stellt eine Generalisierung von TeachML dar.

6.1 Modularisierung des Quelldokumentes

Die Notwendigkeit der Trennung von Inhalt, Struktur, Layout und Metadaten, die in Kapitel 5 begründet wurde, führte zu der in diesem Kapitel im einzelnen beschriebenen Modularisierung.

Die Modularisierung wird in den „Document Type Definitions“ (dtd's)[9] für die jeweiligen Dokumente definiert. Die Aufteilung und deren Zuordnung zu den festgelegten Hierarchieebenen findet sich in Bild 2. Die folgende Aufzählung erläutert die Module im einzelnen:

- Inhaltlich/Pädagogisch getriebene Ebene
Diese Ebene enthält alle Inhalte des Lehrmaterials, sie fasst die textuellen und multimedialen Anteile zusammen, betrachtet aber nicht deren Strukturierung. Die Dokumentenbeschreibung unterteilt sich in die Definition von Textabsätzen (*Inhalt.dtd*), Multimedialen Objekten (*Objekt.dtd*) und Zeichenformatierungen (*Zeichen.dtd*).
- Ebene der Metadaten
Hier werden die zu den einzelnen Inhaltselementen notwendigen Metadaten beschrieben. Das Format der Metadaten orientiert sich am LOM-Standard (Learning Objects Model) und wird in der *Meta.dtd* beschrieben [15].
- Ebene der Strukturierung der Materialien und

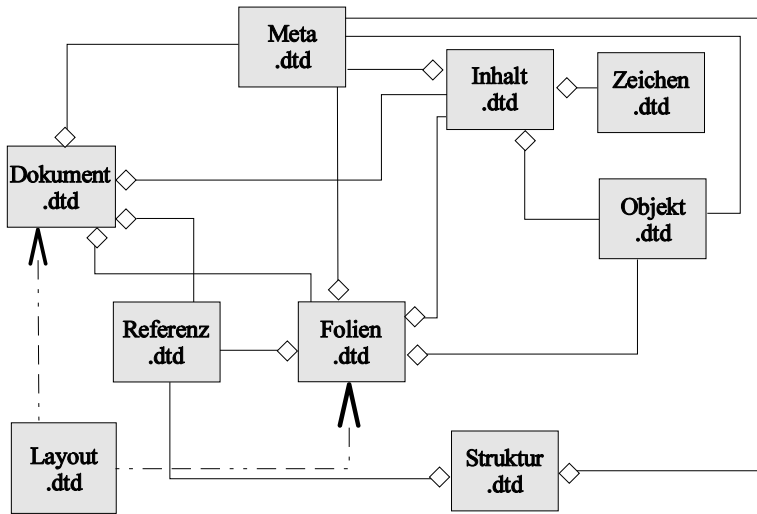


Abbildung 2: Abhängigkeiten der DTD's

ihrer semantischen Zusammenhänge in einer Lehreinheit

Auf dieser Ebene wird der strukturelle Zusammenhang der Inhaltselemente beschrieben. Die Struktur setzt sich aus den in Kapitel Lehr- und Lernobjekte und deren Struktur beschriebenen Elementen zusammen und wird in der *Struktur.dtd* beschrieben. Auf dieser Ebene werden im allgemeinen keine Inhaltselemente direkt eingebettet, sondern als Verweise auf ein Inhaltsdokumente realisiert (*Referenz.dtd*). (Siehe auch Kapitel 6.4)

- Beschreibung von Folien-Elementen

Folien bilden eine Ausnahme in diesem Kanon. Die Einführung dieser Ebene geschieht aus rein pragmatischen Gründen. Hauptfaktor ist hier die Unterstützung von folienbasierten Autorenwerkzeugen und deren Möglichkeiten. Abgebildet wird hier die Rahmenstruktur von Folien, welche eine freiere Kombination von textuellen und grafischen Elementen ermöglicht (*Folien.dtd*). Diese Folienelemente sind nicht nur für Präsentationszwecke interessant, sondern können auch die Grundlage für annotierte und erweiterte Begleitmaterialien bilden.

- Vollständiges Dokument

Die Zusammenfassung aller oben beschriebenen Elemente in ein Dokument, das die geschlossene, vollständige Bearbeitung einer Lehreinheit in einem Autorenwerkzeug ermöglicht. Besonderes Augenmerk liegt hier auf der Aufwandsersparnis die sich ergibt wenn nicht zwei getrennte Dokumente für z.B. eine Druck- und eine Präsentationsversion gepflegt werden müssen (*Dokument.dtd*).

- Layout Beschreibungen

In der Layout-Beschreibung können Hinweise zu besonderen Layout-Anforderungen abgelegt werden. Diese dienen in erster Linie der Rücktransformation in das native Ausgangsformat, können aber durchaus auch für die Aufbereitung innerhalb des Portals herangezogen werden.

Nachdem die Struktur des autorensystemunabhängigen Formates in groben Zügen dargestellt wurde, soll nun in den folgenden Abschnitten genauer auf die einzelnen Module und in den folgenden Kapiteln auf deren Anwendung eingegangen werden.

6.2 Inhaltselemente und deren Struktur

Die Inhaltselemente spiegeln die in Kapitel 5 beschriebenen Absatzformate wieder. Um eine eindeutige Unterscheidung zwischen textuellen und grafischen Elementen zu gewährleisten, werden diese in verschiedenen DTD's definiert. Die Definitionen haben hierbei den gleichen Elementnamen, jedoch verschiedene Namensräume. Auf diese Weise können die jeweiligen Elementeigenschaften getrennt beschrieben werden. Dies führt zu einer wesentlichen klareren Beschreibung der einzelnen Elemente im Sinne einer Eigenschaftskapselung.

Die Elemente haben dabei die folgende Struktur:

- Textuelle Elemente


```
<itoccontent:Absatztyp type="simple">
  text
</itoccontent:Absatztyp>
```
- Grafische Elemente


```
<itobjekt:Absatztyp           ty-
  pe="formula|illustration|animation|simulation
  |table|list">
  <meta:title>
    title
  </meta:title>
  <itobjekt:[formula|illustration|animation|
  simulation|table|list]>
    objekttyp-spezifische Elemente
  </itobjekt:[formula|illustration|animation|
  simulation|table|list]>
  <meta:annotations>
    Annotationen
  </meta:annotations>
</itobjekt:Absatztyp>
```

Der Inhalt der objekttyp-spezifischen Elemente ist hierbei fest definiert. Dies wird dadurch ermöglicht, dass die Formate der multimedialen Objekte festgelegt wurden. Die Zuordnung von Medientypen zu dem jeweilig verwendeten Format ist in Tabelle 1 dargestellt.

Medientyp	Format
Formeln	MathML
Illustrationen	SVG, X3D
Animationen	SVG+JavaScript, X3D
Simulationen	JAVA2 Applets

Tabelle 1: Medientypen und deren Formate [10] [11] [12] [13] [14]

6.3 Metadaten

Die Definition der Metadatenstruktur basiert auf dem Learning Object Model (LOM) des Learning Technology Standards Committee (LTSC)[16]. Nach eingehendem Studium der Standards für Metadaten, hier sind im wesentlichen Dublin Core, Ariadne und LOM zu nennen und der Betrachtung anderer Projekte (CANDLE [17], Ariadne [18]) sowie der Handhabbarkeit der Menge der einzufügenden Metadaten wurde hier eine Untermenge von LOM ausgewählt. Diese enthält die notwendigen Elemente zur Beschreibung und Wahrung von Urheberrechten, Version, Pädagogik und Inhalt. Es werden hierbei Metadaten für Lern Einheiten, Lern Module, Multimediale Elemente (siehe Kapitel 6.4 und alle inhaltlichen Elemente und Objekte gefordert.

6.4 Lehr- und Lernobjekte und deren Struktur

Die in ITO definierte Struktur für Lehr- und Lernobjekte sieht vor, dass Inhalte anhand der im folgenden genannten Objekte zusammengefasst und strukturiert werden (siehe Abbildung 3):

- Lehreinheit
- Lehrmodul
- Multimediales Element
- Verbindungselement

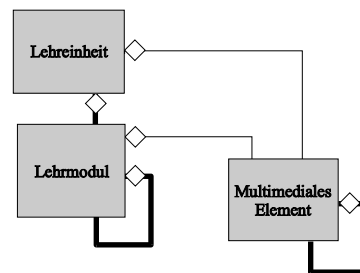


Abbildung 3: Zusammensetzung der Lehrelemente

Eine LehrEinheit (LE) befasst sich mit einem klar umrissenen Fachgebiet und besteht aus einer Menge von Lehrmodulen, die in ein übergeordnetes organisationelles, curriculares Konzept eingebettet sind. Diese Einbettung umfasst u.a. das Betreuungskonzept, organisationelle Lernbedingun-

gen, Lehrmodul-übergreifende Designentscheidungen wie Corporate Identity, etc.

Ein LehrModul (LM) ist ein thematisch geschlossenes, wiederverwendbares Lernobjekt mit klar definierter Lernaufgabe. Es kann inhaltlich in weitere Lehrmodule substrukturiert sein. Technisch gesehen besteht ein Lehrmodul aus einer Menge multimediale Elemente und Einleitungen und Überleitungen zwischen den Substrukturen, die als Verbindungselemente (VE) realisiert werden. Multimediale Ele-

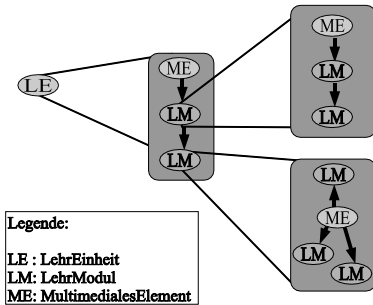


Abbildung 4: Aufbau einer Lehrereinheit

mente (ME) enthalten Referenzen auf wiederverwendbare Textelemente, Folien, Grafik, Video, Audio, Animation, Simulation oder bereits definierte Multimediale Elemente.

Die Struktur dieser Dateien wird in der *Struktur.dtd* definiert, welche sich an dem Resource Description Framework (RDF)[19] orientiert.

Hierbei ergibt sich die Granularität dieser Objekte aus dem inhaltlichen Zusammenhang der im Quelldokument enthalten ist. Das Zugrundelegen eines solchen Domänenkonzeptes bildet bei weitem die stabilste Plattform für eine Umsetzung. Implizit ergibt sich, dass die Domänen im allgemeinen bereits vom Autor in herkömmlichen Materialien eingeführt werden. Die Strukturierung von Textbüchern und Vorlesungsfolien zeigt dies im allgemeinen sehr deutlich.

Greifen wir als Beispiel die Begleitmaterialien zu einer Vorlesung auf, welche in diesem Fall als Buch ausgeführt sind, so finden wir das Domänenmodell in der Kapitelstruktur wiedergespiegelt.

Auf Basis der *Struktur.dtd* ist es möglich, Lehrereinheiten und deren semantische und pädagogische Strukturen zu beschreiben. Bild 4 zeigt den prinzipi-

ellen Aufbau einer Lehrereinheit (LE) und seiner eingebetteten Lehrmodule. Es wird deutlich, dass nicht nur sequenzielle Strukturen, sondern auch semantische Zusammenhänge wie sie in Seminaren gefordert werden, modellierbar sind.

6.5 Die Struktur der Folienelemente

Die Struktur der Folienelemente im autorensystemunabhängigen Format bildet die Rahmenstruktur aus präsentationszentrierten Autorenwerkzeugen ab. Diese ermöglicht dem Autor an der freien Positionierung von Text- und Grafikelementen festzuhalten und präsentationsgerechte Zusammenstellungen von Lehrelementen zu erzeugen. Die einzelnen Folien können dann als Multimediale Elemente in eine Lehrereinheit oder ein Lehrmodul eingebettet und dort beschrieben, annotiert und referenziert werden.

6.6 Die Zusammenfassung in ein geschlossenes Dokument

Die Zusammenfassung von inhaltlichen sowie Strukturierungs- und Folienelementen in eine einzige Datei dient zur Unterstützung künftiger moderner Autorentools und zur geschlossenen Speicherung eines Abzuges einer ausgewählten Lehrereinheit. Ermöglicht wird damit auch der Export von Lehrereinheiten zu anderen Lehrsystemen. Wesentlicher Unterschied zur Strukturdefinition durch die *Struktur.dtd* ist, dass die Strukturierung des Dokumentes hierbei in Kapitel und Unterkapitel im Sinne eines Skriptes erfolgt, in welches auch Folienelemente an beliebiger Stelle eingefügt werden können. Es ist somit leicht möglich aus einem Dokument sowohl eine Druck- als auch eine Präsentationsversion mit Anmerkungen zu generieren.

7 Transformation der nativen Formate

Die Transformation der nativen Formate der verschiedenen Autorenwerkzeuge erfolgt entweder direkt oder über einen Zwischenschritt. Die Unterstützung eines ebenfalls XML-basierten Formates als Eingabeformat für den Transformator bildet einen guten Ausgangspunkt, da somit das Implementieren verschiedener Filter für binäre native Formate

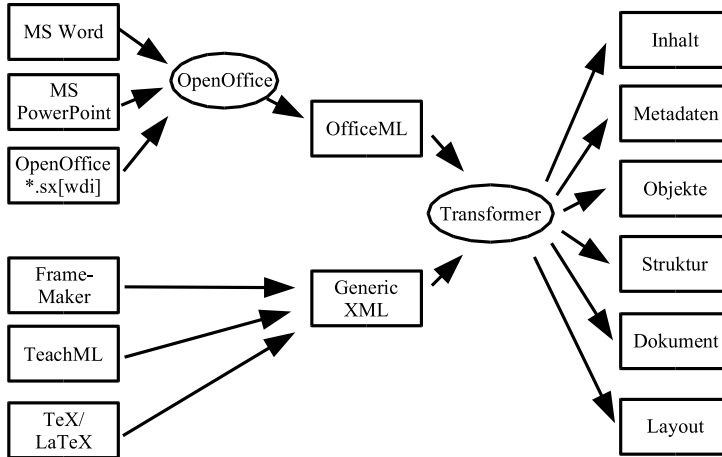


Abbildung 5: Transformation der nativen Dokumente

nicht notwendig ist. Das opensource Produkt OpenOffice nutzt ein XML-basiertes Dokumentenformat zur Speicherung und verfügt weiterhin über gute Ein- und Ausgabefilter für Fremdformate. So benutzen wir OpenOffice [20] als Mediator für den ersten Schritt der Transformation. Das von OpenOffice erzeugte Dokumentenformat hat zwar wesentliche Einschränkungen in der Struktur des Dokumentes [21], lässt sich jedoch unter Einhaltung der in Kapitel 5 beschriebenen Randbedingungen leicht in das beschriebene Zwischenformat transformieren (siehe Abbildung 5). Hilfreich ist hierbei vor allem das von der OpenOffice-Gruppe getriebene Festhalten an offenen Standards, wie beispielsweise SVG [11] und MathML [10]. So kann ein beliebiger XML-Parser zum Einlesen des OpenOffice Dokumentes verwendet werden, eine einfache Heuristik genügt um die Transformation durchzuführen und ein großer Teil der in den Dokumenten eingebetteten grafischen Objekte kann nahezu unverändert übernommen werden.

Die Implementierung des Transformationswerkzeugs wird hierbei vollständig mit Java und Extensiblen Stylesheets (XSLT)[22] durchgeführt um das System sowohl leicht anpassbar, als auch plattformunabhängig zu gestalten.

8 Präsentation der Inhalte

Im letzten Schritt wird aus den separat gespeicherten Informationen über den Inhalt, das Layout, die Struktur und den Metadaten ein Ausgabeformat ad hoc, sowohl für die Inhalte als auch die Navigation, generiert. Diese Aufgabe übernehmen die in Abbildung 6 dargestellten Ausgabetransformatoren. Ausgabemedien sind für das Web und den Druck vorgesehen. Außerdem ist es über eine Rücktransformation möglich, Quellformate für die Autorentools zu generieren.

Bei der Web-Darstellung für den Lernenden steht eine pädagogisch sinnvolle Präsentation im Vordergrund. Heutige Lehrformen werden im Portal abgebildet sowie die verschiedenen Endnutzertypen (Studenten, Schüler, berufliche Weiterbildung, etc.) berücksichtigt. Darüber hinaus kann eine Lernkontrolle stattfinden. Dem Lernenden soll auch die Möglichkeit gegeben werden, verschiedenste Kommunikationsplattformen innerhalb einer virtuellen, kooperativen Lernumgebung zu verwenden (E-Mail, Diskussionsforum, Chat, Voice over IP)[23].

Nutzerspezifischen Daten werden in einem Nutzerprofil verwaltet. Damit ist es möglich jedem Nutzer eine für seine Bedürfnisse angepasste Lernum-

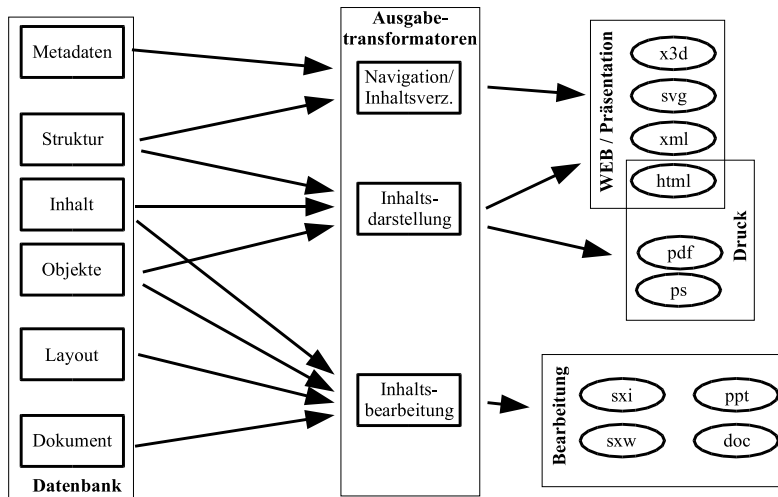


Abbildung 6: Transformation des autorenwerkzeugunabhängigen Formates zur Ausgabe

gebung zu generieren. Diese kann auch von ihm individuell konfiguriert werden. Dies erleichtert auch das Erschließen der Inhalte für sensorisch behinderte Lernende. Die Generierung des Ausgabeformats hängt auch von den verschiedenen Endsystemen (Webbrowser, Ausgabegerät, Betriebssystem, etc.) ab [24].

Die ad hoc Generierung des Ausgabeformats bietet viele Vorteile:

- Für jeden Autor kann eine individualisierte Arbeitsumgebung generiert werden, die seinem gewohnten Arbeitsstil und Werkzeugen angepasst ist.
- Für jeden Lernenden kann eine individualisierte Lernumgebung generiert werden, die seinen Lernbedürfnissen und Lerngewohnheiten angepasst ist.
- Die Wartung der so abgelegten Inhalte ist flexibel, da jederzeit ein Ausgabeformat geändert oder neu hinzugefügt werden kann. Dabei werden keine Parallelversionen erzeugt, die einzeln gepflegt werden müssen. Änderungen im System fließen dabei sofort in die Ausgabe mit ein.
- Einfache Anpassung der Ausgabe an unterschiedliche Endsysteme, so zum Beispiel Medientumwandlung falls das Quellformat auf

dem Endsystem nicht unterstützt wird.

- Spezielle Ausgabeformate für beispielsweise CD-/DVD-Kurse lassen sich generieren.

Zusammenfassung

In diesem Beitrag wurde eine Methode vorgestellt, Material für ein E-Learning Portal aufzubauen, die die Autoren nicht an ein Autorensystem bindet und die einfache Wiederverwendung von Inhalten verschiedener Autoren ermöglicht. Die Methode trennt Inhalt, Layout, Struktur und Metadaten innerhalb des Systems durch die Transformation in das hier beschriebene modulare Zwischenformat und generiert diese Daten automatisiert. Der Abruf und die Aufbereitung dieser Module aus dem System kann dann anhand der Arbeitsumgebung und der Fähigkeiten des Lernenden ad hoc geschehen.

Literatur

- [1] Rob Koper. *Modeling units of study from a pedagogical perspective*, <http://eml.ou.nl/introduction/articles.html>, June 2001

- [2] Information Technology Online (ITO), *ITO Homepage*, <http://www.ias.uni-stuttgart.de/ito/index.htm>
- [3] Rolf Brugger. *Die Erstellung von wiederverwendbaren Inhalten für Web-basierte Kurse*, Virtueller Campus: Szenarien - Strategien - Studium, E. Wagner, M. Kindt (Hrsg.), Medien in der Wissenschaft, Band 14, Waxmann Verlag, Münster, 2001
- [4] World Wide Web Consortium, *Extensible Markup Language (XML) Spezifikation*, <http://www.w3.org/xml>
- [5] World Wide Web Consortium, *HyperText Markup Language (HTML) Spezifikation*, <http://www.w3.org/MarkUp>
- [6] World Wide Web Consortium, *Cascading StyleSheets Level 2 Spezifikation*, <http://www.w3.org/Style/CSS>
- [7] Projekt TARGETEAM, *Targeteam Homepage*, <http://www11.in.tum.de/forschung/projekte/targeteam/index.html.de>
- [8] Projekt TARGETEAM, *The TeachML Language*, <http://www11.in.tum.de/forschung/projekte/targeteam/extension/html/targeteam4.html>
- [9] World Wide Web Consortium, *Document Type Definition (DTD) Spezifikation*, <http://www.w3.org/graphics/xml>
- [10] World Wide Web Consortium, *MathML Spezifikation*, <http://www.w3.org/math>
- [11] World Wide Web Consortium, *Scalable Vector Graphics (SVG) Spezifikation*, <http://www.w3.org/graphics/svg>
- [12] Web3D Consortium, *X3D Spezifikation*, <http://www.web3d.org/x3d>
- [13] ECMA, *ECMA Script Spezifikation*, <http://www.ecma.ch/ecma1/STAND/ECMA-262.HTM>
- [14] SUN, *Java Language Homepage*, <http://java.sun.com>
- [15] IEEE Learning Technology Standards Committee, *LOM Standard*, http://ltsc.ieee.org/doc/wg12/LOM_WD6-1_without_tracking.htm
- [16] IEEE Learning Technology Standards Committee, *LTSC Homepage*, <http://ltsc.ieee.org>
- [17] Cooperative And Network Distributed Learning Environment, <http://www.candle.eu.org>
- [18] Alliance of Remote Instructional Authoring and Distribution Networks for Europe (ARIADNE), *ARIADNE Homepage*, <http://ariadne.unil.ch>
- [19] World Wide Web Consortium, *Resource Description Framework Spezifikation*, <http://www.w3.org/rdf>
- [20] Open Office Organisation, *OpenOffice Homepage*, <http://www.openoffice.org>
- [21] Open Office Organisation, *OpenOffice Dokumenten Spezifikation*, http://xml.openoffice.org/xml_specification_draft.pdf
- [22] World Wide Web Consortium, *Extensible Stylesheet Language Spezifikation*, <http://www.w3.org/TR/xsl/>
- [23] Schulmeister, Rolf, *Virtuelle Universität - virtuelles Lernen*, München [u.a.], Oldenbourg, 2001, Kapitel 7
- [24] K. Meißner, S. Röttger, F. Wehner (2001), *Dynamische Visualisierung modularer, XML-basierter Kursdokumente*, in: Proceedings of "11. Arbeitstreffen der GI-Fachgruppe 1.1.5/7.0.1 „Intelligente Lehr-/Lernsysteme“, Dortmund, Germany.