

Modulare Aufbereitung von multimedialen Lerninhalten für eine heterogene Lernumgebung

Khaldoun Ateyeh, Jutta A. Mülle, Peter C. Lockemann

Universität Karlsruhe, Institut für Programmstrukturen und Datenorganisation
(IPD), 76128 Karlsruhe, Germany, E-Mail ateyeh@ira.uka.de, muelle@ira.uka.de

Interner Bericht 1999-17

Universität Karlsruhe

Fakultät für Informatik

ISSN 1432-7864

Kurzfassung

Der Einsatz multimedialer Lerninhalte ist durch die vielfältig vorhandenen und sich ständig verbessernden technischen Möglichkeiten ein aktuell weit verbreitetes Thema. Ihre Erstellung gestaltet sich allerdings äußerst aufwendig. Daher beschäftigen wir uns mit der kooperativen Entwicklung von multimedialen Lehr-/Lerninhalten, die von verschiedenen Autoren in unterschiedlichen Kontexten für verschiedene Zielgruppen eingesetzt werden können. Unser Ansatz basiert auf dem Modularisierungskonzept zur Unterstützung der Wiederverwendbarkeit von Lernmaterialien. Um solche „universellen“ Lernmodule zu entwickeln, die sowohl die Bedürfnisse verschiedener Zielgruppen als auch verschiedener Autoren erfüllen, werden mehrere Modularisierungsebenen eingeführt und die Erzeugung verschiedener Modulsichten ermöglicht.

Schlüsselworte: Lernmodule, Multimedia Design, Kooperative Entwicklung von Lernmodulen, Hypermedia Design.

1 Einleitung und Motivation

Die Entwicklung multimedialer Lerninhalte ist ein technisch mühsamer, konzeptionell iterativer und damit insgesamt zeitaufwendiger Prozess. Wichtige charakteristische Merkmale des Entwicklungsprozesses multimedialer Systeme, die ihn vom Entwicklungsprozess anderer Softwaresysteme unterscheiden, sind die Interdisziplinarität der Entwicklergruppen (Informatiker, Psychologen, Graphikdesigner, Medienspezialisten, Fachexperten,...), die hohen Anforderungen an Anschaulichkeit, Kreativität und künstlerische Gestaltung sowie die Beachtung von psychologischen und ergonomischen Aspekten neben rein programmier-technischen bzw. ingenieurmäßigen Tätigkeiten im Entwicklungsprozess [1]. Zur Zeit entwickelte Lerninhalte stehen diesen Merkmalen entgegen. Ihre monolithische Struktur macht die Erweiterung, Überarbeitung, Aktualisierung und Wiederverwendung der Lerninhalte sehr aufwendig. Gesucht wird insbesondere nach einer Möglichkeit, um die mit viel Aufwand entwickelten multimedialen Lerninhalten wiederverwenden zu können.

Ein erster Schritt liegt bei der Wiederverwendung multimedialer Bausteine, wie z.B. einzelne Animationen, Simulationen, Video-Clips, Bilder, Audio; die Bausteinwiederverwendung ist derzeit auch Thema in verschiedenen anderen Forschungsvorhaben.

Doch sind diese Bausteine häufig zu feingranular, um den Lehrkontext gebührend zu berücksichtigen. Erwünscht ist also ein Bausteinkonzept, das stärker lehrinhaltsbezogen ist. Dies wiederum wirft die Frage auf, ob man bei einer inhaltsorientierten Wiederverwendbarkeit nicht Nutzen aus den Eigenschaften des Modularisierungskonzepts ziehen kann, so wie es aus dem Bereich der Softwaretechnik bekannt ist. Dem Ziel, den Entwicklungsaufwand für multimediale Lernelemente zu verringern, stellt sich im Projekt Virtueller Hochschulverbund Karlsruhe (ViKar) [2] ein auf verschiedenen sogenannten Autorengruppen beruhender Ansatz. Das Besondere hieran ist, dass diese Gruppen hochschulartübergreifend sind; sämtliche Hochschulen in Karlsruhe sind an ihnen beteiligt. Die Autorengruppen haben das Ziel, multimediale Lernmodule gemeinsam zu entwickeln. Entstehen soll eine modular aufgebaute Sammlung multimedialer Lernmodule, die dann hochschulartspezifisch zu Kursen kombiniert werden können, welche sich also jeweils an Studierende verschiedener Hochschularten (Universitäten (Uni), Fachhochschulen, (FH) und Berufsakademien (BA)), aber auch an interessierte Praktiker, die sich beruflich weiterbilden möchten, richten.

Neben dieser „Makro“-Ebene der Kombination ist auch noch eine „Mikro“-Ebene denkbar, in der man je nach Bedarf Teile aus den Lernmodulen ausblenden kann. So ist etwa denkbar, dass ein Modul für unterschiedliche didaktische Konzepte verwendbar, für verschiedene Einsatzbereiche, wie Uni, FH, BA, und für mehrere

Verwendungszwecke, wie z.B. Vorlesung, Übung, Selbststudium einsetzbar sein sollte, um die Wiederverwendbarkeit sinnvoll zu realisieren.

So schön und interessant eine solche Vision von Lernmodulen ist, die ähnlich den Softwaremodulen¹ von jedem benutzt werden können, besteht das Problem, dass erstmals Wege und Mittel gefunden werden müssen, die die Entwicklung solcher Module möglich machen.

Im weiteren werden wir zuerst die Anforderungen an ein solches Konzept aufführen, einen kurzen Überblick über die einschlägige Literatur geben und dann nach der grundsätzlichen Vorgehensweise für die modulare Entwicklung von Lerninhalten einen Prozess beschreiben, der die Entwicklung von Lernmodulen möglich macht. Als Beispielszenario wird ein Teil aus der Relationentheorie und Normalisierung aus dem Gebiet der relationalen Datenbanken verwendet.

2 Anforderungen an die Entwicklung multimedialer Lerninhalte

Die Anforderungen leiten wir aus dem Einsatzbereich an verschiedenen Hochschulen ab, wie er im ViKar-Projekt charakterisiert ist, und ergänzen ihn um einige etwas realisierungstechnischen Aspekte, die bei der Entwicklung eines praktikablen Konzeptes von vorne herein mit berücksichtigt werden sollen.

- Wiederverwendbarkeit: Im Gegensatz zu anderen Projekten wie z.B. ARIADNE [3] ist unser Ziel, die Wiederverwendbarkeit nicht nur auf der Ebene der multimedialen Bausteine (Animation, Simulation, Audio, Text,...) zu fördern, sondern auch die Wiederverwendbarkeit auf einer höheren Ebene, nämlich auf der Ebene der Lernmodule zu unterstützen. Die Lernmodule sollen von verschiedenen Lehrenden oder Lernenden in verschiedenen Kontexten eingesetzt werden können.
- Hochschulartübergreifende Lernmodule: Die Lernmodule sollen sich an verschiedene Zielgruppen (Uni, FH, BA,...) richten. Verschiedene Zielgruppen haben verschiedene Charakteristika (Vorwissen, Lernziele,...), die bei Entwurf und Gestaltung der Module berücksichtigt werden müssen.
- Unterstützung von Präsenz- und Fernstudium: Die entwickelten Lernmaterialien sollen sowohl das Präsenz- als auch das Fernstudium unterstützen; dies erfordert verschiedene Sichten auf das Lernmaterial.

¹ Das Problem der Wiederverwendbarkeit ist auch im Software-Bereich noch nicht vollständig gelöst. Dies gilt insbesondere dann, wenn mehr Semantik an die Module geknüpft wird, speziell kritisch ist das Auffinden geeigneter Module, bzw. überhaupt die Definition, was geeignet heisst.

- Anpassungsfähigkeit:

Anpassung an die Studenten: Das Lernmaterial soll sich an den verschiedenen Merkmalen der Lernenden wie Lernendertyp, Lernstil, Vorkenntnisse orientieren, sowie an die Bedürfnisse der Lernenden anpassen. Die Lernmaterialien sollen dazu verschiedene Sichten anbieten, zwischen denen der Lernende bei Bedarf wechseln kann, um die passende Sicht für seine Lernsituation und seinen Lernstil wählen zu können.

Anpassung an die Autoren (Dozenten): Verschiedene Autoren haben verschiedene Bedürfnisse. Sie benutzen verschiedene Didaktiken, die für ihre Zielgruppen geeignet sind. Autoren sollen die Möglichkeit haben, ihre eigenen spezifische Sichten auf die Lernmaterialien zu entwickeln. Sie sollen die Möglichkeit haben, Lernmodule so zu konfigurieren, dass sie ihre eigenen didaktischen Ideen und Bedürfnisse unterstützen.

- Kooperation: Die Unterstützung der Hochschulart-übergreifenden Zusammenarbeit bei der Entwicklung von Lehreinheiten soll gefördert werden. Es soll möglich sein, Teile oder komplette Lernmodule parallel zu bearbeiten. Dies führt zur effizienten Aufteilung der Aufgaben auf die Mitglieder einer Autorengruppe.
- Unterstützung von Standards: Um die Entwurfsumgebung offen und erweiterbar zu gestalten, sollen wo möglich Standards, wie z.B. für Metadaten IMS [4] oder XML [5] eingesetzt werden.
- Automatisierung: Es soll möglich sein, möglichst viele Schritte des Entwicklungsprozesses von Lernmaterialien zu automatisieren, um den Entwicklungsaufwand zu reduzieren.

3 Verwandte Ansätze

In letzter Zeit sind auf dem Markt mehrere sogenannte Lernsysteme erschienen. Beispiele für diese Produkte sind: IBM Learning Space [6], Oracle Learning Architecture [7], WebCT [8], Hyperwave Training Space (GENTLE) [9][10]. Lernsysteme stellen eine Lernplattform zur Verfügung, sie enthalten Komponenten zur Erstellung und Verwaltung von Lerninhalten sowie zur Nutzer- und Kursverwaltung; andere wichtige Komponenten sind solche zur Unterstützung der Kommunikation und Kooperation zwischen den Benutzern (Chat, Mail, Tele Conferencing, etc.). Die existierenden Lernsysteme verwenden für die Erstellung von Lerninhalten selbst lediglich Werkzeuge zur Erstellung von Bausteinen, wie z.B. zur Animation. Da es uns um die Wiederverwendbarkeit größerer Einheiten geht, können die genannten Systeme bestenfalls Basisfunktionalität beitragen.

In die von uns geforderte Richtung könnten eher die in den letzten Jahren entstandenen Ansätze zur Modellierung von Hypermedia-Anwendungen [14][15][16] gehen. Diese Modelle wie RMM [15] oder OOHDM[16] konzentrieren sich allerdings auf Präsentationsaspekte wie den Entwurf von Navigationsstrukturen zwischen den Hypermedia-Objekten. Wir fordern jedoch, dass die Modellierung und Strukturierung der Hypermedia-Objekte selbst im Mittelpunkt steht. Wichtige Aspekte, die wir bei der Modellierung von Hypermedia-Objekten verfolgen, sind Wiederverwendbarkeit und kooperatives Design, was in den oben genannten Modellen bisher nur am Rande betrachtet wurde. Das ARIADNE-Projekt [3][17] ist ein Ansatz in diese Richtung, untersucht die Möglichkeit der Wiederverwendbarkeit von Courseware allerdings, im Gegensatz zu uns, lediglich auf der Ebene der Multimedia-Bausteine und nicht von Lernmodulen.

Die genannten Ansätze – so wie die Wiederverwendbarkeit auch an sich – legen eine Trennung von Inhalt und Präsentation nahe, wie es im Model View Controller (MVC) von Small Talk [11] oder Java Swing [12], oder auch in SGML [13] und XML verfolgt wird.

4 Grundsätzliche Vorgehensweise

Die besondere Herausforderung an dem von uns verfolgten Ansatz liegt in der Bestimmung der Lernmodule dergestalt, dass sie einen geschlossenen Lehrinhalt widerspiegeln, andererseits aber so minimal aber auch flexibel gestaltet sind, dass sie ohne größere Eingriffe in unterschiedliche Kurse eingebaut werden können und durch einfache Handhabung auf eine der in Kapitel 2 genannten Nutzungsumgebungen eingestellt werden können. Unser Grobkonzept [19] zur Erstellung multimedialer Lehrmaterialien postuliert die Zerlegung des Lehrstoffes in sogenannte Lernmodule. Ein *Lernmodul* ist eine thematisch abgeschlossene, wiederverwendbare, semantische Lerneinheit, die Motivation und Lösung einer bestimmten Problemstellung enthält und semantisch nicht mehr sinnvoll weiter zerlegt werden kann. Syntaktisch besteht ein Lernmodul aus einer oder mehreren Bildschirmseiten, welche wiederum aus unterschiedlichen Multimediaabausteinen, wie z.B. Text, Grafik, Video, Audio, Animation, zusammengesetzt sind. Die Wiederverwendbarkeit von Bausteinen selbst ist natürlich insbesondere bei multimedialem „Mehrwert“ ebenfalls von Interesse, da ihre Erstellung sich sehr aufwendig gestaltet.

Ziel des Grobkonzeptes ist die Erstellung eines Kurses, der aus Lernmodulen zusammengestellt wird, und in unterschiedlichen Präsentationsformen (auch Lehrformen genannt), wie z.B. Volltext, Vorlesungsfolien, Kurzform, und mit verschiedenen Navigationsstrukturen, wie Guided Tour, Glossar, exploratives Lernen, den Lernenden zur Verfügung gestellt wird. Es werden zwei Entstehungs-

phasen unterschieden: die *Modulerstellung durch Autoren*, deren Ergebnisse - die Lernmodule und ihre logischen Abhängigkeiten - in einer Modulverwaltungs-komponente abgelegt werden, und die *Kurserstellung durch Dozenten*, in der der Kurs aus den Modulen zusammengestellt wird, sowie für die Module entsprechende Präsentationsformen und Navigationsstrukturen entlang der Module des Kurses ausgewählt werden.

Im weiteren werden wir ein verfeinertes Modularisierungskonzept für Lernmodule vorstellen, in dem mehrere Abstraktionsebenen, wie in Abbildung 1 dargestellt, eingeführt werden.

Ausgangspunkt ist ein sogenanntes *Integrationsmodul*, das noch weitgehend Didaktik-frei ist, d.h. keine Informationen über die Lernform beinhaltet, die Positionierung der Inhalte offen lässt oder die Art der Gestaltung und Präsentation der Inhalte frei hält. Das hier enthaltene Lehrmaterial ist also noch nicht geprägt von den unterschiedlichen Nutzern und Dozenten sowie noch unabhängig von den Navigationsstrukturen zur dynamischen Nutzung des Lehrmaterials. Es enthält andererseits die gesamten inhaltlichen Informationen, die von den verschiedenen Nutzern benötigt werden.

Aus dem Integrationsmodul entstehen über eine Art Sichtenbildung die sogenannten *Sichtenmodule*, die im Hinblick auf ihre Nutzung strukturiert sind und die Aspekte Lernform, Zielgruppen und Autorenbedürfnisse berücksichtigen. Bei der Sichtenbildung erfolgt eine Auswahl der zu übernehmenden Modulbestandteile sowie ggfs. eine Umstrukturierung. Es muß auch möglich sein, neue Bestandteile, die nur von einem Autor benutzt werden und daher nur in dessen Sicht erscheinen, aufzunehmen.

Auf der nächsten Ebene wird die Präsentation der definierten Sichtenmodule, erstellt. Hierzu werden sowohl psychologische als auch pädagogische und didaktische Faktoren berücksichtigt. Ergebnis dieses Schritts sind die *Präsentationsmodule*. Schließlich wird aufbauend auf den Präsentationsmodulen eine *Lerneinheit* oder auch Kurs erstellt, in dem unter Berücksichtigung der Abhängigkeiten der Module untereinander eine Auswahl der in der Lerneinheit enthaltenen Module sowie evtl. mehrere Navigationsstrukturen über die Module festgelegt werden.

Auf allen Modularisierungsebenen unterscheiden wir zwischen einem Modultyp und den Instanzen eines Modultyps. In den Entwurfphasen werden die Modultypen entworfen und die Spezifikation² der einzelnen Instanzen festgelegt. Ein Beispiel für einen Modultyp ist die Kurzformsicht mit Attributen (Bestandteilen) wie Titel, Beschreibung und Motivation; eine Instanzspezifikation hierzu ist z.B. eine Kurzform 3NF mit Attributen wie Titel und Beschreibung aber ohne Motivation (siehe Abb. 3). Die Instanziierung oder die

² Die Spezifikation der Module legt u.a. fest, aus welchen Bestandteilen sie bestehen.

Erstellung der einzelnen Module (Instanzen eines Modultyps) erfolgt dann in der Implementierungsphase. Im Beispiel werden in der Kurzform 3NF ein Titel und eine konkrete Beschreibung realisiert, d.h. den Attributen (Bestandteilen) werden konkrete Werte zugewiesen.

5 Der Entwicklungsprozess

Die Abstraktionsebenen für Multimedia-Lerninhalte wirken sich direkt auf die Phasen des Entwicklungsprozesses aus. Abbildung 2 gibt einen Überblick. In der Analysephase werden die zu betrachtenden Zielgruppen und die grobe

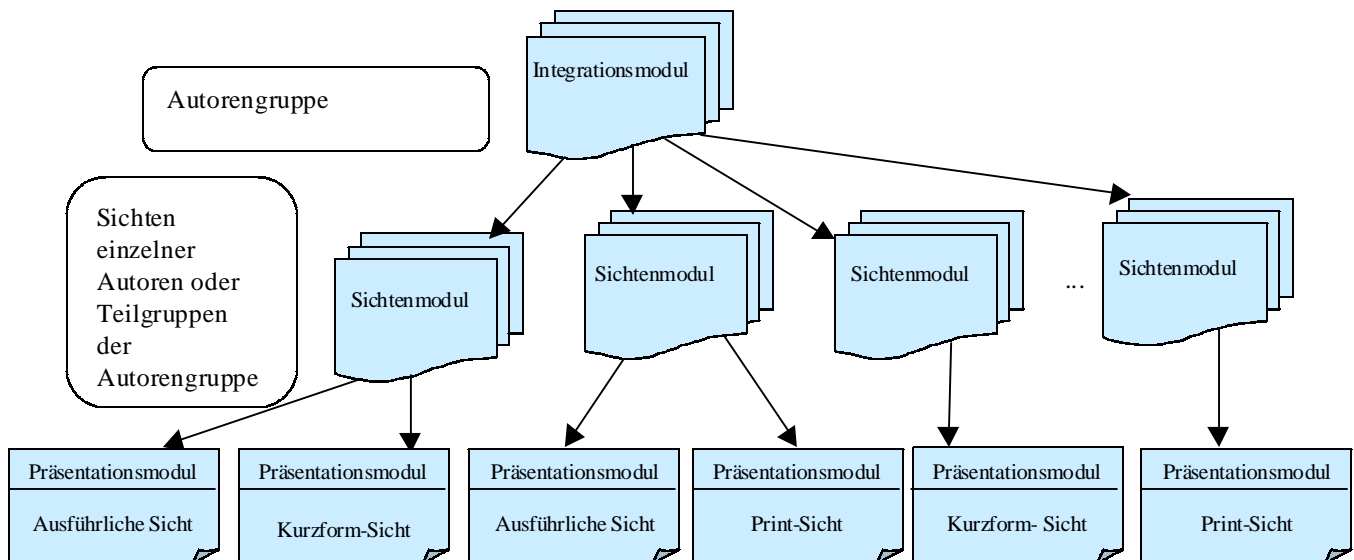


Abbildung 1: Modularisierungsebenen

Lernziele ermittelt. Die Entwurfs- und Implementierungsphase sind jeweils in Teilphasen aufgeteilt. Diese Teilphasen können teilweise parallel ablaufen, z.B. kann man während des Entwurfs in Teilphase 2 mit der Implementierung der Ergebnisse der Entwurfsteilphase 1 beginnen. Im folgenden werden wir die Phasen des Entwicklungsprozesses genauer betrachten.

5.1 Anforderungsanalyse

Diese Phase dient zur Analyse und Abgrenzung des Umfangs des Themenbereiches. In diese Phase gehen Lehrmaterialien ein, die in unterschiedlicher Form vorliegen, wie z.B. Lehrbücher, Inhalte und Vorstellungen

gen, die nur „in den Köpfen“ der Autoren existieren, oder auch Lehrmaterial, das bereits in elektronischer Form verfügbar ist. Ziele dieser Phase sind:

- Identifizierung der Zielgruppen: Welche Zielgruppen sollen die zu entwickelnden Lernmaterialien unterstützen (Uni, FH, BA, Praktiker...)?
- Identifizierung der Lernziele, d.h. Abgrenzung des Themengebietes: welche Lernziele sollen mit dem Inhalt verfolgt werden?

5.2 Entwurf

Im Entwicklungsprozess durchläuft ein Lernmodul verschiedene Stadien. In jedem Stadium gehört ein Lernmodul zu einem Modultyp, der bestimmte Eigenschaften hat und bestimmte Anforderungen erfüllt (siehe Abbildung 1). Der Entwurfsprozess beschäftigt sich mit dem Design der verschiedenen Modultypen.

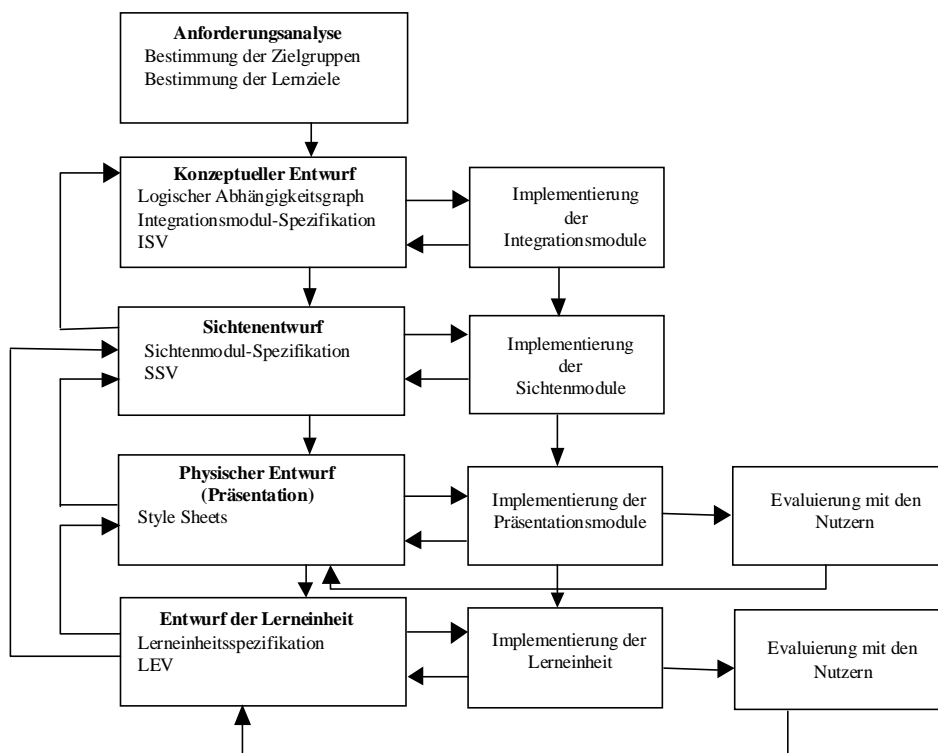


Abbildung 2: Der Entwicklungsprozess

Konzeptueller Entwurf (Integrationsmodul)

Diese Phase hat die Strukturierung des Lehrmaterials in Module zur Aufgabe, d.h. die Einteilung in semantisch nicht weiter zerlegbare Einheiten. Diese Module werden dann in einem logischen Abhängigkeitsgraphen verknüpft werden, um darzustellen, welche Lehrinhalte Voraussetzung zum Verständnis anderer sind. Es entsteht ein konzeptuelles Modell des Lehrmaterials, welches noch unabhängig ist von den unterschiedlichen Nutzern und Dozenten (Autoren) sowie den Navigationsstrukturen zur dynamischen Nutzung des Lehrmaterials. Die Module sind in dieser Phase Didaktik-frei, sie beinhalten keine Informationen über die Lernform, die Positionierung der Inhalte oder die Gestaltung und Präsentation der Inhalte. In dieser Phase werden die Module Integrationsmodule genannt. Integrationsmodule stellen eine Art gemeinsames Framework für die Autoren dar. Sie beinhalten alle gemeinsamen Informationen, die von den Autoren für die Erstellung der Lernmodule benötigt werden. Ergebnisse dieser Phase sind:

- **Logischer Abhängigkeitsgraph:** Der logische Abhängigkeitsgraph beschreibt die Aufteilung des Themenbereiches in Module und wie diese Module voneinander logisch³ abhängen.
- **Integrationsmodulspezifikation:** Hier wird eine Spezifikation für jedes Integrationsmodul erstellt. Die Spezifikation enthält zwei Arten von Informationen:
 - Verwaltungsinformationen: Hier werden alle Informationen, die zur Verwaltung der Integrationsmodule notwendig sind, beschrieben (z.B. Modulname, Autor etc.). Standards für Metadaten wie IMS [4] werden berücksichtigt.
 - Liste der Modulbestandteile: Zu einem Modulbestandteil gehören z.B. Definition und Motivation.
- **Integrationsmodulstruktur-Vorlage (ISV):** Die ISV definiert einen Integrationsmodultyp. Sie ist mit dem Konzept des Datentyps in den imperativen Programmiersprachen oder mit dem Konzept der Klasse in objektorientierten Programmiersprachen zu vergleichen. Sie definiert die inhaltlichen Bestandteile eines Modultyps. Für die Realisierung der Vorlagen ist die Document Type Definition (DTD) von SGML/XML geeignet. Falls keine ISV vorhanden ist, die für den Themenbereich geeignet ist, wird an dieser Stelle eine ISV entworfen. Es ist denkbar, dass man eine allgemeine ISV entwirft, die für mehrere Themenbereiche geeignet ist. ISVs werden in einem ISV-Repository verwaltet, wodurch die Wiederverwendbarkeit der ISVs unterstützt wird.

³ Logische Abhängigkeiten zwischen Modulen sind z.B. dass für das Verständnis des Lernmoduls A der Inhalt des Lernmoduls B vorausgesetzt wird.

Im Bezug auf unser Anwendungsszenario ist der Abhängigkeitsgraph des Themenbereichs Relationentheorie und Normalisierung mit den Bestandteilen jedes Moduls in Abbildung 3 dargestellt. Eine mögliche ISV kann wie folgt aussehen⁴:

```

<!ELEMENT Integrationsmodul (title, keyword*, motivation?, definition*,
description*, method*, algorithm*, note*, example*, summary?, exercise*)>
<!ELEMENT title (figure?, #CDATA)>
<!ELEMENT keyword (figure?, #CDATA)>
<!ELEMENT motivation (reference | #CDATA)>
<!ELEMENT reference EMPTY>
<!ELEMENT definition(reference | #CDATA)>
<!ELEMENT description (reference | #CDATA)>
<!ELEMENT method (reference | #CDATA)>
<!ELEMENT algorithm (reference | #CDATA)>
<!ELEMENT note #CDATA>
<!ELEMENT example (reference | #CDATA)>
<!ELEMENT summary #CDATA>
<!ELEMENT exercise (reference)>
<!ATTLIST Integrationsmodul
id #CDATA
management reference>
<!ATTLIST reference
id #CDATA
mediaType CDATA>

```

5.2.1 Sichtenentwurf (Sichtenmodul)

Auf dieser Ebene werden die Module im Hinblick auf ihre Nutzung strukturiert. Wichtige Aspekte dabei sind:

- Lernform: Für welche Lernformen (Fernstudium, Präsenzstudium, Selbststudium) werden die Module eingesetzt? Module die als Folien in einer Vorlesung benutzt werden, werden anders strukturiert als die Module, die in einem Web-Basierten Training eingesetzt werden.
- Zielgruppen: Für welche Zielgruppen werden die Module eingesetzt? Verschiedene Zielgruppen haben verschiedene charakteristische Merkmale, die bei der Strukturierung beachtet werden sollen.

⁴ Für die Beschreibung der Vorlagen verwenden wir die XML-Notation für die DTD.

- Autorenbedürfnisse: Wie soll der Lehrstoff didaktisch aufbereitet werden? Autoren sollen die Möglichkeit haben die Module so zu konfigurieren, daß sie ihre eigenen didaktischen⁵ Methoden und Interessen widerspiegeln.

Um dem oben genannten Aspekten gerecht zu werden, werden durch Konfigurieren aus den Integrationsmodulen Sichten aufgebaut, die für die spezifischen Merkmale der einzelnen Autoren, Lernenden und Lernformen zugeschnitten sind. Mögliche Sichten sind Kurzform-Sicht (Dozentsicht), die z.B. für die Nutzung als Folien in einer Vorlesung verwendet werden kann, eine ausführliche Sicht (Studentensicht), in der entsprechend einem (Online-)Skriptum ausführlich die Lehrinhalte des Moduls enthalten und beschrieben sind, eine Schnelldurchgangssicht, in der z.B. keine Beweise und Beispiele enthalten sind, usw. . Die Konfigurierung erfolgt dann durch folgende Schritte:

- Auswahl von Bestandteilen aus dem Integrationsmodul, die für die Sichten des jeweiligen Autors benötigt werden.
- Ergänzung des Moduls um neue, nur durch den Autor genutzte Bestandteile, wie z.B. eine Motivation für den im Modul enthaltenen Lehrstoff.
- Neue Strukturierung und Ergänzung der ISV in Hinblick auf die zu unterstützenden Sichten. Im Normalfall wird für jede Sicht eine Sichtvorlage von der ISV abgeleitet. Es ist aber denkbar, dass ein Autor (oder mehrere Autoren) eine einzige Vorlage für alle seine/ihre Sichten erstellt. Wir bezeichnen die von der ISV abgeleitete Vorlage Sicht-Struktur-Vorlage (SSV).

Ergebnisse dieser Phase sind:

- **Sichtenmodulspezifikation:** Hier wird die Spezifikation der Integrationsmodule für jede Sicht verändert, so dass sie die Ergebnisse der Konfigurierung (Auswahl von Bestandteilen, Ergänzung um neue Bestandteile) enthalten.
- **SSV:** für jede Sicht wird, falls keine SSV vorhanden ist, an dieser Stelle eine SSV entworfen. Eine SSV beschreibt ähnlich einer ISV die logische Struktur einer Klasse von Sichtmodulen; sie definiert einen Sichtenmodultyp. Es ist denkbar, dass jeder Autor eine auf ihn angepasste allgemeine SSV entwirft, die für mehrere Themenbereiche geeignet ist.

In unserem Anwendungsszenario wäre z.B. für jeden Autor (Uni, FH), jeweils eine spezifische Kurzform-Sicht von Bedeutung. Sie wären aber auch an einer gemeinsamen ausführlichen Sicht interessiert. Damit würde jeder Autor einen Konfigurierungsschritt für seine spezifische Kurzformsicht durchführen. Für die

⁵ Der Einsatz von Multimedia in der Lehre stellt neue Anforderungen an die Didaktik. Im Projekt Vikar beschäftigt sich eine Gruppe von Didaktikern und Pädagogen mit Multimedia-Didaktik.

gemeinsame ausführliche Sicht würden die Autoren den Konfigurierungsschritt gemeinsam durchführen. Die SSVs sehen dann wie folgt aus:

Kurzformsicht der Uni:

```
<!ELEMENT KurzFormUni ( title, motivation?, definition*, description*, method*, algorithm*, note*, example*, summary?)>
```

Kurzformsicht der FH:

```
<!ELEMENT KurzFormFH ( title, motivation?, definition*, description*, method*, algorithm*, note*, example*, exercise*, summary?)>
```

Gemeinsame ausführliche Sicht für Uni und FH:

```
<!ELEMENT AusführlicheSichtUni+FH (title, motivation?, definition*, description*, method*, algorithm*, example*, summary?, exercise*)>
```

5.2.2 Präsentationsentwurf (Präsentationsmodul)

Auf dieser Ebene wird das äußere Aussehen der auf der Sichtenebene definierten Sichtenmodule erstellt. Für jede Sicht wird ein geeignetes Style Sheet entworfen. Bei der Erstellung des Style Sheets werden sowohl psychologische als auch pädagogische und didaktische Faktoren berücksichtigt. Ergebnis dieses Schritts sind die Präsentationsmodule. Die Definition der Style Sheets stellt eine natürlich ganz erhebliche Anforderung an Kreativität und künstlerische Fähigkeiten. Die Gestaltung des Aussehens der Module ist meistens ein interaktiver sog. User Centered Prozeß [19], der iterativ über die Schritte Design, Implementierung und Evaluierung mit den Benutzern erfolgt (siehe Abbildung 2). Ergebnisse dieser Phase sind die Style Sheets für jedes Modul.

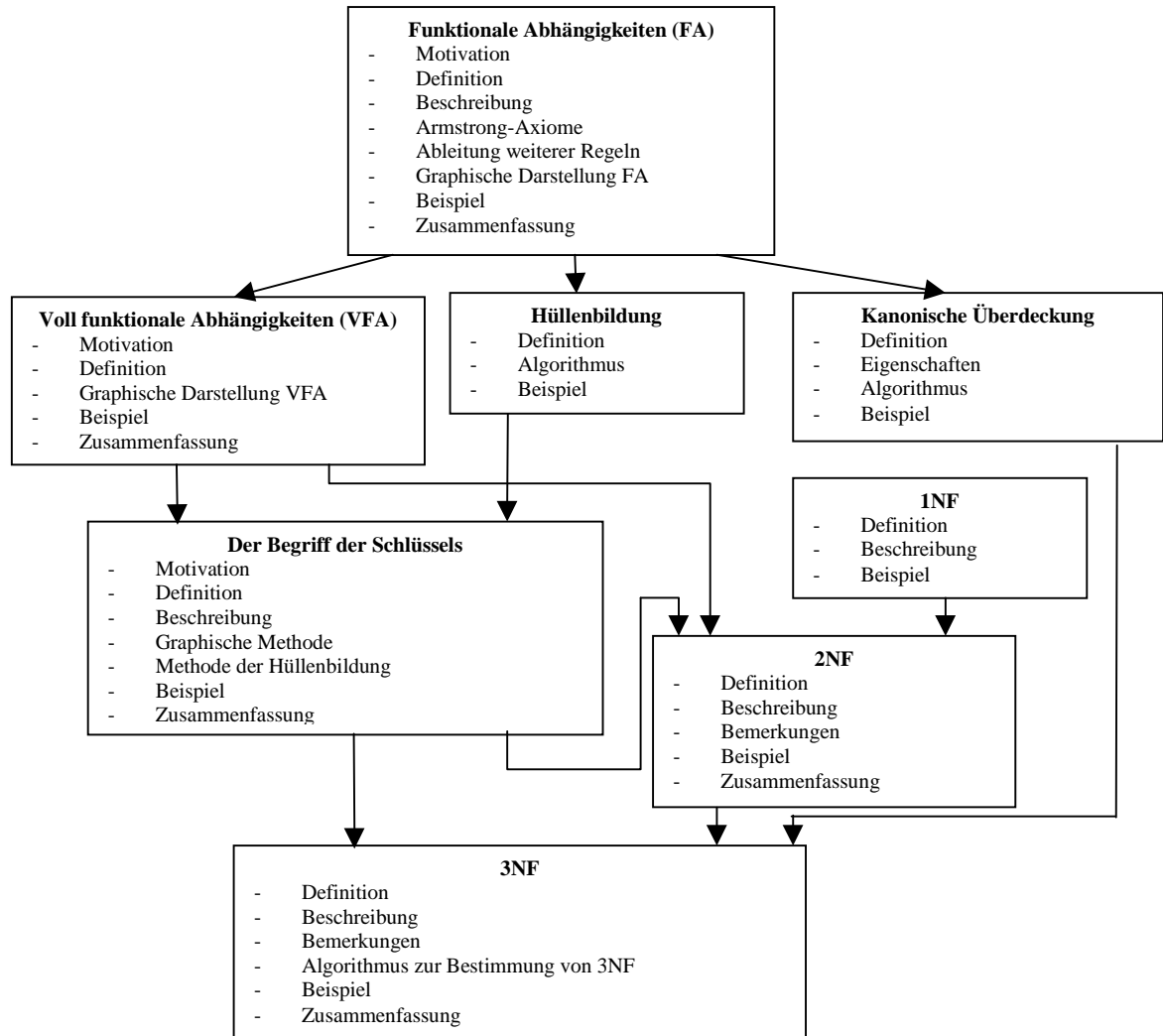


Abbildung 3: Der Abhängigkeitsgraph für einen Ausschnitt aus dem Themenbereich Relationentheorie und Normalisierung

5.2.3 Entwurf der Lerneinheit

Im nächsten Schritt wird unter Nutzung der Präsentationsmodule und mit Berücksichtigung des logischen Abhängigkeitsgraphen eine **Lerneinheit** zusammengestellt, die neben der Auswahl der enthaltenen Module evtl. mehrere Navigationsstrukturen entlang der Module festlegt. Es handelt sich hier wiederum um eine Konfigurationsaufgabe, nun aber auf der Ebene der Menge der Module, d.h. ein Modul selbst ist hier nicht mehr intern konfigurierbar. Beispiele für Navigationsstrukturen auf Lerneinheiten sind explorative Navigation durch das Lehrmaterial, Suche nach Stichworten oder sonstigen Suchbegriffen, eine Guided Tour, ein Glossar oder auch eine hierarchisch organisierte Anordnung der Module einer Lerneinheit. Die Erstellung von Lerneinheiten ist dank der Abhängigkeitsgraphen automatisierbar, was die Erstellung der Navigationsstrukturen angeht, nachdem die Auswahl der in die Lerneinheit eingehenden Module interaktiv mit dem Dozenten geklärt wurde. Für jede Art der Navigationsstrukturen kann man Vorlagen entwerfen, die immer wieder von den verschiedenen Autoren für die Realisierung ihrer Lerneinheiten benutzt werden können. Ein System, das ein derartiges Konzept realisiert, ist Hyperwave (GENTLE) [9][10].

Ergebnisse dieser Phase sind:

- **Lerneinheitsspezifikation:** hier wird eine Spezifikation für jede Lerneinheit erstellt. Die Spezifikation beinhaltet die Namen der Module, die zu einer Lerneinheit gehören und wie sie von einander abhängen. Bei der Spezifikation wird der Abhängigkeitsgraph berücksichtigt.
- **Lehreinheitsvorlage (LEV):** Entwurf oder Auswahl einer Vorlage, die eine geeignete Navigationsstruktur für die Lerneinheit beschreibt.

5.3 Produktion von Modulinhalten (Implementierung)

In dieser Phase werden die Inhalte aufbereitet und die erstellten Vorlagen mit diesen Inhalten gefüllt. Es handelt sich hier um die Erstellung der spezifizierten Instanzen in den Entwurfsphasen. Im folgenden wird die Erstellung von Inhalten (Instanzen) auf den verschiedenen Ebenen betrachtet.

5.3.1 Erstellung der Integrationsmodule

Abbildung 4 zeigt den Erstellungsprozess von Integrationsmodulen.

- Schritt 1: mit Hilfe der Verwaltungsinformationen in der Spezifikation wird nach dem Modul in den Modul-Repositories gesucht. Falls das Modul gefunden wird, wird es, falls notwendig, an eigene Anforderungen angepasst.

Falls das Modul nicht gefunden wird, wird für jeden Inhaltsbestandteil in der Liste der Modulbestandteile ein Entwicklungsprozess gestartet (siehe Abbildung 4). Bei dem Entwicklungsprozess von Modulbestandteilen handelt es sich um die Entwicklung eines Multimedia-Bausteins, der den Bestandteil realisiert. Der Entwicklungsprozess der Multimedia-Bausteine ist dem Entwicklungsprozess der Module sehr ähnlich. Man sucht den Baustein in Baustein-Repositories (wie ARIADNE [3]). Falls man ihn nicht findet, wird man ihn je nach Art (Animation, Text, Audio,..), mit einem geeigneten Werkzeug entwickeln.

- Schritt 2: Nachdem alle Bausteine vorhanden sind, die eine Realisierung der Modulbestandteile bereitstellen, wird die Modulstruktur-Vorlage (ISV) mit den Bausteinen gefüllt. Um dies mit dem Software-Konzept des Datentyps oder der Klasse zu vergleichen, entspricht diese Füllung der Vorlagen mit den Inhalten, der Instanzierung von den Klassen. Es wird ein Instanz eines Integrationsmodultyps erzeugt.

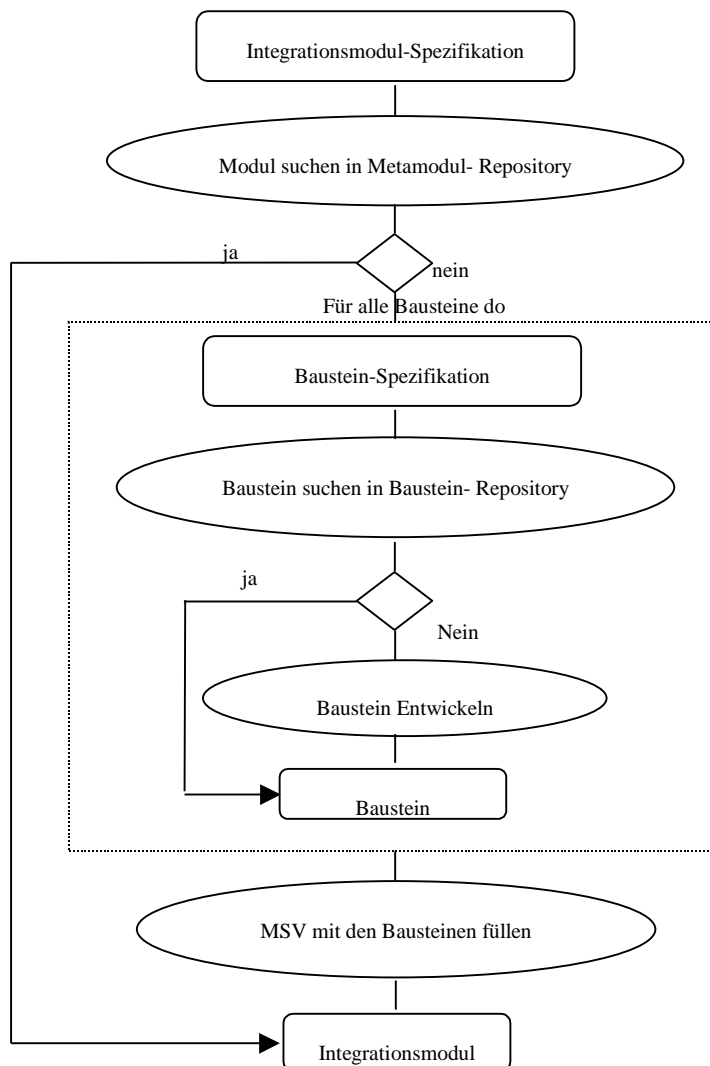


Abbildung 4: Erstellung von Integrationsmodulen

5.3.2 Erstellung der Sichtenmodule

Nach der Konfiguration werden gleich spezifizierten Bestandteile der Sichtenmodule von den Integrationsmodulen übernommen. Für die neuen Bestandteile wird ein Entwicklungsprozess ähnlich dem bei der Entwicklung der Modulbestandteilen aber für die Integrationsmodule gestartet (siehe Abbildung 4).

5.3.3 Erstellung von Präsentationsmodulen

Von den Sichtenmodulen und den zugehörigen Style-Sheets werden die Präsentationsmodule automatisch generiert.

5.3.4 Erstellung von Lerneinheiten

Die erstellten Vorlagen aus der Entwurfsphase der Lerneinheiten werden instanziiert, d.h. mit den spezifizierten Modulen gefüllt.

6 Zusammenfassung

Für die Erstellung multimedialer Lerninhalte wurde ein Konzept vorgestellt, das – im Vergleich zu der Nutzung multimedialer Bausteine, wie Animationen, Simulationen, etc. - auf der Wiederverwendbarkeit von größeren Einheiten, sog. Lernmodulen beruht. Die Lernmodule werden auf verschiedenen Ebenen entwickelt und erlauben damit die Anpassung an unterschiedliche didaktische Konzepte, an unterschiedliche Lernformen und Nutzungsformen der Lernenden sowie an die unterschiedlichen Anforderungen, die aus der Hochschulart, in der sie eingesetzt werden, resultieren. Es wurde ein Entwurfsprozess entwickelt, der auf den verschiedenen Modulebenen zwischen der jeweiligen Spezifikation und der Instanziierung, d.h. dem „Füllen mit Inhalten“ unterscheidet. Unsere bisherigen Evaluierungen basieren auf XML, wo gerade die Trennung von Inhalt und Präsentation als Grundkonzept verfolgt wird. Als Szenario für die Evaluierung dienen Lernmodule aus dem Bereich der Datenbanktechnik, derzeit vor allem aus dem Bereich der Relationen- und Normalformmentheorie.

Vorrangig vor der Erstellung der Entwicklungsumgebung hat eine Evaluation stattgefunden, um die Erstellung möglichst zielgerichtet und an die realen Anforderungen anpassen zu können. Die nächsten Arbeiten umfassen eine weitere Evaluierung an Hand einer umfassenden Fallstudie, wofür aus Gründen der Akzeptanz der Nutzer zu diesem Zweck schon Teile der Entwicklungsumgebung verfügbar sein müssen, sowie eine erste prototypische Realisierung der Entwicklungsumgebung zur Erstellung wiederverwendbarer Lerninhalte.

Literatur:

- [1] D.Boles: Erstellung multimedialer Dokumente und Anwendungen: Verfahren und Werkzeuge. Workshop Software-Engineering für Multimedia-Systeme im Rahmen der GI '97-Jahrestagung, 1997.
- [2] Virtueller Hochschulverbund Karlsruhe (ViKar), <http://vikar.ira.uka.de/>
- [3] ARIADNE, <http://ariadne.unil.ch/>
- [4] EDUCAUSE IMS, <http://www.imsproject.org>
- [5] W3C: Extensible Markup Language (XML), <http://www.xml.com/axml/axml.html>
- [6] IBM Learning Space; <http://www.lotus.com/home.nsf/welcome/learnspace>
- [7] Oracle Learning Architecture; <http://ola.us.oracle.com>
- [8] WebCT; <http://www.webct.com/>
- [9] GENTLE, <http://wbt-2.iicm.edu/>
- [10] HYPERWAVE, <http://www.hyperwave.de/>
- [11] S. Lewis: The art and Science of SmallTalk; Prentice Hall, 1995.
- [12] D. Geary: Graphic Java 2, Mastering the JFC; Prentice Hall
- [13] E. Wilde: Wilde's WWW, technical foundations of the world wide web; Springer 1999.
- [14] Garzotto, F., Paolini, P., and Schwabe, D.: HDM A model-based approach to Hypermedia application design. ACM Transactions on Information Systems, 11, 1 (Jan. 1993), 1-23.
- [15] F. Isakowitz. A. Stohr, P. Balasubramanian; RMM: A Methodology for Structured Hypermedia Design. Communications of the ACM, August 1995, Vol. 38, No. 8.
- [16] D. Schwabe, C. Rossi: The Object-Oriented Hypermedia Design Model. Communications of the ACM, August 1995, Vol. 38, No. 8.
- [17] Erik Duval: An Open Infrastructure for Learning – the ARIADNE project. <http://www.cs.kuleuven.ac.be/cwis/departement/personeel/>
- [18] Stucky, W. et al.: Modularer Aufbau von Hochschulübergreifenden Lerninhalten im Bereich Informationssysteme. Interner Projektbericht, VIKAR, Teilprojekt 2.3, Oktober 1999
- [19] Khaldoun Ateyeh: Generic Graphical User Interfaces for ODMG Object Databases, Diplomarbeit, Universität Karlsruhe und University of Manchester, November 1998.