Textverarbeitung und XDiML

Elektronisches Publizieren von Dissertationen und Habilitationsschriften – Warum wir beim elektronischen Publizieren die Mitarbeit der Autoren brauchen

Sabine Henneberger
shenneberger@cms.hu-berlin.de

An der Humboldt-Universität beschränkt sich das elektronische Publizieren von Dissertationen und Habilitationsschriften nicht auf die Veröffentlichung von PDF-Dokumenten im Internet.

Oft stellen Autoren, aber auch Betreiber von Dokumentenservern, die Frage, warum das so sei. Darauf möchte dieser Artikel eine Antwort geben.

Er beschreibt, wie auf der Grundlage von XML ein Standard angewendet wird, der im Hinblick auf Archivierung und Informationsretrieval wesentlich höheren Qualitätsansprüchen genügt als das bloße Speichern von PDF-Dokumenten. Die Methode zur Erzeugung des Langzeitarchivierungsformats XDiML wird kurz und anschaulich erläutert, so dass deutlich wird, warum die Mitarbeit der Autoren unverzichtbar ist.

Was ist XDiML?

Um zu verstehen, was XDiML ist, muss man zunächst etwas weiter ausholen und sich damit vertraut machen, was unter XML zu verstehen ist. XML steht für *Extended Markup Language* und ist, sehr allgemein ausgedrückt, ein Standard zur Strukturierung von Text [8].

Jedem Leser mehr oder weniger vertraut oder zumindest als Begriff bekannt ist HTML, ebenfalls eine *Markup Language* speziell zur Beschreibung von Webseiten, aber eben nicht *Extended* ... und das heißt hier, weniger strengen Regeln unterworfen. Ein Absatz kann sowohl in HTML als auch in XML so aussehen:

```
<p>Das ist ein Beispiel.</p>
```

XML ist ein Standard, der die Einhaltung wesentlich strengerer Regeln verlangt, z. B. dass jedes Element abgeschlossen werden muss. Etwa nur `<br>` als Zeilenumbruch ist so nicht erlaubt, dagegen `<br/>` schon.

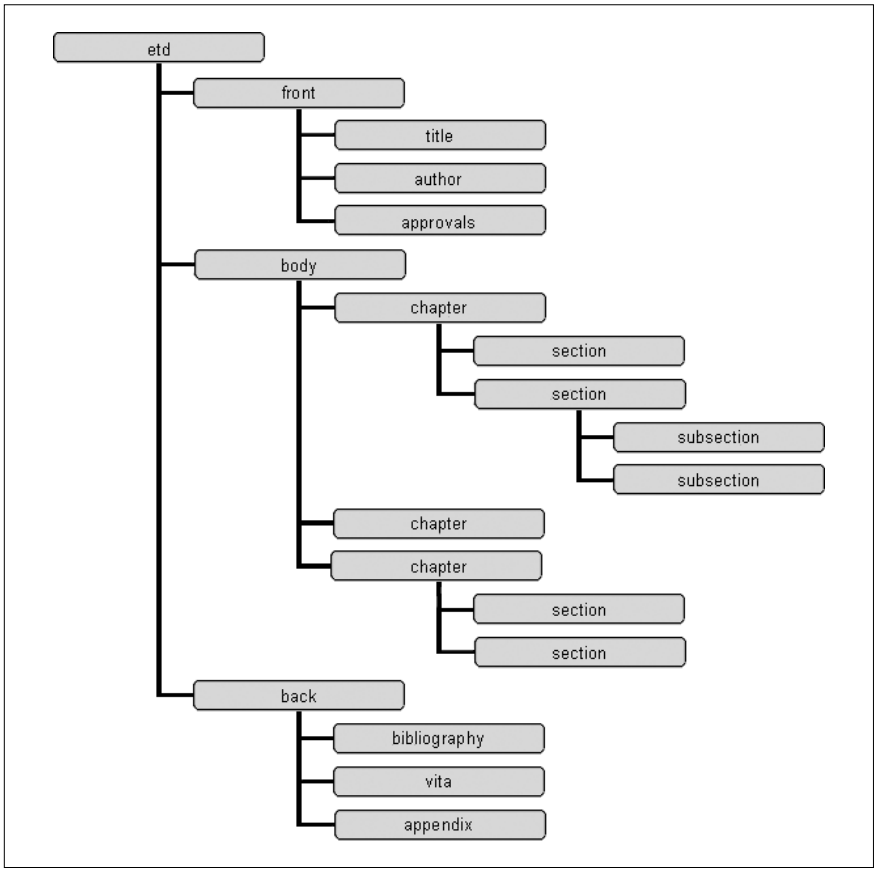


Abb. 1: Baumstruktur eines XDiML-Dokuments.

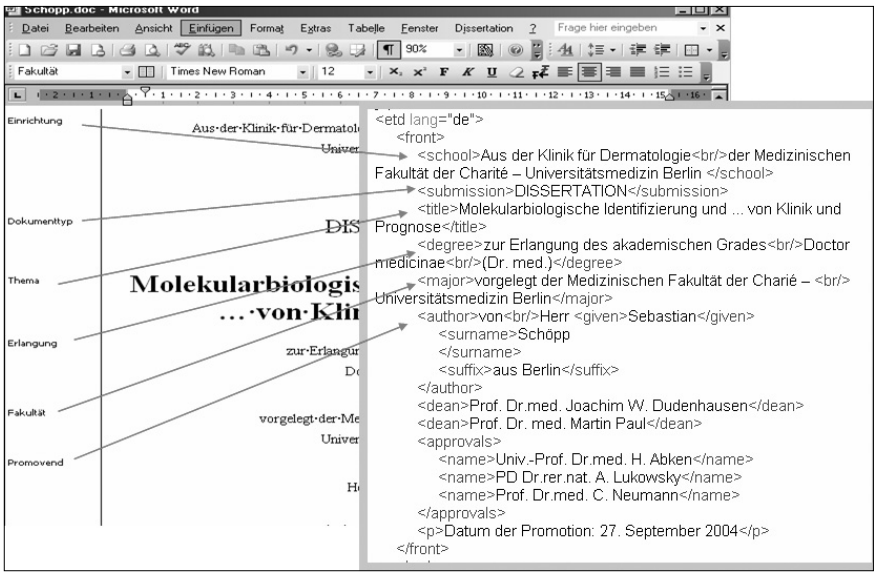


Abb. 2: Zuordnung der Formatvorlagen zu XDiML-Elementen.

Das `p` im Beispiel wird als Element bezeichnet. Ein Element kann aber auch noch Attribute besitzen. Viele kennen wieder aus HTML das Element `img` mit den Attributen `src`, `width` und `height`, welches so aussehen könnte: `<img src="Bild.jpg" width=100 height=200>`, zur Einbindung einer Grafik in die Webseite. In XML müsste das so aussehen: `<img src="Bild.jpg" width="100" height="200"/>`. Die Werte der Attribute müssen in Anführungszeichen eingeschlossen sein, der Schrägstrich am Ende bedeutet den Abschluss des Elementes, welches keinen eigentlichen Inhalt hat.

Elemente, Attribute und Regeln für deren Verwendung müssen an den Typ des Dokumentes angepasst werden. Wenn wir z. B. den Dokumenttyp *Einkaufsliste* betrachten, würden hier wenige Elemente, Attribute und Regeln ausreichen:

```
<liste tag="Montag">
<essen>Brot</essen>
<essen>Butter</essen>
<putzen>Waschpulver</putzen>
<kosmetik>Seife</kosmetik>
<kosmetik>Zahnpasta</kosmetik>
<kosmetik>Haarspray</kosmetik>
</liste>
```

Welche Elemente und Attribute in der XML-Struktur enthalten sein dürfen oder müssen und welchen Regeln die Struktur genügen muss, wird in einer *Document Type Definition* (DTD) oder einem so genannten XML-Schema beschrieben, welche oder welches auch wiederum Bestandteil des XML-Dokumentes ist. Ein weit verbreiteter Dokumenttyp ist *DocBook*, ein Typ, der vor allem zur Strukturierung von Dokumentationen jeder Art verwendet werden kann. DocBook braucht immerhin schon mehr als 300 Elemente, um die für eine Dokumentation wichtigen Merkmale beschreiben zu können [7].

Dissertationen und Habilitationsschriften sind allerdings noch etwas komplizierter aufgebaut. Zur Strukturierung dieser benötigt man wesentlich mehr Elemente und Attribute und damit natürlich eine Vielzahl von Regeln. Diese sind in einer speziellen DTD, der XDiML-DTD, beschrieben.

Nachzulesen ist dies alles unter *http://edoc.hu-berlin.de/diml/* [3]. Zum besseren Verständnis, wie das XDiML-Dokument einer Dissertation aufgebaut sein könnte, sehe man sich die schematische Übersicht in Abb. 1 an.

Die Baumstruktur ist deutlich erkennbar, wobei die Textinhalte, die *Blätter* des Baumes, hier nicht dargestellt sind.

Kann man einen Autor aber überzeugen, seinen Text mühsam in solch einer Struktur unterzubringen? Sicher nicht, da es so viele verschiedene Textverarbeitungssysteme gibt, in denen das Schreiben wesentlich komfortabler ist.

Gerade die Vielfalt macht ein Problem deutlich: Für welche der heute mit diesen Textverarbeitungssystemen erstellten Dokumente wird es auf Grund ihrer komplizierten Formate in 50 Jahren überhaupt möglich sein, die Inhalte sichtbar zu machen? Ein Textfile aber, so einfach strukturiert wie oben zu sehen, kann dann mit Sicherheit noch geöffnet und gelesen werden.

Ein anderer Aspekt ist die Möglichkeit der Suche nach dem Inhalt bestimmter Elemente, also eines komfortablen Informationsretrievals. In einem PDF-Dokument kann man mit einer Volltextsuche einen Begriff auffinden, man kann aber nicht gezielt in bestimmten Textbestandteilen suchen, z. B. den Begriff »Berliner Miniaturschwein« in einer Bildunterschrift, um eben ein Bild eines solchen aufzufinden. In einem XDiML-Dokument kann man das [2].

Dazu kommt, dass sich seit der Etablierung des XML-Standards 1998 XML zu einem universell austauschbaren Format für die Textverarbeitung entwickelt hat. So verwenden StarOffice und OpenOffice XML als internes Format für die Datenspeicherung und Microsoft Word ist ab Version 2003 in der Lage, XML-Dokumente zu im- und exportieren [5, 6]. Wir kommen später auf diese Textverarbeitungssysteme zurück.

Man sieht: Ein XML- und damit auch ein XDiML-Dokument ist in mehrfacher Hinsicht einem Textdokument in proprietärem Format überlegen.

Wie erzeugt man ein XDiML-Dokument?

Man kann ein XDiML-Dokument in einem XML-Editor schreiben. Von solchen gibt es inzwischen eine große Auswahl. Stellvertretend sollen hier als Beispiele EMax, XMetal, XMLSpy oder <oxygen/> genannt werden. Die Preisspanne reicht von Freeware bis teuer.

In diesen Editoren wird das Schreiben von Dokumenten schon sehr gut unterstützt. Es sind allerdings keine WYSIWYG(*what you see is what you get*)-Editoren, und wer möchte schon ständig seine Phantasie aktivieren, um zu entscheiden, ob ein Wort, welches im Element `<em>` eingeschlossen ist, also kursiv erscheinen soll, an dieser Stelle auch die entsprechende Wirkung entfaltet?

StarOffice, OpenOffice und Word 2003 in der Professional Edition z. B. bieten bereits ebenfalls an, XML-Dokumente zu editieren, aber auch das wird wohl nur wenige Autoren veranlassen können, ihre Dissertation auf diesem doch recht unbequemen Weg zu schreiben. Außerdem kommt kein Autor um ein ansprechendes Layout herum, soll die Arbeit doch druckbar und gut lesbar sein.

Einen Ausweg bieten hier z. B. Microsoft Word 2003, StarOffice 7 und FrameMaker: Man kann das Dokument wie gewohnt schreiben und dann nach XML exportieren, entweder in ein vom Programm vorgegebenes XML-Standard-Format oder unter Verwendung eines passenden Filters in ein beliebiges XML-Format.

Wie aber wird gewährleistet, dass z. B. eine Textzeile, die den Titel einer Dissertation enthält, in ein Element `<title>` überführt wird? Der Leser einer Dissertation erkennt die strukturelle Funktion dieser Textzeile und kann sie als Titel identifizieren, der Computer benötigt ein maschinenlesbares Merkmal (siehe dazu auch [1]).

Die Lösung des Problems sind Formatvorlagen, die, in einer Dokumentvorlage zusammengefasst, die Markierung der Textbestandteile entsprechend ihrer strukturellen Funktion ermöglichen. In Abb. 2 sehen wir im Hintergrund das Word-Dokument einer Dissertation in der Normalansicht. In der linken Spalte sind die Formatvorlagen der Absätze angezeigt. Im Vordergrund ist das »Zieldokument«, also das XML-Dokument, mit den zu erzeugenden und zuzuordnenden Elementen zu sehen.

Anhand dieser Markierungen ist es nun möglich, die Textbestandteile in die richtigen XDiML-Elemente zu überführen, vorausgesetzt, der Autor hat die Formatvorlagen richtig zugeordnet [5]. Aber dazu später.

Hat erst einmal jeder Textbestandteil die richtige Formatvorlage zugewiesen bekommen, kann mit Hilfe des schon erwähnten Filters in Form eines abzuarbeitenden Programmes aus dem XML-Format des Textverarbeitungsprogrammes (bei Microsoft Word 2003 spricht man von WordprocessingML) ein anderes, also auch ein XDiML-Format, erzeugt werden. Als Filter wird ein XSLT-Programm verwendet.

Abb. 3: Transformation von WordprocessingML nach XDiML.

XSLT steht für *Extensible Stylesheet Transformation Language*, eine Programmiersprache, die dafür entwickelt wurde, eine XML-Struktur in eine andere XML-Struktur oder ein anderes Format zu überführen. Dieses Programm selbst ist übrigens auch wieder ein XML-Dokument.

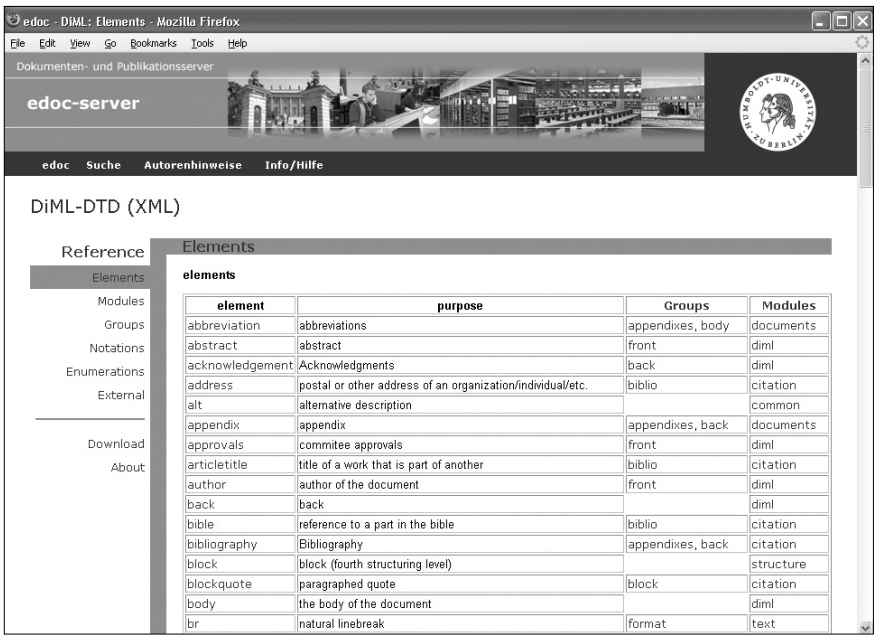
Elektronisches Publizieren an der Humboldt-Universität und XDiML

Für die Arbeitsgruppe *Elektronisches Publizieren* besteht die elektronische Veröffentlichung einer Dissertation oder Habilitationsschrift nicht wie in der überwiegenden Zahl der Institutionen, die dieses tun, in der ausschließlichen Publikation eines PDF-Dokumentes im Internet.

Die Lesbarkeit eines PDF-Dokumentes ist von der dazu passenden Visualisierungssoftware abhängig, die heute Acrobat Reader oder Adobe Reader heißt. Niemand kann vorhersehen, wie sich die Systeme in den nächsten Jahren entwickeln werden und ob es später Software geben wird, die mit den heute erstellten PDF-Dokumenten irgend etwas anfangen kann. Liegt dagegen ein Dokument in Form von XML vor, ist es jederzeit möglich, die Informationen daraus für ein im Moment völlig unbekanntes System aufzubereiten und somit auch lesen zu können.

Ein weiteres Ziel ist es, ein elektronisches Dokument nicht nur zu speichern und sichtbar zu machen, sondern auf Grundlage der Möglichkeiten, die sich aus der Verwendung von XML als Speicherformat ergeben, Langzeitarchivierung, Informationsretrieval und weltweite Verbreitung auf höchstem Niveau anzubieten [4].

Um die Erzeugung eines XDiML-Dokumentes zu ermöglichen, ist die Mitarbeit der Autoren erforderlich. Die meisten Autoren verwenden Microsoft Word als Textverarbeitungssystem. Wie oben beschrieben, ist die Markierung der Textabsätze mit vorgegebenen Formatvorlagen, die in einer von uns entwickelten Dokumentvorlage enthalten sind, die Grundlage für die Konvertierung nach



element	purpose	Groups	Modules
abbreviation	abbreviations	appendixes, body	documents
abstract	abstract	front	diml
acknowledgement	Acknowledgments	back	diml
address	postal or other address of an organization/individual/etc.	biblio	citation
alt	alternative description		common
appendix	appendix	appendixes, back	documents
approvals	commitee approvals	front	diml
articletitle	title of a work that is part of another	biblio	citation
author	author of the document	front	diml
back	back		diml
bible	reference to a part in the bible	biblio	citation
bibliography	Bibliography	appendixes, back	citation
block	block (fourth structuring level)		structure
blockquote	paragraphed quote	block	citation
body	the body of the document		diml
br	natural linebreak	format	text

Abb. 4: Online-Referenz der Dissertation Markup Language *(XDiML)* [3].

XDiML. Diese Dokumentvorlage, im übrigen auch eine Dokumentvorlage für diejenigen, die mit StarOffice oder OpenOffice arbeiten, und alle Informationen, die es rund um das elektronische Publizieren an der Humboldt-Universität gibt, findet man auf unserer Webseite *http://edoc.hu-berlin.de/* unter *»Unser Service für Autoren«*. Daneben bieten wir telefonischen Support, persönliche Beratungen und einen monatlich stattfindenden Kurs an.

Autoren, die ihrer Publikationspflicht durch eine elektronische Publikation nachkommen möchten, können ihre Dokumente sehr bequem über unsere Upload-Seite zu uns übertragen. Die Mitarbeiter werden innerhalb weniger Arbeitstage überprüfen, ob alle Voraussetzungen für das elektronische Publizieren erfüllt sind, und sich gegebenenfalls mit den Autoren in Verbindung setzen, um noch vorhandene Unklarheiten auszuräumen. Ist alles korrekt bearbeitet, erhalten sowohl Autor als auch die Hochschulschriftenstelle der Universitätsbibliothek eine Information darüber und die Publikationsbescheinigung kann abgeholt werden. Für uns ist allerdings der Vorgang noch nicht beendet, sondern der wichtige Teil, nämlich die Konvertierung nach XDiML, kann jetzt beginnen.

Literatur

[1] Becker, O.: Serielle Transformationen von XML. Dissertation, Berlin 2004.

[2] Dobratz, S.: Elektronisches Publizieren – Etablierung eines neuen Service für die Universität durch UB und CMS. *cms-journal*, Sonderheft Nr. 1, 2004.

[3] Dokumenten- und Publikationsserver der Humboldt-Universität, XDiML DTD, http://edoc.hu-berlin.de/diml

[4] Dokumenten- und Publikationsserver der Humboldt-Universität, Leitlinien, http://edoc.hu-berlin.de/e_info/leitlinien.php

[5] Henneberger, S., Schulz, M.: Save as XDiML II- The Package for Electronic Theses, Vortrag auf der OpenOffice.org – Konferenz 2004, Berlin (OOoCon2004).

[6] Lenz, E., McRae, M., St. Laurent, S.: *Office 2003 XML*. O'Reilly, 2004.

[7] Megginson, D.: *Structuring XML Documents*. Prentice Hall PTR, 1998.

[8] W3C – World Wide Web Consortium, http://www.w3.org/