

Die Website des Master-Studiengangs Bibliotheks- und Medienmanagement - Technische und inhaltliche Neukonzeption und Realisierung

Masterarbeit

im Modul „Informationsadministration“

Master-Studiengang Bibliotheks- und Medienmanagement
der Fachhochschule Stuttgart -
Hochschule der Medien

Ulrich Wesser

Erstprüfer: Prof. Bernhard Hütter

Zweitprüfer: Prof. Dr. Wolf-Fritz Riekert

Bearbeitungszeitraum: 01.10.2003 bis 17.02.2004

Stuttgart, Februar 2004

Kurzfassung

Diese Arbeit beschreibt die technische und inhaltliche Neukonzeption der Website des Master-Studiengangs Bibliotheks- und Medienmanagement im Fachbereich Information und Kommunikation der Fachhochschule Stuttgart - Hochschule der Medien. Da zur Verwirklichung der Anforderungen ein Web-Publishing-System eingesetzt werden soll, werden an der FH vorhandene Systeme auf ihre Tauglichkeit für diesen Zweck geprüft und kurz bewertet. Ausgewählt wird die Web-Application-Server-Software Zope in der Version 2.5.1. Ausführlich beschrieben wird die Konkretisierung und technische Umsetzung der Anforderungen mit Hilfe zahlreicher vom Verfasser erstellter oder von dritter Seite bezogener und durch den Verfasser angepasster Programmmodule für die OpenSource-Software Zope.

Schlagerworte: Website, Master-Studiengang Bibliotheks- und Medienmanagement, Intranet, Content-management, Web-Application-Server, Open Source Software, Zope.

Abstract

This master thesis describes the reorganization of the website of the Master-Course Library- and Media Management at the Hochschule der Medien - University of applied science, department of information and communication. This reorganization is concerning not only the content of the website but also its technical basis. The new website requires the use of a web-publishing-system. Therefore a research of such systems that were available at the Hochschule der Medien was carried out. The web-application-server-software Zope version 2.5.1. appeared to be the best choice. The thesis ends with a detailed description of the interactive functions in the intranet. In many cases the installation of those required a special adaption of some free available software-modules, called "products", for the open source Software Zope.

Keywords: website, Master-course library and media management, intranet, content management, web application server, open source software, Zope.

Vorwort

Bei der Konzeption einer Website - sei es eine vollständig neue Produktion oder auch die Überarbeitung eines bestehenden Angebotes - ist eine ausführliche Abstimmung mit dem Auftraggeber über Aussage, Zielgruppe und Funktion unabdingbar. Nur durch eine umfassende und weitreichende Information kann die gewünschte Intention eines Web-Angebotes dargestellt und dem Nutzer vermittelt werden. Aus diesem Grund wurden ausführliche persönliche und telefonische Gespräche und zahlreiche Kontakte per E-Mail mit dem Studiengangsleiter des Master-Studiengangs Bibliotheks- und Medienmanagement und Betreuer dieser Arbeit, Herrn Prof. Bernhard Hütter, durchgeführt. Für diese Gespräche und die weiteren wertvollen Ratschläge und Anregungen möchte ich mich bei Herrn Prof. Hütter herzlich bedanken.

Ebenso gilt mein Dank Herrn Prof. Dr. Wolf-Fritz Riekert für die Bereitschaft, die Zweitkorrektur dieser Arbeit zu übernehmen.

Da die in dieser Arbeit konzipierte Website über einen Studiengang an der HdM mit den anderen Web-Angeboten der Hochschule harmonisieren und korrespondieren muss und die Schnittstellen der verschiedenen Angebote genau definiert werden mussten, war auch immer wieder die Kommunikation mit den anderen Betreuern von Web-Angeboten an der Hochschule notwendig. In diesem Zusammenhang möchte ich mich besonders bei meinen beiden Kollegen Herrn Florian Fahrbach und Herrn David Prüm sehr herzlich bedanken, mit denen derartige Gespräche, die immer von einer offenen, sach- und problemorientierten Grundhaltung geprägt waren, stets unkompliziert und problemlos möglich waren.

Stuttgart, im Februar 2004

Inhaltsverzeichnis

Kurzfassung	II
Abstract	II
Vorwort	III
Hinweise zur technischen Realisierung	VII
I. Einleitung	1
II. Ausgangssituation - das bisherige Webangebot	3
A. Internetbereich	4
1. Inhalte	4
2. Technik	4
B. Intranetbereich	5
1. Inhalte	5
2. Technik	6
III. Anforderungen an die neue Website	8
A. Internetbereich	9
1. Inhalte	9
2. Technik	9
B. Intranetbereich	10
1. Inhalte	10
2. Technik	12
IV. Mögliche Methoden und Werkzeuge des Website- und Content- Managements	14
A. Individuelle Programmierung	15
B. Low-Cost-, Freeware- und OpenSource-Systeme	16

C.	Einsatz und Adaption eines an der HdM vorhandenen Content-Management-Systems	18
1.	Livelihood / OpenText	18
2.	IBT-Server / Time4you	19
3.	Zope	20
D.	Auswahl eines Systems	21
V.	Zope - Technische Grundlagen	24
A.	Installation und Konfiguration	24
B.	Grundlegende Funktionsweise von Zope	25
1.	Klassen, Objekte und Methoden	26
2.	Produkte und ZClasses	29
3.	Benutzer- und Rechteverwaltung in Zope	31
VI.	Produktion der neuen Website	33
A.	Migration der bestehenden Inhalte	33
1.	Internetbereich	33
2.	Intranetbereich	34
B.	Konkretisierung der Anforderungen	34
1.	Internetbereich	35
a.	Newsletter - Abonnement	35
b.	Newsletter - Archivierung	36
2.	Intranetbereich	36
a.	Skriptearchiv	36
b.	Bereich für Arbeitsgruppen (Upload- / Download-Bereich) ...	37
c.	Upload-Bereich zur Abgabe von Prüfungsleistungen	37
d.	Studierendenprofile (inkl. Alumni)	38
e.	Dozentenprofile (inkl. Lehrbeauftragte)	39
f.	Interner Infodienst	40
C.	Technische Umsetzung der neuen Funktionen	40
1.	Internetbereich	40
a.	Newsletter - Abonnement	40

b. Newsletter - Archivierung	44
2. Intranetbereich	45
a. Skriptearchiv	54
b. Bereich für Arbeitsgruppen (Upload- / Download-Bereich) ...	58
c. Upload-Bereich zur Abgabe von Prüfungsleistungen	60
d. Studierendenprofile (inkl. Alumni)	62
e. Dozentenprofile (inkl. Lehrbeauftragte)	66
f. Interner Infodienst	68
VII. Ausblick	72
Literatur- und Quellenverzeichnis	74
Abbildungsverzeichnis	77
Glossar	78
Anhang	83
A. Verzeichnis der verwendeten Zope-Produkte und Quellcodes	83
B. Inhalt der beiliegenden CD-ROM	86
C. Erklärung	87

Hinweise zur technischen Realisierung

Sofern in dieser Arbeit Quelltext von Programmcode dargestellt wird, wird dieser durch die Verwendung der Schriftart `Courier New` gekennzeichnet. Wo es notwendig wird, längere URLs umzubrechen, wird der Zeilenumbruch ohne die Anzeige eines Trennstrichs oder eines Leerzeichens durchgeführt.

Die technische Umsetzung dieser Arbeit ist auf einem Web-Server des Fachbereichs Information und Kommunikation der Hochschule der Medien realisiert. Die URL lautet: `http://bmm2.iuk.hdm-stuttgart.de`. Der Zugang dazu ist bis zur abschließenden Bewertung der Arbeit auf authentifizierte Nutzer beschränkt.

Mögliche Nutzerkennungen sind:

Nutzername	Passwort	Kennung gehört zur Gruppe:
<code>korrektor</code>	<code>bmmonzope</code>	Korrektor
<code>student1</code>	<code>student1</code>	Student
<code>student2</code>	<code>student2</code>	Student
<code>dozent1</code>	<code>dozent1</code>	Dozent
<code>dozent2</code>	<code>dozent2</code>	Dozent
<code>huetter</code>	●●●●●	Manager (Administratoren)
<code>riekert</code>	●●●●●	Manager (Administratoren)

Die Nutzerkennung `korrektor` gewährt auch Zugriff auf die Management-Konsole von Zope. Damit können, neben dem Quellcode, auch die eingesetzten Produkte und die Systemeinstellungen von Zope betrachtet werden. Eine Änderung des Quellcodes ist jedoch nicht möglich, da diese Kennung im Systembereich nur lesenden Zugriff besitzt.

Wenn mit der Kennung `korrektor` die Zope-Management-Konsole aufgerufen werden soll, so darf dafür als Start-URL **nicht** die oben genannte BMM2-URL, sondern nur die URL des Stammverzeichnisses des Servers aufgerufen werden. Diese URL lautet `http://morpheus.iuk.hdm-stuttgart.de`. Die Management-Konsole von Zope wird aufgerufen, indem an die vollständige URL

die Zeichenfolge `/manage` angehängt wird. In diesem Fall heißt die entsprechende URL also: `http://morpheus.iuk.hdm-stuttgart.de/manage` .

Alle in der Arbeit erwähnten Pfadangaben von Dateien, Objekten oder Verzeichnissen gehen davon aus, dass die Management-Konsole über diese URL `http://morpheus.iuk.hdm-stuttgart.de/manage` aufgerufen wird.

Die Software Zope, alle verwendeten Produkte und das Zope-Datenfile `data.fs` können auch auf einem lokalen PC installiert und aufgerufen werden. Alle dazu notwendigen Programme und Daten befinden sich auf der beiliegenden CD. Eine genaue Anleitung für den Installationsvorgang ist ebenfalls auf der beiliegenden CD enthalten.

I. Einleitung

Der Master-Studiengang „Bibliotheks- und Medienmanagement“ startete im April 2001 an der damaligen Fachhochschule Stuttgart - Hochschule für Bibliotheks- und Informationswesen. Primäre Zielgruppe des Studiengangs waren damals und sind bis heute „Personen, die ursprünglich ein bibliothekarisches, dokumentarisches, informationswissenschaftliches oder ein sonstiges, auf Medien ausgerichtetes Erststudium abgeschlossen haben“¹ und die nach dem Abschluss dieses Erststudiums „möglichst bereits seit einigen Jahren im Informations- oder Medienbereich gearbeitet haben“². Aufgrund verschiedener Gründe stellt dieser Studiengang eine Besonderheit in der Hochschullandschaft dar. Neu war zum damaligen Zeitpunkt die Organisation als Teilzeitstudium. Der Unterrichtsstoff von zwei Vollzeitsemestern wird auf insgesamt vier Semester verteilt, so dass sich die regelmäßig zu erbringenden Anwesenheitszeiten halbieren. Dadurch soll die Aufnahme des Studiums auch neben einer bestehenden Berufstätigkeit oder auch noch während einer Familienpause ermöglicht werden. Diese Teilzeitstruktur mit reduzierten, zeitlich geblockten Anwesenheitszeiten an der Hochschule ermöglicht auch weiter entfernt wohnenden Interessenten die Teilnahme am Studium. „Der Einzugsbereich erstreckt sich derzeit von Hamburg im Norden bis Bozen/Südtirol im Süden“³.

Angesichts dieser Besonderheiten ist es verständlich, dass konventionelle Informationswege für Studierende dieses Studiengangs nur sehr eingeschränkt nutzbar sind. Das Durchlesen der Aushänge am schwarzen Brett oder ein kurzer klärender Besuch im Studierendensekretariat ist oft mangels Anwesenheit an der Hochschule über die Dauer von mehreren Wochen nicht möglich. Auch wurde diese ungewohnte Studiengangsstruktur nur unzureichend von den Arbeitsabläufen in der Studierendenverwaltung berücksichtigt: auf Termine wurde zu kurzfristig hingewiesen, die persönliche Anwesenheit der Studierenden (z.B. für die Rückmeldung) war zwingend notwendig - was von den Betroffenen teilweise erhebliche

¹ Hütter, Bernhard: Aktuelles zum Teilzeit-Masterstudiengang »Bibliotheks- und Medienmanagement«, S. 609.

² Ebd.

³ Ebd.

zeitliche und/oder finanzielle Aufwendungen erforderte. Zwar bemühten sich sowohl Verwaltungs- als auch Lehrpersonal, dieser speziellen Situation der Studierenden des Master-Studiengangs gerecht zu werden, indem z.B. die E-Mail als primäres Kommunikationsmittel etabliert wurde, jedoch unterliefen den Beteiligten immer wieder Versäumnisse, die zu Zwischenfällen oder Missverständnissen führten⁴. Auch aus diesem Grund wurden in die geplante Realisierung der Internetseite des Studiengangs, als eines orts- und zeitunabhängig nutzbaren Mediums, große Erwartungen hinsichtlich der Informationsversorgung gesetzt. Um dieser besonderen Situation gerecht zu werden, ist allerdings ein klassischer Internetauftritt, der nur rein rezipierend genutzt werden kann, nur bedingt geeignet. Neben der Lieferung aktueller relevanter Informationen ist es für den Studienbetrieb auch wichtig, ein Arbeitsmittel zur Verfügung gestellt zu bekommen, das tiefergehende Informationen, z.B. in Form eines Material-Archives, anbietet, das die Zusammenarbeit von räumlich entfernten Arbeitsgruppen unterstützt, das hochwertige, nicht allgemein verfügbare, eventuell sogar schützenswerte Daten zur Verwendung durch einen geschlossenen Benutzerkreis bereit hält und das möglichst auch die Erledigung von notwendigen formalen Tätigkeiten im Studienbetrieb virtualisieren kann.

Somit kommt vor allem dem Intranet - dem geschlossenen zugriffsgeschützten Bereich exklusiv für Studierende und Lehrende dieses Studiengangs - bei der Neukonzeption der Website für den Master-Studiengang Bibliotheks- und Medienmanagement eine besondere Bedeutung zu.

⁴ Nach: Boeckh, Dorothee; Beier, Nina; Schäfer, Christina: Alles Master - oder was?, S. 850 - 851.

II. Ausgangssituation - das bisherige Webangebot

Die bis heute verwendete und gepflegte Website des Master-Studiengangs Bibliotheks- und Medienmanagement⁵, die durch die in dieser Arbeit konzipierte neue Website ersetzt werden soll, wurde im Sommersemester 2001 von den Teilnehmern des Seminars "Internet und Intranet" des Moduls Informationsadministration geplant und realisiert. Zum einen konnte dadurch von den zahlreichen Erfahrungen profitiert werden, über die mehrere Studierende als Administratoren von Websites oder Webservern bereits verfügten, zum anderen sollten die Erfahrungen der Studierenden aus deren Studienalltag und die daraus resultierenden Informationsbedürfnisse gleich in Konzeption und Inhalt der Website mit einfließen.

Die komplette Website besteht überwiegend aus konventionellem HTML-Code. Nur vereinzelt wurden kleinere JavaScript-Programmteile eingebaut (z.B. für Roll-over-Effekte bei Abbildungen oder bei Pop-Up-Fenstern) oder vorgefertigte aus einer öffentlich zugänglichen CGI-Quelltext-Bibliothek bezogene Perl-Script-Module (z.B. zum Erstellen von Diskussionsforen) eingesetzt.

Aufgrund der flexiblen Gestaltungsmöglichkeiten bezüglich Layout und Bildschirmaufteilung wurde die komplette Website durch den Einsatz einer Frame-Struktur realisiert. Die gestalterischen Vorzüge einer durch Frames möglichen Umrahmung der Seiteninhalte (z.B. mit dauerhaft sichtbaren Navigationselementen links oder mit einem auch beim Scrollen der Seiteninhalte permanent sichtbaren Kopf- bzw. Fußbereich) wurden von den Seminarteilnehmern für so wichtig erachtet, dass die bekannten typischen Nachteile von Frame-basierten Layouts (z.B. die nicht vorhandene Möglichkeit einer Verlinkung von untergeordneten Seiten, die tiefer in der Site-Struktur abgelegt sind, oder die differierende Darstellung bei unterschiedlichen Bildschirmauflösungen) bewusst in Kauf genommen und toleriert wurden⁶.

⁵ Erreichbar unter <http://bmm.iuk.hdm-stuttgart.de>

⁶ Die Vor- und Nachteile einer Frame-basierten Website wurden während des Seminartermins am 21.06.2001 von den Seminarteilnehmern erörtert.

Auf der grafisch aufwändig gestalteten Eingangsseite erscheinen drei Menüpunkte: “Lehrangebot”, “Organisation” und “Intranet”. Während der Bereich “Intranet” passwortgeschützt und ausschließlich nur für Studierende und Lehrende des Studiengangs zugänglich ist, sind die Bereiche “Lehrangebot” und “Organisation” frei zugänglich.

A. Internetbereich

1. Inhalte

Mit der Unterteilung des öffentlich zugänglichen Internetbereiches in die beiden Unterpunkte “Lehrangebot” und “Organisation” sollte eine klare Trennung zwischen den eigentlichen fachlichen Inhalten des Studiums und den sonstigen Rahmenbedingungen erzielt werden. Während im Bereich “Lehrangebot” Informationen über Studieninhalte, Module, Seminare, zu erbringende Prüfungsleistungen und Dozenten untergebracht sind, finden sich im Bereich Organisation die jeweils aktuellen Terminpläne über Seminar- und Modultermine sowie Verlinkungen auf einige wichtige Formulare. Daneben wurden noch zahlreiche Links zu speziellen externen Informationsquellen aufgeführt, wie z.B. zu Mitwohnzentralen, Mitfahrzentralen oder zu Fahrplänen des öffentlichen Personennahverkehrs. Die Zusammensetzung dieser Links wurde speziell auf die Bedürfnisse und die besondere Situation der Teilzeitstudierenden des Studiengangs ausgerichtet. Trotz dieser klaren inhaltlichen Ausrichtung wurde aber auf eine Übernahme dieser Informationen in den geschützten Intranetbereich verzichtet, da es als wünschenswert und nützlich angesehen wurde, so auch potenziellen Studieninteressierten einen Eindruck über Eigenheiten und Schwierigkeiten eines Teilzeitstudium zu ermöglichen.

2. Technik

Die technische Umsetzung der Seiten des Internetbereiches wurde nahezu vollständig durch reines HTML ausgeführt. Grundlegende Gestal-

tungsmerkmale wurden durch ein zentrales Cascading Style Sheet realisiert. Pflege und Administration der Inhalte wurden ausschließlich von einem Webmaster durchgeführt, meist auf Veranlassung des Studiengangsleiters. Somit mussten auch anlässlich einer größeren Umstrukturierung nach dem Sommersemester 2003 (Umstrukturierung bzw. Wegfall von Modulen, Integration des Musikbibliothekarischen Aufbaustudiums in den Master-Studiengang) alle anstehenden Änderungen und Neuerungen unmittelbar im HTML-Quellcode vorgenommen werden.

B. Intranetbereich

1. Inhalte

Der zugangsbeschränkte Intranetbereich wurde von den Seminarteilnehmern konsequent auf die wichtigsten Informationsbedürfnisse der Studierenden ausgerichtet. Hierzu gehörten:

- Kontaktangaben zu allen Studierenden (Name, E-Mail-Adressen dienstlich und privat, Telefon- und Faxnummern, gewählte Module), von allen Dozenten und möglichst auch allen Lehrbeauftragten
- Gesammelte Lehrmaterialien (Skripte, Folien, Links, Software) zu allen Lehrveranstaltungen der einzelnen Module
- Spezifizierung der Prüfungsanforderungen
- Zusammenstellung wichtiger Termine
- Diskussionsforen

Längere Zeit nach Abschluss des Seminars und nach Inbetriebnahme der Website wurden noch zwei weitere Themenbereiche ergänzt. Dabei handelte es sich um die Einbindung von Themenvorschlägen der Professoren für mögliche Masterarbeiten sowie um einen für alle Studierenden und Lehrenden gleichermaßen zugänglichen Bereich, der den Upload und Download von Dateien ermöglichen sollte. Ausschlaggebend für die Einrichtung dieser Option waren die positiven Erfahrungen im Seminar "Datenbanken", in dem mit Hilfe einer beim Web-Dienstleister Yahoo eingerichteten E-Group dem Dozenten und den anderen Seminarteilneh-

mern Arbeitsergebnisse zur Verfügung gestellt wurden. Durch diesen Up-/Downloadbereich sollte unabhängig von Drittanbietern die Möglichkeit für eine kontinuierliche Zusammenarbeit von Arbeitsgruppen - gerade auch über längere Abwesenheitsphasen hinweg - geschaffen werden.

2. Technik

Der Zugriff auf das Intranet war nur durch die Eingabe eines Nutzernamens und eines Passwortes möglich. Diese Angaben wurden unabhängig vom Standort des aufrufenden PC (innerhalb oder außerhalb der Hochschule) abgefragt. Durch die Anbindung des Webservers an die WindowsNT-Domäne des Fachbereichs konnten die eingetragenen NT-Domänen-User zur Authentifizierung herangezogen werden, womit die Einrichtung und Pflege einer eigenen Benutzerverwaltung umgangen werden konnte.

Bei der Realisierung des Intranetbereiches wurde konsequent auf eine möglichst unkomplizierte und direkte Bedienung Wert gelegt, Design-Elemente wurden nur sehr sparsam eingesetzt. Die Menüstruktur sollte möglichst klar und schnell erfassbar, die Benennung der Menüpunkte möglichst eindeutig sein. Die gewünschten Ziele sollten möglichst schnell mit möglichst wenigen Aktionen erreicht werden. Aus diesem Grund wurde der Aufruf der Modul-Materialien durch ein Pull-down-Menü realisiert. Das Einstellen neuer Materialien oder das Ändern und Ergänzen der sonstigen Inhalte war nur durch den Administrator der Website möglich.

Das Layout des Kontakte-Bereichs wurde mittels konventioneller HTML-Tabellen realisiert. Auch in diesem Bereich mussten die häufig auftretenden Änderungswünsche dem Webmaster persönlich oder per E-Mail mitgeteilt werden.

Die CGI-Skripte zur Nutzung und Administration der Diskussionsforen wurden einem frei zugänglichen Archiv für CGI-Skripte⁷ entnommen und auf die lokale Umgebung und das Layout der Studiengangs-Website abgestimmt. Für jedes Modul wurde ein eigenes Diskussionsforum eingerichtet.

Der Up-/Downloadbereich wurde durch einen Link realisiert, der nicht per http, sondern per FTP auf ein Verzeichnis des Web-Servers verwies. Wurde dieser Link mit dem Internet Explorer aufgerufen, so war der Nutzer in der Lage, per drag-and-drop Dateien in dieses Verzeichnis zu laden. Die Rechte innerhalb dieses Verzeichnisses waren so ausgelegt, dass grundsätzlich alle Nutzer mit Zugang zum Intranetbereich in der Lage waren, Dateien einzustellen und zu lesen. Das Recht zum Löschen einer Datei hatte jedoch lediglich der Inhaber der Datei - also die Person, die die betreffende Datei auch in den Up-/Downloadbereich eingestellt hatte. Ein eigener Login-Vorgang wurde für den FTP-Bereich nicht realisiert. Grundsätzlich wurde bei der Einrichtung dieses Bereiches davon ausgegangen, dass alle zugangsberechtigten Intranet-Nutzer auch Schreibrechte auf dieses FTP-Verzeichnis bekommen sollten.

⁷ Es handelt sich hierbei um "Matt's Script Archive", administriert und aktualisiert von Matt Wright, erreichbar unter der URL: <http://worldwidemart.com/scripts/>.

III. Anforderungen an die neue Website

Um noch von den Erfahrungen der Studierenden des Jahrgangs 2001/2003 profitieren zu können, hat der Verfasser gegen Ende des Sommersemesters 2003 durch eine Mail alle Studierenden des Master-Studiengangs gebeten, ihm Änderungswünsche und Korrekturvorschläge für die Neukonzeption der Website des Studiengangs mitzuteilen. Dabei wurde explizit darauf hingewiesen, dass alle Themenbereiche (Inhalt, Layout, Technik) berücksichtigt werden sollten. Zwar kamen hierauf nur zwei konkrete Rückmeldungen, doch war zumindest einer der Berichte das Ergebnis einer ausführlichen Diskussion innerhalb einer Seminargruppe, so dass letztendlich doch von der Mitwirkung einer größeren Anzahl von Studierenden ausgegangen werden kann⁸. Daneben wurden zwischen dem Studiengangsleiter des Master-Studiengangs, Herrn Prof. Bernhard Hütter, und dem Verfasser mehrere persönliche Gespräche geführt, in denen auch diese studentischen Anmerkungen besprochen wurden⁹. Die in diesem Kapitel aufgeführten Vorschläge nehmen vor allem Bezug auf diese beiden Rückmeldungen durch die Studierenden und auf die Gespräche des Verfassers mit dem Studiengangsleiter.

In keiner dieser mündlichen oder schriftlichen Rückmeldungen wurden Verbesserungsvorschläge hinsichtlich Design oder Layout der Website gefordert. Aus diesem Grund wurden keine grundlegenden Änderungen in diesen Bereichen geplant. Lediglich dort, wo es in Folge weit reichender inhaltlicher oder struktureller Änderungen notwendig wurde, wurde eine Anpassung des Layouts oder Designs vorgenommen.

⁸ Es handelt sich hierbei um eine schriftliche Rückmeldung von Frau Dorothee Boeckh vom 10.10.2003 und um die Flipchart-Aufschriebe einer Seminargruppe, die von Herrn Thomas Becker protokolliert wurden.

⁹ Die Gespräche zwischen dem Verfasser und dem Studiengangsleiter des Master-Studiengangs Herrn Prof. Bernhard Hütter fanden statt am 13.08.2003 und am 16.12.2003.

A. Internetbereich

1. Inhalte

Für den öffentlich zugänglichen Internetbereich wurden von Seite der Studiengangsleitung zwei Anforderungen formuliert:

- Studieninteressenten sollen die Möglichkeit bekommen, durch Eintragen Ihrer E-Mail-Adresse einen Newsletter abonnieren zu können, der, in regelmäßigen Abständen erscheinend, über aktuelle anstehende Veränderungen, Planungen und Tendenzen informiert. Das Abonnement dieses Newsletters und der Versand neuer Ausgaben durch den Studiengangsleiter soll direkt über die Website des Studiengangs möglich sein.
- Ältere Ausgaben dieses Newsletters sollen auf der Website in einem Archiv zum Download bereit gehalten werden. Ebenso wie der Versand soll auch das Einstellen der jeweils neuen Newsletter in die Archiv-Website möglichst unkompliziert und einfach möglich sein.

Die Integration dieser beiden Applikationen ist deshalb wichtig, weil sich durch zahlreiche Modifikationen in der Vergangenheit (Wegfall von Modulen, Umstrukturierungen der Module, Änderungen beim angestrebten akademischen Grad) und durch geplante zukünftige Änderungen (die angestrebte Kooperation mit der Hochschule für Technik und Wirtschaft Chur oder die Akkreditierung) Struktur und Profil des Master-Studiengangs stark verändert haben und zumindest in nächster Zeit auch immer wieder ändern werden. Über einen derartigen Infodienst können auch längerfristig planende Studieninteressenten stets aktuell über derartige Entwicklungen informiert werden.

2. Technik

Hinsichtlich der technischen Grundlage des Internetbereichs wurden keine Verbesserungsvorschläge genannt. Aufgrund der bisherigen Erfahrung des Verfassers als Webmaster mehrerer Internetseiten erscheint es sinnvoll, bei der Aktualisierung der Website zu prüfen, ob von dem Ein-

satz einer Frame-Struktur abgewichen werden kann, um so die bereits erwähnten Nachteile eines Frame-basierten Layouts zu vermeiden.

B. Intranetbereich

1. Inhalte

Die meisten Änderungs- und Korrekturwünsche betreffen das Intranet-Angebot des Studiengangs.

- Das Einstellen von Materialien in die Skriptesammlung soll zukünftig direkt von den Lehrenden vorgenommen werden können, um so die Verzugszeit bis zu Erreichbarkeit der Unterlagen im Netz möglichst gering zu halten.
- Der Upload-/Download-Bereich zum Austausch von Dateien soll erhalten bleiben, muss aber hinsichtlich seiner technischen Stabilität und seiner Benutzerfreundlichkeit verbessert werden. Die Zugriffsrechte sollen wie ursprünglich geplant realisiert werden: Jeder Studierende soll die Möglichkeit haben, Dateien in diesen Bereich abzulegen; alle eingestellten Dateien sollen durch alle Nutzer gelesen werden können; lediglich das Löschen von Dateien soll dem Nutzer vorbehalten bleiben, der die Dateien in diesen Bereich eingestellt hat.
- Um die Abgabe von Arbeitsergebnissen und Prüfungsleistungen zu ermöglichen, soll ein nicht öffentlich zugänglicher Upload-Bereich eingerichtet werden. Nach Einstellen der Datei durch den/die Studierenden unter Angabe des zuständigen Dozenten ist die Datei nur für den zuständigen Dozenten sichtbar und aufrufbar.
- Haus- und Semesterarbeiten sollen in einem eigenen Hausarbeiten-Archiv für alle Studierenden und Lehrenden zugänglich sein.
- Um die gegenseitige Kommunikation und den Erfahrungsaustausch unter den Studierenden zu stärken, sollen umfangreiche Studierendenprofile eingerichtet werden, die detaillierte Angaben über Dienststelle, Funktion und Position aller Studierenden enthalten. Nach Beendigung

des Studiums sollen diese Angaben in einen eigenen Alumni-Bereich überführt werden.

- Ähnlich wie die Studierendenprofile sollen auch ausführliche Dozentenprofile im Intranet präsent sein. Neben den hauptamtlich an der Hochschule lehrenden Professoren sollten auch Lehrbeauftragte in diese Übersicht mit einbezogen werden. Bei den Inhalten der jeweiligen Einträge sollte besonderer Wert auf die Angabe spezifischer Studiengang-relevanter Informationen Wert gelegt werden, z.B. auf genaue Angabe der Funktionen (z.B. Modulverantwortlicher), Mitwirkungen bei weiteren Lehrveranstaltungen, Ausbildungs- und Tätigkeitsprofil, etc.
- Zahlreiche für Studierende wichtige Informationen erreichen diese nur verspätet oder gar nicht. Daher soll im Intranetbereich ein interner Infodienst eingerichtet werden, der schnell und kompakt unmittelbar für den Studienbetrieb relevante Informationen präsentiert. Beispiele wären: ablaufende Fristen, Terminverschiebungen, Änderungen bei Raumbelegungen oder Dozenten, Krankheitsfälle, interne oder hochschulpolitische Neuigkeiten (Akkreditierung, Änderung des Abschlussgrades, o.ä.), Hinweis auf einen neuen Stundenplan, etc.
- In den letzten zwei Jahren, in denen die bisherige Studiengangs-Website in Benutzung war, wurden nur zwei Einträge in die Diskussionsforen vorgenommen. Aufgrund dieser Erfahrung kann davon ausgegangen werden, dass Diskussionsforen in dem neuen Webangebot des Studiengangs entbehrlich sind. Eine Weiterführung der Diskussionsforen in der neuen Website wird daher nicht angestrebt.

2. Technik

Aufgrund zahlreicher Problembereiche von Studierenden¹⁰ bei der Nutzung des Download-/Upload-Bereiches per FTP-Zugriff soll dieses Konzept aufgegeben werden und die Anwendung auf einer anderen technischen Basis realisiert werden. Viele Studierende, die den Zugriff von ihrem Arbeitsplatz aus versucht haben, sind durch Firewall-Einstellungen oder auch durch andere technische Schutzmechanismen daran gehindert worden. Teilweise war die Nutzung sogar innerhalb des Gebäudes der Hochschule nicht möglich, da die Installation älterer Internet Explorer ohne FTP-Funktionalität ein Update notwendig machte, das aber aufgrund eingeschränkter Zugriffsrechte nur mit Administrator-Rechten an dem betroffenen Rechner durchgeführt werden konnte.

Ein zentraler bei allen Gesprächen und Rückmeldungen auftauchender Kritikpunkt bezog sich auf die Aktualität der Inhalte und besonders auf die teilweise hohe Verzugszeit bis zum endgültigen Einstellen von Informationen in die Website. Besonders bei zeitnah benötigten Informationen (z.B. aktualisierte Terminübersichten oder neue Unterrichtsmaterialien) wurde die Verzugszeit als unangemessen hoch empfunden. Die Ursache für diese Missstände muss in erster Linie darin gesehen werden, dass die bisherige Website auf Basis statischer HTML-Seiten realisiert wurde. Dieses starre Konzept macht für jede kleinste Änderung den Umweg über den hauptverantwortlichen Webmaster notwendig - vor allem deshalb, weil bei statischen HTML-Seiten die Gefahr besteht, durch Veränderungen des Inhalts das Layout der Seiten zu beeinträchtigen. Das Delegieren der Verantwortung für bestimmte Bereiche ist prinzipiell zwar möglich, setzt dann aber bei den Bereichsverantwortlichen Kenntnisse in HTML und einem HTML-Editor voraus sowie so viel technische Kenntnisse, dass der Datenabgleich mit dem Webserver z.B. per FTP vorgenom-

¹⁰ Nachdem der Up-/Download-Bereich auf FTP-Basis in die bisherige Studiengangs-Website implementiert und am 18.10.2001 offiziell zur Nutzung frei gegeben wurde, erreichten den Verfasser aufgrund seiner Funktion als Webmaster dieser Website zahlreiche Problemmeldungen von Studierenden und Dozenten.

men werden kann. Dazu hin sind exakte Absprachen zwischen dem Webmaster und den jeweiligen Bereichsverantwortlichen notwendig, um ein gegenseitiges Überschreiben der Änderungen beim Aktualisieren der Inhalte zu vermeiden.

Um dieses ganze Problemfeld grundlegend bewältigen zu können, wird übereinstimmend von der Studiengangsleitung, den Rückmeldung gebenden Studierenden und dem Verfasser vorgeschlagen, die Erstellung und Pflege der neuen Website durch den Einsatz eines Content-Management-Systems zu automatisieren. “Die konzeptionelle Grundlage der Automatisierung ist [...], [dass] die Inhalte von ihrer Darstellung separiert werden”¹¹. Durch den Einsatz eines solchen CMS könnte die Verantwortung über bestimmte inhaltliche Bereiche delegiert werden, ohne die Inhalte anderer Bereiche oder das Layout der gestamten Website der Gefahr unbeabsichtigter Veränderungen auszusetzen. “Die Trennung von Inhalt und Layout bietet den wesentlichen Vorteil, dass der Prozeß der Inhaltserstellung von der Website-Programmierung unabhängig ist. [...] Für die Erstellung einer neuen Website muss der Autor keinen Techniker bzw. Webmaster mehr bemühen”¹². Eine derart durchgeführte Dezentralisierung und Delegierung der inhaltlichen Betreuung hat unmittelbaren Einfluss auf die Verzugszeit zwischen Erstellung von Inhalten und deren Publikation im Web. “Einer der ersten bemerkbaren Effekte nach der Einführung eines WCMS [= Web-Content-Management-System] ist die drastische Veränderung der Aktualisierungszeiträume. Inhalte gehen direkt vom Schreibtisch der Verantwortlichen ins Web und müssen keine Umwege mehr nehmen“¹³.

¹¹ Web Content Management. Websites professionell betreiben, S. 90.

¹² Ebd., S. 91.

¹³ Ebd.

IV. Mögliche Methoden und Werkzeuge des Website- und Content-Managements

Bei der Prüfung und Bewertung von Content-Management-Systemen hinsichtlich ihrer Eignung für die Aufnahme der überarbeiteten Website des Master-Studiengangs Bibliotheks- und Medienmanagement waren zwei grundlegende Anforderungen zu berücksichtigen.

Zum einen muss das CMS in der Lage sein, nicht nur Web-Content zu automatisieren und zu verwalten, sondern auch die Anwendung bestimmter Applikationen zu ermöglichen, durch die z.B. die im Kapitel III. konkret aufgeführten Anforderungen realisiert werden könnten. Da, je nach Status des Nutzers (Studierender, Lehrender, Administrator), unterschiedliche Funktionen ausgeführt werden sollen, muss, neben der bloßen Fähigkeit, solche interaktiven Anwendungen z.B. in Form von Plug-Ins anzubinden, auch die Möglichkeit gegeben sein, eine differenzierte Rechtevergabe anzuwenden. Optimal könnte auf diese Anforderungen durch eine vollständige individuelle Entwicklung für die Bedürfnisse dieses Studiengangs eingegangen werden. Allerdings sprengt eine solche Entwicklung klar den finanziellen Rahmen der Hochschule. "Eine vollständige CM-Lösung kann eine Investition in sechsstelliger Höhe bedeuten - inklusive Schulung und Beratung (gern vornehm als 'Consulting' bezeichnet)"¹⁴. Bei diesen Dimensionen muss dann zu Recht die Frage nach der Verhältnismäßigkeit von Einsatz und Ergebnis gestellt werden - selbst wenn die finanzielle Situation der Hochschule nicht so aussehen würde, wie sie im folgenden Absatz näher beschrieben wird.

Die zweite grundlegende Voraussetzung ist, dass der Einsatz des gewählten Produktes vollkommen kostenneutral erfolgen soll. Das Ministerium für Wissenschaft und Forschung Baden-Württemberg hat erhebliche finanzielle Kürzungen der Etats aller Fachhochschulen des Landes vorgenommen. Allein im Jahr 2004 muss die HdM aufgrund dieser Kürzungen 665.000 Euro Haushaltsmittel einsparen. Da über 80 Prozent der Finanzmittel für vertraglich gebundene Personalkosten eingesetzt werden müssen, verbleibt lediglich der vergleichsweise geringe Sach-

¹⁴ Rohse, Frank; Lennartz, Sven: Wuchern mit Pfründen, S. 186.

mitteletat zur Aufnahme der einzusparenden Mittel¹⁵. Außerdem muss beachtet werden, dass bis jetzt die Realisierung von Master-Studiengängen an der ehemaligen HBI, der ehemaligen HDM und der jetzigen HdM allein mit vorhandenen Finanz- und Personalmitteln durchgeführt werden musste, da vom zuständigen Ministerium für Wissenschaft und Kunst Baden-Württemberg für die Einrichtung von Master-Studiengängen keinerlei zusätzliche Personalkapazität oder zusätzliche Sachmittel bereit gestellt wurden¹⁶. Der Master-Studiengang Bibliotheks- und Medienmanagement verfügt also über keine eigenen ihm speziell zugewiesenen Personalressourcen, er ist stattdessen auf Kapazitäten von Mitarbeitern, die dem Fachbereich oder den grundständigen Studiengängen zugeordnet sind, angewiesen. Somit war der mögliche finanzielle Rahmen klar abgesteckt: Die Neukonzeption der Master-Studiengangs-Website - und vor allem deren Realisierung mittels eines CMS - darf keinerlei Kosten verursachen.

Nach Berücksichtigung dieser beiden grundlegenden Voraussetzungen stehen lediglich folgende Optionen zur Wahl: eine komplette Eigenentwicklung, die von den Personalressourcen und Sachmitteln des Fachbereichs getragen wird, die Anwendung und Anpassung einer kostengünstigen Freeware-, Low-Cost- oder OpenSource-Lösung oder die Berücksichtigung eines bereits an der HdM vorhandenen Systems.

A. Individuelle Programmierung

“Fertighaus, Bausatz, selbst mauern oder mieten? Wer eine CMS-verwaltete Site aufbauen will, muss sich entscheiden, wie maßgeschneidert das Content-Management-System auf seine Systemumgebung zugeschnitten sein soll, wie genau es seinen Anforderungen an das Layout entsprechen muss und wie viel Entwicklungsarbeit er bereit und in der Lage ist, dafür selbst aufzuwenden”¹⁷. Ein individuell programmiertes CMS (z.B. auf ASP- oder PHP-Basis) bietet sicherlich ein Höchstmaß an Anpassungsvermögen und kann zudem noch

¹⁵ Nach: HdM-Infoletter. https://inhouse.hdm-stuttgart.de/intranet/intranet_infoletter/intradok_20031215142650/download, S. 6.

¹⁶ Diese Aussage wurde in einem Telefonat am 21.01.2004 vom Leitenden Verwaltungsdirektor der Hochschule der Medien, Herrn Gerold Müller, bestätigt.

¹⁷ Bager, Jo: Hüter der Inhalte. <http://www.heise.de/kiosk/archiv/ct/02/20/172/>

auf einer der bereits vorhandenen Hardware-Plattformen eingesetzt werden. Allerdings müssen diese Vorteile erkaufte werden durch ein Höchstmaß an Entwicklungsarbeit. Da auch hierfür letztendlich finanzielle Ressourcen in Form von Arbeitsstunden von Mitarbeitern, Fortbildung u.ä. in beträchtlichem Umfang eingesetzt werden müssen, stellt die Möglichkeit der individuellen Programmierung keine echte Alternative dar.

Ein weiterer Nachteil einer solchen individuellen Lösung liegt in der ausgeprägten Abhängigkeit von dem ausführenden Mitarbeiter. Da es sich bei der Tiefe des benötigten Fachwissens um ein ausgesprochenes Expertenwissen handeln würde, kann schon die Suche nach einer Vertretung für Urlaubs- oder Krankheitsfälle Probleme aufwerfen.

B. Low-Cost-, Freeware- und OpenSource-Systeme

Mittlerweile sind Content-Management-Systeme als eigene Anwendung auch in das Bewusstsein von Betreibern privater Websites oder kleinerer Intranet-Lösungen gelangt. "Open-Source-Plattformen, günstige Desktop-Massenware und ASP-Anwendungen machens möglich: Auch Privatleute und Unternehmen mit kleinem Online-Budget müssen nicht auf den Komfort eines Content-Verwalters für die Pflege ihrer Website verzichten"¹⁸. Darunter befinden sich auch einige preisgünstige Mietlösungen, die von Providern ihren Kunden zur Verfügung gestellt werden. Bei genauerer Betrachtung zeigt sich aber, dass nur wenige Produkte die Anforderungen und Voraussetzungen, die in den Absätzen III.A., III.B. und IV. definiert wurden, erfüllen können.

- Mietlösungen sind für Websites der HdM nicht einsetzbar, da sie von den anbietenden Providern lediglich für Kunden zur Verfügung gestellt werden, die auch bei ihnen die Webseiten "gehostet" haben. Da die HdM aber selbst Web-Server unterhält, benötigt sie die Dienste von Providern lediglich zur Anmietung von Domain-Namen und nicht für die Bereit-

¹⁸ Bager, Jo; Sachs, Andreas: Inhalts-Dompteure. <http://www.heise.de/kiosk/archiv/ct/02/20/178/>

stellung von Web-Servern oder Web-Serverdiensten. Außerdem haben Mietlösungen laufende Kosten zur Folge.

- Desktop-Anwendungen für den Massenmarkt können die erforderlichen Leistungsmerkmale nicht erfüllen. Entweder steht gar keine oder nur eine sehr rudimentäre Benutzerverwaltung (z.B. Unterscheidung nur zwischen zwei Gruppen: Administrator und Anwender) zur Verfügung, oder Layout und Design sind auf eine bestimmte Anzahl vorgegebener Vorlagen begrenzt¹⁹. Doch gerade die Möglichkeit einer stark ausdifferenzierten Rechtevergabe ist für die verschiedenen geplanten Anwendungszwecke des Intranet-Bereiches zwingend notwendig. Erweiterungen von Design-Vorlagen oder Funktionalitäten (z.B. durch PlugIns) sind - wenn überhaupt - nur gegen Aufpreis möglich²⁰.
- Einzig die OpenSource-Produkte, bzw. deren Ableger, bieten eine ausreichend ausgebaute und auch für spezielle Anforderungen individuell noch weiter ausbaubare Funktionalität²¹. Aufgrund des öffentlich zugänglichen Quellcodes werden OpenSource-Produkte, wie z.B. das auf PHP basierende PHP-Nuke, teilweise stark verändert und weiter entwickelt. "Aus PHP-Nuke ist mittlerweile eine Reihe von Schwesterprojekten hervorgegangen. Eines der populärsten ist Post-Nuke. Im Unterschied zu PHP-Nuke, dessen Basissystem ein einzelner Programmierer weiterentwickelt, arbeitet eine große Entwicklergemeinschaft an Post-Nuke"²². Da diese Entwicklergemeinschaften mitunter sehr umfangreich werden können, steht dem Anwender oftmals eine hervorragend ausgebaute Infrastruktur zur Verfügung: Über das WWW lassen sich neue Versionen, Patches und Bugfixes, Erweiterungsmodule, Dokumentationen, Anwenderforen und FAQs frei abrufen.

¹⁹ Nach: Ebd.

²⁰ Nach: Ebd.

²¹ Nach: Rohse, Frank; Lennartz, Sven: Wuchern mit Pfründen, S. 190.

²² Bager, Jo; Sachs, Andreas: Inhalts-Dompteure. <http://www.heise.de/kiosk/archiv/ct/02/20/178/>

C. Einsatz und Adaption eines an der HdM vorhandenen Content-Management-Systems

Anfang des Jahres 2004 werden an der Hochschule der Medien mehrere Web-Publishing- bzw. Content-Management-Systeme mit den unterschiedlichsten Schwerpunkten in verschiedenen Abteilungen der Hochschule eingesetzt.

- Das Produkt Livelink der Firma OpenText Corp. wird im Fachbereich Information und Kommunikation der HdM verwendet.
- Der IBT-Server der Fa. Time4you wird ab September 2003 als die zentrale E-Learning-Plattform der Hochschule eingesetzt, die vom Kompetenzzentrum e-Learning betreut wird.
- Seit Frühjahr 2002 ist das OpenSource-Produkt Zope das zentrale Administrationsinstrument für die offizielle Website der Hochschule und für zahlreiche weitere Webangebote von Studiengängen, Abteilungen, Projekten, Veranstaltungen, Stiftungen und Vereinen, die mit der Hochschule der Medien in Verbindung stehen.

1. Livelink / OpenText

Im Dezember 2001 wurde das Produkt Livelink der Firma OpenText im Rahmen eines Sondermittelantrags (Schwerpunktprogramm IV: Förderungsschwerpunkt Informations- und Kommunikationssysteme, Multimedia) beschafft. Die Beschaffung umfasst die Nutzungsrechte und einen Wartungsvertrag für die Dauer von 3 Jahren. Die Kosten dafür beliefen sich auf ca. 90.000 DM²³. Livelink ist ein professionelles kommerzielles Produkt, um collaborative-working zwischen Mitarbeitern, Arbeitsgruppen, Abteilungen oder ganzen Unternehmenszweigen zu realisieren. "Open Text Livelink lässt sich als Kollaborationslösung für größere Unternehmen charakterisieren und wird als installierbares Produkt [...] angeboten.

²³ Diese Angaben beruhen auf einem Gespräch, das der Verfasser am 7.1.2004 mit dem technischen Leiter des IT-Service des Fachbereichs Information und Kommunikation, Herrn Peter Bauer, geführt hat.

Das Livelink-Kernsystem bildet die Kommunikation im Unternehmen in drei Schichten ab: dem Unternehmensarbeitsbereich, die Projektarbeitsbereiche und die persönlichen Bereiche der Mitarbeiter²⁴. Zum Funktionsumfang von Livelink gehören Dienste zur virtuellen Teamarbeit, zur Automatisierung von Geschäftsabläufen, zum Dokumenten-Management²⁵. Livelink arbeitet konsequent Web-basiert, die Gestaltung der einzelnen Angebote kann HTML-konform ohne Einschränkungen erfolgen. Livelink wird eingesetzt zur Verwaltung der Homepage des IT-Service, wie auch in Projekten und Seminaren mehrerer Studiengänge des Fachbereichs Information und Kommunikation.

2. IBT-Server / Time4you

Seit dem September 2003 wird an der HdM der IBT-Server der Firma Time4you als zentrale e-Learning-Plattform der Hochschule eingesetzt. Time4you stellt die Software kostenlos zur Verfügung und verspricht sich von der Zusammenarbeit mit der Hochschule neue Impulse in der Entwicklung und Erprobung neuer Ideen²⁶. Neben den für die E-Learning-Funktionalitäten notwendigen Programm-Modulen ist auf dem an der HdM vorhandenen System noch das Modul web-authering installiert²⁷. Der IBT-Server versteht sich als ein Content-Management-System für Großprojekte zur Durchführung von virtuellen Kursen, electronic collaboration und distance work²⁸. Die Möglichkeiten, die der IBT-Server bietet, zielen eindeutig auf den primären Anwendungszweck als e-Learning-Plattform: projektbezogene Pinwand, Einsendung von Arbeiten an den Tutor inkl. Möglichkeit, dessen Kommentare zu lesen, direkte Kommunikation angemeldeter User über PopUp-Fenster, Diskussionsforen, Online-Chat, Evaluation des Kurses durch die Teilnehmer, Benut-

²⁴ Das Produkt Livelink 9.1. <http://www.vrtpj.de/products/livelink.html>.

²⁵ Nach: Livelink 9.1. http://www.contentmanager.de/itguide/produkt_530_livelink_91.html.

²⁶ Nach: e-Learning-Newsletter. <http://www.hdm-stuttgart.de/e-learning/elearning.pdf>, S. 7.

²⁷ Diese Aussage beruht auf einer Zusammenstellung vom 20.01.2004 von Frau Alexandra Wenzel, Mitarbeiterin im Kompetenzzentrum E-Learning der HdM.

²⁸ Nach: Basiswissen. <http://www.time4you.de/ibt/www/ibt/home/products/esuite.xhtml>.

zergalerie, chronologische Liste aller besuchten Modulseiten, etc.²⁹. Eine freie Gestaltung der Inhalte der einzelnen Projektseiten ist möglich.

3. Zope

Zope ist ein lizenzfrei nutzbares OpenSource-System, das sich selbst weniger als Content-Management-System bezeichnet, sondern eher als "Open Source Web Application and Content Management Server"³⁰. Zope bietet verschiedene Dienste, mit deren Hilfe verschiedenste web-basierte Anwendungen realisiert werden können - darunter natürlich auch Anwendungen, die unerfahrenen Anwendern das Content-Management einer Website ermöglichen. "Ein Web-Applikationsserver fungiert als Bindeglied (Middleware) zwischen dem Browser, der vom Programmierer entwickelten Software (Applikation) und einer Datenbank"³¹. Diese Offenheit gegenüber den verschiedensten Einsatzzwecken und Anwendungen hat aber auch Nachteile: Hinsichtlich des Bedienungskomforts und der notwendigen Vorarbeiten ist ein "nacktes" Zope-System kommerziellen "Out-of-the-box"-Lösungen unterlegen³². Für ein OpenSource-Produkt erscheint Zope aber unerwartet etabliert. Einerseits typisch für Open Source Produkte: Eine große Zope-Community stellt zahlreiche für Zope geschriebenen Applikationen (sog. Zope-Produkte) zur Verfügung, die - aufgrund des frei änderbaren Quellcodes - an ihre individuellen Umgebungen angepasst werden können. Andererseits jedoch lebt mittlerweile eine ständig steigende Anzahl von Firmen von der kommerziellen Entwicklung von Zope-Produkten oder von Lehrgängen in der Anwendung von Zope. Verlage bieten Handbücher zu Zope, Lehrbücher zu der Zope-eigenen Programmiersprache Python, Anleitungen zu verschiedenen Zope-Produkten und Bücher zu speziellen Zope-Problemen an. Mitunter

²⁹ Nach: e-Learning-Newsletter. <http://www.hdm-stuttgart.de/e-learning/elearning.pdf>, S. 5 und 6.

³⁰ SuperGuide for Newbies (Part I of III) - Welcome to the World of Zope. http://www.zopemag.com/Guides/SuperGuide_Newbies.html.

³¹ Zope. Content-Management- & Web-Application-Server, S. 2.

³² Nach: Rohse, Frank; Lennartz, Sven: Wuchern mit Pfründen, S. 190.

haben Internet-Provider auf ihren Webservern Zope installiert, das sie dann ihren Kunden für das Management von deren eigenen Website anbieten³³.

An der HdM wird Zope bereits mehrfach eingesetzt. Neben der offiziellen Homepage der Hochschule, betreut von einem hauptamtlichen Content-Manager, sind auch die Websites mehrerer Studiengänge aus allen Fachbereichen, die überwiegend von Studiengangs-Assistenten betreut werden, auf Zope-Systemen realisiert. Der Fachbereich Information und Kommunikation setzt zwei Webserver unter Zope ein: einen Linux-Server und einen Windows2000-Server.

D. Auswahl eines Systems

Da schon bei der Aufstellung von potenziell geeigneten Systemen auf eine möglichst niedrige bzw. überhaupt nicht vorhandene Kostenbelastung Wert gelegt wurde, bleibt als wichtigstes Entscheidungskriterium für die Auswahl die Leistungsfähigkeit der Systeme. Welches ist so flexibel und durch den Betreuer ausreichend manipulierbar, dass alle für die Website gewünschten Applikationen damit realisiert werden können?

Eine komplett selbst hergestellte Lösung kann zwar auf die Anforderungen an die Website maßgeschneidert werden, ist jedoch mit einem so hohen Arbeitsaufwand verbunden, dass die dafür notwendige Personalkapazität in der momentanen Situation der Hochschule auf absehbare Zeit weder vom Studiengang noch vom Fachbereich bereit gestellt werden kann.

Von den kostenfrei verfügbaren Programmen können letztendlich nur die Open Source-Lösungen die entsprechende Leistung und eine möglichst weitreichende Erweiterungsfähigkeit bieten.

Von den bereits an der HdM eingesetzten Systemen scheint der IBT-Server von Time4you am wenigsten für diesen Zweck geeignet zu sein. Die strenge Ausrichtung auf e-Learning Projekte schafft zwar die technische Grundlage,

³³ Nach: Bager, Jo; Sachs, Andreas: Inhalts-Dompteure. <http://www.heise.de/kiosk/archiv/ct/02/20/178/>.

um die genannten Anforderungen realisieren zu können, gibt aber andererseits eine starke projektorientierte Ausrichtung vor, die von einer kontinuierlichen zeitlichen Abfolge von Arbeitsschritten ausgeht und dann letztendlich zur Beendigung und Schließung eines fertig bearbeiteten Projektes führen soll. Für die eigentliche Ausrichtung des Systems - den produktiven Einsatz als E-Learning-Plattform in der Lehre - ist dies eine gute Voraussetzung, genauso wie die durchgehend gewährleistete Betreuung durch das Kompetenzzentrum e-Learning. Allerdings handelt es sich bei der neuen Website des Studiengangs BMM nicht um eine zeitlich befristete sondern dauerhaft angebotene Website, die auch öffentlich zugängliche Bereiche enthält und nur einzelne technische Möglichkeiten des Systems nutzen würde, um spezielle Anwendungen zu realisieren. Sie widersetzt sich in ihrer Grundvoraussetzung diesem projektorientierten Konzept und kann nicht an einen solchen kontinuierlichen Arbeitsablauf gebunden werden. Außerdem stellt die freiwillige Übereinkunft zwischen HdM und Time4you, auf deren Basis das System der HdM zum Einsatz in der Lehre zur Verfügung gestellt wird, noch einen kleinen Unsicherheitsfaktor dar, da derartige Übereinkünfte von den einzelnen Vertragspartnern natürlich auch aufgekündigt werden können. Ein ähnlicher Umstand stellt den gravierendsten Kritikpunkt bei dem System Livelink da. Obwohl dieses System technisch und funktional geeignet wäre, besteht eine große Unsicherheit durch das Auslaufen des Lizenz- und Wartungsvertrages Ende 2004. Zwar wird innerhalb des Fachbereichs Information und Kommunikation der Hochschule nicht mit dem Schlimmsten gerechnet - einem Entzug der Nutzungslizenzen - , doch muss zumindest davon ausgegangen werden, dass der bestehende Wartungsvertrag nicht weiter verlängert werden wird. Eine kostenpflichtige Verlängerung des Wartungsvertrages, finanziert aus regulären Haushaltsmitteln, ist bei der momentanen finanziellen Lage der Hochschule nicht denkbar³⁴. Doch auch eine Kündigung des Wartungsvertrages bei Fortführung der Nutzungslizenz

³⁴ Diese Angaben beruhen auf einem Gespräch, das der Verfasser am 07.01.2004 mit dem technischen Leiter des IT-Service des Fachbereichs Information und Kommunikation, Herrn Peter Bauer, geführt hat.

hätte Auswirkungen auf den Betrieb. Der Zustand des Systems würde zu diesem Zeitpunkt zementiert werden, weitere Software-Updates, schon die Inanspruchnahme von Supportunterstützung, wären ausgeschlossen.

Bei Betrachtung aller dieser Umstände scheint das Open Source-Produkt Zope noch die ausgewogenste Variante zwischen technischer Leistungsfähigkeit, Offenheit hinsichtlich möglicher Erweiterungen, Sicherheit in Betrieb und Entwicklung und voraussichtlich notwendigem Arbeitseinsatz darzustellen. Die eigene Entwicklung von Modulen mit einer speziellen gewünschten Funktionalität ist zwar möglich, aber nicht immer notwendig, da eine Vielzahl von dokumentierten Zope-Produkten zur freien Verwendung zur Verfügung steht. Darüber hinaus ist eine Vernetzung mit den anderen an der HdM bestehenden Webservern möglich. So können auf anderen Servern installierte Zope-Produkte direkt benutzt oder übernommen und in das eigene Angebot integriert werden. Bei Zope scheint durch die vergleichsweise hohe Verbreitung ein Grad der Etablierung erreicht zu sein, der eine gewisse Zukunftssicherheit hinsichtlich der technischen Entwicklung und des Supports bietet. Bei der Verwendung von Zope ist sicher ein erhöhtes Maß an Entwicklungs- und Betreuungsaufwand zu leisten, allerdings steht eine gut ausgebaute Infrastruktur zur Unterstützung und möglichen Wertschöpfung zur Verfügung. Durch die Verbreitung von Zope innerhalb der HdM sind kompetente Ansprechpartner zur Beratung und für Vertretungen im Krankheit- oder Urlaubsfall vorhanden.

Nach Berücksichtigung aller dieser aufgeführten Gründe wurde deshalb Zope vom Verfasser als zukünftige technische Basis zur Entwicklung der neuen Website des Master-Studiengangs Bibliotheks- und Medienmanagement ausgewählt.

V. Zope - Technische Grundlagen

A. Installation und Konfiguration

Für das Produzieren und “Hosten” der neuen Website des Studiengangs BMM ist es möglich, einen am Fachbereich Information und Kommunikation schon vorhandenen Webserver mit dem Gerätenamen Morpheus mit zu verwenden. Dieses Gerät ist ein handelsüblicher PC, der mit Windows2000 (Server) ausgestattet ist. Auf diesem Server wurde bereits Zope in der Version 2.5.1. für PC unter Microsoft Windows als Web-Content-Management-System installiert und eingesetzt. Dieser Server beherbergt die Homepage der Expertengruppe “Bibliothek und Schule” des Deutsche Bibliotheksverbandes (Nachfolgeorganisation der schulbibliothekarischen Arbeitsstelle des ehemaligen dbi Berlin), die in Kooperation mit dem Fachbereich Information und Kommunikation der HdM auf der Basis von Zope bereit gestellt wird. Durch den Einsatz eines speziellen Zope-Produktes, des “Virtual-Host-Monster”, wurde es ermöglicht, innerhalb von Zope virtuelle Hosts anzulegen und zu verwalten und dadurch mehrere getrennte Webangebote ohne den Einsatz einer anderen Web-Server-Software mit jeweils einem eigenen Domain-Namen störungsfrei nebeneinander zu betreiben. Sobald die neue Website durch die Studiengangsleitung freigegeben wird, ist es dadurch also problemlos möglich, den bisherigen Domain-Namen `http://bmm.iuk.hdm-stuttgart.de` vom bisherigen Webserver Pan auf den Zope-Webserver Morpheus umzuschalten. Da es sich um eine sog. Subdomain von `iuk.hdm-stuttgart.de` handelt, die von dem Fachbereich IuK eigenständig verwaltet und zugeordnet werden, kann diese Umstellung kurzfristig und ohne Kontaktaufnahme mit externen Providern geschehen. Auch Morpheus ist, wie alle anderen Webserver des Fachbereichs, außerhalb des Fachbereichs-Firewalls mit dem Internet verbunden.

B. Grundlegende Funktionsweise von Zope

Zope ist ein Web-Applikation-Server, der durchgehend objektorientiert aufgebaut ist, der die Festlegung stark ausdifferenzierter Nutzerrechte gestattet, der eine integrierte Datenbank, einen Webserver, eine eigene Script-Sprache (DTML) und eine eigene Programmiersprache (Python) bereit hält³⁵. Bei Bedarf kann auch SQL oder Perl-Code in Zope eingesetzt werden. Zope bietet somit eine Arbeitsumgebung, um verschiedenartige Funktionen in einer web-basierten Umgebung ablaufen zu lassen. Die Spanne dieser Funktionen reicht von der bloßen Anzeige (z.B. von HTML-Dokumenten) über interaktive Anwendungen (E-Shopping-System, Module für Content-Management, Diskussionsforen) bis hin zu hochkomplexen individuell programmierten Spezialanwendungen. Die überwiegend zur Steuerung der Zope-Funktionen verwendete DTML (Document Template Markup Language) orientiert sich in Aufbau und Syntax an HTML und ist daher für Anwender mit HTML-Vorkenntnissen leicht erlernbar. Zope braucht keine Client-Installation, sondern arbeitet ausschließlich über den Browser.

Bei der Arbeit mit Zope werden drei verschiedene "Ansichten" unterschieden: Als Erstansicht wird der reine Zope-Quellcode bezeichnet, wie DTML-Tags oder Python-Programmcode. Als Zweitansicht (siehe Abb. 1.) wird die sog. Management-Konsole von Zope bezeichnet. Über die Management-Konsole werden Dokumente editiert, Objekte verwaltet, Produkte erzeugt oder importiert und die Systemeinstellungen von Zope administriert. Der Management-Screen wird aufgerufen, indem an die URL der Zope-Seite das Suffix `/manage` angehängt wird. Ein Anwender, der eine von einem Zope-Server bereit gestellte Ressource im Browser betrachtet, befindet sich hingegen in der sog. Drittansicht³⁶. In dieser Drittansicht ist in aller Regel kein DTML- oder Python-Code sichtbar, sondern lediglich der durch Zope für die Darstellung im Browser generierte HTML-Code.

³⁵ Nach: Zope. Content-Management- & Web-Application-Server, S. 2.

³⁶ Nach: Ebd., S. 69 - S. 70.

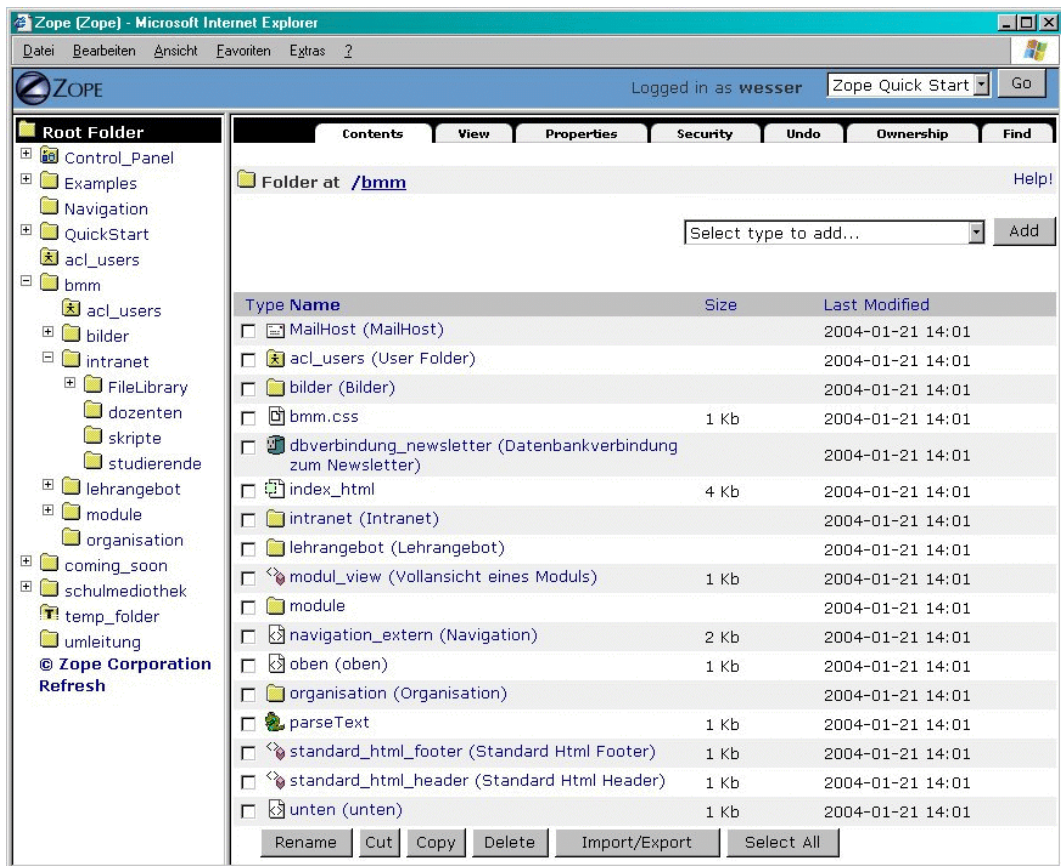


Abb. 1: Die Management-Konsole von Zope entspricht der sog. "Zweitansicht"

1. Klassen, Objekte und Methoden

Alle Daten und Dateien, die innerhalb von Zope abgespeichert werden, werden als Objekte bezeichnet. Auch die Ordner, die in Zope zum Erstellen einer Hierarchiestruktur und zum Aufnehmen anderer Objekte angelegt werden können, sind derartige Objekte. Objekte verfügen über bestimmte Eigenschaften (auch als Attribute oder properties bezeichnet), die bestimmte Werte besitzen können. Standard-Attribute, die jedes Objekt in Zope besitzt, sind die "ID", vergleichbar dem Dateinamen, und ein frei wählbarer "title", der das Objekt genauer bezeichnen kann. "Die objektorientierte Philosophie orientiert sich an der realen Welt. Sie geht davon aus, daß die Welt aus Objekten besteht, die man voneinander unterscheiden kann. Objekte besitzen Zustände, führen Aktionen aus und

kommunizieren miteinander³⁷. Werden in Zope neue Objekte gebildet, so werden diese immer von sog. Klassen abgeleitet.

Soll ein Objekt eine Aktion ausführen können, benötigt es dafür eine entsprechende Methode. Der Unterschied zwischen einer DTML-Datei und einer DTML-Methode ist auf den ersten Blick nicht ersichtlich, da in beiden Objekt-Arten DTML-Befehle eingesetzt werden können, die dann beim Aufruf der Datei ausgeführt werden. Bei näherer Betrachtung zeigen sich aber zwei wichtige Unterschiede. Bei DTML-Dateien können mehr als nur die Standard-Attribute hinzugefügt und definiert werden. Im Zope-Management-Screen erscheint dazu ein Reiter *properties*. Bei DTML-Methoden ist dies nicht möglich. Der zweite Unterschied: Eine DTML-Datei bezieht sich immer auf sich, eine DTML-Methode bezieht sich hingegen - da sie ja nur über die Standard-Attribute verfügt - immer auf die Attribute des übergeordneten Elements. Ein Beispiel: Mit dem DTML-Befehl `<dtml-var id>` kann die ID eines Objektes abgefragt und angezeigt werden. Wird dieser Befehl in einer DTML-Datei aufgerufen, so wird immer exakt die ID dieser aufrufenden DTML-Datei selbst ausgegeben. Wird jedoch der Befehl in einer DTML-Methode ausgeführt, so wird die ID des Ordners ausgegeben, in dem sich diese Methode befindet - da dieser in diesem Beispiel das übergeordnete Objekt darstellt³⁸.

Die folgende Abbildung soll anhand eines Beispiels die Beziehungen zwischen den verschiedenen Zope-Objekten erläutern³⁹.

³⁷ Ebd., S. 21.

³⁸ Nach: Hörtzsch, Matthias; Rapko, Andreas: Content Management und Zope, S. 45.

³⁹ Die Abbildung wurde durch den Verfasser erstellt nach einem Beispiel aus: Zope. Content-Management- & Web-Application-Server, S. 21 - S. 22.

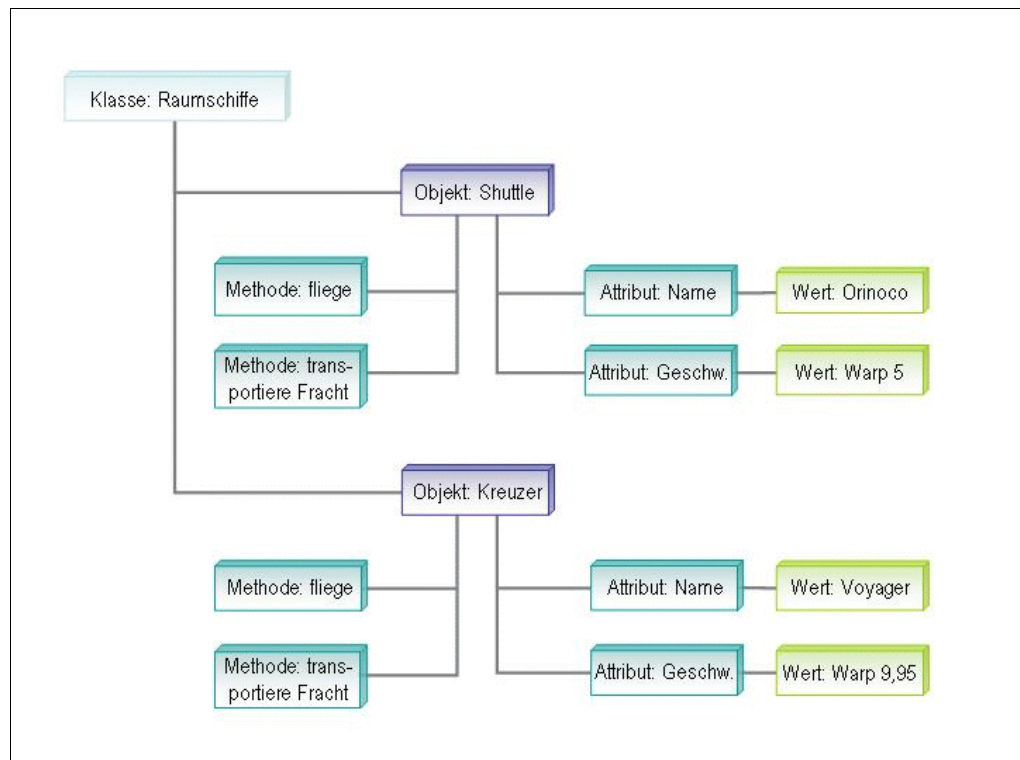


Abb. 2: Klassen, Objekte, deren Attribute und Werte in Zope

Objekte in Zope können “folderish objects” - ordnerähnliche Objekte - oder “non-folderish objects” sein. Das besondere Merkmal von ordnerähnlichen Objekten ist, dass diese andere Objekte aufnehmen können. Eine Eigenschaft von Zope bei der Adressierung von Objekten ist, dass nicht zwingend ein eindeutiger Pfad zu den Objekten angegeben werden muss. Findet Zope ein gesuchtes Objekt nicht im aktuellen Ordner, werden der Reihe nach von unten nach oben in der Hierarchie alle übergeordneten Ordner durchsucht. “Das bedeutet, daß ein Objekt ein anderes Objekt aus darüberliegenden Ordnern automatisch aufrufen kann, ohne daß ein Pfad angegeben werden müsste”⁴⁰. Wenn also für eine komplette Website Standard-Formatierungen angewendet werden sollen, so reicht es aus, diese Formatierungen in eine Datei z.B. mit dem Namen `standard_html_header` zu schreiben, diese Datei im Root der Website zu speichern und in allen Dateien, die in diesem und in einem darunter liegenden Ordner gespeichert werden, durch Einfügen des DTML-Befehls

⁴⁰ Zope. Content-Management- & Web-Application-Server, S. 40.

`<dtml-var standard_html_header>` den Inhalt des Objektes `standard_html_header` anzeigen zu lassen. Dabei ist es nicht notwendig, jeweils den genauen Pfad von `standard_html_header` mit anzugeben.

2. Produkte und ZClasses

Das Verfahren, eine gewünschte Funktion in Zope über DTML-Dokumente und DTML-Methoden zu realisieren, kann, vor allem bei größeren Projekten, schnell unübersichtlich werden. In diesen Fällen ist es sinnvoller, ein eigenes Produkt zu erstellen. In einem Zope-Produkt sind alle Objekte und Methoden enthalten, die zur Ausführung der gewünschten Funktion notwendig sind⁴¹. Ein großer Vorteil: Produkte werden zentral gespeichert und stehen daher an jedem Punkt innerhalb einer Zope-Installation zur Verfügung. So ist das Erzeugen eines Produktes besonders dann sinnvoll, wenn innerhalb der Hierarchiestruktur einer Website von verschiedenen Positionen aus auf eine Funktion zugegriffen werden soll. Dies kann sogar auch über die Grenzen mehrerer verschiedener Websites hinaus geschehen - vorausgesetzt, diese Websites befinden sich auf dem selben Zope-Server.

Neben sehr speziellen individuellen Produkten gibt es auch viele Produkte, die Zope-Entwickler programmiert haben, um häufige, wiederholt auftretende Anforderungen zu bewältigen. Diese Produkte werden dann oft von den Autoren auf einschlägigen Websites der Zope-Gemeinde zum freien Download angeboten. "Eine komplette Liste an Produkten findet man unter »Download« bei Zope.org"⁴². Betreiber dieser Website <http://www.zope.org> ist die Firma Zope Corporation⁴³, die auf kommerzieller Basis individuelle Lösungen mit Zope-Produkten oder Python-Programmierungen entwickelt. Im Jahr 2000 wurde auch PythonLabs., die Entwicklungsfirma der Programmiersprache und Zope-Grundlage Python,

⁴¹ Nach: Ebd., S. 298.

⁴² Hörtzsch, Matthias; Rapko, Andreas: Content Management und Zope, S. 199.

⁴³ Die eigentliche Firmen-Homepage der Zope-Corporation lautet: <http://www.zope.com>. <http://www.zope.org> ist ausschließlich ein Tausch- und Beratungsforum für alle Anwendungen und Fragen rund um das OpenSource-Produkt Zope.

Teil der Zope Corporation⁴⁴. Aus diesem Grund kann die Website <http://www.zope.org> - was die Qualität der dort bereit gestellten Zope-Produkte anbelangt - als seriös und vertrauenswürdig eingestuft werden. Jeder Zope-Entwickler oder Zope-Anwender ist gut beraten, auf den verschiedenen einschlägigen Zope-Websites oder Zope-Usergroups⁴⁵ nachzuschlagen, bevor er die eigene Entwicklung eines Produktes angeht⁴⁶.

Eine Möglichkeit, mit überschaubarem Aufwand eigene leistungsfähige Produkte zu erstellen, ist die Arbeit mit sog. ZClasses. Wenn über die ganze Hierarchie einer Website hinweg Objekte eines bestimmten Typs immer wieder benötigt werden, so ist es sinnvoll, eine neue Klasse für derartige Objekte zu erstellen, nach deren Eigenschaften dann die einzelnen Objekte gebildet werden - vergleichbar mit der Funktion einer Matrize. Der Zope-Anwender ist dann nicht mehr nur auf die Objekte beschränkt, die ihm von Zope standardmäßig im "Available-Objects-Menue" angeboten werden⁴⁷. Diese Vorgehensweise wurde in der neuen BMM-Website für das Erzeugen und Darstellen der Module, der Skripte, der Hausarbeiten, der Lehrkräfte und der Studierenden angewendet. Sie soll am Beispiel der Module näher erläutert werden.

Die verschiedenen Module sind die grundlegenden Elemente des Curriculums des BMM-Studiengangs. Der Zugriff auf die Module ist sowohl vom Internet- als auch vom Intranetbereich aus sinnvoll und wünschenswert. Um ihn zu gewährleisten, wurde ein zentrales Produkt namens `modul`, beruhend auf der ZClass `modul`, angelegt. In diesem besonderen Fall wurde die dazugehörige ZClass `modul` als "ordnerähnlich" (folderish object) gekennzeichnet. Dies hat zur Folge, dass alle Objekte, die nach dieser Klasse erstellt wurden, weitere Objekte beherbergen können. Im Falle der Module ist dieses Verfahren sinnvoll, da so jedem Modul die fünf Lehrveranstaltungen, aus denen jedes Modul besteht, zugeordnet

⁴⁴ Nach: About Zope Corporation. <http://www.zope.com/Corporate/CompanyProfile>.

⁴⁵ Die Homepage der deutschen Zope-Usergroups lautet: <http://www.dzug.org>.

⁴⁶ Nach: Content Management mit ZOPE. Webapplikationen mit dem CMF und ZOPE-Produkten entwickeln, S. 307.

⁴⁷ Nach: Zope. Content-Management- & Web-Application-Server, S. 169.

werden können. Dieses Produkt `modul` ermöglicht es somit, neue Objekte von der Klasse `modul` zu erstellen, die dann über die Eigenschaften verfügen, die im sog. `propertysheet` der `ZClass modul` festgelegt wurden. Außerdem kann auf alle Objekte von der Klasse `modul`, auf alle darin enthaltenen Objekte von der Klasse `lehrveranstaltung` und auf alle Eigenschaften dieser Objekte innerhalb der ganzen Website z.B. mit dem `<dtml-var>`-Befehl zugegriffen werden.

Alle für diese Arbeit verwendeten oder eigens erzeugten Zope-Produkte sowie deren Herkunft und deren Funktion sind im Anhang A auf Seite 83 aufgeführt und beschrieben.

3. Benutzer- und Rechteverwaltung in Zope

Die Zope-eigene Sicherheitsstruktur ist stark ausdifferenziert. Durch die Möglichkeit, sog. Rollen (z.B. Administrator, authentifizierter User, anonymer User, etc.) zu definieren, die dann einzelne Nutzer zugewiesen bekommen, können die Nutzer relativ einfach einer bestimmten Stufe in der "Zope-Rechte-Hierarchie" zugeordnet werden⁴⁸. In Zope sind bestimmte Rollen vordefiniert, wie z.B. die Rollen `Anonymous` und `Manager`. Die Rolle `Anonymous` gilt für alle Besucher der betreffenden Website, die sich nicht authentifizieren müssen. Wenn also der Rolle `Anonymous` das Recht `view` entzogen wird, können nur noch eingetragene Benutzer die Website aufrufen. Bei jedem Zope-Objekt - gleich ob Datei oder Ordner - besteht die Möglichkeit, Rollen anzulegen und diesen Rollen bestimmte Rechte für das konkrete Objekt zu vergeben. Gleichzeitig werden auch immer die Rechte für alle Objekte geändert, die sich in der Hierarchie unterhalb des betreffenden Objektes befinden. (Bei einem Ordner wären dies alle darin befindlichen Objekte, einschließlich der Unterordner).

⁴⁸ Nach: Content Management mit ZOPE. Webapplikationen mit dem CMF und ZOPE-Produkten entwickeln, S. 117.

Die Erstellung und Verwaltung von Benutzern geschieht in Zope über das Anlegen eines Ordners namens `acl_users`. In diesem Ordner können Benutzer angelegt, gelöscht oder bestimmten Rollen zugeordnet werden. Diese Benutzer-Informationen gelten immer in dem Daten-Ordner, in dem der `acl_users`-Ordner angelegt wurde, sowie in allen Ordnern die sich unterhalb dieses Daten-Ordners befinden. Soll also ab einer bestimmten Hierarchiestufe auf andere Benutzerdaten zugegriffen werden, so muss in dem betreffenden Ordner ein `acl_user`-Folder mit den gewünschten Benutzerdaten angelegt werden. Wenn in einem bestimmten Ordner ein bereits bestehender Benutzer z.B. Manager-Rechte bekommen soll, so kann dazu auch ein sog. `Slave_user_folder` eingerichtet werden. In diesem Ordner können keine neuen Nutzer eingerichtet werden, er übernimmt die Benutzer des höher gelegenen `acl_user`-Folders, bietet aber die Möglichkeit, diesen vorhandenen Benutzern neue Rollen, z.B. die Rolle `Manager`, zuzuweisen.

Fazit: Rechte werden in Zope nicht Benutzern, sondern Rollen vergeben. Ein Benutzer erhält bestimmte Rechte, in dem er der entsprechenden Rolle zugewiesen wird.

VI. Produktion der neuen Website

A. Migration der bestehenden Inhalte

1. Internetbereich

Da der Internetbereich der alten Studiengangs-Website stets aktuell gehalten wurde, konnte dieser komplette Bereich ohne inhaltliche Änderungen als Ausgangsbasis für die weitere Entwicklung der Website auf den Zope-Server übernommen werden.

In technischer Hinsicht entstand aber ein Problem durch die bisherige Verwendung von Frames. Der Einsatz von Frame-basierten Web-Seiten in Zope ist zwar möglich, doch in den meisten Fällen nicht nötig. Zope arbeitet stattdessen mit einem Standard-header und einem Standard-footer, die jeweils an den Anfang und das Ende einer Seite bei deren Aufruf dynamisch angefügt werden. Dadurch wird der größte Nachteil von Frame-Seiten vermieden: das Unterschlagen der URL des Frame-Inhalts und das alleinige Anzeigen der URL des Framesets - wodurch das Anlegen eines Bookmarks für einzelne Frameinhalte unmöglich wird. Ein kleiner Nachteil dieser Arbeitsweise macht sich aber im Design der Seiten bemerkbar. Wenn in einem Frameset die Länge einer HTML-Datei den ihr zugedachten Platz übersteigt, erscheint ein Scrollbalken ausschließlich in dem Frame der HTML-Datei. Bei der Zope-spezifischen Lösung durch Header und Footer erscheint der Scrollbalken nicht im spezifischen Frame dieser HTML-Datei, der Scrollbalken erscheint vielmehr über die komplette Bildschirmhöhe hinweg. Der Inhalt des Footers erscheint erst, wenn tatsächlich mittels Scrollbalken oder Cursorstasten an das Ende der Datei gesprungen wird. Der Eindruck der bildschirmfüllenden Umrahmung der inhaltstragenden Seiten, wie er bei der bisherigen Website durch den Einsatz von Frames erzeugt wurde, kann mit der Zope-eigenen Lösung nicht hergestellt werden. Da außerdem nur wenige Dateien bei der Umstellung von Frames auf die Zope-spezifische Lösung mit Header

und Footer geändert werden müssen, wurde im Internetbereich vorläufig die bisherige Framestruktur beibehalten.

2. Intranetbereich

Anders als im Internetbereich müssen bei der Übernahme der Intranet-Inhalte mehr Faktoren berücksichtigt werden.

- Zwar können manche Inhalte noch verwendet werden (wie z.B. die Informationen zu den Lehrinhalten einzelner Module), andere Teile müssen aber komplett entfernt werden (wie z.B. Skripte und Informationen zu nicht mehr angebotenen Modulen).
- Durch den Wegfall bisheriger und die Aufnahme neuer Gliederungspunkte kann die bisherige Menüstruktur nicht beibehalten werden.
- Da im Intranetbereich zukünftig auch die Möglichkeit zur Kooperation von Seminar- oder anderen Arbeitsgruppen gegeben sein soll, stellt es für den Nutzer eine große Erleichterung dar, wenn es zukünftig möglich ist, bevorzugte Seiten (z.B. Skripte und Informationen zu den momentan durch einen Studierenden belegten Modulen) durch Setzen eines Bookmarks abspeichern zu können. Dies spricht eindeutig gegen den Einsatz von Frame-basierten Webseiten.

Da dieses Intranet im Gegensatz zu dem öffentlich zugänglichen Internet mehr Arbeitsinstrument als Repräsentationsobjekt sein soll, sich also die Form weitgehend der Funktion unterordnen soll, wurden bei dessen Erstellung die gleichen Grundsätze befolgt, die bei der Erstellung des ursprünglichen Intranets durch die Seminargruppe im Sommersemester 2001 angewendet wurden. Diese wurden bereits in Kapitel II.B.2. auf Seite 6 dargestellt.

B. Konkretisierung der Anforderungen

Nachfolgend werden die in den Kapiteln III.A.1. für das Internet und III.B.1. für das Intranet formulierten Wünsche und Anregungen genauer interpretiert und hinsichtlich ihrer Realisierbarkeit überprüft. Dabei werden

einzelne Teilbereiche definiert, die als Arbeitsschritte bei der technischen Umsetzung herangezogen werden können. Somit entsteht für jede dieser einzelnen Applikationen ein klares Anforderungsprofil - eine Art Pflichtenheft für den gewünschten Leistungsumfang der einzelnen beschriebenen Funktionen.

1. Internetbereich

a. Newsletter - Abonnement

- Das Abonnieren des Newsletters durch Interessierte am Studiengang soll einfach direkt über die Studiengangs-Website möglich sein.
- Um unbeabsichtigte oder vorsätzliche Falschangaben zu minimieren, bekommt jeder Abonnent direkt nach dem Eintragen zu Kontrollzwecken eine E-Mail an die eingetragene E-Mail-Adresse.
- Um das Löschen von Abonnenten durch Unberechtigte zu vermeiden, bekommt jeder Abonnent in der Kontroll-Mail auch einen Link zugeschickt, durch dessen Aufruf er sich aus dem Verteiler austragen kann. Um Missbrauch weitestgehend auszuschließen, soll dieser Link so zusammengesetzt sein, dass ihn Dritte ohne Ansicht der Kontroll-Mail nicht nachvollziehen können.
- Für die Manager der Website (Studiengangsleiter, Webmaster) sollen weitere Administrationsfunktionen zur Verfügung stehen, z.B. die Ansicht der Abonnenten-E-Mail-Adressen oder das manuelle Löschen von Abonnenten.
- Der Versand der Newsletter soll auch über eine WWW-Oberfläche standortungebunden möglich sein. Der Zugriff auf diese Seite muss auf die Administratoren beschränkt sein.
- Aus Gründen des Datenschutzes sollen die einzelnen E-Mail-Adressen der Abonnenten nicht in der Newsletter-Mail erscheinen.

- Zur Kontrolle des Mailvorgangs sollen der Studiengangsleiter und der Webmaster zusätzlich zu den Adressaten aus dem Verteiler eine Mail mit dem Newsletter erhalten.

b. Newsletter - Archivierung

- Gleichzeitig mit dem Versand eines neuen Newsletters soll auch eine Kopie davon in einem Archiv abgelegt werden, auf das über die Studiengangs-Website auch "Nicht-Abonnenten" Zugriff auf die Newsletter haben.
- Dieses Archiv sammelt automatisch fortlaufend alle versendeten Newsletter.

2. Intranetbereich

Wie beim bisherigen Intranetangebot bleibt auch weiterhin der Zugang zum Intranet nur den Studierenden und Lehrenden des Master-Studiengangs Bibliotheks- und Medienmanagement vorbehalten.

a. Skriptearchiv

Das Skriptearchiv soll als zentrale Sammelstelle für alle offiziellen (also durch die Lehrenden verwendeten oder autorisierten) Materialien und Quellen dienen.

- Der Abruf von Quellen soll durch alle authentifizierten Nutzer im Intranet möglich sein.
- Das Einstellen und Löschen von Material soll nur durch Lehrende oder durch Administratoren erfolgen können. Studierende sollen explizit nicht die Rechte zum Einstellen von Skripten besitzen, da bei dem Einstellen durch Studierende nicht davon ausgegangen werden kann, dass in jedem Fall die Zustimmung des Dozenten zur Veröffentlichung vorliegt. Diese ist aber zwingend notwendig, da der Dozent letztendlich das Urheberrecht über seine Erzeugnisse inne hat.

- Neben dem Einstellen von Dateien soll auch das Veröffentlichen anderer Information, wie z.B. Bemerkungen, Literaturangaben, URLs o.ä., möglich sein.

b. Bereich für Arbeitsgruppen (Upload- / Download-Bereich)

Im Upload-/Download-Bereich sollen alle authentifizierten Nutzer Dateien zum freien Dateiaustausch oder zum Zwischenlagern deponieren können.

- Das Einstellen und Herunterladen von Dateien soll unkompliziert und einfach erfolgen können.
- Als technische Basis soll nur das WWW dienen. Andere Protokolle, wie z.B. FTP, sollen nicht verwendet werden.
- Schreibender und lesender Zugriff soll durch alle authentifizierten Nutzer erfolgen können.
- Ausnahme: Das Löschen von Dateien soll nur demjenigen Nutzer vorbehalten sein, der auch die Datei in diesen Bereich eingestellt hat.

c. Upload-Bereich zur Abgabe von Prüfungsleistungen

Um den Studierenden die Abgabe von Prüfungs- und Prüfungsvorleistungen ohne Anwesenheit an der Hochschule zu ermöglichen, soll die neue Website eine entsprechend gesicherte Upload-Möglichkeit zur Verfügung stellen. Nach erfolgter Korrektur kann die Arbeit durch den betreuenden Dozenten für alle Intranet-Benutzer zugänglich gemacht werden.

- Der Upload soll unkompliziert über eine WWW-Oberfläche erfolgen können.
- Zeitgleich mit dem erfolgtem Upload soll der zuständige Dozent per E-Mail benachrichtigt werden.
- Die Daten sollen nach dem Upload vorerst nur durch den betreuenden Dozenten abrufbar sein.

- Der betreuende Dozent hat, bei ausreichender Prüfungsleistung, die Möglichkeit, die Sperrung der Datei aufzuheben und diese dadurch im Hausarbeiten-Archiv öffentlich zugänglich zu machen.

d. Studierendenprofile (inkl. Alumni)

Um schnelle und zielgerichtete Kontakte zwischen den Studierenden zu ermöglichen, soll das schon vorhandene Verzeichnis aller Studierenden dieses Studiengangs ausgebaut werden.

- Neben Name und Kommunikationsdaten sollen auch der Arbeitgeber und die Position des Studierenden in dem Betrieb vermerkt werden können.
- Ehemalige Studierende sollen nach dem gleichen Schema in einem Alumni-Verzeichnis erfasst werden.
- Auch soll die Möglichkeit gegeben sein, ein Bild einzustellen.

Die mehrfach von Seite der Studierenden vorgeschlagene zwangsweise Veröffentlichung dieser Informationen⁴⁹ im Intranet des Studiengangs durch die Hochschule lässt sich aus juristischen Gründen so nicht verwirklichen. Bei der Verarbeitung personenbezogener Daten "gelten für alle Universitätsrechenzentren und sonstigen öffentlich-rechtlichen Forschungseinrichtungen in Landesträgerschaft die jeweiligen landesrechtlichen Datenschutzvorschriften, insbes. die Landesdatenschutzgesetze [...]"⁵⁰. Nach §4, (1) Landesdatenschutzgesetz ist die "Verarbeitung personenbezogener Daten [...] nur zulässig, 1. wenn dieses Gesetz oder eine andere Rechtsvorschrift sie erlaubt oder 2. soweit der Betroffene eingewilligt hat"⁵¹. Da aber weder das Landesdatenschutzgesetz noch eine andere Rechtsvorschrift die zwangsweise Erhebung

⁴⁹ Diese Anforderung wurde sowohl in der schriftlichen Rückmeldung von Frau Dorothee Boeckh vom 10.10.2003 als auch in dem von Herrn Thomas Becker angefertigten Protokoll der Flipchart-Notizen der Seminargruppe formuliert.

⁵⁰ Checkliste für Rechenzentren. <http://www.dfn.de/content/beratung-weiterbildung/recht/imdfn/checkliste>

⁵¹ Gesetz zum Schutz personenbezogener Daten (Landesdatenschutzgesetz LDSG). <http://www.baden-wuerttemberg.datenschutz.de/Home/Recht/LDSG/ldsgp.zip>, S. 3.

von personenbezogenen Daten für den hier angestrebten Zweck - nämlich das Erstellen ausführlicher Studierendenprofile - berücksichtigt, kann die Erhebung der Studierendendaten allein auf Basis des o.g. Satzes 2 erfolgen. Zur Erhebung der Studierendendaten wird also zwingend die Einwilligung der Betroffenen benötigt.

Auch der Umstand, dass die erhobenen personenbezogenen Daten in diesem Fall nur dargestellt, ansonsten aber nicht weiter verarbeitet werden, ändert nichts an der juristischen Sachlage. Laut §3, (2), 4., b) Landesdatenschutzgesetz umfasst der Begriff der Verarbeitung auch die Bereithaltung personenbezogener Daten zur Einsicht oder zum Abruf durch Dritte⁵². Somit muss festgestellt werden, dass der Hochschule keine Handhabe zur Verfügung steht, mit der die Angabe personenbezogener Daten durch die Studierenden erzwungen werden kann. Die Erstellung solcher differenzierter Studierendenprofile kann also nur auf der freiwilligen Angabe der Daten beruhen. Dies bedeutet:

- Alle im Intranet eingetragenen Studierenden oder Alumni müssen ihren Datensatz selbst erstellen, ändern und auch löschen können.

e. Dozentenprofile (inkl. Lehrbeauftragte)

Da sich bereits jetzt auf der Website des Master-Studiengangs, auf der zentralen Website der Hochschule und auf der Website des Fachbereichs Information und Kommunikation mehrere auf freiwilligen Angaben beruhende Profile von Dozenten befinden, kann davon ausgegangen werden, dass mit der Renovierung der Dozentenprofile auf der neuen Studiengangs-Website nicht mit juristischen Problemen gerechnet werden muss. Grundsätzlich gilt aber auch bei den Dozenten und bei den Lehrbeauftragten: Die Angabe personenbezogener Daten muss, wie bei den Studierenden auch, auf Freiwilligkeit beruhen.

⁵² Nach: Ebd., S. 3.

f. Interner Infodienst

Auf der Startseite des Intranets soll eine Zusammenstellung von Terminen, Fristen, Hinweisen und sonstiger wichtigen Informationen angezeigt werden.

- Das Einstellen neuer Meldungen soll möglichst einfach durch Webmaster und/oder Studiengangsleitung erfolgen können.
- Die Anzeige der Meldungen ausschließlich auf der Startseite des Intranets ist ausreichend. Sie muss nicht auf den untergeordneten Seiten des Intranets fortgeführt werden.
- Nach Ablauf der Gültigkeit einer Meldung soll diese sich selbsttätig aus dem Anzeigebereich für aktuelle Meldungen entfernen.

C. Technische Umsetzung der neuen Funktionen

1. Internetbereich

a. Newsletter - Abonnement

Die in Kapitel VI.B.1.a., Seite 35 definierten Anforderungen wurden umgesetzt, indem die Zope-eigene Datenbank ZGadfly verwendet wurde. Diese Datenbank wird bei jeder Installation von Zope automatisch mit installiert. Sie verwendet die Abfragesprache ZSQL, eine stark vereinfachte Form von SQL⁵³. In dieser Datenbank wurde eine neue Tabelle erstellt, in der die E-Mail-Adressen der Abonnenten zusammen mit einer eindeutigen ID abgespeichert werden. Die Eingabe der E-Mail-Adressen wird über eine Website mit einem Eingabeformular vorgenommen. Zu Kontrollzwecken geht eine Bestätigungsmail an die eingegebene E-Mail-Adresse, in der eine URL enthalten ist, durch deren Aufruf der Newsletter abbestellt werden kann. Von den Administratoren kann eine eigene Website aufgerufen werden, die in einer tabellarischen Übersicht alle Abonnenten anzeigt und das Lö-

⁵³ Eine Beschreibung von ZSQL und den wichtigsten ZSQL-Abfragen ist zu finden auf: Gadfly SQL constructs. Version: 1.2. <http://gadfly.sourceforge.net/sql.html>

schen einzelner Abonnenten über einen Link zulässt. Über eine weitere Website, die nur Administratoren zugänglich ist, werden die Newsletter verschickt. Hierzu muss in die beiden Textfelder auf der Website der gewünschte Betreff und der Newslettertext eingegeben werden. Der Newsletter wird als reine Textnachricht verschickt, da Nachrichten in reinem Textformat von allen E-Mail-Programmen angezeigt werden können. Webmaster und Studiengangleitung erhalten zu Kontrollzwecken eine Kopie des Newsletters per E-Mail.

Arbeitsschritte:

- Erstellen einer neuen Datenbankverbindung⁵⁴ im Root über das “Available-Objects-Menue”.
- Aufrufen der Datenbankverbindung und Erzeugen einer neuen Tabelle `newsletter_email` mit den beiden Spalten `nlid` und `email` im Format `character`⁵⁵.

Eintragen neuer Abonnenten:

- Erstellen einer DTML-Methode `eingabeformular`⁵⁶, die im Bereich “Lehrangebote” der Website verlinkt wird und von Abonnements-Interessenten zum Eingeben ihrer E-Mail-Adresse aufgerufen werden kann. `eingabeformular` enthält ein Formularfeld, aus dem die eingegebene E-Mail-Adresse ausgelesen wird, daneben wird aus den aktuellen Angaben zu Datum und Uhrzeit des Servers eine 14-stellige ID generiert. Diese beiden Werte (E-Mail-Adresse und ID) werden an die nachfolgend beschriebene ZSQL-Methode `speichern` übergeben. Gleichzeitig wird aus diesen beiden Werten der Link zum Abbestellen des Newsletters generiert, der, zusammen mit dem in der DTML-Methode `eingabeformular` enthaltenen Benachrichtigungstext, an die eingegebene E-Mail-Adresse verschickt wird. Bei einer leeren Eingabe wird eine Fehlermeldung ausgegeben. Außerdem enthält diese Datei einen versteckten Link

⁵⁴ Name: `dbverbindung_newsletter`, Position: / (Root)

⁵⁵ Befehl: `create table newsletter_email (nlid varchar, email varchar)`

⁵⁶ Name: `eingabeformular`, Position: `/lehrangebot/newsletter/`

auf den passwort-geschützten Administrationsbereich. (Siehe nachfolgende Abbildung Nr. 3).

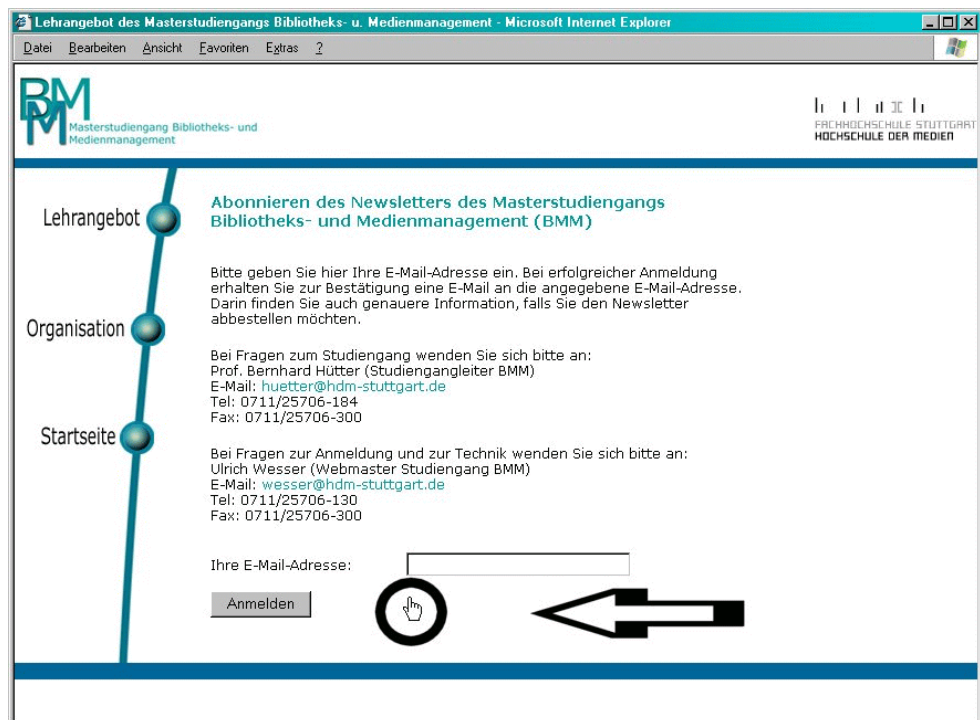


Abb. 3: Position des versteckten "hot spot" mit dem Link zur Newsletter-Administration

- Die ZSQL-Methode `speichern`⁵⁷, übernimmt die beiden Werte (ID und Email) aus dem Eingabeformular und schreibt diese in die Tabelle der Datenbank.

Löschen einer E-Mail-Adresse durch den Abonnenten selbst:

- Der Aufruf der dem Abonnenten per E-Mail mitgeteilten URL startet die DTML-Methode `austragen_html`⁵⁸, die die nachfolgend beschriebene ZSQL-Methode `austragen` aufruft und dem Abonnenten eine Bestätigung anzeigt.

⁵⁷ **Name:** `speichern`, **Position:** `/lehrangebot/newsletter/`, **Inhalt:** `insert into newsletter_email values (<dtml-sqlvar nlid type="string">,<dtml-sqlvar email type="string">)`

⁵⁸ **Name:** `austragen_html`, **Position:** `/lehrangebot/newsletter/`

- Die ZSQL-Methode `austragen`⁵⁹ löscht den entsprechenden Datensatz aus der Datenbank-Tabelle.

Anzeigen und Löschen von E-Mail-Adressen durch Administratoren:

- Der Administrationsbereich kann durch Anklicken des “hot spot” (siehe Abbildung Nr. 3) auf dem Eingabeformular aufgerufen werden. Es kann aber auch im Browser direkt die URL `http://bmm2.iuk.hdm-stuttgart.de/lehrangebot/newsletter_admin` eingegeben werden. Von hier aus führt ein entsprechend beschrifteter Link zur nachfolgend beschriebenen DTML-Methode `loeschen_html`.
- Die DTML-Methode `loeschen_html`⁶⁰ listet durch den Aufruf der nachfolgend beschriebenen ZSQL-Methode `ausgabe` die komplette Abonnenten-Datenbank auf. Durch Links, die neben jeder E-Mail-Adresse erscheinen und die die nachfolgend beschriebene ZSQL-Methode `loeschen` aufrufen, können einzelne Einträge aus der Datenbank gelöscht werden.
- Die ZSQL-Methode `ausgabe`⁶¹ liest beim Aufruf alle eingetragenen Daten aus der Datenbank aus.
- Die ZSQL-Methode `loeschen`⁶² löscht die in der vorher beschriebenen DTML-Methode `loeschen_html` angeklickte E-Mail-Adresse aus der Datenbank.

Versand von Newslettern:

- Über die Newsletter-Administrations-Website ist auch die DTML-Methode `versand_html`⁶³ verlinkt. Sie enthält zwei Textfelder, in

⁵⁹ **Name:** `austragen`, **Position:** `/lehrangebot/newsletter/`, **Inhalt:** `delete from newsletter_email where nlid=<dtml-sqlvar loesch_id type="string"> and email=<dtml-sqlvar email type="string">`

⁶⁰ **Name:** `loeschen_html`, **Position:** `/lehrangebot/newsletter/admin/`

⁶¹ **Name:** `ausgabe`, **Position:** `/lehrangebot/newsletter/admin/`, **Inhalt:** `select * from newsletter_email`

⁶² **Name:** `loeschen`, **Position:** `/lehrangebot/newsletter/admin/`, **Inhalt:** `delete from newsletter_email where nlid=<dtml-sqlvar loesch_id type="string"> and email=<dtml-sqlvar email type="string">`

⁶³ **Name:** `versand_html`, **Position:** `/lehrangebot/newsletter/admin/`

die der Betreff und der Text des neuen zu versendenden Newsletters eingegeben werden kann.

- Beim Absendevorgang werden die DTML-Methode `verteiler`⁶⁴ und das Python-Script `parseText`⁶⁵ ausgewertet.
- Die DTML-Methode `verteiler` bekommt alle gespeicherten E-Mail-Adressen aus der Datenbank von der ZSQL-Methode `ausgabe` übergeben, ergänzt dazu noch die E-Mail-Adresse des Webmasters und generiert aus diesen Angaben das BCC-Feld (BlindCarbonCopy) der Newsletter-Mail. Dadurch wird die Angabe der einzelnen E-Mail-Adressen in der Newsletter-Mail unterdrückt.
- Vor der Übergabe an den Mail-Server durchläuft der E-Mail-Text noch das Python-Script `parseText`, das nach maximal 72 Zeichen einen Zeilenumbruch durchführt, dabei aber einzelne Worte nicht durchtrennt. Dadurch erscheint der Newsletter-Text in jedem E-Mail-Programm mit sauber durchgeführten Zeilenumbrüchen.
- Außerdem wird die nachfolgend beschriebene DTML-Methode `eintragErzeugen` aufgerufen, die den Newsletter als neue DTML-Datei in das Newsletter-Archiv schreibt.

b. Newsletter - Archivierung

Das Newsletter-Archiv füllt sich automatisch, da durch den Absendevorgang durch `versand_html` der neue Newsletter gleichzeitig auch im Archivbereich gespeichert wird.

- Nach dem Absenden einer neuen Newsletter-Mail durch `versand_html` wird das Python-Script `eintragErzeugen`⁶⁶ automatisch aufgerufen.

⁶⁴ Name: `verteiler`, Position: `/lehrangebot/newsletter/admin/`

⁶⁵ Nach: Wrapping email at 72 characters. <http://www.zopelabs.com/cookbook/1028832719>

⁶⁶ Name: `eintragErzeugen`, Position: `/lehrangebot/newsletter/archiv/`

- Dieses Python-Script schreibt den Newsletter-Text und den Newsletter-Betreff in eine DTML-Datei, generiert eine eindeutige ID als Dateinamen und legt diese Datei in dem Verzeichnis `archiv` ab.
- Die im Verzeichnis `archiv` liegende `index_html` ist auf der Website als “Newsletter-Archiv” verlinkt. Beim Aufrufen dieser Datei wird die DTML-Methode `view`⁶⁷ gestartet.
- Die DTML-Methode `view` durchsucht das Verzeichnis `archiv` und liefert den Inhalt und den Namen von jeder gefundenen DTML-Datei zur Darstellung an die Datei `index_htm`.

2. Intranetbereich

Zur Authentifizierung der sich am Intranet anmeldenden User sind grundsätzlich zwei Methoden denkbar: die Authentifizierung der Nutzer über die Anbindung an bereits an der Hochschule vorhandene Authentifizierungsverfahren sowie die Authentifizierung der Nutzer durch lokal auf dem Zope-Server vorgehaltene Nutzerdaten. Die Anbindung an externe Authentifizierungsverfahren ist prinzipiell realisierbar, da für Zope Produkte existieren, die die Anbindung sowohl an einen LDAP-Server als auch an einen WindowsNT-Domänen-Server ermöglichen. Leider war zum Zeitpunkt der Erstellung dieser Arbeit der an der HdM vorhandene WindowsNT-Domänen-Server aufgrund einer Inkonsistenz der Nutzerdatenbank so stark beschädigt, dass eine Anbindung des Zope-Servers an diese NT-Domäne nicht mehr möglich war. Die Installation einer neuen WindowsNT-Domäne wurde zwar eingeleitet, war jedoch noch nicht so weit fortgeschritten, dass die Anbindung des Zope-Servers an diese neue WindowsNT-Domäne erfolgreich getestet werden konnte. Dieser Schritt kann jedoch problemlos nach Fertigstellung der neuen WindowsNT-Domäne nachgeholt werden. Selbst im Produktivbetrieb wäre eine solche Umstellung noch mit tolerierbarem Aufwand zu bewältigen, da es sich bei den Studierenden und Lehrenden des Master-Studiengangs Bibliotheks-

⁶⁷ Name: `view`, Position: `/lehrangebot/newsletter/archiv/`

und Medienmanagement nur um eine begrenzte Anzahl an Personen handelt.

Bei der Entscheidung für eine der beiden genannten Möglichkeiten ist besonders die gewünschte Rolle der Studierenden nach Abschluss ihres Studiums zu beachten. Wenn das Intranet grundsätzlich auch den Alumni zur Verfügung stehen soll, dann muss bei der Anbindung der Benutzerauthentifizierung an einen Server des Fachbereichs gewährleistet sein, dass in dessen Nutzerverwaltung Alumni als langfristige Nutzer angelegt sind. Sollte die dauerhafte Berücksichtigung von Alumni des BMM-Studiengangs in den Benutzerdaten des Fachbereichs nicht gewünscht oder geplant sein, so sollte die Benutzerauthentifizierung und -verwaltung dauerhaft lokal auf dem Zope-Server vorgenommen werden. Für die Benutzer selber entstehen dadurch keinerlei Nachteile. Beim Anlegen neuer Benutzer durch den Webmaster sollten sinnvollerweise die selben Login-Daten (Namenskürzel und Passworte) wie im Novell-Netz verwendet werden. Jeder BMM-Intranet-Benutzer hat, wie auch im Novell-Netz oder in der NT-Domäne, die Möglichkeit, sein Passwort selbst zu ändern. Dies wurde durch die Einbindung von zwei DTML-Methoden namens `change` und `changePassAction`⁶⁸ möglich, die dieses Problem behandeln und unter <http://www.zope.org> zum Download angeboten werden. Der Aufruf dieser Option wurde in den Menüpunkt "Administration" auf der Intranet-Startseite integriert, so dass alle Nutzer problemlos von dem vorgegebenen Passwort zu einem gewohnten Standard-Passwort wechseln können. Dadurch wäre das für die Studierenden wichtigste Argument für eine Anbindung des Zope-Servers an die NT-Domäne oder den LDAP-Server des Fachbereichs - nämlich das Vermeiden von unterschiedlichen Passwörtern - entkräftet.

Für den Intranetbereich musste eine neue Gestaltung und Navigation entworfen und realisiert werden, da die bisherige Navigation nur sehr aufwändig und kompliziert um die neuen Menüpunkte hätte erweitert

⁶⁸ How to allow users to change their own password (mini howto). <http://www.zope.org/Members/msx/ChangeOwnPassword-mini-howto>.

werden können. Das durch den Verfasser neu erstellte Menü wurde in den Header⁶⁹ und Footer⁷⁰ eingefügt. Die Inhalte dieser beiden Dateien werden beim Aufruf aller DTML-Dateien an deren Anfang bzw. Ende eingefügt und angezeigt. Somit müssen in den inhaltstragenden Dateien keine eigenen Gestaltungsmerkmale mehr definiert werden.

Das Intranet ermöglicht grundsätzlich zwei Einstiegswege. Zum einen werden im Startmenü des Intranets alle potenzielle Informationsquellen aufgeführt. So können direkt alle Skripte, alle Studierenden, alle Alumni, etc. durch einen Mausklick angezeigt werden. Zum anderen ist aber auch der Einstieg über ein bestimmtes Modul möglich. Innerhalb der Informationen zu einem bestimmten Modul können dann alle Studierenden und alle Skripte zu diesem Modul angezeigt werden. Um diesen Weg zu ermöglichen, wurden - wie in Kapitel V.B.2. ab Seite 30 erwähnt - alle Module inklusive aller dazu gehörenden Lehrveranstaltungen als ein eigenes Zope-Produkt angelegt. Vor der eigentlichen Erstellung dieses Produktes sind aber noch kleinere Vorarbeiten notwendig. Sowohl die Module als auch die Lehrveranstaltungen sind durch eindeutige Kürzel gekennzeichnet: die Module durch eine Abkürzung (MU, IA, ...), die fünf Lehrveranstaltungen pro Modul durch je eine Ziffer zwischen 1 und 5. Damit diese Angaben beim Anlegen neuer Objekte als PullDown-Menue zur Verfügung stehen, müssen im Root von Zope zwei zusätzliche sog. lines-properties angelegt werden, mit den Namen `modul_abkuerzung` und `lv_nr`. Auf diese Line-properties wird dann innerhalb des Produktes `modul` bzw. `lehrveranstaltungen` Bezug genommen, sodass zur Eingabe bei den Modulen nur die Werte zur Verfügung stehen, die in der lines-property `modul_abkuerzung` definiert sind. Die Erstellung des Produktes `modul` und der darin enthaltenen ZClass `modul` sowie aller nachfolgend beschriebenen Produkte wurde durch den Verfasser anhand der Beispiele durchgeführt, die im Titel "Zope - Content management & Web-Application-Server" im Kapitel 6.3 (Klassen und Objekte), Kapitel

⁶⁹ Name: `standard_html_header`, Position: `/bmm/intranet/`

⁷⁰ Name: `standard_html_footer`, Position: `/bmm/intranet/`

6.4 (Eine ZClass erstellen) und Kapitel 6.5 (Arbeiten mit der ZClass) ausführlich beschrieben werden⁷¹.

Die einzelnen Arbeitsschritte zum Erstellen des Produktes `modul` sind:

- Im Root von Zope: Anlegen einer property `modul_abkuerzung`⁷² vom Typ `lines` mit den Werten `BWL`, `DB`, `IA`, `KM`, `MU`, `WM`, `DMB`, `MIW`.
- Anlegen einer property `lv_nr`⁷³ vom Typ `lines` mit den Werten `1`, `2`, `3`, `4`, `5`.

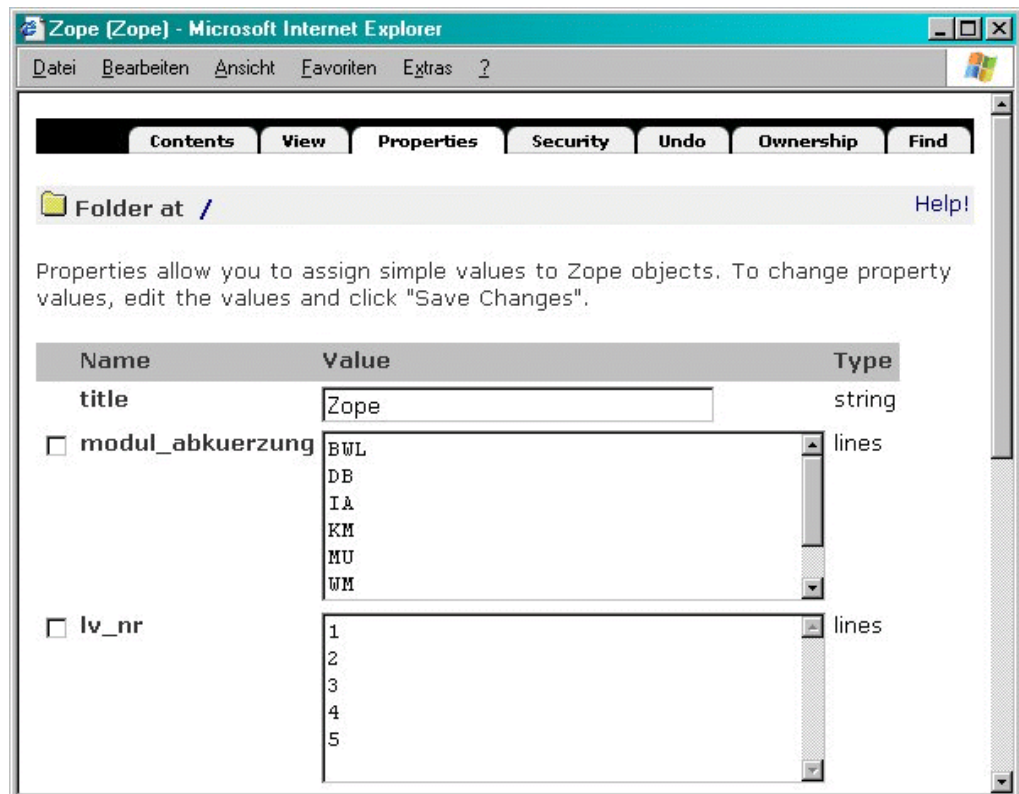


Abb. 4: Definieren der lines-properties im Root der Zope-Installation

Danach kann das eigentliche Produkt `modul` angelegt werden:

- Aufrufen des `Control_panels` (befindet sich im Root einer jeden Zope-Installation).
- Aufrufen des `Product management`.

⁷¹ Nach: Zope. Content-Management- & Web-Application-Server, S. 172 - 221.

⁷² Name: `modul_abkuerzung`, Position: / (Root), Reiter: `properties`

⁷³ Name: `lv_nr`, Position: / (Root), Reiter: `properties`

- Über Schaltfläche "Add Product" : Erstellen eines neuen Produktes namens `modul`⁷⁴.
- Aufrufen des Produktes `modul`. Hinzufügen einer ZClass `modul`⁷⁵.
- Aufrufen der ZClass `modul`.
- Ergänzen eines titles (beschreibender Name) und eines Metatypes. Der Metatype wird später im "Available objects"-Menü in Zope als Name der ZClass angezeigt, von dem neue Objekte (in diesem Fall Module) abgeleitet werden können.
- Base classes hinzufügen. Über die Base classes wird definiert, ob Objekte dieser neuen Klasse von der Zope-eigenen Suchmaschine durchsucht werden sollen, ob diese neuen Objekte vom Typ `modul` Dateien enthalten können und, wenn ja, von welchem Dateityp. In diesem Fall mussten zwei Base classes angewählt werden: `Zcatalog: CatalogAware` (um diese Objekte für die Zope-Suchmaschine durchsuchbar zu machen) und `Zclasses: Object Manager` (weil die neuen Objekte der Klasse `modul` andere Objekte vom Typ `lehrveranstaltung` enthalten sollen).
- Das Häkchen bei dem Kontrollkästchen "Create Constructor objects?" muss aktiviert sein, da dadurch zwei DTML-Methoden, eine Formular-datei `modul_addForm`⁷⁶ und die Skriptdatei `modul_add`⁷⁷, die die Befehle zum Erzeugen eines neuen Objekts vom Typ `modul` enthält, automatisch erstellt werden.
- Dann muss noch festgelegt werden, welche Eigenschaften die zukünftigen Objekte dieser neuen Klasse besitzen sollen. Dies geschieht, indem in der ZClass in ein propertysheet die gewünschten properties eingetragen werden. Hier werden, ähnlich wie in einer Datenbank, Felder definiert und mit Namen versehen. Diese Felder bezeichnen die Attribute (properties), die ein zukünftiges Objekt vom Typ `modul` besit-

⁷⁴ Name: `modul`, Position: `/Control_panel/Products/`

⁷⁵ Name: `modul`, Position: `/Control_panel/Products/modul/`

⁷⁶ Name: `modul_addForm`, Position: `/Control_panel/Products/modul/`

⁷⁷ Name: `modul_add`, Position: `/Control_panel/Products/modul/`

zen kann. Im vorliegenden Fall der Module sind sinnvolle properties: das Kürzel des Moduls (verwendet die in der lines-property `modul_abkuerzung` definierten Inhalte), die Modulbezeichnung, eine genauere Beschreibung des Modulinhaltes, der Name des Modulverantwortlichen, die E-Mail des Modulverantwortlichen. Im propertysheet müssen eindeutige Namen ohne Leerzeichen verwendet werden. Daher wurde ein propertysheet⁷⁸ erstellt, das folgende properties enthält: `modul_kuerzel`, `title`, `modul_beschreibung`, `modul_sprecher`, `modul_sprecher_email`.

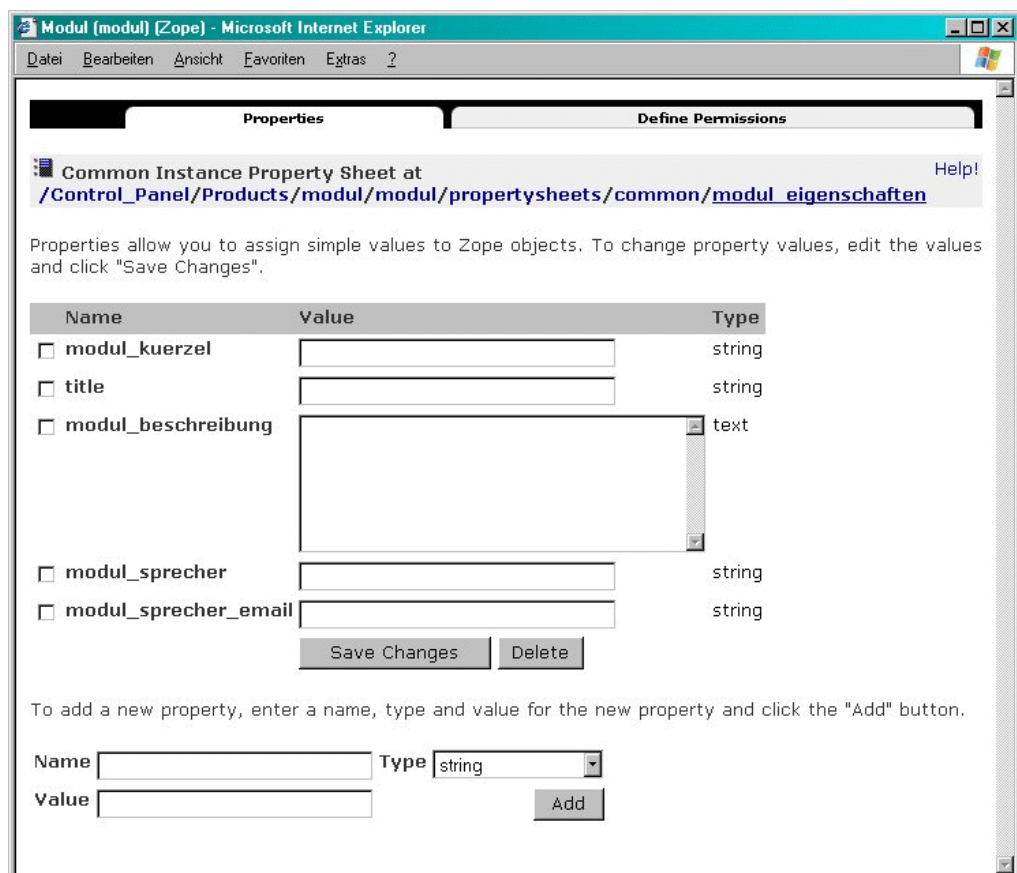


Abb. 5: Definition des property Sheets `modul_eigenschaften`

- Innerhalb des Produktes `modul` muss jetzt noch die automatisch erstellte DTML-Methode `modul_addform` angepasst werden. Dieses Eingabeformular erscheint auf dem Bildschirm, sobald ein neues

⁷⁸ Name: `modul_eigenschaften`, Position: `/Control_panel/Products/modul/modul/property sheets/common/`

Objekt vom Typ `modul` angelegt werden soll. Hierin können die Felder der einzelnen `properties` ausführlicher beschrieben werden. Außerdem wird in dieser DTML-Methode die Verbindung zwischen dem Feld `modul_kuerzel` im `property sheet` der `ZClass modul` und der `lines-property modul_abkuerzung` im `Root` hergestellt.

Als Nächstes muss dann die `ZClass lehrveranstaltung` angelegt werden. Da das Verfahren in den Grundzügen mit der Herstellung der `ZClass modul` identisch ist, wird es hier verkürzt dargestellt.

- Erstellen einer `ZClass lehrveranstaltung`⁷⁹.
- Darin: Auswahl der Base classes: `Zcatalog: CatalogAware` und `Zclasses: Object Manager`.
- Automatisches Erstellen der “constructor input forms” aktivieren.
- Erneutes Aufrufen der `ZClass lehrveranstaltung`: Erstellen eines `property sheets`⁸⁰ mit den Eigenschaften `title`, `beschreibung`, `dozent`, `dozent_email`, `lv_nummer` (verwendet die in der `lines-property lv_nr` definierten Inhalte).

⁷⁹ **Name:** `lehrveranstaltung`, **Position:** `/Control_panel/Products/modul/`

⁸⁰ **Name:** `lehrveranstaltung_eigenschaften`, **Position:** `/Control_panel/Products/modul/lehrveranstaltung/property sheets/common/`

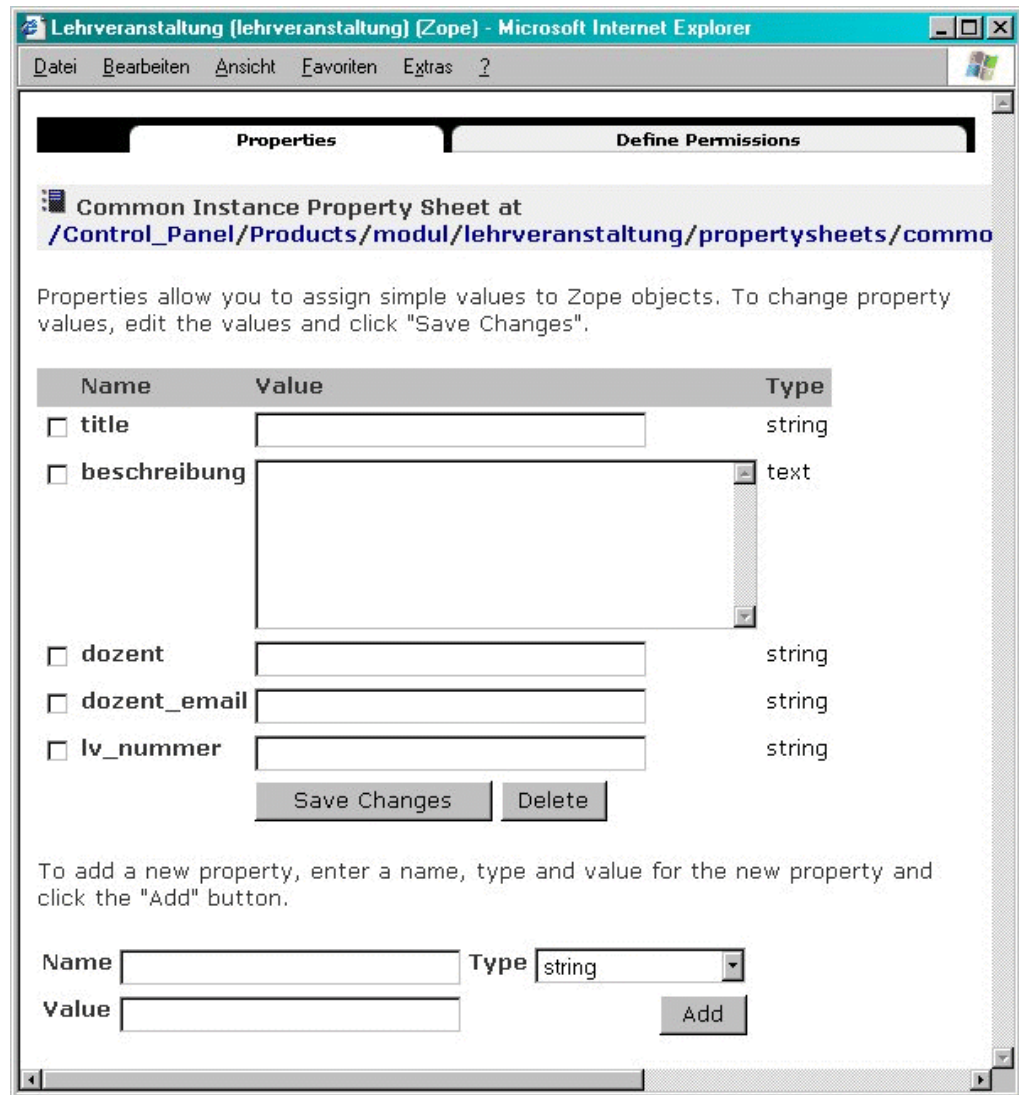


Abb. 6: Definition des property Sheets lehrveranstaltung_eigenschaften

- Innerhalb des Produktes modul muss noch die automatisch erstellte DTML-Methode lehrveranstaltung_addform⁸¹ angepasst werden (z.B. sinntragende Bezeichnung der Eingabefelder und sinnvolle Anordnung bzw. Reihenfolge), die zum Erstellen neuer Objekte der Klasse lehrveranstaltung verwendet wird.
- Aufrufen der ZClass modul: Im Reiter subobjects kann festgelegt werden, dass die ZClass lehrveranstaltung ein subobject des Produkts modul sein soll.

⁸¹ Name: lehrveranstaltung_addform, Position: /Control_panel/Products/modul/

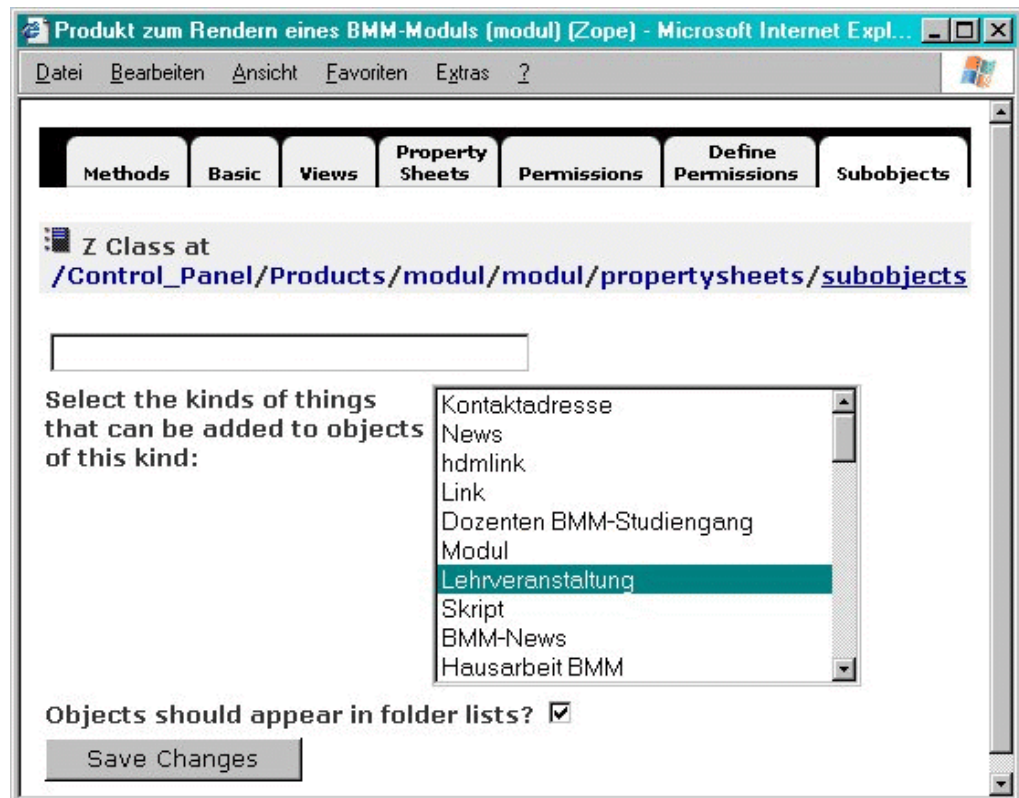


Abb. 7: Objekte vom Typ `lehrveranstaltung` werden als subobjects von Objekten des Typs `modul` definiert

Durch das Anlegen dieses Produktes ist es möglich, von jedem Standort innerhalb der Hierarchie der BMM-Website auf die Eigenschaften der Module (genauer gesagt: auf die Eigenschaften der Objekte, die von der Klasse “Modul” abgeleitet wurden) zuzugreifen. Da der Zugriff auf die Modulinformationen nicht nur aus dem Intranet, sondern sinnvollerweise von der ganzen BMM-Website aus erfolgen können muss, wurde der Ordner zum Sammeln der Modul-Objekte im BMM-Verzeichnis (also: im Root der BMM-Website) des Servers angelegt.

Die Anzeige der einzelnen Modulnamen auf der Eingangsseite des Intranets⁸² wird durch eine Abfrage realisiert, durch die die `title` aller Modul-Objekte ausgelesen und angezeigt werden. Die ausführliche Anzeige der Informationen zu einem Modul geschieht durch den Aufruf einer DTML-Methode `modul_view`⁸³, die eine “Suchschleife” über alle

⁸² Name: `index_html`, Position: `/bmm/intranet/`

⁸³ Name: `modul_view`, Position: `/bmm/`

dem betreffenden Modul zugeordneten Lehrveranstaltungen durchführt und die Eigenschaften aller zugeordneten “Lehrveranstaltungs-Objekte” anzeigt. Das gleiche Ergebnis wird erzielt, wenn von der Eingangsseite des Intranets der Link “Alle Module anzeigen” angewählt wird und von dort aus genauere Informationen zu einem bestimmten Modul angewählt werden. Hierzu wird auch die gleiche DTML-Methode `modul_view`⁸³ eingesetzt.

a. Skriptearchiv

Auch das Skripte-Archiv wird als eigenes Produkt in Zope erstellt. Für dieses Produkt müssen keine zusätzlichen Lines-properties im Root von Zope erstellt werden, da die für das Produkt `modul` erstellten properties `modul_abkuerzung` und `lv_nr` auch für das Produkt `skript` genutzt werden können. Da die einzelnen Arbeitsschritte beim Hinzufügen und Definieren eines neuen Produktes nahezu identisch sind, wird der Vorgang hier nur noch verkürzt wiedergegeben.

Die einzelnen Arbeitsschritte:

- Erstellen eines neuen Produktes namens `skript`⁸⁴.
- Darin: Erstellen einer ZClass `skript`⁸⁵.
- Ergänzen eines Meta types mit dem Namen `Skript`.
- Hinzufügen der Base classes: `Zcatalog: CatalogAware` und `Zclasses: OFS File` (weil die neuen Objekte von der Klasse `skript` andere Objekte, nämlich die eigentlichen Dateien mit den Skripten, enthalten sollen).
- “Constructor input forms” automatisch erstellen lassen.
- Aufrufen der Zclass `skript`. Darin: Erstellen eines `propertysheets`⁸⁶ mit den Eigenschaften `title`, `abstract`, `weitere_url`, `modul_kuerzel` (verwendet die in der lines-property `modul_`

⁸⁴ Name: `skript`, Position: `/Control_panel/Products/`

⁸⁵ Name: `skript`, Position: `/Control_panel/Products/skript/`

⁸⁶ Name: `skript_eigenschaften`, Position: `/Control_panel/Products/skript/skript/propertysheets/common/`

abkuerzung definierten Inhalte), `lv_nummer` (verwendet die in der lines-property `lv_nr` definierten Inhalte), `dozent`, `email_dozent`, `login_skript`, `filename`, `datum`.

- Innerhalb des Produktes `skript`: Anpassen der automatisch erstellten DTML-Methode `skript_addform`⁸⁷, die zum Hinzufügen neuer Objekte verwendet wird. In dieser DTML-Methode wird die Verbindung zwischen dem Feld `modul_kuerzel` im propertysheet der Zclass `skript` und der lines-property `modul_abkuerzung` im Root hergestellt. Ebenso wird die Verbindung zwischen dem Feld `lv_nummer` im propertysheet der Zclass `skript` und der lines-property `lv_nr` im Root hergestellt. Außerdem wird durch eine `<dtml-if>`-Schleife festgelegt, wie das Feld `login_skript` gefüllt wird. Ist dem angemeldeten Nutzer die Rolle “Dozent” zugeordnet, so wird das Feld `login_skript` mit der Login-Kennung des Nutzers gefüllt. Ist dem angemeldeten Nutzer die Rolle “Manager” zugeordnet, so kann das Feld `login_skript` frei ausgefüllt werden. Angehörige anderer Rollen haben keinen Zugriff auf die properties der Skript-Objekte.

Die Darstellung der eingestellten Skripte wird über zwei weitere DTML-Dateien geregelt.

- Die DTML-Datei `skripte_liste_view`⁸⁸ sucht mittels eines `<dtml-in>`-Befehls die Namen und Titel aller vorhandenen Skripte im Verzeichnis `/bmm/intranet/skripte` und listet die Angaben auf.
- Die DTML-Datei `skripte_lv_view`⁸⁹ wird von den Detailangaben zu den Lehrveranstaltungen aufgerufen und zeigt nur die Skripte an, die zu der ausgewählten Lehrveranstaltung gehören.

⁸⁷ Name: `skript_addform`, Position: `/Control_panel/Products/skript/`

⁸⁸ Name: `skripte_liste_view`, Position: `/bmm/intranet/`

⁸⁹ Name: `skripte_lv_view`, Position: `/bmm/intranet/`

Das Einstellen neuer Skripte ist Dozentinnen und Dozenten sowie Managern über zwei Verlinkungen möglich. Ein Link “Neues Skript einstellen” erscheint im Bereich “Administration”, ein zweiter Link “Neues Skript einstellen” erscheint am Anfang der Gesamtliste aller eingestellten Skripte. Beide Links sind nur dann sichtbar, wenn dem angemeldeten User entweder die Rolle “Dozent” oder die Rolle “Manager” zugewiesen wurde.

Etwas aufwändiger war die Realisierung des Verfahrens zum Löschen der Skripte durch den jeweils dafür verantwortlichen Dozenten. Da Skripte im Auftrag von Dozenten auch vom Webmaster eingestellt und gelöscht werden sollen, kann zur Identifizierung des Urhebers eines Skriptes nicht der “Besitzer” (Funktion: Owner) des betreffenden Objektes herangezogen werden. Deshalb wird hierzu das Feld `login_skript` abgefragt. Beim Einstellen eines Skriptes durch einen Dozenten, wird in das Feld `login_skript` automatisch die Login-Kennung des angemeldeten Nutzers eingetragen. Dieser Vorgang kann durch den einstellenden Dozenten nicht beeinflusst werden. Wenn ein Manager, wie z.B. der Webmaster, ein Skript im Auftrag eines Dozenten einstellt, bekommt der Webmaster die Möglichkeit, das Feld `login_skript` mit der Login-Kennung des auftraggebenden Dozenten manuell zu füllen, wodurch auch wieder eine eindeutige Zuordnung zwischen Dozent und Skript hergestellt wird. Da der Inhalt dieses Feldes durch Dozenten nicht manipuliert werden kann, ist es möglich, dieses Feld zur sicheren Identifikation des verantwortlichen Dozenten eines Skriptes zu verwenden.

Der Löschvorgang selbst wurde durch zwei weitere Objekte realisiert, einer DTML-Datei und einer DTML-Methode.

- In der DTML-Datei `skripte_liste_view` wird ein Link auf die Datei `skripte_liste_delete`⁹⁰ integriert, der nur für Nutzern den die Rolle “Manager” oder “Dozent” zugeordnet wurde, sicht-

⁹⁰ Name: `skripte_liste_delete`, Position: `/bmm/intranet/`

bar und ausführbar ist. Dieser Link auf die Datei `skripte_liste_delete` wird auch im Administrationsbereich aufgeführt - dort ebenfalls nur sichtbar für Angehörige der Rollen "Manager" und "Dozent". Wenn die DTML-Datei `skripte_liste_delete` durch einen Manager aufgerufen wird, so werden alle Skripte aufgelistet und mit einem Button versehen, durch den die DTML-Methode `delete_skripte`⁹¹ aufgerufen wird, die dann das entsprechende Skript löscht. Ruft ein Dozent diesen Link auf, so werden nur die Skripte aufgelistet, bei denen der Inhalt des Feldes `login_skript` mit der Login-Kennung des angemeldeten Nutzers identisch ist.

- Wenn bei einem Skript aus der durch `skripte_liste_delete` angezeigten Liste der "Löschen"-Button angeklickt wird, wird die ID des betroffenen Skriptes an die DTML-Methode `delete_skripte` übergeben.
- Die DTML-Methode `delete_skripte` enthält den eigentlichen Lösch-Befehl. Dieser wird aber nur ausgeführt, wenn dem aufrufenden Nutzer entweder die Rolle "Manager" zugeordnet ist, oder wenn die Login-Kennung des angemeldeten Nutzers mit dem Inhalt des Feldes `login_skript` des zum Löschen ausgewählten Skriptes identisch ist. Somit kann ein Dozent tatsächlich nur die von ihm publizierten, oder die ihm zugeordneten Skripte löschen. Auf Skripte von anderen Dozenten, kann die DTML-Methode `delete_skripte` nicht angewendet werden.

Nutzer, denen die Rolle "Student" zugewiesen wurde, besitzen im ganzen Verzeichnis `/bmm/intranet/skripte` kein Schreibrecht - und damit auch keine Berechtigung zum Löschen von Dateien.

⁹¹ Name: `delete_skripte`, Position: `/bmm/intranet/`

b. Bereich für Arbeitsgruppen (Upload- / Download-Bereich)

In jeder Zope-Installation wird automatisch ein Verzeichnis mit dem Namen `examples` angelegt. Um dem Anwender einen Eindruck von der Leistungsfähigkeit von Zope zu vermitteln, sind in diesem Verzeichnis Programmmodule gespeichert, die die Realisierung eines Diskussionsforums, eines Webshops und einer File library demonstrieren sollen. Da die dort gezeigte Anwendung der File library in weiten Teilen den Anforderungen entspricht, die an den einzurichtenden Datei-Upload-/Download-Bereich gestellt wurden, wurden die Programmteile dieser File library vom Verfasser übernommen und hinsichtlich der noch nicht erfüllten Anforderung - Löschen der Datei durch den Nutzer, der diese Datei auch eingestellt hat - angepasst.

Die einzelnen Arbeitsschritte:

- Kopieren der Dateien⁹² (mehrere Python-Script-Dateien, eine PageTemplate-Datei zur Darstellung) in das Verzeichnis `File Library` innerhalb des BMM-Intranets.
- Anpassen von Layout und Design in `index_html`.
- Ergänzen des Skripts `addFile`⁹³ und des PageTemplate-Files `index_html`⁹⁴ um folgende Funktion: Wenn neue Dateien in die File library eingestellt werden, bekommen diese Dateien in ihrem Attribut `title` den Namen des momentan angemeldeten, die Datei hochladenden Nutzers eingetragen.
- Die Möglichkeit des Löschens von Dateien durch den jeweils einstellenden Benutzer wurde durch eine neu erstellte DTML-Datei `liste`⁹⁵ und eine neu erstellte DTML-Methode `delete`⁹⁶ realisiert. Die DTML-Datei `liste` wird über einen Link innerhalb der `index_html` (erscheint beim Aufruf des Menüpunktes "Upload-/Down-

⁹² Name der Python-Skript-Dateien: `addFile`, `getFiles`, `getLink`, `getSize`. Name der PageTemplate-Datei: `index_html`

⁹³ Name: `addFile`, Position: `/bmm/intranet/FileLibrary/`

⁹⁴ Name: `index_html`, Position: `/bmm/intranet/FileLibrary/`

⁹⁵ Name: `liste`, Position: `/bmm/intranet/FileLibrary/`

⁹⁶ Name: `delete`, Position: `/bmm/intranet/FileLibrary/`

load-Bereich”) aufgerufen und listet alle Dateien auf, die von dem momentan angemeldeten Nutzer eingestellt wurden, indem eine Suchschleife den Namen des angemeldeten Nutzers mit der Eintragung im Feld `title` der gespeicherten Dateien abgleicht. Dadurch bekommt jeder Nutzer nur die Dateien angezeigt, die auch seinen Namen im Feld `title` tragen - also die Dateien, die er selbst eingestellt hat. Neben den Dateinamen erscheint ein Schaltfläche Löschen, die nach dem Anklicken die DTML-Methode `delete` aufruft. `delete` löscht die Datei mit dem ausgewählten Namen und gibt die aktualisierte Dateiliste wieder zurück an `liste`, die diese Dateiliste dann erneut ausgibt.

Da ein angemeldeter Nutzer weder seine Login-Kennung noch die Einträge in der Eigenschaft `title` der hochgeladenen Dateien beeinflussen kann, kann dieses Verfahren als sicher angesehen werden.

- Als letzte Maßnahme muss im Verzeichnis `files`⁹⁷, in dem die hochgeladenen Dateien abgespeichert werden, den Rollen `student` und `dozent` die Rechte zum Löschen von Dateien erteilt werden⁹⁸. Dieses Beispiel einer File Library wurde als Zope Page Template realisiert. Ein Page Template ist eine spezielle Art von Zope-Objekt, mit dessen Hilfe Design und Inhalt einer Website konsequent getrennt werden können. Komplexe Programmierungen können jedoch in einem Page Template nicht realisiert werden⁹⁹. Das Einpassen der als Page Templates realisierten File Library in die Website hat sich als etwas sperrig und unpraktisch herausgestellt, da in ein Page Template nicht ein vorhandener Footer oder Header durch den sonst üblichen `<dtml-var>`-Befehl eingebunden werden kann. Aus diesem Grund mussten das Navigationsmenü und die Gestaltungsmerkmale (z.B.

⁹⁷ Name: `files`, Position: `/bmm/intranet/FileLibrary/`

⁹⁸ Aufrufen des Verzeichnisses `/bmm/intranet/FileLibrary/Files/` in der Management-Konsole, Einstellen der Rechte über den Reiter `security`.

⁹⁹ Nach: Hörtzsch, Matthias; Rapko, Andreas: Content Management und Zope, S. 119.

auch das Einbinden des Style sheets) direkt in den Quellcode des Page Templates eingefügt werden. Bei eventuell notwendigen Änderungen in der im Header angesiedelten Navigation muss beachtet werden, dass sich die Navigation innerhalb dieses Page Templates nicht automatisch mit anpasst.

c. Upload-Bereich zur Abgabe von Prüfungsleistungen

Die Möglichkeit, Haus- oder Prüfungsarbeiten per Upload auf den Server dem zuständigen Dozenten zukommen zu lassen, wurde eng verknüpft mit der Erstellung eines Hausarbeiten-Archives, das weitgehend nach dem Muster des Skriptearchivs angelegt werden konnte. Das Konzept des Produktes `skript` wurde um folgende Faktoren erweitert:

- Bei der Herstellung des Produktes `hausarbeit`¹⁰⁰ erhält die ZClass `hausarbeit`¹⁰¹ in ihrem `property sheet`¹⁰² folgende properties: `title` (Titel der Arbeit), `abstract` (abstract), `absender` (Name des Autors), `email_absender` (E-Mail-Adresse des Autors), `modul_kuerzel` (Modul - greift auf die `lines-property` `modul_abkuerzung`¹⁰³ zurück), `lv_nummer` (Nummer der Lehrveranstaltung in dem die Arbeit erstellt wurde - greift auf die `lines-property` `lv_nr`¹⁰⁴ zurück), `dozent` (Name des betreuenden Dozenten), `email_dozent` (E-Mail-Adresse des betreuenden Dozenten), `filename` (Feld für den Dateinamen der Arbeit), `datum` (Datum des Hochladens) sowie eine weitere Eigenschaft mit der Bezeichnung `sichtbar`, die die Werte `gesperrt` oder `freigegeben`

¹⁰⁰ **Name:** `hausarbeit`, **Position:** `/Control_panel/Products/`

¹⁰¹ **Name:** `hausarbeit`, **Position:** `/Control_panel/Products/hausarbeit/`

¹⁰² **Name:** `hausarbeit_eigenschaften`, **Position:** `/Control_panel/Products/hausarbeit/hausarbeit/propertysheets/common/`

¹⁰³ **Name:** `modul_abkuerzung`, **Position:** `/ (Root)`

¹⁰⁴ **Name:** `lv_nr`, **Position:** `/ (Root)`

annehmen kann, welche in der zugeordneten `lines-property` `freigabe`¹⁰⁵ definiert wurden.

- Die DTML-Methode `hausarbeit_addForm`¹⁰⁶, über die Studierende neue Objekte vom Typ `hausarbeit` erstellen können, füllt die Eigenschaft `sichtbar` grundsätzlich mit dem Wert `gesperrt`. Die Vorgabe dieses Wertes kann durch Studierende nicht beeinflusst werden.
- Der Constructor¹⁰⁷ des Produktes `hausarbeit` wird um einen Absatz ergänzt, in dem mit Hilfe des `<dtml-sendmail>`-Befehls bei jedem Einstellen einer neuen Hausarbeit ein Text an den betreuenden Dozenten gemailt wird, in dem er von dem neuen Eintrag unterrichtet wird. Zu diesem Zweck werden die properties `dozent` und `email_dozent` ausgewertet.
- In dieser Mail bekommt der Dozent auch Name und E-Mail-Adresse des einstellenden Studierenden, das Kürzel des Moduls und die Nummer der Lehrveranstaltung in der die Arbeit geschrieben wurde, sowie den Titel der Arbeit mitgeteilt.
- Weiterhin bekommt der Dozent auch zwei URLs mitgeteilt: eine URL, über die er sich die Arbeit herunterladen kann, sowie eine weitere URL, die auf das `edit_propertiesForm`¹⁰⁸ des Produktes `hausarbeit` verweist. Durch Aufruf dieser zweiten URL kann der Dozent in diesem `edit_propertiesForm` die Eigenschaft `sichtbar` der von dem Studierenden neu eingestellten Arbeit von `gesperrt` auf `freigegeben` ändern.
- Die DTML-Datei `index_html`¹⁰⁹, die beim Anklicken des Menüpunktes “Hausarbeitenarchiv” im Intranet ausgeführt wird, überprüft durch einen `<dtml-if>`-Befehl, ob bei den enthaltenen Ob-

¹⁰⁵ Name: `freigabe`, Position: / (Root)

¹⁰⁶ Name: `hausarbeit_addForm`, Position: /Control_Panel/Products/hausarbeit/

¹⁰⁷ Name: `hausarbeit_add`, Position: /Control_Panel/Products/hausarbeit/

¹⁰⁸ Name: `edit_propertiesForm`, Position: /Control_Panel/Products/hausarbeit/hausarbeit/property sheets/methods/

¹⁰⁹ Name: `index_html`, Position: /bmm/intranet/hausarbeit/

jekten vom Typ "Hausarbeit" die Eigenschaft `sichtbar` auf `frei-`
`gegeben` gestellt wurde. Nur die Einträge, bei denen dies der Fall
ist, werden angezeigt. Solange also ein Dozent die an ihn gerichtete
Haus- oder Prüfungsarbeit nicht freigeschaltet hat, wird diese im
Hausarbeitenarchiv nicht angezeigt.

Dieses Verfahren bietet vor allem für die betroffenen Lehrkräfte eine
komfortable und unkomplizierte Möglichkeit, die mit einer ausreichen-
den Note bewerteten Hausarbeiten der BMM-Allgemeinheit zugäng-
lich zu machen. Allerdings muss darauf hingewiesen werden, dass auch
schon vor der Freigabe durch den betreuenden Dozenten die Arbeit
zumindest theoretisch auch für andere Lehrkräfte zugänglich ist. Der
Wert der Eigenschaft `sichtbar` regelt lediglich die Anzeige der Ar-
beit, nicht aber die Zugriffsrechte auf die Datei. Sollte also ein anderer
als der betreuende Dozent den Dateinamen der Arbeit - der dem be-
treuenden Dozenten in der automatischen Mail mitgeteilt wird - zur
Kenntnis bekommen, so kann dieser ebenfalls die Arbeit herunter-
laden, einsehen und durch Anpassung der zweiten in der Mail genann-
ten URL auch deren Eigenschaften verändern. In manchen Fällen ist
dies vielleicht sogar gewünscht, da so der betreuende Dozent die Mög-
lichkeit hat, durch einfaches Weiterleiten der Benachrichtigungsmail
einem Kollegen Zugriff auf die Datei zu geben - z.B. zur Übernahme
der Zweitkorrektur oder einer weiteren Begutachtung. Ein zufälliges
oder vorsätzliches Erraten des Dateinamens einer Arbeit scheint aber
weitgehend ausgeschlossen, da der Dateiname aus dem Systemdatum
und der Systemzeit zum Zeitpunkt des Einstellens generiert wird und
daher eine Vielzahl möglicher Kombinationen existiert.

d. Studierendenprofile (inkl. Alumni)

Im Gegensatz zum Produkt `skript` muss für das Produkt `student`
wieder eine zusätzliche Lines-property im Root von Zope erstellt wer-
den. Über diese lines-property mit dem Namen `student_status` und

den möglichen Werten `Student`, `Alumni`, `Löschen` wird später, bei der Auflistung aller Objekte vom Typ `student`, abgefragt, ob ein Objekt bei der Auflistung der aktiven Studierenden, der Alumni, oder auch gar nicht angezeigt werden soll.

Die einzelnen Arbeitsschritte:

- Erstellen eines neuen Produktes namens `student`¹¹⁰.
- Darin: Erstellen einer ZClass `student`¹¹¹.
- Eintragen des Metatypes mit dem Namen `Student`.
- Hinzufügen der Base classes: `Zcatalog: CatalogAware` und `Zclasses: OFS Image` (notwendig, da die neuen Objekte von der Klasse `student` auch Bilder enthalten sollen).
- “Constructor input forms” automatisch erstellen lassen.
- Innerhalb der ZClass `student`: Erstellen eines `propertysheets`¹¹² mit den Eigenschaften `title` (enthält den Nachnamen), `vorname`, `email` (für die FH-eigene E-Mail-Kennung), `status` (verwendet die in der `lines-property student_status` definierten Inhalte), `dienststelle` (für Bezeichnung und Adresse der Arbeitsstelle), `position` (für Beschreibung der Funktion), `module` (verwendet die in der `lines-property modul_abkuerzungen` definierten Inhalte), `email_dienstl`, `telefon_dienstl`, `telefax_dienstl`.
- Innerhalb des Produktes `student`: Anpassen der automatisch erstellten DTML-Methode `student_addform`¹¹³, die zum Hinzufügen neuer Objekte verwendet wird. (In dieser DTML-Methode wird die Verbindung zwischen dem Feld `module` im `propertysheet` der ZClass `student` und der `lines-property modul_abkuerzung` im Root sowie zwischen dem Feld `status` im `propertysheet` der

¹¹⁰ Name: `student`, Position: `/Control_panel/Products/`

¹¹¹ Name: `student`, Position: `/Control_panel/Products/student/`

¹¹² Name: `student_eigenschaften`, Position: `/Control_panel/Products/student/student/propertysheets/common/`

¹¹³ Name: `student_addform`, Position: `/Control_panel/Products/student/`

ZClass `student` und der lines-property `student_status` im Root hergestellt).

- Anders als bei den bisherigen selbst erstellten Produkten sollen bei dem Produkt `student` die Studierenden selbst ihre Datensätze (exakt: das von ihnen erstellte Objekt vom Typ `student`) editieren können. Deshalb ist es notwendig, für diese Funktion zwei weitere DTML-Methoden zu erstellen, die diesen Zweck erfüllen - vergleichbar den beiden Methoden `student_add` und `student_addForm`, die die Erstellung neuer Objekte vom Typ `student` steuern.

Diese beiden Methoden `edit_properties`¹¹⁴ und `edit_propertiesForm`¹¹⁵ befinden sich innerhalb der ZClass `student` und werden durch Anwählen des Reiters `Methods` sichtbar. Sie sind inhaltlich ähnlich aufgebaut wie die beiden Methoden `student_add` und `student_addForm` und unterscheiden sich nur in folgenden Punkten:

- Mit dem `edit_propertiesForm` können vorhandene Objekte editiert werden. Deshalb dürfen nicht, wie bei der Neueingabe mit dem `student_addForm`, leere Felder zu den Eigenschaften ausgegeben werden, statt dessen müssen die vorhandenen Werte der Eigenschaften ausgelesen und in dem Formular angezeigt werden.
- Sobald die Schaltfläche `Speichern` aktiviert wird, wird die Methode `edit_properties` aufgerufen, durch die die aktualisierten Werte des aufgerufenen Objekts abgespeichert werden. In dieser Methode wurde auch die Meldung "Ihre Änderungen wurden gespeichert" eingefügt, die, zusammen mit einem Link auf das aktualisierte Objekt, das damit direkt zur Kontrolle angezeigt werden kann, ausgegeben wird.

¹¹⁴ Name: `edit_properties`, Position: `/Control_Panel/Products/student/student/property sheets/methods/`

¹¹⁵ Name: `edit_propertiesForm`, Position: `/Control_Panel/Products/student/student/property sheets/methods/`

Die Neueingabe eines Studenten-Datensatzes wird über einen Link im Administrationsbereich (Link “Administration” auf der Startseite des Intranets) durchgeführt. Dieser Link erscheint nur, wenn der angemeldete Nutzer die Rolle “Student” zugewiesen bekommen hat. Der Link ruft das `student_addForm` aus dem Produkt `student` auf. Die Kennung des eingebenden Nutzers wird zusammen mit den neuen Daten als Besitzer dieses neuen Studierenden- Objekts abgespeichert. Die Anzeige der Daten der Studierenden kann über mehrere Wege erfolgen.

- Die DTML-Datei `student_liste_view`¹¹⁶ listet die Felder `title` und `vorname` (also: Nachname und Vorname) aller Studierenden auf, die als Status `Student` eingetragen haben. Diese Datei kann über den Link “Verzeichnis der Studierenden” auf der Startseite des Intranets aufgerufen werden.
- Die DTML-Datei `alumni_liste_view`¹¹⁷ hat die gleiche Funktion, selektiert aber alle Studierenden, die als Status `Alumni` eingetragen haben. Diese Datei kann über den Link “Verzeichnis der Alumni” auf der Startseite des Intranets aufgerufen werden.
- Alle Einträge in diesen Listen enthalten einen Link zu dem vollständigen Eintrag jedes aufgelisteten Studierenden. Diese Vollansicht wird von der DTML-Methode `student_view`¹¹⁸ erzeugt. Diese Methode bekommt von `student_liste_view` oder `alumni_liste_view` die ID des gewünschten Studierenden übergeben und zeigt dann dessen vollständige Daten an. Dabei spielt es keine Rolle, ob der Aufruf ursprünglich von `student_liste_view` oder `alumni_liste_view` stammt.
- Wenn der Besitzer des angezeigten Objekts und der angemeldete Nutzer identisch sind, wird ein Link angezeigt, der zu dem `edit_`

¹¹⁶ Name: `student_liste_view`, Position: `/bmm/intranet/`

¹¹⁷ Name: `alumni_liste_view`, Position: `/bmm/intranet/`

¹¹⁸ Name: `student_view`, Position: `/bmm/intranet/`

`propertiesForm` führt, über das der Nutzer seinen Datensatz bearbeiten kann.

Eine weitere Möglichkeit der Anzeige von Studierenden Daten kann aus der Anzeige der Modulinformationen aufgerufen werden. Die ausführlichen Modulinformationen, die von der Intranet-Startseite zu jedem Modul aufgerufen werden können, enthalten einen Link, der alle Studierenden des ausgewählten Moduls auflistet. Dieser Link ruft die DTML-Datei `student_modul_view`¹¹⁹ auf. Diese DTML-Datei bekommt das Kürzel des aufgerufenen Moduls übertragen, selektiert alle Studierenden mit dem Status `Student`, selektiert daraus alle Studierenden, die das ausgewählte Modul bei ihren Daten eingetragen haben, und listet diese auf.

e. Dozentenprofile (inkl. Lehrbeauftragte)

Zur Einrichtung der Dozentenprofile wurde nahezu das gleiche Verfahren wie bei der Einrichtung der Studierendenprofile angewendet - bis auf geringfügige Unterschiede sind die beiden Produkte `student` und `dozent` identisch. Ein davon abweichendes Vorgehen war allerdings bei folgenden Teilbereichen notwendig:

- Damit bei einer später notwendig werdenden Überarbeitung des öffentlich zugänglichen Internetangebots auch von dort auf die Dozentendaten zugegriffen werden kann, wurde das Verzeichnis zur Speicherung der Dozenten-Objekte¹²⁰ nicht in den zugriffsgeschützten Intranetbereich, sondern in den öffentlich zugänglichen Internetbereich gelegt.
- Als `properties` der ZClass `dozent`¹²¹ wurden Name, Vorname, akad. Grad, E-Mail, Telefon und Vita eingerichtet. Außerdem wurden zwei weitere Felder eingerichtet, die mit der `lines-property` `modul_`

¹¹⁹ Name: `student_modul_view`, Position: `/bmm/intranet/`

¹²⁰ Name: `dozenten`, Position: `/bmm/`

¹²¹ Name: `dozent`, Position: `/Control_Panel/Products/dozent/`

abkürzung verknüpft wurden: In ein Mehrfachauswahlfeld können Module eingetragen werden, für die der Dozent eventuell als Modulverantwortlicher verantwortlich ist, ein weiteres Mehrfachauswahlfeld erlaubt die Markierung aller Module, in denen der Dozent unterrichtet.

Auch bei der Möglichkeit, die Dozent-Objekte nach ihrer Erstellung zu korrigieren, wurde ein anderes Verfahren angewendet.

- Studierende können ihren Datensatz nur dann bearbeiten, wenn sie ihn auch selbst eingegeben haben und damit als “Besitzer” dieses Objekts registriert sind. Eine Erstellung von Datensätzen durch den Webmaster ist somit ausgeschlossen, da dabei die Kennung des Webmasters als Besitzer des erstellten Objekts eingetragen werden würde.
- Damit die Eingabe von Dozentendaten auch durch den Webmaster möglich wird, ohne die Bearbeitungsmöglichkeit durch den betroffenen Dozenten selbst zu beschneiden, wird nicht der Besitzer des Datensatzes überprüft, sondern der Inhalt eines eigenen Feldes (genauer: eine property der ZClass `dozent`) namens `login_name`. Dieses Feld wird beim Anlegen des Datensatzes eines Dozenten durch den Webmaster mit dem Login-Namen des Dozenten gefüllt. Durch den Vergleich des Feldinhalts mit dem Login-Namen des angemeldeten Dozenten wird diesem die Bearbeitung seines eigenen Datensatzes ermöglicht. Eine unerwünschte Bearbeitung dieses Feldes `login_name` durch den Dozenten selbst ist ausgeschlossen, da in das entsprechende `edit_propertiesForm`¹²² eine `<dtml-if>`-Schleife eingebaut wurde, die das Feld nur zu Bearbeitung ausgibt, wenn der angemeldeten Login-Kennung die Rolle “Manager” zugewiesen wurde.

¹²² Name: `edit_propertiesForm`, Position: `/Control_Panel/Products/dozent/dozent/propertysheets/methods/`

f. Interner Infodienst

Um schnell und unkompliziert wichtige Termine, Änderungen und Nachrichten für den Studienbetrieb veröffentlichen zu können, wird auf der Startseite des Intranets ein Bereich zur Anzeige aktueller Meldungen eingerichtet. Nutzer mit Managerrechten können direkt über einen Link auf der Startseite neue Meldungen erstellen.

Ein relativ umfangreiches Produkt, das genau einem solchen Zweck dient, existiert schon innerhalb der Zope-Installation des verwendeten Servers. Dieses Produkt mit der Bezeichnung "News" wurde von Herrn Florian Fahrbach, Content Manager der HdM Stuttgart, entwickelt und realisiert. Es steuert auf der ebenfalls auf diesem Server beheimateten Website der Expertengruppe "Bibliothek und Schule" des Deutschen Bibliotheks-Verbandes die Eingabe und Anzeige aktueller Meldungen. Bei einer ersten Begutachtung schien dieses Produkt auch gut geeignet für den Einsatz auf der neuen BMM-Website zu sein, allerdings zeigte sich dann bei genauerer Betrachtung, dass es auf erheblich mehr Anforderungen eingeht (Anzeige von Ansprechpartnern, verknüpften Dateien und Bildern, Anzeige mehrerer zusätzlicher URLs und Quellen) und sich dadurch sehr komplex und auch etwas unübersichtlich darstellt. Ein weiteres Problem besteht darin, dass bei einem durch beide Websites gemeinsam genutzten Produkt eine individuelle Formatierung, z.B. von der Eingabeseite für neue Meldungen, unterbleiben muss. Da die Anforderungen an ein derartiges Produkt für den Einsatz auf der BMM-Website aber überschaubar waren und ein nur für die BMM-Website geltendes Produkt problemlos formatiert und in den Arbeitsablauf auf der BMM-Site eingepasst werden konnte, wurde vom Verfasser, nach dem Modell des bereits vorhandenen, ein in der Leistungsfähigkeit angepasstes Produkt speziell zum Einsatz auf der neuen BMM-Website hergestellt. Als sehr hilfreich stellte sich die Übernahme der DTML-Methoden und -Dateien heraus, die zur Steuerung und zur Anzeige der Meldungen dienen, da vor allem die zeitgesteuerte Verlagerung der einzelnen Meldungen von dem Anzeigen-

bereich für aktuelle Meldungen in das Meldungsarchiv verhältnismäßig aufwändigen Programmcode erfordert. Die einzelnen Arbeitsschritte:

- Erstellen eines neuen Produktes namens `bmmnews`¹²³.
- Innerhalb des Moduls: Erstellen einer neuen ZClass `bmmnews`¹²⁴ mit folgenden properties: `title` (Überschrift der Meldung), `startseite` (weitere Meldungszeile, die nach der Überschrift nur auf der Startseite erscheint), `inhalt` (ausführlicher Meldungsinhalt zur Anzeige bei der Vollansicht der Meldung und im Archiv), `autor` (Urheber der Meldung), `datum` (Datum der Eingabe der Meldung), `von` (Anfang des Gültigkeitszeitraums der Meldung), `bis` (Ende des Gültigkeitszeitraums der Meldung).
- Die beiden für die Erstellung neuer Meldungen verantwortlichen DTML-Methoden, der constructor `bmmnews_add`¹²⁵ und das constructor-Formular `bmmnews_addForm`¹²⁶, wurden durch Einbindung des Headers und Footers an das Layout des BMM-Intranets angepasst. Diejenigen Felder im constructor-Formular `bmmnews_addForm`, die Datumsangaben enthalten, wurden mit dem aktuellen Tagesdatum vorbesetzt, die `id` von neuen Meldungen (genauer: von den neu erstellten Objekten vom Typ `bmmnews`) wird automatisch aus dem Systemdatum generiert und im Formular nicht zum Editieren angezeigt.

Die Anzeige der Meldungen auf der Startseite des Intranets und im Meldungsarchiv sowie die Vollanzeige von einzelnen Meldungen wird durch die beiden DTML-Methoden `news_footer`, `news_view` und die DTML-Datei `archiv` gesteuert.

¹²³ Name: `bmmnews`, Position: `/Control_Panel/Products/`

¹²⁴ Name: `bmmnews`, Position: `/Control_Panel/Products/bmmnews/`

¹²⁵ Name: `bmmnews_add`, Position: `/Control_Panel/Products/bmmnews/`

¹²⁶ Name: `bmmnews_addForm`, Position: `/Control_Panel/Products/bmmnews/`

- Die Methode `news_footer`¹²⁷ ist in die Startseite des BMM-Intranets eingebunden. In dieser Datei werden alle im Verzeichnis `bmm_news` liegenden Meldungen auf ihren Gültigkeitszeitraum (die Spanne zwischen den beiden Datumsangaben, die in den Eigenschaften `von` und `bis` eingetragen sind) hin überprüft und selektiert. Liegt das aktuelle Tagesdatum innerhalb dieses Zeitraums, werden die Werte der Eigenschaften `title` (die Überschrift der Meldung) und `startseite` (zusätzlicher kurzer Text zur Anzeige auf der Startseite) angezeigt.
Außerdem wird durch einen Zähler überprüft, ob mehr als fünf Meldungen diesen Kriterien entsprechen. Sollte dies der Fall sein, werden nur die fünf neuesten Meldungen angezeigt.
- Der Link zum Erstellen einer neuen Meldung ist in der Datei `news_footer` mit einer `<dtml-if>`-Schleife umgeben. Nur wenn dem angemeldeten User die Rolle “Manager” zugewiesen wurde, ist dieser Link sichtbar. Bei Aufruf dieses Links wird das Eingabeformular `bmmnews_addForm` gestartet.
- Die Vollanzeige einer Meldung wird durch das Aufrufen des Links “Mehr dazu ...” erreicht, der in der Meldungsübersicht auf der Intranet-Startseite nach jeder Meldung erscheint. Dieser Link ruft die `id` einer Meldung zusammen mit der DTML-Methode `news_view`¹²⁸ als Parameter auf. Dieser Aufruf bewirkt, dass von der betroffenen Meldung die Überschrift, der gesamte Text des Inhaltsfeldes, der Autor und das Erstellungsdatum angezeigt werden.
- Der Link zum News-Archiv auf der Startseite ruft die DTML-Datei `archiv`¹²⁹ auf. Dies bewirkt die Anzeige aller vorhandenen Meldungen, wobei die neuesten Meldungen am Anfang erscheinen. Vom Meldungsinhalt werden allerdings nur die ersten 500 Zeichen aus-

¹²⁷ Name: `news_footer`, Position: `/bmm/intranet/`

¹²⁸ Name: `news_view`, Position: `/bmm/intranet/`

¹²⁹ Name: `archiv`, Position: `/bmm/intranet/bmm_news/`

gegeben. Durch den Link “Mehr zu dieser Meldung ...” wird die betroffene Meldung vollständig angezeigt.

VII. Ausblick

Eine Website ist kein statisches Objekt, vielmehr ähnelt sie einem lebendigen Organismus. Über ihre ganze Lebensdauer hinweg wird eine Website ständig Änderungen, Aktualisierungen und Korrekturen erfahren, inhaltliche Änderungen werden strukturelle Anpassungen notwendig machen, neue Angebote werden manche ältere überflüssig werden lassen. Obwohl sich die in dieser Arbeit neu realisierte Website des Studiengangs Bibliotheks- und Medienmanagement durchaus als ein einsatzfertiges Produkt versteht, das ohne Verzugszeit sinnvoll im Echtbetrieb zum Einsatz kommen kann, muss sie auch als ein Ausgangspunkt verstanden werden, der für die Darstellung des Studiengangs eine solide und gleichzeitig flexible Basis bereitstellt. Durch den objektorientierten Aufbau von Zope können die Eigenschaften der Objekte einer Klasse schnell aktualisiert oder auch ergänzt werden, durch das Erstellen neuer Objektklassen lassen sich auch weiter reichende Veränderungen in Struktur und Aufbau des Studiums schnell und flexibel abbilden.

Allerdings sollten schon in näherer Zukunft einige Entscheidungen getroffen werden, die dann auch unmittelbar auf den Echtbetrieb der Website Auswirkungen haben werden.

- Als erstes muss über die Anbindung der Nutzerverwaltung der neuen BMM-Website an mögliche Authentifizierungsquellen des Fachbereiches befunden werden. Hierbei ist vor allem der gewünschte Status der BMM-Alumni hinsichtlich Intranet-Benutzung und das im Fachbereich festzulegende Verfahren mit dieser Benutzergruppe ausschlaggebend. Daneben sollte schon der (voraussichtliche) Status der Studierenden im Falle einer Kooperation mit der Hochschule für Technik und Wirtschaft in Chur berücksichtigt werden.
- Bei den nächsten größeren Aktualisierungen von Inhalten des öffentlich zugänglichen Internetbereichs sollten auch in diesem Bereich die momentan noch vorhandenen statisch eingebundenen Informationen durch die Integration Zope-generierter dynamischer Inhalte ersetzt werden. (Zum Beispiel durch den Einsatz des Modul-Produktes). Daneben sollten auch die

momentan im ganzen Internet-Bereich noch eingesetzten frame-basierten Webseiten durch eine Umstellung des Layouts auf Seiten mit einem Zope-Header und Zope-Footer abgelöst werden.

- Auch eine Überarbeitung des gesamten Designs muss mittelfristig angedacht werden. Die grafisch aufwändiger gestaltete Startseite könnte dabei weiter zum Einsatz kommen, auch die verwendeten Farben könnten sich weiterhin am Corporate Design des Fachbereichs Information und Kommunikation orientieren, jedoch könnten vor allem die Navigationselemente innerhalb der einzelnen Bereiche dynamisch generiert werden, wodurch eine weitere Vereinfachung des Administrationsaufwandes erzielt werden könnte.

Insgesamt betrachtet ist diese neue Website des Master-Studiengangs Bibliotheks- und Medienmanagement durch ihre technische und gestalterische Flexibilität für kommende Anforderungen, wie sie etwa im Zuge die angestrebte Kooperation mit der Hochschule für Technik und Wirtschaft Chur entstehen werden, gut gerüstet.

Literatur- und Quellenverzeichnis

1. **About Zope Corporation.** <http://www.zope.com/Corporate/Company> Profile (Datum des Zugriffs: 12.04.2004).
2. **Bager, Jo:** Hüter der Inhalte. Websites mit Content-Management-Systemen verwalten. <http://www.heise.de/kiosk/archiv/ct/02/20/172/> (Datum des Zugriffs: 20.12.2003).
3. **Bager, Jo; Sachs, Andreas:** Inhalts-Dompteure. Acht günstige Lösungen für Content-Management. <http://www.heise.de/kiosk/archiv/ct/02/20/178/> (Datum des Zugriffs: 20.12.2003).
4. **Basiswissen.** Der IBT[®] SERVER garantiert in Ihrer Organisation reibungslose Lern-, Informations- und Kommunikationsprozesse. <http://www.time4you.de/ibt/www/ibt/home/products/esuite.xhtml> (Datum des Zugriffs: 28.12.2003).
5. **Boeckh, Dorothee; Beier, Nina; Schäfer, Christina:** Alles Master - oder was? Halbzeit beim neuen Studiengang "Bibliotheks- und Medienmanagement" in Stuttgart. In: Bibliotheksdienst 07/2002, S. 845 - 849.
6. **Checkliste für Rechenzentren.** <http://www.dfn.de/content/beratung-weiterbildung/rechtimdfn/checkliste> (Datum des Zugriffs: 28.12.2003).
7. **Content Management mit ZOPE.** Webapplikationen mit dem CMF und ZOPE-Produkten entwickeln. Hrsg. von Stephan Richter. Frankfurt: Software & Support Verlag GmbH, 2003.
8. **e-Learning-Newsletter.** Ausgabe 2, November 2003. <http://www.hdm-stuttgart.de/e-learning/elearning.pdf> (Datum des Zugriffs: 30.12.2003).
9. **Gadfly SQL constructs.** Version: 1.2. <http://gadfly.sourceforge.net/sql.html> (Datum des Zugriffs: 08.01.2004).
10. **Gesetz zum Schutz personenbezogener Daten (Landesdatenschutzgesetz LDSG)** in der Fassung vom 18. September 2000 (GBI. S. 648), geändert durch Gesetz vom 4. Februar 2003 (GBI. S. 108). <http://www.baden-wuerttemberg.datenschutz.de/Home/Recht/LDSG/ldsgp.zip> (Datum des Zugriffs: 29.12.03).

11. **HdM-Infoletter.** Ausgabe 3, Dezember 2003. https://inhouse.hdm-stuttgart.de/intranet/intranet_infoletter/intradok_20031215142650/download (Datum des Zugriffs: 28.12.03).
12. **Hörtzsch, Matthias; Rapko, Andreas:** Content Management und Zope. Bonn: mitp-Verlag, 2003.
13. **Hoffmann, Claus:** Das Intranet. Ein Medium der Mitarbeiterkommunikation. Konstanz: UVK Verlagsgesellschaft, 2001.
14. **How to allow users to change their own password (mini howto).** <http://www.zope.org/Members/msx/ChangeOwnPassword-mini-howto> (Datum des Zugriffs: 25.01.2004).
15. **Hütter, Bernhard:** Aktuelles zum Teilzeit-Masterstudiengang »Bibliotheks- und Medienmanagement«. In: BuB Forum für Bibliothek und Information, Heft 10/11 2003, S. 609 - 612.
16. **Karl, Günther:** Internet Information Server 5. Landsberg: verlag moderne industrie Buch AG & Co. KG, 2002.
17. **Kuhlmann, Gregor; Müllmerstadt, Friedrich:** SQL. Der Schlüssel zu relationalen Datenbanken. Reinbek: Rowohlt, 2001.
18. **Livelihood 9.1.** http://www.contentmanager.de/itguide/produkt_530_livelihood_91.html (Datum des Zugriffs: 06.01.2004).
19. **Münz, Stefan; Nefzger, Wolfgang:** HTML 4.0 Handbuch. Poing: Franzis' Verlag, 1999.
20. **Das Produkt Livelihood 9.1.** <http://www.vrtprj.de/products/livelihood.html> (Datum des Zugriffs: 28.12.2003).
21. **Rohse, Frank; Lennartz, Sven:** Wuchern mit Pfründen. Inhalte mit Content Management Systemen verwalten. In: CT, Heft 21/2000, S. 184 - 195.
22. **Spona, Helma:** Web-Datenbanken. Landsberg: verlag moderne industrie Buch AG & Co. KG, 2002.
23. **SuperGuide for Newbies (Part I of III) - Welcome to the World of Zope.** http://www.zopemag.com/Guides/SuperGuide_Newbies.html (Datum des Zugriffs: 21.01.2004).

24. **Was ist PHP?** Kapitel 1. Einleitung. <http://www.php-homepage.de/manual/introduction.php> (Datum des Zugriffs: 25.01.2004).
25. **Web Content Management.** Websites professionell betreiben. Bonn: Galileo Press GmbH, 2001.
26. **Wrapping email at 72 characters.** <http://www.zopelabs.com/cookbook/1028832719> (Datum des Zugriffs: 13.01.2004).
27. **Zope.** Content-Management- & Web-Application-Server. Hrsg. beehive GmbH, Heidelberg: dpunkt.verlag, 2001.
28. **Zope.** Aspekte des Z Objects Publishing Environments. Hrsg. von Stephan Richter. Frankfurt: Software & Support Verlag GmbH, 2001.

Abbildungsverzeichnis

Abb. 1: Die Management-Konsole von Zope entspricht der sog. "Zweitansicht"	26
Abb. 2: Klassen, Objekte, deren Attribute und Werte in Zope	28
Abb. 3: Position des versteckten "hot spot" mit dem Link zur Newsletter-Administration	42
Abb. 4: Definieren der lines-properties im Root der Zope-Installation	48
Abb. 5: Definition des property Sheets <code>modul_eigenschaften</code>	50
Abb. 6: Definition des property Sheets <code>lehrveranstaltung_eigenschaften</code>	52
Abb. 7: Objekte vom Typ <code>lehrveranstaltung</code> werden als subobjects von Objekten des Typs <code>modul</code> definiert	53

Alle Abbildungen wurden vom Verfasser hergestellt.

Die Abbildung 2 beruht inhaltlich auf einem Beispiel, das im Titel "Zope. Content-Management- & Web-Application-Server" auf den Seiten 21 bis 22 beschrieben wird.

Alle anderen Abbildungen sind Screen-Shots, die die Zweit- oder Drittansicht von Zope bei der Anwendung der beschriebenen Produkte zeigen.

Glossar¹³⁰

Anonymous:

Anonymous ist eine Standardrolle in Zope. Diese Rolle wird zunächst jedem Zope-Anwender zugeteilt, so lange, bis er sich mit seinem Namen und Passwort anmeldet.

ASP:

Active Server Pages sind Webseiten, die nicht im aufrufenden Client, sondern auf dem Server ablaufen und neben HTML-Code auch speziellen ASP-Code enthalten können¹³¹.

Authenticated User:

Der Authenticated User ist der zum aktuellen Zeitpunkt eingeloggte Benutzer. Ist kein Benutzer eingeloggt, so ist Anonymous der Authenticated User.

Authentifizierung:

Mit Authentifizierung wird das Identifizieren eines Benutzers gegenüber Zope bezeichnet. Siehe auch: *Login*.

Available-Objects-Menü:

Das Available-Objects-Menü zeigt an, welche Objekte am aktuellen Ort innerhalb von Zope angelegt werden können. Um ein Objekt anzulegen, wird der Meta-Typ des Objekts aus dem Pull-down-Menü Available Objects ausgewählt.

Besitzer:

Siehe: *Owner*.

Case Sensitivity:

Python ist »case sensitive«, das heißt, es wird zwischen Groß- und Kleinbuchstaben unterschieden.

¹³⁰ Sofern nicht anders angegeben, wurde das Glossar in Auszügen entnommen aus: Zope. Content-Management- & Web-Application-Server, S. 389 - 394.

¹³¹ Nach: Spona, Helma: Web-Datenbanken, S. 32.

CMS:

Content-Management-System

Drittansicht:

Man befindet sich immer dann in der Drittansicht, wenn man sich ein Zope-Objekt über einen Webbrowser ansieht.

Drop-down-Menü »Available Objects«:

Siehe: *Available-Objects-Menü*.

DTML-Tag:

Wie im HTML-Bereich bezeichnet ein Tag eine Sprachanweisung, also eine Art Befehl. In DTML-Tags können natürlich nur DTML-Sprachanweisungen verwendet werden.

Eigenschaften:

Siehe: *properties*.

Erstansicht:

Unter Erstansicht versteht man den gesamten Quellcode, der Zope implementiert.

Externe Methode:

Eine externe Methode ist ein Zope-Objekt, das eine Verbindung zwischen einem im File-System liegenden Python-Modul und Zope herstellt.

folderish object:

Siehe: *ordnerähnliches Objekt*.

Klassen:

Eine Klasse stellt das Schema dar, nach dem Objekte abgeleitet werden. In einer Klasse werden die Methoden und Attribute definiert, über die die davon abgeleiteten Objekte verfügen.

Login:

Über ein Login-Fenster kann sich ein Benutzer mit Namen und Passwort anmelden und somit gegenüber Zope identifizieren.

Management-Ansicht:

Siehe: *Management-Bildschirm*.

Management-Bildschirm:

Der Management-Bildschirm stellt eine Oberfläche zur Verfügung, in der mit den Objekten eines Zope-Servers gearbeitet werden kann. Siehe auch: *Zweitansicht*.

Manager

Manager ist eine Standardrolle in Zope. Diese Rolle hat zunächst alle Rechte außer dem Recht „Take Ownership“.

nicht-ordner-ähnliches Objekt:

Ein nicht-ordner-ähnliches Objekt ist ein Zope-Objekt, das keine weiteren Objekte enthalten kann. Nicht-ordnerähnliche Objekte sind z.B. DTML-Methoden, DTML-Dokumente oder Dateien (File).

non-folderish object:

siehe: *nicht-ordner-ähnliches Objekt*

Objekte:

Objekte werden von Klassen abgeleitet und besitzen Methoden und Attribute. Alle Objekte einer Klasse haben dieselben Methoden.

ordner-ähnliches Objekt:

Ein ordner-ähnliches Objekt ist ein Zope-Objekt, das andere Objekte enthalten kann. Ordner-ähnliche Objekte sind z.B. Ordner (Felder) oder ein ZCatalog.

Owner:

Owner ist eine Standardrolle in Zope. Diese Rolle hat standardmäßig nur das Recht "Take Ownership".

Ownership:

Jedes Objekt in Zope hat einen »Besitzer« (Owner), mit Ausnahme der bei der Installation von Zope angelegten Objekte. Der Benutzer, der ein Zope-Objekt anlegt, wird automatisch der Besitzer dieses Objekts.

Permissions:

Permissions oder auch Rechte legen fest, welche Handlungen ein Benutzer in Zope durchführen darf. Sie können nur an Rollen - und somit nur indirekt an Benutzer - vergeben werden.

PHP:

(Akronym für "PHP: Hypertext Preprocessor") ist eine weit verbreitete und für den allgemeinen Gebrauch bestimmte Open Source Skriptsprache, welche speziell für die Webprogrammierung geeignet ist, und in HTML eingebettet werden kann¹³².

Produkt-Verzeichnis:

Das Produkt-Verzeichnis ist ein Unterverzeichnis von `/lib/pytbon/Products/` und enthält alle für das entsprechende Produkt notwendigen Dateien.

Properties:

Properties sind die Eigenschaften oder auch Attribute eines Objektes.

Python:

Python ist die Programmiersprache, in der in Zope auch individuelle komplexe Anwendungen realisiert werden können.

Rechte:

Siehe: *Permissions*.

Reservierte Wörter:

Reservierte Wörter sind Wörter, die nicht als Bezeichner für Objekte benutzt werden dürfen, da sie bereits in Python oder Zope existieren.

Rolle:

Jedem in Zope existierenden Benutzer kann für ein beliebiges Zope-Objekt eine/mehrere lokale Rolle/n zugeteilt werden. Diese Rolle bestimmt dann, welche Rechte der entsprechende Benutzer bezüglich dieses Objekts hat. Das Setzen von lokalen Rollen kann die Rechte eines Benutzers bzgl. des Objekts einschränken oder erweitern.

¹³² Was ist PHP? Kapitel 1. Einleitung. <http://www.php-homepage.de/manual/introduction.php>

Tableiste:

Die Tableiste befindet sich am oberen Rand des rechten Frames im Zope-Management-Bildschirm. Mit ihr kann man in die einzelnen Ansichten eines Objekts gelangen.

User:

User ist der englische Begriff für Benutzer. Ein Benutzer wird in Zope mit Name, Passwort, Domains und Rolle(n) angelegt.

UserFolder:

Im UserFolder (acl_users) werden in Zope die einzelnen Benutzer angelegt und verwaltet.

WCMS:

Web-Content-Management-System

ZClasses:

Mit Hilfe von ZClasses können ohne eigentlichen Programmieraufwand schnell eigene Zope-Produkte erstellt werden.

Zope-Produkte:

Zope-Produkte stellen in Python programmierte Distributionen dar, die auf einem Zope-Server installiert werden können. Ist ein Produkt installiert, können auf dem Zope-Server Objekte erstellt werden, die dann die im Produkt implementierten Funktionalitäten aufweisen.

Zweitansicht:

Die Zweitansicht umfasst den gesamten Management-Bildschirm. Wenn Sie also im Management-Bildschirm arbeiten, befinden Sie sich in der Zweitansicht.

Anhang

A. Verzeichnis der verwendeten Zope-Produkte und Quellcodes

1. **ChangePassAction**

(<http://www.zope.org/Members/msx/ChangeOwnPassword-mini-howto>)

ermöglicht authentifizierten Benutzern, ihr Passwort selbst zu ändern.

2. **dozent**

(wurde vom Verfasser erstellt)

ermöglicht als eigenes Produkt über die ganze Website hinweg das Erstellen von Objekten vom Typ `dozent` sowie den Zugriff auf diese Objekte. Die von der ZClass `dozent` abgeleiteten Objekte vom Typ `dozent` sind als ordner-ähnliche Objekte angelegt, da sie neben den Informationen zu jeder Lehrkraft auch ein Bild beinhalten können.

3. **hausarbeit**

(wurde vom Verfasser erstellt)

Das Produkt `hausarbeit` ermöglicht das Erzeugen von Objekten vom Typ `hausarbeit` und den Zugriff auf diese Objekte innerhalb des gesamten Intranetbereichs. Die von der ZClass `hausarbeit` abgeleiteten Objekte vom Typ `hausarbeit` sind als ein ordner-ähnliche Objekt angelegt, da sie neben den Informationen zu jeder Hausarbeit auch die Datei der Arbeit selbst beinhalten müssen. Neben dem Erzeugen neuer Objekte führt die Constructor-Methode `hausarbeit_add` als weitere Aktion noch den Versand einer E-Mail durch.

4. **HTMLDocument 1.0**

(<http://www.zfl.uni-bielefeld.de/personal/mjablonski/HTMLDocument.tar.gz>)

ermöglicht das Einstellen konventioneller HTML-Dateien, deren Header und Footer dann durch den systemweiten Standard-Header und -Footer von Zope ersetzt werden, wodurch eine Anwendung des vorgegebenen Seiten-Layouts erzwungen werden kann.

5. **modul**

(wurde vom Verfasser erstellt)

ermöglicht als eigenes Produkt über die gesamte Website hinweg das Erstellen neuer Objekte mit der Klasse `modul` sowie den Zugriff auf diese Objekte. Das Produkt `modul` ist als ein ordner-ähnliches Objekt angelegt, da es Objekte vom Typ `lehrveranstaltung` enthält.

6. **News**

(wurde vom Verfasser erstellt nach dem Muster eines ähnlichen umfangreicheren Produktes, das von Herrn Florian Fahrbach, Content Manager der HdM Stuttgart, zum Einsatz auf der Website der Expertengruppe "Bibliothek und Schule" des Deutsche Bibliotheks-Verbandes erstellt wurde)

ermöglicht die Eingabe und Anzeige von aktuellen Meldungen, die dann nach Ablauf einer Verfallsfrist in ein News-Archiv verschoben werden.

7. **ParseText**

(<http://www.zopelabs.com/cookbook/1028832719>)

setzt bei längeren Texteingaben (z.B. durch Texteingabefelder) die Zeilenlänge automatisch auf maximal 72 Zeichen, ohne einzelne Worte zu zerstückeln. Streng genommen ist `ParseText` ist kein Produkt, sondern ein Python-Skript.

8. **SiteAccess2 incl. VirtualHostMonster**

<http://zope.org/Members/4am/SiteAccess2/2.0.0b2/SiteAccess-2.0.0b2-nonbin.tgz>

erlaubt virtuelles Hosting auf einem Zope-Server (Zuordnen eines domain-Namens zu einem Unterverzeichnis innerhalb der Zope-Hierarchie) ohne die Anwendung weiterer Web-Server-Software.

9. **skript**

(wurde vom Verfasser erzeugt)

Das Produkt `skript` ermöglicht das Erzeugen von Objekten vom Typ `skript` und den Zugriff auf diese Objekte innerhalb des gesamten Intranetbereichs. Die von der ZClass `skript` abgeleiteten Objekte vom Typ `skript` sind als ordner-ähnliche Objekte angelegt, da sie neben den

Informationen zu jedem Skript auch die Datei des Skriptes selbst beinhalten müssen.

10. **SlaveUserFolder 0.1.0**

(<http://zope.org/Members/srichter/Products/SlaveUserFolder/0.1.0/SlaveUserFolder-0.1.0.tgz>)

ermöglicht es, bereits in Zope vorhandenen Benutzern für ein bestimmtes Verzeichnis andere Rechte zuzuteilen, als sie in der Rolle festgelegt wurden, die dem Nutzer zugeordnet ist.

11. **student**

(wurde vom Verfasser erzeugt)

ermöglicht als eigenes Produkt innerhalb des Intranets das Erstellen von Objekten vom Typ `student` sowie den Zugriff auf diese Objekte. Die von der ZClass `student` abgeleiteten Objekte vom Typ `student` sind als ordner-ähnliche Objekte angelegt, da sie neben den Informationen zu jedem Studierenden auch ein Bild beinhalten können. Auch die Alumni können durch das Produkt `student` erfasst werden, da diese durch den Wert der Eigenschaft `status` gekennzeichnet sind und selektiert werden können.

B. Inhalt der beiliegenden CD-ROM

- **Masterarbeit „Die Website des Master-Studiengangs Bibliotheks- und Medienmanagement - Technische und inhaltliche Neukonzeption und Realisierung“**
Text als PDF-Datei
- **Online- und Internet-Quellen**
als Link und als Volltext
- **Zope Version 2.5.1.**
Die für diese Arbeit verwendete Zope-Version zur Installation auf PC unter Windows.
- **Verwendete externe Zope-Produkte**
aus öffentlich zugänglichen Zope-Produkte-Archiven
 - HTMLDocument 1.0
 - SlaveUserFolder 0.1.0
- **Datei data.fs**
Zope-Datenbank-Datei mit allen Inhalten und Programmierungen der Website
- **Anleitung zur Installation von Zope, den verwendeten Produkten und des Datenfiles auf einem lokalen PC**

C. Erklärung

Hiermit erkläre ich, dass ich die vorliegende Masterarbeit selbstständig angefertigt habe. Es wurden nur die in der Arbeit ausdrücklich benannten Quellen und Hilfsmittel benutzt. Wörtlich oder sinngemäß übernommenes Gedankengut habe ich als solches kenntlich gemacht.

Stuttgart, 17.02.2004