

Natalia Fahrenbruch

**Aufbau eines Anwendungssystems zur
Erstellung Dynamischer Websites**

Diplomarbeit

BEI GRIN MACHT SICH IHR WISSEN BEZAHLT



- Wir veröffentlichen Ihre Hausarbeit, Bachelor- und Masterarbeit
- Ihr eigenes eBook und Buch - weltweit in allen wichtigen Shops
- Verdienen Sie an jedem Verkauf

Jetzt bei www.GRIN.com hochladen
und kostenlos publizieren



Bibliografische Information der Deutschen Nationalbibliothek:

Bibliografische Information der Deutschen Nationalbibliothek: Die Deutsche Bibliothek verzeichnet diese Publikation in der Deutschen Nationalbibliografie; detaillierte bibliografische Daten sind im Internet über <http://dnb.d-nb.de/> abrufbar.

Dieses Werk sowie alle darin enthaltenen einzelnen Beiträge und Abbildungen sind urheberrechtlich geschützt. Jede Verwertung, die nicht ausdrücklich vom Urheberrechtsschutz zugelassen ist, bedarf der vorherigen Zustimmung des Verlanges. Das gilt insbesondere für Vervielfältigungen, Bearbeitungen, Übersetzungen, Mikroverfilmungen, Auswertungen durch Datenbanken und für die Einspeicherung und Verarbeitung in elektronische Systeme. Alle Rechte, auch die des auszugsweisen Nachdrucks, der fotomechanischen Wiedergabe (einschließlich Mikrokopie) sowie der Auswertung durch Datenbanken oder ähnliche Einrichtungen, vorbehalten.

Copyright © 2004 Examicus Verlag
ISBN: 9783656999287

Natalia Fahrenbruch

Aufbau eines Anwendungssystems zur Erstellung Dynamischer Websites

Examicus - Verlag für akademische Texte

Der Examicus Verlag mit Sitz in München hat sich auf die Veröffentlichung akademischer Texte spezialisiert.

Die Verlagswebseite www.examicus.de ist für Studenten, Hochschullehrer und andere Akademiker die ideale Plattform, ihre Fachtexte, Studienarbeiten, Abschlussarbeiten oder Dissertationen einem breiten Publikum zu präsentieren.

FACHHOCHSCHULE KÖLN
University of Applied Sciences Cologne

FAKULTÄT FÜR WIRTSCHAFTSWISSENSCHAFTEN
Faculty of Economics and Business Administration

Diplomarbeit
(Drei - Monats - Arbeit)

zur Erlangung
des Diplomgrades
Diplom-Kauffrau (FH)
im Studiengang Betriebswirtschaftslehre
der Fachhochschule Köln

***„Aufbau eines Anwendungssystems zur
Erstellung Dynamischer Websites“***

vorgelegt am: 19. Februar 2004
von cand. rer. pol.: Natalia Fahrenbruch

Danksagung

Diese Gelegenheit möchte ich wahrnehmen, um mich bei allen Personen zu bedanken, die mich bei der Erstellung dieser Diplomarbeit unterstützt haben.

An erster Stelle danke ich meinen lieben Eltern, die mir während des gesamten Studiums geholfen haben. Für ihr enormes Engagement möchte ich ihnen hiermit ganz herzlich danken.

Ein großer Dank gilt meinen wissenschaftlichen Betreuern Herrn Prof. Dr. Otto K. Amon und Herrn Prof. Dr.-Ing. Werner Müller, die mir mit konzeptionellen Anregungen und fachlichen Ratschlägen jederzeit zur Seite standen.

Besonders möchte ich mich auch bei meinem Freund Nico Laudor bedanken, der mich bei der Verwirklichung tatkräftig und moralisch unterstützt hat. Für seinen unermüdlichen Einsatz gebührt ihm ein großes Dankeschön.

Inhaltsverzeichnis

Danksagung	II
Inhaltsverzeichnis	III
Abkürzungsverzeichnis	VIII
Abbildungsverzeichnis	XIII
Tabellen- und Symbolenverzeichnis	XVIII
1 Grundlagen	1
1.1 Problemstellung und Zielsetzung	1
1.2 Aufbau der Arbeit.....	3
1.3 Begrifflichkeiten.....	3
1.3.1 World Wide Web (WWW)	4
1.3.2 Website.....	6
1.3.3 Dynamische vs. Statische Webseiten.....	7
1.3.3.1 Statische Webseiten.....	7
1.3.3.2 Dynamische Webseiten.....	8
1.3.3.3 Anwendungsbeispiele.....	12
1.3.4 Software	14
1.3.4.1 Typologisierung von Software	17
1.3.4.2 Aufwendungen	23
1.3.4.3 Sicherheit.....	24
1.3.5 Open-Source-Software.....	26
1.3.5.1 Open Source - Lizenzmodelle.....	27
1.3.5.2 Stärken und Schwächen von Open-Source-Software.....	30
1.3.5.3 Einsatzbereiche/Ausgewählte Open-Source-Projekte.....	33
1.3.6 Erstellung	36
2 Konfigurierung des Anwendungssystems	38
2.1 Betriebssystem (Windows und Linux).....	39
2.1.1 Öffentlicher Server (Praxiseinsatz).....	39
2.1.2 Testserver (Erstellungsumgebung).....	40

2.2	Webserver (Apache).....	42
2.2.1	Entwicklung des Apache.....	42
2.2.2	Lizenzbestimmungen	44
2.2.3	Funktionsweise des Apache	44
2.2.4	Leistungsmerkmale des Apache.....	49
2.2.5	Stärken und Schwächen des Apache.....	51
2.3	Webservererweiterung (PHP)	54
2.3.1	Webseitenerzeugung mittels CGI-Skript und Servererweiterung....	54
2.3.2	Entwicklung von PHP	56
2.3.3	Lizenzbestimmungen	58
2.3.4	Funktionsweise von PHP	58
2.3.5	Leistungsmerkmale des PHP	61
2.3.6	Stärken von PHP	61
2.4	Datenbankserver (MySQL).....	64
2.4.1	Aufbau eines Datenbanksystems.....	64
2.4.2	Relationales Datenmodell	66
2.4.3	Entwicklung von MySQL	68
2.4.4	Lizenzbestimmungen	69
2.4.5	Funktionsweise von MySQL	70
2.4.6	Stärken und Schwächen von MySQL	72
2.4.7	Komponenten der MySQL-Distribution	75
2.4.8	Benutzeroberflächen für MySQL.....	76
2.4.8.1	mysql (MySQL-Monitor).....	76
2.4.8.2	WinMySQLadmin.....	77
2.4.8.3	MySQL Control Center (MySQLCC).....	77
2.4.8.4	phpMyAdmin	78
2.5	Hilfsmittel zur Webseitenerstellung.....	81
2.5.1	HTML-Editoren	81
2.5.1.1	WYSIWYG-Editoren	82
2.5.1.2	HTML-Texteditoren.....	87
2.5.2	Web-Browser	90
2.5.3	Grafikprogramme.....	92
2.5.4	Sonstige Tools.....	97

3	Architektur des Anwendungssystems.....	98
3.1	Integration von OSS in das Betriebssystem Win2000 Pro.....	100
3.2	Kompatibilität der OSS-Elemente.....	100
4	Zusammenstellung des Anwendungssystems.....	102
4.1	Systemvoraussetzungen	102
4.2	Implementierung des Apache Webservers.....	103
4.2.1	Vorbereitung	103
4.2.1.1	Voraussetzung für den Betrieb des Webservers.....	103
4.2.1.2	Binäry vs. Sourcecode.....	104
4.2.1.3	Beschaffung der Software	104
4.2.2	Installation des Apache	106
4.2.3	Funktionstest	112
4.2.4	Starten, Stoppen und Neustarten des Webservers.....	115
4.2.5	Konfiguration des Servers.....	118
4.2.5.1	Authentifizierung	134
4.2.5.2	Apache-Konfiguration testen	136
4.3	Implementierung von PHP Modul	137
4.3.1	Beschaffung der Software.....	137
4.3.2	Installation von PHP	137
4.3.3	Funktionstest	140
4.3.4	Konfiguration von PHP.....	142
4.4	Implementierung von MySQL-Datenbankserver.....	149
4.4.1	Beschaffung der Software.....	149
4.4.2	Installation von MySQL.....	150
4.4.3	Administration mit WinMySQLAdmin	153
4.4.4	Starten und Stoppen des MySQL-Servers/ Einrichtung als Dienst	156
4.4.5	Funktionstest	159
4.4.6	Konfiguration des MySQL-Servers	160
4.4.7	Administration mit phpMyAdmin	162
4.4.7.1	Beschaffung der Software und Installation.....	162
4.4.7.2	Funktionstest	165
4.4.7.3	Konfiguration von phpMyAdmin.....	166
4.4.7.4	Benutzerverwaltung	167

5	Fallstudie	171
5.1	Struktur der Website.....	172
5.2	Seitenlayout.....	173
5.2.1	Trennung von Layout und Inhalt.....	173
5.2.2	Die Hauptnavigation	181
5.3	Seiteninhalte	185
5.3.1	Formulare: Interaktivität der Website	185
5.3.1.1	Formularelemente.....	186
5.3.1.2	HTML-Formulare erstellen/Formularelemente einfügen.....	188
5.3.1.3	Parameterübergabe mit POST und GET	190
5.3.1.4	Formularauswertung mit PHP	192
5.3.2	Sessionmanagement	199
5.3.2.1	Sessions mit URI.....	199
5.3.2.2	Sessions mit versteckten Feldern	199
5.3.2.3	Nutzung von Cookies	200
5.3.2.4	Textdateien und Datenbanken.....	200
5.3.3	Mailen mit PHP: Kontaktformulare	201
5.3.4	Arbeit mit Dateien: Besucherzähler	209
5.3.5	Cookies: Besucher wiedererkennen	213
5.3.5.1	Cookies setzen mit PHP	214
5.3.5.2	Persönliche Begrüßung	215
5.3.5.3	Web-Umfrage.....	217
5.3.6	Datenbankgestütztes News-System	223
5.3.6.1	Datenbank und Tabelle anlegen	223
5.3.6.2	Speicherung von Binärdateien	225
5.3.6.3	Tabellendaten einfügen, anzeigen und ändern	226
5.3.6.4	Die „öffentliche“ Newsseite.....	228
5.3.6.5	Administration des News-Systems.....	242
5.3.7	Weitere praktische Anwendungen	254
5.4	Sicherheitskopien erstellen.....	261
6	Nutzwertanalyse	263

7	Literaturverzeichnis.....	268
7.1	Fachbücher und Printmedien.....	268
7.2	Elektronische Medien.....	271
	Glossar	275

Inhaltsverzeichnis zum Anhang (auf der CD)

Anhang **A**: URI (Uniform Resource Identifier)

Anhang **B**: HTTP (Hypertext Transfer Protocol)

Anhang **C**: HTML (Hypertext Markup Language)

Anhang **D**: Statuscodes

Anhang **E**: Die „Open Source Definition“

Anhang **F**: Beispielkonfiguration httpd.conf

Anhang **G**: Beispielkonfiguration php.ini

Anhang **H**: PHP-Editoren

Anhang **I**: PHP-Profianwendungen

Anhang **J**: PHP-Funktionen

Anhang **K**: Grafik-Tools

Anhang **L**: E-Mail-Korrespondenz

Abkürzungsverzeichnis

Abb.	Abbildung
abgk.	abgekürzt
Abk.	Abkürzung
allg.	allgemein(e)
ANSI	American National Standards Institut
API	Application Programming Interface
APR	Apache Portable Runtime
ASCII	American Standard Code for Information Interchange
ASF	Apache Software Foundation
ASP	Active Server Pages
BS	Betriebssystem
BSD	Berkeley Software Distribution
BSI	Bundesamt für Sicherheit in der Informationstechnik
bspw.	beispielsweise
bzw.	beziehungsweise
CD	Compact Disc
CERN	Conseil Européen pour la Recherche Nucléaire
CFML	ColdFusion Markup Language
CGI	Common Gateway Interface
CPU	Central Processing Unit
CSS	Cascading Style Sheets
CVS	Comma Separated Value
Daemon	Disk and Execution Monitor
DB	Database, Datenbank
DBMS	Datenbankverwaltungssystem
d.h.	das heißt
DHCP	Dynamic Host Configuration Protocol
DHTML	Dynamic HTML
DLL	Dynamic Link Library
DNS	Domain Name System

DOS	Disc Operating System
DSO	Dynamic Shared Objects
dt.	Deutsch(e)
DTD	Document Type Definition
DWMX	Dreamweaver MX
EDV	Elektronische Datenverarbeitung
E-Mail	Electronic Mail
ERP	Enterprise Resource Planning
FAQ	Frequently Asked Questions
FH	Fachhochschule Köln
FSF	Free Software Foundation
FTP	File Transfer Protocol
f.	für
engl.	Englisch(e)
evtl.	eventuell(e)
EULA	End User License Agreement
f	folgende [Seite]
ff	fort folgende [Seiten]
FSF	Free Software Foundation
FTP	File Transfer Protocol
GB	Giga Byte
GIF	Graphics Interchange Format
GIMP	GNU Image Manipulation Programm
GmbH	Gesellschaft mit beschränkter Haftung
GNU	GNU is not UNIX
GPL	General Public License
ggf.	gegebenfalls
GUI	Graphical User Interface
htdocs	hypertext documents
HTML	Hypertext Markup Language
HTTP	Hypertext Transfer Protocol
HTTPd	Hypertext Transfer Protocol daemon
HTTPS	Hypertext Transfer Protocol over SSL
IBM	International Business Machines

i.d.R.	in der Regel
i.e.S.	im engeren Sinne
IETF	Internet Engineering Task Force
i.H.v.	in Höhe von
IIS	Internet Information Server
Internet	International Network
inkl.	inklusive
IP	Internet Protocol
IRC	Internet Relay Chat
ISP	Internet Service Provider
IT	Information Technology
i.V.m.	in Verbindung mit
i.w.S.	im weiteren Sinne
JPEG	Joint Photographic Experts Group
JSP	Java Server Pages
Kap.	Kapitel
KB	Kilo Byte
LAMP	Linux-Apache-MySQL-PHP (-Konfiguration)
LGPL	Lesser General Public License
LZW-Verfahren	Lempel-Ziv-Welch-Verfahren
Mac OS	Macintosh Operating System
max.	maximal
MB	Mega Byte
MDI	Multiple Document Interface
MHz	Mega Herz
MIME	Multipurpose Internet Mail Extensions
m.	mit
MPM	Multi Processing Modules
MS	Microsoft
MSI	Microsoft-Installer
mSQL	Mini SQL
MXP	Macromedia Extension Package
MySQLCC	MySQL Control Center
NCSA	National Center for Supercomputing Applications

NIC	Network Information Centers
NNTP	Newsgroup Service Transfer Protocol
NPL	Netscape Public License
NT	New Technology
NWA	Nutzwertanalyse
ODBC	Open Database Connectivity
OS	Operating System
OSS	Open Source Software
o.V.	ohne Verfasser
PC	Personal Computer
RDBMS	Relationales DBMS
RDF	Resource Description Framework
Perl	Practical Extraction and Report Language
PHP	Personal Homepage bzw. PHP Hypertext Preprocessor
PHP/FI	Personal Homepage/Form Interpreter
PNG	Portable Network Graphics
POSIX	Portable Operating System Interface for UniX
PWS	Personal Web Server
RFC	Requests for Comments
s.	siehe!
s.a.	siehe auch
SAPI	Server Application Programming Interface
S.	Seite
SGML	Standard Generalized Markup Language
sog.	sogenannte
SMTP	Simple Mail Transfer Protocol
SQL	Structured Query Language
SSI	Server Side Includes
SSL	Security Socket Layer
Stand n.f.	Stand nicht feststellbar
Tab.	Tabelle
TCO	Total Cost of Operations
TCP	Transmission Control Protocol
TCP/IP	Transmission Control Protocol/Internet Protocol

u.	und
u.a.	unter anderem(n)
u.ä.	und ähnliche(s)
Übsg.	Übersetzung
UNIX	Uniplexed Information and Computing System
u./o.	und/oder
URC	Uniform Resource Characteristics
URI	Uniform Resource Identifier
URL	Uniform Resource Locator
URN	Uniform Resource Name
US	United States
usw.	und so weiter
u.U.	unter Umständen
VBScript	Visual Basic Script
Vgl.	Vergleiche!
Vgl.a.	Vergleiche auch
vs.	versus
W3C	World Wide Web Consortium
WAMP	Windows-Apache-MySQL-PHP (-Konfiguration)
WS	Workstation
WWW	World Wide Web
WYSIWYG	What You See Is What You Get
XML	eXtensible Markup Language
z.B.	zum Beispiel
z.T.	zum Teil
ZEND	Zeev Suraski Andi Gutmann

Abbildungsverzeichnis

Abb. 1: Gliederungsschema für Begrifflichkeiten.....	3
Abb. 2: Kommunikationsschema für statische Webseiten	7
Abb. 3: Kommunikationsschema für dynamische Webseiten	9
Abb. 4: Beispiel für eine statische Webseite	12
Abb. 5: Beispiel für eine Dynamische Website	13
Abb. 6: Klassifizierung von Software	15
Abb. 7: Beispiele für Anwendungssoftware	16
Abb. 8: Die aktuellen Software-Kategorien	17
Abb. 9: Die wesentlichen Kriterien der aktuellen Software-Kategorien.....	22
Abb. 10: Lebenszyklus der Software	23
Abb. 11: Vor- und Nachteile von Open-Source-Software	32
Abb. 12: Erstellungsphasen	36
Abb. 13: Marktdurchdringung der Webserver (August 1995- Mai 2003)	43
Abb. 14: Das "Powered by Apache"-Logo	44
Abb. 15: Interaktion von APR und Betriebssystem aus Sicht eines Entwicklers .	46
Abb. 16: Funktionsweise des durch mpm_winnt implementierten Laufzeitverhaltens	48
Abb. 17: Webseitenerzeugung mittels CGI-Skript	55
Abb. 18: Verbreitung von PHP	57
Abb. 19: Das offizielle "PHP"-Logo	58
Abb. 20: Entstehung einer dynamischen Webseite mit PHP.....	60
Abb. 21: Aufbau eines Datenbanksystems	65
Abb. 22: Darstellung der Tabelle "autor"	66
Abb. 23: Beispiel eines Relationenmodells	67
Abb. 24: Das offizielle "MySQL"-Logo	68
Abb. 25: Datenbank-Client-Server-Architektur	71
Abb. 26: Multithreading	71
Abb. 27: Komponenten des Softwarepaketes MySQL	75
Abb. 28: Das offizielle "phpMyAdmin"-Logo	78
Abb. 29: phpMyAdmin-Startseite	79

Abb. 30: phpMyAdmin-Datenbankansicht.....	80
Abb. 31: phpMyAdmin-Tabellenansicht.....	80
Abb. 32: Syntax-Highlightings für verschiedene Sprachen	89
Abb. 33: Das "Weaverslave"-Logo	89
Abb. 34: "The Gimp"-Logo.....	93
Abb. 35: Architektur des Anwendungssystems.....	99
Abb. 36: Marktdurchdringung des PHP Moduls	101
Abb. 37: Erfolgreicher Kontakt zu dem lokalen Rechner mit Ping-Befehl.....	103
Abb. 38: Die Fehlermeldung des Windows-Installers.....	105
Abb. 39: Start der Apache 2-Installation	106
Abb. 40: Grundlegende Servereinstellungen.....	107
Abb. 41: Feature-Auswahl bei der benutzerdefinierten Installation.....	108
Abb. 42: Änderung des Apache-Installationsverzeichnis.....	110
Abb. 43: Apache-Icon im System-Tray.....	110
Abb. 44: Verzeichnisbaum der Apache-Installation	111
Abb. 45: Startseite nach erfolgreicher Apache-Installation.....	113
Abb. 46: Die Einstellungen für den Einsatz des Internet Explorer 5.5	113
Abb. 47: access.log	114
Abb. 48: Überprüfung der Log-Dateien	114
Abb. 49: Allgemeine Optionen der Dienst-Steuerung.....	115
Abb. 50: So wird der Apache Monotor gestartet.....	116
Abb. 51: Apache-Icon mit möglichen Aktionen zur Schnellsteuerung.....	116
Abb. 52: Zentrale Steuerung des Apache-Monitors	117
Abb. 53: Der Apache Service Monitor	117
Abb. 54: Steuerung des Apache Servers über das Konsolenfenster.....	118
Abb. 55: Laufzeitverhalten des Servers nach Einschalten der KeepAlive- Funktionalität	124
Abb. 56: Dynamisches Verzeichnislisting bei fehlender Indexdatei.....	130
Abb. 57: Fehlermeldung bei fehlender Indexes-Angabe.....	131
Abb. 58: Eingabemaske zum Einloggen im Administrationsbereich.....	135
Abb. 59: Eintrag im Startmenü zum Testen der httpd.conf.....	136
Abb. 60: Überprüfung der Syntax	136
Abb. 61: Basisverzeichnis der PHP-Installation.....	138

Abb. 62: Ausgabe des "phptest"-Skripts im Browserfenster nach erfolgreicher Installation	140
Abb. 63: phpinfo()-Statusbericht	141
Abb. 64: Start der MySQL-Installation	150
Abb. 65: Hinweis bei Änderung des Installationsverzeichnis	151
Abb. 66: Auswahl der Komponenten bei der benutzerdefinierten Installation ...	151
Abb. 67: Verzeichnisstruktur der MySQL-Installation	152
Abb. 68: Eingabemaske beim ersten Start von WinMySQLAdmin	153
Abb. 69: WinMySQLAdmin als Ampel-Icon im System-Tray	153
Abb. 70: Benutzeroberfläche von WinMySQLAdmin	154
Abb. 71: WIN NT-Menüeinträge im WinMySQLAdmin-Icon	155
Abb. 72: So wird WinMySQLAdmin gestartet	155
Abb. 73: Sicherheitsabfrage zum Herunterfahren des MySQL-Servers	156
Abb. 74: MySQL-Service starten/MySQL-Server als Dienst einrichten	156
Abb. 75: MySQL als Windows-Dienst (de)installieren, starten und stoppen	157
Abb. 76: Steuerung des MySQL-Dienstes über Windows-Dienst-Manager	158
Abb. 77: MySQL-Server als Standalone-Programm starten	158
Abb. 78: Test der MySQL-Installation	159
Abb. 79: my.ini-Datei in WinMySQLAdmin editieren	160
Abb. 80: Ein Ausschnitt aus der Datei config.inc.php	163
Abb. 81: Die Startseite von phpMyAdmin nach erfolgreicher Installation	165
Abb. 82: Anonymen Benutzer entfernen	168
Abb. 83: Neuen Benutzer in MySQL anlegen	169
Abb. 84: Der neue Benutzer mit eingeschränkten Rechten in der Tabelle "user"	170
Abb. 85: Die Ordnerstruktur der Website kinder Überraschung Eis	172
Abb. 86: Das Rohlayout in der Entwurfsansicht von Macromedia DWMX	174
Abb. 87: Eine einfache Tabelle mit zwei Zeilen und vier Zellen (Spalten)	175
Abb. 88: Ein Ausschnitt des zerlegten Layouts in der Code- u. Entwurfsansicht	175
Abb. 89: Das Layout in zerlegter Form	177
Abb. 90: Zwischenstand: Layoutvorlage mit korrigierten Links in header.inc ...	180
Abb. 91: Die Hauptnavigation in der Datei footer.inc im Quelltext	181
Abb. 92: Ein Rollover-Bild mittels DWMX erzeugen	182
Abb. 93: Die Menüleiste Ü-Eis ist aktiviert	184
Abb. 94: Die Registerkarte "Formulare"	186

Abb. 95: Der Eigenschaften-Inspektor in der Ansicht "Formulare"	188
Abb. 96: Das Demonstrationsformular m. den wichtigsten Formularelementen	189
Abb. 97: Parameterübergabe mit GET	190
Abb. 98: Fehlermeldung bei Reload einer durch POST übergebenen Seite.....	191
Abb. 99: Auswertung der Eingaben des Demonstrationsformulars	192
Abb. 100: Das Kontakt-Formular	202
Abb. 101: Rückmeldung bei freigelassenen Eingabefeldern.....	204
Abb. 102: Ein unvollständig ausgefüllter Formular	204
Abb. 103: Sendebestätigung mit einem Dankeswort.....	208
Abb. 104: Die automatisch generierte E-Mail aus dem Kontakt-Formular in Outlook Express	208
Abb. 105: Der grafische Counter.....	213
Abb. 106: Die "Willkommen"-Seite mit dem Formular zur Abfrage des Benutzernamens.....	215
Abb. 107: Persönliche Begrüßung mit Cookie realisiert	216
Abb. 108: Das Umfrage-Formular: Verhindern von Mehrfachabstimmung	217
Abb. 109: Umfrageergebnisse: Grafische Auswertung.....	220
Abb. 110: Ein Popup-Fenster mit den Umfrageergebnissen	222
Abb. 111: Neue Datenbank anlegen	223
Abb. 112: Neue Tabelle anlegen	223
Abb. 113: Die Definition der Tabellenfelder inkl. der Zuordnungswerte	224
Abb. 114: Einen Datensatz mit phpMyAdmin einfügen	226
Abb. 115: Aktuelle Zeitangaben mit der SQL-Funktion now() ermitteln	226
Abb. 116: Die news-Tabelle mit Beispieldatensätzen in der Datenansicht.....	227
Abb. 117: Die News-Seite mit einem datenbankgestützten News-System	230
Abb. 118: Eine Meldung m. "langem" Text u. dem Verweis auf die Detailseite	233
Abb. 119: Newsseite m. der Möglichkeit, ZURÜCK und WEITER zu blättern.	236
Abb. 120: Ein HTML-Dokument (m. mehreren Grafiken) dynamisch anzeigen	238
Abb. 121: Grafikdateien dynamisch anzeigen.....	239
Abb. 122: Sprungmenü zum Navigieren zwischen übrigen Detailseiten.....	240
Abb. 123: Die Begrüßungsseite im Administrationsbereich	242
Abb. 124: Schematische Darstellung der Abläufe im Administrationsbereich des News-Systems	243
Abb. 125: Das Skript news_list.php in Aktion	244

Abb. 126: Das Formular zum Aufnehmen neuer Meldungen	245
Abb. 127: Das Formular zum Bearbeiten vorhandener Meldungen.....	250
Abb. 128: Die Startseite der Rubrik "POESIEALBUM"	254
Abb. 129: Formular zum Eintrag ins Gästebuch	255
Abb. 130: Ein Popup-Fenster mit der Anleitung zum Dateiupload	255
Abb. 131: Gästebuch-Einträge, mit der Möglichkeit, ZURÜCK und WEITER zu blättern	256
Abb. 132: Rückmeldung auf die fehlenden Formular-Eingaben.....	257
Abb. 133: Die automatisch generierte E-Mail in Outlook Express	257
Abb. 134: Das Gewinnspiel-Formular.....	258
Abb. 135: Die automatisch generierte E-Mail aus dem Gewinnspiel-Formular in Outlook Express	259
Abb. 136: Farbwahl individuell: Der Besucher wählt seine Farbe selbst!	260
Abb. 137: Administration des Gästebuchs	260
Abb. 138: Eine Sicherheitskopie der Datenbank "eis" erstellen.....	262

Tabellen- und Symbolenverzeichnis

Tab. 1: Übersicht der verschiedenen Lizenzmodelle	27
Tab. 2: Lizenzgebühren für MySQL-Server.....	70
Tab. 3: WYSIWYG-Editoren im Überblick.....	84
Tab. 4: Übersicht der gängigsten Webbrowser.....	92
Tab. 5: Aktuelle Preise der proprietären Bildbearbeitungsprogramme	93
Tab. 6: Marktdurchdringung der Apache-Module	100
Tab. 7: Beschreibung der Apache-Distribution.....	112
Tab. 8: "Globale Umgebung"	126
Tab. 9: "Hauptserver-Konfiguration"	133
Tab. 10: Allgemeine Parameter der php.ini.....	146
Tab. 11: Modulspezifische Parameter der php.ini.....	148
Tab. 12: Die wichtigsten ausführbaren Programme in MySQL.....	152
Tab. 13: Die wichtigsten Einstellungen der my.ini	161
Tab. 14: Die wichtigsten Einstellungen der Datei config.inc.php.....	166
Tab. 15: Zugriffsrechte des Benutzers von MySQL	170
Tab. 16: Beschreibung der Formularelemente	187
Tab. 17: Nutzwertanalyse	267

© Copyright(-Vermerk)

→ „siehe“, Verweis auf einen anderen Eintrag oder Glossar

1 Grundlagen

1.1 Problemstellung und Zielsetzung

Die Erstellung, Einrichtung, und Betreuung von Websites war in der Vergangenheit nur mit Spezialwissen zu bewältigen und dementsprechend teuer auf dem Markt zu haben. Zum einen wurden hohe Beträge für den Kauf spezieller Software aufgewendet und zum anderen mussten Spezialisten engagiert werden, die die Installation, Administration und Pflege übernahmen. Diesen Aufwand konnten und wollten sich viele Betreiber von Websites nicht leisten. Zudem waren die Unternehmen bei der Gestaltung und Aktualisierung der Webinhalte von Fachkräften in der Weise abhängig, dass sie auf die Verfügbarkeit dieser angewiesen waren.

In der jüngeren Vergangenheit sind mit dem Aufkommen der Open Source-Gemeinde die notwendigen Softwareprodukte (von denen einige im Verlauf dieser Diplomarbeit vorgestellt werden) zu günstigen Preisen, wenn nicht sogar umsonst (inklusive ausführlicher Installations- und Softwareanleitungen) im Internet zu beziehen. Außerdem finden sich dort die Grundzüge der Open Source-Bestimmungen (Lizenzen), auf dessen Grundlage diese Programme entwickelt, vertrieben und genutzt werden können.

Des Weiteren gibt es zahlreiche Literatur zu dem Themengebiet „Open Source Software“ (deren hochrangigsten Vertreter sind: Linux, Apache, PHP, MySQL) wie das Literaturverzeichnis am Ende dieser Diplomarbeit dokumentiert.

Nicht zuletzt hat das Internet (mit seinen Diensten: www, ftp, mailto, usenet news) dafür gesorgt, dass aus Ideen und anfänglichen Programmcodes ausgewachsene und meist „brauchbare“ Software entstanden ist. Eine Folge war, dass sich kommerzielle Anbieter mit einer neuen Preispolitik im Markt positionieren mussten. Zusätzlich ermöglicht sie jedem, der Interesse an der Erstellung von Websites hat, dies kostengünstig (abgesehen von dem Downloadaufwand der Software, die meist mehrere MB groß ist) mit „fast“ professionellen Hilfsmitteln zu erstellen.

Mit der Entwicklung neuer Technologien zur einfachen Gestaltung und Anbindung an Datenbanken von Websites ist es mehr und mehr möglich die Wartung des Webauftritts in die eigenen Hände zu nehmen. Die Bestrebung von Website-Betreibern, dynamische Inhalte anzubieten, scheitert oft an dem damit verbundenen Aufwand. Die Verwendung von Open Source Software bietet jedoch ein angemessenes Preis-Leistungsverhältnis. „Eines der signifikanten Vorteile bei der Nutzung von Open Source Software ist das Fehlen jeglicher Anschaffungskosten. Insbesondere der Mittelstand, der durchschnittlich 20 bis 80 EDV-Arbeitsplätze samt der notwendigen Infrastruktur unterhalten muss, profitiert von der lizenzkostenfreien Alternative. Je EDV-Arbeitsplatz können jährlich im Durchschnitt 1.000 bis 2.500 Euro alleine an Update- und Lizenzgebühren gespart werden. Durch die Einführung von Open Source Software lassen sich weiterhin die Kosten für Anschaffungen und Inbetriebnahme um mehr als 50% verringern.“¹

Die Erstellung einer Webpräsenz kann in vielen Fällen direkt im Anschluss an die Registrierung der gewünschten Internetadresse erfolgen. Vor der Veröffentlichung sollte jedoch das mit den Open Source Software-Elementen (Apache, PHP und MySQL) zusammengestellte System auf einer internen Anwendungsumgebung getestet werden.

Die Internet Service Provider haben sich auf die entsprechenden Technologien eingestellt und bieten verschiedene Konfigurationsmöglichkeiten an. Unter anderem kann darunter auch die Verwendung von PHP-Skripten auf den Webservern der Internet Service Provider erlaubt sein.

Ziel dieser Diplomarbeit ist es, den Aufbau eines Anwendungssystems zu beschreiben, das die Erstellung von dynamischen Websites unterstützt. Dabei wird darauf geachtet, dass es sich ausschließlich um Open Source Software-Elemente handelt, die frei im Internet zugänglich und kostenlos erhältlich sind.

¹ <http://www.riccut.de/produkte/oss>, 26.06.2003

1.2 Aufbau der Arbeit

Zu Beginn der Arbeit werden die Problemstellung und die Zielsetzung definiert sowie relevante Begrifflichkeiten erläutert.

Daraufhin folgt eine Vorstellung der in dieser Arbeit verwendeten Open-Source-Software. Außerdem werden im zweiten Kapitel Hilfsmittel zur Webseiten-erstellung beschrieben.

Das dritte Kapitel beschäftigt sich mit der Darstellung der Beziehungen unter den Open-Source-Software-Elementen.

Im vierten Kapitel wird auf die Implementierung der Software und insbesondere auf die Installation und Konfiguration der Programme eingegangen.

Im fünften Kapitel wird die Funktionalität des Anwendungssystems im Rahmen einer Fallstudie für ein fiktives „kinder Überraschung Eis“ getestet.

Das sechste Kapitel beinhaltet die abschließende Bewertung des Systems und die Darstellung der Ergebnisse in einer Nutzwertanalyse.

1.3 Begrifflichkeiten

Die Begriffe werden nach folgendem Schema in den Untergliederungspunkten definiert (s. Abb. 1):

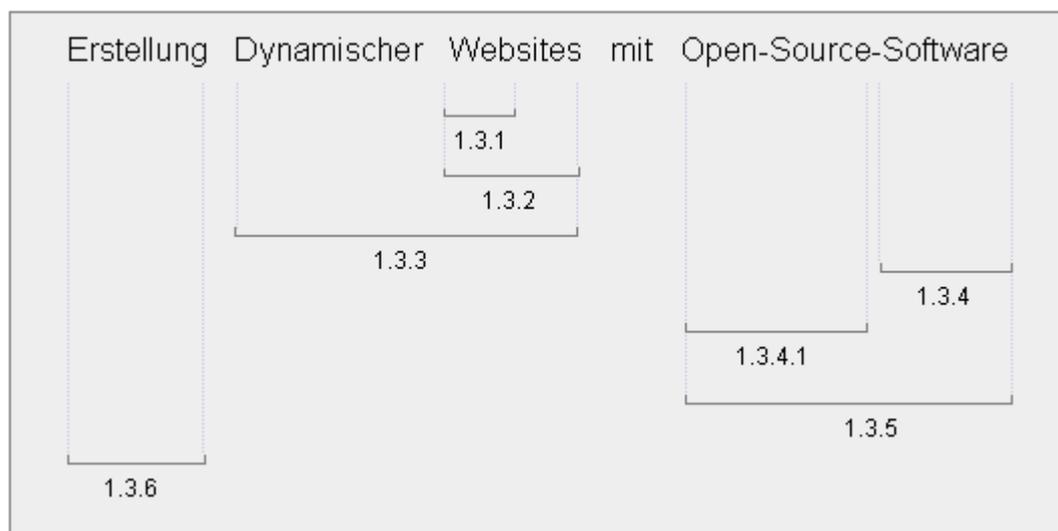


Abb. 1: Gliederungsschema für Begrifflichkeiten

1.3.1 World Wide Web (WWW)

Im allgemeinen Sprachgebrauch werden die Begriffe WWW und Internet fälschlicherweise synonym verwendet. „Als Internet wird die Einheit aller Personen, Firmen, Organisationen verstanden, die sich unter Einhaltung bestimmter Standards und Normen zusammenschließen. Das können auf der einen Seite Informationsanbieter (die Server) sein, auf der anderen Seite gehören auch alle Informationsnutzer (die Surfer [bzw. Clients]) dazu.“² Das Internet trägt den Beinamen „Netz der Netze“, „das weltweit andere Netze verbindet und mittels verschiedener Protokolle eine Reihe von Anwendungsdiensten, vor allem das *World Wide Web* [...], zur Verfügung stellt.“³

Die anwendungsnahen Dienste im Internet basieren auf standardisierten Protokollen, die zwischen Client und Server verwendet werden. Im Wesentlichen handelt es sich dabei um

- World Wide Web: Hypertext Transfer Protocol (→HTTP, S. 283),
- File Transfer: File Transfer Protocol (→FTP, S. 282),
- E-Mail: Simple Mail Transfer Protocol (→SMTP, S. 288),
- Usenet News: Network News Transfer Protocol (NNTP) und
- Terminal Emulation: Telnet Protocol (Telnet).

Das Standardisierungsgremium IETF (Internet Engineering Task Force) steuert und überwacht die Weiterentwicklung der im Internet verwendeten Protokolle. Entwurf und Veröffentlichung der Standards erfolgt in Form von RFC⁴ im Internet.

Viele der Kernprotokolle sind seit der Frühzeit des Internets weitgehend unverändert im Einsatz (z.B. Telnet seit 1972, FTP seit 1973, SMTP seit 1982). Die Aufmerksamkeit, die das Internet in den letzten Jahren in der Öffentlichkeit und in der Wirtschaft erfahren hat, ist in erster Linie auf das World Wide Web zurückzuführen.⁵

² Krause, J.: PHP 4, 2003, S. 41

³ Stahlknecht, P. u.a.: Wirtschaftsinformatik, 2002, S. 89

⁴ Die Abk. RFC steht für Requests for Comments und bezeichnet Dokumente, die Vorschläge für Standards, Technologien oder Vorgehensweisen innerhalb des Internets enthalten.

⁵ Vgl. Stahlknecht, P. u.a.: Wirtschaftsinformatik, 2002, S. 113

Der Begriff **World Wide Web** (auch als WWW, W3 oder kurz Web bezeichnet) wurde 1989 von Tim Berners-Lee geprägt. Das ursprüngliche Ziel war die Verknüpfung der auf verschiedenen Computersystemen vorhandenen Informationen⁶ (vor allem wissenschaftliche Texte) zu einem großen Informationssystem. Dadurch sollten die Informationsflüsse zwischen den Mitarbeitern verbessert und beschleunigt werden. Berners Lee entwickelte und kombinierte als Basis des Web drei einfache Protokolle/Spezifikationen⁷:

- Für die Adressierung einer Ressource⁸: URI (Uniform Resource Identifier)
- Für die Übertragung einer Ressource: HTTP (Hypertext Transfer Protocol)
- Für die Darstellung einer Ressource: HTML (Hypertext Markup Language)

Die erste Version der Web-Spezifikation wurde 1991 vom CERN (Abk. für: **Conseil Européen pour la Recherche Nucléaire**, europäisches Forschungszentrum für Teilchenphysik, <http://www.cern.ch>) freigegeben.⁹

Die Weiterentwicklung und Standardisierung des Web wird vom *World Wide Web Consortium* (→W3C, <http://www.w3.org>, S. 290) koordiniert. Dieses unabhängige Gremium wird von Industrie und Wissenschaft getragen und von vielen Staaten unterstützt.¹⁰

Das zentrale Element des Web ist das multimediale Dokument (*Webseite*, S. 6), das verschiedene Medien wie Texte, Bilder, Audio- und Videodaten integriert und mit weiteren Dokumenten auch anderer Rechner im Internet über Querverweise (*Links*) verknüpft (*Hypertext-Prinzip*). Die Eingangsseite zu einem mehrere Seiten umfassenden Informationsangebot wird als *Homepage* bezeichnet.

Das Web basiert auf dem *Client-Server-Modell*. Die grafische Aufbereitung von Webseiten erfolgt auf der Client-Seite vorwiegend in sog. *Webbrowsern* (S. 90), die das Blättern und Navigieren durch verknüpfte Dokumente ermöglichen.

⁶ →Information, S. 283. Umfassende Hinweise zu Grundbegriffen gibt die Norm DIN 44300 (*Grundbegriffe der Informationsverarbeitung*)

⁷ Vgl. TEIA: Apache, 2001, Abschnitt 1.1, <http://www.teia.de>, 24.05.2003

⁸ →Ressource, S. 287

⁹ Vgl. Hansen, H. R. u.a.: Wirtschaftsinformatik, 2001, S. 1192

¹⁰ Vgl. Stahlknecht, P. u.a.: Wirtschaftsinformatik, 2002, S. 113

Eine Webseite wird auf Anfrage des Webbrowsers von einem *Webserver* geliefert. Die Kommunikation zwischen Webbrowser und Webserver wird durch das *HTTP* geregelt. (Weitere Informationen zu HTTP: siehe Anhang B).

Jede Webseite und jede andere Ressource wird über eine eindeutige Adresse in Form der *URL* definiert, bspw. `http://www.wi.fh-koeln.de/bwl/orga/` (Weitere Informationen zu URL: siehe Anhang A).

Die Seitenbeschreibungssprache *HTML* ist der Standard für die Darstellung der Informationen einer Webseite. Mit Hilfe vordefinierter Markierungen (Tags) können die Inhalte einer Webseite (Texte, Bilder usw.) angeordnet und formatiert werden. (Weitere Informationen zu HTML: siehe Anhang C).

1.3.2 Website

Unter einer Website (auch Webpräsenz, -auftritt oder einfach Site genannt) wird ein virtueller Ort im Web verstanden, an dem ein oder mehrere zusammengehörige HTML-Dokumente, sog. (Web-)Seiten zu finden sind, die durch eine einheitliche Navigation zusammengefasst und durch Hyperlinks miteinander vernetzt werden. Diesem Ort wird durch die URL eine eindeutige Adresse im Netz zugewiesen, die ihn von allen anderen Sites unterscheidbar macht und den gezielten Zugriff erlaubt.¹¹ Außer der URL-Adresse sind keine weiteren Informationen notwendig, um auf die damit bezeichnete Ressource im Netz zugreifen zu können.¹²

Die Homepage¹³ (die „Startseite“ eines Webauftritts) dient als Eingangsseite in ein solches Netz. Sie stellt die gesamte Website in einen thematischen Kontext und ermöglicht die Navigation durch dessen Teilinhalte. Von ihr aus wird der Besucher durch die Website interaktiv geführt. Dadurch wird die Homepage zum zentralen Orientierungspunkt einer Website und trägt wesentlich dazu bei, diese als eigenständige Einheit im Web zu identifizieren.

¹¹ Vgl. Kurz, A.: Net-Lexikon, <http://www.net-lexikon.de/Web-Site.html>, 11.04.2003

¹² Vgl. TEIA: Apache, 2001, Abschnitt 1.4.2, <http://www.teia.de>, 24.05.2003

¹³ Wird im allgemeinen Sprachgebrauch fälschlicherweise oft synonym mit Website verwendet.

1.3.3 Dynamische vs. Statische Webseiten

1.3.3.1 Statische Webseiten

Am einfachsten werden Informationen im Web in Form von statischen Webseiten zur Verfügung gestellt. Ihr Zweck ist die Darstellung von Informationen auf der Basis eines Hypertextsystems: Die in statischen Webseiten enthaltenen Informationen werden mittels HTML-Tags formatiert und als fertige Datei auf der Festplatte des Servers abgelegt. Wird die zu dieser Datei passende URL aufgerufen, liest der Webserver die zugehörige Datei von seiner Festplatte und stellt sie dem Webbrowser ohne Änderungen zur Verfügung (s. Abb. 2).

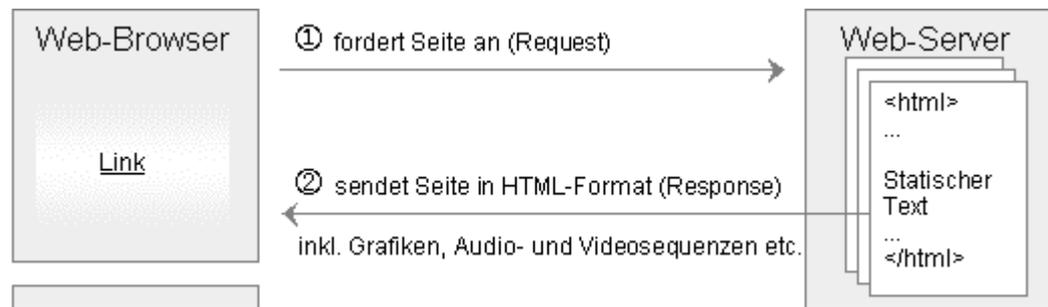


Abb. 2: Kommunikationsschema für statische Webseiten ¹⁴

Statische Webseiten werden per Hand erstellt, gespeichert und sind damit festgelegt. Die Informationen, die dem Besucher bei statischen Webseiten übermittelt werden, sind fest in den HTML-Dateien verankert. „Eine Trennung von Format und Inhalt ist nicht vorgesehen. Wie bei einem gedruckten Buch wird die Seite immer so angezeigt, wie sie erstellt wurde. Vorteil gegenüber einem Buch sind die Verweise auf verwandte Artikel durch Hyperlinks.“¹⁵ Nachteil der statischen Webseiten ist, dass jede Änderung des dargestellten Inhalts manuell bzw. über entsprechende Tools in eine HTML-Seite eingefügt werden muss, was bei häufigen Änderungen mit einem hohen Aufwand verbunden ist.

¹⁴ Vgl. Saamer, S.: Portale, 2000, S. 21

¹⁵ Saamer, S.: Portale, 2000, S. 21

Für kleinere Websites mit einer überschaubaren Zahl an HTML-Dokumenten und seltener Aktualisierung ist das statische Verfahren ausreichend. Sobald die Zahl der Seiten wächst und die Site möglichst täglich aktualisiert werden soll, sind andere Techniken notwendig, Dynamische Verfahren, bei denen der Inhalt der Webseite, nicht die Webseite selbst ausgetauscht wird.

1.3.3.2 Dynamische Webseiten

Im Unterschied zu statischen Webseiten, die in unveränderter, für alle Benutzer gleicher Form zum Abruf bereitstehen, werden dynamische Webseiten erst dann erzeugt, wenn der Benutzer sie anfordert. Mit dynamisch sind die Inhalte der Webseite gemeint, die sich nach aktuellen Erfordernissen (wie Benutzereingaben) ändern. Diese Inhalte können aus verschiedenen externen Informationsquellen (z.B. Datenbanken) stammen. „Der Begriff Dynamische Webseiten bezeichnet nicht Webseiten mit Multimedia-Effekten, sondern solche Seiten, die – im Gegensatz zu statischen Webseiten – erst auf Anfrage eines Besuchers hin, quasi ‚zur Laufzeit‘, aus einem Datenpool erstellt werden. Der Server bezieht Inhalte dynamischer Seiten (die sogenannte ‚dynamische Inhalte‘ oder ‚Dynamic Content‘) aus einer Datenquelle wie bspw. eine MySQL-Datenbank oder eine XML^[16]-Datei.“¹⁷

Im Gegensatz zu statischen Webseiten enthalten dynamische Webseiten neben dem HTML-Code weitere Programmieranweisungen, die auf dem Webserver ausgeführt werden und die zurückgesendete HTML-Seite erst generieren. Die Webseite kann so individuell und zeitnah auf unterschiedliche Ereignisse, bspw. Formulareingaben, reagieren (s. Abb. 3).

¹⁶ XML steht für eXtensible Markup Language.

¹⁷ Martins, F. P. u.a.: Dynamic Content in: Internet Intern: PC Praxis-Intern, 2/03 Mai-Juli, S. 50

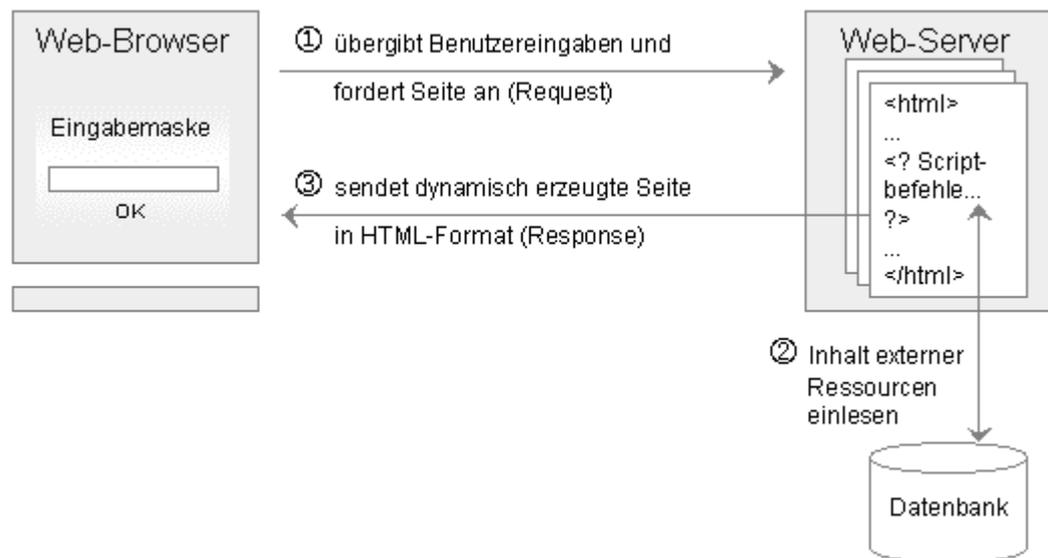


Abb. 3: Kommunikationsschema für dynamische Webseiten¹⁸

Dynamische Webseiten müssen nicht zwangsläufig von einer Datenbank unterstützt werden.¹⁹ Kontaktformulare, Counter und einfache Gästebücher lassen sich bspw. ohne Datenbank erstellen. Dagegen benötigen z.B. Adressbücher, News- und Katalogsysteme i.d.R. eine Datenbank. „Heute [erlangt] die Integration von Informationssystemen in Webprojekte und der Zugriff auf Informationsressourcen in einer verteilten Umgebung eine wachsende Bedeutung. Umfragen in Webagenturen zeigen, dass heute mehr als 80 % aller Webapplikationen direkt oder indirekt mit der Webanbindung von Datenbanksystemen zu tun haben.“²⁰ „Allen Systemen ist gemeinsam, dass Webinhalte nicht mehr statisch, sondern dynamisch erzeugt werden. Zur Verwaltung der Webinhalte werden in der Regel Datenbanken benutzt. Besonders viele Systeme setzen dabei auf kostengünstige Open-Source-Komponenten wie das Datenbanksystem MySQL, den Webserver Apache und die Scriptsprache für serverbasierte Anwendungen PHP.“²¹

¹⁸ Vgl. Saamer, S.: Portale, 2000, S. 23

¹⁹ Vgl. Krause, J.: PHP 4, 2003, S. 34

²⁰ Däßler, R.: MySQL, 2003, S. 35

²¹ Siehe FN 20

Vorteile dynamischer Websites

Zusammenfassend lassen sich die folgenden Vorteile dynamischer Websites nennen²²:

- Trennung von Layout und Inhalt

Da dynamische Webseiten aus *verschiedenen Informationsquellen* erzeugt werden, ist eine Trennung von Layout (design) und dem Inhalt dynamischer Webseiten (dynamic content) möglich. (Die „Rohinformationen“ sind meist strukturiert in Form von Datenbanken hinterlegt und werden z.B. mit HTML-Templates²³ kombiniert, um im Webbrowser darstellbare Seiten zu erhalten.) Der Pflegeaufwand statischer Webseiten ist hoch, z.B. wenn eine Site-weite Layoutänderung durchgeführt werden soll. Bei dynamischen Sites werden in diesem Fall nur HTML-Templates oder eine allgemeine Layout-Vorschrift (Document Type Definition, DTD²⁴) geändert, und alle hiernach erzeugten Seiten verwenden diese Änderung automatisch wieder.

- Zeitnahe Erstellung der dargestellten Informationen (Aktualität)

Weil die Seiten dynamischer Websites in kurzen Abständen oder zeitnah, d.h. „im Augenblick des Abrufes“²⁵ erzeugt werden, stellen Sie den *aktuellen* Informationsstand dar.

- Interaktivität

Die zeitnahe Erzeugung der Seiten ermöglicht es, auf Nutzereingaben *reagieren* zu können. Dies ist für den Erfolg einer Website entscheidend, denn „dynamisch erstellte Webseiten machen erst Sinn, wenn der Nutzer damit interagieren kann.“²⁶

²² Vgl. Lillich, J.: Dynamische Websites, <http://www.unix-ag.uni-kl.de/...>, 11.04.2003

²³ →Templates, S. 289

(Weitere Informationen zur Arbeit mit Templates: McCarty B.: PHP 4, 2002, S. 427ff)
 Definitionsdatei (.css-Datei, Cascading Style Sheets).

²⁵ Krause, J.: PHP 4, 2003, S. 33

²⁶ Krause, J.: PHP 4, 2003, S. 333

Client- und serverseitige Verarbeitung dynamischer Websites

Das Standardformat des Web ist HTML. Diese Seitenbeschreibungssprache wurde für die Präsentation von statischen Inhalten konzipiert. Die ursprünglich einzigen dynamischen Elemente waren Hyperlinks, mit denen sich andere statische Seiten und Grafiken aufrufen ließen. HTML ist nicht in der Lage, dynamische Inhalte (also Inhalte, die von Aktionen des Nutzers abhängig sind) anzuzeigen und zu verarbeiten. Um dieser Beschränkung abzuweichen, sind Erweiterungen entwickelt worden, mit denen sich HTML-Seiten um interaktive Funktionen ergänzen lassen. „Alle HTML-Erweiterungen sind dazu gedacht, Daten und Benutzereingaben zu verarbeiten.“²⁷

Da eine Web-Anwendung eine Client/Server-Anwendung ist, lassen sich die aktuellen Technologien nach dem Ort der Verarbeitung unterscheiden:

- Die Verarbeitung erfolgt auf dem Client, also im Webbrowser des Nutzers. JavaScript und VBScript funktionieren nach diesem Prinzip.
- Die Verarbeitung erfolgt auf dem Server. Dieses Verfahren wird u.a. von CGI (Common Gateway Interface, bspw. Perl), JSP (Java Server Pages), ASP (Active Server Pages) und PHP (PHP Hypertext Preprocessor) verwendet.

Diese Unterscheidung hat Konsequenzen: Die Verarbeitung auf dem Client setzt voraus, dass alle Ressourcen auf dem Client verfügbar sind. Da client-basierte Skriptsprachen keine Funktionen besitzen, um auf das Dateisystem oder auf eine Datenbank zuzugreifen, muss dies mit server-seitigen Skripten geschehen. Server-basierte Erweiterungen sind i.d.R. leistungsfähiger, weil sie auf alle Ressourcen zugreifen können, die sich auf dem Server befinden. Dies ermöglicht erst die Durchführung häufig vorkommender Vorgänge wie z.B. die Suche nach bestimmten Daten, die Auswahl spezifischer Informationen oder die Übermittlung von Daten an den Server.

²⁷ o.V.: S. 15, <http://www.czernohous.de/php/php-skript.pdf>, 11.04.2003

1.3.3.3 Anwendungsbeispiele

... für Statische Webseiten

Statische Seiten sind typisch für Webauftritte mit *Plakatcharakter*, bei denen keine direkte Rückkopplung zum Anbieter (Server) vorgesehen ist und deren Inhalte selten oder nicht aktualisiert werden (s. Abb. 4).



Abb. 4: Beispiel für eine statische Webseite²⁸

Private Homepages, Hilfesysteme oder Online-Dokumentationen basieren oft auf statischen Webseiten.

²⁸ <http://multimedia.zawiw.uni-ulm.de/~webdemo/webkurs/static-de.html>, 10.06.2003

... für Dynamische Webseiten

Folgende Abbildung zeigt eine Website, die deutlich sichtbar mit dynamisch generierten Inhalten arbeitet.

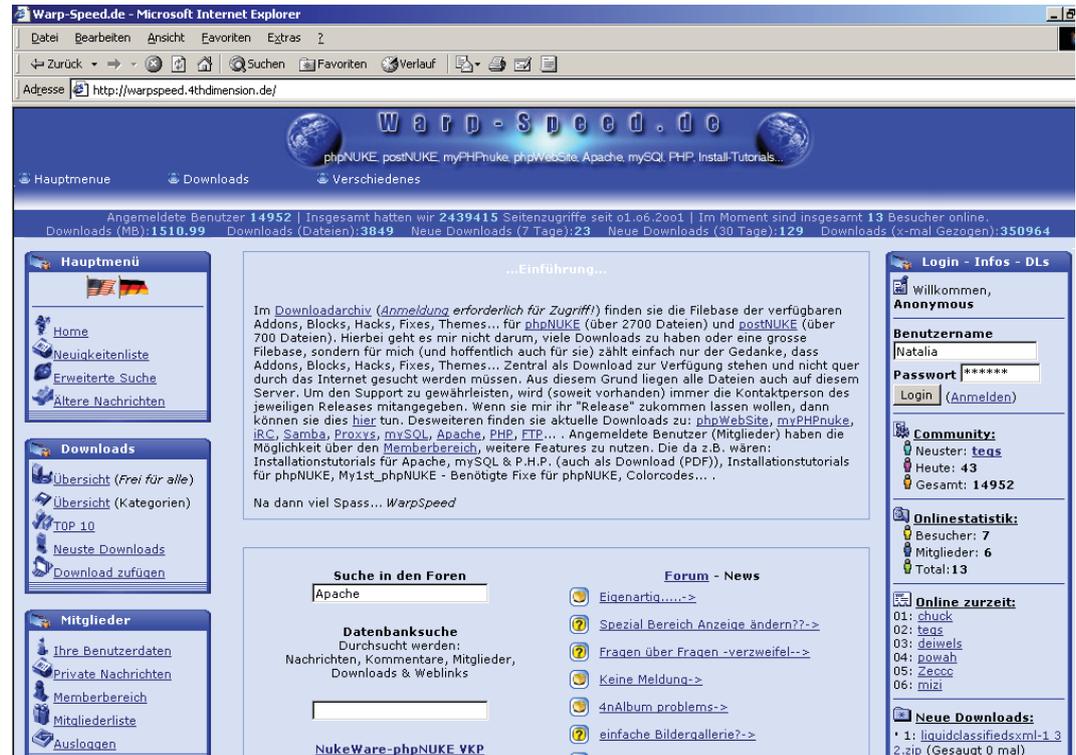


Abb. 5: Beispiel für eine Dynamische Website ²⁹

Ein Beispiel: Der Benutzer erfasst den Namen eines Autors in einem Formular. Der Webbrowser sendet diese Daten an den Webserver, der nach den verfügbaren Buchtiteln dieses Autors in einer Datenbank sucht. Das Suchergebnis wird zu einer HTML-Seite automatisch zusammengestellt und zum Client gesendet.

Oder hat sich z.B. der Benutzer durch einen Login-Mechanismus identifiziert, werden ihm wichtige Informationen hervorgehoben, das Seitenlayout entspricht seinen optischen Vorlieben.

²⁹ Warp-Speed.de: <http://warpspeed.4thdimension.de>, 10.06.2003

1.3.4 Software

Eine kennzeichnende Eigenschaft von Rechnern ist, dass sie programmierbar sind. Dies bedeutet, dass dem Rechner mitgeteilt werden muss, welche Arbeitsschritte er verrichten soll. Die Summe der Arbeitsschritte zur Verrichtung einer Aufgabe wird als *Programm* bezeichnet.³⁰

Software ist „der Sammelbegriff für (Computer-)Programme“.³¹ Der Begriff Software bezeichnet alles, „was auf einem Computer ausgeführt wird, sowohl das fertige, nur auf einer Architektur ausführbare Programm (Binary) als auch den vom Programmierer geschriebenen, menschenlesbaren Programmtext (Source).“³² „Software ist strikt betrachtet nicht mit Daten identisch, da Daten keine Instruktionen enthalten und auf dem Rechner nicht ausgeführt, sondern verarbeitet werden.“³³

Software wird in drei Klassen eingeteilt: **Systemsoftware** (Systemprogramme), **Entwicklungssoftware** (Entwicklungsprogramme) und **Anwendungssoftware** (Anwendungsprogramme) (s. Abb. 6).

„Die *Systemsoftware* stellt die grundlegenden Dienste für andere Programme zur Verfügung, insbesondere den Zugriff auf eine konkrete Rechnerplattform. Die zentralen Dienste der Systemsoftware werden zusammenfassend auch als *Betriebssystem* bezeichnet.“³⁴

Das Betriebssystem (S. 39) ist das Bindeglied zwischen der Hard- und Software, so dass durch Treiberprogramme die Hardware erst angesprochen werden kann und der Anwender über andere Software (z.B. Anwendungsprogramme) in Interaktion mit dem Rechner treten kann. Dabei „steuert und überwacht [das Betriebssystem] die Ausführung von Anwendungsprogrammen auf der Maschinenkonfiguration (der Rechnerhardware).“³⁵

³⁰ Vgl. Hansen, H. R. u.a.: Wirtschaftsinformatik, 2001, S. 150

³¹ Hansen, H. R. u.a.: Wirtschaftsinformatik, 2001, S. 150

³² o.V.: kurz, http://www.oreilly.de/german/freebooks/os_tb/toc.html, 10.04.2003

³³ Kurz, A.: Net-Lexikon, <http://www.net-lexikon.de/Software.html>, 10.04.2003

³⁴ Siehe FN 31

³⁵ Siehe FN 31

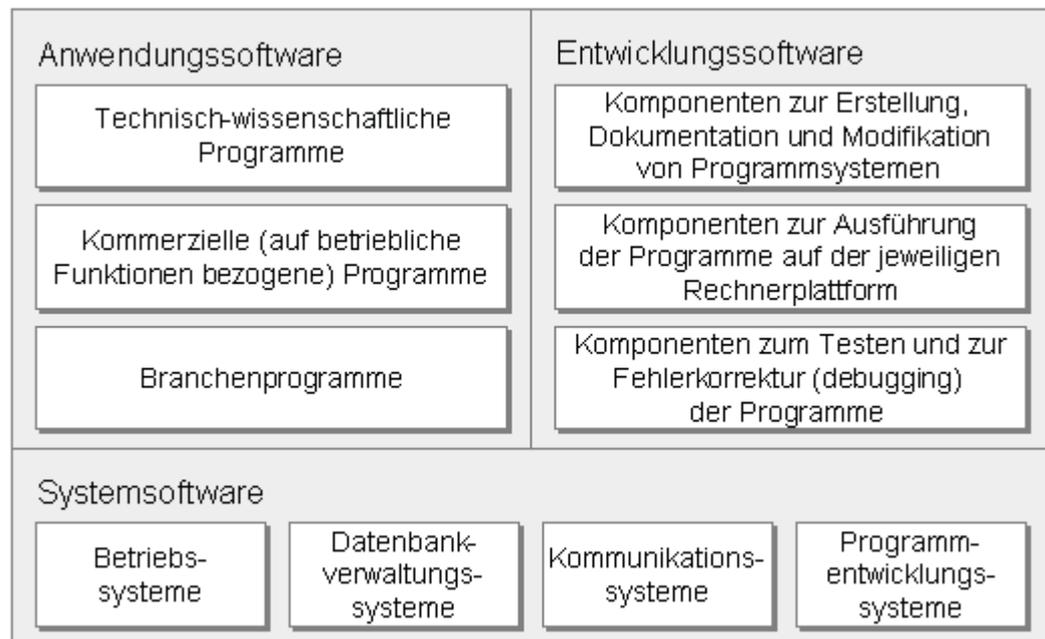


Abb. 6: Klassifizierung von Software³⁶

„Die *Entwicklungssoftware* setzt auf dem Betriebssystem auf und ermöglicht die Programmierung [...] beliebiger Programme.“³⁷ Die Entwicklungssoftware besteht aus den in der Abb. 6 genannten Komponenten. Hierbei wird häufig zwischen einer Entwicklungs- und einer Laufzeitumgebung unterschieden, wobei letztere nur für die Ausführung der entwickelten Software notwendig ist.³⁸

Ein wichtiger Teil einer Entwicklungsumgebung ist die zugehörige Programmiersprache. „Programme werden in einer *Programmiersprache* entwickelt und enthalten detaillierte Anweisungen an den Rechner, wie eine bestimmte Problemstellung zu lösen ist. Die Summe aller Programmiersprachenanweisungen innerhalb eines Programms wird als [...] *Quellprogramm* ([...] auch Quellcode oder Quelltext, engl.: source code) bezeichnet. Es kann mittels *Compiler* und/oder *Interpreter* [...] auf einem Rechner in den Maschinencode übersetzt und ausgeführt werden.“³⁹ Mit Hilfe der Entwicklungssoftware werden sowohl System- und Anwendungs- als auch die Entwicklungssoftware selbst erstellt.

³⁶ Vgl. Hansen, H. R. u.a.: Wirtschaftsinformatik, 2001, S. 151

Vgl.a. Hansen, H. R.: Wirtschaftsinformatik, 1992, S. 355

³⁷ Hansen, H. R. u.a.: Wirtschaftsinformatik, 2001, S. 151

³⁸ Siehe FN 36

³⁹ Siehe FN 37

Die *Anwendungssoftware* (Anwendungsprogramme, Anwendungen, Applikationen, engl.: application programm, user programm) wird zur Lösung der fachlichen Probleme der Anwender eingesetzt.⁴⁰ Dazu gehören die in der Abb. 6 genannten Programme. Voraussetzung für den Einsatz der Anwendungen ist die Systemsoftware.

Bei Anwendungssoftware wird zwischen Standard- und Individualsoftware unterschieden (s. Abb. 7).

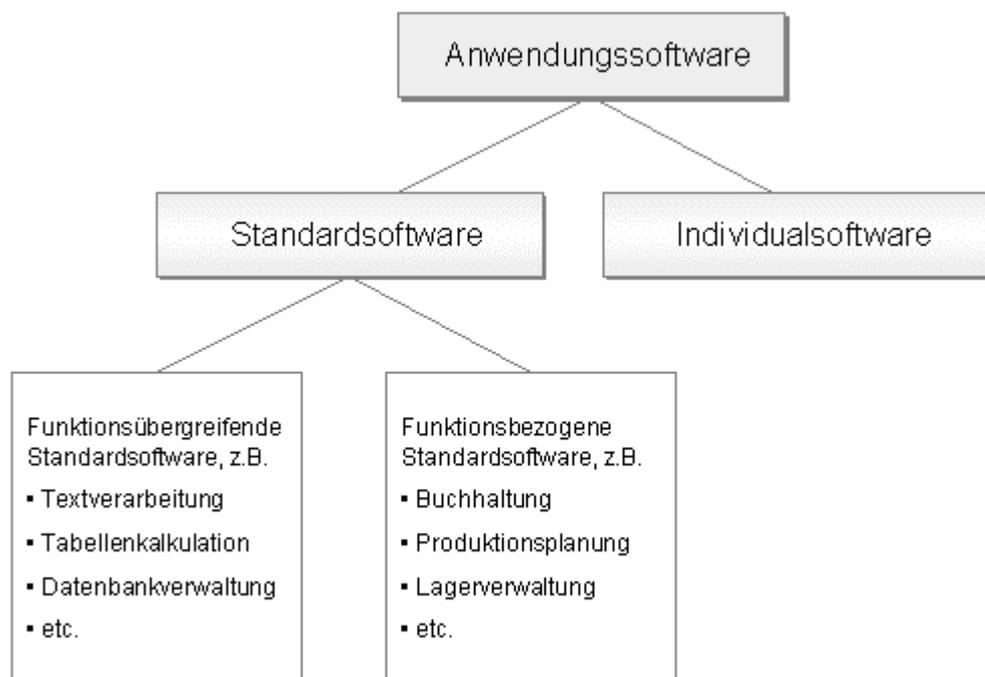


Abb. 7: Beispiele für Anwendungssoftware⁴¹

Als *Standardsoftware* „(Standardprogramme, engl.: packaged software) werden fertige Programme bezeichnet, die auf Allgemeingültigkeit und mehrfache Nutzung hin ausgelegt sind.“⁴²

Individualsoftware (Individualprogramme, engl.: custom software) umfasst hingegen Programme, „die entsprechend der Bedürfnisse eines einzelnen Anwenders entwickelt und nur von diesem eingesetzt“⁴³ werden.

⁴⁰ Siehe FN 36

⁴¹ Vgl. Keil, P.: Open Source, S. 7, <http://www.it-vergabe.de/...>, 8.06.2003

⁴² Hansen, H. R. u.a.: Wirtschaftsinformatik, 2001, S. 152

⁴³ Keil, P.: Open Source, S. 6, <http://www.it-vergabe.de/...>, 8.06.2003