

jetzt lerne ich

**Start
ohne
Vorwissen**

HTML5

CHRISTOPH PREVEZANOS


Markt+Technik

Inhaltsverzeichnis

Willkommen	13
1 Einstieg – Nützliche Hintergründe zum Einlesen	15
1.1 Warum eigentlich noch HTML lernen?	16
1.1.1 Eigene Webdokumente per Hand schreiben	16
1.1.2 HTML-Code aus Editoren aufräumen und optimieren	17
1.1.3 Templates für CMS-Systeme erstellen	18
1.2 HTML – Hintergründe und Entwicklung	19
1.2.1 Die Entstehung von HTML	20
1.2.2 Der Umstieg von HTML 4 zu HTML 5	23
1.2.3 Die Praxistauglichkeit von HTML 5	24
1.3 Was Sie über das Internet wissen sollten	25
1.3.1 WWW – das World Wide Web	25
1.3.2 Der Browser als Benutzeroberfläche	28
1.3.3 Was Sie zu Webservern wissen sollten	32
2 Grundbefehle zum Erstellen von HTML-Dokumenten	37
2.1 So funktionieren HTML-Tags	38
2.1.1 Die Syntax von HTML-Tags	38
2.1.2 Der Umgang mit Leerzeichen und Tabstopps	40
2.1.3 Verbotene Zeichen und Symbole	41
2.2 Der grundlegende Seitenaufbau	42
2.2.1 DOCTYPE – was haben Sie überhaupt vor?	42
2.2.2 HTML – das Dokument eröffnen und beenden	43
2.2.3 HEAD – der Kopf Ihres Dokuments	43

2.2.4	CHARSET – den Zeichensatz definieren	44
2.2.5	TITLE – Ihrem Dokument einen Titel geben	45
2.2.6	BODY – ab hier gibt es Inhalte	45
2.3	Abschnitte, Bereiche & Überschriften definieren	47
2.3.1	H1 – Überschriften festlegen	47
2.3.2	HGROUP – Überschriften gruppieren	50
2.3.3	SECTION – Sinnabschnitte definieren	51
2.3.4	ARTICLE – selbstständige Texte festlegen	52
2.3.5	ASIDE – Randinformationen deklarieren	53
2.3.6	NAV – Navigationselemente kennzeichnen	55
2.3.7	HEADER – Kopfbereiche anlegen	56
2.3.8	FOOTER – Fußbereiche erstellen	57
2.4	Übungen	58
3	Besondere Auszeichnungen für Absätze, Wörter & Elemente	59
3.1	Auszeichnungen für Absätze	59
3.1.1	P – Absätze erzeugen	60
3.1.2	BR – Zeilenumbrüche einfügen	61
3.1.3	NOBR & WBR – Zeilenumbrüche steuern	63
3.1.4	CITE – Titel, Sachnamen und Bezeichnungen	64
3.1.5	Q & BLOCKQUOTE – Zitate hervorheben	65
3.1.6	ADDRESS – Kontaktinformationen	66
3.1.7	FIGURE – Illustrierende Elemente deklarieren	67
3.1.8	PRE – Textpassagen vorformatieren	68
3.1.9	HR – Linien für die Texttrennung	70
3.1.10	Kommentare in den HTML-Code einfügen	70
3.1.11	ü – Sonderzeichen im Text verwenden	71
3.2	Auszeichnungen für interaktive Elemente	73
3.2.1	SCRIPT – Java-Skripte in Dokumente einbetten	73
3.2.2	DETAILS – Zusatzinformationen per Widget	75
3.2.3	MENU – eigene Menüs in Webseiten erstellen	76
3.2.4	COMMAND – Befehle in Menüs erstellen	78
3.2.5	CANVAS – der interaktive Multimedia-Container	79
3.3	Auszeichnungen für Einzelelemente	82
3.3.1	EM & STRONG – wichtige Worte betonen	82
3.3.2	B – Wörter hervorheben	83
3.3.3	I – Stimmungen, Dialoge & Eigennamen	83
3.3.4	MARK – wichtige Wörter farblich markieren	84
3.3.5	SUP & SUB – Wörter hoch- oder tiefstellen	85
3.3.6	SMALL – Wörter kleiner schreiben	85
3.3.7	TIME – Zeit und Datum markieren	86
3.3.8	ABBR – Abkürzungen markieren	87
3.3.9	DFN – Definitionen deklarieren	87

3.3.10	CODE, VAR, SAMP & KBD – Computertexte	88
3.3.11	RUBY – Aussprache für asiatische Texte	89
3.3.12	INS & DEL – Überarbeitungen markieren	90
3.4	Übungen	91
4	Aufzählungen und Listen erstellen	93
4.1	UL – unsortierte Listen erstellen	94
4.2	OL – nummerierte Listen erstellen	95
4.2.1	Die Werte von nummerierten Listen anpassen	96
4.3	DL – Definitionslisten erstellen	97
4.3.1	Listen ineinander verschachteln	99
4.4	Übungen	102
5	Links – Webseiten miteinander verknüpfen	103
5.1	So funktionieren Links – Sprungmarken und Anker	103
5.2	Interne und externe Links erstellen	105
5.2.1	Links auf externe Webseiten	106
5.2.2	Interne Links erstellen	109
5.2.3	Verweise zu bestimmten Textstellen erzeugen	112
5.2.4	Ziele zum Öffnen von Links angeben	115
5.3	Weitere Funktionen für Ihre Links	118
5.3.1	Medium, Sprache oder Typ angeben	118
5.3.2	Eine Link-Basis angeben	120
5.3.3	Link-Typen – definieren Sie Ihre Links	121
5.4	Links zu Funktionen – E-Mail, FTP & Co.	123
5.4.1	Verschlüsselte Verbindungen	124
5.4.2	Dateien per FTP herunterladen	124
5.4.3	E-Mails per Klick erstellen	127
5.4.4	Links auf Newsgroups	129
5.4.5	Sonstige Link-Funktionen	130
5.5	Übungen	132
6	Bilder in die Webseite einbauen	133
6.1	Die Grafikformate im Überblick	134
6.2	So fügen Sie Bilder richtig ein	137
6.3	Die Größe der Bilder anpassen	142
6.4	Grafische Links erstellen	145
6.5	Image Maps – Bilder mit anklickbaren Bereichen erstellen	148
6.6	Übungen	154
7	Tabellen: Komplexe Inhalte übersichtlich darstellen	155
7.1	Der Grundaufbau von Tabellen	156
7.1.1	TABLE – Tabellen öffnen und schließen	156

7.1.2	TR & TD – Zeilen und Spalten definieren	157
7.1.3	Spaltenüberschriften hinzufügen	160
7.1.4	Eine Überschrift für die Tabelle	161
7.2	Tabellen strukturieren – Kopf-, Haupt- und Fußbereich	163
7.2.1	Einen Tabellenkopf deklarieren	163
7.2.2	Der Hauptteil – der Tabellenkörper	164
7.2.3	Abschluss – der Fußbereich	165
7.3	Tabellenzellen miteinander verbinden	167
7.3.1	Eine Zelle über mehrere Spalten spannen	168
7.3.2	Eine Zelle über mehrere Zeilen spannen	170
7.3.3	Zeilen und Spalten gleichzeitig kombinieren	172
7.4	Den Aufbau der Tabelle vorgeben	174
7.5	Übungen	179
8	Multimedia – Videos, Musik, Flash & Java	181
8.1	Die Grundbefehle zum Einbinden von Objekten	182
8.1.1	OBJECT – der Standardbefehl für Multimedia	182
8.1.2	EMBED – der alternative Befehl	188
8.1.3	IFRAME – universell einsetzbar	190
8.2	Videos in die Homepage einbinden	197
8.2.1	VIDEO – Filme direkt im Browser einbinden	197
8.2.2	Alternative Möglichkeiten für Videos	201
8.2.3	Das richtige Format für Ihre Internet-Videos	206
8.3	Arbeiten mit Audiodateien	209
8.3.1	AUDIO – der Befehl extra für Musikinhalte	209
8.3.2	Audiodateien mit anderen Befehlen einbinden	211
8.3.3	Audioformate für das Internet	214
8.4	Flash-Elemente richtig verwenden	216
8.5	Java-Applets in der Homepage nutzen	220
8.6	ActiveX in der Webseite nutzen	223
8.7	Geodaten, Drag & Drop, Offline-Arbeit & Co.	225
8.8	Übungen	230
9	Formulare erstellen und verarbeiten	231
9.1	Das Grundgerüst für Formulare erstellen	232
9.1.1	Hintergrund – so funktionieren Formulare	232
9.1.2	Ein neues Formular definieren	234
9.1.3	Einfache Eingabefelder erstellen	239
9.1.4	Schaltflächen für die Verarbeitung	242
9.1.5	Bereiche und Gruppen definieren	244
9.2	Eingabefelder für Text erstellen	247
9.2.1	Einfache Textfelder erzeugen	248
9.2.2	Die Feldgröße und Eingabelänge bestimmen	249
9.2.3	Beispiele oder Hilfetexte mit angeben	250

9.2.4	Werte vorgeben, sperren oder erzwingen	251
9.2.5	Textbereiche für längere Eingaben erzeugen	253
9.3	Spezielle Input-Typen für das Eingabefeld	256
9.3.1	Suchbegriffe deklarieren	257
9.3.2	E-Mail-Adressen überprüfen	257
9.3.3	Felder für URLs – Internet-Adressen	258
9.3.4	Felder für die Telefonnummer	259
9.3.5	Felder für Nummern und Zahlen erstellen	260
9.3.6	Das Datum mit speziellen Feldern darstellen	261
9.3.7	Eingabefelder für die Uhrzeit	262
9.3.8	Wertebereiche – Maximum, Minimum und Schritte	264
9.3.9	Passwörter im Browser ausblenden	266
9.3.10	Einen Farbwähler erstellen	267
9.3.11	Unsichtbare Informationen mitschicken	267
9.3.12	Dateien per Formular hochladen	267
9.4	Buttons, Checkboxes, Listen und Gruppen erstellen	269
9.4.1	Radio-Buttons – Auswahl-Knöpfe erstellen	270
9.4.2	Checkboxen – anklickbare Listen hinzufügen	273
9.4.3	Ausklappbare Auswahllisten erstellen	275
9.4.4	Lange Auswahllisten mit Gruppen strukturieren	280
9.4.5	Datenlisten – unsichtbare Hilfsdaten anbieten	282
9.5	Weitere Formularelemente	284
9.5.1	BUTTON – beliebige Schaltknöpfe erstellen	285
9.5.2	KEYGEN – der Schlüsselgenerator	286
9.5.3	PROGRESS – den Fortschritt anzeigen	287
9.5.4	METER – Füllbalken erzeugen	287
9.5.5	PATTERN – Zahlenfolgen überprüfen	288
9.5.6	OUTPUT – Ergebnisse deklarieren	289
9.6	Kleiner Exkurs: CGIs mit Perl und PHP	289
9.7	Übungen	294
10	Metatags – Zusatzinformationen für die Welt	295
10.1	So funktionieren Meta-Tags	296
10.2	Hintergrundinformationen zum Dokument	297
10.3	Erstellungsdatum, Events oder Ablaufdatum	299
10.4	Suchmaschinen einladen oder abweisen	301
10.5	Sonstige Meta-Informationen	305
10.6	Übungen	307
11	Exkurs HTML 4: Framesets in Webseiten	309
11.1	Das Frame-Konzept	309
11.2	Die Grundbefehle für Framesets	312
11.3	Die Frames mit Inhalten füllen	314

12	Cascading Style Sheets – erste Schritte in der Webformatierung	317
12.1	Einstieg – so funktioniert CSS	318
12.1.1	STYLE – CSS-Formatierungen einleiten	319
12.1.2	Die Syntax für CSS-Formatierungen	320
12.1.3	Farben mit CSS bestimmen	322
12.2	CSS richtig in HTML einbinden	323
12.2.1	Option 1 – Elemente direkt formatieren	324
12.2.2	Option 2 – CSS im Header deklarieren	324
12.2.3	Option 3 – CSS in eine externe Datei auslagern	325
12.3	HTML-Elemente mit CSS formatieren	327
12.3.1	Body-Bereich – das Layout der Seite festlegen	328
12.3.2	Schriften und Texte beliebig formatieren	330
12.3.3	Den Text und andere Elemente ausrichten	333
12.3.4	Den äußeren Abstand einstellen	337
12.3.5	Den inneren Abstand einstellen	338
12.3.6	Objekte ganz exakt ausrichten	340
12.3.7	Hintergrund, Rahmen und Bilder	348
12.3.8	Listen definieren und formatieren	353
12.3.9	Tabellen mit CSS formatieren	356
12.3.10	Links, MouseOver und Zeiger einstellen	361
12.4	Mit eigene Elementen und Bereichen arbeiten	366
12.4.1	DIV & SPAN – Bereiche definieren	366
12.4.2	CLASS und ID – Elemente identifizieren	367
12.4.3	Eigene CSS-Klassen definieren	367
12.4.4	Elemente mit selbst definierte Klassen formatieren	369
12.4.5	Bereiche mit selbst definierten Klassen formatieren	370
12.4.6	Bedingungen bei der Zuweisung von Klassen	370
12.5	Übungen	371
	Anhang A: Lösungen zu den Übungen	373
	Anhang B: Kurzreferenz HTML	395
	Anhang C: Kurzreferenz CSS	401
	Anhang D: Sonderzeichen	406
	Anhang E: MIME-Types	415
	Anhang F: Farben in HTML	421
	Anhang G: Links & Programme	425
	Stichwortverzeichnis	427

Einstieg – Nützliche Hintergründe zum Einlesen

Das Erlernen von HTML, das Verwalten eines Webserver und die Techniken der aktuellen Webbrowser stellen eigentlich drei eigene Themenbereiche dar. Trotzdem gehören sie einfach zusammen, wenn man sich mit der Webseiten-Erstellung beschäftigt. Wer eine vernünftige Homepage schreiben möchte, muss HTML beherrschen.

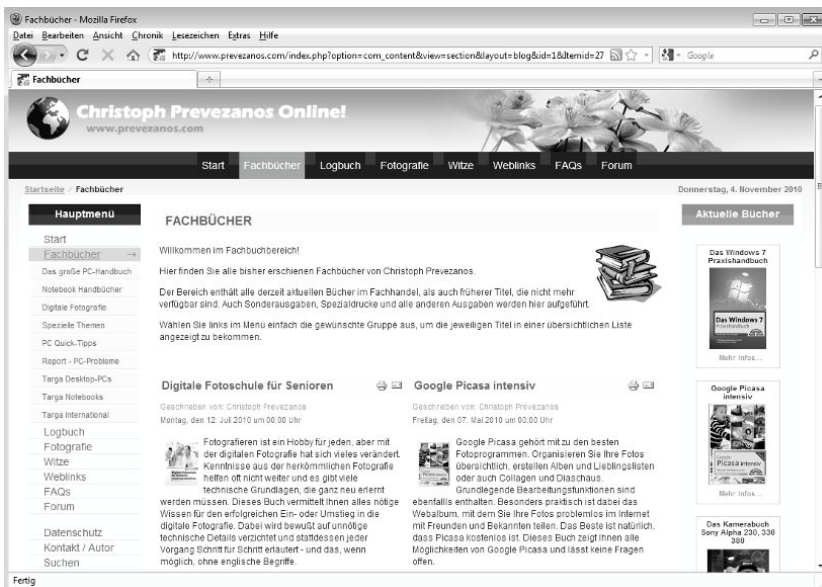


Abb. 1.1: Die eigene Homepage zu erstellen ist gar nicht schwer.

Außerdem sollte er über die Fähigkeiten und Funktionsweisen eines Webservers zumindest grundlegend informiert sein. Dadurch lassen sich die Projekte gezielter planen und die Möglichkeiten des World Wide Webs besser verstehen und umsetzen. Und zuletzt sollte natürlich jeder Homepage-Autor über die aktuellen Browser informiert sein. Immerhin zeigen sie das Ergebnis der ganzen Arbeit an. Beachtet man hier gewisse Standards oder auch Eigenheiten nicht, wird die Homepage möglicherweise falsch angezeigt.

1.1 Warum eigentlich noch HTML lernen?

Möchten Sie sich eine eigene Webseite erstellen, werden Ihnen hierzu überall Homepage-Baukästen, fertige Vorlagen, vollautomatische Verwaltungssysteme und ähnliche Hilfen angeboten. Sie müssen sich also fast gar nicht mehr um die Seite selbst kümmern, sondern nur noch um Ihre Inhalte. Da stellt sich natürlich schnell die Frage, warum Sie heute eigentlich noch HTML lernen sollten. Es gibt sehr viele Gründe hierfür, denn das gesamte World Wide Web basiert nach wie vor fast ausschließlich auf HTML. Jeder, der sich in irgendeiner Weise mit dem Thema Homepage oder Online-Inhalt beschäftigt, kommt zumindest um Grundkenntnisse nicht herum.

1.1.1 Eigene Webdokumente per Hand schreiben

Um eine eigene Webseite in HTML zu schreiben, benötigen Sie nicht mehr als einen Texteditor und HTML-Kenntnisse. Schon können Sie Ihre gesamte Seite mit allen Inhalten, Formatierungen und Grafiken erstellen, abspeichern und online stellen. Selbst aufwendige Webseiten mit Multimedia-Inhalten lassen sich auf diese völlig einfache Weise erstellen. Allerdings macht das in der Praxis wenig Sinn. Zum einen wäre dieses Vorgehen für große Webseiten viel zu kompliziert und unübersichtlich, zum anderen erhalten Sie völlig statische Seiten, die auch in Zukunft immer per Hand aktualisiert werden müssen. Für große Internetseiten und Portale stellen dynamische Verwaltungsprogramme, sogenannte **Content-Management-Systeme** (CMS), die deutlich bessere Wahl dar.

Trotzdem gibt es viele Bereiche, in denen das rein manuelle Erstellen von HTML-Code durchaus sinnvoll ist.

- Die erste kleine Homepage mit einer Handvoll Inhaltsseiten lässt sich so schon in kürzester Zeit erstellen, ohne dass Sie sich in komplizierte HTML-Editoren einarbeiten müssen. Deren Handhabung ist oft schwieriger zu erlernen als HTML selbst.
- Vielleicht möchten Sie nur schnell ein paar Inhalte für Freunde und Bekannte bereitstellen. Das können die Fotos vom letzten Urlaub sein, ein kleines Reisetagebuch oder eine Wegbeschreibung für die nächste Party. Auch für eine reine Online-Visitenkarte benötigen Sie weder aufwendige HTML-Editoren noch teuren Webspace bei einem Provider.
- Möchten Sie bei eBay ein paar Artikel einstellen, können Sie während der Eingabe die Artikel-Seite mit eigenen HTML-Befehlen gestalten. Schon mit sehr grundlegenden Befehlen geben Sie Ihrer Auktion ein interessanteres

Aussehen, heben die wichtigen Informationen hervor und platzieren die Fotos an der richtigen Stelle. Das steigert die Verkaufschancen und den Preis in der Regel erheblich.

- In vielen öffentlichen Foren, Gästebüchern oder Blogs darf man seine Beiträge oft mit zusätzlichen HTML-Befehlen individuell gestalten. So werden sie nicht übersehen, und Ihr Beitrag kommt richtig zur Geltung.

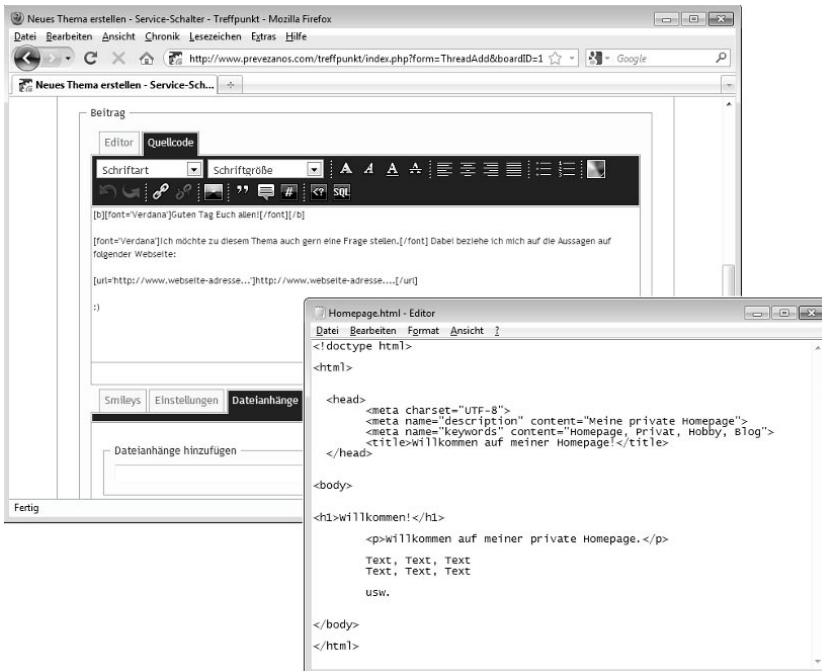


Abb. 1.2:
HTML für Auktionen, Foren, Gästebücher oder direkt im Editor

1.1.2 HTML-Code aus Editoren aufräumen und optimieren

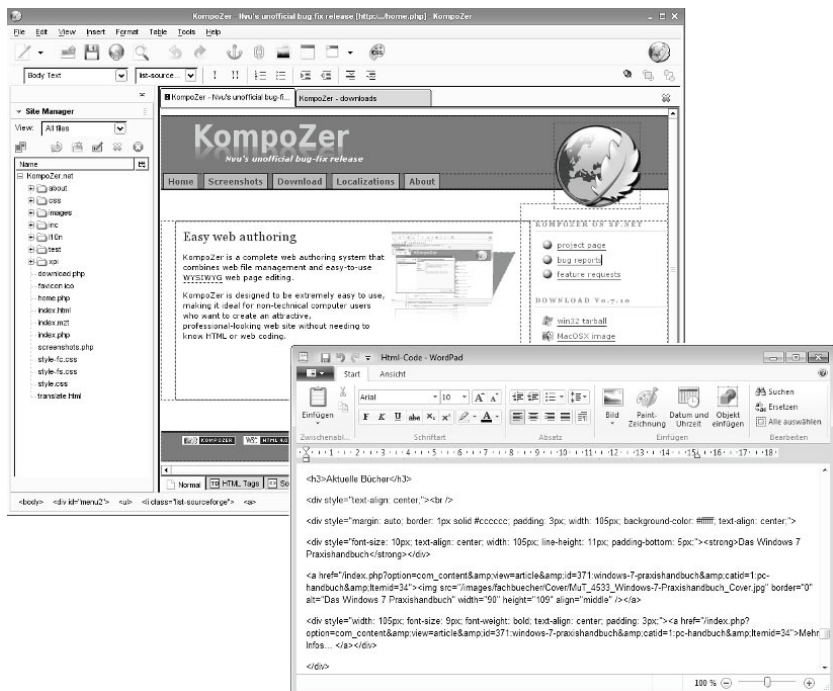
Möchten Sie eine größere Homepage mit vielen Unterseiten und einer aufwendigen Gestaltung erstellen, werden Sie sicherlich auf einen guten HTML-Editor zurückgreifen. Das Angebot ist sehr groß und reicht von kostenlosen Editoren aus dem Internet bis zu professionellen Werkzeugen wie Microsoft Expression Web (ehemals Frontpage) oder Adobe Dreamweaver. Unabhängig davon, für welchen HTML-Editor Sie sich entscheiden, ist das Prinzip doch bei allen Programmen gleich. Sobald Sie Ihre Homepage damit erstellt haben, werden die Inhalte in ganz normales HTML übersetzt und als herkömmliche Textdateien gespeichert.

- Möchten Sie kleine Änderungen an der fertigen Seite vornehmen, Inhalte nachträglich entfernen oder hinzufügen, müssen Sie nicht unbedingt das komplette Projekt wieder in dem HTML-Editor öffnen und erneut exportie-

ren. Das lässt sich auch ganz einfach direkt an den HTML-Seiten vornehmen und geht so meist viel schneller.

- Fast alle HTML-Editoren erstellen einen mehr oder weniger unsauberen Code. Oft werden einfache Formatierungen viel zu kompliziert umgesetzt, unnötige Fragmente bleiben zurück, oder es schleichen sich kleine Fehler ein. HTML basiert letztendlich auf einer logischen Strukturierung der Inhalte. Und genau diese Logik kann man von einer Software nur sehr begrenzt erwarten.

Abb. 1.3:
Den Code eines
HTML-Editors
aufräumen und
optimieren



1.1.3 Templates für CMS-Systeme erstellen

Statische HTML-Dateien sind ein Auslaufmodell und werden zunehmend durch dynamische Seiten ersetzt. Dabei kommen meist sogenannte **Content-Management-Systeme (CMS)** zum Einsatz. Ein CMS verwaltet die Inhalte in einer Datenbank auf dem Webserver. Diese Inhalte werden beim Besuch der Webseite aus der Datenbank ausgelesen und mittels einer Layout-Vorlage angezeigt. Diese Vorlage enthält sämtliche Formatierungen der Webseite, Schriften, Spalten, Farben, Grafiken und was noch so dazugehört. Es erfolgt also eine vollständige Trennung von Inhalt und Gestaltung.

Content-Management-Systeme sind nicht nur für professionelle Online-Redaktionen gedacht. Selbst die eigene kleine Homepage, das private Tagebuch oder die Homepage des Sportvereins lässt sich damit einfach und übersichtlich ver-

walten. Das Beste daran ist, dass Content-Management-Systeme nicht groß, teuer und aufwendig sein müssen. Es gibt eine ganze Reihe kostenloser Programme aus der Open-Source-Welt, die professionellen Systemen in nichts nachstehen. Zu den bekanntesten Vertretern gehören sicherlich Joomla! und WordPress.

Ist das Content-Management-System erst einmal installiert, müssen Sie sich darum kaum noch kümmern, weil es selbstständig auf dem Webserver läuft. Aber die Vorlage, das sogenannte Template, muss von Ihnen erstellt und gepflegt werden. Hierfür benötigen Sie sehr tiefgehendes Wissen über HTML und CSS, denn alle Ihre Inhalte werden vom Server automatisch durch diese Vorlage geschoben. Das Ergebnis landet dann im Browser des Besuchers. Es kann durchaus schwierig sein, eine komplette Webseite innerhalb einer großen Vorlage zu organisieren, aber es erspart viel Arbeiten und Mühe. Einfacher und schneller kann man das komplette Aussehen eines Webserver kaum verwalten.



Abb. 1.4:
Joomla!
und
WordPress als
CMS für Blogs &
Artikel

1.2 HTML – Hintergründe und Entwicklung

Die Abkürzung HTML steht für **Hypertext Markup Language**, und umgangssprachlich wird oft nur von **Hypertext** gesprochen. Der vollständige Name von HTML beschreibt im Grunde die Funktionsweise bereits sehr gut. HTML wird oft als **Auszeichnungssprache** bezeichnet, weil die Elemente eines Dokuments

markiert und somit als bestimmte Komponenten des Textes deklariert werden. Man könnte also auch durchaus von einer Gliederungssprache sprechen. In HTML wird also festgelegt, welcher Teil des Textes die Überschrift darstellt. Was ist Fließtext, wo beginnen und enden die Absätze, welches Element stellt ein Bild dar, wo gibt es Tabellen, Listen oder Aufzählungen ...? Im Grunde wird mit HTML ein Dokument nach seinen Abschnitten und Inhalten strukturiert und organisiert. Dazu kommen im Text selbst unsichtbare Inhalte, die Informationen über den Autor, das Erstellungsdatum oder andere Hintergrundinformationen beinhalten können. Sie dienen der Sortierung, der Organisation und dem Durchsuchen von größeren Textsammlungen.

Beim Erstellen von HTML-Seiten wird sehr häufig von **programmieren** gesprochen. Eigentlich ist das völlig falsch, denn in einer Programmiersprache werden anhand von Anweisungen, Schleifen, Variablen usw. logische Befehle und Abfolgen berechnet bzw. abgearbeitet. Damit hat HTML wirklich überhaupt nichts zu tun. HTML ist in keiner Weise eine Programmiersprache, und deshalb sollte man besser vom »Schreiben in HTML«, »Erstellen von HTML-Dokumenten« oder auch vom »Deklariere in HTML« sprechen. Sie können sich auch selber eine Bezeichnung aussuchen, aber sprechen Sie nicht vom »Programmieren«. Inzwischen ist es üblich, richtige Programme mit Anweisungen und logischen Operatoren in HTML zu integrieren. Allerdings handelt es sich dann um Java-Skripte, Perl, PHP oder Ähnliches. Das sind Programmiersprachen, die in HTML eingebettet werden können oder als Ergebnis HTML ausgeben, dadurch aber natürlich nicht Teil von HTML selbst werden.

1.2.1 Die Entstehung von HTML

Die Urversion von HTML entstand im Jahre 1989 und wurde von Tim Berners-Lee am CERN in Genf definiert. Dieser erste Entwurf beschränkte sich auf sehr einfache Standards. Der Text wurde ausschließlich logisch strukturiert, und die Elemente wurden entsprechend markiert. Schon sehr bald reichte das nicht mehr aus, und es kamen mit HTML 1.0 und HTML 2.0 viele Erweiterungen hinzu. So gab es nun z. B. Markierungselemente für Fettschrift, kursive Darstellung oder Unterstreichungen. Auch Bilder und Grafiken ließen sich nun integrieren. HTML entwickelte sich sehr schnell weiter und bot immer mehr Möglichkeiten, um den eigenen Text möglichst logisch und strukturiert zu markieren. Allerdings gab es bis dahin keinerlei gestalterische Möglichkeiten. Wie ein Text auszusehen hatte oder was für ein Layout gewünscht war, konnte in keiner Weise festgelegt werden. Diese Möglichkeiten wurden größtenteils mit den HTML-Versionen 3.0 bis 3.2 eingeführt. Auf einmal war es möglich, mit HTML das Aussehen der Dokumente stark zu beeinflussen und sie mit richtigen Formatierungen bzw. einem Layout auszustatten.

Mit der Zeit wurden die Dokumente immer komplexer und der Code immer unüberschaubarer. Die Vermischung von Inhalt und Formatierung führte zu großen Problemen, und fast jeder Browser interpretierte die Anweisungen unterschiedlich. Auf jedem Computer sah ein HTML-Dokument also anders aus. Mit der Version HTML 4.0 kehrte man im Grunde zu den Anfängen zurück und entfernte jegliche Formatierungen aus dem HTML-Code. Es fand wieder eine Trennung

zwischen den Inhalten und der Darstellung statt. So enthielt eine HTML-Datei jetzt nur noch den Inhalt, und die Formatierungen wurden in eine externe Datei ausgelagert. Das Prinzip der **Cascading Style Sheets** (CSS) war erfunden. Diese Trennung von Inhalten und Formatierungen stellt bis heute eine besonders wichtige Entwicklung dar, und sie ist immer noch nicht vollständig abgeschlossen. Nur so lassen sich komplexe Webseiten erstellen und verwalten, weil sie nicht mehr einzeln per Hand gepflegt und formatiert werden müssen.

Inzwischen sind wir bei HTML 5 angekommen. Die Entwicklung ist zwar noch lange nicht abgeschlossen, und das W3C-Konsortium spricht ausschließlich von einem »Working Draft« (= Arbeitsentwurf). Trotzdem ist schon jetzt klar, dass HTML 5 einen großen Sprung darstellen wird und vieles anders macht. HTML ist eben eine überaus lebendige und dynamische Sprache. Trotzdem werden Sie alles, was Sie heute über HTML lernen, auch in Zukunft gebrauchen können. Selbst wenn viele neue Befehle hinzukommen oder bestehende wieder entfallen, so bleibt das Grundprinzip doch immer gleich ... zumindest solange es das World Wide Web in der heute bekannten Form gibt und dies auf den Wurzeln von HTML basiert.

- Das Grundkonzept und somit die erste Version von HTML wurde im Jahre 1989 von Tim Berners-Lee festgelegt.
- Im Jahre 1992 wurde eine verbesserte Version von HTML veröffentlicht. Sie trägt noch keine Versionsnummer und dient nur der Strukturierung von Text.
- Im Jahre 1993 wurde HTML um neue Attribute wie Fettschrift, kursive Darstellung und die Einbindung von Bildern und Grafiken erweitert. Eine Versionsnummer gab es weiterhin nicht.
- Ende 1993 wurde an einer grunderneuerten Version mit dem Namen »HTML+« gearbeitet. Allerdings wurde diese nie offiziell verabschiedet oder zum Standard erklärt.
- Erst Ende 1995 erschien mit HTML 2.0 eine völlig neue Version von HTML, die grundlegend überarbeitet worden war und viele neue Funktionen bot, z. B. die Dateneingabe durch den Benutzer anhand von Formularen.
- Fast ein Jahr später sollte die Version HTML 3.0 verabschiedet werden. Allerdings wurde zu diesem Zeitpunkt der Browser Netscape Navigator in der Version 3 veröffentlicht, der so viele neue Funktionen bot, dass der neue HTML-Standard bereits veraltet war. HTML 3.0 wurde deshalb nie offiziell verabschiedet.
- Aus diesem Grund folgte Anfang 1997 die Version HTML 3.2, welche an die neuen Möglichkeiten der aktuellen Browser angepasst war. Es gab zahlreiche neue Funktionen wie z. B. den Aufbau von Tabellen, die Steuerung des Textflusses, das Einbinden von externen Programmen (sogenannten Applets) und vieles mehr.
- Ende 1997 folgte bereits die Version HTML 4.0. Diese stellt insofern einen Meilenstein dar, weil wieder eine strikte Trennung von Inhalten und Formatierungen erfolgte. Diese wurde anhand von Stylesheets vorgenommen. Außerdem gab es zahlreiche neue Funktionen.

- Fast zwei Jahre später folgte Ende 1999 die Version HTML 4.01. Sie zeichnete sich vor allem durch viele Anpassungen bestehender Befehle, Korrekturen und Optimierungen aus. HTML 4.01 ist der auch heute noch gültige Standard.
- An der Version HTML 5 wird bereits seit April 2008 gearbeitet. Sie soll viele neue Funktionen bieten und durch die Integration von XHTML vor allem dynamischen Inhalten aus Content-Management-Systemen gerecht werden. Wann HTML 5 fertig wird und als offizieller Standard verabschiedet werden kann, ist derzeit noch völlig unklar.

Neben dem klassischen HTML begann im Jahre 2000 die Entwicklung von **Extensible Hypertext Markup Language** (XHTML). Sie basierte auf HTML 4.01 und beinhaltete fast den vollständigen Befehlssatz. Allerdings wurde hierbei ganz besonders strikt zwischen Inhalt und Formatierung unterschieden. Es waren keinerlei Formatierungsbefehle im HTML-Code mehr erlaubt. Diese wurden ausschließlich durch CSS-Dateien, Templates oder anderen Vorlagearten hinzugefügt. Damit sollte XHTML vor allem den Ansprüchen moderner Content-Management-Systeme gerecht werden, die immer mehr Einzug in das World Wide Web nehmen.

- Anfang 2000 wurde die Version 1.0 von XHTML verabschiedet, welche aus Gründen der Kompatibilität noch leichte Formatierungen im HTML-Code erlaubte.
- Mitte 2001 erschien der Standard XHTML Version 1.1 und nahm eine ganz strikte Trennung von Inhalt und Formatierung vor. Außerdem entfielen einige Befehle aus dem HTML-Standard, die als nicht mehr zeitgemäß angesehen wurden.
- Die Version XHTML 2.0 sollte nicht mehr auf HTML 4.01 basieren, sondern völlig neue und moderne Befehle beinhalten. Außerdem sollte die Trennung von Inhalt und Formatierung damit vollendet werden. Im Sommer 2009 wurde die Entwicklung von XHTML 2.0 eingestellt, weil diese Funktionen stattdessen Teil von HTML 5 werden sollen.

Auch die heute bekannten **Cascading Style Sheets** (CSS) haben sich erst im Laufe der Zeit entwickelt. Am Anfang gab es keine Versionen, und es wurde schlicht von CSS gesprochen. Erst mit der strikteren Trennung von Inhalten und Layout durch HTML war man auch bei CSS gezwungen, klare Standards zu schaffen.

- Die erste konkrete Definition von CSS 1.0 erfolgte im Jahre 1996. Ein Jahr später, also 1997, wurde sie offiziell in HTML 4.0 aufgenommen.
- Etwa zwei Jahre später, Mitte 1998, erschien die Version CSS 2.0 und erweiterte die Möglichkeiten der Formatierung und Gestaltung um viele neue Befehle. Vor allem sollten die Kompatibilitätsprobleme der verschiedenen Browser durch strikte Standards gelöst werden.
- Seit 2007 ist die Version CSS 2.1 der neue Standard und beinhaltet vor allem kleinere Änderungen, Korrekturen und Optimierungen im Befehlssatz. Au-

Berdem werden die neuen Möglichkeiten der aktuellen Browser besser berücksichtigt.

- An der Version CSS 3.0 wird bereits seit einigen Jahren gearbeitet. Sie soll viele neue Möglichkeiten bringen, modulares Arbeiten ermöglichen und zwischen Anzeigegeräten wie PC, Notebook, Tablet-PC, Handy usw. unterscheiden können. Ähnlich wie HTML 5 bildet CSS 3.0 bisher nur einen Entwurf, und wann mit einer Verabschiedung zu rechnen ist, ist völlig unklar.

Bei so vielen Entwicklungen und Standards wird es schnell verwirrend. Soll man nun HTML lernen, besser XHTML, und was ist mit CSS? Eigentlich tun HTML, XHTML und CSS alle dasselbe – sie deklarieren Textelemente und formatieren diese für die Anzeige. Deshalb stellt HTML 5 genau den richtigen Weg dar, denn es versucht, viele Entwicklungen unter einem Standard zusammenzufassen. Nur so lassen sich die Inhalte des Internets modern, dynamisch, multimedial und dauerhaft speichern.

1.2.2 Der Umstieg von HTML 4 zu HTML 5

Bei neuen Programmen und neuen Standards ist es immer ein wenig schwierig, sich mit den aktuellen Entwicklungen vertraut zu machen. Der Umstieg möchte erlernt werden und fordert oft ein wenig Arbeit. Das ist bei HTML natürlich nicht anders, trotzdem stellt der Umstieg von HTML 4 auf HTML 5 eine Besonderheit dar. Hier werden ganz wichtige Dinge am Grundprinzip einer Sprache verändert, die weltweit das Internet bestimmt. Es geht dabei nicht um ein paar Optimierungen, so wie bei den Vorgängern, sondern um grundsätzliche Dinge.

- HTML 5 bringt einen großen Schwung ganz neuer Befehle mit sich. Sie erweitern die Möglichkeiten bei der Homepage-Gestaltung teilweise erheblich.
- Viele bestehende Befehle werden ganz neu definiert. So funktioniert z. B. das Einbinden von Multimedia-Inhalten wie Video und Musik ganz anders.
- Manche Befehle sind ersatzlos gestrichen worden. Einige bisher bekannte Funktionen lassen sich einfach nicht mehr durchführen.
- Die Trennung von Inhalten und Layout erfährt mit HTML 5 einen großen Sprung. HTML selbst besitzt praktisch keine Layout-Befehle mehr, und es muss zwingend CSS zum Einsatz kommen.
- HTML, XHTML und CSS rücken immer näher zusammen. Im Grunde vereint HTML 5 alle drei Entwicklungen unter einem Dach. Man kann also nicht mehr einfach nur HTML lernen, sondern muss sich auch mit den anderen Themen beschäftigen.

Für alle HTML-Autoren und Homepage-Betreiber ergeben sich daraus viele neue Herausforderungen. Wer sich bereits mit HTML 4 auskennt und seine Seiten damit geschrieben hat, muss in vielerlei Hinsicht umlernen. Besonders beschwerlich dürfte dabei sein, dass viele altbekannte Befehle in HTML 5 gestrichen wurden. Oft wird sich die Frage stellen: »Wie mache ich das denn jetzt?« Das betrifft sehr stark, aber nicht ausschließlich, sämtliche Formatierungen, die

in HTML selbst gar nicht mehr vorkommen. Sie müssen alles in CSS auslagern und den bestehenden Code komplett aufräumen.

Möchten Sie gerade in HTML einsteigen, scheint das zunächst leichter zu sein, weil kein Umlernen notwendig ist. Trotzdem ist es schwieriger, weil Sie automatisch gezwungen sind, mindestens CSS mit zu erlernen. Auch XHTML müssen Sie von Anfang an beherrschen, obwohl es nicht mehr so heißt und in HTML 5 integriert wurde. HTML 5 ist viel umfassender und komplexer als alle Vorgänger. Sie müssen also von Anfang an mehr erlernen.

1.2.3 Die Praxistauglichkeit von HTML 5

HTML 5 wird von allen Seiten freudig erwartet – von HTML-Autoren, Inhaltsanbietern, Browser-Herstellern, Webserver-Betreibern, Designern usw. Das ist verständlich, denn die vielen neuen Funktionen bieten tolle Möglichkeiten und werden der rasanten Entwicklung im Internet gerecht. Trotzdem steht die Praxistauglichkeit von HTML 5 derzeit noch infrage.

- HTML 5 wird bereits seit vielen Jahren entwickelt, und es ist überhaupt kein Ende abzusehen. Es wird lediglich von einem Arbeitsentwurf gesprochen. An eine Verabschiedung des Standards ist noch überhaupt nicht zu denken.
- Die Befehle und Funktionen haben sich im Laufe der Entwicklung oft grundlegend geändert. Was gestern noch Teil von HTML 5 war, kann morgen schon wieder gestrichen werden.
- HTML 5 enthält viele Ideen und Pläne für das zukünftige Internet. Ob sich diese Wunschfunktionen tatsächlich umsetzen lassen, weiß derzeit noch niemand.
- Das W3C-Komitee selbst gibt das Jahr 2022 für die Fertigstellung an. Das ist natürlich nicht ernst gemeint, zeigt aber, dass derzeit niemand konkret absehen kann, wohin die Reise gehen wird.

Vom heutigen Zeitpunkt aus betrachtet macht es keinen Sinn, alle Möglichkeiten von HTML 5 zu erlernen und in die eigene Homepage einzubauen. Vielleicht ist dieses Wissen morgen schon wieder überflüssig. Vor allem muss auch jemand diese Befehle, die noch keinerlei Standard entsprechen, verarbeiten können. Was nützt schließlich eine Homepage in einer Formatierung, die niemand verstehen oder anzeigen kann?

Die HTML 5-Unterstützung ist bei den aktuellen Browsern nur zum Teil gegeben. Mozilla Firefox, Google Chrome, Opera und Apple Safari bemühen sich bereits um HTML 5 und unterstützen überraschend viele Funktionen. Ändert sich der Arbeitsentwurf mal wieder, ist jeweils ein schnelles Update notwendig. Der Internet Explorer kämpft immer noch mit HTML 4 und CSS 2 und interpretiert beides nicht zu 100% korrekt. Erst mit der Version 9 wagt man sich an HTML 5 heran.

Sehr schnell stellt sich die Frage, was einem HTML 5 im Moment eigentlich nützt. Welche Befehle darf man schon verwenden, und welche vermeidet man besser? Sie sollten in jedem Fall bereits HTML 5 erlernen, denn so bleibt Ihnen

der mühsame Umstieg von der Version 4 erspart, welcher früher oder später ganz sicher kommen wird. Außerdem bietet HTML 5 schon heute viele tolle Funktionen, die sicherlich bestehen bleiben und einen echten Mehrwert im Internet bieten. Davon abgesehen versucht HTML 5 durchaus abwärtskompatibel zu sein, und alle aktuellen Browser werden noch über eine sehr lange Zeit sämtliche alle Standards von HTML 4 unterstützen.

Das Ziel dieses Buchs ist es, Ihnen praxistaugliches HTML beizubringen. Sie sollen also sofort funktionierende Webseiten erstellen können. Grundsätzlich erläutern die Kapitel immer die neuesten Standards von HTML 5 und zeigen Ihnen, wie diese richtig eingesetzt werden. Manche Funktionen sind in HTML 5 aber noch nicht ausreichend implementiert. Damit die Webseite funktioniert, müssen Sie also auf HTML 4 zurückgreifen. Das wird dann deutlich angemerkt und hervorgehoben. Außerdem schreibt nicht jeder seine HTML-Dokumente bereits in HTML 5. Sie werden bei den Arbeiten anderer HTML-Autoren also meist noch auf HTML 4 treffen. Damit Sie diese Eigenheiten erkennen und richtig damit umgehen können, finden Sie immer wieder Verweise auf bestimmte HTML 4-Elemente.

1.3 Was Sie über das Internet wissen sollten

Wissen Sie eigentlich genau, wie das Internet funktioniert und mit welcher Technik Ihre Homepage anderen Besuchern zur Verfügung gestellt wird? Wenn Sie nicht gerade Unix-Spezialist oder Techniker in einem Rechenzentrum sind, mussten Sie sich bisher nicht mit diesen Themen auseinandersetzen. Das ist auch völlig in Ordnung, denn die Details der Technik bringen dem reinen Anwender wenig Nutzen. Aber ein gewisses Grundverständnis zum Internet sollten Sie schon mitbringen. Das ist nicht nur für den täglichen Gebrauch des Internets hilfreich, sondern beim Erstellen einer eigenen Homepage durchaus wichtig. Immerhin sollten Sie wissen, was da am anderen Ende der Welt mit den von Ihnen veröffentlichten Daten gemacht wird. Sind Ihnen die Zusammenhänge klar, werden Sie Ihre Homepage viel gezielter und praxisorientierter erstellen.

1.3.1 WWW – das World Wide Web

Der Begriff **World Wide Web** (WWW) wird heute meist als Synonym für das Internet selbst verwendet. Umgangssprachlich mag das in Ordnung sein, da sich der größte Teil der Internetinhalte inzwischen tatsächlich ins WWW verlagert hat. Technisch und historisch gesehen ist das aber falsch.

Dass das Internet ein weltweiter Verbund von Computern ist, weiß heute jeder, der sich auch nur annähernd mit dem Thema beschäftigt. Allerdings beschreibt dies nur die rein technische Vernetzung der Computer bzw. der Unternetze zueinander. Mit den Inhalten, wie diese gespeichert und transportiert werden, hat das zunächst noch gar nichts zu tun. Im Internet gibt es viele verschiedene Dienste, die unterschiedliche Arten von Informationen transportieren.

Hierzu gibt es sogenannte Protokolle. Diese legen fest, um welche Art von Informationen es sich handelt und wie diese übertragen werden. Ein typisches Beispiel stellt die Datenübertragung per **File Transfer Protocol** (FTP) dar. Damit lassen sich größere Datenmengen von einem Computer auf den anderen transferieren – Sie kennen das als Upload und Download. Auf so einem Protokoll basiert auch der weltweite E-Mail-Dienst. Durch das Protokoll **Simple Mail Transfer Protocol** (SMTP) ist genau festgelegt, wie eine E-Mail-Adresse, ein Server oder die Nachricht selbst auszusehen haben. Nur so funktioniert der Austausch weltweit. Letztendlich besteht das Internet also aus ganz vielen Diensten, die jeweils auf einem eigenen Protokoll basieren. Weitere Beispiele sind der Chat per **IRC**, die Kommandozeile per **Telnet**, die Nachrichten per **NTTP**, das Abrufen von Post per **POP3** und viele andere.

Abb. 1.5:
Mit dem WWW
Informationen
weltweit
abrufen



Beim World Wide Web (WWW) ist das nicht anders. Es basiert auf dem Protokoll **Hypertext Transfer Protocol** (HTTP) und dient in erster Linie dem Austausch von Dokumenten, Texten und ähnlichen Inhalten im Internet. Das Web, wie wir es heute kennen, wurde 1989 von Tim Berners-Lee am CERN (Europäische Organisation für Kernforschung) in Genf in der Schweiz begründet. Das Ziel der Entwicklung bestand darin, die vielen Forschungsergebnisse auf möglichst einfache Art und Weise mit den Kollegen austauschen zu können. Dabei sollten die Textinhalte in einer genormten Form strukturiert und abgespeichert werden, sodass sie von allen Forschern abgerufen werden konnten. Kompatibilitätsprobleme wie bei herkömmlichen Dateien aus Textverarbeitungen sollte es genauso wenig geben wie die Notwendigkeit, die Texte als Download auf den eigenen Computer übertragen zu müssen. Sie sollten online abrufbar sein. Ganz wichtig war auch die Idee, im Text verwendete Zitate, Verweise, Quellen oder auch weiterführende Informationen direkt verknüpfen zu können.

Insgesamt basiert das World Wide Web auf ein paar sehr wichtigen Standards:

- **HTTP** stellt das Protokoll für die Übertragung der Inhalte im Netzwerk dar.
- **HTML** ist die Sprache, in der die Dokumente geschrieben und organisiert werden.
- **URLs** bieten die Möglichkeit, Inhalte miteinander zu verknüpfen und somit von einem Dokument zum nächsten zu springen.
- **Webserver** sind öffentliche Computer, auf denen die HTML-Dokumente gespeichert sind und diese im Internet für die Besucher bereitgestellt werden.
- **Browser** sind die Anzeigeprogramme auf den Computern der Anwender, mit denen die HTML-Dokumente von den Servern abgerufen und gelesen werden.

Am Anfang bestand das World Wide Web fast ausschließlich aus reinen Textinhalten. Doch es hat sich sehr schnell weiterentwickelt und erlaubte schließlich auch Bilder und Grafiken in den Dokumenten. Inzwischen sind auch Audio-In-

halte, Animationen, Videos und interaktive Inhalte wie Datenbanken, Foren und soziale Netzwerke eine Selbstverständlichkeit. Besonders die Verknüpfung der Inhalte miteinander hat zu einer völlig neuen Art des Lesens und Recherchierens geführt. Heute liest man nicht einfach nur noch einen Text, sondern durch die Verknüpfungen gleich ein ganzes Dutzend nacheinander, nebeneinander oder im Wechsel. Das führt dazu, dass man beim Recherchieren im Web sehr schnell von einem Dokument zum nächsten springt und sich innerhalb kürzester Zeit in völlig anderen Inhalten am anderen Ende der Welt befindet. Dieses Phänomen hat den Begriff des **Surfens** geprägt. Wer im Internet surft, lässt sich von einem Link zum nächsten treiben und kommt so in den Genuss ganzer Wellen von Informationen.

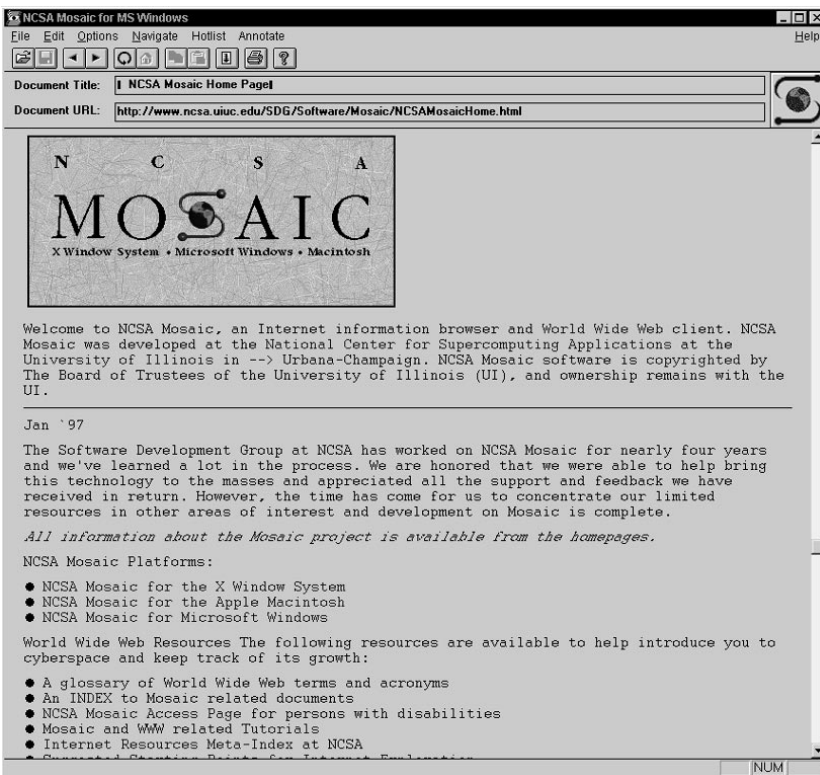


Abb. 1.6:
Der Mosaic-Browser machte das WWW für alle verfügbar (Abbildung: NCSA/University of Illinois).

Inzwischen ist das World Wide Web zur multimedialen Arena des Internets aufgestiegen und hat die meisten anderen Protokolle fast vollständig verdrängt. Kaum jemand kennt noch die Vorgänger wie Gopher, Finger & Co. Vielen Anwendern ist diese Entwicklung nicht bekannt, sodass das World Wide Web heute sehr oft als das Internet selbst angesehen wird.

Abb. 1.7:
Das W3C legt
die Standards
für das Web fest
(Abbildung:
W3C).



Damit dieses komplexe Gefüge aus Webservern, HTML-Dokumenten, Links und Browsern funktionieren kann, muss es natürlich Standards geben. Hierzu wurde das **World Wide Web Consortium (W3C)** im Jahre 1994 gegründet. Der Vorsitzende ist Tim Berners-Lee. Allerdings ist das W3C keine zwischenstaatlich anerkannte Organisation und hat sich im Grunde selbst gegründet. Seine Vorgaben sind weder zwingend, noch stellen sie eine Norm im Sinne der ISO-Spezifizierung dar (Internationale Organisation für Normung). Vielmehr wird dabei von **Empfehlungen** gesprochen. Trotzdem wird das W3C weltweit anerkannt, und fast alle Entwickler sind bemüht, die Vorgaben einzuhalten. Nur so ist gewährleistet, dass ihre Inhalte in aller Welt mit allen Browsern abgerufen werden können. Wer sich nicht an die Standards hält, läuft letztendlich Gefahr, dass seine Inhalte im Internet nicht gelesen werden können.

1.3.2 Der Browser als Benutzeroberfläche

HTML stellte eine völlig neue Art von Dokumententyp dar. Die Inhalte wurden mit speziellen Befehlen deklariert und strukturiert. Kein normales Programm konnte sie auswerten und anzeigen. Deswegen wurde von Tim Berners-Lee zusammen mit der HTML-Syntax auch gleich ein neues Anzeigeprogramm entwickelt. Der Webbrowser, meist einfach nur als Browser bezeichnet, stellte am Anfang ein reines Anzeigeprogramm für in HTML geschriebene Texte dar. Mit ihm wurden die Dateien per HTTP (Hypertext Transfer Protokoll) über das Internet abgerufen. Der Browser wertete die Befehle in den Dokumenten aus und generierte daraus eine Bildschirmseite des Textes. Der Name Browser kommt übrigens aus dem Englischen und bedeutet so viel wie »stöbern«, »sich umsehen« oder auch »schmökern«.

Da HTML am Anfang lediglich zum Strukturieren der Inhalte gedacht war, stellte der erste Browser mit dem Namen **WorldWideWeb** bzw. später **Nexus** einen reinen Textbrowser ohne grafische Fähigkeiten dar. Er wirkte eher wie ein puristisches Bibliotheksprogramm, zum Auswerten und Anzeigen großer Textbestände. Mit der Verbreitung und Weiterentwicklung von HTML wuchs auch der Webbrowser und wurde mit immer neuen Funktionen ausgestattet. Durch den Erfolg des World Wide Webs entwickelten auch andere Unternehmen Webbrowser. Zu den Urahnen aller Webbrowser zählt NCSA Mosaic, der zuerst eine grafische Benutzeroberfläche besaß, Seiten abspeichern konnte (Lesezeichen) und versuchte, das Lesen der Seiten über eine Formatierung angenehmer zu machen.

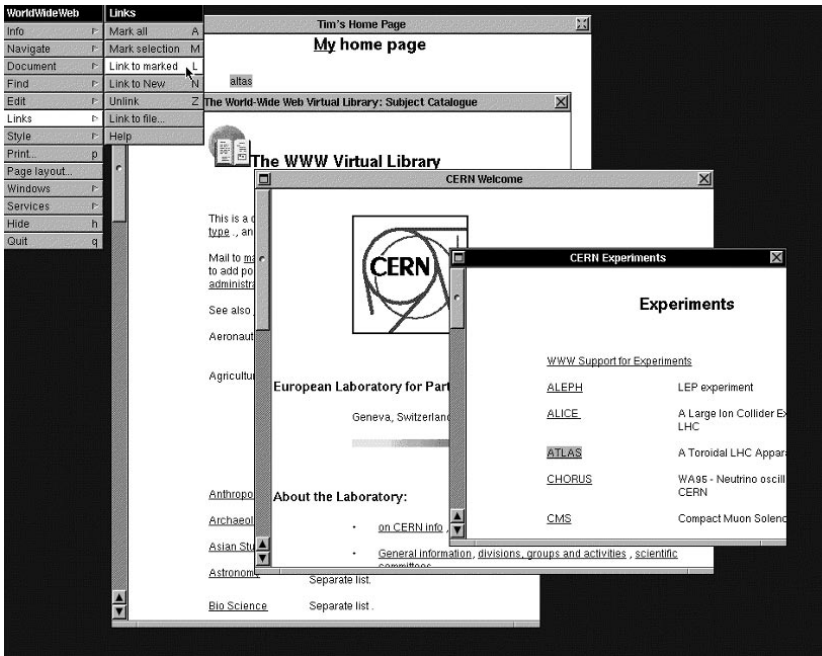
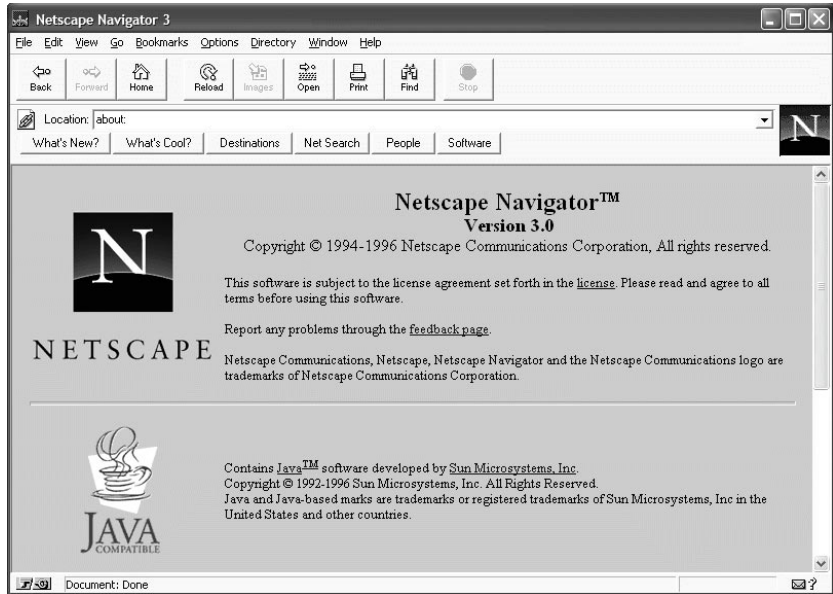


Abb. 1.8:
Nexus – der
erste Browser
für das World
Wide Web

Der große Boom des Internets stellte sich ein, als Marc Andreessen, vorher Entwicklungschef für den Mosaic-Browser, sich mit der Netscape Communication Corporation selbstständig machte. Er entwickelte 1994 den Netscape Navigator und läutete damit ein neues Zeitalter für das World Wide Web ein. Netscape prägte das WWW in großem Maße und definierte die Anforderungen an einen modernen Browser. Netscape war der erste Browser, der weitestgehend so aussah und funktionierte, wie wir das heute kennen. Im Grunde basieren alle modernen Browser von der Art und Handhabung her auf dem Netscape Navigator, weshalb er sehr gern als der Urvater aller Browser angesehen wird.

Netscape dominierte lange das Internet, und es gab keinerlei ernst zu nehmende Konkurrenz. Erst als die Firma Microsoft im Jahre 1995 bemerkte, dass sie die vielleicht wichtigste Entwicklung der Gegenwart verschlafen hatte, veränderte sich das Bild. Microsoft entwickelte den Internet Explorer und verteilte ihn kostenlos mit seinem Windows-Betriebssystem. Es dauerte ein paar Jahre, aber schließlich überwand Microsoft das Netscape-Monopol und wurde zu einer echten Konkurrenz.

Abb. 1.9:
Der Netscape Navigator 3 brachte völlig neue Möglichkeiten beim Browsen mit sich und beeinflusste das ganze WWW.



Damit brach der sogenannte **Browserkrieg** aus. Microsoft und Netscape kämpften mit harten Bandagen um die Marktführerschaft im Internet. Das tat dem World Wide Web teilweise gut, richtete aber auch riesigen Schaden an. Microsoft und Netscape überboten sich laufend mit neuen Versionen ihrer Browser, erfanden immer mehr neue Funktionen und trieben so die Entwicklung von HTML, CSS und des Internets allgemein mit riesigen Schritten voran. Im Gegenzug wurde auf Standards keinerlei Rücksicht genommen. Jeder Browser brachte eigene Funktionen und Eigenschaften mit, welche die HTML-Dokumente anders verarbeiteten. Das führte dazu, dass jedes Dokument in jedem Browser völlig anders aussah. Viele Befehle wurden vom jeweiligen Konkurrenzprodukt auch gar nicht verstanden. Im Internet machte sich ein echtes Chaos breit, und vor dem Besuch einer Webseite musste man überprüfen, für welchen Browser diese geschrieben wurde. Am Ende konnten bestimmte Seiten und Dienste gar nicht mehr genutzt werden, wenn man den falschen Browser verwendete.

Letztendlich gewann Microsoft den Browserkrieg. Das lag nicht unbedingt an der Qualität des Produkts, sondern an den schier unerschöpflichen Finanzmitteln und einem Stab von über 1.000 Entwicklern für die Internetabteilung. Das war mehr, als die Netscape Communication Corporation insgesamt besaß. Netscape verblasste immer schneller auf der Internet-Bühne. Im Jahre 1998 gründete Marc Andreessen die Mozilla-Stiftung und schenkte ihr neben diverser Finanzmittel den Source-Code des Netscape-Browsers. Schließlich wurde die Netscape Communication Corporation von AOL aufgekauft und bis auf den Namen völlig lahmgelegt. Das war das Ende von Netscape. Viele Jahre hatte nun Microsoft die Monopolstellung im Internet eingenommen. Und mangels Konkur-

renz wurde die Entwicklung des Internet Explorers weitestgehend eingestellt. So dümpelte ein völlig veraltetes Produkt jahrelang weltweit auf den Computern herum.

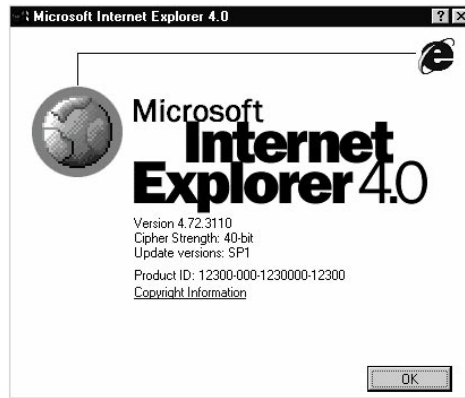


Abb. 1.10: Mit dem Internet Explorer 4 drängte Microsoft massiv auf den Browser-Markt.

Im Jahre 2004 brachte die unabhängige und durchaus finanzstarke Mozilla-Stiftung einen neuen Browser heraus: Firefox war geboren. Dieser neue Browser war klein, schlank, schnell und vor allem modern. Im Gegensatz zum inzwischen hoffnungslos veralteten Internet Explorer entsprach Firefox den modernen Bedürfnissen des Internets und der Anwender. Multimedia, Plug-Ins und Interaktivität lauteten die Schlagwörter. Dazu kam, dass sich Firefox vollständig an die Standards von HTML, CSS und XHTML hielt. So sah sich Microsoft erneut herausgefordert und kurbelte die Entwicklung des Internet Explorers wieder an. Ende 2007 kam schließlich der Internet Explorer 7 auf den Markt und versuchte jahrelange Versäumnisse nachzuholen. Ein neuer Browserkrieg entstand zum Glück nicht, weil schnell klar war, dass das Mittel zum Überleben im Einhalten der Standards lag. Firefox und der Internet Explorer führen ein halbwegs friedliches, gemeinsames Dasein im Internet.

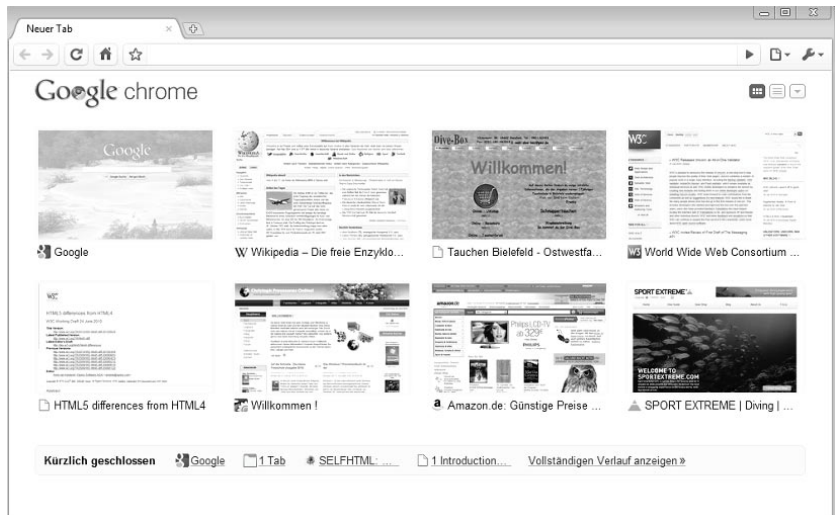
Mit dem Internet Explorer 8 holte Microsoft viele der Versäumnisse gegenüber Firefox 3.6 auf. Wirklich spannend wird es ab Firefox 4 und Internet Explorer 9, weil beide Browser dann voll auf HTML 5 setzen und dafür vollmundig versprechen, 100 % standardkonform zu sein.

Natürlich gibt es noch andere Browser wie z. B. Google Chrome, Apple Safari, Opera oder Konqueror. Einige führen eher ein Nischendasein, andere fangen an, immer beliebter zu werden, z. B. Google Chrome. Da inzwischen alle Browser bemüht sind, sich an die W3C-Standards zu halten, sind die Unterschiede in der Anzeige und Handhabung der Internetseiten gering. Vielmehr sind es die Sonderfunktionen wie Tabbed Browsing, Sicherheitsoptionen, Kennwortspeicherung, die Lesezeichenverwaltung und vieles mehr, die den jeweiligen Browser kennzeichnen. Es ist also in weiten Teilen eine Frage des Geschmacks, ob Sie heute den Internet Explorer, Mozilla Firefox, Google Chrome oder einen ganz anderen Browser verwenden.

Abb. 1.11:
Microsoft Internet Explorer 8
& Mozilla Firefox 3.6



Abb. 1.12:
Googles eigener Browser
überrascht mit
guten Leistungen
und gehört
neben Internet
Explorer und
Firefox zu den
drei »Großen«.



1.3.3 Was Sie zu Webservern wissen sollten

Damit Ihre Webseite mit einem Browser weltweit abgerufen werden kann, muss sie natürlich im Internet veröffentlicht werden. Hierzu dienen Webserver. Dabei handelt es sich um recht normale Computer, die in der Regel in einem Rechenzentrum stehen. In einem persönlichen Verzeichnis liegen Ihre HTML-Doku-

mente und warten darauf, von einem Webbrowser angefordert zu werden. Passt dies, schickt der Webserver die HTML-Dateien über das Protokoll HTTP über das Internet zum Computer des Anwenders. Dort wertet der Browser die Befehle aus und zeigt die HTML-Seite an. Das Prinzip eines Webserver ist immer gleich. Dabei spielt es keine Rolle, ob Sie eine kostenlose Homepage bei Ihrem Internetprovider nutzen, den privaten Webspaces des E-Mail-Kontos verwenden oder ein eigenes Konto bei einem der großen Hosting-Provider gebucht haben. Letztendlich tun sie alle das Gleiche, nämlich das Verschicken von HTML-Dokumenten über das Protokoll HTTP.

Es gibt sehr viele Programme, die die Aufgabe eines Webserver übernehmen können. Wie in fast allen Bereichen haben sich aber auch hier ein paar wenige Programme weitestgehend durchgesetzt. Als Quasistandard gilt im Internet der **Apache Webserver**. Dabei handelt es sich um eine freie Open-Source-Software. Sie darf also von jedem, privat oder kommerziell, kostenlos verwendet werden. Der Apache Webserver kommt in den meisten Rechenzentren weltweit zum Einsatz. Wenn Sie also bei einem Hosting-Provider ein Konto für Ihre Homepage buchen, stehen die Chancen sehr gut, dass der Server auf einem Unix-Betriebssystem mit Apache läuft (offizielle Homepage – <http://httpd.apache.org>).

Des Weiteren genießt das Paket **Microsoft Internet Information Services (IIS)** eine recht große Verbreitung. Dieser Webserver läuft ausschließlich auf Windows-Systemen und unterstützt zusätzliche Microsoft Webtechnologien wie Active Server Pages (ASP) oder .NET-Applikationen (offizielle Homepage – <http://www.iis.net>).

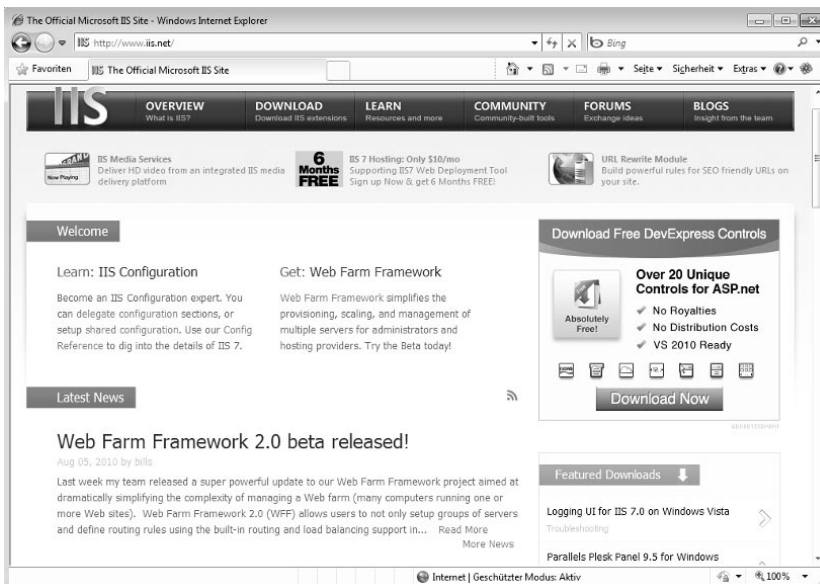
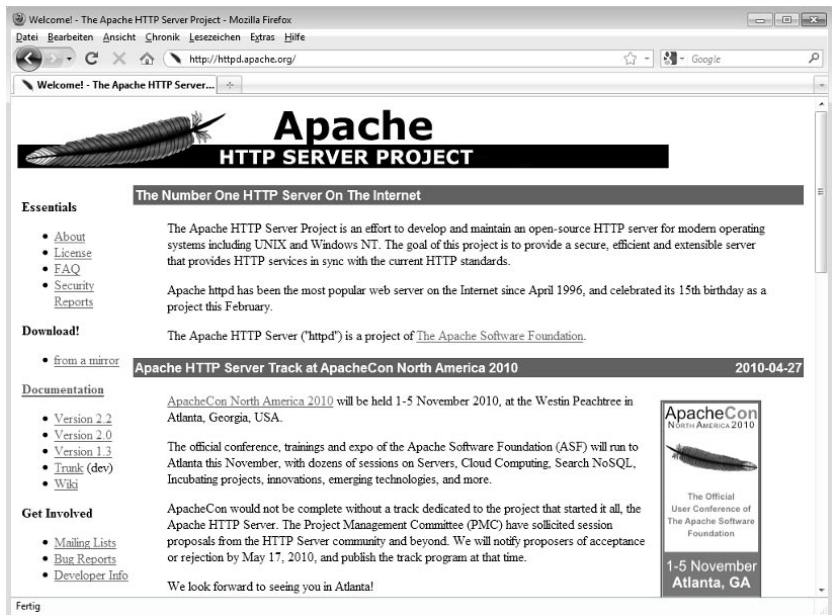


Abb. 1.13: Microsoft Internet Information Services läuft nur unter Windows und ist kostenpflichtig. Einige Versionen von Windows Vista bieten eine kostenfreie Minimal-Version.

Die Besonderheit von Apache besteht darin, dass er nicht einfach nur HTML-Dateien durch das Internet schickt, sondern die Möglichkeiten durch die Integration von Skriptsprachen wie Perl oder PHP enorm erweitert. Auf dem Server laufen also richtige Programme, die entweder Eingaben durch die Besucher verarbeiten oder auch die eigene Webseite in Echtzeit aus verschiedenen Bausteinen zusammenstellen. Besonders wichtig ist auch die nahtlose Einbindung von SQL-Datenbanken und anderer externer Inhalte. Im einfachsten Fall lassen sich damit z. B. Gästebücher erstellen, im größten Fall sind dies ganze Online-Shops, Netzwerkverwaltungssysteme oder Webportale mit einem professionellen Content-Management-System. Dazu kommen noch viele weitere Funktionen wie Live-Streams für Audio- und Video, Sicherheitsverschlüsselungen, Benutzeranmeldungen und vieles mehr.

Abb. 1.14:
Apache ist einer der meistgenutzten und am weitesten verbreiteten Webserver.



Wirklich interessant wird es, sobald man die Art und Weise betrachtet, mit der die Benutzerkonten unter Apache verarbeitet werden können. In der Regel besitzen Sie bei einem Homepage-Anbieter ein Benutzerkonto mit dazugehörigem Namen. Meist gelangen Sie dann über die Adresse Ihres Anbieters plus Ihren Benutzernamen an Ihre Webseite.

<http://www.internet-anbieter.de/benutzerkonto/>

Natürlich werden auf diesem Server auch Hunderte oder gar Tausende andere Kunden ihr Benutzerkonto liegen haben. Das ist für Apache überhaupt kein Problem, und jedes Konto kann völlig einzeln verwaltet werden. Möchten Sie Ihre Seite anstatt über die Adresse des Anbieters mit einer eigenen Domäne erreichen, ist auch das möglich. Man spricht dabei von virtuellen Servern. Sobald je-

mand z. B. die Domäne `http://www.meine-homepage.de` aufruft, leitet Apache diese Anfrage an Ihr Benutzerverzeichnis auf der Festplatte des Servers weiter. Das hat eigentlich die Adresse `http://www.internet-anbieter.de/benutzerkonto/`, aber das versteckt Apache vollkommen. Niemand bekommt etwas von diesem virtuellen Server mit. Für die Besucher sieht es so aus, als hätten Sie einen richtigen eigenen Server im Internet. Dies ist das gängige Prinzip bei allen Hosting-Providern. Das sieht nicht nur gut aus, sondern bietet Ihnen noch eine ganze Reihe weiterer Möglichkeiten, z. B. eine eigene E-Mail-Adresse, Sicherheitszertifikate, eine einfachere Verlinkung durch den Wegfall der Provider-Adresse und vieles mehr.

In den meisten Fällen müssen Sie sich um den Webserver und die Technik dahinter keine Gedanken machen. Das regelt alles Ihr Homepage-Anbieter für Sie. Sobald Sie sich aber mit größeren Homepage-Projekten beschäftigen oder ein eigenes Content-Management-System installieren wollen, wird das interessant. Dann muss die Software zum Webserver passen und alle notwendigen Funktionen unterstützen. Zum Beispiel muss die Skriptsprache PHP verstanden und verarbeitet werden können, weil Programme wie Joomla! oder WordPress damit programmiert sind. Außerdem ist eine Unterstützung der Datenbank MySQL notwendig, weil hier die Inhalte des CMS gespeichert werden. Damit müssen Sie sich zu diesem Zeitpunkt erst einmal nicht beschäftigen. Zum einen wollen Sie erst einmal die Grundlagen von HTML erlernen, zum anderen gehört das eher zum Themenbereich »Webseiten installieren und verwalten«. Es ist aber auf jeden Fall hilfreich, wenn Sie davon schon einmal gehört haben und ungefähr wissen, wozu ein Webserver eigentlich gut ist.

Und nun kann es direkt mit HTML losgehen!



*Abb. 1.15:
Hier geht's
zum HTML-
Grundkurs!*

Stichwortverzeichnis

A

Absätze 60
abschicken, Formulare 233
Abstand
– außen (CSS) 337
– innen (CSS) 338
– Objekte (CSS) 340
– Padding (CSS) 339
abweisen, Suchmaschinen 302
Accept-Charset 237
ActiveX einbinden 223
Adobe Flash einbinden 216
alternativer Text, Bilder 139
Andreessen, Marc 29
Anführungszeichen 41
Anker 112
Apache Webserver 33
Applets, Java 220
Archiv, Suchmaschinen 303
ASCII-Text 37
ASCII-Zeichensatz 41
Audio einbinden 209
Aufbau 46
Auflistungen (CSS) 353
aufräumen, Code 17
Auktionen 16
Ausrichtung
– Objekte (CSS) 340
– Schriften (CSS) 333

äußerer Abstand (CSS) 337
Auswahllisten, Formular 275
Autocomplete 238
Autoplay, Videos 199
Autor 297

B

Basis-Link 120
Befehle öffnen und schließen 39
Bereiche, Tabellen 163
Berners-Lee, Tim 28
Beschreibung 297
Bilder
– Breite und Höhe 143
– Dateigröße 136
– dpi 142
– einbinden 137
– Formate 134
– Formulare 243
– Größe 142
– Hintergrund (CSS) 349
– Image Maps 148
– Links 145
– Text 139
– Titel 140
– Verzeichnis 137
blinde Tabellen 178
Blockzitate 65
Body 45

Breite und Höhe (CSS) 341
Browser allgemein 28
Browser-Cache 306
Browser-Fenster, Link 116
Browserkrieg 30
Buttons, Formular 270

C

Cache, Browser 306
Cache, Suchmaschinen 303
Cascading Style Sheets
– Ausrichtung Text 333
– Außenabstand 337
– Body-Bereich 329
– Breite und Höhe 341
– Class 367
– Div 366
– Fettdruck 330
– Float 345
– Funktionsweise 318
– Hintergrund 20
– Hintergrundbilder 349
– Hintergrundfarbe 348
– ID 367
– Innenabstand 338
– Kapitälchen 330
– Kombinationen 327
– Kursiv-Schrift 330
– Links 361
– Listen 353
– Mauszeiger 364
– maximale Breite 343
– maximale Höhe 343
– MouseOver 363
– Objekte ausrichten 340
– Padding 339
– Rahmen 351
– Schriften 330
– Schriftfamilien 332
– Schriftfarbe 331
– Schriftgröße 330
– Span 366
– Style 319
– Syntax 320
– Tabellen 356
– tiefstellen 331
– übergroße Objekte 343
– verlinken 326
– zentrale Datei 325
– Z-Index 344

CGI 289
Checkboxes, Formular 273
Chrome, Google 31
Class 367
Code aufräumen 17
Code, Sonderzeichen 71
Common Gateway Interface 289
Content-Management-System 16
Controls 198
Cross-Site-Scripting 196
CSS
– Ausrichtung Text 333
– Außenabstand 337
– Body-Bereich 329
– Breite und Höhe 341
– Class 367
– Div 366
– Fettdruck 330
– Float 345
– Funktionsweise 318
– Hintergrund 20
– Hintergrundbilder 349
– Hintergrundfarbe 348
– hochstellen 331
– ID 367
– Innenabstand 338
– Kapitälchen 330
– Kombinationen 327
– Kursiv-Schrift 330
– Links 361
– Listen 353
– Mauszeiger 364
– maximale Breite 343
– maximale Höhe 343
– MouseOver 363
– Objekte ausrichten 340
– Padding 339
– Rahmen 351
– Schriften 330
– Schriftfamilien 332
– Schriftfarbe 331
– Schriftgröße 330
– Span 366
– Style 319
– Syntax 320
– Tabellen 356
– übergroße Objekte 343
– verlinken 326
– zentrale Datei 325
– Z-Index 344
Cursor (CSS) 364

D

Dateien herunterladen 125
Dateien, Formular 267
Dateiendung 37
Datum 299
Definitionslisten 97
DELETE 236
deutsche Umlaute 72
Div 366
Dokumenteninformationen 297
Dokumententitel 45
Dokumententyp 42
DPI, Bilder 142
Dreamweaver 17

E

Editoren, HTML 17
Eingabefeld
– allgemein 232
– Auswahlliste 275
– Buttons 270
– Checkboxes 273
– Dateien 267
– erstellen 247
– Gruppen 244
– Gruppieren 280
– Kennwörter 266
– Text 248
– Textbereiche 253
– vorbelegen 251
– Wertebereich 264
einladen, Suchmaschinen 302
E-Mail-Links 127
Embed 188
Enctype 236
Endung, Datei 37
Erstellungsdatum 299
externe Links 106

F

Farbe
– Erläuterung 323
– Hintergrund (CSS) 329, 348
– Schrift 322
– Schriften (CSS) 331
Fett-Druck (CSS) 330
Firefox 31
Flash einbinden 216
Float (CSS) 345
Foren 17
Format, Videos 200

Formatierung, Schriften (CSS) 330

Formulare
– Accept-Charset 237
– Auswahllisten 275
– Autocomplete 238
– Buttons 270
– CGI 289
– Checkboxes 273
– Dateien 267
– DELETE 236
– Eingabefelder 247
– Enctype 236
– eröffnen 234
– Felder 232
– GET 235
– Gruppen 244
– Gruppieren 280
– Hintergrund 232
– Kennwörter 266
– Label 240
– Novalidate 238
– Perl 290
– PHP 291
– POST 236
– PUT 236
– Schaltflächen 242
– Textbereiche 253
– Textfelder 248
– unsichtbare Felder 267
– verschicken 233
– vorbelegen 251
– Wertebereich 264
Fotos
– Breite und Höhe 143
– Dateigröße 136
– dpi 142
– einbinden 137
– Formate 134
– Formulare 243
– Größe 142
– Hintergrund (CSS) 349
– Links 145
– Text 139
– Titel 140
– Verzeichnis 137
Frontpage (MS) 17
Funktionen, Links 123

G

Gästebücher 17
GET 235

- GIF-Format 134
- Gliederung, Tabellen 163
- Grafiken
 - Breite und Höhe 143
 - Dateigröße 136
 - dpi 142
 - einbinden 137
 - Formate 134
 - Formulare 243
 - Größe 142
 - Hintergrund (CSS) 349
 - Links 145
 - Text 139
 - Titel 140
 - Verzeichnis 137
- grafische Links 145
- Größe
 - Bilder 142
 - Multimedia 183
 - Objekte (CSS) 341
 - Schriften (CSS) 330
- Großschreibung 39
- Gruppen, Formular 244, 280

- H**
- Header 43
- herunterladen, Dateien 125
- Hintergrundbilder (CSS) 349
- Hintergrundfarbe (CSS) 329, 348
- hochstellen 85
- Höhe und Breite (CSS) 341
- Hyperlink
 - allgemein 104
 - Bilder 145
 - Darstellung (CSS) 361
 - E-Mail 127
 - extern 106
 - Formatierung (CSS) 361
 - Funktionen 123
 - intern 109
 - MouseOver (CSS) 363
 - Syntax 104
 - Textstellen 112
 - Ziel 115
- Hypertext Markup Language 19

- I**
- ID 367
- IFrame-Injektion 196
- IFrames, allgemein 190
- Image Maps 148

- Index 297
- Informationen, Seite 295
- Inhaltsbereich 45
- Injektion, Iframe 196
- innerer Abstand (CSS) 338
- interne Links 109
- Internet Explorer 29
- ISO-8859-1 44

- J**
- Java einbinden 220
- Joomla! 18
- JPEG-Format 134

- K**
- Kapitalchen (CSS) 330
- Kennwörter, Formular 266
- Kleinschreibung 39
- Kombinationen, CSS 327
- Kommentare 70
- Kompozer 17
- Kopfbereich 43
- Kursiv-Schrift (CSS) 330

- L**
- Label 240
- Landkarten, Image Maps 148
- Layout mit Tabellen 178
- Leerzeichen 40
- Leserichtung 90
- Linien 70
- Link
 - allgemein 104
 - Basis 120
 - Bilder 145
 - Darstellung (CSS) 361
 - E-Mail 127
 - extern 106
 - Formatierung (CSS) 361
 - Funktionen 123
 - intern 109
 - MouseOver (CSS) 363
 - Syntax 104
 - Textstellen 112
 - Typen 121
 - Ziel 115
- Link-Basis 120
- Listen
 - allgemein 93
 - mit CSS 353
 - nummeriert 95

- sortiert 95
 - unsortiert 94
- Listenelement 94
Loop 199

M

- Mail-Links 127
Mauszeiger (CSS) 364
maximale Breite (CSS) 343
maximale Höhe (CSS) 343
Meta-Informationen 295
Microsoft Expression Web 17
Microsoft Internet Information Services 33
Mindestbreite (CSS) 343
Mindesthöhe (CSS) 343
MouseOver (CSS) 363
Mozilla Firefox 31
Multimedia
– ActiveX 223
– Audio 209
– einbinden 182
– Flash 216
– Größe 183
– Java 220
– Musik 209
– Parameter 186
Musik einbinden 209

N

- Netscape Navigator 29
neues Browser-Fenster 116
Nexus 28
Notizen 70
Novalidate 238
nummerierte Listen 95

O

- Object 182
Objekte ausrichten (CSS) 340
öffnen & schließen (Tags) 39
OL, sortierte Listen 95

P

- Padding (CSS) 339
Param, Object 186
Parameter, Multimedia 186
Perl 290
PHP 291
PNG-Format 134
POST 236

- Poster 198
präformatierter Text 68
Preload 199
Protokolle 25
Proxies 305
PUT 236

R

- Rahmen
– grafische Links 145
– mit CSS 351
– Tabellen 156
Range 264
Relationship 121

S

- Sandbox 194
Schaltflächen, Formular 242
Schriften
– Ausrichtung (CSS) 333
– CSS 330
– Farbe 322
– Farbe (CSS) 331
– Fettdruck (CSS) 330
– Größe (CSS) 330
– hochstellen (CSS) 331
– Kapitälchen (CSS) 330
– kursiv (CSS) 330
– tiefstellen (CSS) 331
Schriftfamilien 332
Schriftfarbe 322
Seamless, IFrame 191
Secure Sockets Layer 124
Seitenaufbau 42
Seitengestaltung, Tabellen 178
Seiteninformationen 295
Seitenkopf 43
sichere Links 124
Sonderzeichen 71
sortierte Listen 95
Sound einbinden 209
Source, Videos 200
Spalten, Tabellen 158
Span 366
Sprache einstellen 44
Sprungmarke 104
SRCDOC 196
Stichwörter 297
Striche 70
Style 319
Style-Datei verlinken 326

Suchmaschinen 296
Syntax, Grundprinzip 38

T

Tabellen
– allgemein 156
– Bereiche 163
– blind 178
– CSS 356
– Rahmen 156
– Spalten 158
– Spaltenüberschrift 160
– Überschrift 161
– Zeilen 157
– Zellen verbinden 167
Tabstopps 40
Tag, Bedeutung 38
Tags öffnen und schließen 39
Templates erstellen 18
Textausrichtung (CSS) 333
Textbereich 45
Textbereiche, Formular 253
Texteditor 37
Textfelder, Formular 248
Textfelder, vorbelegen 251
Textstellen, Links 112
tiefstellen 85
Titel 45
Titel, Bilder 140
Trennlinien 70

U

übergroße Objekte (CSS) 343
Überschrift, Spalten 160
Überschrift, Tabellen 161
Überschriften 47
Umlaute 72
Umrandung
– grafische Links 145
– mit CSS 351
– Tabellen 156
unsichtbare Informationen,
Formular 267
unsortierte Listen 94

Upload, Formulare 267
UTF-8 41, 44

V

Validate 238
verbinden Zellen 167
verbotene Zeichen 41
verschachteln, Listen 99
verschicken, Formulare 233
Verschlüsselung 124
Verzeichnis, Bilder 137
Videos einbinden 197
vorformatierter Text 68
Vorlagen erstellen 18

W

W3C 28
Webserver, allgemein 32
weiterleiten 306
Wertebereich, Formular 264
WordPress 18
World Wide Web Consortium 28
World Wide Web, Hintergründe 25

X

XHTML 22

Y

Y
– unter Y steht gar kein Eintrag, aber
es taucht nun im Index auf
(*_freu_*) 15

Z

Zeichensatz einstellen 44
Zeiger, Maus (CSS) 364
Zeilen, Tabellen 157
Zeilenhöhe (CSS) 334
Zeilenumbruch 61
Zeit 299
Zellen verbinden 167
zentrale Datei, CSS 325
Ziel, Links 115
Z-Index (CSS) 344
Zitate 65