

Fachhochschule Dortmund
Fachbereich Nachrichtentechnik
Studienrichtung Telekommunikationstechnik
Studiengang Informations- und Kommunikationstechnik

**Planung einer selbstverwaltenden
Ehemaligendatenbank für den internen
Gebrauch**

Ingenieurmäßiges Arbeiten

von

André Blase und Carsten Sander

16.02.2001

Labor für Kommunikationselektronik
Betreuer: Prof. Dr. – Ing. Jürgen Hetsch

Like ancient Rome, Microsoft's empire has a tax, called
Windows, that most everyone pays to be part of the empire.

- von einer unbekanntem Webseite –

Inhaltsverzeichnis

Einleitung	1
Kapitel 1	
Datenbanken	4
1.1 Datenbanksysteme	4
1.2 Datenbankserver	8
1.2.1 Das Betriebssystem	8
1.2.2 Der Webserver.....	9
1.2.3 Die Datenbank	10
1.2.4 Dynamische Webseitengenerierung	12
1.2.5 Zusätzliche Module	15
Kapitel 2	
Planung und Vergleichsaspekte	17
2.1 Planungsphase	17
2.2 Vergleichsphase	20
2.2.1 Vergleich der Betriebssysteme	20
2.2.2 Vergleich der Webserver-Software	25
2.2.3 Vergleich der Datenbank-Software	31
2.2.4 Vergleich der Skriptsprachen	33
2.2.5 Vergleich der möglichen Kombinationen	39
2.3 Hardwareanforderungen	40
2.4 Erstellung eines Konzeptpapiers	42

Kapitel 3

Implementierung	47
3.1 SuSE Linux	47
3.2 Apache Webserver	51
3.3 MySQL	55
3.4 PHP	58
3.5 Sendmail	60
3.6 Spiegelung	61

Kapitel 4

Schlußbemerkung und Ausblick	64
4.1 Erfahrungen	64
4.2 Erweiterungen	66

A Anhang	69
A.1 Fußnoten	70
A.2 Glossar	73
A.3 Abbildungsverzeichnis	90
A.4 Literaturverzeichnis	91

Einleitung

Das vorliegende Ingenieurmäßige Arbeiten stellt die Planung, Konzeption und Implementierung eines Systems zur datenbankgestützten Verwaltung eines Ehemaligenverzeichnisses vor. Der erste Abschnitt dieser Einleitung liefert hierfür die Motivationsgrundlage, indem die Problematik der bisherigen, konventionellen Vorgehensweise bei der Verwaltung der Ehemaligendaten erläutert wird. Als Möglichkeit zur Beseitigung dieser Schwierigkeiten, insbesondere der fehlenden Gesamtübersicht der Abschlußprüfungsthemen und dem raschen Veralten der Adreßdaten, wird hier der Einsatz eines sich weitgehend selbstverwaltenden Datenbanksystems betrachtet. Der inhaltliche Aufbau dieser Arbeit ist im letzten Abschnitt der Einleitung dargelegt.

Zum Abschluß einer Studienlaufbahn an der Fachhochschule Dortmund im Fachbereich Nachrichtentechnik werden von jedem Studenten im Rahmen der Prüfungsordnung zwei schriftliche Arbeiten – Ingenieurmäßiges Arbeiten und die Diplomarbeit – erstellt. Eine fächerübergreifende Verwaltung der in diesen Arbeiten abgehandelten Themen sowie der Absolventenadreßdaten gibt es bislang nicht, auch eine Aktualisierung der persönlichen Daten kann aufgrund des fehlenden Kontaktes nach Abschluß des Diploms nicht erfolgen.

Bei eventuellen Rückfragen von Mitgliedern des Fachbereichs oder externen Anfragen, zum Beispiel von Firmen, wird bisher auf das System der Mundpropaganda vertraut, welches den Anfragenden meist zu den privaten Aufzeichnungen des IA- bzw. DA-betreuenden Professors führt. Besonders nach einer Zeitspanne von mehr als einem Jahr ist die Aktualität der Adreßdaten nicht gewährleistet und somit eine gesicherte Kontaktaufnahme mit dem ehemaligen Studenten nicht mehr möglich.

Mit zunehmender Verbreitung einfach bedienbarer Informationssysteme, insbesondere des World Wide Web (WWW), liegt es nahe, eine Verwaltung und Aktualisierung der relevanten Daten über dieses Medium durchzuführen. Außerdem kann durch die Trennung von Intranet und Internet ein Vollzugriff auf die Datenbank bei internem Gebrauch und ein restriktiver Zugriff für Internetanfragen festgelegt werden, welches für die Gewährleistung des Datenschutzes von Notwendigkeit ist. Bei der Aktualisierung der persönlichen Daten durch den jeweiligen Studenten mittels WWW-Frontend sorgen Verschlüsselungsmechanismen für die nötige Sicherheit und Einhaltung des Datenschutzgesetzes.

Der Einsatz des oben beschriebenen integrierten Datenbanksystems für die Verwaltung der Adreßinformationen kann Abhilfe für die eingangs beschriebenen Schwierigkeiten bringen. Neben den offensichtlichen Vorteilen, wie der Möglichkeit zur redundanzfreien und konsistenzbewahrenden Datenhaltung, ergeben sich eine Reihe weiterer Argumente für diese Lösung. So lassen sich aufgrund der durch das Datenbanksystem erzwungenen Trennung von Struktur und Inhalt der Informationen recht einfach Ausgabemasken für die unterschiedlichen Anfragen entwickeln (Adreßdaten, IA- und DA-Themenübersicht). Des weiteren erleichtert ein Datenbanksystem auch die Implementierung von Suchmaschinen, die mit vergleichbarem Leistungsumfang in einer rein statischen WWW-Version aufgrund der fehlenden semantischen Struktur eines WWW-Dokumentes nur mit hohem Aufwand zu erreichen sind.

Das Ziel dieser Arbeit ist die Planung zur Erstellung einer weitgehend sich selbst verwaltenden Ehemaligendatenbank für den internen Gebrauch sowie deren technische Umsetzung in Form einer Referenzimplementierung. Ausgehend von den in dieser Einleitung vorgestellten Schwierigkeiten, die bei der bisherigen Verwaltung der Ehemaligendaten zu beobachten sind, werden in Kapitel 1 allgemein der Aufbau und die prinzipielle Funktionsweise von Datenbanksystemen erläutert, da die Datenbank die zentrale Rolle bei diesem Projekt darstellt. Es enthält ferner die für das in den restlichen Kapiteln vorgestellte Datenbanksystem notwendigen datenbanktheoretischen Grundlagen. Im Anschluß daran werden in Kapitel 2 die

Planungsaspekte sowie ein Vergleich zwischen den beiden weitverbreitetsten Softwarepaketen für Microsoft- und Unix-basierende Rechnerstrukturen aufgezeigt, welche dann mit in die zu dieser Arbeit entwickelte Referenzimplementierung – beschrieben in Kapitel 3 – einfließt. Kapitel 4 beinhaltet eine Zusammenfassung und gibt einen Überblick über einige Möglichkeiten zur Erweiterung des in dieser Arbeit vorgestellten Systems.

Kapitel 1

Datenbanken

Dieses Kapitel beschäftigt sich mit den für das Ehemaligenverzeichnis wesentlichen datenbanktheoretischen Grundlagen. Ausgehend von einer kurzen Betrachtung verschiedener konzeptioneller Datenmodelle wird auf die Zusammenstellung von Serversystemen durch unterschiedliche Software-Werkzeuge und Plattformen eingegangen.

1.1 Datenbanksysteme

Eine Datenbank ist eine strukturierte Sammlung von logisch zusammengehörenden Informationen (in elektronischer Form). Ein Datenbanksystem stellt dabei – neben der erforderlichen Infrastruktur zur Speicherung von Informationen – Mechanismen zur Verfügung, diese Daten zu verwalten, d.h. Daten einzugeben, abzufragen, zu ändern und zu löschen. Es bildet somit für unterschiedliche Einsatzgebiete in bezug auf solche Datenbankmanipulationsoperationen eine anwendungsunabhängige Abstraktionsebene. Dem Benutzer präsentieren sich die Schnittstellen zu einem Datenbanksystem entweder als interaktives Programm oder als in eine Programmiersprache eingebettetes Interface (das Application Programming Interface (API)).

Die Entwicklung der Datenbanksysteme begann ca. 1970, als in der Industrie in zunehmenden Maße Computersysteme zur Verwaltung von Datenbeständen eingesetzt wurden. Bald zeigte sich, daß die simple Speicherung von Daten ohne geregelte Zugriffsverfahren zu einer Reihe von Problemen wie Redundanz und Inkonsistenz der Informationen führte. Basierend auf theoretischen Arbeiten von

E. F. Codd und I. J. Heath¹ waren Mitte der siebziger Jahre die ersten Datenbanksysteme auf dem Markt verfügbar.

Ausgehend von den Unzulänglichkeiten der bis dahin üblichen Datenverwaltung, lassen sich die Ziele bei dem Einsatz eines Datenbanksystems nach A. Achilles² wie folgt zusammenfassen:

- **Redundanzvermeidung**
Alle Daten werden nur einmal gespeichert.
- **Abstraktion**
Da allein das Datenbanksystem die Zugriffe auf die Daten verwaltet, müssen beispielsweise bei Ergänzung oder Umgestaltung der Datenstruktur nicht alle darauf aufbauenden Applikationen angepaßt werden.
- **Datenschutz**
In einer Mehrbenutzerumgebung ist es in der Regel erforderlich, den Zugriff auf einzelne Informationen einzuschränken.
- **Zuverlässigkeit**
Die gespeicherten Informationen müssen jederzeit abrufbar sein.
- **Datenintegrität**
Änderungen am Datenbestand (insbesondere das Löschen von Informationen) dürfen nicht zu Inkonsistenzen oder Anomalien führen.

Datenbanksysteme lassen sich anhand der Strukturierung der Daten charakterisieren, auf denen sie operieren. Klassische Beispiele für diese Struktur sind das hierarchische, das Netzwerk- und das relationale Modell – vergleichsweise neu ist der objektorientierte Ansatz. Im folgenden werden die wesentlichen Merkmale der einzelnen Modelle kurz vorgestellt. Eine ausführliche Diskussion findet sich in dem Buch *Datenbankeinsatz* von S. M. Lang und P. C. Lockemann³.

Dem hierarchischen Datenmodell liegt eine Baumstruktur zugrunde. Der Zugriff auf einzelne Daten muß dabei immer von der Wurzel aus erfolgen, welche somit den

Einstiegspunkt darstellt. Um an tiefer liegende Informationen in den Knoten des Baumes zu gelangen, muß der Anwender oder das Anwendungsprogramm Kenntnis von der Struktur des Baumes besitzen, um diesen durchlaufen zu können.

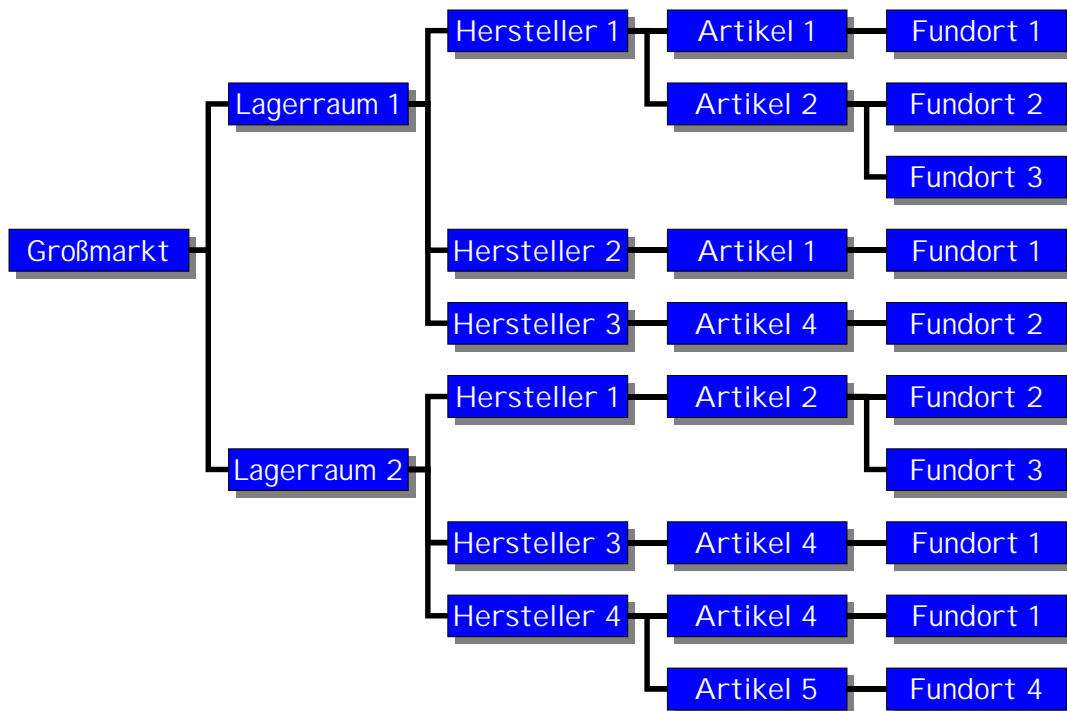


Abb. 1 – Beispiel für ein hierarchisches Datenbankmodell

Das Netzwerkmodell ist dem hierarchischen sehr ähnlich. Es läßt sich insofern als Verallgemeinerung des hierarchischen Ansatzes verstehen, als es die Zugriffsmöglichkeiten nicht auf feste Einstiegspunkte beschränkt, von denen aus nur in eine Richtung durch die Datenstruktur „navigiert“ werden kann.

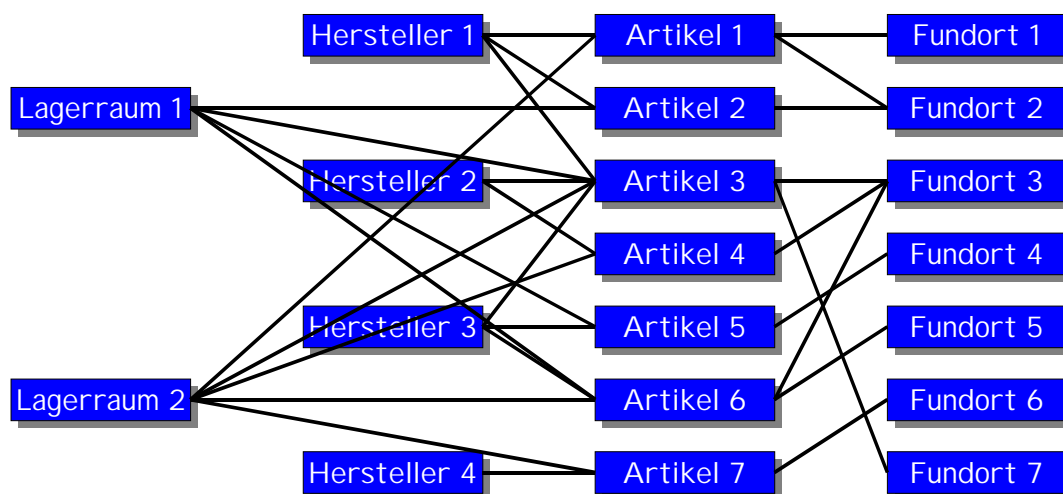


Abb. 2 – Beispiel für ein Netzwerk-Datenbankmodell

Die meisten zur Zeit verfügbaren Datenbanksystemen basieren auf dem sogenannten relationalen Modell und bilden somit eine Art Standard auf dem Gebiet der computergestützten Datenverwaltung. Dies hat im wesentlichen zwei Gründe. Zum einen bietet es – informell ausgedrückt – durch die Strukturierung der darin enthaltenen Daten in Tabellenform (im recht komplexen Gebiet der Datenbanktheorie) auch weniger versierten Anwendern eine vertraute Sichtweise auf die Informationen. Zum anderen existiert mit der sogenannten relationalen Algebra ein mathematischer Formalismus zur Beschreibung der Struktur und Manipulation von Datenbeständen. Daraus resultieren optimale Ausgangsbedingungen für die technische Umsetzung dieses Konzeptes auf einem klassischen Computersystem. Ferner ist auf dieser Basis die Implementierung von Datendefinitions- und Manipulationssprachen möglich, die auch Anwendern ohne detaillierte Programmierkenntnisse die Datenverwaltung und –abfrage erleichtern.

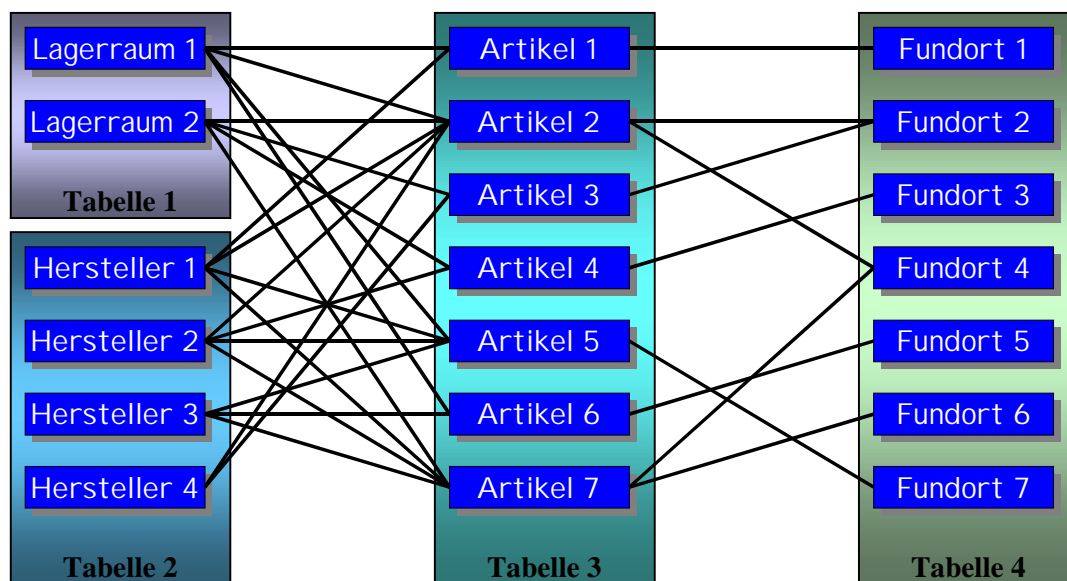


Abb. 3 – Beispiel für ein relationales Datenbankmodell

Die Prinzipien, auf denen der objektorientierte Ansatz aufbaut, entstammen der objektorientierten Programmierung. So verwaltet ein objektorientiertes Datenbanksystem seine Daten als Instanzen von Objekten, die wiederum bestimmten Klassen („Datentypen“) zugeordnet sind. Dabei stehen sowohl die Klassen als auch die Objekte in einer hierarchischen Beziehung zueinander. Die Manipulation des Datenbestandes erfolgt über speziell für jede Klasse zu definierende Methoden

(„Funktionen“). Ein Nachteil des objektorientierten Konzeptes ist jedoch die Tatsache, daß im Vergleich zu dem relationalen Ansatz keine einheitliche formale Definition eines Datenmodells existiert.

1.2 Datenbankserver

Die in der Einleitung durchgeführten Betrachtungen des Funktionsumfangs setzen den Einsatz von insgesamt vier Softwarekomponenten auf dem zu planenden Server voraus. Diese Hauptbestandteile können durch weitere funktionsergänzende Module zu einer Reihe von Ausbaumöglichkeiten des Serviceangebots, aber auch zu einer größeren Komplexität des Datenbankservers führen.

1.2.1 Das Betriebssystem

Der essentielle Bestandteil eines jeden PC-Systems ist sein Betriebssystemkern, im allgemeinen kurz Betriebssystem genannt.

Definition für „Betriebssystem“ (operating system) nach [DIN 44300]⁴:

„Die Programme eines digitalen Rechensystems, die zusammen mit den Eigenschaften der Rechanlage die Grundlage der möglichen Betriebsarten des digitalen Rechensystems bilden und insbesondere die Abwicklung von Programmen steuern und überwachen.“

Bei jedem Start eines Computers wird das Betriebssystem durch das sogenannte „Booten“ (Ursprung: amerikanisches Kunstwort aus „boot“ = Stiefel) geladen. Seine hauptsächliche Aufgabe besteht grundsätzlich aus dem Verwalten von Speicher jeglicher Art. Hierzu zählen der Hauptspeicher und der Speicher auf Datenträgern, wie zum Beispiel Disketten, Festplatten, CD-Roms, DVD oder ähnliche (externe) Speichermedien. Betriebssysteme sind auf die Hardware abgestimmt und stellen dem Benutzer eine Schnittstelle in Form einer mehr oder weniger aufwendigen Programmier- bzw. Kommando-Sprache zur Verfügung, derer sich der Anwender zu bedienen hat, um einen Dienst des Betriebssystems anzufordern (zum Beispiel Treiber).

Als die zur Zeit meistgenutzten Betriebssysteme sind Windows, Unix, Linux, BeOS, FreeBSD und OS/2 zu nennen. Die Nennung von kleineren Systemkernels und -derivaten würde diese Liste sehr stark anwachsen lassen. Obwohl für alle Systemkerne Vergleichsaspekte aufgestellt werden können, wie zum Beispiel Architektur, technische Qualitäten, das Vorhandensein von benutzerfreundlichen Bedienoberflächen, Beigaben zur jeweiligen Distribution, verfügbare Zusätze für eine stetige Erweiterung, benötigter Speicherplatz bzw. Hardwareanforderungen oder die Anschaffungskosten, handelt es sich immer um ein komplexes System, welches die Anforderungen und Bedürfnissen des Benutzers oder des Anwendungsgebietes mit ineinandergreifenden Komponenten befriedigen soll. Der Einsatz eines bestimmten Betriebssystems muß also immer im Hinblick auf den größtmöglichen Nutzen und das Vorhandensein von Erweiterungen erfolgen, denn die beste und stabilste Architektur ist unbedeutend, wenn es an Anwendungen oder Erweiterbarkeit mangelt. Weitere Anforderungen bei der Wahl eines Betriebssystems bilden die Sicherheit, die Einrichtung eines Mehrbenutzerbetriebes, die Netzwerk- und Multimediafähigkeit, sowie die allgemeine Verwaltungsmöglichkeiten.

1.2.2 Der Webserver

Als Webserver werden sowohl Rechner, als auch Software bezeichnet, die WWW-Komponenten einem Client – im allgemeinen ein Web-Browser – zur Verfügung stellen und auf dessen Anfragen entsprechend reagieren. Bei dem Zugriff auf diese Komponente, meist eine HTML-Seite mit eingebetteten Bildern, Skripten oder weiteren Verknüpfungen, wird eine Kopie der angeforderten Webseite erstellt und im Zusammenspiel mit dem Browser auf der Clientseite visualisiert. Abschließend wird diese Verbindung wieder abgebaut und für weitere Anfragen bereitgestellt. Sollte bei diesem Prozeß ein Fehler auftreten, werden diese durch die Webserver-Software abgefangen. So wird zum Beispiel bei einem Aufruf eines nichtexistierenden Dokumentes (auch „ungültige Seite“ genannt) dem Client eine vorher definierte Fehlermeldung übermittelt oder direkt eine Umleitung auf eine angepaßte Seite erfolgen. Des weiteren kümmert sich ein Webserver um sogenannte Server Side Includes (SSI) zur einfachen Erweiterung von gewöhnlichen HTML-Dateien und stellt das Common Gateway Interface (CGI) zur Verfügung, mit dessen Hilfe Programme ausgeführt werden können, die dynamische Webseiten generieren (zum

Beispiel bei Datenbankabfragen). Letztendlich können mit seiner Hilfe auch Log-Files über Zugriffe erstellt und bearbeitet werden, sowie mehrere virtuelle Webserver auf nur einem System mit einer einzigen öffentlichen IP-Adresse eingerichtet werden.

Laut einer fortlaufenden Studie des Serviceanbieters Netcraft⁵ sind der Apache und der Microsoft Internet Information Server (IIS) die derzeit meist eingesetzten Webserver zum Ende des 20. Jahrhunderts. Noch zu nennen wären WebLogic, Netscape-Enterprise, AOLserver und aus Nostalgiegründen der NCSA HTTPd (National Center for Supercomputing Applications, eine amerikanische Forschungseinrichtung), welcher der Urvater aller Webserver ist, jetzt aber so gut wie keine Anwendung mehr findet, so daß sogar auf der eigenen Webseite zu einer Benutzung des Apache-Webserver angeraten wird.

1.2.3 Die Datenbank

Ergänzend zu den Ausführungen in der Einleitung wird an dieser Stelle speziell auf die Software für eine Datenbankeinstellung und -verwaltung eingegangen, welche auch als Datenbank Management System (DBMS) bezeichnet wird.

Definition für „Datenbank“ nach Brockhaus⁶:

„System zur Beschreibung, Speicherung und Wiedergewinnung von umfangreichen Datenmengen, die von mehreren Anwendungsprogrammen genutzt werden. Es besteht aus der Datenbasis, in der die Daten abgelegt werden, und den Verwaltungsprogrammen (Datenbanksoftware), die die Daten entsprechend den vorgegebenen Beschreibungen abspeichern, auffinden oder weitere Operationen mit ihnen durchführen.“

Definition für „Datenbank Management System“ nach A. G. Taylor⁷:

„Ein DBMS ist einen Satz von Programmen, welche dazu dienen, Datenbanken und die mit ihnen verknüpften Anwendungen zu definieren, verwalten und zu bearbeiten.“
(*„a set of programs used to define, administer, and process databases and their associated applications“*).

Somit ist ein DBMS die Software, welche zwischen den spezifischen Anwendungen und den Daten vermittelt und welcher hierfür typischerweise folgende Komponenten implementiert worden sind:

- **Data Definition Language (DDL)**

Diese enthält den Befehlssatz zum Erstellen, Modifizieren oder Löschen einer Datenbank. Es ist die Spezifikation des konzeptionellen und internen Schemas der Datenbank und dient Administratoren und Designer als Grundlage.

- **Data Manipulation Language (DML)**

Diese enthält den Befehlssatz zum Verwalten einer Datenbank, d.h. zum Eintragen, Ändern, Löschen oder Auswählen von Daten aus einer Datenbank. Ein wichtiger Bestandteil der DML ist eine Query Language, die das Auffinden und Aussuchen von Daten ermöglicht.

- **Data Control Language (DCL)**

Diese enthält den Befehlssatz zur Sicherung einer Datenbank. Bestandteile dieser Sicherheit sind u.a.:

- **Security System**

Eine Zugangskontrolle, welche einen unbefugten Zugriff auf Daten oder Veränderung der Daten bzw. der Datenbank verhindert.

- **Integrity System**

Eine Sicherung der Unversehrtheit der Daten und Vermeidung von Inkonsistenzen.

- **Concurrency Control System**

Eine Regelung für den gleichzeitigen Zugang zur Datenbank, insbesondere wenn Veränderungen an einem Datensatz vorgenommen werden sollen.

- **Recovery Control System**

Ein Mittel zur Wiederherstellung der Datenbank im Falle von Hardware- oder Softwareproblemen.

- **User-accessible Catalog**

Eine Gewährleistung, daß alle zugänglichen Daten letztendlich auch einsehbar sind.

- **View Definition Language (VDL)**

Diese erlaubt jedem Nutzer sich seine eigenen Sichtweisen (views) der Datenbank zu schaffen.

Ein Datenbank Management System verwaltet die physischen Details der Datenspeicherung, so daß die Anwendungsprogramme letztendlich nur für die logische Struktur der Daten und ihrer Beziehungen verantwortlich sind und sich nicht zusätzlich um die Art der Speicherung kümmern müssen. Ein DBMS, welches auf verschiedenen Arten von Computern (Mainframes, PCs, Organizer) läuft, nennt man skalierbar („scaleable“), was insbesondere in Netzwerkumgebungen wichtig ist, um verschiedene Arten von Computern (vom Großrechner bis zum Notebook und PDA (Personal Digital Assistant)) miteinander zu verbinden.

Als Entscheidungskriterien bei der Auswahl des richtigen Datenbank Management Systems wären die Faktoren Applikationsgebundenheit, Hardwareanforderungen, Sicherheitsmerkmale, Größenordnung der zu speichernden Daten und Administrationsaufwand zu nennen, welche letztendlich festlegen, ob zum Beispiel Oracle, MySQL, mSQL, Informix, ADABAS, Microsoft Access, Microsoft SQL, Paradox oder dBase am besten für das jeweilige Einsatzgebiet geeignet ist.

1.2.4 Dynamische Webseitengenerierung

Mittels Web-Applikationssprachen wird die Möglichkeit zur Erzeugung von dynamischen Webseiten bereitgestellt, welche tägliche oder gar stündliche Aktualisierungen, Anbindung an bestehende Datenbanken und Informationsdienste oder die Personalisierung des Webauftritts vereinfachen. Durch das Ausführen von kleinen Programmen oder Skripten auf der Server- oder der Clientseite können somit HTML-Seiten erstellt werden, welche auf Benutzeranfragen zugeschnitten sind, indem sie nicht statisch auf dem Webserver abgelegt wurden, sondern die erforderlichen Informationen aus anderen Applikationen und Komponenten (zum Beispiel aus einer Datenbank) des Webserver zusammentragen und eine HTML-Seite generieren.

Definition für „Skript“ nach Dr. S. Wigard⁸:

„Im engeren Sinne versteht man unter ‚Skript‘ ein (meist kleineres) Programm zur Automatisierung von häufig wiederkehrenden Aufgaben, welches von einem (Kommando-)Interpreter unmittelbar ausgeführt wird. Andere Programme werden dagegen vorab von einem Compiler in Maschinencode übersetzt und können dann ohne einen Interpreter ablaufen.“

Bei der Benutzung dieser Art der Webprogrammierung, welche unter der Bezeichnung Dynamic HTML zusammengefaßt wird, unterscheidet man zwischen einer Skriptausführung auf Seiten des Servers (Beispiele sind Common Gateway Interface (CGI), Server-Side Includes (SSI) und Active Server Pages (ASP)) oder auf Seiten des Clients (zum Beispiel Browser-PlugIns, ActiveX Controls, Java Applets, Visual Basic Script (VBScript)).

Die einfachste Möglichkeit für das Publizieren von dynamischen Inhalten wird durch das Common Gateway Interface (CGI) ermöglicht. Hierbei ruft der Webserver ein ausführbares Programm auf und übergibt diesem einige Umgebungsvariablen, welche das CGI-Programm mit dynamisch erzeugtem HTML-Code kombiniert und eine Webseite erstellt, die wiederum vom Webserver als Resultat an den Client weiterleitet wird. Ein Vorteil von CGI ist seine Flexibilität, da jede ausführbare Datei in der Lage ist, HTML-Seiten zu erzeugen (Grundgerüst des HTML-Codes ist eine unformatierte Textdatei) und somit die Verwendung der Programmiersprache, wie zum Beispiel C oder Java, nicht vorgeschrieben ist. Damit ist eine CGI-Sprache auf nahezu jeder Plattform einsetzbar. Im Falle einer Systemumstellung, zum Beispiel von HP-Unix auf Solaris, werden die vorhandenen Skripte in das CGI-Verzeichnis portiert und sind auch dort ohne Einschränkungen einsetzbar. Als Schwachpunkt zählt das Starten und Beenden des CGI-Programms bei jedem Aufruf, da das Betriebssystem jedesmal einen Prozeß startet, welcher Speicher reserviert und erst nach Beendigung des Prozesses diesen wieder freigibt. Dies führt bereits bei einigen wenigen Aufrufen pro Minute zu einer großen Hardware- und Ressourcenauslastung des Systems und kann ernste Performanceprobleme nach sich ziehen, insbesondere wenn sich die Abrufzahlen auf mehrere Hits pro Sekunde steigern. Zudem lassen

sich identische CGI-Aufrufe nicht zwischenspeichern, sondern müssen bei jedem Programmaufruf ausgeführt werden.

Eine Lösung des oben genannten CGI-Performance-Problems ist der Einsatz von Webserver-Erweiterungen, damit die erstellten Programme nicht extern, sondern innerhalb des Webserver-Prozesses ablaufen können. Hierdurch entfällt das aufwendige Starten und Beenden von Prozessen, da der Webserver die Erweiterungen nach der ersten Anforderung in den Speicher lädt und für alle gleichzeitigen Zugriffe dieselbe Instanz der Servererweiterung verwendet werden. Das Manko der Webserver-Erweiterungen ist die fehlende Portierbarkeit auf andere Systeme, da diese sich genau an das Application Programming Interface (API) des jeweiligen Servers halten müssen. Microsoft-Server basieren zum Beispiel auf der ISAPI, während die Server von Netscape die NSAPI unterstützen. Doch trotz dieser Einschränkung gelten sie immer noch als die schnellste Möglichkeit dynamische Seiten zu liefern, wenn die optimale Performance eine Rolle spielt.

Aufgrund der rapiden Leistungssteigerung bei der Prozessorentwicklung ist es sinnvoll, einen gesunden Mittelweg zwischen Portierbarkeit, komfortabler Entwicklung und Performance zu finden. Hier finden Techniken wie Active Server Pages (ASP) oder PHP Hypertext Preprocessor (PHP) Anwendung, denn bei beiden Lösungen handelt es sich um Webserver-Erweiterungen, die HTML-Code mit Hilfe von Skriptsprachen generieren. Durch die Einbindung der Skriptbefehle durch spezielle Tags in eine HTML-Seite ist die Trennung von reinen HTML-Seiten und dynamischen Seiten fließend und im Gegensatz zu CGI-Programmen muß nicht der gesamte HTML-Code dynamisch generiert werden. Die Einbettung der Skripte erleichtert deshalb die Webseitenprogrammierung – in der Regel mit einem Texteditor oder einem komfortablen HTML-Editor wie Microsoft Frontpage oder Dreamweaver von Macromedia – und sorgt für mehr Übersicht, da der Ausgangspunkt wieder die Webseite und nicht ein Programm ist.

Eine Weiterentwicklung obiger Technik stellen die Web Application Server dar, welche eigenständige, vom Webserver unabhängige Server sind, die sich nur um die Programmlogik und HTML-Erzeugung kümmern. Was zunächst wie ein Nachteil

aussieht, ermöglicht aber den Einsatz von hochspezialisierten Servern, die ganz auf verteilte Internet-Applikationen ausgerichtet sind und durch die getrennte Architektur zum Webserver eigene Methoden zur Verarbeitung und Überwachung in Anspruch nehmen können, was sich wiederum positiv auf die Performance des Webservers auswirkt. Diese Features stellen allerdings Mindestanforderungen an die verwendete Technik oder Skriptsprache, denn aufgrund des verbindungslosen HTTP-Protokolls ist zum Beispiel die Wiedererkennung des Benutzers sowie die Aufzeichnung seiner Anfragen von großem Interesse. Einige Systeme bieten hier ein komfortables Session-Management an, mit dem sich benutzerabhängige Daten auf den unterschiedlichsten Webseiten lesen oder schreiben lassen. Weitere Standardanforderungen sind neben der unabdingbaren Formularauswertung das Verschicken von eMail-Nachrichten, das Kodieren von Inhalten zur Weitergabe und das Auslesen der Browser-Information.

1.2.5 Zusätzliche Module

Einer stetigen Erweiterung des Datenbankservers durch Module verschiedenster Gebiete steht im Grunde nichts im Wege, doch es sollte ein Auge auf eventuelle Performanceeinbußen durch eine übermäßige Bereitstellung von Funktionen geworfen werden. Die Einrichtung von zusätzlichen Serviceleistungen und Features sollte also immer im Einklang mit der jeweils vorhandenen Hardware geschehen, da Datenbanken im allgemeinen stetig anwachsen und somit zum Beispiel eine Suchroutine, die während der ersten Stunden des Servers noch zügig erfolgt, nach Jahren anwachsender Datenbestände schon etwas mehr Zeit in Anspruch nehmen kann.

In der Regel sind allgemeine Module bereits in der jeweiligen Softwaredistribution enthalten. In den meisten Fällen (zum Beispiel bei der Kombination Linux und Apache) können Installationen auf ein Minimum der Anforderungen heruntergeschraubt werden und sich auf die Bedürfnisse des gesamten Projektes einstellen. Mögliche Zusätze wären zum Beispiel Sicherheitsprogramme, die durch Backup-routinen oder Firewall-Einstellungen für mehr Sicherheit sorgen, Serviceprogramme, die durch Datenreorganisation und performancesteigernde Aktivitäten den reibungslosen Betrieb des Servers sicherstellen oder, wie in unserem Fall, ein Mailprogramm,

welches durch einen regelmäßigen eMailkontakt Newsletter oder Erinnerungsschreiben an gespeicherte Adressaten verschickt.

Kapitel 2

Planung und Vergleichsaspekte

Zu einer erfolgreichen Projektierung mit anschließender Implementierung gehört stets eine detaillierte Planung, welche mögliche Schwachstellen und Fehler aufdecken soll und somit einen reibungslosen Aufbau und Betrieb des Projektes gewährleistet. Im folgenden Kapitel wird auf die verschiedenen Planungsaspekte eingegangen und die in Kapitel 1 besprochenen Hauptbestandteile der Softwareimplementierung einem Vergleich unterzogen. Hierbei wird das Hauptaugenmerk auf die gängigsten Softwareprodukte gelegt, welches eine Betrachtung von Microsoft- und Unix-basierenden Rechnerstrukturen nach sich zieht. Abschließend werden aufgrund von Hardwareanforderungen und der vorhandenen Infrastruktur erste Empfehlungen ausgesprochen und Projektgrundlagen abgesteckt.

2.1 Planungsphase

Es gilt ein System zu entwerfen, welches eine Datenbank enthält, die sowohl aus dem Intranet als auch aus dem Internet erreichbar sein soll, jedoch aufgrund des Datenschutzes verschiedenen Benutzergruppen nur spezifische Zugriffsmöglichkeiten geben darf. Außerdem sollen Benachrichtigungs- und Administrationsmechanismen für einen weitgehend autarken Betrieb sorgen und somit zusätzliche administrative Eingriffe von außen minimieren, wenn nicht sogar gänzlich ausschließen. Die Einsicht in die Datensätze soll ohne die Installation eines zusätzlichen Softwarepaketes erfolgen, was schließlich eine serverseitige Datenverarbeitung und Erstellung des Ausgabedokumentes voraussetzt. Hierfür bietet sich

die Benutzung von dynamischen Webseiten an, da diese mit gesammelten und gespeicherten Daten eine Ausgabe nach den Bedürfnissen des Besuchers erstellen können, die von jeder Arbeitsstation mittel eines Browsers abrufbar sind.

Betrachten wir die verschiedenen Planungsaspekte gestaffelt nach Gebieten und versuchen erste Ziele abzustecken:

Hardware:

Hierbei sollte ein Mittelweg zwischen Ausstattung und Sicherheit gefunden werden, da zum einen der Kostenaufwand möglichst gering ausfallen soll, zum anderen aber ein Verlust der Daten durch Hardwaredefekte nach Möglichkeit ausgeschlossen oder zumindest minimiert werden muß. Also reicht der Einsatz eines Standard-PCs vollkommen aus und da der Rechner nicht als Arbeitsstation genutzt wird, richten sich die Hardwareanforderung hauptsächlich nach der zu benutzenden Software. Sicherheitsmechanismen, wie zum Beispiel Festplatten-RAID, redundante Netzteile oder eine USV (unterbrechungsfreie Stromversorgung), müssen mit den uns gegebenen Vorgaben im Sinne der Kostenminimierung vernachlässigt werden, sollten aber bei Erweiterung des Servers durchaus wieder in die Planung mit einfließen.

Software:

Die Wahl der Software wird beeinflusst durch die Komplexität des zu erstellenden Systems. Wie bei der Hardwareausstattung kann auch hier ein Einsatz von Standardkomponenten aufgrund der einfachen Umstände in Betracht gezogen werden. Bereits angesprochen wurden die vier Grundbausteine der Software (Betriebssystem, Webserver, DBMS und Skriptsprache) und als zusätzliche Module werden ein eMail-Client und eine Backup-Lösung benötigt. Der Client dient zur Automatisierung der Adressenaktualisierung und das Backup soll einem Datenverlust nach einem Absturz, Hardwareausfall oder -defekt entgegenwirken. Weitere Softwarepakete werden durch den reinen Servereinsatz überflüssig und sollten nicht zu einer unnötigen Ressourcenbeschaffung in Form von Hardware führen. Zum Beispiel ist der Verzicht auf einen Virens scanner trotz Internetanbindung in Erwägung zu ziehen, da von außen keine potentiell verseuchten Dateien aufgespielt werden können

(Upload) und auch das Ausführen von möglicherweise Viren enthaltenen Programmen auf dem Server nicht möglich ist. Aufgrund des von uns geplanten Systems werden Eingriffe von Außen auf ein Minimum reduziert, so daß eine ständige Überwachung durch ein Virenschild oder ähnlichem zwar empfehlenswert aber nicht zwingend notwendig ist.

„Manpower“

Da es sich um ein hauptsächlich autarkes System handelt, bei dem systemgefährdende Eingriffe von Außen auf ein Minimum reduziert werden, sollte der Einsatz von Spezialkenntnissen nur im Falle der Servererweiterung und Implementierung von neuen Komponenten notwendig sein – oder im Reparaturfall. Dies vermeidet im normalen Betrieb die zusätzliche Beschäftigung bzw. den kostenaufwendigen Eingriff eines Administrators oder die besondere Schulung von Mitarbeitern. Im Hinblick auf eine zukünftige Ausweitung des Funktionsumfangs und der Wartung des Systems wäre allerdings ein Grundwissen über die Systemkomponenten zu empfehlen und deshalb werden Fortbildungsmaßnahmen, zum Beispiel Schulungen, in unseren weiteren Betrachtungen mit einfließen. Bei Abfragen der Datenbank werden keine besonderen Kenntnisse vorausgesetzt, da der Besucher mit einer ihm sehr gut bekannten Oberfläche, nämlich seinem Browser, arbeitet und damit Datensätze ändern oder ergänzen kann.

Betriebskosten

Durch die Verwendung von Standardkomponenten und der vorhandenen Infrastruktur, wozu auch die permanente Anbindung an das WWW zählt, lassen sich die Betriebskosten nur auf den Stromverbrauch eines zusätzlichen PCs und eventuelle Ersatzteile bei Hardwareverschleiß beziffern. Weitere besondere Maßnahmen, wie zum Beispiel der Gebrauch einer Kühleinheit, Bereitstellung eines eigenen Serverraumes, zusätzliche Raumsicherheits- und Datenleitungskosten entfallen bei dieser Dimensionierung des PCs.

2.2 Vergleichsphase

Aufgrund der geplanten Ziele und Anforderungen, welche an das gesamte Serversystem gestellt werden, und das Ineinandergreifen der bereits erwähnten Softwarekomponenten, werden in den nächsten Unterkapiteln die gängigen Produkte mit ihren Vorzügen und Einschränkungen vorgestellt und einem Vergleich unterzogen, damit alle Erwartungen erfüllt werden können und sich jeweils die optimale Variante herauskristallisiert. Schon die Auswahl des Betriebssystems beinhaltet eine Vorentscheidung für die weitere Software, denn eine Kompatibilität auf Betriebssystembasis gibt es bekanntlich bis heute nicht.

2.2.1 Vergleich der Betriebssysteme

Die Liste der freien und kommerziellen Betriebssysteme ist lang und jedes hat seine speziellen Vorzüge, aber wenn man Verbreitung und somit ein Zusammenspiel mit anderen Produkten zugrunde legt, lichtet sich der Betriebssystemdschungel und zwei Vertreter bleiben übrig: Microsoft Windows und eine Vielzahl von Unix-Distributionen. Letztere werden im allgemeinen Hausgebrauch (im Gegensatz zu aufwendigen Serversystemen) hauptsächlich von Linux dominiert, welches sich trotz des Open Source-Systems (Quellcode ist öffentlich verfügbar) nicht hinter dem kommerziellen Produkt Windows verstecken muß und in Punkto Sicherheit, Stabilität und echtes 32 Bit Multitasking ohne Probleme mithalten kann. Da für das hier zu planende Serversystem ein großes Maß an Datensicherheit und Zugriffsbeschränkungen erforderlich sind, fallen Windows 95/98/ME aus den Betrachtungen heraus und nur die Serverschiene von Microsoft (Windows NT und Windows 2000) wird in weiteren Beschreibungen und Vergleichen herangezogen.

Verbreitung und Distributionen:

Microsoft Windows ist das meistverwendete Betriebssystem auf der ganzen Welt und nach der Einführung von Windows 2000 etablierte es sich durch die vielen Neuerungen, wie zum Beispiel SMTP-Server, Telnet-Server und Einschränkungen des verfügbaren Festplattenplatzes (sogenannte Disk-Quotas), zu einem ausgereiften Netzwerkbetriebssystem, was bislang eher von Unix- und Novell-Servern dominiert worden ist. Im Zuge des – uns teilweise unverständlichen – Microsoft-Hasses

avancierte Linux mit seinen bekanntesten Distributionen SuSE, Debian, RedHat, Caldera und Mandrake zu einer oft eingesetzten SoHo-Lösung (Smalloffice/Homeoffice = Angebote für kleine Netzwerklösungen) bis hin zu Midrange-Solutions. Beide Systeme werden ausreichend mit der von uns benötigten Software unterstützt und auch Support und Hilfestellungen in digitaler (Internet & eMail) und gebundener Form (Bücher & Zeitschriften) sind zur Genüge vorhanden.

Produktkosten und Lizenzfragen:

Hier liegt der Vorteil eindeutig bei eine Linuxinstallation, denn durch die Regelung des GNU General Public License (GPL) ist dieses Betriebssystem kostenlos erhältlich und einsetzbar, ohne Rücksicht auf die Benutzung als Einzelplatz- oder Multiusereinsatz oder ob es einem privaten oder kommerziellen Zweck dient. Lediglich für eine Distribution durch Drittanbieter, wie zum Beispiel durch die Firma Suse GmbH – die in Deutschland meistverkaufte Distribution – wird ein Entgelt erhoben, welches sich aber auf die Auslieferung des Linuxkernels auf einem Medium (CD oder DVD), die ausführliche Dokumentation in Buchform und die Bereitstellung eines Supportes bezieht, nicht aber auf den Source-Code selbst. Im Falle der SuSE-Distribution belaufen sich die Kosten auf 89 DM für eine Personal Edition und auf 129 DM für eine Professional Edition für das SuSE Linux 7.0 (Quelle: SuSE GmbH⁹), welches das gesamte Softwareangebot einmal auf sechs CDs oder einer DVD (beides im Paket enthalten) anbietet. Bei Windows gestaltet sich die Preispolitik schon etwas schwieriger, denn Microsoft unterscheidet zwischen Einzelplatz- und Mehrplatzlizenzen, ob mit oder ohne OptionPack, als Enterprise Edition und diverse andere Kombinationen und Ausführungen. Als Beispiel für den hohen Preis dieses kommerziellen Produktes sollen die Kosten eines Standardserver inklusive fünf Clients dienen. Laut der Firma BtB OHG¹⁰ liegt der Preis des Windows NT 4.0 Servers inklusive OptionPack bei 2.265 DM und des Windows 2000 Servers bei 1.599 DM.

Des weiteren entstehen bei der Anschaffung der PC-Hardware Kosten, wobei der Windows-Server auch hier nicht gerade als Kostverächter bezeichnet werden kann, denn die Hardwareanforderungen hinsichtlich der Performance und Ressourcen sind im Vergleich zu denen eines Linux-Systems immens. In der Regel veranschlagt man

für den produktiven Einsatz von Windows 2000 eine Rechnerausstattung mit einem Intel PII 400 MHz, 256 MB Arbeitsspeicher und ca. 1 Gigabyte Festplattenspeicher nur für die Systemdateien – zusätzliche Softwarekomponenten wie Webserver oder Datenbank noch nicht mit eingerechnet. Linux wiederum kann auch auf älteren Maschinen ausgeführt werden, welche technisch nicht mehr auf dem neuesten Stand sind, wobei die untere Grenze aber im Falle einer ausgeglichenen Auslastung der Ressourcen bei einem Pentium 233 MHz und 128 MB Arbeitsspeicher anzusiedeln ist.

Funktionalität und Performance:

Ein Redakteur der c't hat in der Ausgabe 17/2000 eine grundlegende Charakterisierung von Windows 2000 und Linux mit einem Bild aus dem Kinderzimmer vorgenommen:

„Linux ähnelt einem Berg bunter Lego-Bausteine - zwar kann man alles Mögliche damit bauen, aber bevor man sich diese Mühe macht, stellen sie nichts dar. Windows entspricht der fertigen Plastik-Ritterburg - auspacken und loslegen, aber natürlich kann man kein Flugzeug draus machen, wenn einem das Ritterspielen langweilig geworden ist.“

Dieser Vorteil von Linux läßt sich durch den immer erweiterbaren Modulaufbau erklären, der den Einbau und Ausschluß von Hardwaretreibern, verschiedene Dateisysteme, jedes Netzwerkprotokoll und viele Funktionen wie die Nutzung bestimmter Prozessorfähigkeiten bei der Erstellung eines Kernels zuläßt. Somit kann jeder Benutzer seinen Kernel optimal an das eigene System und die eigenen Ansprüche anpassen und unnötige Ressourcenbindungen, wie sie durch die relativ starre Installation eines Windows-Betriebssystems auftreten, vermeiden. Die Treiber-
verfügbarkeit für Linux ist zwar eingeschränkt, da wir aber die Installation eines Servers mit nicht ständig wechselnden Standardkomponenten betrachten, fällt dieser Punkt eher unbedeutend für weitere Betrachtungen aus. Auch Büroanwendungen und Multimedia fallen in diese Kategorie, da der zu planende Server im stand-alone-Modus betrieben wird, was sogar einen Betrieb ohne Monitor, Tastatur und Maus ermöglicht.

Beide Systeme werden als „multi-user“-fähig bezeichnet, doch nur Linux wird den Definitionsansprüchen gerecht, denn hier können wirklich mehrere Benutzer gleichzeitig am System arbeiten und Anwendungen unter Mitbenutzung der Serverhardware starten, sofern die Berechtigungen des Benutzer dies zulassen. Bei Windows wird diese Fähigkeit mit gleichzeitigem Dateien-, Drucker- und Netzwerkzugriff begründet, aber eine Ausnutzung der zum Teil höheren Rechenleistung des Servers und Dezimierung von Last auf der Arbeitsstation ist im Gegensatz zu Linux-Servern nicht möglich. Ein Windowsnutzer kann nur solche Anwendungen benutzen, die in zwei Teilen geschrieben wurden, im allgemeinen Client/Server-Anwendungen genannt. Gebraucht wird diese Fähigkeit insbesondere für Aufgaben wie Administration oder rechenintensive Batchabläufe, wo ein direkter Serverzugriff unzweifelhaft von Vorteil ist.

Eine weitere Schwäche von Windows ist das antiquierte Festplattendesign, welches durch die Vergabe von Laufwerksbuchstaben dem Gesamtsystem hardware-bedingte Grenzen aufbürdet. Unter Unix – und somit auch Linux – können gemeinsame Netzwerk-Dateisysteme an jeden Punkt der Verzeichnisstruktur gemountet (eingehängt und verfügbar gemacht) werden, was dazu führt, daß eine Netzwerkressource mehrere Festplatten oder verschiedene Rechner umfassen kann und damit dem Administrator gestattet, existierende und dem Benutzer bekannte Verzeichnisstrukturen beizubehalten, trotzdem aber den verfügbaren Festplattenspeicher auf dem Server für den Benutzer transparent zu erweitern (zum Beispiel nach Einbau einer neuen Festplatte zwecks Erweiterung des Speicherplatzes). Dieser Unterschied zwischen den UNIX- und Windows-Betriebssystemen unterstreicht einmal mehr die ursprünglichen Absichten ihrer jeweiligen Entwickler, denn Unix wurde als Client/Server-Betriebssystem für die professionelle Nutzung entwickelt, während Windows und seine Nachfolger von DOS abstammen, einem Betriebssystem, welches niemals für den Einsatz in Client/Server-Umgebungen gedacht war, noch weniger als Server.

Systemmanagement:

Hierbei punktet eindeutig das Microsoft-System, denn wenn man nicht gerade ein sehr versierter Benutzer im Umgang mit Kommandozeilenbefehlen ist, erleichtert die

graphische Oberfläche von Windows die Arbeiten am Server ungemein. Dies wirkt sich natürlich bei Bedarf auf den Schulungsaufwand von Mitarbeitern aus und auch der doch häufiger vorkommende private Umgang mit Windowsversionen erleichtert ein Umdenken auf die Serverplattform bzw. die ähnliche Benutzeroberfläche. Unix bietet zwar GUIs (Guided User Interface) anstelle von CLIs (Command Line Interface) an, doch in der Regel können sie nicht mit der jahrelangen Erfahrung Microsofts bezüglich graphischen Desktopdesigns mithalten.

Einen negativen Eindruck hinterlassen die Redmonder auch mit dem ständigen Neubooten nach systemrelevanten Änderungen, denn wenn Linux schon seit früheren Versionen ladbare Gerätetreiber implementieren kann, welche einen Reboot verhindern, sollte die Entwicklung solcher serverrelevanten Features – jeder Ausfall verursacht Kosten und Verzögerungen – für ein Windowssystem mit höchster Priorität gehandhabt werden.

Sicherheit:

Da es sich in unserem Fall um Daten handeln, welche dem Datenschutzgesetz unterliegen und somit besondere Sicherheit gegen unbefugten Zugriff gewährleistet werden muß, ist eine Betrachtung dieses Vergleichsaspekts mit besonderer Aufmerksamkeit vorzunehmen. Dies hat schließlich dazu geführt, daß bestimmte Betriebssysteme bereits zu Beginn der Auswahl ausgeschlossen worden sind.

Schon die grundlegende Absicherung von Dateien, nämlich das Löschen von Systemdateien durch nichtprivilegierte Benutzer, wird unter Linux besser berücksichtigt, denn dieses Recht ist nur dem Root (bei Windows ist dies der Administrator) vorbehalten und eine Ernennung eines Benutzers zum User mit root-ähnlichen Rechten, wie bei Windows, gibt es nicht. Zudem legt das Sicherheitskonzept von Unix bereits bei der Erstellung von Dateien fest, welche Rechte der Eigentümer, die Gruppe und die Allgemeinheit hat, diese zu ändern oder gar auszuführen. Das Windows-NT-Prinzip ist im Vergleich zur Unix-Datensicherheit rudimentärer und einfacher gehalten und erst mit Windows 2000 wurde die Benutzeradministration einer komplette Überarbeitung unterzogen. Das Kennwortsystem von Windows wiederum ist im allgemeinen sicherer als das von Unix, denn

es ist schwieriger, Paßwörter des Windows NT/2000-Logarithmus zu knacken, da sie einen längeren Enkryptionsschlüssel verwenden.

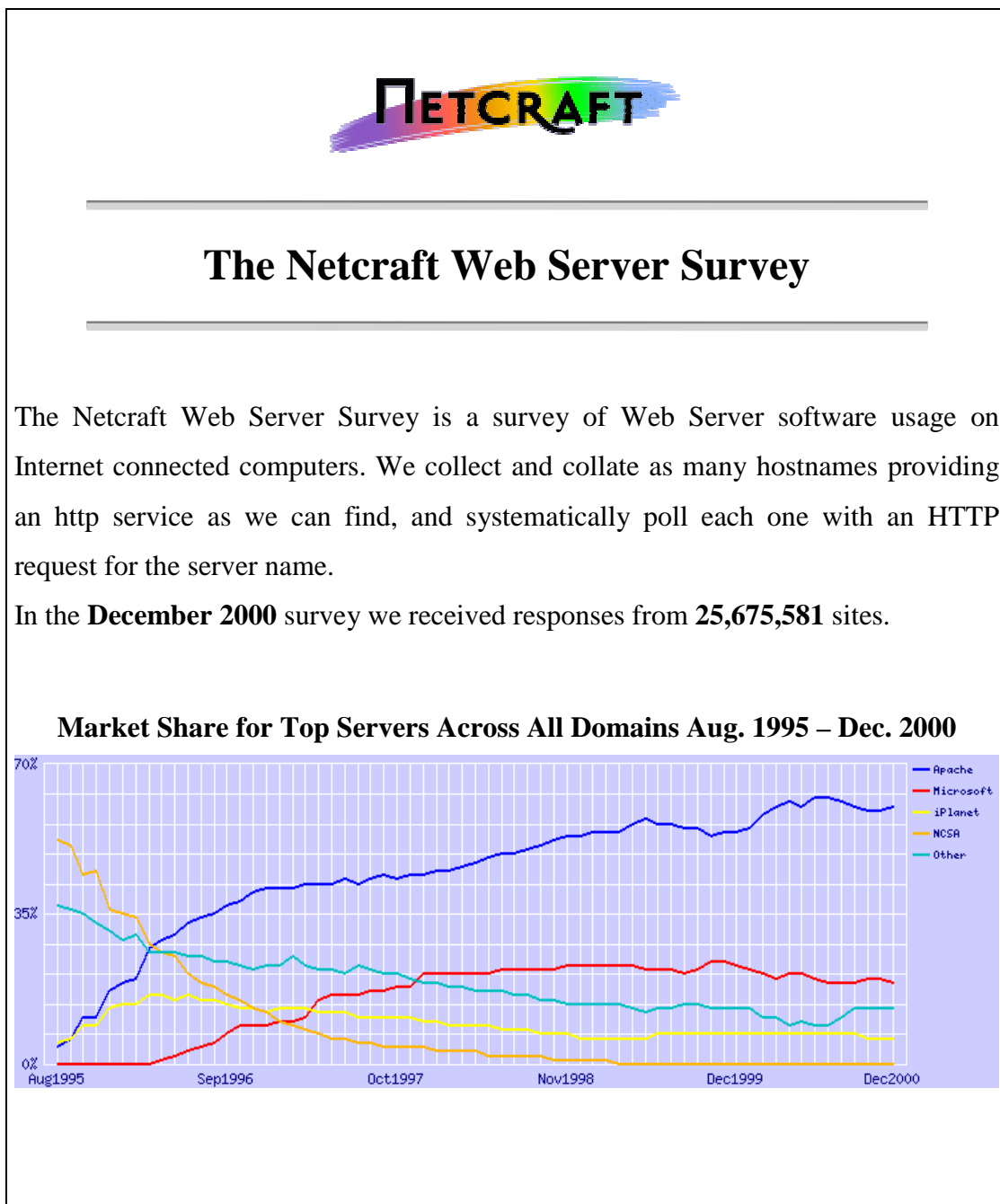
Trotz vieler Vorbehalte gegen Windows und die Anfälligkeit gegen Angriffe, zum Beispiel durch Viren oder Trojaner, welche durch die große Verbreitung der Microsoft-Betriebssysteme im privaten und somit weniger gesicherten Bereich forciert wird, einigen sich die beiden Kontrahenten auf ein Unentschieden in weiteren Sicherheitsfragen (zum Beispiel Filesystem oder Firewall). Insbesondere die letzten Attacken in der Internetwelt haben bei der Entwicklung von Windows 2000 einen Innovationsvorschub bewirkt und die Sicherheitslücken des Betriebssystemkerns sind in etwa mit denen eines Unix-/Linux-Systems zu vergleichen. Hier bedarf es schon einiger Hackererfahrungen und fundierten Kenntnissen, um Sicherheitsbarrieren zu überwinden. Der große Verbreitungsgrad von Windows führt in diesem Fall auch zu einer Vermehrung derer, die sich mit diesem System auskennen, welches wiederum einen erhöhten Angriff auf Windowssicherheitssysteme zur Folge haben kann. Außerdem bieten unausgereifte Zusatzprogramme öfter mehr Angriffsfläche als der eigentliche Betriebssystemkern, wie uns der I-LOVE-YOU-Virus im Mai 2000 eindrucksvoll an Microsofts Outlook demonstriert hat – andere Mailprogramme wurden von den Attacken verschont bzw. die Auswirkungen für den eigenen PC fielen deutlich geringer aus.

2.2.2 Vergleich der Webserver-Software

Durch die Entscheidung, daß als Betriebssystem Windows oder Linux zum Einsatz kommt, beschränkt sich die Auswahl des Webserver auf das reine Windowsprodukt Internet Information Server (im Folgenden als IIS bezeichnet) und dem sowohl für Linux als auch für Windows erhältliche Apache-Webserver. Im Gegensatz zum IIS und dem Apache für Linux ist die Apache-Version für Windows kostenpflichtig und da mächtige Features des Apaches unter einem Microsoft-Betriebssystem nicht verfügbar sind (weil sie eben auf Unix-Systemkonzepten basieren), wird jener aus den weiteren Betrachtungen als möglicher Einsatzkandidat herausfallen. Übrig bleiben also zwei kostenlose Softwarepakete – der Apache ist wie Linux ein GPL-Produkt und der IIS ist bei der Kostenaufstellung der Betriebssysteme bereits berücksichtigt worden – welche auch die Spitzenpositionen bei den weltweiten

Webservereinsätzen nach einer Erhebung¹¹ des Internet-Serviceanbieters Netcraft (siehe Abbildung 4) halten. Demnach wird der Apache auf ca. 60 % und der IIS auf ca. 20 % der Internet-Webserver eingesetzt, was die Vertrauensfrage in Punkto Sicherheit und Stabilität eindeutig zu Gunsten des Apache ausfallen läßt. Wir werden im Folgenden weiterhin eine wertneutrale Betrachtung der Vorzüge und Schwachstellen vornehmen.

Ein Auszug der Netcraft-Webseite vom 17. Januar 2001:



Fortsetzung auf der nächsten Seite

Top Developers

<u>Developer</u>	<u>November 2000</u>	<u>Percent</u>	<u>December 2000</u>	<u>Percent</u>	<u>Change</u>
Apache	14193854	59.69	15414726	60.04	0.35
Microsoft	4776220	20.09	5027023	19.58	-0.51
iPlanet	1643977	6.91	1722228	6.71	-0.20

Top Servers

<u>Server</u>	<u>November 2000</u>	<u>Percent</u>	<u>December 2000</u>	<u>Percent</u>	<u>Change</u>
Apache	14193854	59.69	15414726	60.04	0.35
Microsoft-IIS	4774050	20.08	5025017	19.57	-0.51
Netscape-Ent.	1605438	6.75	1682737	6.55	-0.20
WebLogic	789953	3.32	890791	3.47	0.15
Zeus	640386	2.69	676526	2.63	-0.06
Rapidsite	347307	1.46	365807	1.42	-0.04
thttpd	226867	0.95	321944	1.25	0.30
tigershark	120213	0.51	139300	0.54	0.03
AOLserver	136326	0.57	125513	0.49	-0.08
WebSitePro	106618	0.45	110681	0.43	-0.02

Active Sites

<u>Developer</u>	<u>November 2000</u>	<u>Percent</u>	<u>December 2000</u>	<u>Percent</u>	<u>Change</u>
Apache	5770634	59.95	6218326	59.54	-0.41
Microsoft	2483691	25.80	2615133	25.04	-0.76
iPlanet	247976	2.58	278724	2.67	0.09

Fortsetzung auf der nächsten Seite

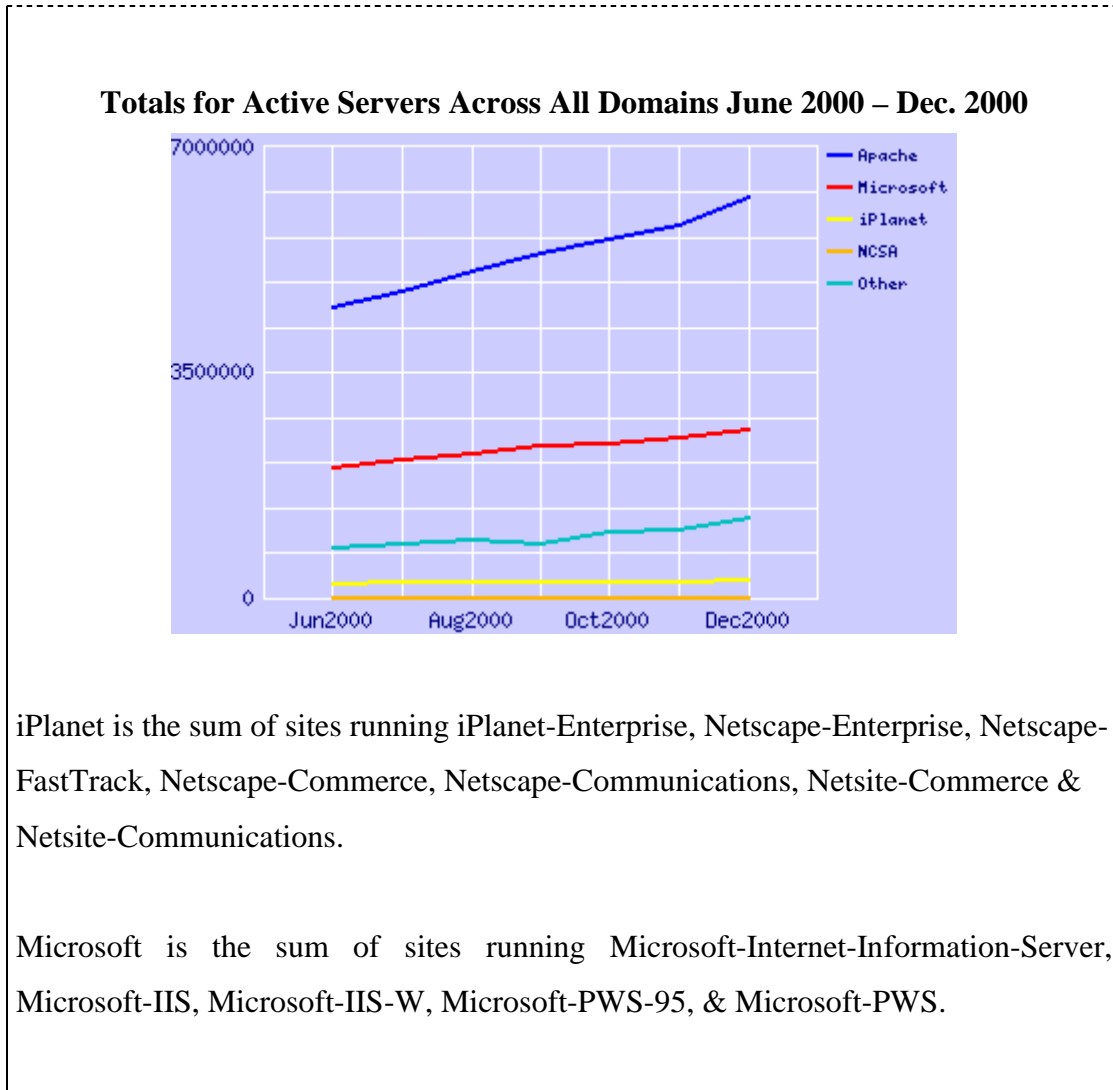


Abb. 4 – Webserver Survey der Firma Netcraft (Auszug der Webseite)

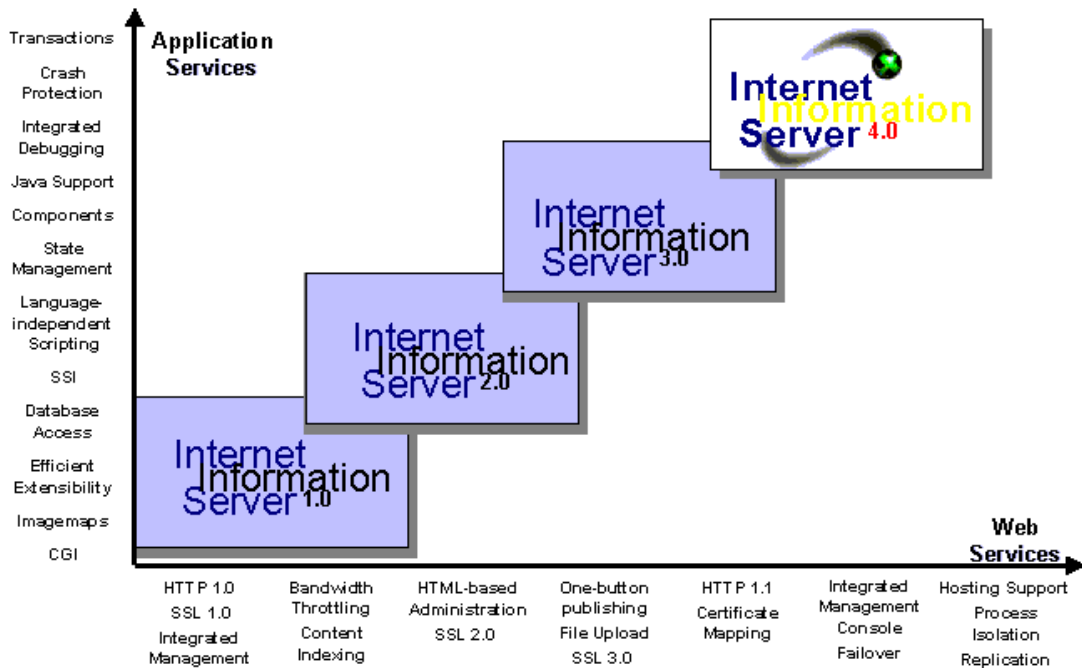
Der Apache-Webserver:

Ein wichtiger Meilenstein in der Entwicklung von Webservern ist das Virtual Hosting, welches einen Betrieb von mehreren Webservern unter einer IP-Adresse bezeichnet – zuerst unterstützt vom Apache. Bei dem Vergleich der Installation und Administration verschafft der Unix-typische Modulaufbau abermals dem Apache einen Vorteil, denn die individuelle und angepaßte Auswahl der Softwaremodule steht der relativ starren Installationsroutine des IIS gegenüber. Diesem modularem Aufbau verdankt der Apache auch unzählige Erweiterungen, die der Server im Laufe der Jahre erfahren hat. In punkto Rechenleistung und Speicherbedarf verhält sich Apache recht genügsam, da er auf ressourcenfressenden Ballast, zum Beispiel eine

aufwendige graphische Oberfläche in der Grundversion, verzichtet. Erst die separate Installation einer frei downloadbaren GUI vereinfacht das Administrieren auch für den unerfahrenen Verwalter, führt aber wiederum zu den zuvor aufgeführten Belastungen des Systems. In der Praxis werden Konfigurationsänderungen und Modulerweiterungen aber nur äußerst selten vorgenommen. Wenn man zudem die oben angesprochene Serverplazierung vornimmt, empfiehlt sich eine Administration von einer anderen Arbeitsstation aus, zum Beispiel per Konsole oder Remote-Zugang. Hervorragend gestaltet sich die Dokumentation und der Support, da die Apache-Foundation einen umfangreichen Vorrat an Manuals, Buglisten, FAQs (Frequently Asked Questions), Foren und sogar einen kommerziellen Supportservice anbietet. Durch den Open Source-Code werden Fehler im Programm schnell behoben und anschließend als neuere Programmversion auf der Apache-Webseite bereitgestellt. Dieser Unix-Webserver ist aufgrund seiner Flexibilität, seinen vielen Leistungsmerkmalen und der einfachen Handhabung (wenn man mit ihm vertraut ist) unschlagbar.

Der Microsoft Internet Information Server:

Lange hieß es, das der IIS „eines der wenigen Dinge ist, die zur Zeit mit Windows NT geliefert werden“. Bei Windows NT noch ausgelagert auf dem OptionPack, ist er nun fester Bestandteil von Windows 2000 und läßt sich in gewohnter Windows-Manier unproblematisch installieren und durch die graphische und Explorer-ähnliche Oberfläche MMC (Microsoft Management Console) einfacher administrieren. Dieses Feature wird aber durch den auftretenden Performanceverlust wieder zur bitteren Pille (siehe Ausführungen beim Apache, Seite 28f).



Quelle: <http://www.microsoft.com/technet/IIS/prodfact/revguide/40revgd.asp>

Abb. 5 – Fortschritte bei der Versionsentwicklung des IIS

Er arbeitet hervorragend mit Zusatzprogrammen von Windows NT/2000 zusammen, die zum Beispiel für eine Beobachtung der Serveraktivitäten oder für ein Sammeln der Leistungsdaten eingesetzt werden können. Auch auf die Windows 2000-beherrschenden Assistenten muß man nicht verzichten, und bei der NT-Version ist eine umfangreiche Dokumentation im OptionPack mit inbegriffen. Lediglich der offizielle Microsoft-Support gestaltet sich nicht gerade hilfreich, denn wenn die Knowledge Base der Webseite nicht mehr weiterhelfen kann, muß man den, im Vergleich zu anderen Softwareanbietern sehr teuren, Telefonsupport in Anspruch nehmen, damit der persönliche Kontakt zu Lösungsvorschlägen führen kann.

Kurz erwähnt - der iPlanet Enterprise Web Server:

Wenn ein robuster Webserver benötigt wird, dann ist dieser, von der Sun-Netscape-Alliance angebotene, Webserver eine sehr gute Wahl, zumal Netscape besonderen Wert auf ausgezeichneten Support und Dokumentation legt. Er wird als Highend-Lösung angepriesen, was der Preis von 3050 DM (Quelle: Infinigate GmbH¹²) verdeutlicht und ihm Platz 3 mit „nur“ ca. 7 % bei der Webserverzählung von

Netcraft¹³ verschafft. Für unsere Zwecke ist dieser Server somit völlig überdimensioniert.

2.2.3 Vergleich der Datenbank-Software

Auch wenn eine Vorauswahl bezüglich der Betriebssysteme Microsoft Windows und Unix/Linux getroffen wird, bleibt eine Vielzahl von möglichen Kandidaten übrig. Bei dem Kriterium der Verbreitung dominieren Microsoft Access, Microsoft SQL und der Oracle-SQL-Server. Im Bereich „Erweiterbarkeit für wenig Geld“ punktet natürlich wieder die GNU General Public License und verhilft mSQL oder MySQL zu einer möglichen Ernennung. Da unser Datenbankserver als SoHo-Lösung mit durchschnittlich ca. 100 Hits pro Woche dienen soll, werden die wohl bekannteste Datenbank und die Lösung für den möglichen Linux-Server in die engere Auswahl genommen. Somit sind also Microsoft Access und MySQL die Datenbanken, welche wir im weiteren Verlauf genauer betrachten werden.

Microsoft hat mit dem Office-Paket eine regelrechte Monopolstellung im Bereich Büroapplikationen und –kommunikation erreicht und jedermann nicht nur zu einer einfachen Benutzung von Textverarbeitung-, Tabellenkalkulations- und Publishing-programmen verholfen, sondern auch frühe Kundenbindung in dem mächtigen Bereich der Datenbankbenutzung geschaffen. Mit Microsoft Access ist jeder auf relativ einfache Art und Weise in der Lage eine Datenbank aufzubauen, diese mit Datensätzen zu füllen, Ein- und Ausgabemasken zu erstellen und letztendlich zu administrieren. Auch hier gestaltet sich die Preispolitik von Microsoft wieder sehr verwirrend und man hat unter anderem die Wahl zwischen der Einzelapplikation oder einem Bundle, dem sogenannten Office-Paket, erhältlich als Developer, Premium oder Professional. Für diese gängigen vier Produkte werden von der Firma BtB OHG¹⁴ folgende Preise angegeben:

- MS Access 2000	799 DM
- MS Office 2000 Developer	2389 DM
- MS Office 2000 Premium	1315 DM
- MS Office 2000 Professional	1395 DM

Bei der Neuentwicklung von Datenbanken wird Access oft nur als Entwurfshilfe benutzt und dann später durch einen SQL-Server ergänzt. Dies geschieht aufgrund der größten Schwachstelle: dem linearen Suchverfahren bei Anfragen. Bei Suchanfragen in Access wird immer die gesamte Datenbankdatei durchsucht, was natürlich bei größeren Datenmengen die Rechnerperformance stark in Mitleidenschaft zieht und Anfragen unnötig verlängert. Der oben erwähnte SQL-Server von Microsoft arbeitet, wie die meisten anderen DBMS auch, bei einem Suchvorgang mit der Baumstruktur, welches ressourcenschonender und auch schneller zu einem Suchergebnis führt. Bei einer einzelnen Arbeitstation ist die ungewöhnliche Suchart von Access nicht unbedingt ein Hindernis, aber wenn mehrere User gleichzeitig auf eine im Netz liegende Datenbankdatei zugreifen wollen, wird zusätzlich zum Performanceeinbruch des Servers die Netzlast auf unnötige Weise erhöht. Des Weiteren ist Access auch bei dem Versuch Festplattenspeicher einzusparen nicht sonderlich hilfreich, da die Datendatei neben den eigentlichen Datensätzen auch viel Ballast, wie zum Beispiel alle Eingabeformulare und Masken, enthält und somit einen zu großen Overhead im Gegensatz zu vergleichbaren Datenbankdateien mit sich führt.

Von Vorteil ist Access' Kompatibilität zu Microsoft ODBC-Treibern (Open Database Connectivity) und zu Sun JDBC-Anbindungen (Java Database Connectivity), da somit eine uneingeschränkte Migration in SQL (Structured Query Language) gewährleistet ist. SQL ist eine Datenbanksprache zur Datenabfrage und Datenmanipulation in relationalen Datenbanken, bei der eine SQL-Engine eine SQL-Anweisung zunächst analysiert und optimiert, bevor sie ausgeführt wird. Die oben genannten Techniken und Kompatibilitäten werden weltweit als Standard anerkannt, aber die zumeist nur rudimentäre Unterstützung von Unix-Systemen zeigt deutlich den Kampf Microsofts gegen Anbieter aus dem Open Source-Bereich.

Diese kontern mit dem überaus leistungsstarken und sehr schnellen MySQL-DBMS, welche sowohl unter Unix als auch unter Windows einsetzbar ist. Bei der Version für Unix-Systeme greift wieder die GPL, wohingegen für die Windowsversion eine Lizenzgebühr von 170 € (Quelle: VoPage¹⁵) entrichtet werden muß. Laut der Zeitschrift i'X¹⁶ ist MySQL wesentlich schneller als andere SQL-Datenbanken, da

Präprozessoren (zum Beispiel Transaktionsmanager und Logmanager) entfallen und kaum Overhead vorhanden ist. Bei MySQL ist die Performance nur von der Hardware abhängig und die neuesten Versionen reizen sie bis zur Leistungsgrenze aus, während hingegen alle anderen Datenbanken die CPU erheblich belasten, das Festplatteninterface aber nie voll ausnutzen. Benchmark-Tests¹⁷ der Firma TPC (Transaction Processing Council) zeigten deutliche Vorteile bei Lesezugriffen gegenüber Oracle, Sybase oder Informix und auch bei den Schreibzugriffen gehört MySQL zum oberen Drittel. Hier sind Nachteile gegenüber DBMS von namhafteren Datenbankherstellern aufgrund einer spezifischen Locking-Art vorhanden. Im Vergleich zu anderen SQL-Datenbanksystemen gilt dieses DBMS, von der in Schweden ansässigen TCX DataKonsult AB, als ein gelungener Kompromiß zwischen Funktionsvielfalt und Arbeitstempo. Auf referenzielle Integrität oder Transaktionen muß man zwar verzichten, doch in kleineren oder in hauptsächlich die Datenbank lesenden Projekten stört das nicht zwangsläufig. Für sichere Update-Vorgänge über mehrere Tabellen bei einem Mehrbenutzerbetrieb muß man schlichtweg die betroffenen Tabellen locken (aus dem Englischen „to lock“ = sperren). Komfort in der Art, wie ihn Desktop-Datenbanken normalerweise liefern, bietet MySQL nicht, da es sich lediglich um einen Server handelt. Da jedoch diverse Zusätze existieren, etwa ODBC- und JDBC-Treiber, spricht nichts dagegen, zum Beispiel Access als Frontend zu verwenden. Ansonsten können für eine leichtere Bearbeitung auch wieder diverse GUIs installiert und verwendet werden, was in unserem Fall aber überflüssig ist, da wir grundsätzlich eine Administrierung und Bearbeitung per Web-Frontend über eine Skriptsprache vorsehen.

2.2.4 Vergleich der Skriptsprachen

Der Hauptbestandteil bei Dynamic HTML (DHTML), also dynamischen Webseiten, ist die Skriptsprache, welche aus vorhandenem HTML-Code und den empfangenen Informationen des Besuchers (zum Beispiel Cookies oder Formulardaten) die gewünschten Abfragen bearbeitet und dem Browser ein lesbares Dokument zurücksendet. Es wird zwischen zwei Skripttypen unterschieden: der client- und der serverseitigen Skriptausführung. Die Bezeichnungen lassen erkennen, daß es sich bei dem ersten Typ um eine Zusammenstellung der Webseite durch den Client (in der Regel der Browser) und bei dem zweiten um eine Zusammenstellung durch den Webserver

handelt. Beide Ansätze beinhalten natürlich Vor- und Nachteile, die von bestimmten Skriptsprachen zum Teil kompensiert werden. Letztendlich bestimmen das Zusammenspiel zwischen Hard- und Software, sowie die Einsatzgebiete, welche Sprache optimal für das Gesamtsystem ist. Bei der clientseitigen Entwicklung dominiert JavaScript, was durch die hohen Benutzung belegt wird. Schwieriger wird die Sache auf Seiten der Serverprogrammierung, wobei Perl hier nach wie vor eine ganz zentrale Rolle spielt, da bereits eine Unmenge an spezifischen Skripten im WWW zum Download bereitgestellt werden, die man einfach in eine Webseite integrieren kann. Sollen aufwendige Datenbanksysteme ans Internet angebunden werden, spielen ASP und PHP auf den jeweiligen Herkunftsplattformen (ASP für Windows und PHP im allgemeinen für Unix) ihre Stärken aus. Gefragt sind stabile, durchschaubare und einfach zu erlernende Systeme, die zudem beliebig skalierbar und erweiterbar sind. Im Folgenden werden die sechs gängigsten Möglichkeiten etwas näher betrachtet:

CGI (Common Gateway Interface)

Wie bereits in Kapitel 1.2.4 beschrieben, handelt es sich hierbei um die einfachste Art der Skriptnutzung, welche aber durch das besondere Handling bei Anfragen mit erheblichen Performanceverlusten verbunden ist. Seine Flexibilität bei Verwendung einer Programmiersprache (zum Beispiel Assembler, C oder Java) und somit seiner Portierbarkeit auf andere Plattformen, verdankt diese Skriptsprache den häufigen Einsatz auf Webseiten mit DHTML.

JavaScript

Dies ist eine kompakte, objektorientierte Programmiersprache, mit der sich Client- und Serveranwendungen erstellen lassen. Unter „kompakt“ versteht man in diesem Zusammenhang, daß die Skriptsprache aus vergleichsweise wenigen, einfach einsetzbaren Befehlen besteht und so gestaltet ist, daß gerade jene Anwender, die nicht mit Programmierung vertraut sind, relativ schnell zu guten Ergebnissen gelangen. Ein weiteres, sehr wesentliches Merkmal von JavaScript ist seine Objektorientierung. Kleinigkeiten, wie zum Beispiel das Öffnen neuer Fenster mit definierten Eigenschaften oder das Anzeigen von Statusmeldungen, hat JavaScript zu einer schnellen Verbreitung und einem vielseitigen Einsatz bei der Webprogrammierung verholfen.

Eingebunden wird ein solcher JavaScript-Befehl direkt in den HTML-Code einer Webseite, abgetrennt durch sogenannte Tags. JavaScript kann sowohl für die Entwicklung von serverseitig als auch clientseitig ablaufenden Skripten verwendet werden, wobei beide Versionen den JavaScript-Kern nutzen, der Schlüsselwörter, Grammatik, Syntax, Regelwerk, das zu Grunde liegende Objektmodell, vorgegebene Objekte und Funktionen definiert. Auf der Clientseite gibt es spezielle Erweiterungen, wie beispielsweise das Window-Objekt, über das die oben angesprochene Steuerung von Fensteraktionen erfolgt. Serverseitig gibt es weitere Klassen, wie zum Beispiel die Sendmail-Klasse, die das Versenden von Mails erlaubt, oder das Datenbankobjekt, über das Verbindungen zu ODBC-kompatiblen Datenbanksystemen erzeugt werden können. Somit ist es eine optimale Skriptsprache für Web Application Server.

VBScript (Visual Basic Script)

Mit VBScript lassen sich wie mit JavaScript beispielsweise Formularberechnungen durchführen, benutzerdefinierte Homepages erstellen oder gar Online-Spiele erstellen. Es ist eine Untergruppe der Visual-Basic-Programmiersprachenfamilie und ähnlich wie bei JavaScript besteht der Quellcode eines VBScript-Programms aus ASCII-Text, welcher direkt in die HTML-Seite mit Tags eingebunden wird. Durch die fehlende Kompatibilität zu anderen Browsern als dem Internet Explorer – VBScript ist eine Entwicklung von Microsoft – wird es allgemein recht selten in der WWW-Programmierung auf Clientseite eingesetzt.

Perl (Practical Extraction and Report Language)

Diese zählt zu den wichtigsten Sprachen für die Entwicklung von interaktiven Inhalten für Webseiten. Perl ist so konzipiert, daß sich damit typische Probleme einfacher und eleganter als mit klassischen Programmiersprachen lösen lassen. Genutzt wird es insbesondere zur Datenbankmanipulation, Text- und Dokumentenverarbeitung, Datenextraktion und -reduktion, Netzwerkadministration und objektorientierten Programmierung. Positiv sind seine Erhältlichkeit für nahezu alle Betriebssysteme und die Möglichkeit der Erweiterung, da seine offene Struktur eine Koexistenz von anderen Skriptsprachen in ein und demselben Dokument und eine Benutzung von objektorientierten Bibliotheken zuläßt. Es ist eigentlich eine

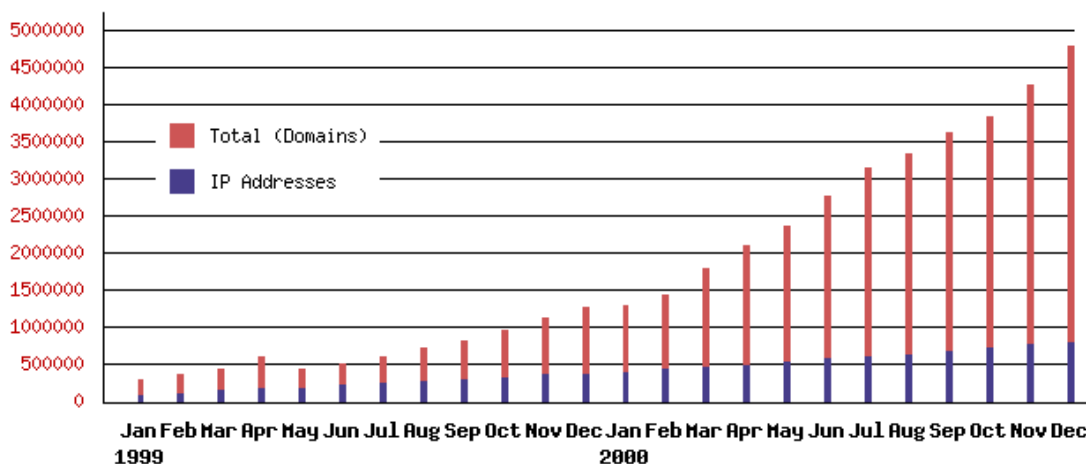
interpretierende Sprache, denn unter Perl wird ein Skript zunächst gelesen, vor der Ausführung aber in Byte-Code verwandelt und erst dann ausgeführt, wenn das ganze Skript fehlerfrei verarbeitet wurde. Somit wird ein syntaktisch falsches Perl-Programm nicht ausgeführt und der kompilierte Byte-Code läuft mit höherer Geschwindigkeit. Bei der Verarbeitung von wiederkehrenden Problemen können sogenannte Perl-Module eingesetzt werden, welche zum größten Teil bereits als fertige Module im Internet zu finden sind.

ASP (Active Server Pages)

Vorab soll darauf hingewiesen werden, daß ASP eigentlich keine Programmiersprache ist, sondern lediglich das Ausführen von Skripten auf der Serverseite erlaubt (Interpreter). In der Praxis wird ASP zusammen mit VBScript oder JScript (die JavaScript-Variante von Microsoft) eingesetzt. Es ist ein fester Bestandteil des Internet Information Servers und sofort nach der Installation des Webservers einsetzbar. Lediglich das Ausführen von Skripten muß für die jeweilige Website erlaubt sein. Wie bei JavaScript findet man auch hier eine symmetrische Architektur, das heißt, daß sowohl der Client als auch der Server eine aktive Rolle bei der Bearbeitung von Skripten haben können und durch Scripting Services auf beiden Seiten wird die Kommunikation zwischen den Objekten sichergestellt. Als Scriptsprachen stehen insbesondere VB- und JavaScript zur Verfügung, wobei in der Praxis die Entwickler überwiegend auf JavaScript setzen. Für die Entwicklung von Serverkomponenten kann jede beliebige Programmiersprache verwendet werden, beispielsweise C++, Java oder auch Visual Basic. Eine der zentralen Einschränkungen von ASP ist die Bindung an den Microsoft Webserver IIS. Sogenannte Adapter, die einen Einsatz auch mit dem Apache- oder dem Netscape-Webserver ermöglichen, sind im allgemeinen recht teuer (Beispiel: Chili!Soft ASP kostet 750 \$ für Linux, Quelle: Chili!Soft Inc.¹⁸). Somit ist dies eine durchdachte, robuste und leicht erweiterbare Technik, wenn als Grundlage die Serverprodukte der Firma Microsoft vorhanden sind. Durch die Verwendung von bereits existierenden Skriptsprachen fällt die Einarbeitungszeit äußerst kurz aus und da ASP-Seiten kompiliert und im Webserver gecached werden, ist die Performance sehr gut.

PHP (PHP Hypertext Preprocessor)

Dieses Produkt ist ein gutes Beispiel dafür, wie schnell sich Open Source-Projekte im Internet durchsetzen, denn in den vergangenen beiden Jahren hat sich PHP einen festen Platz unter den Skriptsprachen erkämpft. Es erlaubt die schnelle Entwicklung von dynamischen Webseiten und die Syntax ist C-ähnlich, teilweise hat man sich auch bei Perl bedient. Die Stärken dieser Skriptsprache liegen aber in dem Bereich der Datenbankanbindung, da es die inzwischen wohl wichtigste Webdatenbank My-SQL, aber auch Oracle, Microsoft SQL Server und Sybase-Datenbanken unterstützt. Selbst der Zugriff auf ODBC-Datenbanken ist möglich, so daß beispielsweise Microsoft Access in eine Website eingebunden werden kann. Insbesondere durch den Nulltarif der Webserver-Komplettlösung LAMP (weitere Informationen folgen in Kapitel 2.2.5) verzeichnet PHP beachtliche Wachstumsraten, so stieg die Nutzung laut Netcraft¹⁹ von bereits 1,4 Millionen Domains im Februar 2000 auf fast 5 Millionen Domains bis zum Dezember 2000.



Quelle: <http://uptime.netcraft.com/up/news>

Abb. 6 – Wachstumsrate der Benutzung von PHP

Technisch gesehen handelt es sich bei PHP um einen Präprozessor (Compilerbestandteil, welcher zuerst die Quelltexte bearbeitet) mit integrierter Skriptsprache. Es ist deshalb unter jedem Webserver einfach wie ein CGI-Skript einsetzbar. Für den Apache Webserver gibt es sogar ein eigenes PHP-Modul, mit dem das Skript im gleichen Prozeßraum wie der Webserver läuft und nicht bei jedem Aufruf neu gestartet werden muß. Eine optimale Performance ist nur bei dieser Installation

gegeben, was PHP zum häufigsten installierten Apache-Modul macht. Der Programmcode und HTML-Code werden in einem Dokument vermischt und es gibt keine fehlerträchtigen und in der Programmierung zeitraubenden Übergänge via CGI. Die Zweiteilung in Seite und Skript wie bei Server Side Includes entfällt, womit sich ein PHP-Skript ohne lange Initialisierung direkt in den HTML-Code einbauen läßt – ähnlich wie JavaScript. Gegenüber seinem Kollegen jedoch läuft ein PHP-Skript nur auf dem Server ab und kann auf dessen Dateien und Hilfsprogramme zugreifen. Diese Skriptsprache ist strikt auf die Arbeit mit HTML-Servern ausgelegt, so reicht zum Beispiel ein einfacher Befehl, um Cookies zu setzen, ein weiterer Befehl sendet ohne Umschweife eine Mail von der Homepage ab. Einschränkungen bei der Version 3 wurden mit PHP 4 fast gänzlich behoben und durch eine komplett neu programmierte Engine wurde nicht nur die Performance erheblich verbessert, sondern allgemein für eine bessere Abstraktion zwischen Webserver und PHP gesorgt (Programmierung auf Session- und Applikationsebene). Mit dem ebenfalls integrierten Session-Handling und der Erweiterungsmöglichkeit über beliebige Java-Objekte ist PHP seinen Konkurrenten mit der neuen Version kaum noch unterlegen.

Anmerkung: Da es sich in fast allen Fällen um Skriptsprachen handelt, die auf Befehlsebene und nicht als eigenständiges Programm in eine Webseite eingebunden werden, entfallen Kosten für ein Programmpaket. Lediglich Erweiterungen, graphische Benutzeroberflächen oder eine Entwicklungsumgebung sind kostenpflichtig, sollen aber als solche in die Skriptsprachenbetrachtung nicht mit einbezogen werden. Nur auf die offensichtliche Feindseeligkeit zwischen Microsoft-Sympathisanten und deren Gegnern möchten wir an dieser Stelle noch einmal explizit hinweisen, denn die Erstellung von Webseiten sollte eigentlich unter höchsten Kompatibilitätsaspekten geschehen, da ein optimales Betrachtungsergebnis für den Besucher das Ziel eines jeden HTML-Programmierers ist, egal ob Internet Explorer, Netscape, Opera, Microsoft Betriebssystem, Apple/MAC, Unix, Linux, BeOS oder andere Komponenten verwendet werden. In den Anfängen des WWW stand dieser Gedanke noch im Vordergrund, doch mit dem Beginn der graphischen Entwicklung von Webseiten wurden Mechanismen wie zum Beispiel ActiveX und Flash-Erweiterungen zum Kriterium dafür, welches Betriebssystem oder Browser die besseren Ergebnisse liefert.

2.2.5 Vergleich der möglichen Kombinationen

Mit den vorangegangenen Betrachtungen der möglichen Komponenten für den Datenbankserver wurde bereits eine Vorauswahl bezüglich des Betriebssystems getroffen. Jetzt fehlt nur noch die Zusammenstellung der vier maßgeblich beteiligten Softwarepakete und die erneute Betrachtung dieser Kombination.

Bei den Betriebssystemen dominieren eindeutig Microsoft und Linux. Auch die Frage nach dem Webserver und der Datenbank wird durch jeweils zwei Kandidaten beantwortet. Nach Einbeziehung des Einsatzgebietes – der Anbindung einer Datenbank – kristallisieren sich Perl und PHP als würdige Vertreter der vierten Softwaregruppe heraus. Dementsprechend lassen sich folgende Kombinationen zusammenstellen:

LAMP: Linux, Apache, MySQL, Perl/PHP

WAMP: Windows, Apache, MySQL, Perl/PHP

WIMP: Windows, Internet Information Server, MySQL, Perl/PHP

WIAP: Windows, Internet Information Server, Access, Perl/PHP

Auf die Vorzüge und Nachteile der Finalisten ist bereits in den Vorbetrachtungen hingewiesen worden, doch in den bisherigen Untersuchungen wurde stets besonderen Wert auf die Prüfung der Wirtschaftlichkeit gelegt. Somit sollen auch bei der endgültigen Entscheidung für eine der Kombilösungen die Anschaffungs- und Folgekosten (Hard-/Softwarepreise, Schulungen) die ausschlaggebende Stimme haben. Nur bei der Auswahl der Skriptsprache haben wir uns von unseren Eindrücken bei den Vorbereitungen für diese Arbeit leiten lassen und unsere durch die Fachhochschule vermittelten Kenntnisse in C bzw. C++ als Grundlage für die Entscheidung genommen.

Schließlich favorisieren wir die Kombination Linux, Apache, MySQL und PHP, da die Hardwareanforderungen zu einem Windowssystem vergleichsweise geringer ausfallen und die GNU General Public License die Softwarekosten auf Null schrumpfen läßt. Alle erforderlichen Programmteile und Module sind kostenlos aus dem Internet ladbar und auch für Hilfestellungen²⁰ – im Linux-Jargon How-To's und

Manpages genannt – wird ausreichend von der stetig wachsenden Linux-Fangemeinde gesorgt. Außerdem genießt das LAMP-Projekt einen großen Verbreitungsgrad in der heutigen Internetwelt, was natürlich auch aus der Dominanz des Apache als Webserver resultiert – unter den Netcraft-Top 50 der Uptime-Systeme (Uptime = Zeitspanne zum letzten Reboot) sind alleine 47 Apache Webserver zu finden²¹.

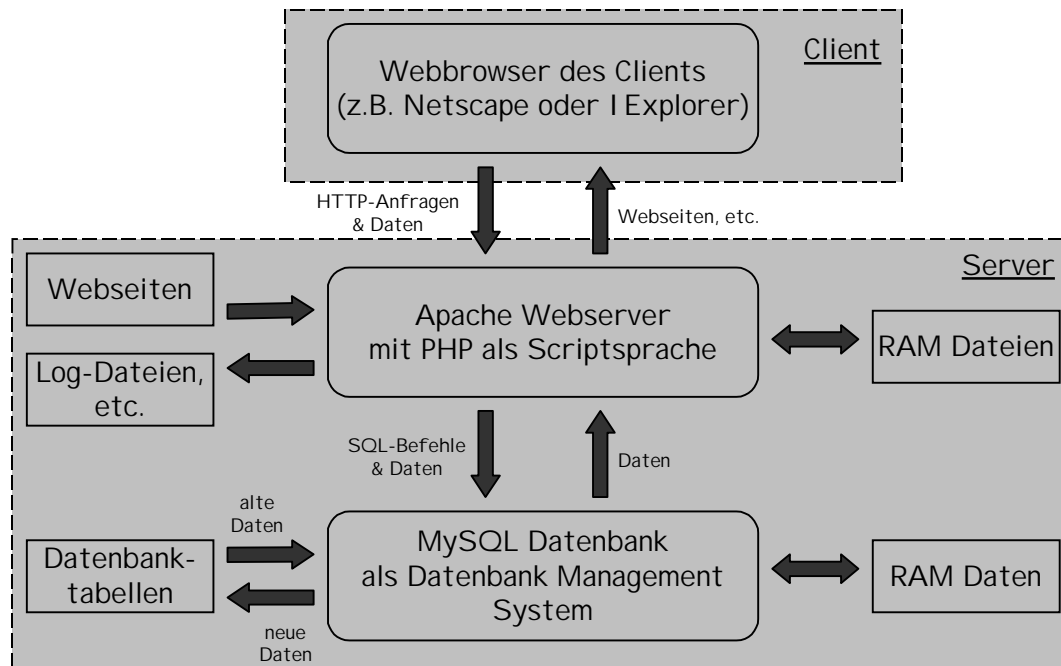


Abb. 7 – Arbeitsweise eines LAMP-Servers

Insgesamt hat sich LAMP als äußerst zuverlässig, robust und sicher erwiesen, wobei die Ansprüche an die Datensicherheit bei einer Datenbank besonders hohe Aufmerksamkeit verdient und letztendlich den Einsatz von bestimmten Softwareprodukten entscheidend beeinflussen kann. Microsoft zum Beispiel hat sich in jüngster Zeit nicht unbedingt als unüberwindbar bezüglich seiner Anwendungen herausgestellt.

2.3 Hardwareanforderungen

Nach Festlegung der zu benutzenden Softwarekomponenten kann nun mit der Dimensionierung des PCs begonnen werden. Das Betriebssystem und die Datenbank sind in erster Linie die richtungsweisenden Bestandteile und somit liegt die Mindest-

anforderung an die CPU bei 233 MHz und an den Speicher bei 128 MB. Wohl bemerkt, dies sind die unteren Grenzen für einen reibungslosen Betrieb des Servers. Empfehlen würden wir eine Prozessorleistung von ca. 350 MHz und einen Gesamtspeicher von 256 MB bei einer Taktung auf 100 MHz – je nach Art und Einstellmöglichkeit der Hardware können wiederum Performanceänderungen auftreten, die Geschwindigkeitseinbußen zur Folge haben (66 oder 100 MHz-Taktung, AMD / PII oder PIII-Prozessoren, FPM / EDO oder SDRAM). Bei der Bemessung des Festplattenspeichers sollte besonderer Augenmerk auf zukünftige Erweiterungen gerichtet werden, zum Beispiel das zusätzliche Abspeichern der Diplomdokumente zum jeweiligen Datensatz, und im Zuge der doch wachsenden Gigabyte-Größen zu einem kleinen Preis empfehlen wir die Benutzung einer 10 GB-Festplatte nur für die Datenbank. Optimal wäre die Benutzung einer zweiten, identischen Festplatte für eine Spiegelung der Daten, um bei einem Hardwaredefekt dem Verlust der gesammelten Daten entgegenzuwirken. Letztendlich tendieren wir somit zu einem Einbau von drei IDE-Festplatten – einer Systemplatte mit ca. 6 GB und die zwei Daten-/Spiegelungsplatten mit je 10 GB. Die restlichen Komponenten können im direkten Vergleich zu heutigen Standard-PC-Systemen unterdimensioniert ausfallen oder sogar ganz weggelassen werden, da der geplante Datenbankserver ohne Eingriff durch Dritte auskommt. Es könnte zum Beispiel auf einen Zukauf von Tastatur, Maus und Monitor verzichtet werden, denn für den Fall, daß doch Arbeiten am Server anfallen, kann die Administrationsfähigkeit mittels Konsolen von einer anderen Arbeitsstation ausgenutzt werden oder man borgt sich kurzfristig oben genannte Peripherie von einem anderen PC. Ansonsten fehlen nur noch ein CD-Rom-Laufwerk (eventuell sollte auch an den Kauf eines DVD-Roms gedacht werden, da manche Linux-Distributionen aus Platzgründen auf DVD ausgeliefert werden), ein Floppy-Laufwerk, eine Netzwerkkarte (ob BNC oder Twisted Pair bzw. 10 MBit oder 100 MBit hängt vom vorhandenen Netzwerk ab) und eine Grafikkarte, welche durchaus auch mit nur 4 MB bestückt sein kann. Bei dem Gehäuse zählt primär nur die ausreichende Kühlung (ggf. empfiehlt sich der Einbau eines Extra-Lüfters), weitere Anforderungen ergeben sich aus der Bauart des Mainboardes (Formfaktor) und der Anzahl der Laufwerke (Einbaumöglichkeit).

2.4 Erstellung eines Konzeptpapiers

Für einen reibungslosen Ablauf bei der Implementierung werden die bisher genannten Anforderungen und Ziele in Form eines Konzeptpapiers zusammengetragen, welches durch die Einbindung weiterer Aspekte letztendlich zu einer Richtlinie für das weitere Vorgehen wird. Abweichungen oder Verbesserungsvorschläge können zwar im Nachhinein jederzeit mit einfließen – sollen auf keinen Fall unberücksichtigt bleiben – müssen aber stets mit der Grundlage und der Zielsetzung vereinbar sein und dürfen den weiteren Ablauf der Implementierung nicht stören oder sogar aufhalten.

Vorraussetzungen

Die Ernennung und Bekanntmachung eines Ansprechpartners hilft bei der Entscheidungsfindung und das Einfließen zusätzlicher Kriterien, wie zum Beispiel Verbesserungsvorschläge. Dieser ist für die Einhaltung der Zielsetzung und der Konzeptionierung verantwortlich und koordiniert den gesamten Ablauf von der Projektierung bis zur Fertigstellung. Somit ist gewährleistet, daß Mißverständnisse, hervorgerufen durch fehlende Absprache oder Unklarheiten, verhindert oder zumindest auf ein Minimum reduziert werden. Regelmäßige Treffen und häufige Gespräche über den aktuellen Fortschritt sind für die Einhaltung von Vorgaben unerlässlich.

hier:

Prof. Dr. – Ing. Jürgen Hetsch, Andre Blase und Carsten Sander

Eine Motivationsgrundlage und eine Problemanalyse wurden bereits auf den vorangehenden Seiten erarbeitet und sollen an dieser Stelle nicht noch einmal im Detail aufgeführt werden.

hier:

Zentralisierung von studentenspezifischen Daten zwecks komfortabler Abfragemöglichkeit durch Fachhochschulmitarbeiter und einer permanenten Aktualisierung durch die Studenten selbst. Ein Passus auf der Eingabemaske wird Studenten auf die grundsätzliche Einhaltung des Datenschutzes hinweisen, zum Beispiel die Datenschutzerklärung des Arbeitsvertrages. Durch die freiwillige Eingabe von Privatdaten entfallen gesonderte Mechanismen und Genehmigungen für die Herausgabe innerhalb der Fachhochschule, da der Student selbst festlegen kann, ob und in welchem Maße seine Daten in diese Zentraldatei aufgenommen werden.

Eine Aufstellung der Aufgaben des fertiggestellten Projektes verschafft einen Überblick über die bereitzustellende Hard-/Software, Dienste und andere Ressourcen, wie zum Beispiel Räumlichkeiten oder menschliche Hilfe bei der Nachbetreuung, Administration genannt.

hier:

1. Sammlung von Studentendaten auf einem zentralen Server (Name, Anschrift, Kontaktmöglichkeiten, Themen der Abschlusarbeiten).
2. Erreichbarkeit des Servers aus dem Intranet und dem Internet heraus.
3. Visualisierung von Datenbankabfragen durch einen standardisierten Viewer (zum Beispiel Browser).
4. Aktualisierung der Datensätze durch den jeweiligen Studenten selbst.
5. Abfragemöglichkeiten sollen benutzerspezifisch sein, d.h. nur autorisierte Mitarbeiter der Fachhochschule Dortmund sollen den vollen Datensatz einsehen können, externe Besucher aber nur den eigenen oder lediglich eine Auflistung der Abschlusarbeiten mit Verweis auf eine Kontaktierung des jeweiligen Betreuers.
6. In Abständen sollen eMails an die Inhaber der Datensätze verschickt werden, welche somit zu einer Aktualisierung der Adreßdaten beitragen.
7. Gewährleistung von Datensicherheit im Falle eines Ausfalls und Einhaltung des Datenschutzgesetzes durch Zugriffsbeschränkungen.

Die Festlegung eines Zeitplans ermöglicht einen permanenten Abgleich zwischen Soll- und Ist-Zustand, so daß sich Verzögerungen frühzeitig erkennen lassen und gegebenenfalls Maßnahmen eingeleitet werden können.

hier:

Die Projektplanung soll im Monat März 2001 abgeschlossen sein, mit einer Fertigstellung der Implementierung wird im Monat Juni 2001 gerechnet.

Planung

Zunächst werden mögliche Werkzeuge für die Realisierung der Projektziele gesammelt und einer gründlichen Untersuchung unterzogen, damit Schwachstellen, Inkompatibilitäten und Fehlerquellen schon im Vorfeld aufgedeckt und für den Projekteinsatz gewertet werden. Hierzu zählen die Betrachtungen und Vergleiche in Kapitel 1 und Kapitel 2. Bei Softwareprodukten spielt auch die Berücksichtigung der Versionsnummer eine große Rolle, da neuere Versionen meist nützliche Funktionserweiterungen beinhalten, die ggf. die Einschränkungen der Vorgänger beheben und somit einen Einsatz wieder rechtfertigen könnten. Weiteres Augenmerk sollte auf zukünftige Entwicklungen und somit einer Erweiterungsmöglichkeit des Projektes geworfen werden, damit spätere Veränderungen nicht zu einer Neuimplementierung des gesamten Systems führen.

hier:

1. Aufstellung der Softwarekomponenten, da diese den größten Einfluß bei der Auswahl der restlichen Komponenten (zum Beispiel Hardware) hat:
 - Betriebssystem: SuSE Linux 7.0 Professional, Kernel 2.2.16
 - Befehlsinterpreter: Perl 5.00503
 - Webserver: Apache 1.3.17
 - DBMS: MySQL 3.23.32
 - Skriptsprache: PHP 4.0.4, Patchlevel 1
 - Mailprogramm: Sendmail 8.10.2-24
 - Software für die Spiegelung: Bestandteil des Linux-Kernels

2. Aufstellung der Hardwarekomponenten unter Berücksichtigung der vorhandenen Infrastruktur:
 - Standard-PC mit den in Kapitel 2.3 angesprochenen Mindestanforderungen (eine komplette Auflistung der tatsächlich implementierten Hardware wird im Rahmen der Diplomarbeit vorgenommen)
3. Auflistung von möglichen Erweiterungen unter Verwendung der obigen Aufstellungen und minimalen Zukauf von Hard- bzw. Software:
 - Einrichtung eines Backupsystems (Bandlaufwerk oder Imageerstellung)
 - geschriebene Abschlußarbeiten sind aus dem Datensatz heraus abrufbar
 - Benutzung der Datenbank auch schon während des Studiums (Praktika)
 - Ausweitung des Projektes auf FH-Ebene mit Anbindung an die Informix-Datenbank der Fachhochschule mittels ODBC

Auf Basis der bis hier festgelegten Komponenten und Anforderungen erfolgt eine Kostenrechnung, die Aufschluß über die Rentabilität geben soll. Sollten die einmaligen und laufenden Kosten den Nutzenfaktor übersteigen, wäre an dieser Stelle eine nochmalige Planung unter Berücksichtigung der bereits erworbenen Erkenntnissen und relevanten Fakten oder gar eine Einstellung des Projektes zu empfehlen.

hier:

Durch die Softwarelösung LAMP entstehen lediglich Kosten durch die Anschaffung der Linux-Distribution SuSE, die restlichen Module fallen in die GPL-Regelung und sind somit kostenlos.

Durch den Einsatz von Linux werden die Hardwarekosten, im Vergleich zu einem Microsoft-Systems, gering gehalten und auch ältere, eventuell schon vorhandene Hardwarekomponenten können im Rahmen einer Kostenreduzierung eingesetzt werden.

Eine genaue Bezifferung der Gesamtkosten kann erst am Tage des Kaufes erfolgen, da Preise von PC-Hardware, insbesondere CPU und Speicher, täglichen Schwankungen ausgesetzt sind. Zum jetzigen Zeitpunkt sieht eine ungefähre Auf-

stellung der einmaligen Kosten wie folgt aus:

- Software: 129,- DM
- Hardware: 1.500,- DM (Standard-PC ohne Monitor)
500,- DM (Monitor)

Wie bereits angesprochen, beschränken sich die monatlichen Kosten auf den zusätzlichen Stromverbrauch eines weiteren Standard-PCs und eventuell anfallende Reparaturkosten.

Kapitel 3

Implementierung

Wenn alle Vergleiche, Kostenrechnungen und Planungen abgeschlossen sind, kann eine Umsetzung des Geplanten erfolgen – die sogenannte Implementierung. In diesem Kapitel möchten wir die von uns ausgewählten Softwarekomponenten genauer beschreiben, da im Rahmen der Diplomarbeit die eigentliche Implementierung erfolgen wird und in der schriftlichen Ausarbeitung Informationen bezüglich der Installationen, Einstellungen und die Source-Codes der von uns erstellten Skripte aufgeführt werden.

3.1 SuSE Linux

Linux hat sich in der letzten Zeit durch den vermehrten Einsatz im privaten Umfeld – als Alternative zu Microsoft-Produkten – rapide zu einem weit verbreiteten Betriebssystem entwickelt und somit einem älteren, noch gebräuchlicheren, zu neuem Schwung verholfen: Unix. Streng nach Definition und Intention bildet Linux aber nur den Kern des Betriebssystems, den sogenannten Kernel, der die grundlegenden Funktionen für die Prozeßsteuerung, den virtuellen Speicher, die Dateiverwaltung und die Ein-/Ausgabeoperationen bereitstellt. Also stellt Linux selbst nur die unterste Ebene des Betriebssystems dar. Im Volksmund wird mit dem Begriff „Linux“ aber das komplette System bezeichnet, den Kernel zusammen mit den vielen Anwendungen, die unter ihm laufen, wie eine vollständige Entwicklungs- und Arbeitsumgebung mit Compilern, Editoren, grafischen Benutzerschnittstellen, Textverarbeitung, Spielen, und vielem mehr.

Der Linux-Kernel ist eine Weiterentwicklung des Netzwerk- und Multitasking-Systems Unix, welche von Linus Torvalds 1991 an der Universität von Helsinki, Finnland, vorgenommen wurde. Hierbei legt er auch heute noch besonderen Wert auf die fortschreitende Entwicklung in Zusammenarbeit mit vielen interessierten Programmierern im Internet, wodurch letztendlich der Benutzer gleichzeitig zum „Beta-Tester“ und „Softwareverbesserer“ werden kann. Jeder kann unter Berücksichtigung bestimmter Konventionen, festgehalten in der GPL, den öffentlich zugänglichen Programmcode (Grundprinzip des Open Source) verändern und somit verbessern. Im allgemeinen kann man sagen, daß es Tausende von Programmversionen gibt, doch letztendlich definiert das Entwicklerteam um Linus Torvalds den aktuellsten Kernel, bei dem Fortschritt und Entwicklungsstufe in eindeutiger Weise beschrieben wird:

- Stabile Kernel, deren Funktionstüchtigkeit nachgewiesen ist, erhalten eine gerade Ziffer nach der Hauptversionsnummer (Beispiel: 2.0)
- Experimental-Kernel, in die neue Funktionen eingebaut werden und an denen gerade gearbeitet wird, erhalten ungerade Ziffern (Beispiel: 2.1)
- An die Unterversion wird dann noch eine weitere ein- oder mehrstellige Zahl angehängt, die den aktuellen Entwicklungsstand anzeigt (Beispiel: 2.0.29)

Zwar hat Linus Torvalds die Rechte am Basis-Kernel und bestimmt noch heute die Entwicklung, aber erst die Zusammenarbeit der Linux-Fangemeinde verschaffte diesem Betriebssystem die Verbreitung, die es heute genießt. Das macht Linux zu einer echten Open Source-Software.

Im Laufe dieser Ausarbeitung wurde schon oft die GPL erwähnt, deshalb soll an dieser Stelle eine kleine Erklärung dieses Systems und dessen Intention erfolgen. Da Linux aus sehr vielen Komponenten mit unterschiedlichem Copyright besteht, läßt sich keine einheitliche Aussage über den rechtlichen Status machen. Üblicherweise behalten sich die Autoren die Rechte an ihren Werken vor, erlauben aber die Nutzung unter den Auflagen der General Public License²² (GPL), welche im Wesentlichen fünf Aussagen trifft:

1. die Software darf nur unter gleichen Bedingungen weitergegeben werden
2. die Software beinhaltet keine Garantien, jegliche Haftung ist ausgeschlossen
3. es dürfen bei Verkauf lediglich Kosten für den Vertrieb berechnet werden
4. es werden keine Lizenz-, Registrierungs- oder Benutzungsgebühren fällig
5. insbesondere müssen stets die Quelltexte verfügbar sein

Die GPL wurde von der Free Software Foundation (FSF) entworfen und dient als Copyright für deren GNU-Programme (GNU = GNU is not Unix), scherzhaft auch als Copyleft bezeichnet. Nicht alle Linux-Programme unterliegen der GPL und der Gebrauch des Titels „Freeware“ und „Shareware“ hat unter Linux fast keine Bedeutung.

Weil Linux freie Software ist, gibt es keine einzelne Organisation oder Instanz, die für die Freigabe und Verteilung der Software zuständig ist. Deshalb steht es jedermann frei, Linux-Software zusammenzustellen und zu verteilen, solange die Einschränkungen durch die GPL beachtet werden. Als Folge davon gibt es viele Distributionen, die man sich per anonymem FTP und im Versand- oder Buchhandel besorgen kann. Als die weltweit verbreitetste wird Red Hat genannt, auf Deutschland bezogen dominiert SuSE aufgrund seiner doch recht einfachen Installation und Konfiguration – insbesondere durch die Weiterentwicklung und grundlegende Verbesserung des YaST (Yet another Setup Tool), welches in der zur Zeit verwendeten Version 2 auch einem Neuling den relativ einfachen Einstieg in die Linux-Welt ermöglicht. Selbst auf eine grafische Oberfläche wie Windows sie bietet, braucht man nicht zu verzichten, denn diverse Module, besonders die neueste Version des KDE (Kool Desktop Environment), bilden eine hervorragende Grundlage für die einfache Benutzung und Administration eines Linux-Systems, so wie man es von Microsoft Windows gewohnt ist.

Viele beenden hier auch schon die Gemeinsamkeiten mit Windows, denn dessen Schwächen sind die Stärken von Linux. So gehört zum Beispiel die Netzwerkfähigkeit und das hohe Maß an Sicherheit eindeutig zu den Vorteilen, sowie eine hohe Stabilität, „echte“ Multitasking- und Multiuserfähigkeit, das umfangreiche Vorhandensein von Netzdiensten bereits bei der Grundinstallation und die

Hardwareunabhängigkeit, welche sich in der geringen Hardwareanforderung wieder spiegelt. Man kann die Abstammung von Unix – einem der meist eingesetzten und sichersten Netzwerksystemen – weder verbergen noch leugnen, denn fast alles, was Unix zu dem hohen Stellenwert in dem Bereich der Netzwerktechnologie verschafft hat, wurde auch in die „Home-Version für Jedermann“ implementiert. Die mächtigen Features offenbaren sich allerdings meist nur dem wirklich erfahrenen Benutzer, da hierfür in der Regel eine Bedienung der recht komplizierten und kryptischen Eingabekonsole oder das Verändern von Konfigurationsdateien, wenn nicht sogar das Kompilieren eines neuen Kernels, notwendig wird. Auch die Verwaltung der User-Rechte gestaltet sich recht komplex, da hier streng zwischen Lesen, Schreiben und Ausführen bei Zugriffen von Eigentümern, Gruppenmitgliedern oder der Allgemeinheit unterschieden wird – und das für jede Datei unterschiedlich definierbar. Doch bei Fragen und Problemen bietet die Linux-Gemeinde ein umfangreiches Hilfeprogramm, denn wenn die Möglichkeiten des Distributionsupportes (SuSE gilt im allgemeinen als ein Paradebeispiel für guten und umfangreichen Support) ausgeschöpft sind, sind im Internet noch unzählige Hilfestellungen, die bereits oben erwähnten How-To's, Manpages und FAQs (Frequently Asked Questions), auf Webseiten von Linux-Fans, Distributoren und Entwicklern zu finden²³.

In unserem speziellen Fall profitieren wir insbesondere von dem modularen Aufbau dieses Betriebssystems, da wir unter Berücksichtigung des Kostenaufwandes auf die Installation von überflüssigen Komponenten verzichten können, um Hardwareressourcen in Form von Festplattenspeicher und Rechenleistung einzusparen. Der Modulaufbau erlaubt andererseits aber auch eine problemlose Nachinstallation benötigter Komponenten, ohne durch eine Deinstallation von gesamten Softwarepaketen die Systemintegrität zu beeinflussen.

Auf der SuSE-Homepage²⁴ heißt es:

„Die SuSE Linux AG ist der internationale Technologieführer und Anbieter von Komplettlösungen auf Basis des Open Source-Betriebssystems Linux. Diese Position ist auf SuSEs einzigartiges Linux-Know-how zurückzuführen. SuSE beschäftigt das weltweit größte Entwicklungsteam für Open Source-Software. Das Unternehmen besitzt die größte öffentlich zugängliche Supportdatenbank und die umfangreichste

Hardwaredatenbank im Linux-Markt. SuSE unterstützt mehr Hardwareplattformen als jede andere kommerziell verfügbare Linux-Distribution.“

Inwiefern diese Angaben nun der Wahrheit entsprechen, soll und kann an dieser Stelle nicht geklärt werden, aber die Entscheidung, daß wir für den geplanten Datenbankserver die SuSE-Distribution verwenden, treffen wir aufgrund des großen Verbreitungsgrades in der Studentenschaft (ermöglicht durch den Verkauf einer Studenten-Edition) und der somit hohen Wahrscheinlichkeit, daß auch in den folgenden Jahren des Betriebes Leute gefunden werden, die bei Problemen helfen oder eine weitere Ausbaustufe des Servers vornehmen können. Somit ist eine Betreuung durch Dritte langfristig gewährleistet, da eine komplexe Einarbeitung in eine neue Arbeitsumgebung, zum Beispiel durch kostenintensive Schulungen, für die meisten Nachfolgeadministratoren entfällt. Auch das Installationskonzept von SuSE besticht durch die zunehmende Vereinfachung durch Assistenten und GUIs, welche besonders von Neueinsteigern gerne benutzt werden. Dieser Distribution ist es fast perfekt gelungen, eingefleischte Windows-User auch für Linux zu begeistern und der immense Vorrat an freien Programmen und Erweiterungen in Form von downloadbaren Modulen führt zusätzlich zu einer wachsenden Attraktivität für Heimanwender und Privatpersonen. Alleine die Verbreitung der Textverarbeitung LaTeX und der Anspruch, auch innerhalb des Studium kompatibel mit anderen Studenten und Dozenten zu bleiben, konfrontiert viele Kommilitonen mit dem Gedanken dieses Betriebssystems einzusetzen.

3.2 Apache Webserver

Das dominierende Sicherheitspotential und die Zuverlässigkeit eines Unix/Linux-Systems, trug auch zu einer raschen Verbreitung des Webservers Apache bei, denn diese Tugenden sind für einen Server unerlässlich, da er ständig erreichbar sein soll und durch die offene Anbindung angreifbar für Hackerattacken ist. Aber auch das Konzept der kostenfreien Verbreitung durch die GPL-Regelung ist ein Grund für den ca. 60-prozentigen Einsatz weltweit.

Diese Zahl verdient um so mehr Bewunderung, wenn man berücksichtigt, daß mit der Entwicklung dieser Software erst Anfang 1995 begonnen wurde, als der NCSA HTTPd (HTTP daemon) bereits unangefochtenen den Spitzenplatz in der Liste der eingesetzten Webserver inne hatte. Mittlerweile hat der Urvater der Webserver diese Position an den Apache verloren und findet fast keine Anwendung mehr. Bemerkenswert ist auch die Entstehungsgeschichte, denn wie bei Linux ist der Apache eine Weiterentwicklung, die von einer handvoll Begeisterter mittels eMailkontakt, Newsgroups und Dateiaustausch entstanden ist. So wurden die Veränderungen in Form von Patches und Bugfixes gesammelt und im April 1995 mit dem NCSA HTTPd Version 1.3 kombiniert. Das war die Entstehung des ersten offiziellen Releases 0.6.2 des Apache Webservers. Der Name beschreibt noch heute seine Herkunft, denn ursprünglich wurde dieser „a patchy server“ genannt, da er aus existierendem Code und selbstgeschriebene Files wie ein „Flickwerk“ zusammengesetzt war (scherzhaft wird er auch als ein "um Fehler bereinigter (a patchy) NCSA-Server" bezeichnet). Durch die englische Aussprache und dem Gleichklang wurde aus dem „a patchy“ später dann der „Apache“.

Das Grundautorenteam beschrieb ihn damals wie folgt:

„Highly configurable, extendable, robust, fast, standards-compliant, pre-forking, efficient, constantly evolving, user motivated, user supported, collaboratively developed, well tested, user satisfying, hugely popular.“

Brian Behlendorf ist der Pressesprecher der Apache-Gruppe, welche heute die Weiterentwicklung kontrolliert, und obwohl er nie eine einzige Zeile Code geschrieben hat, gilt er als einer der führenden Köpfe des Projektes. Bereits weniger als ein Jahr nach dem Erscheinen des ersten Release im Jahr 1995 hatte Apache den NCSA-HTTPd-Server vom ersten Platz verdrängt. Auch dies ist, wie bei Linux, ein sehr gutes Beispiel für die Möglichkeiten und Vorteile der Freeware-Entwicklung im Internet.

Drei Eigenschaften heben den Apache besonders hervor:

1. Er unterstützt sowohl „name based“ als auch „IP based virtual domains“. Das bedeutet, daß jede Domain des Webserver eine eigene öffentliche IP-Adresse erhält, oder daß sich alle Domains eine IP-Adresse teilen können, da der Webserver die Verwaltung bei Zugriffen vornimmt, das sogenannte Virtual-Hosting. Die zweite Möglichkeit wird hauptsächlich im Rahmen der Kostenersparnis eingesetzt, da der Erwerb einer öffentlichen IP-Adresse mitunter einen hohen Preis haben kann.
2. Er ist in der Lage, mit Unix-Datenbankdateien zu arbeiten und auf Routinen des GNU Datenbank Managers zurückzugreifen, wodurch eine höhere Geschwindigkeit bei dem Durchsuchen der Autorisierungsdateien (Zugriffsrechte der Nutzer und ihre Paßwörter) erreicht wird. In unserem Projekt ist diese Funktion und das Sicherheitskonzept von Unix zwingend notwendig für die Zugriffsbeschränkung auf die Datenblätter der Studenten und bei dem Einloggen der Benutzer, da die Daten im Internet bzw. Intranet verfügbar sein werden. Es muß gewährleistet sein, daß die relevanten Dokumente nur von solchen Nutzern aufgerufen werden können, die entsprechend den datenschutzrechtlichen Regelungen Einsicht in die Listen nehmen dürfen.
3. Wie Linux ist auch der Apache modular aufgebaut und kann durch das Einspielen von weiteren Komponenten in seinem Funktionsumfang ergänzt werden. Der Server selbst steht im Main-Modul und der Anwender hat die Möglichkeit, sich den Server nach eigenen Anforderungen aus den verschiedenen Modulen zusammensetzen, indem die nicht benötigten in einer Konfigurationsdatei auskommentiert werden. Der modulare Aufbau bringt mehrere Vorteile mit sich, da nur die wirklich benötigten Applikationen in den Server kompiliert werden und der Apache kleiner bleibt als eine Standardversion und somit weniger Speicherplatz belegt. Weniger Module bedeuten gleichzeitig aber auch eine geringere Angriffsfläche, was die Sicherheit des Systems erhöht. Über ein Application Programming Interface (API) besteht darüber hinaus die Möglichkeit, selbst programmierte Module in den Server zu kompilieren und den Apache noch besser auf die eigenen Anforderungen abzustimmen.

Mit der Version 1.3.4 wurden die bisher notwendigen drei Konfigurationsdateien (*httpd.conf*, *srn.conf* und *access.conf*) in einer Datei zusammengefaßt, der *httpd.conf*, und die Anordnung der Befehlszeilen wurden übersichtlicher gestaltet.

In unserem Fall werden die Module „mod_php“, „mod_perl“ und „mod_ssl“ statisch in den Webserver eingebunden. Dadurch erreichen wir Geschwindigkeitsvorteile gegenüber einer dynamischen Einbindung, da wir dieses Kriterium einer besseren Wartungsmöglichkeit bei der Neukompilierung eines Moduls vorziehen. In dem seltenen Fall einer solchen Ergänzung ist es möglich, den Dienst für ein paar Minuten zu stoppen, um eine Kompilierung durchführen zu können. Die ersten beiden Module ermöglichen die enge Zusammenarbeit mit unserer MySQL-Datenbank durch die Einbindung der Schnittstellen für Perl und PHP, welche zur Erstellung von dynamischen Webseiten benötigt werden. Das dritte dient der zusätzlichen Verschlüsselung und somit Absicherung des Datensatzes bei Zugriffen über das Internet. Der Secured Socket Layer (SSL) ist ein Verfahren, welches sichere Transaktionen im Internet durch verschlüsselte Übertragung möglich macht. Hierbei verwendet das SSL-Protokoll sowohl symmetrische als auch asymmetrische Verschlüsselung und ermöglicht Datenintegrität, Server- und Client-Authentifizierung und Datenverschlüsselung. Bei der symmetrischen Verschlüsselung gibt es nur einen Schlüssel zum ver- und entschlüsseln, d.h. beide Kommunikationspartner muß dieser Schlüssel bekannt sein, was natürlich dazu führen kann, daß bei Kontakt mit mehreren Partnern der Schlüssel vielen Personen bekannt ist. Für die Verwendung eines solchen Verfahrens spricht die vorteilhafte Geschwindigkeit während der Kommunikation. Für die asymmetrische Verschlüsselung werden zwei Schlüssel verwendet, einem Public Key (öffentlicher Schlüssel) und einem Private Key (privater Schlüssel). Hierbei kann nur der jeweils andere Key zur Entschlüsselung benutzt werden, was eine Geheimhaltung trotz häufigen Kontaktes ermöglicht. Der Absender braucht nur den öffentlichen Schlüssel des Empfängers zu kennen, um zum Beispiel eine Nachricht zu schicken, die der Empfänger mit seinem eigenen Schlüssel wieder entschlüsseln kann. Ein Absender kann folglich seinen öffentlichen Schlüssel allgemein bekannt machen, so daß jemand anderes verschlüsselte Nachrichten senden können – der private Schlüssel bleibt dagegen geheim, damit niemand weiterer die Daten eines Empfangs entschlüsseln kann.

Die weiteren Funktionen und Einsatzgebiete eines Apache Webservers sind für unsere Zwecke nicht erforderlich und die Hauptmerkmale sollen an dieser Stelle nur eine kurze Erwähnung finden:

- **Proxyserver**

Zwischenspeicherung von Seiteninhalten zwecks Routing zwischen einer öffentlichen IP-Adresse und einem privaten Adreßraum (gegenüber den Clients im LAN tritt der Proxy als Server auf und gegenüber den externen Servern als Client).

- **Indexerstellung**

Anzeigen der Verzeichnisstruktur mit allen abgelegten Dateien, z.B. als FTP-Ersatz.

- **HTTP-Header**

Erstellen bzw. Weiterleiten von Metafiles und expires:-Headern („Ablaufdatum“ einer Webseite), zur Übermittlung von zusätzlichen Informationen, wie Suchbegriffe oder sogenannten Dokumentverfallsdaten, die einem Cache-Zwischenspeicher aufzeigen, ob die Datei im Cache noch aktuell ist oder neu abgerufen werden muß.

3.3 MySQL

Zwar ist MySQL eine Weiterentwicklung der Datenbank mSQL (Mini SQL), aber trotzdem wurde der Quelltext komplett neu geschrieben und im Mai 1995 veröffentlicht. Dieser Neuentwurf wurde nötig, als sich mSQL bei der Verbindung von Tabellen mit Routinen als zu langsam und unflexibel entpuppte. MySQL ist ein relationales Datenbank Managementsystem (RDBMS), das von der Firma TCX DataKonsult AB in Schweden entwickelt wurde, welches seit der Version 3.21.29 Datenbanken bis zu mehreren Terrabyte Umfang bei praktisch gleichbleibender Geschwindigkeit verkraften kann – so die Angaben der Herstellerfirma. Hierbei wird die Abfragesprache SQL verwendet, die in den 70er Jahren von der Firma IBM

entwickelt wurde und bis heute als Standardabfragesprache für relationale Datenbanken gilt.

Die hohe Geschwindigkeit und die Implementierung eines leistungsfähigen ANSI-SQL-Dialektes sind neben der liberalen Kostenpolitik – im Unix-Bereich weitgehend kostenfrei durch GPL – mit die bedeutendsten Gründe für die Beliebtheit dieses Datenbankservers. Als besonders beliebte Einsatzgebiete im WWW gelten virtuelle Warenkörbe und Bilddatenbanken, da sich hier die Mischung aus Geschwindigkeit und einfacher Struktur als optimal herausgestellt haben.

Zwar hat MySQL die fast dreimal höhere Geschwindigkeit gegenüber vergleichbaren Produkten, aber dafür unterstützt diese Datenbank SQL nicht in vollem Umfang – es wurde aus Performancegründen darauf verzichtet. So werden zum Beispiel nicht unterstützt oder einfach ignoriert:

- **Transaktionen**

Verlaufsprotokolle und Zusammenfassung von mehreren Anweisungen zu einer Arbeitseinheit, um die Fehleranfälligkeit gering zu halten und eine Datenintegrität zu gewährleisten.

- **Stored procedures**

Ein oder mehrere SQL-Anweisungen, die in einer separaten Datenbank unter einem vereinfachten Namen abgelegt werden (gespeicherte Prozeduren).

- **Trigger**

Automatisierung von Überwachungsaufgaben.

- **Referenzielle Integrität**

Bei jeder Veränderung wird quergeprüft, ob eine Verletzung zwischen der Mastertabelle und der Detailtabelle vorliegt (Auffinden von Anomalien).

- **View**

Möglichkeit einer selbstdefinierten Ansicht von bereits existierenden Datenbanktabellen.

- **Commit**

Permanente Übernahme der letzten Änderungen (Datenbank muß Transaktionen unterstützen).

- **Rollback**

Rücknahme der letzten Änderungen, die noch nicht mit „commit“ übernommen worden sind (Datenbank muß Transaktionen unterstützen).

TCX plant aber die Implementierung der noch fehlenden Komponenten in neuere Versionen, doch man möchte dies möglichst ohne Geschwindigkeitseinbußen bewerkstelligen. Auffallend ist die Stabilität des Servers, denn auch nach mehreren vorsätzlich verursachten Abstürzen des Betriebssystems Linux entstanden keine Dateninkonsistenzen, weder bei Systemdateien, noch bei den Datenbankdateien.

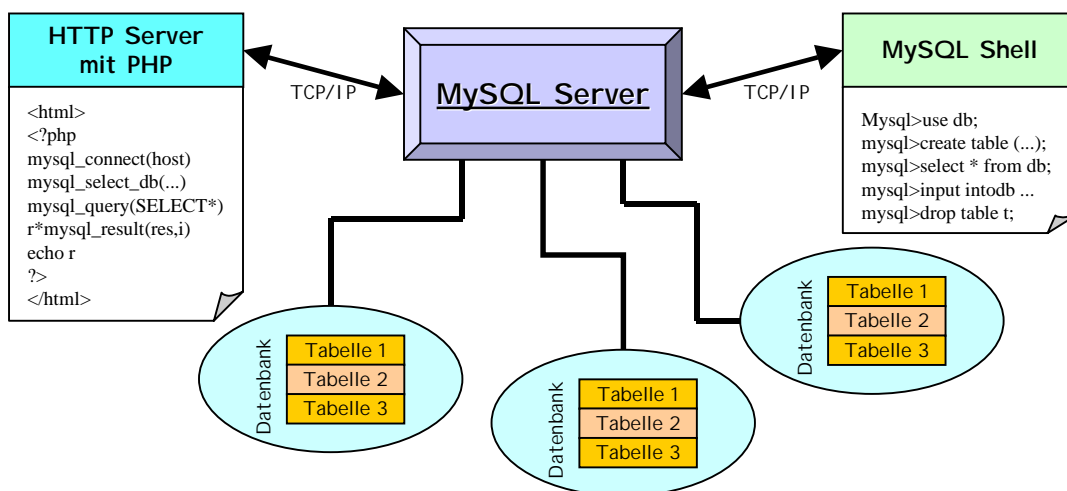


Abb. 8 – MySQL Architektur

MySQL ist eine echte Multi-User- / Multi-Thread-Anwendung, d.h. die Datenbank führt viele Aufgaben zur gleichen Zeit aus, so als würden mehrere Instanzen der Anwendung gleichzeitig laufen. Dieses Design ist natürlich komplexer als das mit nur einem Thread arbeitende mSQL-Design und der Geschwindigkeitsvorteil mehrerer simultaner Abfragen wiegt die Geschwindigkeitsverluste durch die erhöhte Komplexität auf jeden Fall wieder auf. Es ist eine Client/Server-Implementierung, die aus einem Server-Daemon „mysqld“ und vielen Client-Programmen, sowie Bibliotheken für PERL, PHP 3, PHP 4 sowie ASP besteht.

3.4 PHP

PHP wurde Anfang 1995 von Rasmus Lerdorf zunächst als PHP/FI (Form Interpreter) speziell für die Realisierung von Webanwendungen geschaffen. Im Laufe der Zeit hat sich die Bedeutung des namensgebenden Kürzels gewandelt, denn ursprünglich stand es für Personal Homepage Tool, doch mittlerweile wird es als Abkürzung für PHP Hypertext Preprocessor verstanden. Als ein Preprozessor wird in der elektronischen Datenverarbeitung ein Programm bezeichnet, das Eingaben aus einer bestimmten Quelle durcharbeitet und danach als Ausgabe wieder ablegt bzw. direkt zur Endverarbeitung weiterreicht. Genau dies wird mit in HTML-Code eingebetteten PHP-Programmen realisiert, bevor der Web-Server irgendwelche Daten an einen Client, zum Beispiel einem Browser, sendet.

Im Gegensatz zu den bisher vorgestellten Softwareprodukten, ist die erste Version von PHP komplett neu entworfen worden und basiert nicht auf bereits vorhandenen Quellcode. Auch bei der Neuentwicklung des Parsers, dieser analysiert die Struktur (Text, Formatierung, Aufteilung und PlugIns) einer Webseite, so daß der Browser diese korrekt darstellt, werden nicht Vereinigungen von Fans als Entwickler genannt, sondern Einzelpersonen. So fallen bei PHP 3 (1997) und PHP 4 (2000) die Namen Zeev Suraski und Andi Gutmans, welche in die vierten Version auch als Sprachkern die Scripting-Engine ZEND (aus den Namen Zeev & Andi) eingebaut haben. Alle PHP-Quellcodes sind im Internet frei erhältlich und unterliegen der GPL, sogar bei einem kommerziellen Einsatz.

PHP 4 ist weitgehend kompatibel zu der Version 3 und realisiert eine konsequente Trennung der Scripting Engine, die aus einem internen Compiler und einem Executor besteht, von den darauf aufsetzenden und schon von PHP 3 her bekannten, sowie zusätzlichen Modulen. Da durch den internen Compiler der PHP-Code in ein Binärformat übersetzt wird, ist die Version 4 bei der Durchführung von Skripten deutlich schneller als sein Vorgänger. Beide Versionen können entweder als beliebig portables CGI-Programm oder als integriertes Modul für eine Reihe von Webservern, hierbei insbesondere auf dem Apache, eingesetzt werden. Insbesondere unterstützt PHP den Zugriff auf dem verbreiteten, für Webanwendungen wegen seiner hohen

Verarbeitungsgeschwindigkeit und seinem geringen Speicherverbrauch besonders gut geeigneten, Datenbankserver MySQL durch eine Vielzahl von Funktionen. Des weiteren verfügt PHP über ODBC-Funktionen, mit deren Hilfe der komfortable Zugriff auf beliebige ODBC-Datenbanken ermöglicht wird.

Wie bereits angesprochen, wird ein PHP-Skript direkt in eine HTML-Seite durch Abgrenzung mittels Tags eingebunden, welche im allgemeinen mit der Dateiendung *php3* oder *php* gekennzeichnet werden, was aber in der Konfigurationsdatei des Webserver individuell festgelegt werden kann. Ein Zugriff auf eine PHP-Seite erfolgt ohne Einsichtmöglichkeit des Skriptes, da dieses bereits serverseitig interpretiert und in HTML-Code umgewandelt wird. Durch den Webserver wird der benötigte Interpreter gestartet, welcher das angeforderte Dokument übersetzt, den PHP Source-Code der Seite ausführt und die modifizierte Webseite an den Client zurückschickt.

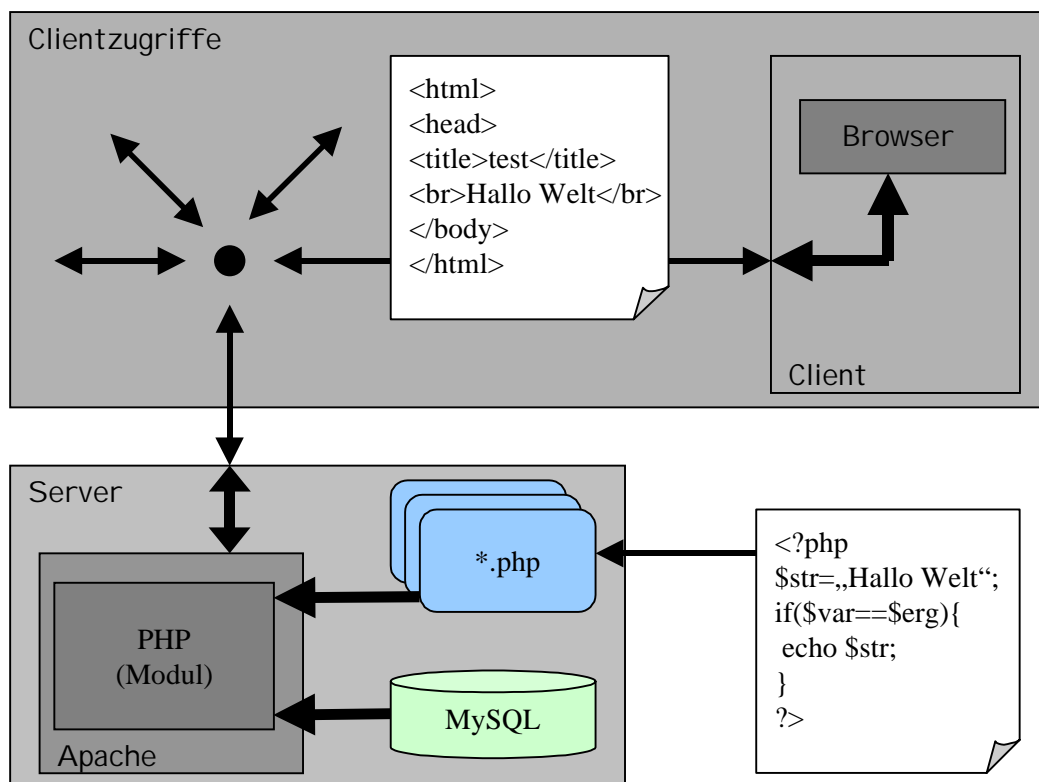


Abb. 9 – PHP als Apache-Modul

Die Syntax dieser Skriptsprache basiert hauptsächlich auf der Programmiersprache C, aber auch einige Eigenschaften von Java und Perl wurden übernommen. Somit

ähnelt eine PHP-Webseite mit seinen Anweisungen (Zuweisungen, Funktionsaufrufe, Schleifen, Bedingungen) und Kommentaren eher einem für C geschriebenen Programmcode als einer gewöhnlichen HTML-Seite. PHP stellt sogar eine Reihe von Sprachkonstrukten zur Verfügung, zum Beispiel Klassen und Objekte, die stark an objektorientierte Programmierung erinnern – im strengen Sinne ist es natürlich keine objektorientierte Sprache.

3.5 Sendmail

Der Daemon Sendmail ist die für Unix-Systeme und –derivate bekannteste Realisation einer Mail-Server-Applikation, welches standardisiert auf Port 25 verbindet bzw. sendet – der englische Fachausdruck für diese Programme lautet Mail Transport Agent (MTA). Sendmail unterstützt neben SMTP (Simple Mail Transport Protocol) weitere Protokolle wie UUCP (Unix to Unix Copy Protocol) und POP (Post Office Protocol). Die Möglichkeiten, die sich dadurch ergeben, lassen schon die gewonnene Komplexität des Programms erahnen, welche nicht mit der Einfachheit des eingesetzten Mailprotokolls verglichen werden kann. Die Konfiguration kann auf zwei Arten erfolgen: entweder durch das Editieren der Konfigurationsdatei *sendmail.cf* oder durch das Erstellen einer m4-Makrodatei, welche durch den Makroprozessor eine gültige *sendmail.cf*-Datei erstellt. Das als letzteres genannte Verfahren wird erst seit der Version 8 (Insidernamen V8) und von der IDA-Erweiterung von Sendmail unterstützt.

Sendmail wurde ursprünglich 1981 von Eric Allman entwickelt und ist mit ca. 80% Marktanteil der dominierende MTA im Internet. Im November 1997 gründeten Eric Allman und Greg Olson mit einigen privaten Geldgebern Sendmail Inc., um eine kommerzielle Version des Programms herzustellen. Die Firma wird weiterhin ein kostenloses MTA herstellen, aber in die kommerziellen Teile werden eine web-basierte Administrationsoberfläche und andere Managementtools integriert sein.

Die von uns verwendete Version beinhaltet ein umfassendes Tool, um Spam-Mails zu kontrollieren und somit einer Zustellung von Datenbank-irrelevanten eMails entgegenzuwirken. Mit Hilfe des MTA werden wir zeitlich gesteuerte Erinnerungs-

emails an die eingetragenen User schicken, die zu einer Überprüfung und ggf. anschließenden Aktualisierung des eigenen Datensatzes auffordern sollen. Inwiefern eine Zustellung von eMails mit Fragen oder Problemen an den Webmasters des Datenbankservers stattfinden und ob diese über die Sendmail-Routine oder über den vorhandenen Mail-Server der Fachhochschule laufen wird, kann bei der Implementierung des Servers festgelegt werden, da hierfür lediglich Einträge auf der Webseite in Form eines Links eingefügt werden müssen. In Sendmail findet dann dementsprechend die Aktivierung oder Deaktivierung der Empfangsoption statt.

3.6 Spiegelung

Nach der Installation der SuSE-Distribution wird eine RAID-Kernelerweiterungen zur Verfügung gestellt, mit der Linear Modus, RAID-0, 1, 4 und 5 als Software-RAID möglich sind. Für das Akronym RAID haben sich zwei Bedeutungen gebildet, nämlich Redundant Array of Independent Disks oder Redundant Array of Inexpensive Disks. Es bezeichnet eine Technik, um mehrere Partitionen miteinander zu verbinden, mit dem Ziel entweder eine Performancesteigerung, eine Steigerung der Datensicherheit oder eine Kombination aus beidem zu erreichen. Dieser Mechanismus kann vor Festplattenfehlern schützen und führt in der Regel auch zu einer Steigerung der Gesamtleistung im Vergleich zu einem Betrieb einer einzelnen Festplatte.

Folgende RAID-Erweiterungen werden als Software-RAID implementiert:

- **Linear Modus (Append)**

Partitionen unterschiedlicher Größe werden über mehrere Festplatten hinweg zu einer großen Partition zusammengefügt und linear beschrieben. Hier ist kein Geschwindigkeitsvorteil zu erwarten und bei Ausfall einer Festplatte sind alle Daten verloren.

- **RAID-0 (Striping)**

Auch hier werden zwei oder mehr Partitionen zu einer großen zusammengefügt, allerdings erfolgt hier der Schreibzugriff nicht linear (Beschreiben der ersten Festplatte bis die Kapazität erschöpft ist, dann folgt die zweite Platte, usw.),

sondern parallel. Dadurch wird ein deutlicher Zuwachs der Datenrate insbesondere bei SCSI-Festplatten erzielt, welche sich für die Dauer des Schreibvorgangs kurzfristig vom SCSI-Bus abmelden können und ihn somit für die nächste Festplatte freigeben – die erzielten Geschwindigkeitsvorteile gehen allerdings zu Lasten der CPU-Leistung. Bei einer Hardwarelösung würde ein RAID-Kontroller diese Arbeit übernehmen, aber der Preis eines guten Kontrollers steht in keinem Verhältnis zur verbrauchten CPU-Leistung eines durchschnittlichen Computers.

- **RAID-1 (Mirroring)**

Eine Festplatte wird auf eine weitere gleichgroße Festplatte oder Partition dupliziert, welches auch als Spiegelung bezeichnet wird. Hierdurch wird eine erhöhte Ausfallsicherheit erreicht, denn bei Ausfall einer Festplatte, bleibt die andere funktionstüchtig – natürlich nur, wenn die gespiegelten Partitionen auf unterschiedlichen Platten liegen. Die zur Verfügung stehende Festplattenkapazität wird durch dieses Verfahren halbiert und ein Geschwindigkeitsgewinn ist nur bei einem Lesezugriff zu erwarten.

- **RAID-4 (Striping & Dedicated Parity)**

Entspricht dem RAID-0 Verfahren, belegt allerdings eine zusätzliche Partition mit Paritätsinformationen, aus denen eine defekte Partition wieder hergestellt werden kann. Diese Funktion benötigt zusätzliche CPU Leistung.

- **RAID-5 (Striping & Distributed Parity)**

Hier werden die Paritätsinformationen zum Restaurieren einer defekten Partition zusammen mit den Daten über alle Partitionen verteilt. Dies führt zu Kapazitätsverlust, denn wenn man zum Beispiel fünf 1 GB-Partitionen zu einem RAID-5 zusammenfaßt, bleiben für die eigentlichen Daten noch 4 GB Platz übrig. Bei dem Schreibvorgang auf einen RAID-5 Verbund wird erst ein Datenblock geschrieben, dann erfolgt die Berechnung der Paritätsinformationen, welche anschließend auch auf den RAID-Verbund geschrieben werden. Das Verfahren führt zwar zu einer schlechteren Schreibgeschwindigkeit, aber der Lesevorgang ähnelt dem RAID-0, was wiederum eine Steigerung der Lese-
geschwindigkeit im Gegensatz zu einer einzelnen Festplatte zur Folge hat.

- **RAID-10 (Mirroring & Striping)**

Diese Technik bezeichnet keinen eigenständigen RAID-Modus, sondern ist ein Kombination aus RAID-0 und RAID-1. Hierbei werden zuerst zwei RAID-0 Verbunde erstellt, die dann mittels RAID-1 gespiegelt werden. Der Vorteil von einem RAID-10 im Gegensatz zu einem RAID-5 besteht in der höheren Performance. Während ein RAID-5 nur relativ wenig Geschwindigkeitsvorteile bringt, ist ein RAID-10 durch die beiden RAID-0-Verbunde für den Fall besser geeignet, wenn man sowohl Redundanz als auch eine hohe Geschwindigkeit erzielen will. Sogar die anschließend notwendige RAID-1 Spiegelung bringt noch einen Vorteil bei der Lesegeschwindigkeit. Weiterhin fällt hierbei die notwendige Berechnung von Paritätsinformationen weg, jedoch eine sehr viel schlechtere Nutzung des vorhandenen Festplattenplatzes muß in Kauf genommen werden, da immer nur 50% der tatsächlichen Kapazität beschrieben werden kann.

Durch die Software-Implementierung der RAID-Verfahren in die Linux-Distribution sind keine speziellen Hardware- oder Festplattenkontroller erforderlich, um viele Vorteile von RAID nutzen zu können und manches läßt sich mit Hilfe der Kernel-erweiterungen sogar flexibler lösen, als es mit Hardwarekontrollern möglich wäre. Für die Verwendung von RAID werden für den Linear, RAID-0 und RAID-1-Modus mindestens zwei leere Partitionen auf möglichst unterschiedlichen Festplatten benötigt. Für RAID-4 und RAID-5 sind mindestens drei Partitionen nötig und für den RAID-10 Modus sogar vier. Unwichtig ist, ob die Partitionen auf (E)IDE- oder SCSI-Festplatten liegen, aber bei Software-RAID mit (E)IDE-Festplatten empfiehlt es sich, jeweils nur ein Platte an einem (E)IDE Kontroller zu benutzen. Im Gegensatz zu SCSI beherrschen (E)IDE-Festplatten keinen Disconnect – können sich also nicht vorübergehend vom Bus abmelden – und können dementsprechend nicht parallel angesprochen werden. An zwei unterschiedlichen (E)IDE Kontrollern ist dies jedoch in Grenzen möglich, schneidet im Vergleich zu SCSI-Kontrollern aber schlechter ab.

Kapitel 4

Schlußbemerkung und Ausblick

Das letzte Kapitel gibt eine kurze Zusammenfassung dieser Ausarbeitung im Rahmen des Ingenieurmäßigen Arbeiten sowie einen Überblick über die während der Implementierungsphase auf einem Testsystem gemachten Erfahrungen. Außerdem werden einige Ansatzpunkte für mögliche Erweiterungen des Datenbankservers vorgestellt, deren Konzeption und Umsetzung im Rahmen eines Ingenieurmäßigen Arbeitens – und einer späteren Diplomarbeit – aus Gründen der Komplexität oder des zeitlichen Aufwandes nicht zu verwirklichen sind.

4.1 Erfahrungen

Als Grundlage einer theoretischen Planung gelten in erster Linie Aufzeichnungen in Form von Büchern, Zeitschriften, Skripten und Webseiten. Um eine technische Umsetzung dieser Arbeit zu garantieren, sind wir nach der Festlegung auf die Kombination LAMP dazu übergegangen, ein vorläufiges PC-System aufzubauen, an dem wir die hier beschriebenen Theorien auch auf Funktionstüchtigkeit und Realisierungsmöglichkeit testeten.

So mußte beispielsweise die Installationsreihenfolge von Webserver und Datenbank abweichend der allgemeinen Aufzählung der LAMP-Komponenten getauscht werden, da bei der Installation der Apache-Module die Einrichtung der MySQL-Module bereits abgeschlossen sein sollten. Ebenso hatten wir Schwierigkeiten bei der Einbindung von bereits in der Distribution enthaltenen Modulen durch

Einschränkungen und Inkompatibilitäten zu anderen. Durch die Verwendung der neuesten Modulversionen wurden aber alle Probleme beseitigt und der volle Funktionsumfang von Apache, MySQL und PHP stand uns zur Verfügung.

Besonders bei der Benutzung von Installations- und Konfigurationsanweisungen ist man auf eine größtmögliche Aktualität angewiesen, denn durch die schnelle Entwicklung neuer Modulversionen tritt eine rasche Veralterung der Dokumente auf. Hierfür hat sich natürlich das Internet als sehr dynamisch bewährt, da die große Linux-Gemeinde die eigenen Foren und Webseiten sehr zuverlässig mit der Dokumentation neuerer Versionen und Hilfestellungen durch Auflistung der Unterschiede zwischen den einzelnen Modulversionen versorgt.

Ein weiteres Problem die ungewohnte Benutzerverwaltung bei dem LAMP-System. So müssen User sowohl unter dem Betriebssystem, als auch für die Datenbank eingerichtet werden, wobei MySQL wiederum zwischen dem *localhost* und einem entfernten Host unterscheidet. D.h. ein Zugriff auf die Datenbank vom Server selbst aus (zum Beispiel über ein Skript wie PHP oder durch Benutzung des Servers als Workstation) unterliegt anderen Voraussetzungen als bei einem Zugriff über das Intra-/Internet mit einem Software-Client wie zum Beispiel MySQL-Front.

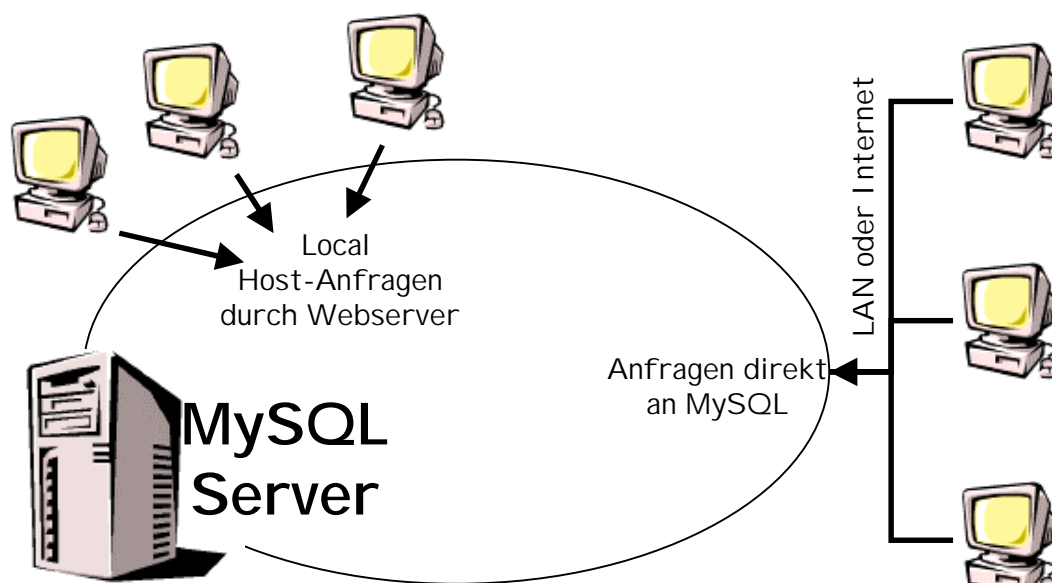


Abb. 10 – Zugriffe auf MySQL bei einem LAMP-Server

4.2 Erweiterungen

Mit dem vorgestellten System zur Verwaltung des Ehemaligenverzeichnisses auf der Basis einer relationalen Datenbank steht ein Werkzeug zur Verfügung, das für den fachhochschulweiten Einsatz ausgelegt ist. Sollte der Fall eintreten, daß die Wahl unseres Datenbank Management Systems MySQL den zukünftigen – aber gegenwärtig nicht abzusehenden – Anforderungen an das Ehemaligenverzeichnisses oder einer Ausbaustufe hiervon nicht gerecht wird, ist ein Wechsel auf eine andere relationale Datenbanksoftware, welche der SQL-Kompatibilität entspricht, problemlos möglich.

Die Erweiterbarkeit war ein wesentliches Kriterium bei der Konzeption des Datenbankservers. Die gegenwärtige Planung und die anschließende Implementierung in der Diplomarbeit bietet daher eine Reihe von Ausbaumöglichkeiten, wie zum Beispiel die Benutzung eines vom Server getrennten Backup-Systems. Die derzeitige Datensicherung erfolgt durch Spiegelung der Festplatten mittels Software-RAID, d.h. im Falle eines rechnerzerstörenden Ereignisses, wie Feuer oder Blitzeinschlag, welcher einen Ausfall aller Laufwerke zur Folge haben kann, besteht keine Möglichkeit mehr an die gespeicherten Daten zu gelangen. Zwar gibt es noch spezialisierte Unternehmen, die eine Datenrettung in solchen Fällen anbieten, doch würde das den Kostenrahmen sprengen, so daß diese Art der Rückgewinnung eher ausgeschlossen werden kann. Im allgemeinen versteht man unter einem Backup-system, welches vom System getrennt ist, die Speicherung der Daten auf einem Datenträger, der an einem anderen Ort als dem Standort des Rechners aufbewahrt werden kann. Hierzu zählen Medien wie Wechselfestplatten, Streamer- oder DAT-Bänder, CD-Rs/CD-RWs und DVD-RAMs.

Eine andere zukünftige Erweiterung ist das Ablegen der Abschlußarbeiten auf dem Server, mit einer Verknüpfung an den Datensatz des jeweiligen Verfassers. Somit könnte bei einem Zugriff auf die Datensätze, welche auch die Themen dieser Abschlußarbeiten beinhaltet, gleichzeitig eine Einsicht in die Dokumente erfolgen. Voraussetzung hierbei ist das einheitlich gewählte Dateiformat, denn die in der Vergangenheit aufgezeigten Probleme mit Versionsänderungen (zum Beispiel alte

MS Office-Dokumente und Version 7.0 oder höher) und Inkompatibilitäten mit anderen Datenverarbeitungen (zum Beispiel MS Office, StarOffice und LaTeX) sollen vermieden werden. Hierzu empfiehlt sich die Benutzung von allgemein anerkannten Standards, wie sie zu Zeit PostScript, Acrobat-Dateien (PDF-Endung) und HTML-Seiten bieten, da diese plattform- und formatierungsunabhängig sind.

Wir empfehlen eine Benutzung der Datenbank bereits während des Studiums, damit der Studentenschaft dieses System vorgestellt werden kann und die Eintragung der Adreßdaten durch die Studenten schon im frühen Stadium stattfindet. Hierzu können Hinweise durch die Professoren oder die lehrenden Mitarbeiter während der Vorlesungen erfolgen. Die Eintragung der Abschlußarbeitsthemen kann dann am Ende durch den betreuenden Professor vorgenommen werden, damit ein einheitliches Erscheinungsbild gewahrt wird. Eine zukünftige Erweiterung in diesem Rahmen bietet die Verwaltung von Studienvoraussetzungen, wie bestandene Praktika oder Klausuren und die Festlegung des Vertiefungsgebietes, welche natürlich nur von einem bestimmten Mitarbeiterkreis eingetragen und verändert werden kann. Da wir uns hierbei lediglich auf das Setzen von Markierungen (zum Beispiel durch eine Checkbox) beschränken wollen, entstehen keine Probleme bezüglich des Datenschutzes, es muß jedoch ein solides Sicherheitskonzept eingerichtet werden.

In vorhergehenden Gesprächen mit der Verwaltung der Fachhochschule wurde auf ein Interesse über eine mögliche Anbindung unserer Datenbank an das DBMS Informix, welches die Fachhochschule einsetzt, per ODBC hingewiesen. Auch die Ausweitung dieses Projektes auf Fachhochschulebene würde großen Anklang finden, da zwar schon mehrere Anfragen an die Verwaltung herangetragen worden sind, aber nie ein Bericht über das Gelingen oder Mißlingen erfolgte.

Während der Recherchen für unser Projekt sind wir auch auf das Vorhandensein von Mechanismen, welche wir in unseren Server implementieren möchten oder in Zukunft anstreben, hingewiesen worden, aber entweder handelt es sich lediglich um einzelne Bestandteile oder wir sind mit den vorhandenen Ergebnissen nicht ganz einverstanden gewesen. So gibt es zum Beispiel bereits eine Verwaltung von Abschlußarbeiten auf WWW-Ebene, doch bei Benutzung dieses Projektes der

Transferstelle ist uns aufgefallen, daß neuere Arbeiten aus dem Studiengang Informations- und Kommunikationstechnik komplett fehlen (zum Beispiel Powerline, Voice over IP und das Projekt der IfKom). Ein weiteres Sammeln von Hochschulschriften zur elektronischen Publikation bietet das Projekt KaFKA, welches durch die Hochschulbibliothek verwaltet wird – hier fehlt gänzlich eine Suchmaschine für den heimischen PC.

Als letzte Erweiterungsstufe des Webservers könnte auch die Bereitstellung von Webspaces erfolgen, so daß interessierten Studenten die Möglichkeit der Einrichtung eines Informationskreises gegeben wird. Hierzu ist als Beispiel die im August 1999 eingerichtete Telcom-Seite²⁵ zu erwähnen, welche aber mit dem fortlaufenden Ausscheiden der Studenten aus dem Erstsemesterjahr 1997/98 einer baldigen Einstellung des Betriebes entgegensteuert. In den drei Jahren des Bestehens wurden diese Seiten hauptsächlich zum Austausch von studienrelevanten Informationen, welche nicht durch die fachhochschulinternen Server abgedeckt worden sind, in Form von eMails und Dateien genutzt. Durch die Verwendung von Disk-Quotas und Zugriffsbeschränkungen, welches das LAMP-System bereits bietet, könnte diese Bereitstellung ermöglicht werden und zu einer noch besseren Kommunikation und Identifikation mit der Fachhochschule oder dem Fachbereich führen.

Inwieweit Erweiterungen jedoch realisierbar und relevant sind, bedarf einer eingehenden Analyse, da alle Veränderungen mit vertretbarem Aufwand verbunden sein sollen.

Anhang

Auf den letzten Seiten dieser Ausarbeitung möchten wir die Erklärung der Fußnoten und ein Glossar zum besseren Verständnis der Ausführungen bereitstellen, welches Begriffe aus der Computerterminologie erklärt. Hierbei wurde insbesondere Wert auf eine für uns spezifische Erläuterung gelegt.

Des weiteren soll ein Abbildungsverzeichnis die Suche nach Erklärungszusätzen erleichtern und den Abschluß dieser Arbeit bildet das Literaturverzeichnis. Letzteres gliedert sich in zwei Teile, wobei der erste eine Aufstellung der verwendeten Literatur enthält und der zweite eine Auflistung von Webseitenadressen ist, welche entscheidende Informationen beigetragen haben.

A.1 Fußnoten

- [1] aus dem Buch *An introduction to database systems* von Christopher John Date, Addison-Wesley Publishing Company Inc., 1995
- [2] aus dem Buch *SQL – standardisierte Datenbanksprache vom PC bis zum Mainframe* von Albrecht Achilles, R. Oldenbourg Verlag GmbH, 1989
- [3] aus dem Buch *Datenbankeinsatz* von Stefan M. Lang und Peter C. Lockemann, Springer Verlag, 1995
- [4] aus dem Buch *Begriffe der Informationstechnik* des Herausgebers DIN Deutsches Institut für Normung e.V., Beuth Verlag GmbH, 1989
- [5] <http://www.netcraft.com/survey>, Januar 2001
- [6] aus dem Buch *Duden Informatik* von Herrmann Engesser, F.A. Brockhaus AG, 1988
- [7] aus dem Buch *SQL for dummies* von Allen G. Taylor, IDG Books Worldwide, 1995
- [8] aus dem Buch *Apache – Das Einsteigerseminar* von Dr. Susanne Wigard, bhv Verlag, 2000
- [9] <http://www.suse.de/de/produkte/susesoft/linux/preise.html>, Januar 2001
- [10] aus dem Online-Shop unter <http://www.btb-online.de>, Januar 2001
- [11] <http://www.netcraft.com/survey>, Januar 2001

- [12] <http://www.infinigate.com/iplanet>, Januar 2001
- [13] <http://www.netcraft.com/survey>, Januar 2001
- [14] aus dem Online-Shop unter <http://www.btb-online.de>, Januar 2001
- [15] http://vopage.ch/webpublishing/neue_seite_2.htm, Februar 2001
- [16] aus einem Artikel von Christian Kirsch in der Zeitschrift *iX*, Verlag Heinz Heise GmbH & Co. KG, Ausgabe 6/2000
- [17] <http://www.tpc.org/rspec.html> & <http://www.tpc.org/wspec.html>, August 2000
- [18] <http://www.chilisoft.com/platforms/linux.asp>, Januar 2001
- [19] <http://uptime.netcraft.com/up/news>, Januar 2001
- [20] Beispiele hierfür sind:
<http://www.linux.org>
<http://www.linux.de/links>
<http://linuxtoday.com>
<http://freshmeat.net>
- [21] <http://uptime.netcraft.com/up/today/top.avg.html>, Februar 2001
- [22] <http://www.gnu.org/copyleft/gpl.html> und abgedruckt in dem Handbuch der SuSE-Distribution 7.0, S.u.S.E. GmbH, 2000
- [23] Beispiele hierfür sind:
<http://www.linux.org>
<http://www.linux.de/links>
<http://linuxtoday.com>
<http://freshmeat.net>

[24] <http://www.suse.de/de/suse/company/index.html>, Februar 2001

[25] <http://welcome.to/telcom>, Februar 2001

A.2 Glossar

Active Server Pages

Microsofts Konzept für dynamische Webseiten, deren Inhalte serverseitig generiert werden (*siehe auch Seite 36*).

ActiveX

Entwicklung von Microsoft, welche die Freigabe von Informationen zwischen Anwendungen erleichtert und die Einbettung beliebiger Objekte (Video, Sound, ...) in fremde Dokumente wie z.B. Webseiten erlaubt. Die in einer ActiveX-Umgebung (Win95/98/ME/NT und MAC) lauffähigen Komponenten werden ActiveX Controls genannt.

Akronym

Ein aus den Anfangsbuchstaben mehrerer Wörter gebildetes Wort.

API

Abkürzung für → *Application Programming Interface*

Application Programming Interface

Programmier- und Anwendungsschnittstelle, die eine Interaktion zwischen Anwendungen und Betriebssystem ermöglicht.

ASP

Abkürzung für → *Active Server Pages*

Backup

Englisch für Sicherheitskopie. Das Betriebssystem und relevante Daten werden mit speziellen Programmen auf diverse Medien gesichert.

Befehlsinterpreter

Interpreter ist die englische Bezeichnung für „Übersetzer“. Es handelt sich dabei um ein Programm, das Befehle einer Programmiersprache in Maschinensprache umwandelt.

Benchmark

Dieser Begriff wurde als Maßstab für Leistungsvergleiche bei Hard- und Software festgelegt.

BNC

Abkürzung für Bayonet Network Connector. Hochfrequenz-Verbindungstechnik, mit der beispielsweise Computer zu Netzwerken verbunden werden. BNC-Kabel zeichnen sich durch gute HF-Eigenschaften aus. BNC-Steckverbindungen werden i.a. für RG58-Kabel (mit Bajonett-Verschluss) bei Übertragungsraten von ca. 10 Mbit/s benutzt.

Browser

Abgeleitet vom Englischen „to browse“ (durchblättern, schmökern, sich umsehen). Hauptsächlich zum Visualisieren von Webseiten benutzt.

Bug

Englisch für Wanze oder Käfer. Umgangssprachliche Bezeichnung für einen Programmfehler. Als die Computer noch mit Relais als Schaltelementen funktionierten, störten manchmal Insekten die Datenverarbeitung indem sie in die Computergehäuse krabbelten (damals noch große Räume) und zwischen den Schaltern zerquetscht wurden. Die Programmierer hatten also echte Bugs in ihrer Anlage.

Bugfix

Umgangssprachlich für ein kleines Programm, welches Bugs aus Programmen beseitigt – oftmals auch als Patch oder Service Pack bezeichnet.

Bundle

Zusammenstellung von Hard- oder Softwarekomponenten zu einem Gesamtpaket.

Bus

Ein System von parallelen Leitungen zur Übertragung von Daten zwischen einzelnen Systemkomponenten, z.B. zwischen Mikroprozessoren, Hauptspeicher, Schnittstellen, Erweiterungskarten und Laufwerken.

Cache

Webseiten werden im Puffer eines → *Proxy-Servers* zwischengespeichert, so müssen sie nicht jedesmal über das Internet geladen werden.

CD-R bzw. CD-RW

Abkürzung für „Compact Disc Recordable“ bzw. „Compact Disc Rewritable“. Die CD-R kann mittels eines CD-Brenners einmal beschrieben werden, die CD-RW mit einem CD-RW-Brenner mehrmals.

CGI

Abkürzung für → *Common Gateway Interface*

Client

An einen Server angeschlossene Arbeitsstation. Er schickt Anfragen des Benutzers in einem speziellen Protokoll an den Server und stellt dessen Antworten in lesbarer Weise auf dem Bildschirm dar.

Common Gateway Interface

Genormte Schnittstelle zwischen Programmen und Webserver (*siehe auch Seite 13f*).

Compiler

Übersetzungsprogramm, welches Programme einer höheren Programmiersprache in den vom Prozessor ausführbaren Maschinensprache umwandelt.

Cookie

Englisch für Kekse. Verbindungsinformationen in Dateiform, die durch den Webserver auf dem Client-Rechner abgelegt werden, um beispielsweise die LogIn-Information bei dem nächsten Besuch wiederzuerkennen,.

CPU

Abkürzung für Central Processing Unit (→ *Prozessor*)

DAT

Abkürzung für Digital Audio Tape (digitale Tonaufzeichnung). DAT-Technologie wird häufig für die Sicherung von Daten verwendet, den sogenannten DAT-Streamern. Es handelt sich hierbei um Bandlaufwerke mit rotierenden Köpfen, wobei die Datensicherung auf speziellen Audio- oder Videokassetten mit 4 oder 8 mm Breite erfolgt. Auf ein Band passen in der Regel 2, 8 oder mehr Gigabyte Daten.

Digital Versatile Disk

Optisches Speichermedium, dessen Kapazität von 2,3 GB bis 17 GB reicht.

Domain

Eine Internet-Domain ist durch einen registrierten Domainnamen wie z.B. *www.vorname-name.de* gekennzeichnet, welche dann von einem Domain Name Server (DNS) in eine bestimmte IP-Adresse übersetzt wird.

DVD

Abkürzung für → *Digital Versatile Disk*

EDO

Abkürzung für Extending Data Out-DRAM (DRAM = Dynamic Random Access Memory). Ein Speichertyp, der schnellere Speicherlesezugriffe als → *FPM-DRAM* ermöglicht. Wurde ursprünglich für den Pentium I mit 66 MHz optimiert.

EIDE

Abkürzung für Enhanced IDE. Es ist eine Verbesserung des IDE-Standards (→ *IDE*), welche das Ansprechen von Festplatten mit einer höheren Kapazität als 528 MB ermöglicht und einen schnelleren Datentransfer erreicht.

FAQ

Abkürzung für Frequently Asked Questions (häufig gestellte Fragen). FAQ-Seiten enthalten kurze und klare Antworten auf die am meisten gestellten Fragen zu einem Thema.

FPM

Abkürzung für Fast Page Mode. War lange der DRAM-Standard, ist aber mittlerweile veraltet und wird von → *EDO*, aber noch mehr von → *SDRAM* übertroffen. Es optimiert den Zugriff auf Daten durch Zwischenspeicherung und eine verbesserte Adressierung.

Firewall

Englisch für Brandschutzmauer. Sicherheitsmechanismus, bei dem verschiedene Netzwerke (typischerweise → *Intranet* und → *Internet*) nicht direkt miteinander verbunden sind. Nur nach einer Reihe von Sicherheitsprüfungen durch eine zwischengeschaltete Hard- oder Software werden Zugriffe zugelassen.

Flash

Datenformat bzw. Software der Firma Macromedia, welche die Benutzung vektorbasierter Grafiken ermöglicht. Die Software wird für diverse Webbrowser als Erweiterung zur Verfügung gestellt.

Forum

Elektronische Form des Schwarzen Brettes. Foren werden zum Informations- und Erfahrungsaustausch genutzt und sind in verschiedene Themenbereiche gegliedert. Andere Bezeichnungen für Forum sind Brett, Newsgroup oder Konferenz.

Frontend

Andere Bezeichnung für eine graphische Benutzeroberfläche.

General Public License

Juristische Absicherung des GNU Projektes (*siehe auch Seite 48f*).

GNU

Projekt der Free Software Foundation (FSF). Ziel ist die Schaffung eines rechtlich freien Raumes (Recht auf Zugang, Veränderung und Benutzung des Programm-codes) durch die Verbreitung eines Unix-kompatiblen Betriebssystems (*siehe auch Seite 49*).

GPL

Abkürzung für → *General Public License*

Header

Bereich am Anfang (Kopf) von Dateien, in dem grundsätzliche Informationen über die Datei gespeichert sind.

Hosting, virtual

→ *Websserver, virtueller*

How-To

Im Linux-Bereich sogenannte Installationsanleitungen für bestimmte Softwarepakete oder Hardwarekomponenten.

HTML

Abkürzung für → *Hypertext Markup Language*

HTML, Dynamic

Zusammenspiel von Skriptsprache, Cascading Style Sheets (CSS = Sprache zur Definition von Vorlagen) und HTML, um veränderbare bzw. an den Besucher angepaßte Webseiten zu generieren.

Hypertext Markup Language

Programmiersprache für Webseiten.

IDE

Abkürzung für Integrated Disk Electronics. Standardisierte Schnittstelle zwischen dem Datenbus der Hauptplatine und angeschlossenen Laufwerken, i.d.R. Festplatten und CD-Rom.

Inkonsistenz

Dateninkonsistenz ist eine Widersprüchlichkeit, die aus den Informationen der Datenbank resultiert.

Integrität, referenzielle

Querprüfung von Datensätzen zur Auffindung von Fehlern.

Internet

Weltumspannendes heterogenes, d.h. aus verschiedensten Rechnern bestehendes, Netzwerk. PCs werden im Internet über das IP-Protokoll und einer eindeutigen Adreßstruktur, den sogenannten IP-Adressen, angesprochen.

Intranet

Internes (Firmen-)Netzwerk.

ISAPI

Abkürzung für Microsofts Internet Server Application Interface. Beschleunigt den Aufruf serverseitiger Programme, indem es als Plug-ins in den Webserver integriert ist. Meist in C oder C++ geschrieben.

Java Applet

Statt Programme auf einem Server zu hinterlegen, können sie auch als Java-Applet kompiliert an den Nutzer übertragen werden, wo sie mit Hilfe eines javafähigen Browsers ausgeführt werden.

Java Database Connectivity

Von Sun entwickelte Datenbankschnittstelle mit vielen Vorteilen gegenüber herkömmlichen Schnittstellen. Damit ist es auf vergleichsweise einfache und effektive Art möglich aus Applets heraus auf externe Datenbanken zuzugreifen.

JDBC

Abkürzung für → *Java Database Connectivity*

Kernel

Das Herzstück des gesamten Linux-Systems. Zuständig für Speicherverwaltung, Führung der Prozeßtafel, Management von Multiuser- und Multitasking-Fähigkeit, die Verwaltung der Zugriffe auf das Dateisystem, die Treiber für die Zugriffe auf die jeweilige Hardware, etc..

Knowledge Base

Zusammenstellung von Problemlösungen und Anleitungen.

Locking

Sperrung eines Datensatzes zur Aufrechterhaltung der Integrität.

Logfile

Datei, in der die Aktivitäten eines Betriebssystems oder Programms protokolliert werden.

Mainboard

Englisch für Hauptplatine. Die zentrale Plattform, auf der die weitere Hardware modular aufgesteckt wird, wie z.B. Speicherbausteine, Grafikkarte, CPU, usw..

Mainframe

Englisch für Großrechner. Meist Hauptrechner eines Computersystems, an das sehr viele Arbeitsstationen angeschlossen sind und der mit Arbeitsspeicher von mehr als 100 MB und Festplattenkapazitäten im TerraByte-Bereich ausgestattet sein kann.

Makro

Kombination einzelner Anweisungen oder Folge von Befehlen und Vorgängen, z.B. eine Kombination von Tasten- und Mausclicks, welche festgehalten und gespeichert worden sind. Wird ein Makro aufgerufen, werden die aufgezeichneten Vorgänge und Aktionen in der entsprechenden Reihenfolge automatisch wieder abgearbeitet.

Manpage

Bei Unix-Systemen liegt die Dokumentation traditionellerweise in Man(ual)-Pages vor, die mit dem Befehl *man* einsehbar sind.

Manual

Englisch für Handbuch. Vom Hersteller mitgelieferte Dokumentation zu Hard- und Software-Produkten.

Meta-File

Datei, welche Informationen zu einem bestimmten HTML-Dokument enthält (z.B. Header und Suchbegriffe).

Midrange

Rechnerausstattung, welche zwischen dem → *SoHo*- und dem Großrechner angesiedelt wird.

Modul

Einzelner Bestandteil eines zusammengesetzten Systems oder Programms.

Multitasking

Fähigkeit, mehrere Aufgaben (Programme) gleichzeitig ausführen zu können (vom Englischen Wort „task“ (*Aufgabe*))

NSAPI

Abkürzung für Netscape Server Application Programming Interface. Beschleunigt den Aufruf serverseitiger Programme, indem es als Plug-ins in den Webserver integriert wird. Meist in C oder C++ geschrieben.

ODBC

Abkürzung für → *Open Database Connectivity*

Open Database Connectivity

Englisch für offene Datenbankverbindung. Standardisierte Methode, die den Zugriff auf Datenbanken erlaubt, ohne dabei zu berücksichtigen, aus welchem Programm oder von welchem Betriebssystem aus der Zugriff erfolgt. Beruht auf einer Spezifikation, die von der SQL-ACCESS-Group (SAG) unter Federführung von Microsoft ins Leben gerufen wurde.

OptionPack

Zubehör-CD für MS Windows NT 4.0, auf dem bei der Serverversion u.a. der Microsoft Internet Information Server enthalten ist.

Organizer

Andere Bezeichnung für PDA (→ *Personal Digital Assistant*)

Overhead

Englische Bezeichnung für die Überlastung eines Systems mit Aktionen, welche die Produktivität verhindern.

Parität

Ein Prüfverfahren in der EDV, um Fehler bei der Datenübertragung zu entdecken. Mit Hilfe eines sogenannten Paritätsbits, ein zusätzlich übertragenes Kontroll-Bit, errechnet der Sender eines Bytes die Quersumme der Bits und setzt ggf. das Paritätsbit. Der Empfänger gleicht die Quersumme der empfangene Daten mittels des Paritätsbits ab. Ist sie gleich, dann war die Datenübertragung fehlerfrei.

Parser

Strukturanalysator (*siehe auch Seite 58*).

Partition

Einheit eines definierten Speicherbereichs einer Festplatte, die als eigenständiges Laufwerk angesprochen und behandelt werden kann .

Patch

→ *Bugfix*

PDA

Abkürzung für → *Personal Digital Assistant*

Personal Digital Assistant

Minirechner im Westentaschenformat für Adressen, Terminverwaltung und andere weniger speicheraufwendige Anwendungen, welcher auch über eine Schnittstelle für den Anschluß an den heimischen PC zwecks Abgleich verfügen kann.

PlugIn

Häufig kostenlose Hilfsprogramme, welche die Funktionalität einer Software oder eines WWW-Browser erweitern.

POP

Abkürzung für → *Post Office Protocol*

Post Office Protocol

Client/Server-Protokoll bei dem der Server Mails empfängt und für den Client bereithält, bis sich dieser am Server anmeldet. Nach Authentifizierung werden die eMails an den Client durchgereicht.

PostScript

Von Adobe System Inc. 1984 auf den Markt gebrachte Seitenbeschreibungssprache für das Ausdrucken und Speichern von Grafiken und Texten. Das System arbeitet system-, größen- und auflösungsunabhängig, wobei die Qualität des Ausdruck sich einzig nach den technischen Möglichkeiten des Ausgabegerätes richtet.

Proxy

Rechner, der stellvertretend für die Clients in einem lokalen Netz Dokumente von externen Servern anfordert, diese zwischenspeichert und an die Clients weiterleitet.

Prozessor

Schaltzentrale eines jeden Computers. In ihm werden die Befehle des Benutzers bzw. der Programme in Maschinensprache abgearbeitet und ausgeführt. Er hat die Kontrolle über das gesamte System und erbringt die eigentliche Rechenleistung.

Publishing-Programm

Programm zu Erstellung von Präsentationsmaterial.

Query Language

Abfragesprache für Datenbanken.

RAID

Abkürzung für → *Redundant Array of Independent Disks*

Reboot

Herunterfahren aller Dienste und Anwendungen und kompletter Neustart des Betriebssystems.

Redundant Array of Independent Disks

oder

Redundant Array of Inexpensive Disks

Zusammenschaltung von Festplatten, die einen Geschwindigkeitsvorteil und/oder eine erhöhte Datensicherheit gewährleisten soll (*siehe auch Seite 61ff*).

Root

Englisch für Wurzel. Zum einen das oberste Verzeichnis eines Unix-Dateisystems, zum anderen die Kennung des Systemadministrators eines Unix-Systems, welcher als einziger Zugang und Rechte zu allen Diensten eines Rechnersystems hat.

SCSI

Abkürzung für Small Computer Systems Interface. Leistungsfähige Schnittstelle zum Verbinden von PC-Komponenten wie Festplatten, CD-Rom, CD-Writer, Wechselplattensystemen, Scannern etc..

SDRAM

Abkürzung für Synchronous DRAM. Der Nachfolger von \rightarrow EDO synchronisiert sich mit dem Systemtakt, der den Prozessor kontrolliert, welches Zeitverzögerungen bei einem Zugriff verhindert. Eine Datenserie, auch Burst genannt, wird zügig übertragen. SDRAM hat eine völlig andere Architektur als klassisches DRAM und ist wesentlich schneller. Diesen Speichertyp gibt es als 168-polige DIMM-Module für Systeme mit 66 MHz und 100 MHz Bustakt.

Secured Socket Layer

Protokoll, welches entwickelt wurde, um sichere Datenübertragung über das Internet zu ermöglichen. Alle neueren Netscape- oder Microsoft-Browser unterstützen dieses Verfahren. SSL nutzt den Public-Key, bei dem mit einem öffentlich zugänglichen Schlüssel codierte Daten nur mit einem ganz bestimmten privaten Schlüssel wieder dechiffriert werden können. Je nach Browser zeigen die SSL-Symbole (Schlüssel oder Vorhängeschloß) eine ungesicherte bzw. eine gesicherte Verbindungen an.

Server

Abgeleitet vom Englischen Wort "to serve" (dienen, jemanden versorgen). Zentraler Rechner in einem Netzwerk, welcher den Arbeitsstationen (Clients) Daten, Speicher und Ressourcen zur Verfügung stellt. Auf dem Server ist das Netzwerk-Betriebssystem installiert, und vom Server wird das Netzwerk verwaltet. Im WWW sind Server Knotenpunkte des Netzes.

Server Side Includes

Ein WWW-Server (HTTP-Server) kann die vom Client abgerufenen Dokumente vor der Auslieferung überarbeiten und z.B. darin enthaltene Befehle ausführen. Server Side Includes werden nur ausgeführt, wenn der WWW-Browser die HTML-Datei über einen installierten WWW-Server aufruft, also mit einer URL-Adresse vom Typ *http://...*. Voraussetzung ist ferner, daß der installierte WWW-Server SSI unterstützt, sonst funktionieren die entsprechenden Anweisungen in der HTML-Datei nicht.

Session

Englisch für Arbeitssitzung.

Simple Mail Transfer Protocol

Standard für den Austausch von eMails zwischen Servern im Netzwerk. eMail-Clients benutzen SMTP, um eMails an einen Server zu schicken, nicht aber zum Empfang.

SMTP

Abkürzung für → *Simple Mail Transfer Protocol*

SoHo

Abkürzung für „Small Office/Home Office“. Rechnerausstattungen für ein kleines Büro oder dem Heimbüro. Gemeint sind vor allem Rechnersysteme für Privatleute, Freiberufler und Selbständige.

Source-Code

1. In einer Programmiersprache geschriebener ASCII-Text, der durch ein Übersetzungsprogramm in eine ausführbare Form umgesetzt wird.
2. Quelltext (Vom Englischen Wort "source") einer HTML-Seite, welche von einem Browser dargestellt werden kann.

Spam

Unerwünschte eMails, meist Werbemails. Der Wert dieser Mitteilungen entspricht "Spam", ein eingetragenes US-Warenzeichen für ein rosafarbenes Dosenfleisch. Allgemeines Urteil der amerikanischen Bevölkerung: unästhetisch und ohne Nährwert.

SSI

Abkürzung für → *Server Side Includes*

SSL

Abkürzung für → *Secured Socket Layer*

Streamer

Andere Bezeichnung für ein Magnetband-Laufwerk. Es wird zur Sicherung von einzelnen Daten, Festplatten oder ganzen Netzwerken eingesetzt. Die Speicherkapazität eines Band liegt abhängig vom verwendeten Typ zwischen 100 MB und mehreren Gigabytes.

Tag

Englisch für Etikett. Steuersymbole zur Formatierung von HTML-Quelltexten oder SGML-Dokumenten.

Telnet

Möglichkeit für Internet-User direkten Zugriff auf andere Computer im Netzwerk zu erhalten. Der Begriff steht außerdem für die standardisierte Terminal-Emulation und das dazugehörige Übertragungsprotokoll.

Transaktion

Anweisungs- bzw. Informationssequenz (*siehe auch Seite 56*).

Trojaner

Computervirus, welcher wie eine harmlose Datei aussieht, nach dem Eindringen aber wichtige Dateien ausspäht oder ändert. Auch in Anwendungen (also Programmen) lassen sich Funktionen verstecken, die z.B. Passwörter ausfindig machen und sie über das Internet an den Späher versenden.

Twisted Pair

Verdrillte Kupferdrahtleitung. Einer der verbreitetsten Kabeltypen (neben BNC), mit dem ein Netzwerk (LAN) verkabelt wird (RJ45-Kabel mit Westernstecker).

Unix to Unix Copy Protocol

Protokoll, welches ursprünglich die Übertragung von Daten zwischen Unix-Rechnern über die serielle Schnittstelle oder Modem ermöglichen sollte. Inzwischen kann UUCP auch auf normalen Netzen und über TCP/IP eingesetzt werden. In den meisten Fällen dient es dazu, eine durch Zusammenpacken von Nachrichten erzielte, möglichst schnelle Übertragung von eMails und News an Rechner zu gewährleisten, die mit einer langsamen Leitung an das Internet angeschlossen sind. Viele Internet-Provider bieten UUCP allerdings gar nicht oder nur auf Nachfrage an.

Unterbrechungsfreie Stromversorgung

Technik, mit der ein Gerät auch bei einer Unterbrechung des öffentlichen Stromnetzes eine Zeitlang weiterarbeiten und beispielsweise noch die veränderten Daten speichern kann.

USV

Abkürzung für → *Unterbrechungsfreie Stromversorgung*

UUCP

Abkürzung für → *Unix to Unix Copy Protocol*

VBScript

Abkürzung für → *Visual Basic Script*

Viewer

Englisch für (Datei-)Betrachter. Für HTML-Dokumente ist der Webbrowser der Viewer.

Visual Basic Script

Untermenge des Visual-Basic Programmiersystems von Microsoft. VBScript sind in HTML-Seiten eingebettete Programme, die von Browsern gelesen werden. VBScript-Programme werden entweder auf dem Computer, auf dem der Browser installiert ist, oder auf dem WWW-Server ausgeführt. (*siehe auch Seite 35*).

Webseite, dynamische

→ *HTML, Dynamic*

Webseite, statische

Eine solche Seite wird aus bestehendem HTML-Code für jeden User gleichartig dargestellt (im Gegensatz zur → *dynamischen Webseite*)

Webserver, virtueller

Eine implementierte Webserverfunktion, welche einen bestimmten Ordner im Dateisystem nach außen hin wie einen eigenständigen Webserver darstellt.

Webspace

Zusammenfassung von Internet-Diensten und Serviceleistungen, die benötigt werden, um eine eigene Internet-Präsenz aufbauen zu können.

World Wide Web

Bezeichnet die über Hyperlinks (Querverweis zu einem HTML-Dokument) untereinander vernetzten HTML-Dokumente im Internet.

WWW

Abkürzung für → *World Wide Web*

A.3 Abbildungsverzeichnis

Abb. 1 – Beispiel für ein hierarchisches Datenbankmodell	6
Abb. 2 – Beispiel für ein Netzwerk-Datenbankmodell	6
Abb. 3 – Beispiel für ein relationales Datenbankmodell	7
Abb. 4 – Webserver Survey der Firma Netcraft (Auszug der Webseite)	28
Abb. 5 – Fortschritte bei der Versionsentwicklung des IIS	30
Abb. 6 – Wachstumsrate der Benutzung von PHP	37
Abb. 7 – Arbeitsweise eines LAMP-Servers	40
Abb. 8 – MySQL Architektur	57
Abb. 9 – PHP als Apache-Modul	59
Abb. 10 – Zugriffe auf MySQL bei einem LAMP-Server	65

A.4 Literaturverzeichnis

Sekundärliteratur:

- A. Achilles
SQL – standardisierte Datenbanksprache vom PC bis zum Mainframe
R. Oldenbourg Verlag GmbH, 1989

- A. G. Taylor
SQL for dummies
IDG Books Worldwide, 1995

- C. J. Date
An introduction to database systems
Addison-Wesley Publishing Company Inc., 1995

- C. Kirsch
Artikel *Datendüse, MySQL: Freie Datenbank* in der Zeitschrift *iX*
Verlag Heinz Heise GmbH & Co. KG, Ausgabe 6/2000

- Dr. S. Wigard
Apache – Das Einsteigerseminar
bhv Verlag, 2000

- Dr. S. Wigard
PHP 4.0 – Das Einsteigerseminar
bhv Verlag, 2000

- H. Engesser
Duden Informatik
F.A. Brockhaus AG, 1988

- J. Krause
Microsoft Active Server Pages
Addison-Wesley Publishing Company Inc., 2000

- K. Schmidt
PHP 4
C&L Computer & Literaturverlag, 2000

- R. D. Stoll und G. A. Leierer
PHP 4 + MySQL
DATA BECKER GmbH & Co. KG, 2000

- R. J. Yarger, G. Reese und T. King
MySQL & mSQL
O'Reilly Verlag, 2000

- R. Stephens u.a.
SQL in 21 Tagen
SAMS, ein Imprint der Markt & Technik GmbH, 1998

- S. M. Lang und P. C. Lockemann
Datenbankeinsatz
Springer Verlag, 1995

- *c't plus ROM - Das Jahresarchiv auf CD-Rom*
Verlag Heinz Heise GmbH & Co. KG, Ausgaben 1998-2000

- *Internet Professionell – Das Jahresarchiv auf CD-Rom*
Ziff-Davis Verlag GmbH, Ausgaben 1999 & 2000

- *S.u.S.E. Linux 7.0 – Das Handbuch*
S.u.S.E. GmbH, 2000

Tertiärliteratur:

- <http://www.apache.org>
- <http://apachetoday.com>
- <http://freshmeat.net>
- <http://infobase.id-pro.de>
- <http://krum.rz.uni-mannheim.de/web-tech>
- <http://linux-2000.org>
- <http://members.aol.com/duenhoel>
- http://sdb.suse.de/sdb/de/html/lmuelle_suselinux_internet.html
- <http://vopage.ch>
- <http://www.altavista.com>
- <http://www.altavista.de>
- <http://www.apache.org>
- <http://www.aspheute.com>
- <http://www.btb-online.de>
- <http://www.chilisoft.com>
- <http://www.dbg.rt.bw.schule.de/lehrer/ritters/info/linux/linux.htm>
- <http://www.dynamic-webpages.de>
- <http://www.fhb.fh-dortmund.de/kafka.htm>
- <http://www.fireball.de>
- <http://www.firmen-info.de/mysql/mysql.html>
- <http://www.glossar.de>
- <http://www.guug.de/~winni/linux>
- <http://www.heise.de>
- <http://www.heise.de/ix>
- <http://www.ideenreich.com>
- <http://www.im.nrw.de>
- <http://www.infinigate.com>
- <http://www.linux.de>
- <http://www.linux.org>

- <http://www.linux.org/info/gnu.html>
- <http://www.linux-news-network.de>
- <http://www.microsoft.com>
- <http://www.mysql.com>
- <http://www.net-graphics.de>
- <http://www.netcraft.co.uk>
- <http://www.online-tutorial.de>
- http://www.oreilly.de/german/freebooks/linux_netz/inhalt.html
- <http://www.pageresource.com/asp/asp1.htm>
- <http://www.php.net>
- <http://www.php.net/manual/de>
- <http://www.php-center.de>
- <http://www.php-resource.de>
- <http://www.phpbuilder.com>
- <http://www.suse.de>
- <http://www.tcx.se>
- <http://www.teamone.de/selfaktuell>
- <http://www.tomix.de/linux/slm>
- <http://www.toppoint.de/~tulli/lamp.html>
- <http://www.transferstelle.fh-dortmund.de>
- <http://www.yahoo.com>
- <http://www.yahoo.de>
- <http://www.zend.com>

Erklärung

Hiermit versichern wir, die vorliegende Diplomarbeit selbstständig angefertigt und keine weiteren als die angegebenen Hilfsmittel und Quellen verwendet zu haben.

Dortmund, den 01.03.2001

.....
Unterschrift André Blase

.....
Unterschrift Carsten Sander