

Peter Micheuz

Internet inside – dynamisch und interaktiv

Skriptsprachen und Datenbanken in einer Client-Server-Umgebung

*Ich lese immer die Einführung,
weil sie manchmal der beste Teil des Werks ist.*

Cecilia Bohl de Faber
Spanische Schriftstellerin (1796– 1879)

1 Einführung

Niemand wird ernsthaft bezweifeln, dass das Internet bereits jetzt die Bildungslandschaft im Allgemeinen und den Informatikunterricht im Besonderen durchdringt und dieser Trend sich in Zukunft weiter verstärken wird. Die Nutzung des Internet als Medium für die Erschließung neuer Lehr- und Lernwelten ist dabei nur ein Aspekt. Er spielt in diesem Beitrag nur so weit eine Rolle, als das Internet zur schnellen Informationsbeschaffung beitragen kann. Vom Standpunkt des Informatikunterrichts hat die **Vermittlung der theoretischen Grundlagen des Internet**, seiner Funktionsweise und der darauf basierenden Entwicklung von dynamischen Webanwendungen eine herausragende Bedeutung. Eine interessante didaktische Schleife, dass nämlich der **Computer Werkzeug und Gegenstand** des Informatikunterrichts zugleich ist, gilt für das Informatiksystem Internet, das Lehr- und Lernmedium zugleich ist, im Besonderen.

Dieser Beitrag befasst sich mit den faszinierenden Möglichkeiten, die Funktionalität, Attraktivität und den Nutzwert von Internetseiten durch **Skriptsprachen** inklusive deren Anbindung an Datenbanken mit vergleichsweise wenig Aufwand wesentlich zu erhöhen. Besonderer Wert wird in diesem Beitrag darauf gelegt, die **Abläufe in Client-Server-Systemen** auf verschiedenen Abstraktionsebenen verständlich zu machen. Die Entscheidung, die einige praxisbezogene Beispiele mit der speziellen Software PHP und MySQL zu realisieren, soll den invarianten Aspekt dieses Beitrages nicht schmälern.

Ein Ausschnitt einer Homepage (<http://www.schulinformatik.at>), siehe Abbildung 1, möge einen ersten Überblick und kleinen Vorgeschmack auf die folgenden Ausführungen geben.

Websites, die den Eindruck mangelnder Wartung vermitteln, obwohl in den Tiefen der Sitestruktur sehr viel Content (Inhalt) dazugekommen ist, lassen sich z. B. durch Zufallssprüche, Umfragen, Newsticker oder durch einen einfachen Zähler beleben. Beim Betrachter hinterlässt dies den **Eindruck von Professionalität und Aktivität**.

Für den Informatikunterricht ist die Frage, wie **dynamisch** (im Hinblick auf wechselnde Inhalte) gestaltete Internetseiten realisiert werden können, in vielerlei Hinsicht sehr interessant:

- Wie und wo sind die Zufallssprüche gespeichert und wie gelangen sie auf den Bildschirm? (Datenspeicherung und Kommunikation)
- Wie kann verhindert werden, dass die Zahl der Seitenaufrufe manipuliert wird? (Vertraulichkeit)

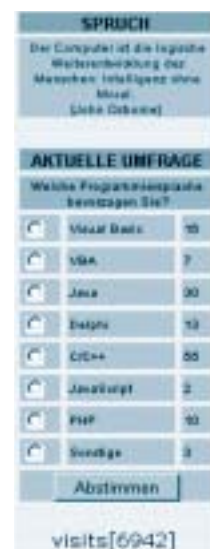


Abb. 1

Fachbereich allgemeinbildende höhere Schule

Das Internet bietet 1.000.000 Antworten auf Fragen, die nie einer gestellt hat.
 Sir Peter Ustinov (*1921)

2 Wie funktioniert das Internet?

„Alles im Internet ist ein Dokument“

Die Idee von Hypertext ist älter als das Internet

Hypertext ist die vernetzte Darstellung von Informationen

Das Internet ist ein großes, aus vielen Millionen, weltweit vernetzten Computern bestehendes Informatik-System. Diese Computer verständigen sich über eine gemeinsame Sprache, das **TCP/IP-Protokoll**. Die Bedeutung des **World Wide Web (WWW)** als wichtigster Teil des Internet liegt in seiner Einfachheit: Alles ist ein Dokument. Damit ist das World Wide Web der bedeutendste und populärste Vertreter der so genannten dokumentenbasierten verteilten Systeme. Und das **Hypertext**-Konzept, in dem die Dokumente miteinander verknüpft (verlinkt) sind, das prägende Gestaltungsmerkmal. Für das weitere Verständnis ist die Unterscheidung der Computer in so genannte **Clients und Server** und deren Kommunikation besonders wichtig. Über den Client stellt der User seine **Anfragen (requests)** an den Server, der daraufhin im Normalfall die angeforderten Dokumente an den Client zurücksendet.

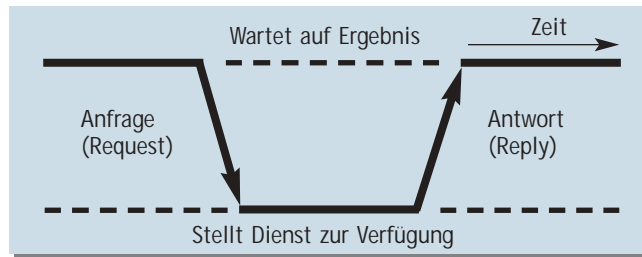


Abb. 2: Schematische Darstellung einer Client-Server-Kommunikation



Abb. 3: Eingabe der Adresse im Internet-Browser



Abb. 4: Anzeige der angeforderten Dokumente und den mit der Homepage verbundenen weiteren Objekte wie z. B. Graphiken.

Das Verstehen dieses Frage-Antwort-Verhaltens von Client und Server ist auch für einen unbedarften Internetsurfer nicht uninteressant. Im einfachsten Fall gibt er im **Internetbrowser** eine (gültige) Internetadresse z. B. <http://www.schulformatik.at> ein und bekommt auf dem Bildschirm die Homepage zu sehen. Was aber bekommt er da zu sehen? Ein Dokument? Welches Dokument? Ist die Internetadresse das Dokument? Diese Fragen sind berechtigt und nicht trivial, da ja in der Internetadresse, – auch als

Micheuz: Internet inside – dynamisch und interaktiv

URL (Uniform Resource Locator) bezeichnet –, (meistens) keine Datei zusätzlich angegeben wird. In unserem Beispiel wird nach dem Aufruf der Adresse „http://www.schulinformatik.at“ vom Webserver – wo immer er auch lokalisiert sein möge – die Datei „index.php“ in einem speziellen Verzeichnis automatisch gesucht und an den Client gesendet. Der Grund, warum zusätzlich zur Adresse http://www.schulinformatik.at nicht das Startdokument, nämlich das Dokument **index.php** eingegeben werden muss, ist folgender: Der Webserver, der die Anforderung des Browsers beantwortet, weiß durch eine entsprechende Konfigurationseinstellung, dass er in dem voreingestellten Wurzelverzeichnis der Homepage (document root) nach Dateien wie index.htm, index.html oder wie in unserem Fall index.php suchen muss. Was es mit der Endung „php“ statt „html“ auf sich hat, ist ein zentrales Thema in diesem Beitrag.

URL: Uniform Resource Locator
Weltweit eindeutige Adresse, hinter der ein 32-stelliges Bitmuster steht
URI: Universal Resource Identifier
(nachzulesen im RFC – request for command – 1630)

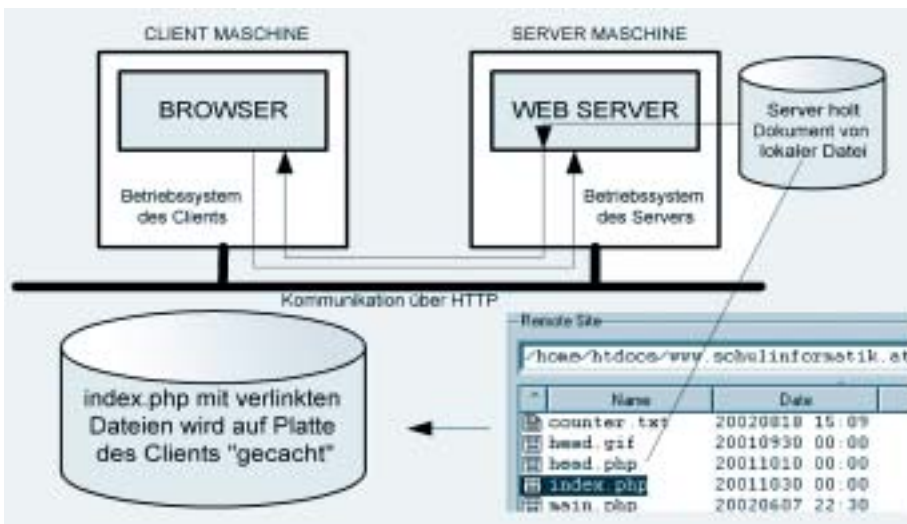


Abb. 5: Schematische Darstellung der Kommunikation im Internet

Die Kommunikation zwischen Client und Server basiert auf dem verbindungsorientierten TCP/IP-Protokoll. Verbindungsorientiert deshalb, weil vor der eigentlichen Internetkommunikation (siehe oben) eine Verbindung mit dem Zugangsprovider (auch: Access Service Provider) bestehen muss. Ein wesentlicher Bestandteil der Protokollfamilie TCP/IP, die auf dieser Verbindung aufsetzt, ist **das Protokoll HTTP (Hypertext Transfer Protokoll)**, das Grundlage des World Web ist. Wird die Seite index.php angefordert, so werden mit ihr auch alle verlinkten Dateien (z. B. Grafiken) übertragen, was die Bezeichnung H für „Hypertext“ in der Protokollbezeichnung rechtfertigt.

TCP/IP = Transmission Control Protocol/Internet Protocol als Teil des Protokollstapel

Warum die erste aufgerufene Seite „index.php“ und nicht „index.htm“zw. „index.html“, wie eigentlich erwartet, heißt, hat damit zu tun, wie Webdokumente vom Webserver interpretiert werden. Kann man davon ausgehen, dass HTML-Dokumente (*.htm, *.html) nur (statische) HTML-Tags, Texte und (dynamische) Javascriptbereiche beinhalten, die clientseitig vom Internetbrowser interpretiert werden, gibt es in PHP-Dateien auch, ähnlich Javascript, einen PHP-Code, der von einem PHP-Interpreter am Server ausgeführt wird. Der grundlegende Unterschied zwischen client- und serverseitigen Prozessen wird im nächsten Kapitel ausführlich behandelt.

Anwendungsschicht: HTTP, FTP, SMTP
Transportschicht: TCP, UDP
Internetschicht: IP
Netzwerkschicht, z. B. Ethernet

HTML: Hypertext Markup Language

PHP: PHP Hypertext Preprocessor (rekursive Definition)

*Wer hohe Türme bauen will,
muß lange beim Fundament verweilen.*

Bruckner, Anton, Dr. h. c.

3 HTML und Skriptsprachen

*HTML ist keine
Programmiersprache*

Gibt man sich mit der Gestaltung von einfachen für das Internet aufbereiteten Dokumenten zufrieden, reicht es völlig, mit einem (Web)Editor HTML-Dateien zu erzeugen und diese auf den Webserver zu übertragen. Das Protokoll MIME (Multipurpose Internet Mail Extensions) erlaubt es, bekannte Dokumenttypen aus bekannten Anwendungsprogrammen wie der Office-Familie (DOC, XLS, PPT-Dokumente) sowie bekannte Grafikdateien (JPG, GIF) oder PDF (Portable Data Format) zu übertragen. Diese Dateien können mit den Internetbrowsern angefordert und nach erfolgreicher Übertragung an den Client mit der entsprechenden Anwendung geöffnet werden. Grafikdateien werden im Normalfall direkt im Browser angezeigt.

Wer ernstlich daran interessiert ist, professionelle Homepages bzw. Websites zu erstellen, kommt an der grundlegenden Kenntnis und Beherrschung der Hypertext Markup Language (HTML) nicht vorbei. Damit ist auch die fachdidaktische Frage beantwortet, ob es im Informatikunterricht Sinn macht und notwendig ist, die Grundzüge dieser „Auszeichnungssprache“ zu vermitteln.

*HTML ist eine
Beschreibungssprache
(Markup Language)*

Ist HTML eine Programmiersprache? Nein, es ist eine Auszeichnungssprache, die mit einer begrenzten Anzahl von ca. 60 „tags“ (tag = Umbau) ein HTML-Dokument beschreibt. Das Beispiel eines HTML-Trainers, der selbst ein HTML-Dokument darstellt und über <http://www.schulinformatik.at> erreichbar ist, möge dies veranschaulichen.

Eine umfangreichere Interaktion im Internet, die über das Klicken von Buttons hinausgeht, bedingt eine formularorientierte Eingabe, die in der Sprache HTML über Formulare und Formularobjekte realisiert wird. Einen sehr guten Überblick über alle Formularobjekte bietet z. B. die Internetquelle von Stefan Münz, <http://www.team-one.de/selfhtml>.

Ebendort findet sich eine ausführliche Beschreibung der ebenso wichtigen Gestaltungsmöglichkeiten durch Tabellen.

Sowohl die Formular-Tags (<FORM> ...) als auch die Table-Tags können mit folgendem HTML-Trainer, der in starker Selbstbezüglichkeit aus HTML-Elementen aufgebaut ist, veranschaulicht werden.

Ein einfacher HTML-Trainer

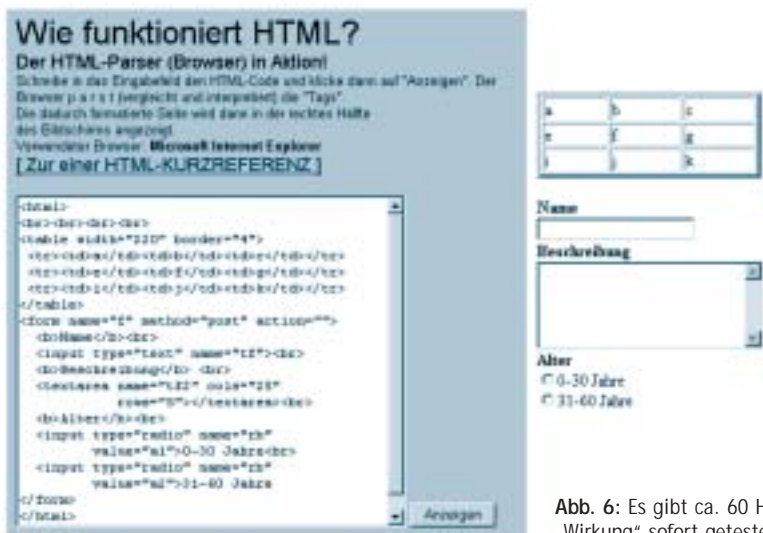


Abb. 6: Es gibt ca. 60 HTML-Tags, deren „Wirkung“ sofort getestet werden kann.

Die Struktur dieses HTML-Trainers besteht aus einem einfachen **Framekonstrukt**, das den Bildschirm in einen linken (Eingabe und Verarbeitung) und in einen rechten Teil (Ausgabe) teilt.

```
Die Datei: „index.html“

<HTML>
<FRAMESET cols=50%,50% border=0>
  <FRAME src="links.html">
  <FRAME src="rechts.html">
</FRAMESET>
</HTML>
```

Die FRAME-Technik, sparsam eingesetzt, ist noch sehr verbreitet

Die dadurch entstehenden Bildschirmbereiche müssen eindeutig mit Namen identifiziert werden

Der im linken Eingabefenster eingetragene HTML-Code wird vom Browser analysiert (geparst) und interpretiert. Das Ergebnis dieser Interpretation wird im rechten Frame angezeigt. Das Herzstück dieses HTML-Trainers ist ein JAVASCRIPT-Abschnitt (fett hervorgehoben), der alle Tags im Eingabefenster „umbaut“, d. h. interpretiert und im rechten Teil ausgibt.

Fachbereich allgemeinbildende höhere Schule

Die 16,7 Millionen Farbzahlen reichen von #000000 (schwarz) bis #FFFFFF (weiß)

Dieses Zahlen liegt das RGB (rot-grün-blau)-Modell zugrunde. Nicht alle sind „webecht“

`<textarea>...</textarea>` markiert ein mehrzeiliges Eingabefeld

In Javascript ist einfache EREIGNISSTEUERUNG möglich.
(z. B. `onclick =`)

Die Variable `inhalt` enthält die Eingaben des Benutzers in das `textarea` – Objekt

Die Methode `write` des Objekts `document`, also ... `document.write` veranlasst die vom Browser bereits geparte und übersetzte Ausgabe auf dem Bildschirm!

Die Datei: „links.html“

```
<BODY bgcolor="#66CCFF">
<FORM>
  <TEXTAREA rows=20 wrap=physical cols=45></TEXTAREA>
  <INPUT onclick=transport() type=button value=Anzeigen>
</FORM>
.....
</BODY>
</HTML>
```

Die Datei: „rechts.html“

```
<HTML>
<HEAD>
  <SCRIPT language="Javascript">
    function transport() {
      inhalt = parent.frames[0].document.
        forms[0].elements[0].value;
      parent.frames[1].document.clear();
      parent.frames[1].document.close();
      parent.frames[1].document.open();
      parent.frames[1].document.write(inhalt);
      parent.frames[1].document.write('<br></body></html>');
    }
  </SCRIPT>
</HEAD>
```

Für einen Programmier-Neuling ist dieser Quellcode mit Sicherheit nicht lesbar, er kann aber Ausgangspunkt für einen Einstieg in die objektbasierte Programmierung mit Javascript sein. Alle Javascript-Objekte wie `parent`, `frames`, `document`, `forms`, `elements` inkl. Objekteigenschaften und Objektmethoden werden mittels „Punkt“-Notation angesprochen und korrespondieren über die Ereignissteuerung (`onclick`) mit dem Formular. Es ist eine didaktische Herausforderung für den Informatik-Unterrichtenden und bedarf einer soliden Vorbereitung, dieses Listing verständlich zu machen.

Vom Client zum Server

Wir wollen nun anhand eines einfachen Problems einen „Paradigmenwechsel“ von clientseitigem Javascript-Code zu serverseitigen Skripten vornehmen. Die Aufgabe ist, den Text „Guten Tag“ mit Hilfe einer Zählschleife dynamisch dreimal auszugeben. Kein sonderlich originelles Beispiel, aber immer noch besser als nur einmal „Hello World“ anzeigen zu lassen. ;-). Um Dynamik in eine statische HTML-Seite einfließen zu lassen und automatisierte Abläufe zu programmieren, gibt es die objektbasierte Skriptsprache JAVASCRIPT, die in vielen Websites unverzichtbarer Bestandteil ist. Diese Sprache ist von der Syntax her sehr stark an C angelehnt, eine Zählschleife wird mit der Wiederholungsanweisung `for (...)` realisiert.

Micheuz: Internet inside – dynamisch und interaktiv

SERVER-FESTPLATTE

```
<HTML>
<b>Jetzt wird 3 x Guten Tag ausgegeben!</b>
<SCRIPT language="Javascript">
for (i = 1; i<=3; i++)
    { document.writeln("Guten Tag."); }
</SCRIPT>
<b>Javascript hat 3 mal Guten Tag gesagt!</b>
</HTML >
```

Dieses Dokument, besser eine Kopie des Dokuments mit Javascript-Code wird mittels HTTP zum Client übertragen.

CLIENT-FESTPLATTE

```
<HTML>
<b>Jetzt wird 3 x Guten Tag ausgegeben!</b>
<SCRIPT language="Javascript">
for (i = 1; i<=3; i++)
    { document.writeln("Guten Tag."); }
</SCRIPT>
<b>Javascript hat 3 mal Guten Tag gesagt!</b>
</HTML>
```

ANZEIGE DURCH DEN BROWSER
Javascript-Code wird interpretiert

CLIENT-BILDSCHIRM

Jetzt wird 3 x Guten Tag ausgegeben!
Guten Tag. Guten Tag. Guten Tag.
Javascript hat 3 mal Guten Tag gesagt!

Dass dem Webdesigner mit Javascript ein mächtiges Mittel in die Hand gegeben ist, Algorithmen zur Ausführung zu bringen, kann hier nur ansatzweise angedeutet werden. Links zu sehr guten ausführlichen Javascript-Kursen finden sich z. B. auf <http://www.schulinformatik.at>.

Das Ablaufschema eines Javascripts, das webbasiert den Euklidischen Algorithmus (Bestimmung des größten gemeinsamen Teilers zweier Zahlen) ermöglicht, ist in Abbildung 7 zu sehen.

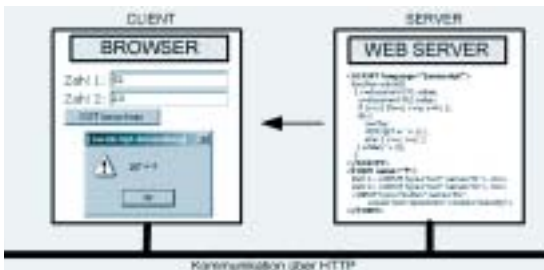


Abb. 7: JAVASCRIPT-CODE wird immer zum Client übertragen! Auch wenn der Code in einer externen Datei (Einbindung von js-Dateien) vorliegt.

Fachbereich allgemeinbildende höhere Schule

PHP, eine mächtige Skriptsprache

War es bisher die einzige Aufgabe des Webservers, das HTML-Dokument bereitzustellen und bei Bedarf eine Kopie anzufertigen und diese per HTTP auf den Client zu transfieren, wählen wir im Folgenden einen völlig anderen Zugang. Wie gehabt, wartet zwar das Webdokument auch auf dem Server auf den Abruf (request) durch den Benutzer, das Skript wird jetzt allerdings auf dem Server abgearbeitet und nur die Ergebnisse (= HTML-Format) werden zusammen mit dem statischen Anteil zum Client übertragen!

Der User sieht vom PHP-Skript überhaupt nichts – nur dessen Auswirkungen

SERVER-FESTPLATTE

```
<HTML>
<b>Jetzt wird 3 x Guten Tag ausgegeben!</b>
<?PHP
for (i = 1; i<=3; i++)
  { echo("Guten Tag"); }
?>
<b>PHP hat 3 mal Guten Tag gesagt!</b>
</HTML >
```

Dieses Dokument wird OHNE den PHP-Code mittels http an den Client übertragen (kopiert)! Das PHP-Script wird bereits auf dem Server interpretiert, der nur mehr HTML-Code mittels der Anweisung „echo“ zurück an den Client schickt!

CLIENT-FESTPLATTE

```
<HTML>
<b>Jetzt wird 3 x Guten Tag ausgegeben!</b>
Guten TagGuten TagGuten Tag
<b>PHP hat 3 mal Guten Tag gesagt!</b>
</HTML> ANZEIGE DURCH DEN BROWSER
NUR MEHR DER HTML-Code wird interpretiert
```

CLIENT-BILDSCHIRM

```
Jetzt wird 3 x Guten Tag ausgegeben!
Guten TagGuten TagGuten Tag
PHP hat 3 mal Guten Tag gesagt!
```

Damit PHP-Skripts ausgeführt werden können, benötigt der Webserver PHP-Unterstützung in Form eines „Middleware“-Moduls

Das ist der fundamentale Unterschied: Im ersten Fall benötigt man keinen Webserver, das HTML-Dokument kann lokal ausgeführt werden! (Webdokument öffnen mit jedem Webbrowser, der Javascript aktiviert hat.) Im zweiten Fall sieht das Dokument verblüffend ähnlich aus. Der geneigte Leser möge die wenigen Unterschiede gleich einem Bilderrätsel festmachen. Äußerlich ist der Unterschied die Dateieindung „php“, die signalisiert, dass es sich hier um ein serverseitiges Script handelt, also Code, der auf einem Webserver interpretiert werden muss. Um die Auswirkung dieses Dokuments (mehrfache Ausgabe durch Schleifenkonstruktion) zu sehen, ist es notwendig, dieses Webdokument auf einem Webserver zu platzieren. Auf diesem muss zusätzlich das **PHP-Modul (Interpreter)** installiert sein. Sehen wir uns das Zusammenspiel von Client und Browser bzw. Server und PHP-Interpreter graphisch an:

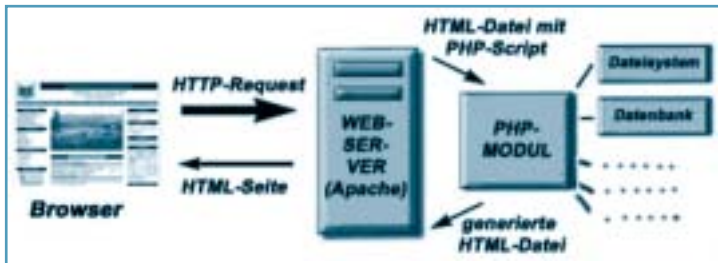
Micheuz: Internet inside – dynamisch und interaktiv

Abb. 8: PHP gibt es wie den Apache Server sowohl für Windows als auch Linux

Iss Kuhmist, Milliarden von Fliegen können sich nicht irren.

PHP – Eine Erfolgsstory sondergleichen

PHP ist alles andere als Mist! Es gehört zu einer Klasse von Sprachen, die als Middleware bezeichnet wird. Diese Sprachen wie z. B. Perl, ASP von Microsoft® oder Java Server Pages, stellen das Bindeglied zwischen dem Webserver und anderen Anwendungen wie z. B. einer Datenbank dar.

Mit PHP kann auch objektorientiert programmiert werden. Professionelle Konzepte moderner Sprachen werden in PHP laufend implementiert.

Warum ist PHP (auch für die SchulInformatik) eine gute Wahl?

- PHP ist einfach (zu lernen) und schnell
- PHP ist plattformunabhängig
- PHP ist sehr kommunikativ
- PHP wird laufend verbessert
- PHP ist sehr weit verbreitet
- Es kostet nichts
- Es arbeitet perfekt mit der Datenbank MYSQL zusammen, einer Datenbank, die sehr schnell und mächtig ist.

Die Geburtsstunde dieser Sprache war der Frühling 1995, als der Däne Rasmus Lerdorf PHP/FI 1.0 „Personal Home Page Tools/Form Interface“ entwickelte. Eine rasante Weiterentwicklung mit echtem Parser und Interpreter und grundlegenden OO-Möglichkeiten führte dazu, dass das Akronym PHP in „PHP HyperText Preprocessor“ unbenannt wurde. Anfang 2000 wurde ein neuer, schnellerer Sprachkern „Zend“ entwickelt, viele neue Funktionen kamen hinzu. Mittlerweile (Jahr 2003) verfügt die Sprache standardmäßig über mehr als 3000 Funktionen für fast jeden Einsatzbereich im Bereich der Webprogrammierung.

Die Verbreitung von PHP kann durch die Tatsache, dass sich die Anzahl der Treffer bei der Suche nach „PHP“ in Google innerhalb kurzer Zeit vervielfacht hat und nun bei 200 Millionen liegt, sehr gut dokumentiert werden

Fachbereich allgemeinbildende höhere Schule

Wo werden eigentlich die
Seitenzugriffe gezählt?

„Willst Du ein Schiff bauen, rufe nicht Menschen zusammen,
um Pläne zu machen, die Arbeit zu verteilen,
Werkzeuge zu holen und Holz zu schlagen,
sondern lehre sie die Sehnsucht, nach dem großen Meer.“

Antoine de Saint-Exupéry

4 Ein Zähler für alle Fälle

Eine fundamentale Fragestellung bei jeder Softwareentwicklung bezieht sich darauf, mit welchen Werkzeugen (Tools) das Problem gelöst werden kann. Die Entscheidung darüber, ob es bereits adaptierbare Module für diverse informatische Problemstellungen gibt oder ob eine Eigenentwicklung eventuell kostengünstiger ist, fällt nicht immer leicht. Eine Softwarefirma trifft im Normalfall jene Entscheidung, die die Kosten und damit den Ertrag optimiert. Im Informatikunterricht kann man sich den Luxus erlauben, mit einem Problem und den Lösungsmöglichkeiten zu spielen und zu experimentieren, ohne gleich an die Kosten-Nutzen-Optimierung denken zu müssen. Es sei denn, der Zeitdruck – z. B. bei einem Projekt mit einer knappen Deadline – macht es notwendig, schnell „Outsourcing“ im Sinne von Zukauf und Adaption von Fremdprodukten betreiben zu müssen.

Unser Auftrag ist, einen Seitenzähler, also einen COUNTER, in einer Homepage einzubauen, um einen Überblick über die Anzahl der Seitenbesuche als Teil einer (noch auszubauenden) Zugriffsstatistik zu gewinnen. Für jeden Homepagebetreiber ein verständlicher Wunsch.

Zum Thema „Counter“ findet man in diversen Suchmaschinen eine fast unüberschaubare Fülle von Einträgen im Millionenbereich! Als erste Übung soll ein Counter eines Fremdanbieters in die Homepage eingebaut werden. Die Qual der Wahl besteht nun darin, aus vielen Anbietern von „Zählern“ sich für einen (guten) zu entscheiden. Auf jeden Fall ist es didaktisch sehr wertvoll, die Funktionsweise eines „Fremdzählers“ unter die Lupe zu nehmen, um das Internet verstehen zu lernen.

Nach Anmelden der eigenen Homepage bei einem „Counter-Provider“ bekommt man im einfachsten Fall die Anweisung (meistens per Mail), ein HTML-Fragment wie z. B.

```
<IMG BORDER="0" SRC="http://www.webhits.de/cgi/
Count.cgi?ft=0&dd=B&df=74354.dat">
```

im gewünschten Bereich der Homepage zu platzieren. Wie ist diese Anweisung zu verstehen? Im Verzeichnis CGI (siehe oben) des Servers des Anbieters wird ein Programm „count.cgi“ mit Parametern aufgerufen, von denen der letzte, nämlich „df=74354.dat“ sich auf eine Datei bezieht, in der der aktuelle Zählerstand gespeichert ist. Dabei ist es für uns zunächst unerheblich, in welcher (Script)Sprache der Counter programmiert ist. Auf jeden Fall ist es ein so genanntes CGI-Script, das über das Common Gateway Interface veranlasst, die Datei zu öffnen, den alten Zählerstand auszulesen, diesen um eins zu erhöhen und in der entsprechenden Form (z. B. graphisch) anzuzeigen. Diese Vorgangsweise über das CGI werden wir nicht weiterverfolgen, es gibt eine andere, eigenständige Lösung dieses Problems mit der Middleware PHP.

Wir werden nun mit diesen Vorgaben einen Zähler mittels PHP realisieren.

Hier ist nicht ersichtlich,
mit welcher Programmier-
sprache die Programm-
datei count.cgi erstellt
wurde, die die Daten-
datei tei74354.dat mit
dem Zähler manipuliert.
Es könnte PERL, C,
weniger wahrscheinlich
PHP sein.

PHP in einer CGI-Version
ist nicht sehr vorteilhaft

CGI = Common Gateway
Interface
Schnittstelle, die vom
Benutzer übergebene
Daten (Daten aus dem
http-Header oder Formu-
lardaten von einem
beliebigen Programm auf
dem Server bearbeiten
lässt und das Ergebnis
über den Webserver zum
Client zurücksendet

Micheuz: Internet inside – dynamisch und interaktiv

Der **Algorithmus (Pseudocode)** für unsere eigenständige Lösung ist:

- Zugriff auf eine Textdatei, die den Zähler speichert.
- Lies den Wert dieser Datei aus.
- Erhöhe den Wert um 1.
- Schreibe den um 1 erhöhten Wert in die Datei zurück.
- Zeige den Wert an.

Der Code sieht folgendermaßen aus:

Datei: zaehler.php

```
1 <?PHP
2 $counter = file("counter.txt");
3 $counter = $counter[0] + 1;
4 $fp = fopen("counter.txt", "w");
5 fwrite($fp, $counter);
6 echo $counter;
7 ?>
```

PHP hat mächtige Funktionen zur Manipulation von Textdateien und Zeichenketten

Beschreibung:

Zeile 1: Der PHP-Code wird initialisiert (der PHP-Interpreter wird angewiesen, ab dieser Zeile den Code zu interpretieren und auszuführen, in Zeile 7 wird der Code-Teil abgeschlossen).

Zeile 2: Die Textdatei „counter.txt“, die in diesem Fall auf dem Server im gleichen Verzeichnis wie die Datei zaehler.php liegt, wird durch die Anweisung \$counter = file(„counter.txt“) ausgelesen, der Wert in der Variablen \$counter gespeichert. Bei einem Zähler besteht der Inhalt der textdatei aus nur einem Zahlenwert, dem Zähler eben.

Zeile 3: Der Wert der Variablen wird um 1 erhöht (inkrementiert).

Zeile 4: Die Datei „counter.txt“ wird mit dem Parameter „w“ (wie „write“) zum Lesen und Schreiben geöffnet. Dabei ist es sehr wichtig, dass diese Datei Schreibrechte hat! Der Betreiber des Webservers muss dies für diese Textdatei ermöglichen (Setzen des Schreibrechts).

Die Datei bekommt ein mit der Variablen \$fp (wie „filepointer“) ein „Handle“ zugewiesen, das heißt die externe Datei „counter.txt“ wird im Programm über diese Variable \$fp (eine von PHP vergebene Zahl) angesprochen. Die Bezeichnung \$fp ist frei gewählt.

Zeile 5: Der um 1 erhöhte Wert der Variablen wird in die Datei zurückgeschrieben.

Zeile 6: Der Wert der Variablen \$counter, der soeben in die Datei geschrieben und somit gespeichert wurde, wird mit der echo-Anweisung angezeigt, ausgegeben bzw. in das HTML-File eingetragen, die anschließend zum Client übertragen wird. Das hat zur Folge, dass man vom PHP-Skript am Client nichts sieht, sondern nur das Ergebnis des Skripts.

Wir können nun diesen Code im Sinne des Modulkonzepts mit der INCLUDE-Anweisung an geeigneter Stelle folgendermaßen einbinden:

Datei: „testseite.php“

```
<HTML>
.....
<?PHP include("zaehler.php") ?>
.....
</HTML>
```

Mit der INCLUDE oder REQUIRE-Anweisung kann eine PHP-Anwendung gut modularisiert werden

Fachbereich allgemeinbildende höhere Schule

Bemerkungen:

Die HTML-Dateien, die auch nur eine Zeile PHP-Code aufweisen, müssen, um vom PHP-Interpreter übersetzt zu werden, standardmäßig mit der Endung *.php enden.

Ein ehrlicher Zähler

Zähler können vom Serverbetreiber sehr leicht manipuliert werden. Kein Zählerkonzept ist perfekt. Er stellt nicht die einzige Möglichkeit dar, die Frequenz der Seitenbesuche zu ermitteln

Eine andere Möglichkeit ist, die LOG-Dateien, die vom Webserver angelegt werden, auszuwerten. Dafür gibt es professionelle Programme

Diese einfachste Lösung eines Zählers, so geeignet sie für das grundsätzliche Verständnis ist, kann nicht der Weisheit letzter Schluss sein! Was ist, wenn jemand die Seite innerhalb kurzer Zeit öfter aufruft? Die Auswertung ist sicher seriöser, wenn man das für viele Fälle unterbindet. Folgender PHP-Code stellt eine Verbesserung dar:

```

1 <?php
2 $counter="counter.txt"; $ip="ip.txt";
3 $fp=fopen($counter,"r"); $zahl= fgets($fp,10);
  fclose($fp);
4 $fp=fopen($ip,"r"); $ip_adresse = fgets($fp,15);
  fclose($fp);
5 if ($ip_adresse <> $REMOTE_ADDR)
6   { $zahl=$zahl+1;
7     echo „$zahl“;
8     $fp=fopen($counter,"w"); fwrite($fp,$zahl);
      fclose($fp);
      $fp=fopen($ip,"w");
      fwrite($fp,$REMOTE_ADDR); fclose($fp);
9   }
10 ?>

```

Es fällt sofort auf, dass in den Zeilen 2-4 eine weitere Textdatei „ip.txt“ dazukommt, die die IP-Adresse speichert. An dieser Stelle wird vorausgesetzt, dass der Begriff IP-Adresse bekannt ist. In dieser Datei wird die max. 15-stellige (xxx.xxx.xxx.xxx) aktuelle IP-Adresse gespeichert. Damit wird der Algorithmus, wie Mehrfacheingaben (in einem Stück) verhindert werden können.

In Zeile 5 findet man die HTTP-Variable \$REMOTE_ADDR, die mit vielen anderen Variablen aus dem „Header“ der Datenpakete vom Clientrechner, der (im Normalfall, wenn man sich bei einem Provider einwählt) eine weltweit eindeutige IP-Adresse hat, ausgelesen werden kann.

Windows User können diese Adresse mit dem Dienstprogramm ipconfig /all (Win 2000/XP) oder mit winipcfg (Win 98) abfragen.

Auf die Möglichkeiten, diese IP-Adresse zu verstecken, soll hier nicht weiter eingegangen werden. Das Internet stellt hier ein Fülle von Möglichkeiten bereit.

Mit einem Seitenaufruf verrät der unbedarfte Internetuser mehr, als er glaubt. Diese Informationen werden bei jeder Anfrage mit dem so genannten IP-Header in jedem TCP/IP-Paket zum Server übertragen

\$REMOTE_ADDR:

Die IP-Nummer des abrufenden Rechners. Dies kann die momentane IP-Nummer des tatsächlichen Abrufers (oft dynamisch vergeben und variabel) oder die IP-Nummer des „Proxy-Servers“ sein, den der Abrufer verwendet.

\$HTTP_REFERER:

Die URL der Seite, die auf diese Seite verwiesen hat.

*Micheuz: Internet inside – dynamisch und interaktiv***\$HTTP_USER_AGENT:**

Die Browserkennung des abrufenden Browsers.

\$HTTP_HOST:

Inhalt des Host-Headers der aktuellen Anforderung, wenn er existiert.

Datei: info.php

```

1  <? PHP
2  $info = "info.txt";
$fp = fopen($info,"a+");
fwrite($fp,"$REMOTE_ADDR\n");
5     $adr=gethostbyaddr($REMOTE_ADDR);
6         fwrite($fp,"$adr\n");
7         fwrite($fp,"$HTTP_REFERER\n");
8         fwrite($fp,"$HTTP_USER_AGENT\n");
9     fwrite($fp,"$HTTP_HOST\n");
10    fwrite($fp,".....\n");
11 fclose($fp);
12 ?>

```

Dieses PHP-Programm öffnet die Datei „info.txt“ und erweitert diese bei jedem Seitenaufruf um 5 Zeilen, die nacheinander die IP-Adresse, die zugeordnete URL (unified resource locator) bzw. Internetadresse (das besorgt die Funktion „gethostbyadress“), die Bezeichnung der aufrufenden Seite, den Typ des aufrufenden Browsers, und die URL des Servers, auf dem das Script läuft, speichert.

Datei auslesen.php:

```

<?
$info = "info.txt";
3  $fp = fopen($info,"r");
4  $inhalt = file($info);
5  foreach($inhalt as $zeile) {
6             echo trim($zeile) . "<BR>\n";
7             }

8  /* VARIANTE: $array von hinten nach vorne durchlaufen
9     $i = sizeof($inhalt);
10    while ($i-) { echo trim($inhalt[$i]) . "<br>\n"; } */
11 ?>

```

Dieses File liest die Datei „info.txt“ aus und zeigt sie zeilenweise am Bildschirm an. Damit die Zeilen auch „umbrochen“ werden, ist in Zeile 6 der HTML-Tag
 angefügt. Die Anweisung `$array = file($info)` in Zeile 4 übergibt alle Zeilen der Textdatei an eine strukturierte (indizierte) Arrayvariable:

Erste Zeile	→	inhalt[0]
Zweite Zeile	→	inhalt[1]
.....		

Mit / ... */ können Skriptteile auskommentiert werden*

Die Schleife in Zeile 5-6 „foreach“ ... liest diese Arrayvariable aus, weist den Komponenten des Arrays inhalt[0], inhalt[1], ... – die Zählung beginnt bei 0 – die Variable \$zeile zu und gibt die Zeilen nacheinander aus.

Fachbereich allgemeinbildende höhere Schule

Eine andere Möglichkeit (Variante) des Durchlaufens ist, die Zeilen direkt durch die indizierte Variable `inhalt[n]` anzusprechen.

Das auskommentierte Codefragment von 8 – 11 [C-Syntax: `/* Kommentar */`] gibt die Textdatei mit der letzten Zeile beginnend umgekehrt aus. In Zeile 9 wird die Zeilenanzahl ausgelesen und in der `while`-Schleife so lange um 1 verkleinert, bis die Zeilenzählvariable 0 ist.

Ein universeller Textfänger

Abb. 9: Einfachste Form eines übersichtlichen text- und server-basierten „Textfängers“

The screenshot shows a web browser window. On the left, there is a text input area with the prompt 'Bitte hier den Text eingeben'. Below it, a text area contains the following HTML code:


```
<h3>Eine Tabelle</h3>
<table border = 1>
  <tr><td>Zelleintrag</td></tr>
</table>
```

 Below the text area are three buttons: 'Eintragen', 'Reset', and 'Einträge rechts löschen'. On the right side of the browser, the rendered output is shown:

Ein zweiter Eintrag.

Und ein dritter!

Eine Tabelle

Zelleintrag

Die strukturgebende Datei (FRAME-Definition) : `index.htm`

```
<html>
<frameset cols="40%,*" frameborder="0" framespacing="0"
  border="0">
  <frame src="eingabe.html" name="eingabe" noresize>
  <frame src="ausgabe.php" name="ausgabe" noresize>
</frameset>
</html>
```

Die Eingabedatei: `eingabe.html`

```
1 <table border="0" width="100%" bgcolor="#CCCCCC">
2 <tr><td><h3><b>TEXTFÄNGER</b></h3></td></tr><tr><td>
3 <form action="ausgabe.php" method="POST" name="Formular"
4   target="ausgabe">
5   Bitte hier den Text eingeben:<br>
6   <textarea name="inhalt"
7     cols="40" 7 rows="6"></textarea><br>
8   <input type="submit" name="eintragen"
9     value="Eintragen"> &nbsp; &nbsp;
10  <input type="reset" name="loeschen" value="Löschen">
11 </form> </td></tr><tr><td>
12 <form action="loeschen.php"
13   method="POST" name="Formular1"
14   target="ausgabe">
15   <br><input type="submit" name="loeschen"
16     value="Alle Einträge rechts löschen">
17 </form>
18 </td></tr>
19 </table>
```

Zwei verschiedene Aktionen (Ereignisse) können ausgelöst werden, indem zwei Formulare verwendet werden

*Micheuz: Internet inside – dynamisch und interaktiv***Die Verarbeitungs- und Ausgabedatei: ausgabe.php**

```
<?php
$fp=fopen("journal.txt","a+");
if(isset($eintragen)) {
    fwrite($fp,nl2br($inhalt));
    fwrite("<hr>");
    fclose($fp);
    include("journal.txt");
}
?>
```

Diese Aktion sollte eigentlich nur dem Administrator vorbehalten bleiben und mit Passwort geschützt werden

Datei: loeschen.php

```
<?php
$fp=fopen("journal.txt","w+");
if(isset($eintragen)) {
    fwrite($fp,"");
    fclose($fp);
}
include("journal.txt");
?>
```

Dieser didaktisch extrem reduzierte Textfänger stellt bereits ein – zugegebenermaßen einfaches – Gästebuch dar. Er ist sogar ein einfacher Chat, wenn in der Texteingabe der „Nickname“ mit eingegeben wird. Es ist auch unschwer zu erkennen, dass er genauso gut als HTML-Trainer fungieren kann, wenn man (wie oben) zulässt, dass HTML-Tags aus dem Text nicht automatisch entfernt werden!

Der große Unterschied zum clientseitigen HTML-Trainer ist erstens, dass die Formulareingaben gespeichert werden, und zweitens, dass jeder im Internet die Einträge online mitverfolgen kann!

Es wird nun ein bisschen komplizierter. Wie können die Texteingaben in umgekehrter Reihenfolge ausgegeben werden, sodass die letzten Eingaben zuoberst aufscheinen? Eine nahe liegender Lösungsansatz ist, die Zeilen der Textdatei als Array von Strings auszulesen und mit dem höchsten Zeilenindex beginnend auszugeben. Nur: Dies funktioniert nicht, weil die Beiträge sich über mehrere Zeilen erstrecken (Absätze), die nicht „umgedreht“ werden dürfen.

PHP stellt für die Manipulation von Textdateien eine hohe Funktionalität zur Verfügung

Eine saubere Lösung für dieses Problem ist folgende: Der eingegebene Text wird in einer temporären (neuen) Textdatei gespeichert, die alte Datei wird zeilenweise angehängt und anschließend gelöscht. Die temporäre Datei wird in den Namen der alten Datei umbenannt.

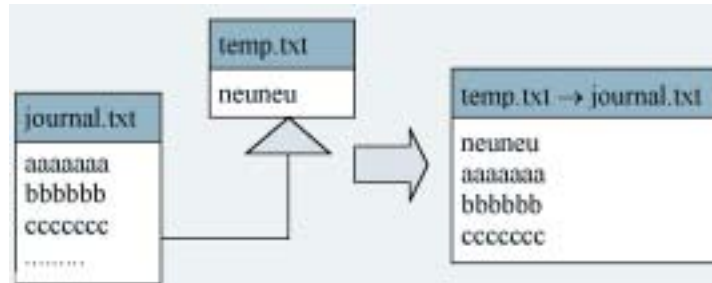
Das sieht dann so aus:

```
1 <?php
2 $fp = fopen("temp.txt","w");
3 fwrite($fp,nl2br($inhalt));
4 fwrite($fp,"<hr>");
5 $zeilen = file("journal.txt");
6 $anzahlzeilen = sizeof($zeilen);
7 for ($i=0; $i < $anzahlzeilen; $i++) {
8     fwrite($fp, $zeilen[ $i]);
9 }
10 fclose($fp);
11 unlink("journal.txt");
12 rename("temp.txt","journal.txt");
13 include("journal.txt");
14 ?>
```

Unabhängig von der Programmiersprache ist das Konzept des direkten Zugriffs auf die Dateien für viele Aufgaben wichtig

Fachbereich allgemeinbildende höhere Schule

Schematische Darstellung:



Bemerkungen:

Zeile 2: Eine Datei „temp.txt“ wird erzeugt. Dabei ist es wichtig, am Server dafür zu sorgen, dass jeder User dort auch ein so gennates „create“-Recht besitzt, um dort Dateien anlegen zu können.

Zeile 3: In diese zunächst leere Datei wird der Text des Formularobjektes (Textarea mit der Bezeichnung „inhalt“) mit Zeilenumbrüchen geschrieben (Funktion nl2br: new line to break).

Zeile 7-9: Die Textzeilen der alten Datei werden an die neue angehängt.

Zeile 11: „unlink“ heißt die PHP-Funktion zum Löschen einer Datei.

Zeile 13: Der neue Eintrag (mit den angehängten alten) wird auf dem Bildschirm ausgegeben.

Professioneller ist in jedem Fall die Datenhaltung in einer Datenbank statt in Textdateien. Siehe Kapitel 5. Die vorliegende Lösung ist aber legitim, wenn man keine Datenbank zur Verfügung hat

KOMPAKTÜBERBLICK: PHP-Sprachelemente

Die grundlegende Syntax von PHP ist wie die von Javascript sehr stark an die Programmiersprache C angelehnt. Variablen beginnen immer mit einem vorangestellten S-Zeichen. Ihr Datentyp wird bei der ersten Zuweisung festgelegt und muss (bei einfachen Aufgabenstellungen nicht unbedingt vorher definiert werden).

Der Strichpunkt (Semikolon) trennt Anweisungen.

Es gelten die grundlegenden Rechenoperatoren +,-,*,/.

Modulooperator: a % b (Rest bei der Division von a durch b) .

Verknüpfung von Zeichenketten: ein Punkt! „.“

Beispiel: \$a = „PHP“ \$b = „ist toll“ \$a.\$b liefert „PHP ist toll“.

Ein C und damit auch PHP-Spezifikum ist der Zähler:

\$i++ bedeutet \$i = \$i + 1, also die Erhöhung des Wertes der Variablen \$i um 1, entsprechend ist \$i-- gleichbedeutend mit \$i = \$i - 1

Vergleichsoperatoren

== gleich	< kleiner
> größer	<= kleiner gleich
>= größer gleich	!= ungleich

Für Basic und Pascal-Programmierer sind die Vergleichsoperatoren (gleich, ungleich) etwas gewöhnungsbedürftig

Logische Operatoren

Und: and z. B.: if (Ausdruck1 && Ausdruck2) ...

Oder: or ... z. B. if (Ausdruck1 || Ausdruck2) ...

Nicht: not ! ... logische Negation

Ablaufstrukturen

Einseitige Abfrage

If (Bedingung) { Anweisung1; Anweisung2; ... }

Zweiseitige Abfrage (Entscheidungsanweisung):

If (Bedingung) { Anweisung1; Anweisung2; ... }

else { Anweisungen ...; }

Schleifen

Zählschleife:

for (Initialisierung; Bedingung; Zustandsänderung) { ... }

Schleife mit Abbruchbedingung:

while (Bedingung) { }

Benutzerdefinierte Funktionen

Funktionen definiert man in PHP Javascript-ähnlich: Man gibt das Schlüsselwort function an, den Funktionsnamen und die Parameter, Anfang und Ende der Funktion werden wie üblich mit den geschweiften Klammern { und } bezeichnet, Rückgabewerte mit return zurückgeliefert.

Selbstgeschriebene Funktionen (userdefined functions) sind ein wesentlicher Aspekt der Modularisierung von Anwendungen

```
<?php
echo "2 mal 2 ist ".ergebnis(2);

function ergebnis($zahl)
{
    $res=$zahl*$zahl;
    return $res;
}
?>
```

Der Gültigkeitsbereich von Variablen muss sorgfältig überlegt werden

Globale, lokale Variablen

Variablen sind in PHP grundsätzlich lokal wirksam. Will man anderswo verwendete Variablen „importieren“ schreibt man vor der Verwendung global <Variablenname>;

```
<?php
echo "2 mal 2 ist ".ergebnis(2);
$res="gleich ";

function ergebnis($zahl)
{
    global $res;
    $res=$res.$zahl*$zahl;
    return $res;
}
?>
```

Fachbereich allgemeinbildende höhere Schule

Textdateien und Zeichenketten spielen in vielen Bereichen der Client-Server-Anwendungen eine herausragende Rolle (jedes PHP-Skript ist eine Textdatei!)

Funktionen zur Manipulation von Textdateien und Strings

Von den unzähligen Stringfunktionen (vergleiche z. B. [http://www.php.org]) sollen hier nur wenige aufgeführt werden, die in diesem Beitrag auch Anwendung finden.

include(„Dateiname“)	Liest den Inhalt der Datei und gibt ihn aus
fopen(„Dateiname“, „p“)	Öffnet eine Textdatei, p steht für Parameter r (nur lesen), r+ (lesen/schreiben), w(nur schreiben), a(nur schreiben, am Ende der Datei)
feof(„Dateiname“)	ist wahr, wenn Ende der Datei erreicht
fgets(„Dateiname“,n)	Liest maximal die nächsten n Zeichen
fwrite(„Dateiname“,Sz)	Schreibt den Inhalt von Sz in die Datei
fclose(„Dateiname“)	Schließt die Datei
\$z = file(„Dateiname“)	Liest den Inhalt der Textdatei aus und übergibt ihn in eine Arrayvariable \$z[i].
\$s = substr(\$z,pos,anz)	Gibt eine Teilzeichenkette von \$z aus, beginnend an der Stelle „pos“ mit „anz“ Zeichen.

*Ich höre und ich vergesse,
ich sehe und ich erinnere mich,
ich tue und ich verstehe.
(Chinesisches Sprichwort)*

5 Von der Textdatei zur Datenbank

Textdateien haben eine Struktur

Ausgangspunkt unserer nächsten Überlegungen ist folgende Schülerdatei „schueler.txt“, die auf dem Webserver liegt.

```
1A Franz Huber
1A Berta Maier
1B Susi Mustermann
1A Herbert Gritsch
1B Silvia Steinwender
.....
```

Der Inhalt einer Textdatei ist schnell ausgegeben

Unsere Aufgabe ist, auf einer Homepage den Inhalt dieser Datei anzuzeigen, diese Liste nach Klassen sortiert anzugeben und Klassenlisten zu generieren. Die erste Aufgabe ist wirklich nicht schwierig zu lösen!

Anzeige aller Schüler(innen):

```
<? readfile(„schueler.txt“); ?>
oder alternativ:
<? include(„schueler.txt“) ?>
```

Dabei wird die Textdatei von der Platte des Webservers zum Client übertragen und dort in einem Stück ausgegeben. Die Einflussnahmen auf das Erscheinungsbild sind sehr gering.

Wir wollen diese Lösung verfeinern, indem wir diese Textdatei in Tabellenform ausgeben. Dazu ist aber ein Mehraufwand notwendig.

Dieser Mehraufwand äußert sich in der zeilenweisen Abarbeitung der Textdatei, der entsprechenden Zerlegung der Zeilen und der Ausgabe in Tabellenform, die eine grundlegende Kenntnis von HTML-Tags voraussetzt.

Micheuz: Internet inside – dynamisch und interaktiv

1A	Franz Huber
1A	Berta Maier
1B	Herbert Müller
1B	Susi Mustermann
1A	Herbert Gritsch
1B	Silvia Steinwender

Für die Lesbarkeit ist doch ein Mehraufwand notwendig. Tabellenstrukturen lassen sich mit den HTML-Tags <table>, <tr>, <td> realisieren

Wir müssen die Textdatei öffnen, Zeile für Zeile lesen, in diesen Zeilen die Klasse - ersten zwei Zeichen - und den Namen mittels „substr(..)“ auslesen und in Tabellenform anzeigen, bis das Dateiende erreicht ist.

Der erforderliche PHP-Code:

```
<?
$liste=fopen("schueler.txt","r");
echo("<table border=0 cellpadding=1 cellspacing=0 >");
$i=0;
while (!feof($liste)) {
    $zeile=fgets($liste,50);
    $kl = substr($zeile,0,2);
    $name = substr($zeile,3,50);
    $i=$i+1;
    echo("<tr><td>$i</td><td>\".$kl.\"</td><td>\".$name.\"</td><tr>");
}
fclose($liste);
echo("</table>");
?>
```

Das Technik des zeilenweisen Abarbeitens einer Textdatei ist keine Erfindung von PHP, sondern findet sich analog in anderen Programmiersprachen wieder. „Solange das Ende der Datei nicht erreicht ist, ...“ Die Textdatei kann auch mit der Funktion file (...) ausgelesen werden (s. o)

Ist die einfache Anzeige der Schülerdaten aus einer Textdatei heraus noch relativ einfach, gestaltet sich die Lösung komplexerer Aufgabenstellungen, wie die sortierte Ausgabe einer Textdatei, als nicht mehr trivial.

Wir führen deshalb einen weiteren „Paradigmenwechsel“ durch und wenden uns dem Kapitel **Datenbanken** zu. Die Datenhaltung in Datenbanken gibt uns ein mächtiges Mittel in die Hand, strukturierte Daten effizient zu manipulieren. Durch den Einsatz eines Datenbanksystems wird sehr viel Logik und Funktionalität in eine andere Sprache, nämlich SQL (=„structured query language“) verlagert.

„Paradigmenwechsel“ gab es im Laufe der Informatikgeschichte viele. PARADIGMA = Muster, Modell

In weiterer Folge werden wir mit einer relationalen SQL-Datenbank arbeiten, die mit PHP besonders gut zusammenspielt und nicht nur für den schulischen Einsatz hervorragend geeignet ist. Das ist MYSQL.

Das Datenbanksystem MYSQL ist nicht nur für schulische Zwecke sehr gut geeignet, sondern steht im professionellen Einsatz und wird stets weiterentwickelt.

An dieser Stelle sei u.a. auf die Homepage <http://www.mysql.org> verwiesen.

KLASSE	VORNAME	NACHNAME
1A	Franz	Huber
1A	Berta	Maier
1B	Herbert	Müller
1B	Susi	Mustermann
1A	Herbert	Gritsch
1B	Silvia	Steinwender

Fachbereich allgemeinbildende höhere Schule

Datenbanken bestehen aus (zweidimensionalen) Tabellen, diese wiederum aus Datensätzen. Die Datensätze (Zeilen) bestehen aus Feldern (Spalten)

Wie kommen die Daten in eine Tabelle (bzw. mehrere Tabellen) einer Datenbank? Dies geschieht entweder direkt am Datenbankserver oder durch den User clientseitig über eine Webschnittstelle. Sind strukturierte Daten aus einer anderen Anwendung vorhanden und zu importieren, bietet jede Datenbank und somit auch MYSQL die Möglichkeit des Datenimports. Die Beschreibung (z. B. mit PHPMYADMIN, s.u.), wie eine Datenbank in MYSQL angelegt wird bzw. in dieser Datenbank TABELLEN mit den entsprechenden FELDERN, wird findet z. B. www.schulinformatik.at.

Wir gehen davon aus, dass innerhalb des RDBMS (relationalen Datenbankmanagementsystems) MYSQL bereits eine Datenbank mit der Bezeichnung SCHULE angelegt ist. In dieser Datenbank existiert obige Tabelle SCHUELER mit den drei Feldern

KLASSE, VORNAME, NACHNAME.

Datenbanken, die aus nur einer Tabelle bestehen, sind die Ausnahme. Aus didaktischen Gründen ist es sinnvoll, mit Eintabellen-Aufgaben zu beginnen

Datenbank schule - Tabelle schueler auf localhost

Struktur Anzeigen SQL Teilw. anzeigen Einfügen Exportieren Operationen

Zeige Datensätze 0 - 5 (0 insgesamt)

SQL-Befehl: [Ändern] [SQL erklären] [PHP-Code erzeugen]
 SELECT * FROM `schueler` ORDER BY `klasse` ASC LIMIT 0, 30

Zeige: 30 Datensätze, beginnend ab 0

[Untereinander] angeordnet und wiederhole die Kopfzeilen nach 100 Datensätzen.

	klasse	vorname	nachname
Ändern Löschen	1A	Franz	Huber
Ändern Löschen	1A	Barth	Maier
Ändern Löschen	1A	Herbert	Gritsch
Ändern Löschen	1B	Herbert	Müller
Ändern Löschen	1B	Susi	Mustermann
Ändern Löschen	1B	Silva	Steinwender

PHPMYADMIN ist ein hervorragendes Tool zur Verwaltung der Datenbank MYSQL über ein Webinterface

Das ist ein Blick auf die Benutzerschnittstelle von PHPMYADMIN, einem sehr mächtigen Verwaltungstool für die MYSQL-Datenbank. Dieses Programm ist selbst in PHP geschrieben und ein ausgezeichnetes Webinterface. Diese Software muss auf einem Webserver installiert und konfiguriert werden und ist nach erfolgreicher Installation für Benutzer intuitiv zu bedienen. Das Anlegen, Verwalten und Abfragen von Datenbanken und Tabellen sowie der Datenimport und -export werden unterstützt.

Anhand dieser überschaubaren Tabelle werden wir alle wesentlichen Aspekte des Zusammenspiels von HTML, PHP und MYSQL kennen lernen.

Aufgabe

Der Inhalt der Datenbanktabelle soll – ohne Formatierung, wie zu Beginn des Kapitels – angezeigt werden.

Micheuz: Internet inside – dynamisch und interaktiv

```

1 <?php
2   echo("ALLE SCHÜLER<br><br>");
3   $verbindung = mysql_connect("localhost","root","");
4   mysql_select_db("schule");
5   //SQL - Abfrage wird formuliert
6   $abfrage="select klasse,vorname,nachname from schueler;";
7   //SQL - Abfrage wird in der Datenbank abgesetzt
8   $tabelle = mysql_query($abfrage);
9   //Datensätze werden aus der Abfragetabelle ausgelesen
10  //datensatz = sog. Assoziatives ARRAY (indiziert!)
11  while($datensatz = mysql_fetch_row($tabelle)) {
12      echo($datensatz[0]." ".$datensatz[1]." ".
13          $datensatz[2]."<br>"); }
14  mysql_close($verbindung);
15 ?>

```

Listing Datenbankzugriff

Der Zugriff auf eine Datenbank folgt einer klaren Abfolge:

- 1) Datenbankserver wählen, Anmelden mit Benutzername und Passwort,
- 2) Datenbank wählen
- 3) Anfrage (SQL-Statement) formulieren
- 4) Das Ergebnis verarbeiten

Die Zeilen, die mit einem „//“ beginnen, sind Kommentarzeilen und werden vom PHP-Interpreter ignoriert.

Damit der Webserver die strukturierten Daten aus der Tabelle der Datenbank an den Client schicken kann, muss über PHP zuerst Verbindung mit dem Webserver aufgenommen werden. Dies passiert durch die Anweisung in Zeile 2. Statt „localhost“ könnte hier auch eine IP-Adresse oder die URL des Datenbankservers stehen. Wenn das PHP-Script auf dem Webserver liegt, auf dem auch der Datenbankserver MYSQL läuft, reicht die Bezeichnung „localhost“. Die Verbindung erfolgt immer über eine Authentifizierung, wobei der User – in diesem speziellen Fall – „root“ ist und kein Passwort vorgegeben ist. Der User „root“ hat mit dem Linux-Superuser – in vielen Fällen ist der Webserver ein Linux-Server – „root“ überhaupt nichts zu tun, sondern ist im System MYSQL in der Datenbank „mysql“ in der Tabelle „user“ angelegt. Hier wird auf eine kurze Einführung in das Rechtssystem von MYSQL in der MYSQL-Dokumentation verwiesen. In Zeile 4 wird mit der Anweisung `mysql_select_db(„schule“)`; die Datenbank ausgewählt und mit ihr Verbindung aufgenommen. Zeile 6: Hier wird in einem String die eigentliche SQL-Abfrage formuliert.

Erst durch die Anweisung `mysql_fetch_row($tabelle)` wird der erste Datensatz ausgelesen und in einer Arrayvariablen gespeichert. Hier werden die durch die SQL-Anweisung eindeutig festgelegten Tabellendaten zeilenweise in ein indiziertes, assoziatives Array eingelesen. Man kann die einzelnen Felder entweder über den Index beginnend bei 0 und korrespondierend mit dem SELECT-Statement ansprechen: `$datensatz[0]`, `$datensatz[1]`, `$datensatz[2]` oder direkt über den Feldnamen `$datensatz[„klasse“]`, `$datensatz[„vorname“]`, `$datensatz[„nachname“]`.

Assoziativ deshalb, weil die Feldnamen und die Inhalte übereinstimmen. Wenn die gesamte Tabelle abgearbeitet werden soll, ist das einfach über eine Schleifenstruktur (Wiederholungsanweisung) zu lösen.

In einer Datenbank gibt es zusätzlich zur Zeilenstruktur (= Datensätze) noch die klare Spaltenstruktur innerhalb der Zeilen (= Felder).

Die Anbindung einer Datenbank und damit der Zugriff auf strukturierte Daten folgt einer ähnlichen Logik wie der Zugriff auf Textdateien. Wir werden sehen, dass der Einsatz von Datenbanken in jedem Fall eine lohnende Sache ist.

Auf dem Webserver muss nicht unbedingt auch ein Datenbankserver laufen. Der (dislozierte) Datenbankserver muss in diesem Fall z. B. über die URL oder über die IP-Adresse angesprochen werden können

Fachbereich allgemeinbildende höhere Schule

Jede Spalte (FELD) soll eine sinnvolle Bezeichnung haben. Eine Zeile (DATENSATZ, RECORD) sollte eine zusammenhängende Informationseinheit über beliebige Objekte darstellen

KLASSE	VORNAME	NACHNAME
1°	Franz	Huber
1A	Berta	Maier
1B	Herbert	Müller
1B	Susi	Mustermann
1A	Herbert	Gritsch
1B	Silvia	Steinwender

Die der Datenbank innewohnende Logik in Form der beschreibenden Abfragesprache SQL macht das Arbeiten selbst mit großen Datenbeständen sehr effizient. Der Zugriff auf einen Datensatz erfolgt hier normalerweise über ein so genanntes Schlüsselfeld, an Hand dessen ein Datensatz eindeutig identifizierbar ist. In unserem obigen Beispiel können wir z. B. den vierten Datensatz mittels des Nachnamens (weil eindeutig), aber nicht mit dem Vornamen (siehe Herbert) und schon gar nicht mit der Klasse eindeutig identifizieren. Für eine gesamte Schülerliste einer Schule ist auf jeden Fall ein Kunstschlüssel (eine Schülernummer, eine Schüler-ID) zu empfehlen.

Auf den vierten Datensatz kann mit der SQL-Abfrage:
 select * from schueler where nachname= „Mustermann“ zugreifen.

Im PHP-MYSQL System sieht das folgendermaßen aus:

```
$abfrage = "select klasse,vorname, name from schueler";
$tabelle = mysql_query($abfrage);
$datensatz = mysql_fetch_row($tabelle);
```

SQL (structured query language) ist ein Standard, der von allen namhaften Datenbanken unterstützt wird

Mit \$datensatz[0] bis \$datensatz[2] (Indexzählung beginnt immer bei 0) oder mit \$datensatz[„klasse“], \$datensatz[„vorname“], ... können wir die einzelnen Feldinhalte auslesen.

Architektur von dynamischen, datenbankgestützten Web-Anwendungen:

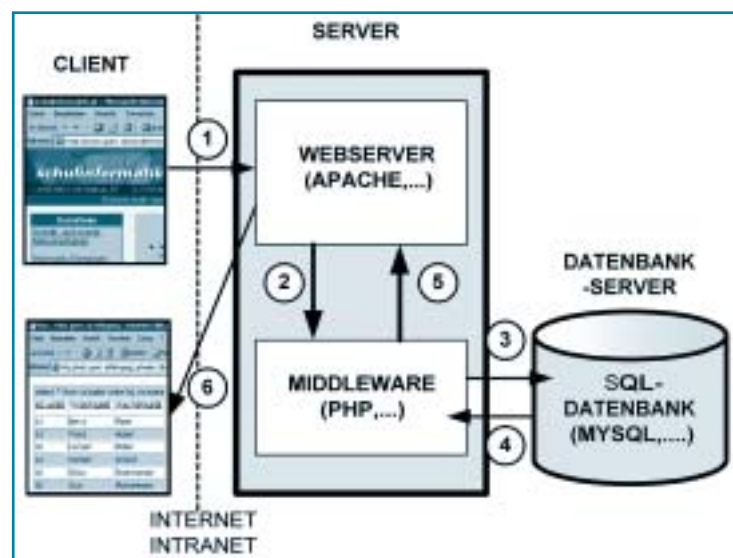


Abb. 10: Das Zusammenspiel des WEBSERVERS, PHP und MYSQL

*Micheuz: Internet inside – dynamisch und interaktiv***Die Abfolge:**

- 1: Der Client fordert eine Seite mit PHP-Code vom Webserver an.
- 2: Der Webserver erkennt den PHP-Code der angeforderten Seite und bindet die Middleware, in unserem Fall den PHP-Parser ein, der den PHP-Code interpretiert und ausführt.
- 3: Sind Befehle für die Datenbankanbindung enthalten, wird der SQL-Datenbankserver aktiv ...
- 4: ... und liefert die Ergebnisse an den PHP-Parser zurück.
- 5: PHP übergibt die Daten aus der entsprechenden Datenbank mit HTML-Tags, die für das Design der Anzeige zuständig sind, an den Webserver zurück.
- 6: Der Webserver sendet nur mehr HTML-Code (und eventuell auch Javascript-Code) an den Client zurück, wo der Browser interpretierend aktiv wird.

Das Verständnis der Datenbankanbindung ist für die Erstellung von Anwendungen sehr wichtig

Bemerkung:

Die Entscheidung für das System APACHE – PHP – MYSQL in diesem Beitrag gründet sich auf die für Schulen nicht zu unterschätzende Tatsache, dass alle Komponenten den Bestimmungen von „open source“-Software unterliegen und somit kostenlos sind.

Als Middleware kann statt PHP auch ASP (Active Server Pages von Microsoft) oder JSP (Java Server Pages) verwendet. Bei den SQL-Datenbanken könnte auch MS SQL oder ORACLE etc. eingesetzt werden.

In diesem Beitrag wird sehr viel Wert darauf gelegt, das Konzept der Client-Server. Anwendungen transparent zu machen. Ein Umstieg auf andere vergleichbare Systeme sollte kein unüberwindbares Problem sein.

LAMP: Linux-Apache-MYSQL-PHP hat in der Windowswelt die Entsprechung: WAMP

*Ich höre und ich vergesse,
ich sehe und ich erinnere mich,
ich tue und ich verstehe
(Chinesisches Sprichwort)*

SQL ist eine beschreibende (deklarative) Sprache. Die innewohnenden Algorithmen sind für den Anwender unsichtbar

6 Praktische Beispiele

Nach etwas Theorie wollen wir uns wieder der Praxis zuwenden und dabei sehen, welche Vorteile der Umstieg von Textdateien auf Datenbanken bringt.

SQL ist eine deklarative, also beschreibende Sprache, in der die gewünschte Auswertung, also das „WAS“ beschreibend formuliert wird. Über das „WIE“ braucht man sich keine Gedanken zu machen. Die effizienten Algorithmen sind Teil der Datenbank und werden je nach Abfrage ausgeführt. Das Problem zum Beispiel, die Schülerliste mit Nachname und Klasse sortiert auszugeben, ist in SQL sehr einfach formuliert:

Die Anweisung: „select nachname, klasse from schueler order by nachname“ liefert das gewünschte Ergebnis. Im Listing Datenbankzugriff ist nur die Zeile beginnend mit Sabfrage=... auszutauschen.

Und wir sind fast fertig. Da nur 2 Felder ausgegeben werden, ist \$datensatz[2] natürlich ersatzlos zu streichen!

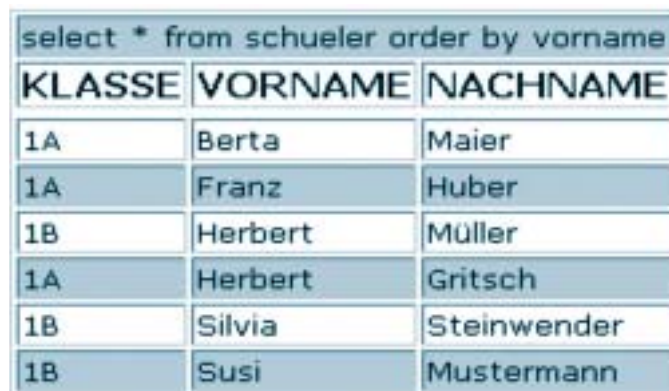
Fachbereich allgemeinbildende höhere Schule

Generische Funktionen sind vielseitig einsetzbar, das Rad muss nicht immer neu erfunden werden

Im Rahmen der Objekt-orientierten Betrachtung ist dies eine Methode

Um für unterschiedliche Tabellen und verschiedene Anfragen nicht immer den Quellcode anpassen zu müssen, wird im Folgenden eine „generische“ parametrisierte Funktion vorgestellt, die eine beliebige Abfrage mittels Aufruf einer benutzerdefinierten Funktion (userdefined function) ausgibt.

Als Parameter werden in unsere Eigenbau-Funktion „anzeige(str datenbank, str abfrage)“ nur die Bezeichnung der Datenbank und die SQL-Abfrage übergeben. Die Anweisung `anzeige(„schule“, „select * from schuler order by vorname“)` liefert die nach Vorname sortierte Ansicht der Tabelle.



KLASSE	VORNAME	NACHNAME
1A	Berta	Maier
1A	Franz	Huber
1B	Herbert	Müller
1A	Herbert	Gritsch
1B	Silvia	Steinwender
1B	Susi	Mustermann

Abb. 11: Das ansehnliche Ergebnis einer einfachen SQL-Abfrage, die nach Vornamen sortiert ist.

Damit lässt sich auf einfache Art und Weise jede Tabelle und Abfrage, deren Ergebnis ja wieder eine Tabelle ist, darstellen und platzieren.

Der Quellcode, der diese brauchbare Ausgabe (Abbildung 11) liefert:

```

1  <?php
2  function anzeige($datenbank,$abfrage)
3  {
4  ?>
5  <STYLE TYPE="text/css">
6      td.he { font-size: 10pt; font-family: verdana;
7              background-color:white }
8
9      td.du { font-size: 10pt; font-family: verdana;
10             background-color:lightgrey }
11 </STYLE>
12 <?
13 $verbindung = mysql_connect("localhost","root","");
14 mysql_select_db($datenbank);
15 $datensatz = mysql_query($abfrage);
16 $anzahlfelder=mysql_num_fields($datensatz);
17 echo("<table border=1>");
18 echo("<tr><td class=' du' align=center
19         colspan = ``.$anzahlfelder.``
20         <font face=' Verdana'>$abfrage</font></td></tr>");
21 echo("<tr>");
22 while ($feldname=mysql_fetch_field($datensatz)) {

```

Stylesheets sind ein mächtiges Mittel, das Erscheinungsbild von HTML-Seiten anzupassen

PHP ist mit einer Vielzahl von MYSQL-spezifischen Funktionen ausgestattet

Micheuz: Internet inside – dynamisch und interaktiv

```

19     echo "<th>".strtoupper($feldname->name)."</th>";
20 }
21 echo("<tr>");
22 $schalter = 0;
23 while($zeilen = mysql_fetch_row($datensatz)) {
24     echo("<tr>");
25     if ($schalter==0) {
26         for($i=0; $i < $anzahlfelder; $i++) {
27             echo "<td class=' he'>$zeilen[ $i]</td>";
28             $schalter = 1;
29         }
30     } else {
31         for($i=0; $i < $anzahlfelder; $i++) {
32             echo "<td class=' du'>$zeilen[ $i]</td>";
33             $schalter = 0;
34         }
35     }
36     echo("</tr>");
37 }
38 echo("</table>");
39 mysql_close($verbindung);
40 }
41 anzeige($db,$anfrage);
42 ?>

```

Zustandsvariablen (hier die Variable `$schalter`) sind kein PHP-Spezifikum, diese Technik gibt es, seit es Programmiersprachen gibt.

Die wesentlichen neuen Teile dieses Listings finden sich in

Zeile 5-8: Hier werden für die Ausgabe der Datensätze CSS-Angaben (Cascading Style-sheets) definiert, die für jede Zelle der Tabelle (TD ... table data - TAG) die Angaben für Schriftgröße, Schrifttyp und Hintergrundfarbe definieren.

Zeile 13: Die PHP-Funktion `mysql_num_fields(...)` liefert die Anzahl der Felder einer Tabelle oder Abfrage zurück.

Zeile 18, 19: `while ($feldname=mysql_fetch_field($datensatz))`
`{ echo „<th>“.strtoupper($feldname->name)."</th>"; }`

Diese Anweisungen liefern die Feldnamen als Überschriften der einzelnen Spalten (TH-Tag; TD-Tag mit Eigenschaft `fett, zentriert`). Damit wird möglich, dass die Funktion sich dynamisch der Anzahl der Felder anpasst.

Zeile 22-35: Hier sorgt die Schaltervariable `$schalter`, deren Wert zwischen 0 und 1 umschaltet, dafür, dass die Ausgabezeilen zur besseren Unterscheidbarkeit alternierend die Hintergrundfarbe wechseln.

Zeile 41: `anzeige(„schule“,„select * from schueler;“);`

Hier wird die Funktion mit den Parametern Datenbank und Abfrage aufgerufen. Diese Anweisung könnte durchaus auch am Anfang dieses Quellcodes stehen, PHP-Funktionen können an beliebiger Stelle stehen. Man nennt diese Art der Interpretierung durch PHP „late binding“.

Abbildung

PHP-Funktionen können an beliebiger Stelle des Skripts stehen

Ein SQL-Trainer im Eigenbau

Nun bedeutet es einen ganz kleinen Schritt, einen SQL-Trainer im Eigenbau vorzustellen. Aufbauend auf der obigen leistungsfähigen Funktion „anzeige(..., ...)“ können wir ein einfaches Formular vorschalten, mit Frametechnik wie bei unserem Textfänger die SQL-Statements in einem Formular absetzen und die Ergebnisse aus der Datenbank übersichtlich darstellen lassen.

Eine Dokumentation findet sich auf <http://www.schulinformatik.at>.



Abb. 12: Das Ergebnis jeder SQL-Abfrage ist wieder eine Tabelle!

Die where-Klausel filtert Daten nach vorgegebenen Kriterien (kann auch komplexerer logischer Ausdruck sein)

Man beachte, dass das Feld in group by-Klausel mit dem select-statement korrespondieren muss. In diesem Fall ist es „Klasse“

Außer „count“ gibt es vier weitere so genannte Aggregatfunktionen (über Zahlenfelder):
 sum (Summe)
 avg (Mittelwert)
 max (Maximum)
 min (Minimum)

z. B. die Klassenlisten interessieren, also die Filterung nach der Klasse 1A, 1B usw.

Die Abfrage „select * from schueler where klasse='1A'“ würde die Schüler der „1A“ anzeigen.

Gruppierungen

Bei vielen Auswertungen geht es darum, große Datenmengen zu „verdichten“ um aus ihnen neue, kompakte Informationen zu erhalten. Das geschieht häufig über sogenannte „Gruppenbildungen“ über entsprechende (Schlüssel-)Felder. Uns interessieren z. B. die Klassengrößen, d. h. eine (Abfrage-)Tabelle, die nur die Klassen und die Schülerzahlen ausgibt. Die Lösung ist in diesem Fall die „group by“-Klausel.

Grundlegende Auswertungen:

Filtern von Daten – Datenselektion








Bei einem großen Datenbestand ist es wichtig, die Daten nicht nur zu sortieren, d. h. sie neu anzuordnen (siehe order by-Anweisung in Abbildung 12), sondern sie nach gewissen Kriterien zu filtern bzw. zu selektieren. So könnten uns



Veranschaulichung durch horizontale Balkendiagramme

Ein Bild (Diagramm) sagt mehr als tausend Worte. Ergänzend zur Ausgabe der Zahlen ist es immer von Vorteil, diese mittels eines geeigneten Diagramms zu veranschaulichen. Eine sehr elegante, weil intelligente, einfache Lösung ist, sich das Image-Tag `` mit dem Parameter „width“ zu Nutze zu machen.

Ein kleiner HTML-Trick lässt auf einfache Weise horizontale Balkendiagramme anzeigen

klasse	anzahl	
1a	28	
1b	29	
1c	27	
1d	24	
1e	22	
2a	28	
2b	27	

```

1  $verbindung = mysql_connect("localhost","root","");
2  mysql_select_db(„workshop“);
3  $abfrage = "select klasse,count(*) as anzahl
              from schueler group by klasse;";
4  $tabelle = mysql_query($abfrage);
5  $gesanzahl = mysql_num_rows($tabelle);
6  echo("<table>");
7  while($datensatz = mysql_fetch_row($tabelle)) {
8      echo("<tr>");
9      for($i=0; $i < mysql_num_fields($tabelle); $i++) {
10         echo "<td>$datensatz[ $i]</td>";
11     }
12     $breite = $datensatz[ 1] /$gesanzahl*200;
13     echo "<td width=' 350'><img src=' green.gif'
14         height=' 10' width=' $breite'></td>";
15     echo("</tr>");
16 } echo("</table>");

```

Vieles in diesem Skript ist bereits bekannt. Die Abfrage liefert 2 Felder zurück, wobei anzumerken ist, dass das berechnete Feld, nämlich die Aggregatfunktion `count(*)` den Aliasnamen „anzahl“ erhält. In Zeile 9 wird eine Schleife über alle durch das `select`-Statement – in diesem Fall nur zwei! – gebildet.

In SQL-Abfragen werden zur besseren Lesbarkeit oft ALIAS-Bezeichnungen verwendet

`<td>$datensatz[0]</td><td>$datensatz[1]</td>` hätten es auch getan.

Es ist aber kein Fehler, wenn man die Programmierung „generischer“ anlegt, also in diesem Fall unabhängig von der Feldanzahl.

Das Highlight, nämlich die horizontale Balkendiagrammdarstellung, wird in Zeile 11 dargestellt. Hier wird eine minimale Grafikdatei „green.gif“, die aus nur einem „grünen“ Pixel besteht, mit den Parametern „height“ und „width“ auf die entsprechende Höhe und Breite (abhängig von der Anzahl, siehe Zeile 10) skaliert.

Mit der hier dargestellten Technik lassen sich viele Tabellen über das Web sehr einfach

Fachbereich allgemeinbildende höhere Schule

und effizient auswerten. Man bedenke dabei den Vorteil, z. B. statistische Auswertungen der Schülerdaten einer Schule, die man am Schuljahresanfang in die Datenbank einspielt, immer am aktuellen Stand zu halten.

Siehe auch Schulhomepage <http://www.gym1.at> unter Statistik.

Eine interaktive Gruppeneinteilung

Mit diesem Formular können Gruppeneinteilungen über weite Entfernungen durchgeführt werden

Anfügen: Gruppe wählen
 Löschen: Namen und keine Gruppe
 Gruppenwechsel: Namen und andere Gruppe

Name:

Gruppe 1
 Gruppe 2

Gruppe 1	Gruppe 2
Arno Dengg	Alois More
Gebetsberger Andrea	Elke Anderwald
Gertraud Scherwitzl	Ingrid Krassnig
Ina Anderwald	Michaela Krainer
Karin Thurner	Mirjam Schönlaub
Kurt F. Scherwitzl	Siegfried Truschner
Mario Moritz	Uta Hörandner
Norbert Sapper	

Eingabeformular und Ausgabetable kombiniert

Dieses Beispiel zeigt, wie man mit relativ wenig Aufwand das Gruppenteilungsproblem (vor allem über große Entfernungen) lösen kann. Jede/r Teilnehmer/in sieht im Internet die Einteilung vor sich und kann sich eintragen, einen Wechsel in eine andere Gruppe vornehmen oder sich wieder austragen.

Die Basis dieser reizvollen Aufgabe – man muss nicht immer von Problemen reden – ist die denkbar einfache Tabelle „netzwerk“ in der Datenbank „lehrgang“. Dieses Programm hat den Praxistest im Rahmen einer Veranstaltung zum Thema Netzwerk sehr gut bestanden.

Die zugrunde liegende Tabelle

EIN EINFACHER SQL-TRAINER

DATENBANK:

SELECT STATEMENT:

ABFRAGE:
 select * from netzwerk;

NAME	GRUPPE
Norbert Sapper	1
Gertraud Scherwitzl	1
Kurt F. Scherwitzl	1
Uta Hörandner	2
Mirjam Schönlaub	2
Mario Moritz	1
Gebetsberger Andrea	1
Elke Anderwald	2
Siegfried Truschner	2

Und nun zum Code, der in einen sehr kurzen Eingabeteil (das Formular), einen Ausgabeteil (die Tabelle) und den Datenbankmanipulationsbereich gliedert ist.

*Micheuz: Internet inside – dynamisch und interaktiv***// DAS EINGABEFORMULAR**

```

1 <h3><b>Eine interaktive Gruppeneinteilung</b></h3>
2 Anfügen: Gruppe wählen<br>
  Löschen: Namen und keine Gruppe<br>
3 Gruppenwechsel: Namen und andere Gruppe<br>
4 <form method="post" action="<? echo $PHP_SELF; ?>">
5   Name: <input type="text" name="name" size="30"><br>
6   <input type="radio" name="gruppe" value=1> Gruppe 1 <br>
7   <input type="radio" name="gruppe" value=2> Gruppe 2 <br>
8   <input type="submit"
9     name="Button" value="Daten absenden">
10  <input type="reset" name="Button2" value="Reset">
11 </form>

```

Das klassische EVA-Prinzip in einem sehr modernen Gewand

//DATENBANKMANIPULATIONSTEIL(ANFÜGEN, ÄNDERN, LÖSCHEN)

```

11 <?
12 if (isset($name)) {
13   $db = mysql_connect("localhost","root","")
14     or die("Keine Verbindung");
15   mysql_select_db("lehrgang",$db)
16     or die("Keine Verbindung mit DB");
17   $anfuegen = "insert into netzwerk (name,gruppe)
18     values ('$name','$gruppe)";
19   $aendern = "update netzwerk set gruppe=$gruppe
20     where name = '$name' ";
21   $loeschen = "delete from netzwerk where name = '$name' ";
22   $anzahl = mysql_query("SELECT * FROM netzwerk
23     where name = '$name'", $db);
24   $anzahlzeilen = mysql_num_rows($anzahl);
25   if ($anzahlzeilen > 0) {
26     if ($gruppe) {
27       mysql_db_query("lehrgang", $aendern, $db)
28         or die("Fehler: Ändern");
29     } else {
30       mysql_db_query("lehrgang", $loeschen, $db)
31         or die("Fehler: Löschen"); }
32   } else {
33     if ($gruppe) {
34       mysql_db_query("lehrgang", $anfuegen, $db)
35         or die("Fehler: Anfügen");
36     }
37   }
38 }
39 >
40 // DIE TABELLARISCHE AUSGABE MIT SQL-SELECT
41 <table border="1">
42 <tr><td>Gruppe 1</td><td>Gruppe 2</td></tr>
43 <tr>

```

SQL kann mehr als nur Anfragen beantworten

INSERT: Ein-/Anfügen
UPDATE: Ändern
DELETE: Löschen

Fehlermeldungen des Interpreters PHP können mit der Funktion „die“ genauer spezifiziert werden

Fachbereich allgemeinbildende höhere Schule

```

37 <td> <? $db = mysql_connect("localhost", "root", "");
      $query = "select name from netzwerk where gruppe=1
              order by name";
38     $rs = mysql_db_query("lehrgang", $query, $db);
39     while ($row = mysql_fetch_row($rs)) {
              echo $row[0]."<br>"; } ?>
40 </td>
41 <td>
42     <? $query = "select name from netzwerk
              where gruppe=2 order by name";
42     $rs = mysql_db_query("lehrgang", $query, $db);
43     while ($row = mysql_fetch_row($rs)) {
              echo $row[0]."<br>"; }?>
44 </td></tr>
45 </table>

```

Mit diesem Beispielskript treffen wir „viele Fliegen mit einem Schlag“. Das EVA-Prinzip zeigt sich hier in einem sehr modernen Gewand. Die Eingabe des Namens und der Gruppe erfolgt durch ein HTML-Formular. Die Daten gelangen mittels TCP/IP bzw. HTTP über beliebige Entfernungen zum Server und lösen dort in Abhängigkeit der eingegebenen Daten in der Datenbank folgende Aktionen aus:

Der PSEUDOCODE kann programmiersprachenunabhängig formuliert werden

```

wenn der Name schon vorhanden ist ...
(bzw. Anzahl der selektierten Datensätze > 0)
    wenn eine Gruppe angegeben wurde ...
        ÄNDERN der Gruppenzuteilung (UPDATE)
    sonst ...
        LÖSCHEN des Namens (DELETE)
sonst ...
    wenn eine Gruppe angegeben wurde
        ANFÜGEN des Namens in der Gruppe (INSERT)

```

Der Grund für diesen Pseudocode ist, zu zeigen, dass die Anwendung von elementarer Logik in jeder Programmierung wesentlich ist. Die Invarianz informatischer Inhalte und ihre Robustheit gegenüber zeitgeistigen Strömungen kommt hier sehr gut zum Ausdruck. Weiters gereicht es nicht gerade zum Nachteil, die Vermittlung von Grundlagen im Kontext mit sehr anwendungsorientierten Beispielen wie diesem zu verknüpfen.

Die Verknüpfung von Formularvariablen und der weiteren Bearbeitung im Skript ist in PHP sehr einfach

Eine ganz andere Verknüpfung, nämlich die der Formularfelder und dem Userinterface mit dem entsprechenden Skript (Code) ist bei PHP in nahezu genialer Weise verwirklicht: Die mit dem Parameter `name=„...“` versehenen Formularfelder (Objekte in Formularen, wie z. B. das Eingabefeld) können im PHP-Skript direkt angesprochen werden, und zwar als PHP-Variablen mit einem vorangestellten Dollarzeichen. So wird im obigen Skript aus

`name` $\hat{=}$ `$name` und aus `gruppe` $\hat{=}$ `$gruppe`.

Zu bemerken ist weiters die Abfrage in Zeile 13: `if (isset($name)) { ... }`

Nur wenn ein Namen eingetragen wurde, wird überhaupt eine Datenbankaktion ausgelöst, was ja durchaus Sinn macht. In Zeile 4 ruft sich das PHP-Dokument, nachdem das Formular abgeschickt wurde, mit der internen Variablen `$PHP_SELF`, die den Namen

Mit der Technik des `self`-Aufrufes, so genannten „Combskripts“, können in PHP sehr kompakte Lösungen realisiert werden

Micheuz: Internet inside – dynamisch und interaktiv

der PHP-Datei selber zurückliefert, selbst auf. In diesem Fall (Eingabe und Ausgabe werden gleichzeitig angezeigt) ist diese Technik in Bezug auf Softwareergonomie und Usability durchaus von Vorteil, weil übersichtlich. Man hätte sich auch der Frametechnik bedienen können: Mit dem Formular in einem Rahmen und dem tabellarischen Ausgabeteil in einem anderen. Aber diese Variante wurde schon an anderer Stelle angewendet und erklärt. Dies könnte als Übung dienen, die Frametechnik zu wiederholen.

Mit der Variablen \$PHP_SELF kann die Bezeichnung des aufrufenden PHP-Skripts ausgelesen werden

Anmerkungen:

Programmieren lernt man nur durch Programmieren. Die besten Einführungen sind im optimalen Fall gute Einstiegshilfen. Der Teufel steckt oft im Detail und die Probleme tauchen erst beim Realisieren auch schon vermeintlich einfacher Aufgaben auf. Oft sind es Kleinigkeiten, ein Fehlen eines Anführungszeichens, das Fehlen einer geschwungenen Klammer zum Beispiel, und man sucht den Fehler manchmal stundenlang. Wer beim Programmieren nie Fehler macht, hat in Wirklichkeit nichts gelernt oder ist nicht von dieser Welt (?).

Programmieren lernt man nur durch Programmieren

Neben den syntaktischen, programmiersprachenabhängigen und logischen, also programmiersprachenunabhängigen Problemen kommt bei der Server-Client-Programmierung als zusätzliche Schwierigkeit noch der vergleichsweise komplexe Ablauf in der Client-Server-Kommunikation hinzu. Aber speziell diese Verständnisprobleme sollten durch den vorliegenden Beitrag weitestgehend ausgeräumt sein.

Ein Ablaufdiagramm möge zum Abschluss dieses Kapitels die Client-Server-Kommunikation bei unserer Gruppeneinteilung veranschaulichen.

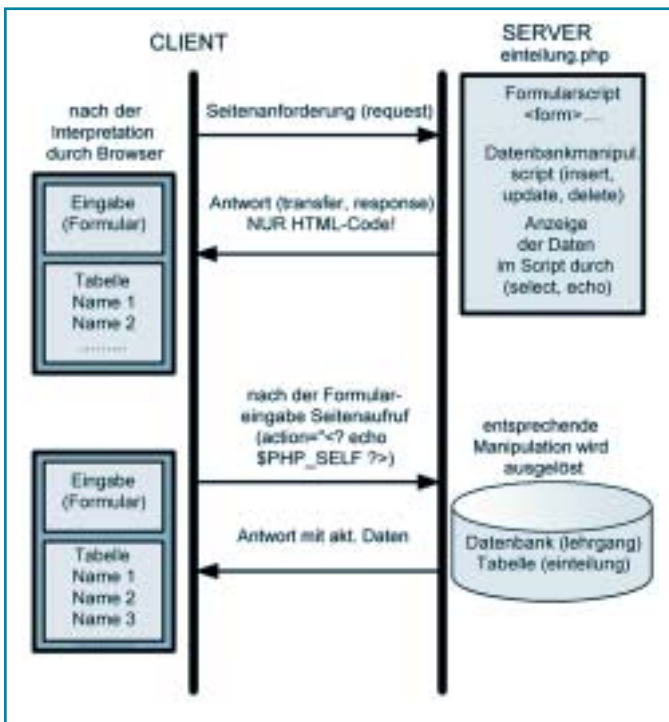


Abb. 13: Der Ablauf der Gruppeneinteilung kann in einem so genannten Sequenz- oder Ablaufdiagramm dargestellt werden.

Fachbereich allgemeinbildende höhere Schule

Jede Technologie entwickelt sich vom Primitiven über das Komplizierte zum Einfachen.

Antoine de Staint-Exupéry

7 Contentmanagement

CMS – Contentmanagementsystem, mehr als ein „buzzword“

Mit der Möglichkeit, (Teile von) Webseiten dynamisch aus Datenbanken zu erzeugen, haben wir ein sehr mächtiges Werkzeug kennen gelernt. Im Informatikunterricht können damit „didaktisch reduzierte“, überschaubare Anwendungen entworfen, modelliert und realisiert werden.

Ein universelles Modul: Artikelverwaltung

Unser heimlicher Auftraggeber wünscht sich eine ganz einfache Verwaltung von Texten (Artikeln) im Inter/Intranet. Ein simples digitales Tagebuch, indem er den Text mit Datum versehen kann. Genauso gut könnten es Sprüche, News oder Klassenbucheinträge oder sonstige Meldungen sein.

Wir benötigen zunächst ein Verwaltungstool, um die nötige Information in der Datenbank zu speichern. Ausgangspunkt ist die Tabelle „beitraege“ in der Datenbank „lehr-gang“. Die Struktur dieser Tabelle ist denkbar einfach, sie besteht nur aus einer ID (Primärschlüssel, der mit dem Attribut „autoincrement“ versehen ist, was bedeutet, dass MYSQL in diesem Feld eine fortlaufende Nummer automatisch vergibt), dem Feld „beitrag“ und dem Datumsfeld „datum“.

Der Datentyp TEXT kann bis zu 65.535 Zeichen aufnehmen. Falls das nicht reicht, kann der Type LONGTEXT eingestellt werden

Feld	Typ	Attribute	Null	Standard	Extra
<u>ID</u>	int(11)		Nein		auto_increment
beitrag	text		Nein		
datum	date		Nein	0000-00-00	

Das Format des Datums kann über PHP gesteuert werden. Aber auch MYSQL hat viele Funktionen implementiert

```

1 <?php
2 if (isset($addbeitrag)) {
    echo("<form action=' $PHP_SELF' method=' post' >");
    echo("<p>Beitrag eingeben: <br />");
    echo("<textarea name=' beitragtext'
        rows=' 10' cols=' 40' wrap></textarea><br />");
    echo("<input type=' submit' name=' button'
        value=' Beitrag speichern' /></p>");
    echo("</form>");
}
3 elseif (isset($updatebeitrag)) {
4     $datenbank = mysql_connect("localhost", "root", "");
    mysql_select_db("lehr-gang");

    $abfrage = mysql_query("SELECT ID, datum, beitrag
        FROM beitraege where
        ID=$updatebeitrag");

    $row = mysql_fetch_row($abfrage);
    $beitragtext=$row[2];
    echo("<form action=' $PHP_SELF? satznr=$updatebeitrag'

```

Micheuz: Internet inside – dynamisch und interaktiv

```

method=' post'>");
    echo("<p>Beitrag ändern: <br />");
    echo("<textarea name=' beitragtext' rows=' 10' cols=' 40'
        wrap>$beitragtext
    </textarea><br />");
    echo("<input type=' submit' name=' buttonupdate'
        value=' Beitrag ändern' /></p>");
5  echo("</form>");
    }
6  else {
    $datenbank = mysql_connect(„localhost“, „root“, „“);
    mysql_select_db(„lehrgang“);
7  if ($button == „Beitrag speichern“) {
        mysql_query(„INSERT INTO beitraege SET beitrag
            =' $beitragtext', datum=CURDATE()“);
    }
8  if (isset($deletebeitrag)) {
        mysql_query(„DELETE FROM beitraege
            WHERE ID=$deletebeitrag“);
    }
9  if (isset($buttonupdate)) {
        mysql_query(„UPDATE beitraege set beitrag=
            '$beitragtext', datum=CURDATE ()
            WHERE ID=$satznr“);
    }
    $abfrage = mysql_query(„SELECT ID,
        DATE_FORMAT(datum,' %d.%m.%Y' ), beitrag
        FROM beitraege“);
    echo("<table border=5>");
    echo("<tr><td colspan=6><p
        align=center><b>Artikelverwaltung</b></p></td></tr>");
10 while ( $row = mysql_fetch_row($abfrage) ) {
        $beitragid = $row[ 0 ];
        $beitragdatum = $row[ 1 ];
        $beitragtext = $row[ 2 ];
        echo("<tr><td>$beitragid</td><td>$beitragtext</td>
            <td width=80>$beitragdatum</td>
11 <td><a href= ' $PHP_SELF? deletebeitrag=$beitragid' >
                Löschen</a></td>
12 <td><a href=' $PHP_SELF? updatebeitrag=$beitragid' >
                Ändern </a></td>
13 <td><a href=' $PHP_SELF? addbeitrag=1' >
                Neu</a></td> „ );
    }
    echo(„</table>“);
}
?>
    
```

Parameterübergaben an das Skript erfolgen mit der Syntax: `datei?variable1=wert` mittels der POST-Methode

Beim Löschen und Ändern muss der Datensatz eindeutig adressiert werden

DATE_FORMAT ist eine Möglichkeit, das gewünschte Datenformat einzustellen

Der Aufruf dieses Skripts liefert:

Artikelverwaltung			
29	Das Problem der Zeitungsberichterstattung liegt darin, daß das Normale uninteressant ist. Below, Sam	03.09.2002	Löschen Ändern
19	Das Problem der Zeitungsberichterstattung liegt darin, daß das Normale uninteressant ist. Below, Sam	03.09.2002	Löschen Ändern
14	Schulden Du Deiner Bank 100 Dollar, so hast Du ein Problem. Schuldest Du ihr jedoch 500 Millionen Dollar hat die Bank das Problem. (J.P. Getty)	03.09.2002	Löschen Ändern
15	Management ist die schließliche aller Künste - die Kunst, Talente richtig einzusetzen. McHarney, Robert Strang	03.09.2002	Löschen Ändern
16	Wenn Du ein Schiff bauen willst, so trommle nicht Männer zusammen, um Holz zu beschaffen, Werkzeuge vorzubereiten, Aufgaben zu vergeben und die Arbeit anzustellen, sondern lehre die Männer die Seemannschaft nach dem weisen ruhigen Mann: Saint-Exupéry, Antoine de (eigentlich: Antoine-Marie-Jacques Graf von)	03.09.2002	Löschen Ändern
17	Das Problem der Zeitungsberichterstattung liegt darin, daß das Normale uninteressant ist. Below, Sam	03.09.2002	Löschen Ändern

Ein Webinterface, das für das Bearbeiten von Tabellen (fast) keine Wünsche offen lässt ...

Fachbereich allgemeinbildende höhere Schule

Ist Programmieren ist eine Kunst? Oder nur Übungssache?

Dieses PHP-Script ist ein kleines Kunstwerk, weil der Aufruf dieser Datei „admin.php“ eine vollständige Verwaltung der Tabelle mit Anfügen, Löschen und Ändern ermöglicht.

Was will man mehr?

Darf denn Softwareentwicklung zur Kunst „entarten“? Soll man an die Programmierung nicht ingenieurwissenschaftlich herangehen – mit den Methoden des Software-Engineerings? Top-Down-Entwürfe, Modularisieren, einen objektorientierten Ansatz versuchen?

Unsere Programmierung im Kleinen genügt den Ansprüchen strukturierter Programmierung, das Programm macht, was es soll. Wir können dieses Skript parametrisieren und als Funktion in unsere Bibliothek aufnehmen und immer dann verwenden, wenn es darum geht, kleine Tabellen zu verwalten. Bei der Realisierung ähnlicher Aufgabstellungen gibt es zwei Hürden zu überwinden:

Eine Hürde ist das vollständige Verstehen des Ablaufs in der Client-Server-Kommunikation, die im obigen Fall durch den zweimaligen Selbstaufruf der Webseite schon ziemlich komplex ist.

Die andere Hürde ist die Beherrschung der Syntax – auch der PHP-Interpreter verzeiht nichts –, und das genaue Unterscheiden zwischen Formularvariablen, PHP-Variablen und den Literalen (= eigentliche Werte, den die Variablen annehmen). Vor allem bei Textvariablen und Textliteralen, die unter Anführungszeichen stehen, können die Ausdrücke sehr verwirrend werden.

Pseudocode:

Sorgfältige Analysen von Programmen sind Voraussetzung für ihre Erweiterbarkeit und Adaption

Die Syntax sollte nicht im Vordergrund stehen

Bei komplexeren Anwendungen ist eine andere Notation anzustreben, z. B. UML

*UML:
Unified Modeling Language*

```

WENN Variable "$addbeitrag" gesetzt
  Anfügeformular anzeigen
  Variable button erhält den Wert "Beitrag speichern",
  auf den in Zeile 7 abgefragt wird
  Formular abschicken und Webseite nochmals aufrufen
  (action=' $PHP-SELF' ...) (SELBSTAUFRUF)
SONST WENN Variable „$updatebeitrag“
  Änderungsformular anzeigen
  Formular abschicken und Webseite nochmals aufrufen, (diesmal
  wird der Parameter satznr mit übergeben
  (action=' $PHP-SELF?satznr=$updatebeitrag' ...),
  weil ja beim Ändern der betroffene Datensatz gezielt
  angesprochen werden muß.
  (SELBSTAUFRUF mit Parameter satznr)
SONST
  wenn Buttonname (vom Anfügeformular) den Wert
  "Beitrag speichern" hat
    neuen Datensatz anfügen (INSERT)
  wenn $deletebeitrag gesetzt ist (siehe Zeile 11)
    Datensatz löschen (DELETE)
  wenn $buttonupdate gesetzt ist
    (siehe Änderungsformular)
    Datensatz mit den geänderten Formulardaten aus
    (ab Zeile 4) ändern (UPDATE)
    (zu ändernder Datensatz wird mit der
    übernommenen Variablen $satznr gefiltert)
Alle Daten der Tabelle inklusive Links ausgeben
LÖSCHEN (mit Parameterübergabe)
ÄNDERN (mit Parameterübergabe)
NEU (ohne Parameterübergabe)

```

Micheuz: Internet inside – dynamisch und interaktiv

Der nächste Schritt ist nun, die so verwalteten Daten, die in der Tabelle „beitraege“ der Datenbank „lehrgang“ gespeichert und verwaltet werden, in mehreren Spielformen auszugeben.

Je nach Content kann nun aus der Tabelle ein Zufallsartikel angezeigt werden, die letzten 3 Einträge oder alle Einträge. Es könnte z. B. auch nach Stichworten gesucht werden.

Mit der Realisierung dieser Funktionalitäten und „Begehrlichkeiten“ hat der Programmierer seine Schuldigkeit getan, die Stunde des Designers bzw. Grafikers ist gekommen. Jetzt sind unter anderem gute HTML-Kenntnisse gefragt, um den Content auch optisch gut darzustellen.

Wir wollen nun aus dieser Tabelle einen Zufallsartikel bzw. Zufallsspruch auswählen und anzeigen.

Ausgabe eines Zufallsartikels:

Im Prinzip müssen wir nur den Ausgabeteil im obigen Skriptum adaptieren.

```
<?
mysql_connect("localhost", "root", "");
mysql_select_db („lehrgang“);
$abfrage = mysql_query („SELECT
DATE_FORMAT(datum, '%d.%m.%Y' ), beitrags
FROM beitraege order by rand()“);
$datensatz=mysql_fetch_row($abfrage);
echo ($datensatz[ 0] . "<br>");
echo ("<i>". $datensatz[ 1] . "</i>");
?>
```

Das Design einer Webanwendung ist sehr wichtig

Die SQL-Anweisung „order by rand()“ durchmischt die Datensätze

Mit der Sortieroption „oder by rand()“ werden die Datensätze zufällig durchmischt. Es reicht dann, den ersten Datensatz auszugeben.

Das war's.

Ein abschließendes, komplexeres Beispiel

An dieser Stelle bietet sich die Gelegenheit, sich dem sehr umfassenden, aber noch nicht normierten Begriff „Contentmanagement“ nicht nur theoretisch, sondern auch praktisch zu nähern.

Die einen verstehen unter einem Contentmanagementsystem (CMS) einen allumfassenden Oberbegriff, der Produktion, Transport (Workflow), Speicherung bis zur automatischen Publizierung von Webcontent für große Websites einschließt, in deren Hintergrund viele Abläufe automatisiert sind und die Trennung von Design und Inhalt ein wesentliches Merkmal ist. Die anderen begnügen sich damit, sich unter einem CMS, ein einfaches Verwaltungssystem für Informationen bzw. Artikel jeder Kategorie vorzustellen. Eine vielleicht am besten zutreffende Aussage war in einem angesehenen Magazin (CYBiz) zu lesen: „CMS ist eine komplexe Software, die der Erstellung, Überwachung, Freigabe, Veröffentlichung und Archivierung von interaktiven Inhalten für Websites dient.“

Die minimale Anforderung an ein WCMS, also (Web)Content Management System, ist die Trennung des Inhaltes von Gestaltung (Design) und Funktionalität (Prozesse, Algorithmen).

Ein Contentmanagementsystem basiert auf der Trennung von Inhalt und Design

Die Begriffsdefinition von Contentmanagementsystemen reicht von der einfachen Verwaltung von Texten bis zu hochkomplexen Portalsystemen mit einer nahezu unüberschaubaren Anzahl von Features

Fachbereich allgemeinbildende höhere Schule

Zum Abschluss soll hier ein methodisch gangbarer Weg skizziert werden, ein einfaches, aber erweiterbares CMS im Schulkontext zu entwerfen und auch einzusetzen, in dem die in diesem Beitrag vorgestellten Techniken angewendet werden. Es geht darum, ein erweitertes Verwaltungssystem für Berichte (Artikel) jeder Art, die kategorisiert sind, zu entwickeln.

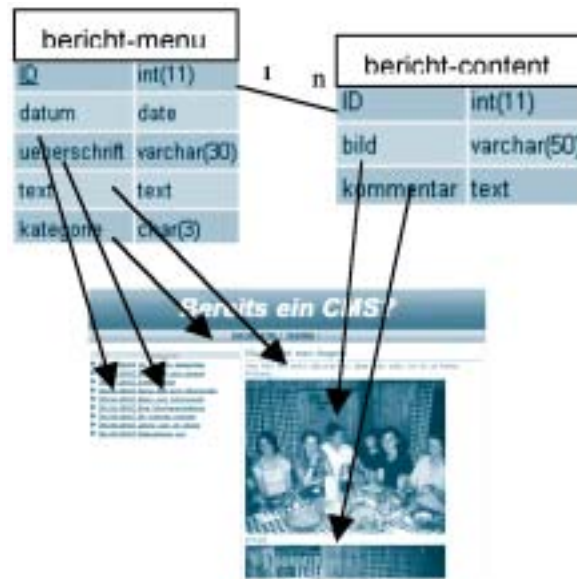
Abb. 14: Das „Skelett“ eines einfachen bebilderten Artikelverwaltungssystems

TOP: Grafik	
Hauptmenü (Kategorien)	
Kategorie	Überschrift
Bericht 1	Text
Bericht 2	BILD 1
.....	Kommentar 1
	BILD2
	Kommentar 2

BOTTOM	

Dieses (als fast klassisch zu bezeichnendes aber den Usability-Empfehlungen gerecht werdende) Layout soll als Vorlage (Template) dienen und bei jedem Seitenwechsel angezeigt werden. Wir werden in diesem Fall keine Frametechnik anwenden, sondern wie die meisten Portalseiten alle Seiten „neu“ aufbauen.

Abb. 15: Die Datenbank setzt sich aus zwei verknüpften Tabellen zusammen



Diese Website hat strukturell eine große Ähnlichkeit mit vielen bebilderten Newssystemen. Alle Texte – Hauptkategorien, Überschriften (Menüpunkte), Text und Bildkommentare sowie die Bildnamen – sind in den 2 Tabellen „bericht-menu“ und „bericht-content“, die über den Schlüssel ID verknüpft sind, gespeichert. Diese Daten werden durch den User mit der Template-Datei, z. B. „berichte.php“ auf dem Server zusammengeführt und im HTML-Format zum Client übertragen.

Die Struktur eines einfachen CMS, das Berichte, angereichert mit Bildern und geordnet nach Kategorien, verwaltet

Durch Template-Dateien bekommt die Website ein einheitliches Design mit wechselnden Inhalten

Wir haben es hier mit einer sauberen Trennung von Content (Texte, Bilder) und der Struktur bzw. dem Layout zu tun. Der PHP-Interpreter sorgt dafür, dass die Inhalte – versehen mit entsprechenden Attributen (Schriftgröße, Art ...) - aus der Datenbank in die Vorlage eingefügt werden.

Micheuz: Internet inside – dynamisch und interaktiv

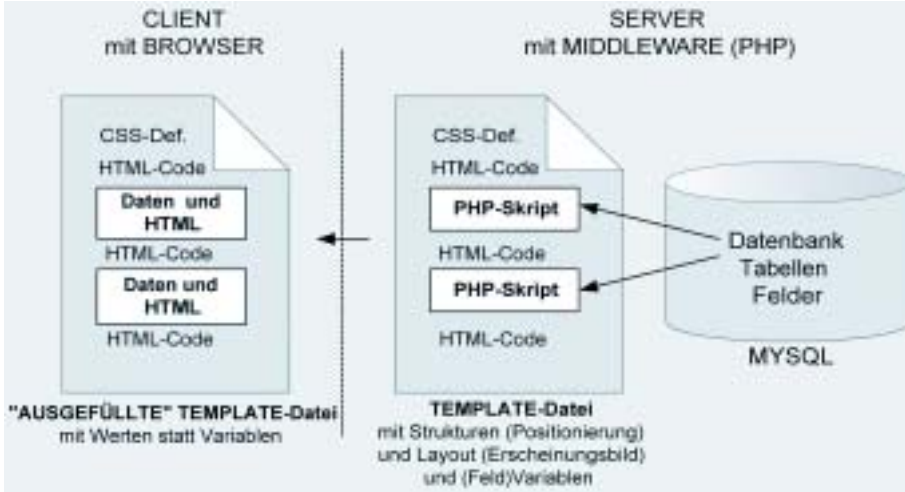


Abb. 16: Die Template-Dateien stellen die Struktur und das Erscheinungsbild mit Platzhaltern zur Verfügung. Je nach Komplexität der Website kann es mehr oder weniger Templates geben

z. B. Grafik	
<?PHP-Skript 1?>	<?PHP-Skript 2?>
BOTTOM	

Abb. 17: Das einfache Layout mit zwei Zellen, in denen die Skripts für den dynamisch aus den Datenbanken generierten Inhalt (Content) sorgen

Skript 1:

erzeugt aus der Tabelle „bericht_menu“ die Menüpunkte mit den Links

```

<?
$db=mysql_connect("localhost","root","");
mysql_select_db("lehrgang");
if (isset($kat) ) {
    $abfrage = mysql_query("select
        DATE_FORMAT(datum,' %d.%m.%Y' ),ueberschrift,text,id
        from bericht_menu
        where kategorie=' $kat'
        order by datum desc");
} else {
    $abfrage = mysql_query("select
        DATE_FORMAT(datum,' %d.%m.%Y' ),
        ueberschrift,text,id
        from bericht_menu order by datum desc");
}

while ($datensatz=mysql_fetch_row($abfrage)) {
    echo("<tr>");

    echo("<td width=' 13' ><font size=' 1' >
        <img border=' 0' src=' f2.gif' width=' 10' >
        </font> </td>");
    echo("<td <font size=' 1' >
        <a href=' \$.PHP_SELF.'?bnr=' \$.datensatz[3].' >
        [ \$.datensatz[0].']' \$.datensatz[1].' </a>
        </font></td>' );
    echo("</tr>");
}
?>
    
```

Hypertext-Referenzen (Links) werden in HTML durch Verweistext angegeben. Man beachte im nebenstehenden Listing die kunstvolle Konstruktion der (maskierten) Anführungszeichen!

Fachbereich allgemeinbildende höhere Schule

Script 2:

erzeugt aus der Tabelle „bericht_content“ die Anzeige der Bilder mit Text

Der HTML-Tag
 zeigt das
 Bild an.
 IMG SRC ... „image
 source“ bedeutet so viel
 wie „Bildquelle“

```
<?
    if (isset($bnr) ) {
        $db=mysql_connect("localhost","root","");
        mysql_select_db("lehrgang");
        $abfrage = mysql_query("select ueberschrift,text
                                from bericht_menu where id=' $bnr' ");
        $datensatz=mysql_fetch_row($abfrage);
        echo("<tr><td class=' ueber' >$datensatz[ 0] </td></tr>");
        echo("<tr><td>$datensatz[ 1] </td></tr>");
        $abfrage = mysql_query("select bild, kommentar
                                from bericht_content where id=' $bnr' ");
        while ($datensatz=mysql_fetch_row($abfrage)) {
            echo("<tr><td><img src='\".$datensatz[ 0] .' \"
                    border=0></td></tr>' );
            echo("<tr><td STYLE=' color: gray; font-size:8pt;
                    font-family: Verdana; font-style:italic' >
                    $datensatz[ 0] </td></tr>");
        }
    }
?>
```

Bemerkungen und Hinweise:

Die Anzeige der Menüpunkte auf der linken Seite erfolgt in Verweisform () mit dem Parameter „bnr“ (Berichtsnummer)!

Dabei wird auf die Template-Datei rückverwiesen, diese allerdings mit dem Parameter „bnr“ aufgerufen. Nur wenn der Aufruf der Templatdatei von diesem Verweis kommt, wird das Script 2 aktiv! Dieses gibt dann den Bericht auf dem Bildschirm in einer mehr oder weniger gediegenen Form (es werden inline-CSS-Angaben verwendet) aus. Hier kommt es (unabhängig von den Daten) darauf an, dass der Webdesigner Geschmack beweist und die Ausgabe farblich abstimmt und mit den richtigen Proportionen bemisst. Wünschenswert wäre hier noch ein Administrationstool für das Hochladen der Bilder. Tipp: Im Internet gibt es für dieses Upload genügend PHP-Skripts, die nur anzupassen sind.

Im einfachen Fall kann der Content mit dem PHPMYADMIN-Tool administriert werden und das File-Upload mit einem FTP-Programm erfolgen.

Mit dieser fundamentalen Technik von Template-Dateien können nun andere Features in die Homepage (= CMS?) eingebaut werden. Um den Kreis zu schließen, wollen wir zum ersten Kapitel zurückkehren und die Umfrage als Teil einer – vom werten Leser zu vervollständigenden – Schulhomepage erklären.

Mit einem webbasierten
 Administrationstool
 müssen auch Bilder und
 Grafiken „upgeloadet“
 werden können

Micheuz: Internet inside – dynamisch und interaktiv

TOP: Grafik		
Hauptmenü		
Menü (dynam. Menü)	CONTENT1	CONTENT2
Die Schule	Startseite	Forum
Aktuelles	Bild der Schule,	Zufallspruch
Berichte	Kurzvorstellung	Umfrage
	
Gästebuch		
.....		
BOTTOM: Menü aus Navigation		

Abb. 18: Umfragen sind ein beliebter interaktiver Bestandteil von vielen Portalsystemen

Die Realisierung der Umfrage aus dem ersten Kapitel (Bild 1):

```

if ($REQUEST_METHOD=="POST") {
    $update="update umfrage set anzahl=anzahl+1
        where beschreibung=' $poll' ";
    mysql_query($update); }
$datensatz=mysql_query("select beschreibung,anzahl
        from umfrage");
$anzahl=mysql_num_rows($datensatz);
echo('<form action="'.PHP_SELF.'" method="post">
    <table cellpadding="3" cellspacing="1"
        bgcolor="CCCCFF" width="100%">');
echo('<tr bgcolor="#AAAAFF"><td colspan="3"
    align="center"><font size="3">
    <b>AKTUELLE UMFRAGE</td></tr>');
echo('<tr bgcolor="#BBBBFF"><td align="center" colspan="3">
    <font style="font-size:8pt;">
    Welche Programmiersprache bevorzugen Sie?</td></tr>');
while ($row=mysql_fetch_row($rs)) {
    echo('<tr><td><input type="radio" name="poll"
        value="'. $row[0] .' ">
    </td><td><font style="font-size:8pt;">' . $row[0] .'
    </td><td><font style="font-size:8pt;">' . $row[1] .'
    </td></tr>'); }
echo('<tr><td align="center" colspan="3">
    <input type="submit" value="Abstimmen"></td></tr>');
echo('</table>
</form>');
}
?>

```

Die Abfrage `$REQUEST_METHOD="POST"` ist eine weitere Möglichkeit, festzustellen, ob es einen Selbstaufruf der Seite gegeben hat

Bei Aktivierung des „Radiobutton“ wird die Seite nochmals aufgerufen, vorher aber der entsprechende Zähler in der Tabelle erhöht (Zeile 2)

Die Abfrage „if (\$REQUEST_METHOD=="POST")“ ist eine weitere Möglichkeit, abzufragen, ob der Aufruf des PHP-Files ein Selbstaufruf war. Als Methode, die Formulardaten an dem Server und somit der Datenbank (UPDATE der Votings) zu übergeben, wurde POST angegeben (`<form method="POST">`).

Fachbereich allgemeinbildende höhere Schule

Es gibt bereits eine unübersehbare Zahl von mehr oder weniger brauchbaren Contentmanagement-systemen

Mit den hier vorgestellten Techniken sollte es möglich sein, ein individuelles CMS zu modellieren und auch zu realisieren

Nach der Abgabe der Stimmen wird die Homepage (mit den aktualisierten Stimmen) angezeigt. Dem geneigten Leser ist überlassen, an dieser Stelle das vom Counter bereits bekannte IP-Locking (jetzt aber durch Speicherung in einer Tabelle der Datenbank) einzubauen.

Je mehr Content in die Datenbank verlagert wird, desto weniger Gestaltungsmittel sind für einen nicht programmierenden „Redakteur“ gegeben, aber umso stringenter ist das Erscheinungsbild der Website und desto besser ist der Inhalt von der Form getrennt. Zufallssprüche, Newsticker (Aktuelles), Berichte und Personallisten (SchülerInnen, LehrerInnen) werden sinnvollerweise aus Datenbanken und Tabellen oder aus Textdateien erzeugt, was aber aus oben angeführten Gründen mehr Programmierarbeit erfordert. Foren, Gästebücher, Umfragen sind ja interaktiv und deshalb per se dynamisch anzulegen. Eine weitere Überlegung ist die Navigation (Menü) wert, da diese ja auch aus einer Datenbanktabelle erzeugt werden könnte. Ebenso verhält es sich mit dem Kopf- und Fußbereich, den man aus Tabellen generieren kann. In der Praxis zeigt es sich, dass es nicht die optimale Lösung gibt. Automatismen haben den Nachteil, dass sie mehr oder weniger unflexibel sind und der Aufwand für eine nachträgliche Änderung manchmal nicht unerheblich ist.

Es gibt bereits eine unübersehbare Zahl von mehr oder weniger brauchbaren Contentmanagementsystemen. Mit den in diesem Beitrag vorgestellten Techniken und etwas Ausdauer lassen sich aber bereits an den eigenen Bedürfnissen orientierte Systeme im Eigenbau realisieren.

► Skriptsprachen - inklusive deren Anbindung an Datenbanken – bieten eine faszinierenden Möglichkeiten, die Funktionalität, Attraktivität und den Nutzwert von Internetseiten mit vergleichsweise wenig Aufwand wesentlich zu erhöhen.