

L.A.M.P. - Eine Alternative zu MS Active Server Pages

Henning P. Schmiedehausen
<hps@intermeta.de>

Vorbemerkungen

- Dieser Vortrag beschäftigt sich mit L.A.M.P.
MS ASP wurde nur erwähnt, damit Sie in den Vortrag kommen.
- Dieser Vortrag beschäftigt sich mit PHP.
Nicht mit PHP3. Aktuell ist PHP Version 4.
Und PHP3, Version 4 klingt blöd.
- Dieser Vortrag ist weitestgehend ideologiefrei.
Weder wird freie Software in den Himmel gelobt
noch auf kommerzielle Software eingeschlagen.
- Auch nicht auf Microsoft.

Was ist L.A.M.P.?

L.A.M.P. besteht aus vier Komponenten:

- L: Linux als Betriebssystem
- A: Apache als Webserver
- M: MySQL als Datenbank
- P: PHP als Scriptsprache

„Erzeugen, Speichern und Ausliefern von Daten“

Was kostet L.A.M.P.?

- Alle Komponenten von L.A.M.P. sind „freie“ Software
- Linux und MySQL: GNU GPL (General Public License)
- Apache: Apache License
- PHP: PHP License (entspricht in etwa der Apache License)
- Für die Installation und Benutzung sind **KEINE** Lizenzgebühren (Geld) zu bezahlen

Warum L.A.M.P.? (I)

- Moderne Websites haben nur noch wenige statische Seiten
- Selbst einfache Webseiten besitzen aktive Anteile:
 - Kopfzeilen
 - Fußzeilen
 - Änderungsdatum
 - Versionen
- Apache bietet für einfache Anforderungen „Server Side Includes“ (SSI)

Warum L.A.M.P.? (II)

- Server Side Includes können keine komplexeren Entscheidungen treffen.
- Aktive Inhalte auf Webseiten sollen über einfaches Einfügen von Text hinausgehen:
 - personalisierte Websites
 - Warenkörbe
 - Feedback- oder Diskussionsforen
 - usw.
- Große Sites sind Datenbanken mit Web-Frontend

Alternativen zu L.A.M.P. (I)

- Linux: *BSD, Solaris (oder jedes andere Unix), Windows NT
- Apache: Roxen WebServer, Zeus, Netscape (iPlanet) Server, IIS
- MySQL: PostgreSQL, Adabas, Informix, Oracle
- PHP: perl, Python, Java, Visual Basic, jede beliebige andere Programmiersprache

Alternativen zu L.A.M.P. (II)

- MS Active Server Pages (nur Windows)
- ColdFusion (proprietär)
- ePerl (seit 1998 nicht weiterentwickelt)

- Java Server Pages (JSP)

Linux

- Linux 0.01 wurde Herbst 1991 freigegeben
- Steht stellvertretend für viele freie Betriebssysteme
- LAMP klingt besser als FAMP oder NAMP
- Linux kann heutzutage von jedermann „aus der Schachtel gezogen“ und installiert werden
- professioneller Support verfügbar
- „Linux hat den cooleren Namen“
- Linus Torvalds

Apache

- Entstand 1995 aus einer Patchsammlung für den NCSA Webserver („a patchy server“)
- Laut unabhängigen Zählungen der meistbenutzte Webserver im Internet
- Für fast jede Plattform verfügbar (auch exotische wie Windows NT)
- einfach durch Module erweiterbar
- „industrial strength“ Webserver
- „correctness more important than speed“ - Dean Gaudet

MySQL

- Seit ca. 1992 verfügbar
- Relationales Datenbankmanagementsystem
- Unterstützt die Verwaltung mehrerer verschiedener Datenbasen gleichzeitig
- mehrstufiges Zugriffs- und Rechtemodell
- Unterstützt nur SQL Subset
- für Abfragen optimiert, somit gut für Webapplikationen geeignet

PHP

- Entstand Mitte 1995 als „personal homepage tool“ (Rasmus Lerdorf)
- wurde gezielt für Webapplikationen entwickelt
- Als Modul für Apache (und viele andere Webserver) implementiert
- eigene Syntax mit Anlehnungen an C++ und perl
- Kernstück des LAMP Systems

Was ist L.A.M.P. nicht?

- „Auspacken, Einschalten, Los“
- L.A.M.P. ist ein Framework für Applikationen und Daten, keine fertige Sammlung von Modulen
- Content Management System (CMS)
- keine „silver bullet“

Microsoft Active Server Pages

- Visual Basic in Webseiten eingebettet
- Ursprünglich nur für Microsoft IIS auf WIN32 verfügbar
- Enge Verknüpfung mit OLE und IIS
- Session-Handling und Persistenz implementiert
- Inzwischen einige Implementationen von ASP für Unix verfügbar (Chili!Soft ASP, InstantASP)
- Verschiedene IDE verfügbar

Gute Eigenschaften von PHP

- Vollständige Programmiersprache
- Integration direkt in HTML Seiten
- Komplexe Elemente (Strings, Objekte,...) vorhanden
- Modifikation der HTTP Header möglich
- Session-Handling und Persistenz sind seit PHP 4 Bestandteil der Sprache

...und schlechte

- PHP hat einen ähnlichen Wildwuchs im Sprachumfang wie perl (zumindest ist aber noch kein die unless not open möglich)
- Über 1400 (!) einzelne Funktionen
- Oft kein Konzept erkennbar (beispielsweise beim Datenbankzugriff)
- Weder funktionale noch objektorientierte Programmierung wird erzwungen
- Keine Trennung von „Form“ und „Funktion“
- Keine IDE, nur rudimentärer Debugger

Ein einfaches Beispiel

Datum und Uhrzeit in eine Webseite einfügen:

```
<HTML>
<HEAD>
<TITLE>Datum und Uhrzeit einfügen</TITLE>
</HEAD>
<BODY BGCOLOR=„white“>
<H1>Datum und Uhrzeit einfügen</H1>
Aktuelle Uhrzeit und Datum:

<?php
echo date();
?>

</BODY>
</HTML>
```

PHP Tags

- `<SCRIPT LANGUAGE=„php“> . . . </SCRIPT>`
- `<? . . . ?>`
- `<?php . . . ?>`
- **ASP kompatibel:**
`<% . . . %>` und
`<%= $variable %>` (Ausgabe eines Variablenwertes)

PHP in einer Webseite

- Es dürfen beliebig viele PHP Blöcke vorkommen
- Die Seite wird von oben nach unten interpretiert, nicht kompiliert
- Die Ausgabe eines PHP Blocks wird direkt in die HTML Seite eingefügt
- PHP Code kann vom HTML durch `require` und `include` getrennt werden

Mehrere PHP-Blöcke

```
[... Header ...]  
<?php  
    srand((double)microtime()*1000000);  
    $test = rand(1,99);  
?>  
Die Zufallszahl <?php echo $test; ?> ist  
<?php if(($test % 2) == 0) { ?>  
gerade  
<?php } else { ?>  
ungerade  
<?php } ?>  
[... Footer ...]
```

PHP und MySQL

- Datenbankverbindungen können persistent sein , d.h. mehrere PHP Programme können sich eine Datenbankverbindung teilen
- PHP verfügt für Zugriffe auf MySQL Datenbanken über etwa 35 Funktionen
- SQL Abfragen können direkt in PHP Datentypen gespeichert werden
(`mysql_fetch_array`, `mysql_fetch_object`)
- Das Datenbankinterface ist MySQL spezifisch

Datenbankabfrage in PHP

```
[...]  
<TABLE>  
<?php  
$link = mysql_connect(„dbserver“, „dbuser“, „dbpasswd“);  
$res = mysql_query(„mydatabase“ „SELECT name FROM userdb“, $link);  
while($row = mysql_fetch_array($res)) {  
?>  
<TR><TD>Benutzername:</TD><TD> <?php echo $row[0]; ?></TD></TR>  
<?php  
}  
mysql_free_result($res);  
mysql_close($link);  
?>  
</TABLE>  
[...]
```

Applikationen in PHP

- IMP Webmail System
- Keystone (Workgroup und Troubleshooting System)
- phpCMS (Content Management System)
- phpMyAdmin (MySQL Administration)
- phpChat (Chat Server)
- phpPolls (Voting Booth)

Referenzen (Quelle: www.php.net)

- The Audi TT Website
- White House 200th anniversary site
- RedHat Software
- Neckermann
- Big Brother - The TV Show
- SAT.1
- ComDirekt Bank
- NASA
- Spiegel Online

Fazit

- L.A.M.P. ist ein System zum Betrieb von Websites mit aktiven Inhalten
- Apache und Linux als Industriestandards bieten ein solides Fundament
- PHP als mächtige Scriptsprache ermöglicht die Erstellung von komplexen Anwendungen
- MySQL bietet eine performante und günstige Lösung zum Speichern von Daten für Webapplikationen

Fragen und Diskussion

Vielen Dank für Ihre Aufmerksamkeit

<ftp://ftp.hometree.net/pub/kongress2000/LAMP.ppt>