

# So setzen Sie eine Entwicklungsumgebung für Magento auf

In dieser Anleitung erfahren Sie, wie Sie Ihre ganz persönliche Serverumgebung aufsetzen können. Dabei wird gezeigt, wie Sie Apache, MySQL und PHP unter Windows und Linux installieren können. Eine solche Serverumgebung ist nötig, um Magento überhaupt installieren zu können. Sollten Sie XAMPP oder ein vergleichbares Tool einsetzen, werden diese Elemente automatisch installiert. Eine ausführliche Anleitung wie Sie XAMPP installieren, finden Sie in Kapitel 2 des Buches.

## ■ Die Windows-Installation

Zunächst widmen wir uns der Installation des Apache-Servers. Laden Sie sich dazu die aktuelle Apache-Version unter <http://httpd.apache.org/download.cgi> herunter. Wenn Sie mit Windows 9.x oder ME arbeiten, benötigt zusätzlich den MSI Installer, der kostenlos unter <http://support.microsoft.com/default.aspx?scid=kb;de;292539> zum Download bereit steht. Andere Windows-Versionen wie XP, Vista oder Windows 7 sind davon nicht betroffen. Für Windows verwenden Sie in aller Regel die Version *Win32 Binary without crypto (no mod\_ssl) (MSI Installer)*.

Gestartet wird die Installation durch Anklicken der heruntergeladenen MSI-Datei. Ein Assistent führt durch die Installation des Apache-Servers. Der Willkommensbildschirm kann mit NEXT übersprungen werden. Akzeptieren Sie anschließend die Lizenzbestimmungen und klicken Sie auf NEXT. Das Dialogfenster *Read this first* kann ebenfalls mit NEXT bestätigt werden. Sobald das Fenster mit den Serverdaten erscheint, geben Sie für eine lokale Installation Folgendes ein: Bei *Network Domain* tragen Sie *127.0.0.1* ein, als Server Name wird *localhost* verwendet. Die E Mail-Adresse spielt zwar bei einer lokalen Installation keine Rolle, ohne sie lässt sich der Apache allerdings nicht installieren.

Mit NEXT geht es zum nächsten Schritt. Bestätigen Sie *Typical* ebenfalls mit NEXT. Ändern Sie bei Bedarf das Installationsverzeichnis über CHANGE und bestätigen Sie auch dieses Fenster mit NEXT. Die Installation kann nun über INSTALL abgeschlossen werden.

Unter Windows 7, Vista, XP und NT läuft der Server nach der Installation als Service. Wenn Sie Apache auf einer Windows-9.x- oder einer ME-Plattform betreiben wollen, müssen Sie ihn per Hand über die *Apache.exe* starten, die im *bin*-Verzeichnis von Apache liegt.

Legen Sie sich anschließend die Verzeichnisse an, in denen später die Magento-Dateien gespeichert werden sollen.

```
C:\apache\htdocs\magento
```

Um zu überprüfen, ob der Server tatsächlich läuft, rufen Sie im Browser *http://localhost* auf. Daraufhin wird der Startbildschirm des Servers angezeigt.

Der Server muss nun noch konfiguriert werden. Die Konfigurationsdatei finden Sie standardmäßig im Verzeichnis *C:\Program Files\Apache Software Foundation\Apache2.2\conf\* unter dem Namen *httpd.conf*. Über diese Datei wird der Apache gesteuert.

Suchen Sie nach folgender Zeile:

```
DocumentRoot "C:/Program Files/Apache Software Foundation/Apache2.2/htdocs"
```

Hier wird der Pfad zum Magento-Verzeichnis angegeben. Sie müssen diesen Wert nur dann verändern, wenn Sie ein anderes Installationsverzeichnis angegeben haben – beispielsweise:

```
DocumentRoot "C:/apache/htdocs/magento"
```

Suchen Sie anschließend nach den Eintrag *ServerRoot*. Auch dort muss gegebenenfalls der Pfad angepasst werden.

```
ServerRoot "C:/apache/htdocs/magento"
```

Nun geht es weiter mit den Dokumentverzeichnissen.

```
<Directory "C:/Program Files/Apache Software Foundation/Apache2.2/htdocs">
...
</Directory>
```

Ersetzen Sie die erste Zeile durch folgenden Eintrag.

```
<Directory "C:/apache/htdocs/html">
```

Auch das gilt natürlich nur, wenn das Installationsverzeichnis tatsächlich *apache* heißt.

Damit steht die Grundkonfiguration für den Apache. In der Konfigurationsdatei sind noch zahlreiche weitere Optionen einstellbar. Informationen hierzu finden Sie unter *http://httpd.apache.org/docs/2.0/*.



### Weiterführende Informationen zur *httpd.conf*

An dieser Stelle konnten nur die wichtigsten Optionen der Apache-Konfigurationsdatei vorgestellt werden. Eine sehr umfangreiche Einführung in die Thematik finden Sie beispielsweise unter *http://web.mit.edu/rhel-doc/4/RH-DOCS/rhel-rg-de-4/s1-apache-config.html*.

Der Server wird nach der Installation automatisch gestartet. Sie können ihn aber jederzeit anhalten und neu starten. Rufen Sie dazu den Apache-Monitor über die Taskleiste auf. Hierüber können Sie den Server starten, anhalten und neu starten.

## Die manuelle PHP-Installation

Es ist noch gar nicht so lange her, dass PHP für Windows eher belächelt wurde. Inzwischen hat sich das grundlegend geändert, die PHP-Windows-Kombination ist durchaus gebräuchlich.

So gibt es mittlerweile einen Installer, der durch die früher doch recht holprig ablaufende Installation führt. Heruntergeladen werden kann dieser unter <http://windows.php.net/download/#php-5.4>. Am besten laden Sie die Zip-Version der Windows-Binaries herunter. (Verwenden Sie die Zip-Version, die im Bereich *Thread Safe* angeboten wird, nicht die aus dem Bereich *Non Thread Safe*.) Entpacken Sie dafür zunächst das heruntergeladene Zip-Archiv in ein beliebiges Verzeichnis wie beispielsweise `C:\PHP`. Kopieren Sie dann die Datei `php.ini-development` aus dem PHP-Verzeichnis in das Windows-Verzeichnis. Üblicherweise liegt dieses direkt unter `C:\`, also `C:\Windows`. Die `php.ini-development` muss in `php.ini` umbenannt werden. Diese `php.ini` ist die Konfigurationsdatei von PHP, über die festgelegt wird, wie PHP arbeiten soll. So kann man über die `php.ini` beispielsweise bestimmen, wie lange ein Skript maximal ausgeführt und ob der Safe-Mode aktiviert werden soll. Die Datei ist in mehrere Abschnitte unterteilt, über die sich jeweils ganz bestimmte Funktionen konfigurieren lassen.

Suchen Sie nun in der Apache-Konfigurationsdatei `http.conf` nach folgendem Eintrag.

```
#
# Dynamic Shared Object (DSO) Support
#
# To be able to use the functionality of a module which was built
# as a DSO you have to place corresponding 'LoadModule' lines
# at this location so the directives contained in it are actually
# available _before_ they are used. Statically compiled modules
# (those listed by 'httpd -l') do not need to be loaded here.
#
# Example:
# LoadModule foo_module modules/mod_foo.so
#
```

Unterhalb dieses Blocks tragen Sie die folgende Zeile ein.

```
LoadModule php5_module C:/PHP/php5apache2_2.dll
```

Suchen Sie im nächsten Schritt nach diesem Block:

```
#
# If the AddEncoding directives above are commented-out, then
# you probably should define those extensions to indicate media
# types:
#
AddType application/x-compress .Z
```

```
AddType application/x-gzip .gz .tgz
```

Jetzt müssen Sie noch festlegen, dass Dateien, die die Endung *.php* besitzen, auch tatsächlich als PHP-Dateien interpretiert werden. Dazu fügen Sie unter der letzten Zeile Folgendes ein.

```
AddType application/x-httpd-php .php
```

Die PHP-Installation ist damit abgeschlossen. Nun müssen Sie nur noch die *php\_mysql.dll* aus dem Verzeichnis *php/ext* ins PHP-Hauptverzeichnis *php* verschieben.

Die gezeigte Variante verwenden Sie, wenn Sie PHP als Apache-Modul einsetzen wollen. Alternativ kann PHP aber auch als CGI-Modul genutzt werden. Dazu muss die *http.conf* ebenfalls angepasst werden. Ergänzen Sie die Datei in diesem Fall um folgende Einträge.

```
ScriptAlias /php5/ "C:/PHP/"  
AddType application/x-httpd-php .php  
Action application/x-httpd "php5/php.exe"
```

Für welche der gezeigten Varianten Sie sich letztendlich entscheiden, bleibt Ihnen überlassen. Die CGI-Version läuft nach wie vor etwas stabiler, PHP als Apache-Modul läuft hingegen ein wenig schneller – wobei die Geschwindigkeitsunterschiede wirklich nur marginal sind.

## MySQL installieren

Nach Apache und PHP muss nun die Datenbank MySQL installiert werden, denn sämtliche Inhalte, die Sie in Joomla! anlegen, werden in einer Datenbank gespeichert. Das entsprechende Windows-Paket von MySQL kann unter <http://www.mysql.de/downloads/mysql> heruntergeladen werden.

Unter *MySQL Community Server 5.5.25a* wählen Sie *Windows (x86, 32-bit)*, *MSI Installer*. Wenn Sie ein 64-Bit-Betriebssystem verwenden, laden Sie sich die Version *Windows (x86, 64-bit)*, *MSI Installer* herunter. In aller Regel werden Sie aber mit einem 32-Bit-System arbeiten. Die Installation wird durch Anklicken der heruntergeladenen MSI-Datei gestartet. Den ersten Schritt überspringen Sie mit **NEXT**. Akzeptieren Sie die Lizenzbestimmungen und klicken Sie erneut auf **NEXT**. Bei der Installationsvariante stellen Sie *Custom* ein und bestätigen mit **NEXT**. Über *Browse* stellen Sie das Installationsverzeichnis (typischerweise *C:\MySQL*) ein. Mit **NEXT** und **INSTALL** wird nun die Installation eingeleitet.

Nach erfolgreicher Installation klicken Sie sich so lange mit **NEXT** durch den Installationsassistenten bis das Fenster *Wizard Completed* angezeigt wird. Überprüfen Sie, ob das Kontrollkästchen *Launch the MySQL Instance Configuration Wizard* aktiviert ist, und klicken Sie auf **FINISH**. Daraufhin öffnet sich der Configuration Wizard, der Assistent zum Konfigurieren von MySQL. Bestätigen Sie das erste Fenster mit **NEXT**. Nun beginnt das eigentliche Einrichten des Datenbankservers. Rufen Sie dazu *Standard Configuration* auf und bestätigen Sie dies mit **NEXT**. Im nächsten Dialogfenster sollten die beiden Kontrollkästchen *Install As Windows Service* und *Launch the MySQL Server automatically* aktiviert sein. Zusätzlich aktivieren Sie *Include Bin Directory in Windows PATH*.

Mit **NEXT** geht es weiter zum nächsten Schritt. Der Haken bei *Modify Security Settings* bleibt gesetzt. Weisen Sie anschließend dem *root* ein Passwort zu. Die Installation kann dann mit

hilfe des Assistenten zu Ende gebracht werden. Klicken Sie abschließend auf EXECUTE und FINISH.

Im letzten Schritt muss nun noch dafür gesorgt werden, dass MySQL über PHP ansprechbar ist. Dazu suchen Sie in der *php.ini* (*C:\Windows*) nach folgender Zeile

```
extension=extensions\php_mysql.dll
```

und entfernen das Semikolon. Dadurch wird erreicht, dass die MySQL-Erweiterung auch tatsächlich geladen wird. Nach einem Neustart des Apache-Servers steht eine vollständige WAMP-Umgebung (Windows Apache MySQL PHP) zur Verfügung.

## ■ Eine sichere Umgebung unter Linux

Bei den meisten Linux-Distributionen ist ein LAMP-System bereits vorinstalliert. Sollte LAMP nicht direkt während der Installation mit eingerichtet worden sein, kann man das ganz einfach nachholen. Im Folgenden zeige ich Ihnen, wie Sie vorgehen müssen, wenn Sie ein LAMP-System für die prominentesten Distributionen nachträglich installieren möchten.

### Die Installation unter OpenSUSE

Unter SuSE wird YaST für die Installation verwendet. Sollten Apache, PHP und MySQL noch nicht vorhanden sein, müssen die folgenden Pakete installiert werden:

- apache2
- apache2-devil
- apache2-mod\_php5
- mysql
- php5\_mysql

Gestartet wird der Apache-Server dann folgendermaßen:

```
/etc/init.d/apache2 start
```

Um ihn anzuhalten, wählen Sie den Befehl *stop*. Eine Übersicht aller Befehle liefert übrigens *help*.

Genauso einfach wird MySQL gestartet.

```
/etc/init.d/mysql start
```

Auch der Datenbankserver wird mit *stop* angehalten.

## Die Installation unter Debian und Ubuntu

Bei diesen Distributionen werden die benötigten Pakete über *apt* installiert:

- `apache-common`
- `php5`
- `mysql-common`
- `mysql-server`

Um PHP zu installieren, müssen Sie folgenden Befehl verwenden.

```
apt-get install php
```

Der Apache-Server wird über `/etc/init.d/apache2 start` gestartet. Wenn Sie MySQL starten wollen, geben Sie `/etc/init.d/mysql start` ein. Beendet werden Apache und MySQL jeweils mithilfe des `stop`-Befehls.