

Chameleon CMS/Shop Framework Systemanforderungen

Letzte Aktualisierung 05.09.2011

1. Betriebssystem

Das Chameleon Framework ist für den Einsatz auf Linux- oder FreeBSD-Servern konzipiert. Der Betrieb auf einem Windows-Server ist nicht getestet und wird von ESONO auch nicht supported.

2. Webserver / Datenbank

Als Webserver sollte **Apache 2.x** eingesetzt werden. Das Apache-Modul **mod_rewrite** ist obligatorisch. Der Betrieb mit einem **lighttpd-Server** ist theoretisch möglich, aber **nicht getestet** worden.

Der Betrieb mit einem **nginx Server** ist möglich. Die ESONO AG kann hier aber keinen Support leisten. Es ist zu beachten, dass der nginx Server in `$_SERVER` teilweise für den Betrieb von Chameleon erwartete Variablen nicht ausgibt, was dazu führt, dass man leichte Modifikationen am Controller vornehmen muss um das Verhalten des Apache zu emulieren.

Als Datenbank wird MySQL ab Version 5.0 benötigt. Andere Datenbanken werden nicht unterstützt. Wichtig: Die Datenbank muss im Format **utf8_general_ci** angelegt werden.

Beispiel einer Apache vhost Config:

```
<Directory /var/www/vhosts/yourdomain.de/httpdocs>
    <IfModule mod_php5.c>
        php_admin_flag engine on
        php_admin_value open_basedir
"/var/www/vhosts/yourdomain.de/httpdocs:/var/www/vhosts/yourdomain.de/private:/tmp"
    </IfModule>
    Options -Includes +ExecCGI FollowSymLinks
</Directory>
```

3. PHP-Konfiguration

PHP muss in der Version **5.2.6** oder neuer installiert werden. Wir empfehlen **5.3** oder neuer. Das Chameleon Framework benötigt folgende PHP-Extensions:

bcmath, ctype, curl, date, dom, gd (2.0.x), hash, iconv, json, libxml, mbstring, mcrypt, mhash, mysql, mysqli, openssl, pcre, posix, session, shmop, SimpleXML, sockets, xml, xmlrpc, zip, zlib

optional:

tidy - zur Bereinigung von Office/Word HTML (wird empfohlen)

memcache (3.0.x) – wenn Memcache als Cache-Storage-Lösung verwendet werden soll und/oder für Session-Storage

Enterprise Shop | Content Management System
CHAMELEON 

Chameleon CMS® ist ein Produkt der ESONO® AG. Die ESONO® AG mit Sitz in Freiburg ist ein langjährig erfolgreich am Markt agierendes Unternehmen für Software-Entwicklungen mit Fokus auf Internet-, Intranet- und Extranetapplikationen. Chameleon CMS® zählt zu den stabilsten und nachhaltigsten Projekt-Entwicklungssystemen, die aktuell verfügbar sind. Mit einem ständig wachsenden Partnernetzwerk aus professionellen Integrationspartnern, ausgewählten Agenturen und Systemhäusern wird bundesweit ein kontinuierlicher Ausbau des Service und der Systemweiterentwicklung vorangetrieben.

Die folgenden Einstellungen müssen an der php.ini oder in der vhost-Konfiguration des Webservers vorgenommen werden:

```
zend.ze1_compatibility_mode = off
short_open_tag = on
register_globals = off
file_uploads = on
allow_url_fopen = on
magic_quotes_gpc = off
session.use_cookies = on
session.use_trans_sid = off
session.serialize_handler = php
memory_limit = 64M (Kommt sehr auf die verwendeten Module an. Bei Im-/Exports wird hier evtl. ein höheres Limit benötigt. Beim Einsatz der GD-Lib ohne ImageMagick und Bilddateien im Mega-Pixel-Bereich muss dieser Wert deutlich erhöht werden.)
```

```
max_execution_time = 300 (Um CronJob-Abbrüche zu vermeiden, sollte der safe_mode deaktiviert werden, damit die CronJobs die execution time selbstständig erhöhen können.)
```

```
post_max_size = 100M
```

```
upload_max_filesize = 100M
```

(Sollten sie FLV Video Dateien mit dem System verwalten wollen, sollten Sie diese Werte entsprechend der maximalen Dateigröße erhöhen.)

Session:

Wenn für Session Management kein Memcache Server verwendet wird, ist zu beachten, dass auf Debian basierenden Systemen, wie z.B. auch Ubuntu in der PHP.ini eingestellt werden muss:

```
session.gc_probability = 1
```

Tuning:

Da das Framework mit sehr vielen Dateien arbeitet und automatisch Datenbank-Classes generiert, ist es ratsam, den PHP-Dateicache zu erhöhen.

```
realpath_cache_size = 64K (Standard ist 16K)
```

Security:

Beim Einsatz der PHP-Security-Erweiterung **suhosin** (wird dringend empfohlen anstatt safe_mode) sind folgende Einstellungen vorzunehmen:

```
suhosin.executor.allow_symlink = On
```

```
suhosin.session.cryptua = Off
```

```
suhosin.stealth = 1
```

```
suhosin.session.encrypt = Off
```

4. Sonstige Server-Software

Wir empfehlen den Einsatz von **ImageMagick** zur Bearbeitung von Bilddateien. Damit Chameleon auf die Binaries von ImageMagick zugreifen kann, müssen diese vom Webserver zugreifbar sein (z.B. durch Kopieren der binaries convert und identify in den bin-Ordner der chroot-Umgebung der Website).

Außerdem ist es empfehlenswert **tidy** einzusetzen, um nicht valide HTML-Formatierungen, die durch copy&paste aus Microsoft Office-Produkten ins System gelangen, effektiv filtern zu können. Alternativ zur Commandline-Version kann auch die tidy PHP-Extension eingesetzt werden (bevorzugt).

Um interne Jobs ausführen zu können muss in regelmäßigen Abständen die URL /cms.php?pagedef=runcrons aufgerufen werden. Dazu müssen sie einen **Cronjob** auf ihrem System einrichten, der diese URL z.B. alle 15min per wget, lynx oder einem anderen Programm aufruft.
Crontab Beispiel:

```
0,15,30,45 * * * * /usr/bin/lynx -dump http://yourdomain/cms.php?pagedef=runcrons >/dev/null 2>&1
```

5. Client Anforderungen

Das Content Management Backend wird primär für den Firefox 3.x entwickelt. Die Bedienung ist aber auch in Internet Explorer 7 sowie Safari und Chrome mit Einschränkungen möglich. Beim Einsatz des Internet Explorer 8 sollte dieser in den IE7-Kompatibilitätsmodus geschaltet werden, um Probleme zu vermeiden.

Zur Bedienung des Content Management Backends werden Cookies benötigt.

Um multiple Datei-Uploads mit der Anzeige von Fortschrittsbalken zu ermöglichen setzt, Chameleon eine Flash-Upload-Komponente ein. Hierfür wird das Flash Plugin ab Version 9.x benötigt.

6. mySQL-Server-Anforderungen

Chameleon verwendet eine mySQL-Datenbank zur Speicherung von Daten. Die Tabellen sind dabei fast alle im *myISAM* Typ angelegt. Ausnahmen sind die **Cache**, **Shop Suchindex** und **Session** Tabellen, die das *InnoDB* Format nutzen.

Zusätzlich wird ab Version 3.2. für eine **Lock** Tabelle der Typ *memory* eingesetzt.

Die folgenden Rechte muss der Datenbank-Benutzer zur Verfügung haben, um keine Funktionen zu beeinträchtigen.

Daten: SELECT, INSERT, UPDATE, DELETE (auf alle Tabellen)

Struktur: CREATE TABLE, CREATE TEMPORARY TABLES, ALTER, INDEX, DROP,

Admin: SHOW TABLES, LOCK TABLES

Tuning Empfehlungen:

Erhöhung des Cachespeichers der INNODB-Engine (default 8MB).

<i>innodb_buffer_pool_size</i>	256M (mindestens, eher 1GB oder höher)
<i>innodb_additional_mem_pool_size</i>	20M

Bitte beachten Sie, dass mySQL sehr viele Parameter bietet und eine individuelle Optimierung abhängig vom Lastverhalten ihrer Applikation und Server erfordert.

Hier hilft der Einsatz eines mySQL Performance Tuning Scripts wie z.B. das: „MySQL Performance Tuning Primer Script“ <http://www.day32.com/MySQL/>

Optional:

Chameleon Shop kann optional für eine schnellere Indizierung des Produkt-Katalogs die Funktion **LOAD DATA INFILE** nutzen (dazu muss in der Config aktiviert sein: CMS_SEARCH_INDEX_USE_LOAD_FILE)

Read-Only-System:

Es ist möglich, aus Sicherheitsgründen Chameleon ohne Schreibrechte zu betreiben, z.B. Frontend-Server und Edit-Server auf getrennten Systemen mit einer shared database. Das Edit-System hat einen mySQL-Benutzer mit den vollen Rechten und das Frontend Auslieferungssystem hat nur Lese-Rechte.

Dies ist nur möglich, wenn Ihre Website keinerlei Benutzerinteraktion hat, für die Schreibzugriffe notwendig sind.

Hier sind folgende Rechte notwendig:

Daten: SELECT (auf alle Tabellen), INSERT, UPDATE, DELETE (Tabellen: _cms_sessions, _cms_cache, _cms_cache_info, _cms_lockmanager)

Struktur: CREATE TEMPORARY TABLES (wenn sie ein Enterprise-System mit Freigabe-System nutzen)

Admin: SHOW TABLES

Hinweis:

Bitte beachten Sie, dass die benötigten Rechte für den mySQL-Benutzer nur für ein System im Auslieferungszustand gelten. Je nachdem, welche Funktionen sie implementieren, können weitere Rechte notwendig sein.

7. Performance Tipps

Wir empfehlen den Einsatz eines **Bytecode Caches**, z.B. eAccelerator, Xcache oder APC.

Bei High-Traffic-Websites ist der Einsatz von **memcached** als Storage für den CMS Object Cache sehr zu empfehlen. Hier ist auch der Einsatz eines Clusters möglich.

Außerdem sollte man das PHP-Session-Management ebenfalls auf memcached umstellen. Bitte beachten, dass für die Sessions eine eigene memcached-Instanz benötigt wird, da sonst beim Flush des Object-Caches auch alle Sessions gelöscht werden.

Statische Seiteninhalte sollten auf ein Content Delivery Network gespiegelt werden oder über einen lighttpd-Webserver ausgeliefert werden (z.B. von lokal in Kombination mit mod_proxy).