

Mit dem Managed Center und den Magento-Einstellungen Schritt für Schritt zur besseren Performance

Geschwindigkeit und Stabilität bilden den Grundstock eines erfolgreichen Onlineshops. Wenn die Ladezeit Ihres Magento-Shops nicht dem Sollwert entspricht, können Sie ihn mit nur wenigen Handgriffen anwendungs-, server- und frontend-seitig optimieren. In diesem Guide zeigen wir Ihnen, wie Sie die Performance mit Hilfe der Magento-Einstellungen und des Managed Centers von maxcluster gezielt steigern können:

Anwendungsoptimierung	2
Aktuelle PHP-Version verwenden	2
Magento-Indexverwaltung optimieren	3
Varnish verwenden	4
Redis verwenden	5
Serverseitige Optimierung	7
OPCache optimieren	7
Software-Komprimierung einrichten	8
MySQL-Einstellungen optimieren	9
FPM optimieren	9
HTTP/2 aktivieren	11
IPv6 nutzen	11
Frontend-Optimierung	12
CSS/JS minimieren	12
Bilder optimieren	12
Alternatives Frontend installieren	12
Weitere Performance-Engpässe lösen	13
Sie haben Fragen?	13

Anwendungsoptimierung

Aktuelle PHP-Version verwenden

Da neuere PHP-Versionen i.d.R. performanter sind als ihre Vorgänger, kann ihr Einsatz im Hinblick auf die Geschwindigkeit eines Onlineshops durchaus lohnenswert sein. Dazu müssen Sie zunächst prüfen, mit welchen PHP-Versionen die von Ihnen genutzte Magento-Version kompatibel ist. Magento ist beispielsweise ab Version 2.4 kompatibel mit PHP 7.4. PHP 8.0 wird aktuell (Stand August 2021) noch von keiner Magento-Version unterstützt.

Im Managed Center von maxcluster kann die PHP-Version unter dem Menüpunkt **Webserver** > **Apache** bzw. **Webserver** > **NGINX** für jeden vHost einzeln eingestellt werden. Dies ist über den Reiter "Modus" in den vHost-Einstellungen möglich.

magento2-demoshop-maxcluster.de

Bearbeiten

ALLGEMEIN | SSL-ZERTIFIKAT | SSL-OPTIONEN | **MODUS** | PHP

Globalen FPM-Pool für diese Domain nutzen

PHP 5.6 | PHP 7.0 | PHP 7.1 | PHP 7.2 | PHP 7.3 | **PHP 7.4** | PHP 8.0

Dynamic Scheduler

Prozesse: 1 | 20

STANDARD WIEDERHERSTELLEN | **SPEICHERN**

Um die korrekte PHP-Version auf der Kommandozeile auszuwählen, kann diese z.B. mit dem Befehl `/usr/bin/php7.4` direkt angesprochen werden.

Alternativ kann die Standardversion über das Managed Center festgelegt werden!



PHP CLI VERSION

- PHP 5.6 aktivieren
- PHP 7.0 aktivieren
- PHP 7.1 aktivieren
- PHP 7.2 aktivieren
- PHP 7.3 aktivieren
- PHP 7.4**
- PHP 8.0 aktivieren

Magento-Indexverwaltung optimieren

Magento nutzt Datenstrukturen in MySQL und Elasticsearch, um Daten wie Preise und Verfügbarkeit schneller abzurufen. Da sich der Status dieser Daten (z.B. bei Preiserhöhungen) auch ändern kann, müssen Indexer regelmäßig neu aufgebaut werden.

Dieser Prozess erfolgt entweder direkt nach dem Speichern einer Änderung oder via Cronjob in einem festgelegten Rhythmus. Da bestimmte Indizes auch durch Benutzeraktionen neu geschrieben werden können, kann ein Echtzeit-Index unter Umständen die Performance beeinträchtigen.

Es ist daher empfehlenswert, entsprechende Indexer auf **“Update by schedule”** zu stellen. Die Details dazu sind in der [Magento-Dokumentation](#) zu finden.

Mit dem Befehl `php bin/magento indexer:show-mode` lässt sich der Status der einzelnen Indexer innerhalb von Magento anzeigen.

```
15:34:22 c-500 ~/magento2-demoshop-maxcluster.de/2.4.2-p1 $ php bin/magento indexer:show-mode
Design Config Grid:      Update by Schedule
Customer Grid:          Update by Schedule
Product Flat Data:      Update by Schedule
Category Flat Data:     Update by Schedule
Category Products:      Update by Schedule
Product Categories:     Update by Schedule
Catalog Rule Product:   Update by Schedule
Product EAV:            Update by Schedule
Stock:                  Update by Schedule
Inventory:              Update by Schedule
Catalog Product Rule:   Update by Schedule
Product Price:          Update by Schedule
Catalog Search:         Update by Schedule
```

Über den Befehl `php bin/magento indexer:set-mode schedule #Indexer#` kann der jeweilige Indexer anschließend so umgestellt werden, dass er nicht mehr in Echtzeit ausgeführt wird.

```
15:34:15 c-500 ~/magento2-demoshop-maxcluster.de/2.4.2-p1 $ php bin/magento indexer:set-mode schedule catalogsearch_fulltext
Index mode for Indexer Catalog Search was changed from 'Update on Save' to 'Update by Schedule'
```

Varnish verwenden

Ein wichtiger Aspekt der Magento Performance ist Caching. Der Magento HTTP Cache ist standardmäßig aktiv und kann in den Full-Page-Cache Varnish gelegt werden, um eine bessere Performance zu erzielen. Eine Anleitung zur Konfiguration finden Sie ebenfalls in der [Magento-Dokumentation](#).

Varnish muss zunächst in den Magento-Einstellungen aktiviert werden. Dies ist unter **Stores > Settings > Configuration > Advanced > System > Full Page Cache** möglich. Der Port muss bei maxcluster auf 8.000 gesetzt werden.

Eine passende VCL-Datei kann ebenfalls direkt dort exportiert werden. Im nächsten Schritt muss Varnish im maxcluster-Interface aktiviert werden, um im Anschluss daran dort die VCL-Datei hinterlegen zu können. Hier ist zu beachten, dass die VCL-Datei immer für den kompletten Cluster gilt. Bei Bedarf sollten daher Ausnahmen für andere Domains hinterlegt werden. Es ist ratsam, auch den SSL-Offloader direkt hier zu aktivieren.

Beim Hinterlegen der VCL-Datei empfehlen wir, den Healthcheck aus der Datei zu entfernen. Dieser ist bedingt durch unser Hochverfügbarkeits-Setup nicht notwendig und beinhaltet das Risiko, dass sich die Downtime nach einer temporären Störung erhöht. Alle notwendigen Varnish Einstellungen befinden sich im Managed Center unter **Webserver > Varnish SSL**.

The image shows a screenshot of the maxcluster management interface. On the left is a navigation menu with categories like 'Dashboard', 'Webserver', 'Datenbank & Suche', 'Cache & Queue', 'Sicherheit & Backups', 'E-Mail', and 'DNS'. The main content area is titled 'Varnish SSL' and includes a status toggle set to 'An' (On). Below this, there are buttons for 'KONFIGURATION' and 'STARTPARAMETER'. A 'Backends' section shows a 'default' backend with IP '127.0.0.1:8000'. An 'SSL' section also has a status toggle set to 'An' and shows the 'SSL Domain' as 'magento2-demoshop-maxcluster.de'. An inset window shows a VCL file editor with the following code:

```
1 # VCL version 5.0 is not supported so it s
2 vcl 4.0;
3
4 import std;
5 # The minimal Varnish version is 6.0
6 # For SSL offloading, pass the following
7
8 backend default {
9     .host = "127.0.0.1";
10    .port = "8000";
11    .first_byte_timeout = 600s;
12 }
13
14 acl purge {
15     "127.0.0.1";
16 }
17
18 sub vcl_recv {
19     if (req.restarts > 0) {
20         set req.hash_always_miss = true;
21     }
22
23     if (req.method == "PURGE") {
24         if (client.ip !~ purge) {
25             return (synth(405, "Method not
26         }
27         # To use the X-Pool header for pur
28         # has been added to the response i
29         # capistrano-magento2 gem for purg
30         if (!req.http.X-Magento-Tags-Patte
31             return (synth(400, "X-Magento-
32         }
33         if (req.http.X-Magento-Tags-Patter
34             ban("obj.http.X-Magento-Tags ~
35         }
36         if (req.http.X-Pool) {
37             ban("obj.http.X-Pool ~ " + req.h
38         }
39     }
```

Redis verwenden

Gerade bei mittleren und größeren Onlineshops kann es im Hinblick auf die Performance sehr lohnenswert sein, den Magento Cache und die Sessions vom Dateisystem in den Keyvalue-Store Redis zu verschieben. Die Details dazu haben wir in einer [eigenen Blogreihe](#) veröffentlicht.

Um Redis bei maxcluster nutzen zu können, müssen über das Managed Center zunächst die Instanzen erstellt werden. Dabei ist es wichtig, die Instanzen logisch zu trennen. Die Sessions sollten daher nicht in derselben Instanz gespeichert werden wie der Magento Cache.

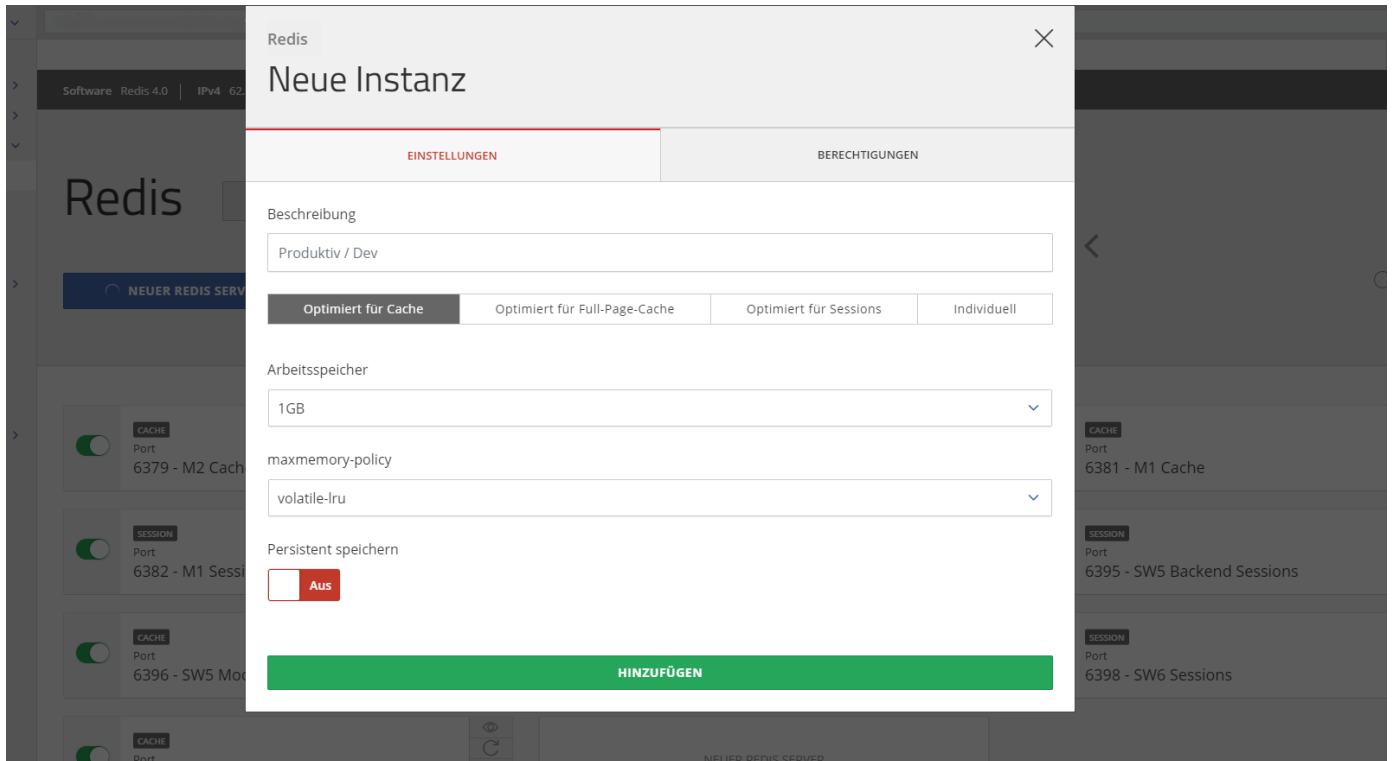
The screenshot displays the Redis management interface. At the top left, the word 'Redis' is shown next to a 'MONITORING' button. Below this is a blue button labeled 'NEUER REDIS SERVER'. The main area contains a grid of instance cards. Each card has a green toggle switch, a 'CACHE' or 'SESSION' label, a port number, and a name. The instances are as follows:

Category	Port	Name
CACHE	6379	M2 Cache
SESSION	6380	M2 Session
CACHE	6381	M1 Cache
SESSION	6382	M1 Sessions
SESSION	6394	SW5 Frontend Sessions
SESSION	6395	SW5 Backend Sessions
CACHE	6396	SW5 Model Cache
CACHE	6397	SW5 Backend Cache
SESSION	6398	SW6 Sessions
CACHE	6399	SW6 Cache

At the bottom center of the grid, there is a button labeled 'NEUER REDIS SERVER'.

Alle angelegten Instanzen werden im Managed Center nach Cache und Session gegliedert aufgeführt.

Um neue Instanzen anzulegen, klicken Sie auf den Button "Neuer Redis Server" und nehmen dann die entsprechenden Einstellungen vor.



Cache-Instanzen müssen ausreichend groß gewählt werden. Eine Einschätzung über die benötigte Kapazität kann die Größe des Caches im Dateisystem geben.

Anschließend muss Magento noch für die Verwendung von Redis konfiguriert werden. Sessions werden in Redis mit folgendem Codeblock abgelegt:

```
bin/magento setup:config:set -q --session-save=redis  
--session-save-redis-port=0 --session-save-redis-db=2  
--session-save-redis-host=/var/run/redis/6379.sock
```

Den Cache können Sie folgendermaßen in Redis ablegen:

```
bin/magento setup:config:set -q --cache-backend=redis  
--cache-backend-redis-port=0 --cache-backend-redis-db=1  
--cache-backend-redis-server=/var/run/redis/6380.sock
```

Serverseitige Optimierung

OPCache optimieren

Damit das PHP-Skript vor der Ausführung nicht jedes Mal in ein maschinenlesbares Format umgewandelt werden muss, ist es möglich, das Ergebnis nach dem ersten Aufruf mit dem PHP-OPCache im Arbeitsspeicher zu cachen. Der aktuelle Status des OPCaches kann über eine spezielle Webseite oder den PHP-Socket mit dem CLI-Tool [cachetool](#) analysiert werden. Bei Bedarf prüft und optimiert der maxcluster-Support den OPCache gerne für Sie.

Es ist auch möglich, den OPCache selbst zu überprüfen. Über den Socket kann der Status ohne PHP-Script ausgelesen werden. Diese Aktion darf jedoch nicht direkt nach einem PHP-Neustart durchgeführt werden, da der OPCache durch den Neustart geleert wird und sich anschließend neu aufbauen muss.

Anhand der vom CLI-Tool ausgelesenen Ergebnisse kann anschließend kontrolliert werden, ob die Hitrate dem Sollwert entspricht (sie sollte über 99 % liegen), und der Cache selbst sowie der String Buffer nicht voll sind. Eine Optimierung ist aktuell nur über den maxcluster-Support möglich.

```
16:43:08 c-500 - $ cachetool --fcgi=/var/run/php7.4-fpm-magento2-demoshop-maxcluster.de.sock opcache:status
```

Name	Value
Enabled	Yes
Cache full	No
Restart pending	No
Restart in progress	No
Memory used	134.12 MiB
Memory free	377.87 MiB
Memory wasted (%)	6.71 KiB (0.0012800097465515%)
Strings buffer size	24 MiB
Strings memory used	9.08 MiB
Strings memory free	14.92 MiB
Number of strings	140190
Cached scripts	6072
Cached keys	11925
Max cached keys	32531
Start time	Tue, 06 Jul 2021 05:04:23 +0000
Last restart time	Never
Oom restarts	0
Hash restarts	0
Manual restarts	0
Hits	1100860642
Misses	6078
Blacklist misses (%)	0 (0%)
Opcache hit rate	99.999447889568

Software-Komprimierung einrichten

Mit Modulen wie gzip und Brotli werden alle ausgelieferten Daten einer Webseite zusätzlich komprimiert. Auf Servern von maxcluster sind sowohl gzip als auch Brotli standardmäßig bereits aktiviert, sodass hier keine weiteren Anpassungen notwendig sind.

Über den Header "content-encoding" können Sie nachvollziehen, welche Komprimierung konkret zum Einsatz kommt.

🔍 Kopfzeilen durchsuchen

Blockieren | Erneut senden ↕

▶ GET https://magento2-demoshop-maxcluster.de/static/version1621960723/frontend/Magento/luma/de_DE/fonts/opensans/bold/opensans-700.woff2

Status	200 OK ⓘ
Version	HTTP/2
Übertragen	17,25 KB (17,25 KB Größe)
Referrer Policy	strict-origin-when-cross-origin

▼ Antwortkopfeilen (384 B)

Unformatiert

- ⓘ [cache-control](#): max-age=31536000, public
- ⓘ [content-encoding](#): br
- ⓘ [content-type](#): application/font-woff2
- ⓘ [date](#): Sun, 01 Aug 2021 13:39:20 GMT
- ⓘ [expires](#): Mon, 01 Aug 2022 13:39:20 GMT
- ⓘ [last-modified](#): Tue, 25 May 2021 16:40:17 GMT
- ⓘ [server](#): nginx/1.20.1
- ⓘ [strict-transport-security](#): max-age=2592000; includeSubDomains; preload
- [X-Firefox-Spdy](#): h2
- ⓘ [x-frame-options](#): SAMEORIGIN

MySQL-Einstellungen optimieren

Die Performance eines Magento-Shops kann auch mit Hilfe der Einstellungen des Datenbankservers verbessert werden. Der InnoDB Buffer Pool sollte beispielsweise groß genug sein, um alle verwendeten InnoDB vollständig im Cache halten zu können.

Der aktuell eingestellte Wert lässt sich in der MySQL-Konsole über den Befehl `show variables like "innodb_buffer_pool_size";` prüfen.

Mit diesem Befehl kann die Größe der InnoDB-Datenbanken eingesehen werden:

```
select sum(DATA_LENGTH) + sum(INDEX_LENGTH) FROM information_schema.TABLES
WHERE ENGINE like "InnoDB" AND table_schema not in ("mysql",
"performance_schema", "information_schema");
```

```
16:57:17 localhost [(none)]> select sum(DATA_LENGTH) + sum(INDEX_LENGTH) FROM information_schema.TABLES WHERE ENGINE like "InnoDB" AND table_schema not in ("mysql", "performance_schema", "information_schema");
+-----+
| sum(DATA_LENGTH) + sum(INDEX_LENGTH) |
+-----+
| 657227776 |
+-----+
1 row in set (3.544 sec)

16:57:27 localhost [(none)]> show variables like "innodb_buffer_pool_size";
+-----+-----+
| Variable_name | Value |
+-----+-----+
| innodb_buffer_pool_size | 8589934592 |
+-----+-----+
1 row in set (0.001 sec)
```

Prüfung und Optimierung sind über den maxcluster-Support möglich.

FPM optimieren

PHP kann über mod_php direkt aus Apache heraus ausgeführt werden. Es ist allerdings wesentlich effizienter, diesen Prozess mit dem FastCGI-Prozessmanager PHP-FPM in einen eigenständigen Dienst auszulagern. Abhängig vom erwarteten Traffic und den verfügbaren Hardware-Ressourcen kann die Anzahl der PHP-Worker angepasst werden.

Die Wahl des PHP-Schedulers hat ebenfalls Auswirkungen auf die Performance: Während der OnDemand-Scheduler die PHP-Auslastung durch Abschalten nicht benötigter Worker sehr gering hält, sorgt der Dynamic Scheduler dafür, dass der Shop auch bei einem Anstieg des Traffics performant bleibt.

Im Managed Center von maxcluster können die FPM-Einstellungen unter dem Menüpunkt **Webserver > Apache** bzw. **Webserver > NGINX** für jeden vHost einzeln vorgenommen werden.

magento2-demoshop-maxcluster.de

Bearbeiten

ALLGEMEIN | SSL-ZERTIFIKAT | SSL-OPTIONEN | **MODUS** | PHP

Globalen FPM-Pool für diese Domain nutzen

PHP 5.6 | PHP 7.0 | PHP 7.1 | PHP 7.2 | PHP 7.3 | **PHP 7.4** | PHP 8.0

Dynamic Scheduler

Prozesse

1 | 20

STANDARD WIEDERHERSTELLEN | **SPEICHERN**

Um zu validieren, ob die Anzahl der PHP-Worker ausreichend ist, kann die PHP-Log-Datei geprüft werden. Bei maxcluster befindet sich diese unter `/var/www/share/logs/php7.4-fpm.log`.

Wenn dort regelmäßig die Meldung `server reached pm.max_children setting` erscheint, sind wahrscheinlich nicht genügend Worker vorhanden. In diesem Fall können Sie die Anzahl der eingesetzten PHP-Worker unter dem Reiter "Prozesse" manuell anpassen.

Prozesse

1 | 20

HTTP/2 aktivieren

Eine serverseitige Aktivierung des 2015 releasteen HTTP-Protokoll-Upgrades HTTP/2 ist unserer Erfahrung nach empfehlenswert. Das lässt sich mit Hilfe von Browser- oder Web-Tools überprüfen. In den Browser-Tools wird der Einsatz von HTTP/2 anhand der Versionsangabe ersichtlich. Wenn dort HTTP/2 angezeigt wird, wurde die Ressource mit HTTP/2 übertragen.

🔍 Kopfzeilen durchsuchen

Blockieren | Erneut senden

▶ GET https://magento2-demoshop-maxcluster.de/static/version1621960723/frontend/Magento/luma/de_DE/fonts/opensans/bold/opensans-700.woff2

Status **200 OK** ⓘ

Version HTTP/2

Bei maxcluster ist HTTP/2 bei allen Kunden mit Ubuntu-Clustern bereits standardmäßig eingebunden. Der 2018 fertiggestellte HTTP/3-Standard ist aktuell noch nicht mit NGINX und Apache kompatibel. Sobald für beide Webserver eine HTTP/3-Unterstützung existiert, wird deren Nutzung zeitnah danach auch bei maxcluster möglich sein.

IPv6 nutzen

Seit vielen Jahren werden IPv4-Adressen im Web nach und nach auf das neuere und performantere IPv6-Protokoll umgestellt. Obwohl einige Provider das IPv4-Protokoll bereits nicht mehr unterstützen, bleiben reine IPv4-Webdienste aber durch Übergangstechniken wie DSLite weiterhin erreichbar. Da diese Übergangstechniken störungsanfällig und aufgrund der zusätzlichen Komplexität schlecht für die Performance sind, ist eine Umstellung auf IPv6 trotzdem sinnvoll.

Um IPv6 bei maxcluster zu verwenden, muss sichergestellt werden, dass ein passender AAAA DNS Eintrag existiert. Serverseitig sind die maxcluster-Server bereits für IPv6 konfiguriert. Sobald der AAAA DNS Eintrag angelegt wurde, ist der Magento-Shop daher per IPv6 erreichbar. Die korrekte IPv6 Adresse ist im Managed Center unter **Webserver > Apache** bzw. **Webserver > NGINX** zu finden.

Sofern auch die DNS-Verwaltung bei maxcluster liegt, kann der AAAA Eintrag im Managed Center unter DNS angelegt werden.

magento2-demoshop-maxcluster.de

AAAA

2a00:f48:1008::231:72:10

Frontend-Optimierung

CSS/JS minimieren

Abhängig von der Größe und Anzahl der vorhandenen CSS-Dateien kann die Performance der Website beeinträchtigt werden. Im Backend von Magento können die Dateien zugunsten der Ladezeit minimiert werden. Eine Komprimierung erfolgt beispielsweise durch das Entfernen überflüssiger Whitespaces, Zeilenumbrüche oder Kommentare. Magento verfügt im Backend über Einstellungen, um diesen Vorgang [automatisch durchzuführen](#).

Bilder optimieren

Auf Clustern von maxcluster sind bereits einige Tools zur Optimierung von Bildern vorinstalliert. Dazu gehören jpegoptim, jpegtran, pngcrush, optipng, pngout und cwebp. Obwohl Magento selbst das WebP-Format noch nicht unterstützt, ist es möglich, dieses mit Plugins nachzurüsten. Die Bildoptimierung selbst kann ebenfalls mit Hilfe eines Plugins durchgeführt werden. Das geschieht im besten Fall direkt nach dem Upload eines neuen Bildes.

Auch über die Kommandozeilen ist es möglich, Bilder zu optimieren. Mit diesen beiden Befehlen ist eine automatische Optimierung von z.B. allen png- und jpeg-Dateien im aktuellen Ordner möglich:

```
find -name "*.png" -exec optipng -o7 -strip all {} \;
```

```
find -name "*.jpg*" -exec jpegoptim -o --strip-all {} \;
```

Alternatives Frontend installieren

Das standardmäßig von Magento vorinstallierte Luma-Theme ist im Hinblick auf die Performance-Optimierung limitiert. Eine mögliche Alternative mit deutlich mehr Optimierungspotenzial ist [Hyvä](#), ein von Grund auf neu entwickeltes und auf Performance ausgelegtes Theme für Magento. Weitere Alternativen sind Lösungen auf Basis einer Progressive Web App (PWA): Dazu zählen z.B. das von Magento entwickelte [PWA Studio](#) oder [Vue Storefront](#), eine PWA-Lösung, die weitestgehend unabhängig vom verwendeten Shopsystem eingesetzt werden kann.

Weitere Performance-Engpässe lösen

Besteht ein spezifisches Problem, das die Performance eines Onlineshops beeinträchtigt, muss zunächst die Ursache des Performance-Engpasses identifiziert werden. Diese kann bspw. in den Servereinstellungen, Hardware-Ressourcen, der PHP-Verarbeitung oder auch dem Magento-Frontend liegen.

→ Servereinstellungen

Prüfen Sie die Parameter der verwendeten Dienste (z.B. die Anzahl an PHP-Workern).

→ Hardware

Stellen Sie sicher, dass zum Betreiben Ihres Onlineshops genügend RAM zur Verfügung steht und die CPU-Load mittelfristig unter der Anzahl der CPU-Kerne liegt. Im Managed Center von maxcluster können Sie die benötigten Hardware-Ressourcen mit nur wenigen Klicks anpassen.

→ Frontend

Mit Tools wie Google Pagespeed Insight oder WebPageTest können Sie Performance-Probleme im Frontend von Magento eingrenzen und identifizieren.

→ PHP-Verarbeitung

Liegt der Flaschenhals in der PHP-Verarbeitung, unterstützt ein Profiler wie Tideways bei der Analyse. Im Managed Center von maxcluster kann unter dem Menüpunkt **Webserver** > **Profiler** ein Account von Tideways, New Relic APM oder Blackfire IO hinterlegt werden.



Sie haben Fragen?

Sie haben eine Frage oder benötigen kompetente Unterstützung bei der Lösung eines auftretenden Problems? Sie erreichen unser Service-Team telefonisch unter der Nummer 05251/414130 oder per E-Mail an support@maxcluster.de.