



Dokumentation Image Cache Management
(v2.x)

Beschreibung

Das Modul "Image Cache" verbessert den Umgang und Aufbau des Bild-Caches. Es ermöglicht gezielten Neuaufbau von Bildern im Cache und optimiert auch den Bild-Upload.

Funktionsmerkmale

Funktion	Beschreibung
Bild-Cache-Invalidierung bei Bild-Upload	Wenn Bilder z. B. Produktbilder hochgeladen werden, kann der Bild-Cache bzgl. des Uploads invalidiert werden.
Aufbau Bild-Cache mit CLI-Kommando	Es ist möglich den Bild-Cache mit einem Magento-CLI-Kommando gezielt aufzubauen. Es werden nur betroffene Bilder aktualisiert.
Aufbau Bild-Cache mit Cron-Job	Es ist möglich den Bild-Cache mit einem konfigurierbaren Magento-Cron-Job gezielt aufzubauen. Es werden nur betroffene Bilder aktualisiert.
Logausgabe des Cron-Jobs	Die Ausführung bzw. das Ergebnis des Cron-Jobs zur Bild-Cache-Generierung kann konfigurativ mit einer Log-Ausgabe überwacht werden
Pipeline-Step-Konfiguration für TechDivision Pacemaker	Das Modul stellt eine Pipeline-Step-Konfiguration (Xml-Datei) bereit, welche bei einer Pacemaker-Installation genutzt wird. Es wird ein zusätzlicher Pipeline-Schritt z. B. bei einem Produktimport ausgeführt.

Contributors

Patrick Mehringer	p.mehringer@techdivision.com	Developer, Maintenance, Documentation
Martin Eisenführer	m.eisenfuehrer@techdivision.com	Maintenance

Requirements / Installation

Magento Version Compatibility

Magento Version	Latest Version
Magento 2.4.* Opensource (CE) / Commerce (EE)	composer require "techdivision/imagecache ^2.1.0"
Magento 2.3.* Opensource (CE) / Commerce (EE)	composer require "techdivision/imagecache ^1.1.0"

PHP Version

Compatible zu PHP Version ~7.3.0 || ~7.4.0

Installation mit Composer

- Um im **TechDivision** Context ein Modul mittels Composer zu installieren, bitte per folgenden Befehl das Repo entsprechend einbinden

```
composer config repositories.repo.met.tdintern.de composer https://repo.met.tdintern.de/
```

Modul Installationsbefehle

Nach Einbindung des **MET-Composer-Repository** folgende Befehle zur Installation ausführen:

```
# add to composer require
composer require techdivision/imagecache ^2.1.0

# run magento setup to activate the module
bin/magento set:up
```

Aktivieren des Moduls

Das Modul ist standardmäßig nach der Installation im Magento Backend verfügbar.

TIP

Die Modulfunktionalität ist initial aktiviert!

[TechDivision >> Performance & Images >> Image Cache Management](#)

Deinstallation

Folgende Punkte sind bei einer Deinstallation eines Moduls stets zu beachten:

- Gibt es Einträge in der Datenbank, die vor dem deinstallieren bereinigt werden müssen?
- Sind evtl. Media Files (Images, Videos ect.) vorhanden, die vorab bereinigt werden müssen?
- Gibt es Konfigurationspfade in der Config (DB), die entfernt werden müssen?
- Caches entleeren
- Indexer neu starten wenn notwendig

```
bin/magento module:uninstall techdivision_imagecache
```

Modul Konfiguration

Navigieren Sie zu [TechDivision >> Performance & Images >> Image Cache Management](#)

Modul-Optionen

Section	Option	Value	Default	Beschreibung
Module Settings	Enable	Yes/No	Yes	Aktiviert bzw. Deaktiviert die Konfiguration und Modulfunktionalitäten
Cache Generation	Enable	Yes/No	Yes	Aktiviert bzw. Deaktiviert die Cache-Generierungsfunktionen
Cache Generation	Enable Cron	Yes/No	No	Aktiviert bzw. Deaktiviert die Cache-Generierungsfunktion über den Cron-Job/Cron-Gruppe
Cache Generation	Schedule for cache generation	Text (Cron-Format)	0 1 * * *	Legt die Ausführzeit des Cron-Jobs fest. Standardmäßig jede Nacht um 01:00 Uhr
Cache Invalidation	Enable	Yes/No	No	Aktiviert bzw. Deaktiviert die Cache-Invalidation beim Hochladen von Bildern
Logging	Log Level	Level Select	Debug	Legt fest ab welchem Log-Level eine Log-Ausgabe für das Modul (z.B. bei Ausführung des Cron-Jobs) geschrieben wird.

Module Settings

Enable [global] Use system value
Enables the module configuration and functionality

Cache Generation

Enable [global] Use system value
Enables the cache generation functionality

Enable Cron [global] Use system value
Enables the cache generation via cron execution

Schedule for cache generation [global]
e.g. "0 1 * * *" (every night at 01:00) Use system value

Cache Invalidation

Enable [global] Use system value
Enables the cache invalidation functionality on uploading images

Logging

Log Level [global] Use system value
Defines the log-level for logging output (eg. cron job)

Bedienungsanleitung Modul-Funktionen

- Installation des Moduls
- Konfigurationseinstellungen des Moduls
- Cache-Generierung via CLI-Kommando
- Cache-Generierung via Cron-Job
- Cache-Invalidierung beim Image-Upload

Cache-Generierung via CLI-Kommando

1. Navigieren Sie im Backend zu **TechDivision >> Performance & Images >> Image Cache Management**
2. Konfigurieren Sie das Modul wie folgt:
 - a. Das Modul muss aktiviert sein.
 - b. Die Cache-Generierung muss aktiviert sein.
3. Speichern Sie die Konfiguration und leeren Sie den Cache.
4. Führen Sie im Quellverzeichnis den Befehl `bin/magento techdivision:images:generatecache` aus.

Das Kommando generiert nur die fehlenden Daten im Cache. Es handelt sich also um eine Delta-Generierung.

```
→ 2.3.4 git:(TDMET-32-demo-modules) X bin/magento techdivision:images:generatecache
Module "TechDivision ImageCache" not enabled!
→ 2.3.4 git:(TDMET-32-demo-modules) X bin/magento techdivision:images:generatecache
Generate cache images for new images (delta)
1000/1000 [=====] 100% 1 min 74.5 MiB | /a/9/a9b7ba70783b617e9998dc4dd82eb3c5.jpg
Product images resized successfully
→ 2.3.4 git:(TDMET-32-demo-modules) X bin/magento techdivision:images:generatecache
Generate cache images for new images (delta)
1000/1000 [=====] 100% 2 secs 76.5 MiB | /a/9/a9b7ba70783b617e9998dc4dd82eb3c5.jpg
Product images resized successfully
→ 2.3.4 git:(TDMET-32-demo-modules) X █
```

TIP

Die Kommando-Option `--generateFullCache` bzw. `-f` generiert den kompletten Bilder-Cache erneut. Dazu wird das Standard-Magento-Commando `bin/magento catalog:images:resize` verwendet.

Cache-Generierung via Cron-Job

1. Navigieren Sie im Backend zu **TechDivision >> Performance & Images >> Image Cache Management**
2. Konfigurieren Sie das Modul wie folgt:
 - a. Das Modul muss aktiviert sein.
 - b. Die Cache-Generierung muss aktiviert sein.
 - c. Die Cache-Generierung via Cron muss aktiviert sein.
 - d. Die Ausführzeit des Crons sollte genau definiert sein (Standard = 01:00 Uhr nächtlich).

TIP | Detaillierte Einstellungen zur Cron Syntax finden Sie in [Wikipedia/Cron](#).

3. Speichern Sie die Konfiguration und leeren Sie den Cache.
4. Der Cron-Job wird in der Tabelle **cron_schedule** angelegt und ausgeführt.

schedule_id	job_code	status	messages	created_at	scheduled_at	executed_at	finished_at
1	886 generate_imagecache_delta	pending	null	2020-05-27 11:50:00	2020-05-27 23:00:00	null	null
2	885 generate_imagecache_delta	success	null	2020-05-27 11:50:00	2020-05-27 12:00:00	2020-05-27 12:00:00	2020-05-27 12:00:10

TIP | Crons können mit dem Standard-Magento-Commando `bin/magento cron:run` ausgeführt werden. Die "Schedule", also die geplante Ausführzeit, wird dabei auch beachtet.

5. Ist das Logging konfiguriert, wird eine Logausgabe während der Ausführung des Cron-Jobs geschrieben.

```
→ 2.3.4 git:(TDMET-32-demo-modules) X tail -f var/log/imagecache.log
[2020-05-27 09:12:15.342328] image_cache_logger.INFO: Start TechDivision Image Cache in delta mode! {"file":"/home/mehringerp/Workspace/MET/Magento-Commerce/2.3.4/vendor/magento/framework/Logger/Monolog.php","line":48,"class":"Magento\Framework\Logger\Monolog","function":"addRecord"}
[2020-05-27 09:12:17.085513] image_cache_logger.INFO: Finish TechDivision Image Cache in delta mode! {"file":"/home/mehringerp/Workspace/MET/Magento-Commerce/2.3.4/vendor/magento/framework/Logger/Monolog.php","line":48,"class":"Magento\Framework\Logger\Monolog","function":"addRecord"}
```

Cache-Invalidierung beim Image-Upload

1. Navigieren Sie im Backend zu **TechDivision >> Performance & Images >> Image Cache Management**
2. Konfigurieren Sie das Modul wie folgt:
 - a. Das Modul muss aktiviert sein.
 - b. Die Cache-Invalidierung muss aktiviert sein.
3. Laden Sie ein Bild z.B. über die Produktpflege im Backend hoch.

Der Cache für das Bild wird automatisch invalidiert.

Pacemaker Verbindung

Mit der Modul-Version 1.1.0 bietet das Modul die Funktionalität des CLI-Kommandos zur Delta-Bildcache-Generierung zur Ausführung innerhalb der TechDivision Pacemaker Process-Pipelines an.

Konfiguration per XML-File

Die Konfiguration ist im Modul vorhanden und ist standardmäßig wie folgt definiert:

```
<?xml version="1.0"?>
<config xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="urn:magento:module:TechDivision_ImageCache:etc/pipeline.xsd">
  <pipeline name="pacemaker_import_catalog" description="Pacemaker Product Import" use-
working-directory="true" expire-in="6 hours">
    <conditions>
      <pipeline_condition
type="TechDivision\ProcessPipelines\Helper\Condition\Pipeline\NoAutoSpawn" description="No
automatic start for this pipeline"/>
    </conditions>
    <step name="media_cache_refresh"
executorType="TechDivision\PacemakerImportBase\Model\Executor\MagentoExecutor"
sortOrder="68" description="Refresh media cache">
      <conditions>
        <step_condition
type="TechDivision\ProcessPipelines\Helper\Condition\Step\AttemptsLimit\Limit1"
description="Try once."/>
        <step_condition
type="TechDivision\ProcessPipelines\Helper\Condition\Step\PreviousStepsCompleted"
description="Previous step needs to be finished."/>
        <step_condition
type="TechDivision\PacemakerImportCatalog\Virtual\Condition\NoConflictingStepInProgress"
description="Avoid conflicts between import steps."/>
      </conditions>
      <arguments>
        <argument key="command" value="techdivision:images:generatecache" />
      </arguments>
    </step>
  </pipeline>
</config>
```

- ☒ Wenn eine Import Pipeline für den Katalog ausgeführt wird
- ☒ wird nun auch der neue Step "media_cache_refresh" ausgeführt
- ☒ welcher das Kommando "techdivision:images:generatecache" ausführt

Backend Pipeline

Das Resultat der Konfiguration während der Ausführung sieht nun folgendermaßen aus:

Pacemaker Pipelines 🔍 2 🧑 admin

[Add New Pipeline Entry](#)

Filters | Default View | Columns

4 records found | 20 per page | 1 of 1

ID	Name	Status	Has Warning	Steps	Created	Started	Finished
<input type="checkbox"/> 60	pacemaker_import_catalog	running	No		Jun 30, 2020 3:57:01 PM	Jun 30, 2020 3:57:09 PM	
<input type="checkbox"/> 61	pacemaker_import_catalog	pending	No		Jul 1, 2020 9:52:25 AM		
<input type="checkbox"/> 62	pacemaker_import_catalog	pending	No		Jul 1, 2020 9:56:23 AM		
<input type="checkbox"/> 63	pacemaker_import_catalog	pending	No		Jul 1, 2020 10:00:20 AM		

Glossar

Begriff	Beschreibung
Cache	Cache bezeichnet in der EDV einen schnellen Pufferspeicher oder Zwischenspeicher, der (wiederholte) Zugriffe auf ein langsames Hintergrundmedium oder aufwendige Neuberechnungen zu vermeiden hilft. Daten, die bereits einmal geladen oder generiert wurden, verbleiben im Cache, so dass sie bei späterem Bedarf schneller aus diesem abgerufen werden können. Auch können Daten, die vermutlich bald benötigt werden, vorab vom Hintergrundmedium abgerufen und vorerst im Cache bereitgestellt werden (read-ahead).
Magento-CLI-Kommando	Magento bietet verschiedenste Kommandos, welche über das Command-Line-Interface ausgeführt werden können. Sie haben folgendes Format <code>bin/magento <COMMAND></code>
view.xml	In dieser Datei werden alle frontend-seitigen Eigenschaften wie z. B. die Größe von Produkt-Bildern konfiguriert. Auch eine benutzerdefinierte ID eines Bildes kann hier mit seiner Größe hinzugefügt werden, welches dann in PHP generiert wird. Zusätzlich können einige Eigenschaften eingestellt werden, ohne dass dabei Template-Dateien überschrieben werden.
TechDivision Pacemaker	Mit Pacemaker stellen wir eine leistungsfähige Plattform zur Verfügung, mit der sich selbst die komplexesten Szenarien abbilden und die größte Datenmenge aus beliebigen Drittsystemen zuverlässig in Magento importiert und dort weiterverarbeitet werden können. https://www.techdivision.com/technologien/pacemaker.html