ItkAutoImageCompressor

Anwenderdokumentation

André Fischer

30.03.2026

Inhaltsverzeichnis

1 Zweck

Du lädst in Shopware ständig Bilder hoch – Produkte, Banner, CMS. Ohne Nachbearbeitung bleiben die Dateien oft größer als nötig: unnötige Metadaten, zu hohe JPEG-Qualität, riesige PNGs. **ItkAutoImageCompressor** greift **direkt nach dem Upload** ein und kann (je nach Konfiguration) verkleinern, Werkzeuge zur verlustbehafteten oder verlustfreien Komprimierung anwerfen und optional WebP erzeugen. Du musst dafür **nichts** im Alltag extra klicken, sobald alles einmal sinnvoll eingestellt ist.

Konkret kann das Plugin (alles optional über Schalter):

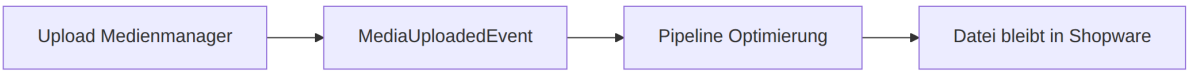
- **JPEG, PNG und/oder WebP** mit den CLI-Tools **jpegoptim**, **optipng** und **cwebp** optimieren
- per `ImageMagick convert` (nicht die PHP-Extension Imagick) Resize, EXIF-Rotation, Trim, Schärfe usw. anwenden
- **WebP-Konvertierung** neben dem Original anlegen
- **Metadaten** (EXIF, IPTC, XMP, GPS) streifen
- **Lazy-Loading-Hinweise** für die Auslieferung vorbereiten, z. B. über Konfigurationen für Placeholder oder Standardverhalten bei Bildern

Wichtig: Auf dem Server müssen die genannten **Kommandozeilen-Tools** installiert sein und für denselben Benutzer erreichbar sein, unter dem **PHP** (FPM oder CLI) läuft. Ohne Tools oder bei blockiertem **exec**/`proc_open` siehst du Fehler oder stilles Überspringen – dann zuerst **php bin/console itk:image:test** und die Logs prüfen.

---

2 Typischer Ablauf für dich im Alltag

2.1 Diagramm: Nach dem Upload



1. Du oder ein Kollege ladet eine Datei im **Medienmanager** hoch – wie gewohnt.

2. Shopware speichert die Datei und feuert intern ein **MediaUploadedEvent**.

3. Das Plugin prüft: Ist es an? Passt die Datei zu Größe, Format und Medienordner?

4. Wenn ja, läuft die Pipeline (Backup optional → ggf. Resize → Korrekturen → Kompression → ggf. WebP).

5. Wenn ein Schritt fehlschlägt, wird in der Regel **geloggt**; der Upload in Shopware bleibt bestehen (du musst nicht nochmal hochladen, aber das Bild ist ggf. unoptimiert).

Für **Bestandsmedien** nutzt du die **Console-Befehle** (siehe unten), weil der Subscriber nur bei **neuen** Uploads triggert.

---

3 Was das Plugin bewusst nicht übernimmt

- Kein Ersatz für professionelles **Studio-Retusche** – es geht um automatisierte, regelbasierte Optimierung.

- Keine **Inhalts-KI** oder intelligente Bildausschnitte (außer dem, was **convert** mit Trim & Co. kann).

- **SVG, GIF, PDF** usw. sind nicht Gegenstand der JPEG/PNG/WebP-Pipeline (werden je nach MIME ignoriert).

- **Kein CDN** und kein externes Image-CDN – alles lokal auf deinem Server bzw. im konfigurierten **mediaPath**.

---

4 Konfiguration: Grundkonfiguration

Konfiguration unter **Erweiterungen → Meine Erweiterungen** → **ITK Auto Image Compressor** → **Konfigurieren** (Sales-Channel-Vererbung wie bei Shopware üblich).

Feld	Bedeutung
Plugin aktivieren	Hauptschalter; wenn aus, reagiert der Upload-Subscriber nicht.
Media-Verzeichnis-Pfad	Absoluter Pfad zum public/media-Verzeichnis (Standard Linux-Beispiel in der Config). Auf Windows/anderen Pfaden anpassen.
Minimale Dateigröße (KB)	Kleinere Dateien werden nicht angefasst (Standard 1 KB).
Maximale Dateigröße (KB)	Größere Dateien werden übersprungen (Standard 10240 KB = 10 MB).
Original sichern	Vor Bearbeitung Kopie des Originals (mehr Speicherbedarf).
Ordner von Optimierung ausschließen	Kommagetrennte **Namen der Medienordner** (kleingeschrieben verglichen). Unterstützt **SQL-ähnliche Muster** mit **%**, z. B. **raw**, **backup**, **%import%**. Liegt eine Datei in einem so erkannten Ordner, wird **keine** Komprimierung/Resize-Pipeline ausgeführt. Standard: **raw,original,backup**.

---

5 Unterstützte Formate (Schalter)

Feld	Bedeutung
JPEG/JPG komprimieren	Nutzt **jpegoptim**.
PNG komprimieren	Nutzt **optipng**.
WebP komprimieren	Nutzt **cwebp** auf bestehende WebP-Dateien.

Mindestens ein Format sollte aktiv sein, sonst gibt es nichts zu tun.

---

6 JPEG-Einstellungen

Feld	Bedeutung
JPEG-Qualität	1–100 (Standard 85).

Feld	Bedeutung
Progressive JPEG	Ladeverhalten im Browser „von grob zu scharf“.

7 PNG-Einstellungen

Feld	Bedeutung
Optimierungsstufe	0–7 für **optipng** – höher = kleinere Datei, langsamer.
Interlacing	Progressives PNG; kann Dateigröße erhöhen.

8 WebP (Optimierung bestehender WebP-Dateien)

Feld	Bedeutung
WebP-Qualität	1–100 (wird bei verlustfrei ignoriert).
WebP verlustfrei	Verlustfreie WebP-Komprimierung.
WebP Multi-Threading	Schneller, mehr CPU.
Komprimierungsmethode	0–6, höher = besser komprimiert, langsamer.
Spatial Noise Shaping (SNS)	0–100, weniger Artefakte.
WebP Auto-Filter	Automatische Filterwahl durch **cwebp**.

9 WebP-Konvertierung (JPEG/PNG → WebP)

Feld	Bedeutung
WebP-Konvertierung aktivieren	Erzeugt zusätzlich zum Original eine WebP-Datei. Das Original kann je nach Konfiguration erhalten bleiben, damit Theme, Thumbnails oder Fremdsysteme nicht plötzlich auf ein anderes Format umgestellt werden.
Konvertierungs-Qualität	Qualität der Zieldatei.
Original beibehalten	Ob JPEG/PNG nach Konvertierung erhalten bleibt (wichtig für ältere Clients).

---

10 Bild-Resize

Feld	Bedeutung
Resize aktivieren	Skaliert Bilder nach Regeln **pro Medienordner**.
Resize-Regeln	Kommagetrennt **ordner=mMaxBreite**. Platzhalter **%** wie bei SQL-LIKE (z. B. **product%=1200**). Standardbeispiel: ***=1080,product=1200,banner=1920,thumbnail=300**. Es wird **nur verkleinert**, nie hochskaliert.

Zu schmale Bilder für eine Regel können im Log `var/log/image_resize_small.log` landen (eigener Logger).

---

11 Metadaten

Feld	Bedeutung
Alle Metadaten entfernen	Steht über Einzeloptionen (globaler Strip).
EXIF / IPTC / XMP / GPS	Feinsteuerung, soweit die Engine das unterstützt.

---

12 Automatische Bildverbesserungen (optional)

Feld	Bedeutung
Automatische EXIF-Rotation	Bild anhand Kamera-Orientierung drehen (Standard oft sinnvoll).
Helligkeit/Kontrast	Automatische Anpassung (experimentell im Alltag).
Schärfung	Nachschärfen.
Automatisches Zuschneiden (Trim)	Entfernt gleichfarbige Ränder; **Trim-Toleranz** in %.
Farbnormalisierung	Konsistentere Farben.

Praktisch läuft das in genau dieser Richtung ab: erst Bild einlesen und prüfen, dann optionale Bildverbesserungen anwenden, danach komprimieren und zum Schluss – wenn aktiviert – zusätzlich WebP erzeugen.

---

13 Lazy Loading (Storefront-Bezug)

Feld	Bedeutung
Lazy Loading aktivieren	Setzt bzw. unterstützt **loading="lazy"** für optimierte Einbindung (Details siehe Implementierung in Subscriber/Templates).
Lazy-Loading-Platzhalter	Optional Base64-Platzhalter.

---

14 CLI-Befehle (Batch / Wartung)

Befehl	Zweck
php bin/console itk:image:optimize	Medien durchgehen und optimieren (siehe `--help` für Filter).
php bin/console itk:image:optimize-all	Alle Medien erneut anfassen.

Befehl	Zweck
php bin/console itk:image:check-resize	Resize-Regeln gegen die Datenbank prüfen (Dry-Run / Analyse).
php bin/console itk:image:test	Diagnose: Config, Engine-Verfügbarkeit, Pfade.

---

15 Logs

Datei	Inhalt
var/log/image_compressor.log	Ablauf Komprimierung, Fehler, übersprungene Dateien.
var/log/image_resize_small.log	Bilder, die für Resize zu klein sind o. Ä.

---

16 Typische Probleme

Nichts passiert beim Upload: Plugin aus; falsche **mediaPath**; Datei außerhalb Min/Max; Ordner ausgeschlossen; MIME nicht in den aktivierten Formaten.

Fehler im Log „Engine not available“: jpegoptim / **optipng** / **cwebp** fehlen oder PATH für PHP-FPM falsch.

Qualität zu schlecht: JPEG-Qualität erhöhen; WebP nicht verlustfrei mit zu aggressiven Methoden kombinieren.

Unterschied CLI vs. Web: Auf dem Server kann `itk:image:test` in der Shell alles grün melden, aber **PHP-FPM** hat einen anderen **PATH** oder `open_basedir` – dann schlägt der Upload trotzdem fehl. Immer beide Kontexte testen.

Windows vs. Linux: Beispielpfade in der Config sind oft Linux-lastig; auf Windows musst du **mediaPath** und Tool-Installation (oder WSL) selbst stimmig machen.

---

17 FAQ

Beeinflusst das Plugin schon hochgeladene Bilder?
Nur, wenn du `itk:image:optimize` / `itk:image:optimize-all` ausführst. Der normale Weg ist „ab jetzt bei jedem neuen Upload“.

Kann ich einen Ordner „roh“ lassen?
Ja – **Ordner ausschließen** mit kommagetrennten Namen und **%**-Mustern (siehe Tabelle oben).

Soll ich „Original sichern“ aktivieren?
Wenn du keine zweite Quelle hast und experimentierst mit aggressiver Qualität: ja. Kostet **Platte**.

Warum steht noch „mogrify“ irgendwo in Metadaten?
Historisch/Label – die Hauptkompression läuft über **jpegoptim / optipng / cwebp** plus **convert**.

Darf ich Sales-Channel-spezifische Config nutzen?
Shopware kann Plugin-Configs pro Verkaufskanal vererben – praktisch, wenn du unterschiedliche Qualität willst. Im Storefront-/Sales-Channel-Kontext arbeitet das Plugin deshalb mit der jeweiligen Kanal-Konfiguration statt blind mit nur einem globalen Satz.