staticcache.ini [Target_example]

Die Einstellungen von [Target_example] in staticcache.ini: Name, SiteAccess, SiteAccessList, DefaultSiteAccess, Kind, Remote, Branch, Folder ...

Source file: `doc/ger-de/beispiele/ini-einstellungen/staticcache/Target_example.md` Last changed 2026-11-05
https://doc.exponential.earth/learninminutes/exphelp/viewer/examples/ini-settings/staticcache/Target_example.md

Der Block `[Target_example]` von `settings/staticcache.ini`, mit 32 Einstellung(en).

Bearbeiten Sie `settings/staticcache.ini` nie selbst: Ein Update ersetzt die Datei. Schreiben Sie die Einstellungen, die Sie ändern, in ein Override, das Vorrang vor der ausgelieferten Datei hat:

- `settings/override/staticcache.ini.append.php` für jeden Siteaccess,
- `settings/siteaccess/<siteaccess>/staticcache.ini.append.php` für einen Siteaccess,
- `extension/<extension>/settings/staticcache.ini.append.php` in einer Erweiterung (gelesen, wenn die Erweiterung aktiv ist).

Name

> Ein Ziel, als Vorlage: Kopieren Sie den Block als [Target_<id>] nach settings/override/staticcache.ini.append.php und fügen Sie <id> zu TargetList[] hinzu. Dieser Block wird nicht gebaut (TargetList[] nennt ihn nicht). Einstellungen, die einem Ziel fehlen, nehmen die hier gezeigten Werte.
>
> Der Name, der in Listen und der Administration angezeigt wird.

Standard: `Example target`.

INI

```
<?php /* #?ini charset="utf-8"?
# settings/override/staticcache.ini.append.php
[Target_example]
Name=Example target
*/ ?>
```

SiteAccess

> Der zu veröffentlichende Front-Siteaccess. Administrations- und Bearbeitungs-Siteaccesses (die eine Anmeldung verlangen oder ein Administrationsdesign nutzen) werden abgelehnt.

Standard: leer.

INI

```
<?php /* #?ini charset="utf-8"?
# settings/override/staticcache.ini.append.php
[Target_example]
SiteAccess=
*/ ?>
```

SiteAccessList

> Mehrere Front-Siteaccesses in einer Kopie, statt SiteAccess: Sites, die sich über ihr URI-Präfix einen Host teilen (site.ini [SiteAccessSettings] MatchOrder mit uri, oder host_uri mit einer URI), etwa site, bold und bold_ger. Jeder außer dem Standard wird in den Ordner seines Präfixes geschrieben (/bold_ger/...); die Links zwischen ihnen bleiben in der Kopie, und Links zu einem nicht aufgeführten Siteaccess führen aus der Kopie hinaus (OutsideLinks). Nennt es welche, wird SiteAccess nicht gelesen.

Standard: eine leere Liste.

INI

```
<?php /* #?ini charset="utf-8"?
# settings/override/staticcache.ini.append.php
[Target_example]
# ein leeres "SiteAccessList[]" zuerst ersetzt die ausgelieferte Liste; ohne es werden die Einträ
ge hinzugefügt
SiteAccessList[]
*/ ?>
```

DefaultSiteAccess

> Der Siteaccess aus SiteAccessList[], der im Stamm der Kopie ausgeliefert wird. Leer: Jeder Siteaccess liegt in seinem Ordner, und der Stamm ist eine Seite, die auf jeden verlinkt. Ein Ziel mit einem Siteaccess hat ihn immer im Stamm.

Standard: leer.

INI

```
<?php /* #?ini charset="utf-8"?
# settings/override/staticcache.ini.append.php
[Target_example]
DefaultSiteAccess=
*/ ?>
```

Kind

> Wohin der Build geht: github-pages (GitHub Pages) oder git (jeder Git-Host).

Standard: `github-pages`.

INI

```ini
<?php /* #?ini charset="utf-8"?
# settings/override/staticcache.ini.append.php
[Target_example]
Kind=github-pages
*/ ?>
```

Remote

> Die Git-Adresse, in die der Build übertragen wird (ohne Passwort oder Token darin).

Standard: leer.

INI

```ini
<?php /* #?ini charset="utf-8"?
# settings/override/staticcache.ini.append.php
[Target_example]
Remote=
*/ ?>
```

Branch

> Der Branch, in den der Build übertragen wird: gh-pages, oder main mit Folder=docs.

Standard: `gh-pages`.

INI

```ini
<?php /* #?ini charset="utf-8"?
# settings/override/staticcache.ini.append.php
[Target_example]
Branch=gh-pages
*/ ?>
```

Folder

> Der Ordner des Branches, in dem die Website liegt: leer für die Wurzel, oder docs.

Standard: leer.

INI

```ini
<?php /* #?ini charset="utf-8"?
# settings/override/staticcache.ini.append.php
[Target_example]
Folder=
*/ ?>
```

Delivery

> Wie GitHub Pages den Build erhält: branch (in Branch übertragen) oder actions
> (ein GitHub-Actions-Workflow veröffentlicht ihn).

Standard: `branch`.

INI

```ini
<?php /* #?ini charset="utf-8"?
# settings/override/staticcache.ini.append.php
[Target_example]
Delivery=branch
*/ ?>
```

Links

> Wie Links geschrieben werden: relative (../about-us/, funktioniert unter jedem Pfad) oder base-path (/meine-site/about-us/, unterhalb von BasePath).

Standard: `relative`.

INI

```ini
<?php /* #?ini charset="utf-8"?
# settings/override/staticcache.ini.append.php
[Target_example]
Links=relative
*/ ?>
```

BasePath

> Der Pfad, unter dem die Website ausgeliefert wird: /<repository>/ für eine GitHub-Projekt-Website, leer oder / für eine eigene Domain oder eine Benutzer-Website. 404.html verlinkt darunter, was auch immer Links sagt, und exp:static:serve liefert den Build darunter aus.

Standard: leer.

INI

```ini
<?php /* #?ini charset="utf-8"?
# settings/override/staticcache.ini.append.php
[Target_example]
BasePath=
*/ ?>
```

PublicURL

> Die öffentliche Adresse der Website (https://owner.github.io/meine-site/), für die Sitemap, robots.txt und die Canonical- und Social-Card-Links. Leer: sie werden wie jeder andere Link geschrieben, und es wird keine Sitemap geschrieben.

Standard: leer.

INI

```
<?php /* #?ini charset="utf-8"?
# settings/override/staticcache.ini.append.php
[Target_example]
PublicURL=
*/ ?>
```

CustomDomain

> Eine eigene Domain (www.example.com): wird in CNAME geschrieben; damit wird die Website
> an der Wurzel der Domain ausgeliefert.

Standard: leer.

INI

```
<?php /* #?ini charset="utf-8"?
# settings/override/staticcache.ini.append.php
[Target_example]
CustomDomain=
*/ ?>
```

Credential

> Die Zugangsdaten, die der Push nutzt: token:<geheimnisname> (oder token:env:<VARIABLE>)
> für ein https-Remote, sshkey:<geheimnisname> für ein SSH-Remote (exp:static:keys <ziel> --generate erzeugt den Schlüssel: sshkey:staticpublish.<ziel>.deploy_key), app:<installation> (eine GitHub-App, mit der GitHub-Anbindung) oder none (der eigene SSH-Agent oder Credential-Helper des Hosts). Nur ein Name; der Wert bleibt in Setup > Geheimnisse und gelangt nur über seine Umgebung zu git.

Standard: `none`.

INI

```
<?php /* #?ini charset="utf-8"?
# settings/override/staticcache.ini.append.php
[Target_example]
Credential=none
*/ ?>
```

CrawlURL

> Die Seiten von dieser Adresse abrufen statt von der SiteURL des Siteaccess
> (eine interne Adresse derselben Website, http://127.0.0.1:8080).

Standard: leer.

INI

```
<?php /* #?ini charset="utf-8"?
# settings/override/staticcache.ini.append.php
[Target_example]
CrawlURL=
*/ ?>
```

LiveURL

> Wohin Links auf das gehen, was der Build nicht enthält (die Suche, die Anmeldung, ein Formular, ein anderer Siteaccess): die Live-Website, https://www.example.com. Leer: https:// und der Host der SiteURL des Siteaccess.

Standard: leer.

INI

```
<?php /* #?ini charset="utf-8"?
# settings/override/staticcache.ini.append.php
[Target_example]
LiveURL=
*/ ?>
```

OutsideLinks

> Links auf Seiten des eigenen Hosts der Website (oder LiveURL), die der Build nicht enthält (eine Modulansicht wie /user/login oder /shop/basket, eine Seite mit Query-String, eine ausgelassene Seite, ein anderer Siteaccess): keep (sie gehen zur Live-Website, oder bleiben ohne eine, wie sie sind) oder unlink (der Link bleibt ein mit seinen Klassen, aber ohne Adresse, mit aria-disabled markiert und von der exp-static.css des Builds ausgegraut, so dass das Aussehen des Themes bleibt). Bilder, Stylesheets, Formulare (siehe Forms) und Links auf andere Hosts werden nie herausgenommen.

Standard: `keep`.

INI

```
<?php /* #?ini charset="utf-8"?
# settings/override/staticcache.ini.append.php
[Target_example]
OutsideLinks=keep
*/ ?>
```

AdminLinks

> Links in einen Siteaccess der Administration (/admin/..., auf dem eigenen Host der Website oder LiveURL): keep (wie OutsideLinks=keep) oder unlink. Die Siteaccesses der Administration sind die, die die Prüfungen des Ziels schon kennen: eine Anmeldung ist nötig, ein Administrationsdesign oder ein Name wie admin oder editor.

Standard: `keep`.

INI

```
<?php /* #?ini charset="utf-8"?
# settings/override/staticcache.ini.append.php
[Target_example]
AdminLinks=keep
*/ ?>
```

MaxPages

> Die meisten Seiten, die ein Build durchläuft (1 bis 20000).

Standard: `10000`.

INI

```
<?php /* #?ini charset="utf-8"?
# settings/override/staticcache.ini.append.php
[Target_example]
MaxPages=10000
*/ ?>
```

MaxDepth

> Wie viele Links von der Startseite entfernt das Durchlaufen geht (0 bis 30). Leer: [CacheSettings] MaxCacheDepth.

Standard: leer.

INI

```
<?php /* #?ini charset="utf-8"?
# settings/override/staticcache.ini.append.php
[Target_example]
MaxDepth=
*/ ?>
```

Assets

> Die kopierten Dateien: referenced (was die Seiten und Stylesheets nennen) oder referenced-and-storage (zusätzlich jedes Bild aus var/<site>/storage, das zu veröffentlichten Inhalten gehört, die der anonyme Benutzer sehen darf).

Standard: `referenced-and-storage`.

INI

```
<?php /* #?ini charset="utf-8"?
# settings/override/staticcache.ini.append.php
[Target_example]
Assets=referenced-and-storage
*/ ?>
```

Extras

> Die Dateien, die neben den Seiten geschrieben werden: nojekyll (.nojekyll, damit GitHub Pages die Dateien unverändert veröffentlicht), 404 (404.html aus der eigenen 404-Seite der Site), robots (robots.txt), sitemap (sitemap.xml, braucht PublicURL), manifest (.static-publish/manifest.json; immer geschrieben), redirects (eine kleine Seite für jede alte Adresse und jeden eigenen Alias einer Seite und für URL-Platzhalter ohne *), readme (README.md), license (LICENSE, aus License unten), workflow (der Workflow von GitHub Actions). CNAME wird immer geschrieben, wenn CustomDomain gesetzt ist.

Standard: eine Liste mit 5: `nojekyll`, `404`, `robots`, `sitemap`, `manifest`.

INI

```ini
<?php /* #?ini charset="utf-8"?
# settings/override/staticcache.ini.append.php
[Target_example]
# ein leeres "Extras[]" zuerst ersetzt die ausgelieferte Liste; ohne es werden die Einträge hinzu
gefügt
Extras[]
Extras[]=nojekyll
Extras[]=404
Extras[]=robots
Extras[]=sitemap
Extras[]=manifest
*/ ?>
```

PostProcess

> Was mit jeder Seite geschieht: strip (immer) entfernt Debug-Ausgaben, Formular-Tokens, Sitzungs-IDs und Nonces der Content Security Policy. Wahlweise: minify (weniger Leerraum, keine Kommentare), precompress (.gz, und .br mit der Brotli-Erweiterung von PHP, neben jeder Textdatei), fingerprint (Stylesheets und Skripte nach dem Hash ihres Inhalts benannt, site.3f2a9c1b.css).

Standard: eine Liste mit 1: `strip`.

INI

```ini
<?php /* #?ini charset="utf-8"?
# settings/override/staticcache.ini.append.php
[Target_example]
# ein leeres "PostProcess[]" zuerst ersetzt die ausgelieferte Liste; ohne es werden die Einträge
hinzugefügt
PostProcess[]
PostProcess[]=strip
*/ ?>
```

Search

> Eine Suchseite für die statische Kopie (search.html, ein Index je Sprache und ein kleines Skript; die Suchformulare der Site gehen dorthin): enabled

> oder disabled.

Standard: `disabled`.

INI

```
<?php /* #?ini charset="utf-8"?
# settings/override/staticcache.ini.append.php
[Target_example]
Search=disabled
*/ ?>
```

Forms

> Was mit Informationssammlern (Formularen) geschieht: hide (der Ersatztext, FormFallbackText) oder endpoint (sie senden an den Formular-Endpunkt der Live-Site; braucht PublicURL oder CustomDomain). Kommentar-, Warenkorb- und Anmeldeformulare werden von einer statischen Kopie nie gesendet.

Standard: `hide`.

INI

```
<?php /* #?ini charset="utf-8"?
# settings/override/staticcache.ini.append.php
[Target_example]
Forms=hide
*/ ?>
```

ExcludePaths

> Ausgelassene Seiten, nach ihrem Pfad in der Website (/members, /shop/*); * passt auf jeden Teil. Links darauf gehen zur Live-Website.

Standard: eine leere Liste.

INI

```
<?php /* #?ini charset="utf-8"?
# settings/override/staticcache.ini.append.php
[Target_example]
# ein leeres "ExcludePaths[]" zuerst ersetzt die ausgelieferte Liste; ohne es werden die Einträge
 hinzugefügt
ExcludePaths[]
*/ ?>
```

ExcludeSections

> Seiten, die nach der Sektion ihres Inhalts ausgelassen werden (Sektions-IDs).

Standard: eine leere Liste.

INI

```
<?php /* #?ini charset="utf-8"?
# settings/override/staticcache.ini.append.php
[Target_example]
# ein leeres "ExcludeSections[]" zuerst ersetzt die ausgelieferte Liste; ohne es werden die Eintr
äge hinzugefügt
ExcludeSections[]
*/ ?>
```

Triggers

> Was einen Build auslösen darf: manual, publish (der Cronjob-Teil staticpublish veröffentlicht die geänderten Seiten nach einer Veröffentlichung von Inhalten), schedule (derselbe Cronjob-Teil veröffentlicht vollständig zu den Zeiten von Schedule), cli.

Standard: eine Liste mit 2: `manual`, `cli`.

INI

```
<?php /* #?ini charset="utf-8"?
# settings/override/staticcache.ini.append.php
[Target_example]
# ein leeres "Triggers[]" zuerst ersetzt die ausgelieferte Liste; ohne es werden die Einträge hin
zugefügt
Triggers[]
Triggers[]=manual
Triggers[]=cli
*/ ?>
```

Schedule

> Wann ein geplanter Build läuft (ein Cron-Ausdruck, in der Zeitzone des Servers: 0 3 * * *, oder @daily), für Triggers[]=schedule.

Standard: leer.

INI

```
<?php /* #?ini charset="utf-8"?
# settings/override/staticcache.ini.append.php
[Target_example]
Schedule=
*/ ?>
```

Notify

> Wann eine Benachrichtigung per E-Mail gesendet wird ([StaticPublishSettings] NotifyEmail): failure, success.

Standard: eine Liste mit 1: `failure`.

INI

```
<?php /* #?ini charset="utf-8"?
# settings/override/staticcache.ini.append.php
[Target_example]
# ein leeres "Notify[]" zuerst ersetzt die ausgelieferte Liste; ohne es werden die Einträge hinzu
gefügt
Notify[]
Notify[]=failure
*/ ?>
```

Webhook

> Eine https-Adresse, an die nach jeder Veröffentlichung und Rücknahme eine Benachrichtigung geht (POST, JSON, Schema exp-static-webhook/1).

Standard: leer.

INI

```
<?php /* #?ini charset="utf-8"?
# settings/override/staticcache.ini.append.php
[Target_example]
Webhook=
*/ ?>
```

License

> Die Lizenz der Inhalte für Extras[]=license: eine SPDX-Kennung (CC-BY-4.0) oder der Text selbst.

Standard: leer.

INI

```
<?php /* #?ini charset="utf-8"?
# settings/override/staticcache.ini.append.php
[Target_example]
License=
*/ ?>
```

Siehe auch

- Alle Blöcke von staticcache.ini
- Alle INI-Dateien
- Die Einstellungsseite

Erzeugt von `./console exp:help:examples --language=ger-de` aus `settings/staticcache.ini`. Bearbeiten Sie diese Seite nicht: Ändern Sie die Quelle oder die deutsche Übersetzung und erzeugen Sie sie neu (wie). Freie Dokumentation: GFDL 1.3 oder später.

Exponential Copyright © 1998-2026 Exponential Foundation and 7x
Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.3 or any later version published by the Free Software Foundation; with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts. A copy of the license is included in the section entitled "GNU Free Documentation License".
This documentation is free documentation, licensed under the GNU Free Documentation License 1.3 or later:
https://www.gnu.org/licenses/fdl-1.3.html.