OPcache und Cache-Vorladen

Zwei verschiedene Arten von „Vorladen“, denen ein PHP-Entwickler in Exponential begegnet - der Opcode-Cache (OPcache) und die Einstellungen, die unter PHP-FPM und Velocity zählen, sowie der Vorlader für den Site-Cache…

Source file: `doc/ger-de/hilfe/php-entwickler/opcache-und-cache-vorladen.md` Last changed 2026-11-04
https://doc.exponential.earth/learninminutes/exphelp/viewer/help/php-developers/opcache-and-cache-preloading.md

Diese Seite richtet sich an PHP-Entwickler, die Code bereitstellen oder sich um die erste Anfrage nach einem Deployment kümmern. Zwei Dinge heißen rund um Exponential „Vorladen“, und sie sind verschieden:

- **OPcache** hält kompiliertes PHP im Speicher, sodass Dateien nicht bei jeder Anfrage erneut geparst werden.
- **Der Vorlader für den Site-Cache** (`bin/php/preload.php`, `exp:preload`) fragt die Seiten einer Site ab, bevor Besucher es tun, sodass der View-Cache der Seiten, die Bild-Aliase und die kompilierten Templates existieren.

Die eigene PHP-Einstellung `opcache.preload` wird von Exponential nicht verwendet; nichts in den mitgelieferten Einstellungen konfiguriert sie.

Inhalt

1. OPcache unter PHP-FPM
2. OPcache unter Velocity
3. Was eine Codeänderung braucht
4. Der Vorlader für den Site-Cache
5. Der Cache-Wärmer
6. Nach einem Deployment

OPcache unter PHP-FPM

Verwenden Sie die Standardwerte von PHP oder die Einstellungen Ihres Hosters; Exponential braucht nichts Besonderes. Zwei Punkte:

- Mit `opcache.validate_timestamps=0` (in der Produktion üblich) werden geänderte PHP-Dateien erst gesehen, wenn PHP-FPM neu geladen wird. Laden Sie nach jedem Deployment von PHP-Code den Pool neu, der die Site tatsächlich bedient.
- Generierte PHP-Dateien (der INI-Cache, kompilierte Templates, die Klassen-Maps in `var/autoload/`) werden zur Laufzeit geschrieben; OPcache nimmt sie wie jede andere Datei auf.

OPcache unter Velocity

Velocity führt die Kommandozeilen-Binärdatei von PHP aus, in der OPcache ausgeschaltet ist, sofern es nicht angefordert wird. `settings/velocity.ini` `[PHPSettings]` `IniOptions[]` liefert deshalb mit:

Option	Warum
`opcache.enable_cli=1`	OPcache für den Serverprozess einschalten; gemessen 18 bis 26 Prozent schneller
`opcache.memory_consumption=256`, `opcache.max_accelerated_files=20000`, `opcache.interned_strings_buffer=32`	Platz für den Kernel, die Erweiterungen und die kompilierten Templates
`opcache.revalidate_freq=0`	Zeitstempel bei jedem Einbinden prüfen
`opcache.file_update_protection=0`	die Anfragezeit eines persistenten Workers ist seine Startzeit, sodass jede später geschriebene Datei (INI-Caches, kompilierte Templates) als „zu neu“ galt und bei jedem Einbinden kompiliert wurde; mit `0` kosten gerenderte Seiten etwa ein Viertel der CPU wie vorher
`apc.shm_size=256M`	Platz für den Antwort-Cache und den SQL-Abfrage-Cache in APCu

Lesen Sie die vollständige Liste in `settings/velocity.ini` und den Hintergrund unter Velocity: Exponential in einem Webserver mit persistenten Workern betreiben.

Was eine Codeänderung braucht

Änderung	Unter PHP-FPM	Unter Velocity
eine PHP-Datei bearbeitet	PHP-FPM neu laden, wenn Zeitstempel nicht validiert werden	`exp:velocity deploy`: Worker behalten die Klassen, die sie geladen haben
eine Klasse hinzugefügt oder umbenannt	Autoloads, dann neu laden	`exp:velocity deploy` (`--kernel` für eine Kernel-Klasse)
eine Kernel-Datei bei eingeschaltetem Engine-Archiv	(nicht zutreffend)	das Archiv wird beim Neustart bei Bedarf neu gebaut (Das Engine-Archiv)
ein Template bearbeitet	nichts, oder Template-Caches leeren	`exp:velocity cache clear` für den Antwort-Cache

Der Vorlader für den Site-Cache

Shell

```
php bin/php/preload.php --apply                                    # the default siteaccess
php bin/php/console exp:preload --apply --siteaccess=<siteaccess>
```

Ein Lauf beginnt bei den Abschnittsseiten einer Site, folgt ihren Links bis zu einer Seitengrenze und einer Linktiefe und fragt sie als anonymer Besucher ab. Er wärmt den View-Cache der Seiten, die Bild-Aliase und die kompilierten Templates vor, also Dateien, die PHP-FPM und Velocity gemeinsam nutzen. Jeweils ein Lauf hält eine Sperre; jeder Lauf wird mit seinen defekten Links festgehalten. **Setup > Seiten vorladen** startet denselben Lauf im Hintergrund

und zeigt ihn an (Caches vorladen, Vorlader für den Site-Cache).

Der Cache-Wärmer

`exp:warm` (`bin/php/warm.php`) durchläuft die veröffentlichten Inhalte als anonymer Besucher, sodass niemand für das erste Rendern nach einem Ablauf bezahlt. Er ergänzt den Antwort-Cache von Velocity und den rollenbewussten HTTP-Cache (HTTP/2 und Cache-Vorwärmen).

Nach einem Deployment

1. Generieren Sie die Autoloads neu und leeren Sie die Caches, die die Änderung braucht (oder `exp:velocity deploy`, das es in der richtigen Reihenfolge erledigt).
2. Laden Sie PHP-FPM neu oder starten Sie die Worker von Velocity neu.
3. Leeren Sie die Caches, die gerenderte Ausgabe enthalten, **nach** dem Neuladen, damit eine zwischendurch von altem Code gerenderte Seite nicht behalten wird.
4. Vorwärmen: `exp:preload` oder `exp:warm`.

Verwandte Seiten

- Velocity und der Opcode-Cache
- Leitfaden zum Deployment
- Leitfaden zu Caches
- Freie Dokumentation: diese Seite darf unter der GNU FDL 1.3 frei kopiert, übersetzt und verbessert werden.

> **Werden Sie Exponential-Partner**
>
> Partner teilen ihre Informationen, stellen Mittel bereit, geben Beiträge zurück und entwickeln die Produkte mit uns, und so wächst Exponential. Eine Partner-Website und ein Dashboard sind für Mitte 2027 geplant: Werden Sie noch heute Partner, damit wir schneller wachsen und neue Funktionen früher liefern.
>
> Partner: hier beginnen | Wie Sie sich bewerben

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.