Das Engine-Archiv (phar) für PHP-Entwickler

Was das optionale Engine-Archiv dist/engine.phar für PHP-Code bedeutet - welche Verzeichnisse es enthält, wie es eingeschaltet wird, warum Code das Wurzelverzeichnis der Installation niemals aus __DIR__ berechnen darf…

Source file: `doc/ger-de/hilfe/php-entwickler/das-engine-phar-fuer-php-entwickler.md` Last changed 2026-11-04
https://doc.exponential.earth/learninminutes/exphelp/viewer/help/php-developers/the-engine-phar-for-php-developers.md

Diese Seite richtet sich an PHP-Entwickler, deren Code möglicherweise mit eingeschaltetem Engine-Archiv läuft. Exponential kann `kernel/`, `lib/` und `autoload/` aus einem Archiv laden, `dist/engine.phar`, statt aus tausend einzelnen Dateien. Es ist standardmäßig aus, es ist keine Performance-Funktion (gemessen macht es keinen Unterschied), und es ist kein Quellcodeschutz. Aber es ändert eine Sache, die PHP-Code beachten muss: Pfade. Die vollständige Referenz ist Unterstützung für das Engine-Archiv (phar).

Inhalt

1. Was im Archiv steckt
2. Die Regel für Ihren Code: Leiten Sie das Wurzelverzeichnis nie aus einer Kernel-Datei ab
3. Ein- und ausschalten
4. Bauen und neu bauen
5. Der Stream-Wrapper phar
6. Es in einem Stacktrace erkennen

Was im Archiv steckt

Aus dem Archiv	Immer von der Festplatte
`kernel/`, `lib/`, `autoload/`	`extension/`, `design/`, `settings/`, `share/`, `bin/`, `var/`, `vendor/`

Nur der Autoloader verhält sich anders: Eine Kernel- oder Bibliotheksklasse wird über `phar://.../engine.phar/...` gelesen. Die Klassen Ihrer Erweiterung werden immer von der Festplatte gelesen.

Die Regel für Ihren Code: Leiten Sie das Wurzelverzeichnis nie aus einer Kernel-Datei ab

Innerhalb des Archivs nennen `__DIR__` und `__FILE__` einer Kernel-Datei einen Pfad **innerhalb des Archivs**, sodass Code, der das Wurzelverzeichnis der Installation als `dirname( __DIR__, 3 )` berechnet, nichts findet. `autoload.php`, das immer auf der Festplatte liegt, veröffentlicht das Wurzelverzeichnis einmal:

PHP

```php
// the installation root, an absolute path, whatever the engine
$root = defined( 'EXP_ROOT_DIR' ) ? EXP_ROOT_DIR : eZSys::rootDir();
$file = $root . '/var/acmereport/export.csv';
```

- Verwenden Sie `EXP_ROOT_DIR` (von `autoload.php` definiert) oder `eZSys::rootDir()`.
- Verwenden Sie `eZSys::varDirectory()`, `eZSys::cacheDirectory()` und `eZSys::storageDirectory()` für die Verzeichnisse der Site.
- Verwenden Sie für Einstellungen und Designs Pfade relativ zum Wurzelverzeichnis der Installation, wie der Kernel; das aktuelle Verzeichnis jedes Einstiegspunkts ist das Wurzelverzeichnis.

Die eigenen Dateien Ihrer Erweiterung liegen auf der Festplatte, sodass `__DIR__` in einer Erweiterungsklasse weiterhin funktioniert. Die Regel ist wichtig, wenn Sie Kernel- oder Bibliothekscode ändern, und für Hilfsfunktionen, die später in den Kernel wandern könnten.

Ein- und ausschalten

INI

```ini
# settings/velocity.ini, or an override
[ServerSettings]
EnginePhar=enabled          ; disabled (default), enabled, or a path to an archive
```

oder die Umgebungsvariable `EXP_ENGINE_PHAR=/path/to/dist/engine.phar` für das, was die Site ausliefert. Velocity exportiert die Variable an seine Worker. Ein fehlendes Archiv hindert den Server am Starten, statt still auf die Dateien zurückzufallen. Leeren Sie die Caches, wenn Sie umschalten, in beide Richtungen.

Bauen und neu bauen

Shell

```shell
php bin/php/console exp:phar build      # write dist/engine.phar (only when something changed)
php bin/php/console exp:phar check      # current or not, which files differ; exit 1 when stale
php bin/php/console exp:phar info       # path, size, version, whether it is in use
php bin/php/console exp:phar clean      # remove it
```

- `dist/engine.phar.index.json` erfasst jede gepackte Datei mit Größe, Zeit und SHA-1. Ein Build und jedes `exp:velocity start`, `restart` und `graceful` baut nur dann neu, wenn eine Datei in `kernel/`, `lib/` oder `autoload/` hinzugefügt, entfernt oder geändert wurde oder sich die Minor-Version von PHP geändert hat.
- **Eine Datei, die sich nicht parsen lässt, behält das alte Archiv.** Jede PHP-Datei wird zuerst geparst; eine fehlerhafte wird genannt, und nichts wird geschrieben. Ein Neustart lehnt dann ab, bevor er den Server stoppt. Führen Sie `php -l` auf dem aus, was Sie geändert haben.
- Das Archiv ist mit dem Release, dem Commit und `-dirty` für einen nicht committeten Baum gestempelt; `exp:phar info` und **Setup > Systeminformationen** (Block *Engine*) zeigen es an.

Bei eingeschaltetem Archiv ist **eine Kernel-Änderung also erst nach einem Neubau live**, den `exp:velocity deploy` und ein Neustart für Sie erledigen.

Der Stream-Wrapper phar

`autoload.php` hebt bei jeder Anfrage die Registrierung des Stream-Wrappers `phar://` von PHP auf, sofern die Laufzeitumgebung nicht selbst darüber gelesen wird (das Engine-Archiv oder ein Werkzeug wie `phpunit.phar` als Einstiegspunkt). Der Grund ist Sicherheit: Eine Datei, die sowohl ein gültiges Bild als auch ein gültiges phar ist, kann hochgeladen werden, und mit registriertem Wrapper würde `include( 'phar://<upload>.jpg/x.php' )` das PHP darin ausführen. Registrieren Sie den Wrapper nicht erneut, und verwenden Sie keine `phar://`-Pfade in Ihrem eigenen Code.

Es in einem Stacktrace erkennen

Ein Stacktrace oder eine Fehlermeldung mit `phar://.../dist/engine.phar/kernel/...` bedeutet, dass das Archiv verwendet wird. Wenn eine Änderung an einer Kernel-Datei „nichts bewirkt“, prüfen Sie zuerst `exp:phar check` und **Setup > Systeminformationen**.

Verwandte Seiten

- Unterstützung für das Engine-Archiv (phar)
- Persistente Velocity-Worker für PHP-Entwickler
- Velocity-Engines
- 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.