



Nils Adermann
@naderman



Private Packagist
<https://packagist.com>

Composer 2.0 wurde am 24.10.2020 veröffentlicht

- Entwicklung seit Sommer 2018
- 28 Code Contributor
- Viele weitere Helfer, die Issues reportet oder kommentiert, oder pre-releases getestet haben
- Finanziell durch Private Packagist Kunden möglich gemacht worden

Vielen Dank!



Didier Sampaolo
@dsampaolo



Replying to @seldaek

My internet connection is really slow, and Composer 2 is like an early Christmas gift. Thank you for your work!



12:47 AM · Nov 14, 2020 · Twitter for Android

Ziele für 2.0

- Performance verbessern
- Besser reproduzierbares Verhalten
 - Die übrigen Bugs in 1.x sind Sonderfälle, die schwer zu debuggen und schwer zu reproduzieren
- Bessere Fehlermeldungen
- Neue Funktionen, die durch BC Breaks / Refactorings einfacher umsetzen geworden sind
- Schmerzfreie Upgrades von 1.x auf 2.0

Warum 2.0 und nicht 1.x?

Performance verbessern

- Was macht Composer langsam?
 - I/O
 - Netzwerk
 - Metadaten JSON Downloads
 - Paketarchiv Downloads
 - Speicherzugriff
 - Schreiben, Lesen und verändern von GB von Speicher
 - CPU
 - Sequentielles extrahieren der Code Archive

Performance verbessern

Wofür benutzt Composer Speicher?

- JSON Repräsentation jeder Version jedes Pakets dass Requirements erfüllen könnte
- Repräsentation von Abhängigkeiten/Konflikten zwischen all diesen Paketversionen für den SAT Solver

Lösungen

- Anzahl der Paketversionen reduzieren, die in Frage kommen
- Abhängigkeiten/Konflikte effizienter repräsentieren

Anzahl der Paketversionen reduzieren, die in Frage kommen

- **Composer 1** Lazy Loading: Beim Erstellen der Speicherrepräsentation von Abhängigkeiten
 - Idee: Solver lädt nur was er wirklich braucht wenn er soweit ist
 - Probleme
 - Solver wartet trotzdem, nur zu späterem Zeitpunkt
 - Unmöglich die Menge der Pakete zu reduzieren bevor alle Abhängigkeiten/Konflikte geladen wurden
 - Paralleler Netzwerkzugriff schwer/chaotisch umsetzbar

Composer 2.0 refactort Prozess in mehrere klar getrennte Schritte:

- Rekursiver Metadaten Download für Paketversionen, die wirklich gebraucht werden
- Reduktion der Paketversionsanzahl im Speicher so weit wie möglich
- Speicherrepräsentation von Abhängigkeiten erstellen

=> BC Break (für Plugins) => 2.0

Abhängigkeiten/Konflikte effizienter repräsentieren

SAT Solver braucht einen booleschen Ausdruck, z.B.

```
foo/bar 1.0 requires baz/qux ^2.0
foo/bar 1.0 conflicts with baz/qux ^2.0
```

```
(- foo/bar 1.0 | baz/qux 2.0.0 | baz/qux 2.0.1 | baz/qux 2.1.0)
(- foo/bar 1.0 |- baz/qux 2.0.0) & (- foo/bar 1.0 |- baz/qux 2.0.1) &
(- foo/bar 1.0 |- baz/qux 2.1.0)
```

Man kann nur eine Version eines Pakets installieren
=> impliziert einen Konflikt zwischen jedem Versionspaar

```
foo/bar 1.0, 1.1, 1.2
```

```
(- foo/bar 1.0 |- foo/bar 1.1) & (- foo/bar 1.0 |- foo/bar 1.2) &
(- foo/bar 1.1 |- foo/bar 1.2)
```

Extremes Wachstum $\binom{n}{2} = \frac{n!}{2(n-2)!}$

	3 versions	6 versions	100 versions	Symfony 500 versions	1000 versions
Composer 1	3 rules	15 rules	4,950 rules	124,750 rules	499,500 rules
Composer 2	1 rule	1 rule	1 rule	1 rule	1 rule

Composer 2.0 benutzt eine einzige besondere Multi Conflict Rule Darstellung für alle diese Rules

```
foo/bar 1.0, 1.1, 1.2
```

```
oneof(foo/bar 1.0, foo/bar 1.1, foo/bar 1.2)
```

- JSON Metadaten & Paketarchiv Downloads
 - **Parallelisierung** von HTTP Requests mit curl multi
 - Benutzung von **HTTP/2** Features für weniger Server Round-Trips
 - Stabiler und vollständiger als Composer 1 Plugin Lösungen (hirak/prestissimo, symfony/flex), die durch das Plugin Interface erschwert wurden
- Packagist.org Protokoll Verbesserungen
 - Übertragene Datenmenge reduziert
 - Effektiverer Cache durch das Trennen von dev Versionen und Releases
 - Verbesserte Stabilität für packagist.org Infrastruktur

Hinweis: Verbesserungen benötigen ext-curl (bei ca. 97% der Composer Nutzer installiert)

Performance verbessern: Archive Extrahieren

- Composer 2.0 entpackt alle heruntergeladenen Archive parallel
 - Benötigt Linux/OS X/WSL
 - Benötigt “unzip” Programm im \$PATH

Performance verbessern

- Was macht Composer langsam?
 - I/O ✓
 - Netzwerk ✓
 - Metadaten JSON Downloads ✓
 - Paketarchiv Downloads ✓
 - Speicherzugriff ✓
 - Schreiben, Lesen und verändern von GB von Speicher ✓
 - Anzahl der Paketversionen reduzieren, die in Frage kommen ✓
 - Abhängigkeiten/Konflikte effizienter repräsentieren ✓
 - CPU ✓
 - Sequentielles extrahieren der Code Archive ✓

- <https://blog.packagist.com/composer-2-0-is-now-available/> von Jordi - @seldaek
Updates: **60% weniger Zeit**
- <https://susi.dev/composer2-perf> von Susey - @sasunegomo
Installs: 44% weniger Zeit
Updates: **80% weniger Zeit** **81% weniger Speicher (1.15GB zu 244MB)**
- <https://developers.ibexa.co/blog/benchmarks-of-composer-2.0-vs-1.10> von Jani Tarvainen - @velmu
Installs: 27% weniger Zeit
Updates: 50% weniger Zeit 72% weniger Speicher (2.5GB zu 711MB)

Partial Update (composer require):
23% weniger Zeit **88% weniger Speicher (648MB zu 81MB)**

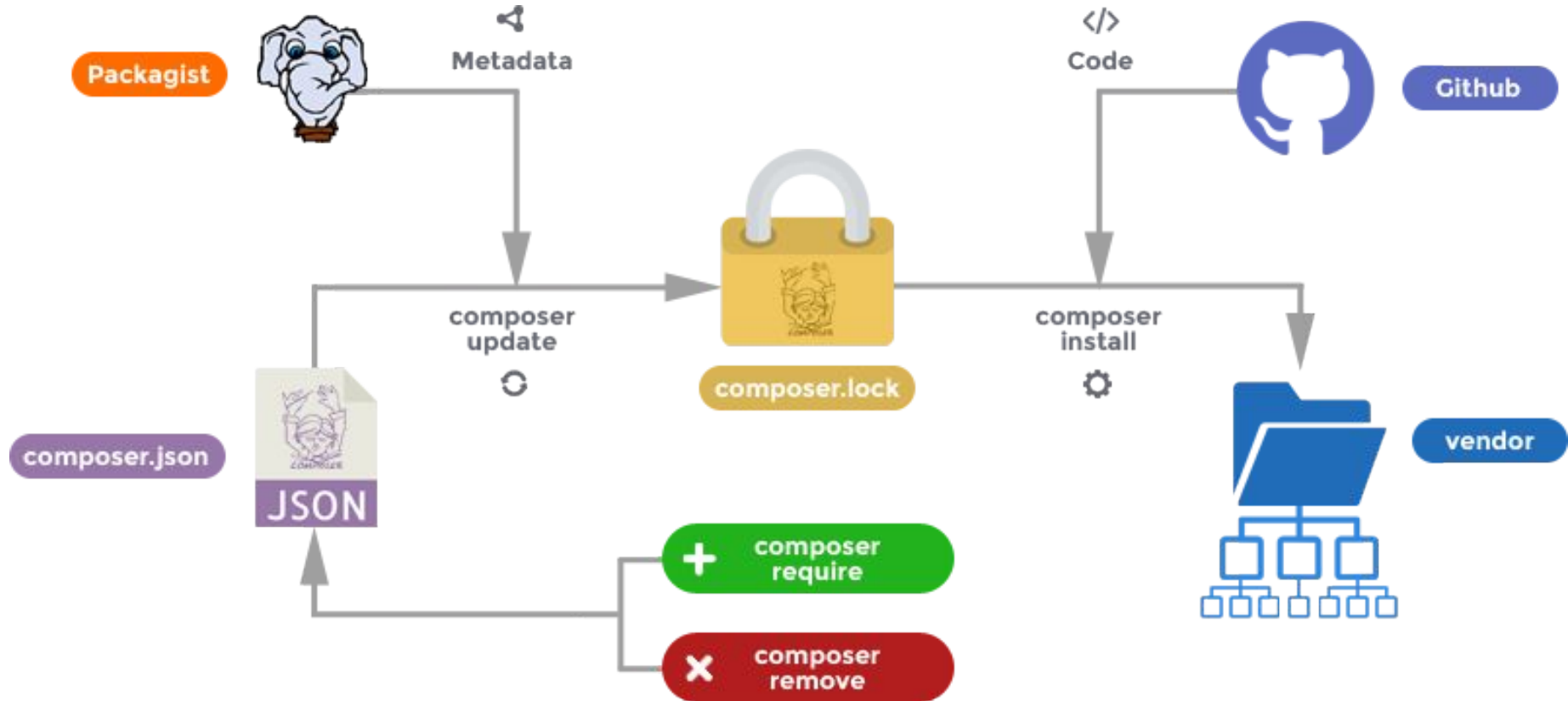
Bessere Reproduzierbarkeit

composer update

VS

composer install

Trennung von update & install



Separating update & install

vendor		
symfony/http-foundation:	5.1.8	previous local upgrade attempt
composer.lock		
symfony/http-foundation:	4.4.16	old production state
composer.json		
symfony/http-foundation:	5.0.*	limited upgrade for now, because of 5.1 issues

```
naderman@saumur: ~/projects/composer/test/symfony-http-foundation$ composer update
```

```
Loading composer repositories with package information
```

```
Updating dependencies
```

```
Lock file operations: 0 installs, 1 update, 0 removals
```

```
- Upgrading symfony/http-foundation (v4.4.16 => v5.0.11)
```

```
Writing lock file
```

```
Installing dependencies from lock file (including require-dev)
```

```
Package operations: 3 installs, 1 update, 1 removal
```

```
- Removing symfony/deprecation-contracts (v2.1.3)
```

```
- Installing symfony/polyfill-php72 (v1.17.0): Extracting archive
```

```
- Installing symfony/polyfill-intl-idn (v1.17.1): Extracting archive
```

```
- Installing symfony/mime (v5.1.8): Extracting archive
```

```
- Downgrading symfony/http-foundation (v5.1.8 => v5.0.11): Extracting archive
```

```
Generating autoload files
```

```
6 packages you are using are looking for funding.
```

```
Use the `composer fund` command to find out more!
```

New Features

Projekt auf PHP8 testen?

```
composer update --ignore-platform-reqs
```

Installiert auf PHP8

Kann Pakete installieren, die PHP Extensions brauchen, die nicht da sind

```
composer update --ignore-platform-req=php
```

Installiert auf PHP8

Überprüft alle PHP Extensions wie üblich

Beim Deployment immer ausführen:

```
composer check-platform-reqs
```

Partial Updates zu bestimmten Versionen

```
// composer.json
"require": {
    "symfony/http-foundation": "^4.0 || ^5.0",
}
```

```
// composer.lock
"packages": [{
    "name": "symfony/http-foundation",
    "version": "v4.4.16",
```

```
$ composer update symfony/http-foundation:5.0.*
```

```
Loading composer repositories with package information
```

```
Updating dependencies
```

```
Lock file operations: 0 installs, 1 update, 0 removals
```

```
- Upgrading symfony/http-foundation (v4.4.16 => v5.0.11)
```

```
Writing lock file
```

```
Installing dependencies from lock file (including require-dev)
```

```
Package operations: 0 installs, 1 update, 0 removals
```

```
- Downloading symfony/http-foundation (v5.0.11)
```

```
- Upgrading symfony/http-foundation (v4.4.16 => v5.0.11): Extracting archive
```

- Repositories sind canonical by default
 - Erstes Repository, das ein Paket mit dem gesuchten Namen enthält, gewinnt
 - Wiederherstellen des alten Merge Verhaltens mit
“canonical”: false
- Begrenzen der Pakete aus einem Repository

```
{  
  "type": "composer",  
  "url": "https://some-third-party.com/composer-repo/",  
  "only": [ "foo/*", "bar/baz" ],  
  "exclude": [ "foo/qux" ]  
}
```

Projekte auf 2.0 umstellen

Best Case

```
composer self-update --2  
composer update / composer install
```

Keine Fehler, alles läuft wie vorher.

Upgrade Probleme mit Plugins

`foo/bar requires composer-plugin-api ^1.0.0 -> no matching package found.`

- foo/bar aktualisieren, wenn es eine Version mit Composer 2.0 Support gibt
- Autor des foo/bar Plugins kontaktieren
- Vorübergehend zum Testen das Plugin entfernen

Fast alle bekannten Plugins sind mittlerweile kompatibel.

- Kompatibel mit Composer 2.0 seit 1.8.0
 - requires PHP ^7.4
- composer require **composer/package-versions-deprecated**
 - Wir haben das Paket geforkt, jetzt mit PHP ^7.0 kompatibel
 - replaces ocramius/package-versions
 - => erfüllt auch alle requires von ocramius/package-versions in Abhängigkeiten
- Für neuen Code der Runtime Paketinformationen braucht, lieber:

Runtime Composer Utilities <https://github.com/composer/composer/blob/master/doc/07-runtime.md>

- automatisch in jedem Composer Projekt verfügbar

```
\Composer\InstalledVersions::isInstalled('vendor/package'); // returns bool
```

```
use Composer\Semver\VersionParser;  
\Composer\InstalledVersions::satisfies(new VersionParser, 'vendor/package', '2.0.*');
```

Autoloading Probleme

Deprecation Warnings in Composer1 überprüfen

Class Foo\Bar located in ./src/SomeName/Bar.php does not comply with psr-4 autoloading standard.

It will not autoload anymore in Composer v2.0. in

phar:///usr/local/bin/composer/src/Composer/Autoload/ClassMapGenerator.php:18

Verzeichnisnamen müssen zu Klassennamen passen wie in PSR-0/4 definiert.

Wie weit ist Composer 2.0?

Schnelle Akzeptanz

- Composer 2.0 auf packagist.org
 - **38%** composer install letzte Woche
 - **30%** composer update letzte Woche

Stabil

- Aktuelle Version 2.0.8
- Offene Issues größtenteils seltene Sonderfälle
- Eine große Entwicklerzahl setzt die neue Version bereits produktiv ein

Jetzt zu Composer 2 upgraden!

```
composer self-update --2
```

```
Updating to version 2.0.x (2.x channel).
```

```
  Downloading (100%)
```

```
Use composer self-update --rollback to return to version 1.10.17
```

Lock Datei commitet? Keinerlei Risiko beim lokalen ausprobieren!

```
Zurück zu v1? composer self-update --1
```

Feedback



Gary Hockin
@GeeH

Replying to @seldaek

Huge congrats to you @naderman and contributors. Thanks for all the time and energy you spend making our lives easier.

1:50 PM · Oct 24, 2020 · Tweetbot for Mac



Alexandre ELISÉ (Coder.Parler.Partager)
@mralexandreise

composer v2 is insanely fast!!! Thanks to @seldaek and others for this speedup. #PHP

4:49 PM · Oct 25, 2020 · Twitter Web App



Cees-Jan Kiewiet
@WyrilHaximus

22s is my LONGEST composer v2 run so far, well done team @packagist @seldaek @naderman!

```
- Installing php-coveralls/php-coveralls (v2.4.2): Extracting archive
- Installing nunomaduro/collision (v5.0.2): Extracting archive
- Installing maglnet/composer-require-checker (2.1.0): Extracting archive
- Installing wyrilhaximus/test-utilities (2.9.0): Extracting archive
- Installing ctuw/block-race (v1.4.0): Extracting archive
- Installing wyrilhaximus/async-test-utilities (2.3.0): Extracting archive
- Installing wyrilhaximus/ticking-promise (1.0.3): Extracting archive
3 package suggestions were added by new dependencies, use 'composer suggest' to see details.
```



Austen Cameron
@austencam

The rumors are true folks. Composer 2 is *so* fast. 🐎

Incredible job @seldaek 🙌🙌

11:46 PM · Oct 26, 2020 · Twitter Web App



Tizian Schmidlin
@TizianTweets

Kudos to @seldaek and all people who worked on #composer v2, deployment in CI is at least twice as fast with composer install and up to three times as fast with composer update! I'm very impressed!

9:57 AM · Nov 10, 2020 · Twitter for Android



Dries Vints
@driesvints

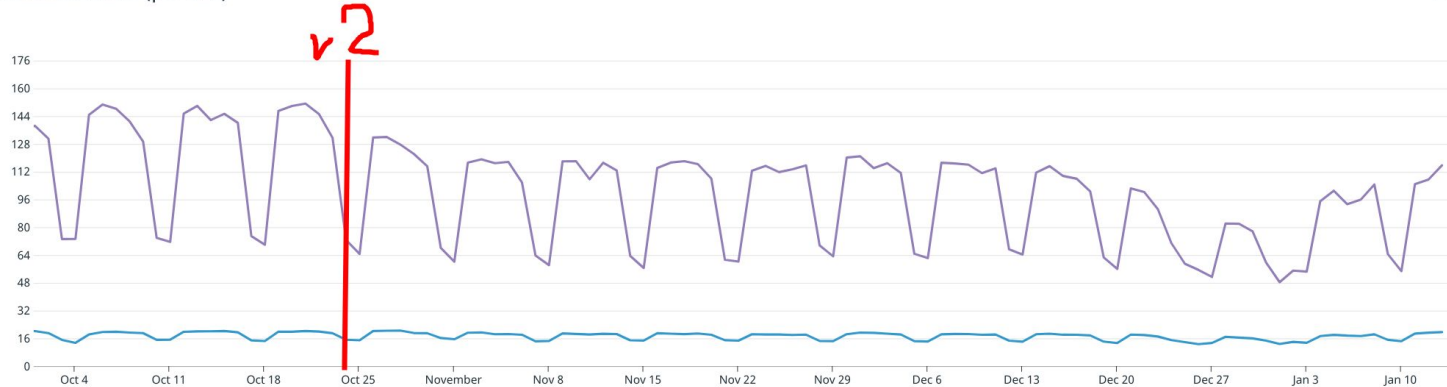
Composer v2 is 🚀👏

10:50 AM · Oct 26, 2020 · Tweetbot for Mac

packagist.org Bandbreite

- Composer 2 hat die Bandbreite von 74 TB/week auf 53 TB/week reduziert
 - Jeder sollte jetzt upraden
 - Composer 2 unterstützt sogar PHP 5.3

Network traffic (per sec)



Wie geht es weiter?

- Minimum PHP 7.1 ab Composer 2.2
 - Keine PHP5 Unterstützung mehr, aber nur internes Refactoring
 - Composer 2.1 wird noch längere Zeit Bug & Security Fixes erhalten
- Abschalten der Composer 1 Schnittstelle auf packagist.org
 - Zeitpunkt hängt von Nutzung ab, bitte upgraden!

Ressourcen

- Packagist Blog: Composer 2.0 is now available!
<https://getcomposer.org/2>
- Changelog
<https://github.com/composer/composer/releases/tag/2.0.0>
- Upgrade Guide
<https://github.com/composer/composer/blob/master/UPGRADE-2.0.md>
- Composer Plugin Readiness for 2.0
<https://github.com/composer/composer/issues/8726>

Questions / Feedback?



Private Packagist
<https://packagist.com>

E-Mail: contact@packagist.com

Twitter: [@naderman](https://twitter.com/naderman)