

PSR-6 and PSR-16 in Netgen Stack

By BM and MB



Netgen.

The need for caching PSR's



Caching PSR's

- Caching is a common way to improve the performance of any project
- Many libraries roll their own caching libraries
- Developers have to learn multiple systems
- The problem of supporting multiple frameworks (TediumStashBundle)

The future looks cached and standardized

- A common interface for caching systems will solve our problems
- Developers can count on the systems working the way they're expecting
- No more a whole assortment of adapters

PSR-6



PSR-6

Goal:

- **Allow developers to create cache-aware libraries that can be integrated into existing frameworks**

PSR-6

Interfaces:

- **CacheItemInterface**
- **CacheItemPoolInterface**
- **CacheException**
- **InvalidArgumentException**

PSR-6 - CacheltemInterface

- *public function getKey()*
- *public function get()*
- *public function isHit()*
- *public function set(\$value)*
- *public function expiresAt(\$expiration)*
- *public function expiresAfter(\$time)*

PSR-6 - CacheItemPoolInterface

- *public function getItem(\$key)*
- *public function getItems(array \$keys = array())*
- *public function hasItem(\$key)*
- *public function clear()*

PSR-6 - CacheItemPoolInterface

- *public function deleteItem(\$key)*
- *public function deleteItems(array \$keys)*
- *public function save(CacheItemInterface \$item)*
- *public function saveDeferred(CacheItemInterface \$item)*
- *public function commit()*

PSR-6 - Libraries

- **Symfony Cache**
- **PHP Cache**
- **Laminas Cache (Zend Cache)**
- **Stash Cache**

PSR-16



PSR-16 (simple cache)

Goals:

- a simple interface for cache operations
- support for operations on multiple keys
- Providing an adapter that turns a PSR-6 into a PSR-Simple-Cache

PSR-16 (simple cache)

Non-Goals:

- Solving all possible edge cases, PSR-6 does this well already.

PSR-16 (simple cache)

Why bother?

- **standardized layer of simplicity on top of PSR-6**
- **barebones by design**
- **to be used only by the most simple cases**
- **layer of convenience on top of PSR-6**

PSR-16 - Libraries

- **Symfony Cache**
- **PHP Cache**
- **Laminas Cache (Zend Cache)**

PSR-(6|16) in Symfony



Symfony Cache Component

- **PSR-6**
- **Cache Contracts**

Symfony Cache - PSR-6

Item:

A single unit of information stored as key/value pair, where the key is the unique identifier of the information and the value is its contents.

Symfony Cache - PSR-6

Pool:

A logical repository of cache items. All cache operations are performed through the pool. Applications can define as many pools as needed.

Symfony Cache - PSR-6

Adapter:

It implements the actual caching mechanism to store the information in the filesystem, in a database, etc.

Symfony Cache - available adapters

The following cache adapters are available:

APCu, Array, Chain, Couchbase, Doctrine, Filesystem, Memcached, PDO & Doctrine DBAL, PHP Array, PHP Files, Proxy, Redis

Symfony Cache - sugar on top

- **AdapterInterface**
- **TagAwareAdapterInterface**
- **PruneableInterface**

PSR-(6|16) in Netgen Stack



PSR-6 demo



PSR-16 demo



Links

- <https://github.com/MarioBlazek/caching-psrs-in-netgen-stack>
- <https://www.php-fig.org/psr/psr-6/>
- <https://www.php-fig.org/psr/psr-16/>
- <https://symfony.com/doc/current/components/cache.html>
- https://symfony.com/doc/current/components/cache/psr6_psr16_adapters.html



Netgen.
STACK


Q&A

Visit us on social networks

 [netgenlabs](#)

 [netgentweets](#)

 [netgenlabs](#)

 [netgen](#)



Netgen.