+ const DEPTH_INFINITY + const NS SABREDAV + \$tree + \$httpResponse + \$httpRequest + \$sapi + \$transactionType + \$protectedProperties + \$debugExceptions + \$resourceTypeMapping + \$enablePropfindDepthInfinity + \$xml + static \$exposeVersion + static \$streamMultiStatus # \$baseUri # \$plugins __construct(\$treeOrNode =null, HTTP\Sapi \$sapi=null) + start() + exec() + setBaseUri(\$uri) + getBaseUri() + guessBaseUri() - addPlugin(ServerPlugin \$plugin) getPlugin(\$name) ⊦ getPlugins() - getLogger() et 24 de plus..

AbstractAdapter # const NS SEPARATOR - commit() + static createSystemCache (string \$namespace, int \$defaultLifetime, string

(string \$dsn, array \$options=[]) _construct(string

\$version, string \$directory, LoggerInterface \$logger=null) - static createConnection

\$namespace=", in \$defaultLifetime=0)

ArrayAdapter

_construct(int \$default Lifetime=0, bool \$storeSerialized =true, float \$maxLifetime=0,

int \$maxItems=0)

- get(string \$key, callable \$callback, float \$beta =null, array &\$metadata=null)

- save(CacheltemInterface

saveDeferred(CacheItemInterface

- delete(string \$key) __construct(string - hasItem(\$key) \$namespace=", int
- \$defaultLifetime=0) getItem(\$key) doSave(array \$values, getItems(array \$keys=[]) int \$lifetime, array - deleteltem(\$key) \$addTagData=[], array - deleteltems(array
- \$removeTagData=[]) doDelete(array \$ids) doDeleteTagRelations

LoggerAwareInterface

setLogger(LoggerInterfa

AbstractTagAwareAdapter

deleteltems(array

invalidateTags(array

- commit()

\$keys)

\$tags)

\$logger)

- (array \$tagData) doInvalidate(array
- \$taglds) # doDeleteYieldTags (array \$ids)
- commit() + clear(string \$prefix=")

\$keys)

+ getValues() + reset()

TagAwareAdapter - const TAGS PREFIX

construct(AdapterInterface \$itemsPool, AdapterInterface \$tagsPool=null, float \$knownTagVersions

invalidateTags(array \$tags)

Ttl=0.15)

- + hasItem(\$key)
- + getItem(\$key)
- + getItems(array \$keys=[])
- + clear(string \$prefix=") + deleteItem(\$key)
- + deleteItems(array \$kevs)
- + save(CacheltemInterface
- saveDeferred(CacheltemInterface \$item)
- + commit()
- + __sleep() + wakeup()
- + __destruct()

PdoAdapter

\$maxIdLength

+ __construct(\$connOrDsn, string \$namespace=", int \$defaultLifetime=0, array \$options=[], MarshallerInterface \$marshaller=null)

- getItem(\$key)
- getItems(array \$keys=[])
- + hasItem(\$key)
- + deleteItem(\$key)
- + deleteltems(array \$keys)
- + clear(string \$prefix=")
- \$callback, float \$beta =null, array &\$metadata=null) + delete(string \$key)
- + save(CacheltemInterface
- # doDelete(array \$ids)

et 7 de plus...

- # doFetch(array \$ids) # doHave(string \$id)

 - # doClear(string \$namespace)
 - # doDelete(array \$ids)
 - # doSave(array \$values,

int \$lifetime) # getId(\$key)

_construct(string \$namespace=". int \$defaultLifetime=0. string \$directory=null, bool \$appendOnly=false)

PhpFilesAdapter

- + prune()
- # doFetch(array \$ids) # doHave(string \$id)
- int \$lifetime)
- # doDelete(array \$ids) # doUnlink(string \$file)

doClear(string \$namespace)

Psr16Adapter # const NS SEPARATOR

- _construct(CacheInterface \$pool, string \$namespace =", int \$defaultLifetime=0)
- # doFetch(array \$ids) # doHave(string \$id)
- # doClear(string \$namespace)
- # doDelete(array \$ids) # doSave(array \$values, int \$lifetime)

\$marshaller=null)

RedisAdapter

doClear(string \$namespace) # doSave(array \$values, int \$lifetime, array

_construct(string

\$namespace=", int

\$defaultLifetime=0,

string \$directory=null,

FilesystemTagAwareAdapter

MarshallerInterface \$marshaller=null)

- \$addTagData=[], array \$removeTagData=[]) # doDeleteYieldTags
- (array \$ids) # doDeleteTagRelations
- (array \$tagData)
- # doInvalidate(array \$taglds)

RedisTagAwareAdapter

- __construct(\$redis, string \$namespace=" int \$defaultLifetime =0, MarshallerInterface \$marshaller=null)
- # doSave(array \$values, int \$lifetime, array \$addTagData=[], array \$delTagData=[])
- # doDeleteYieldTags (array \$ids)
- (array \$tagData)
- # doInvalidate(array \$taglds)

ApcuAdapter

- construct(string \$namespace=", in \$defaultLifetime=0, string \$version=null MarshallerInterface \$marshaller=null)
- static isSupported()
- # doFetch(array \$ids) # doHave(string \$id)
- # doClear(string \$namespace)
- # doDelete(array \$ids)
- # doSave(array \$values, int \$lifetime)

CouchbaseBucketAdapter

- construct(\CouchbaseBucket \$bucket, string \$namespace =", int \$defaultLifetime=0, MarshallerInterface \$marshaller=null)
- static createConnection
- # doFetch(array \$ids)
- # doClear(string \$namespace)
- int \$lifetime)

- # doDelete(array \$ids)

- (\$servers, array \$options=[])
- static isSupported()
- # doHave(string \$id)
- # doSave(array \$values,

CouchbaseCollectionAdapter

- construct(Collection \$connection, string \$namespace=", int \$defaultLifetime
- =0, MarshallerInterface \$marshaller=null) - static createConnection
- (\$dsn, array \$options=[])
- static isSupported() # doFetch(array \$ids)
- # doClear(\$namespace)

doHave(\$id)

doDelete(array \$ids) # doSave(array \$values, \$lifetime)

DoctrineAdapter

- + __construct(CacheProvider \$provider, string \$namespace
 - =", int \$defaultLifetime=0) + reset()

doDelete(array \$ids)

int \$lifetime)

doSave(array \$values,

- # doFetch(array \$ids)
- # doHave(string \$id) + prune() # doClear(string \$namespace) # doFetch(array \$ids)
 - # doHave(string \$id) # doClear(string \$namespace)

int \$lifetime)

getId(\$key)

\$maxIdLength

doDelete(array \$ids) # doSave(array \$values,

DoctrineDbalAdapter

array \$options=[], MarshallerInterface

_construct(\$connOrDsn,

configureSchema(Schema

\$schema. Connection

string \$namespace=",

int \$defaultLifetime=0,

\$marshaller=null)

\$forConnection)

createTable()

_construct(string \$namespace=", int \$defaultLifetime=0, string \$directory=null, MarshallerInterface \$marshaller=null)

FilesystemAdapter

MemcachedAdapter

- # \$maxIdLength construct(Memcached \$client, string \$namespace
- =". int \$defaultLifetime =0, MarshallerInterface \$marshaller=null)
- static isSupported() static createConnection + get(string \$key, callable (\$servers, array \$options=[])
- doSave(array \$values, int \$lifetime)
- # doFetch(array \$ids) # doHave(string \$id)
- # doClear(string \$namespace)

- + static isSupported()
- # doSave(array \$values,

- _construct(\$redis,
- string \$namespace=' int \$defaultLifetime =0. MarshallerInterface

- # doDeleteTagRelations