PSR-4: Autoloader

The key words “MUST”, “MUST NOT”, “REQUIRED”, “SHALL”, “SHALL NOT”, “SHOULD”, “SHOULD NOT”, “RECOMMENDED”, “MAY”, and “OPTIONAL” in this document are to be interpreted as described in RFC 2119.

1. Overview

This PSR describes a specification for autoloading classes from file paths. It is fully interoperable, and can be used in addition to any other autoloading specification, including PSR-0. This PSR also describes where to place files that will be autoloaded according to the specification.

2. Specification

1. The term “class” refers to classes, interfaces, traits, and other similar structures.

2. A fully qualified class name has the following form:

    ```
    \<NamespaceName>(\<SubNamespaceNames>)*\<ClassName>
    ```

    1. The fully qualified class name MUST have a top-level namespace name, also known as a “vendor namespace”.

    2. The fully qualified class name MAY have one or more sub-namespace names.

    3. The fully qualified class name MUST have a terminating class name.

    4. Underscores have no special meaning in any portion of the fully qualified class name.

    5. Alphabetic characters in the fully qualified class name MAY be any combination of lower case and upper case.

    6. All class names MUST be referenced in a case-sensitive fashion.

3. When loading a file that corresponds to a fully qualified class name …

    1. A contiguous series of one or more leading namespace and sub-namespace names, not including the leading namespace separator, in the fully qualified class name (a “namespace prefix”) corresponds to at least one “base directory”.

2. The contiguous sub-namespace names after the “namespace prefix” correspond to a subdirectory within a “base directory”, in which the namespace separators represent directory separators. The subdirectory name MUST match the case of the sub-namespace names.

3. The terminating class name corresponds to a file name ending in `.php`. The file name MUST match the case of the terminating class name.

4. Autoloader implementations MUST NOT throw exceptions, MUST NOT raise errors of any level, and SHOULD NOT return a value.

3. Examples

The table below shows the corresponding file path for a given fully qualified class name, namespace prefix, and base directory.

FULLY QUALIFIED CLASS NAME	NAMESPACE PREFIX	BASE DIRECTORY	RESULTING FILE PATH
\Acme\Log\Writer\File_Writer	Acme\Log\Writer	./acme-log-writer/lib/	./acme-log-writer/lib/File_Writer.php
\Aura\Web\Response\Status	Aura\Web	/path/to/aura-web/src/	/path/to/aura-web/src/Response/Status.php
\Symfony\Core\Request	Symfony\Core	./vendor/Symfony/Core/	./vendor/Symfony/Core/Request.php
\Zend\Acl	Zend	/usr/includes/Zend/	/usr/includes/Zend/Acl.php

For example implementations of autoloaders conforming to the specification, please see the examples file. Example implementations MUST NOT be regarded as part of the specification and MAY change at any time.