# 自动加载规范

**此规范已被弃用** - 本规范已于2014年10月21日被标记为弃用，目前最新的替代规范为 [PSR-4] 。

本文是为自动加载器（autoloader）实现通用自动加载，所需要遵循的编码规范。

## 规范说明

* 一个标准的 命名空间(namespace) 与 类(class) 名称的定义必须符合以下结构： \<Vendor Name>\(<Namespace>\)\*<Class Name>；
* 其中Vendor Name为每个命名空间都必须要有的一个顶级命名空间名；
* 需要的话，每个命名空间下可以拥有多个子命名空间；
* 当根据完整的命名空间名从文件系统中载入类文件时，每个命名空间之间的分隔符都会被转换成文件夹路径分隔符；
* 类名称中的每个 \_ 字符也会被转换成文件夹路径分隔符，而命名空间中的 \_ 字符则是无特殊含义的。
* 当从文件系统中载入标准的命名空间或类时，都将添加 .php 为目标文件后缀；
* 组织名称(Vendor Name)、命名空间(Namespace) 以及 类的名称(Class Name) 可由任意大小写字母组成。

## 范例

* \Doctrine\Common\IsolatedClassLoader => /path/to/project/lib/vendor/Doctrine/Common/IsolatedClassLoader.php
* \Symfony\Core\Request => /path/to/project/lib/vendor/Symfony/Core/Request.php
* \Zend\Acl => /path/to/project/lib/vendor/Zend/Acl.php
* \Zend\Mail\Message => /path/to/project/lib/vendor/Zend/Mail/Message.php

## 命名空间以及类名称中的下划线

* \namespace\package\Class\_Name => /path/to/project/lib/vendor/namespace/package/Class/Name.php
* \namespace\package\_name\Class\_Name => /path/to/project/lib/vendor/namespace/package\_name/Class/Name.php

以上是使用通用自动加载必须遵循的最低规范标准， 可通过以下的示例函数 SplClassLoader 载入 PHP 5.3 的类文件，来验证你所写的命名空间以及类是否符合以上规范。

## 实例

以下示例函数为本规范的一个简单实现。

<?php  
  
function autoload($className)  
{  
 $className = ltrim($className, '\\');  
 $fileName = '';  
 $namespace = '';  
 if ($lastNsPos = strrpos($className, '\\')) {  
 $namespace = substr($className, 0, $lastNsPos);  
 $className = substr($className, $lastNsPos + 1);  
 $fileName = str\_replace('\\', DIRECTORY\_SEPARATOR, $namespace) . DIRECTORY\_SEPARATOR;  
 }  
 $fileName .= str\_replace('\_', DIRECTORY\_SEPARATOR, $className) . '.php';  
  
 require $fileName;  
}

## SplClassLoader 实例

以下的 gist 是 一个 SplClassLoader 类文件的实例，如果你遵循了以上规范，可以把它用来载入你的类文件。 这也是目前 PHP 5.3 建议的类文件载入方式。

* <http://gist.github.com/221634>