**1、APC缓存简介**

APC，全称是Alternative PHP Cache，官方翻译叫”可选PHP缓存”。它为我们提供了缓存和优化PHP的中间代码的框架。 APC的缓存分两部分:系统缓存和用户数据缓存。

**系统缓存**

它是指APC把PHP文件源码的编译结果缓存起来，然后在每次调用时先对比时间标记。如果未过期，则使用缓存的中间代码运行。默认缓存

3600s(一小时)。但是这样仍会浪费大量CPU时间。因此可以在php.ini中设置system缓存为永不过期(apc.ttl=0)。不过如果这样设置，改运php代码后需要重启WEB服务器。目前使用较多的是指此类缓存。

**用户数据缓存**

缓存由用户在编写PHP代码时用apc\_store和apc\_fetch函数操作读取、写入的。如果数据量不大的话，可以一试。如果数据量大，使用类似memcache此类的更加专著的内存缓存方案会更好

**缓存key生成规则**

APC的缓存中的每个slot都会有一个key，key是

apc\_cache\_key\_t结构体类型，除了key相关的属性，关键是h字段的生成。 h字段决定了此元素落于slots数组的哪一个位置。对于用户缓存和系统缓存，其生成规则不同。 用户缓存通过apc\_cache\_make\_user\_key函数生成key。通过用户传递进来的key字符串，依赖PHP内核中的hash函数（PHP的hashtable所使用的hash函数：zend\_inline\_hash\_func），生成h值。

系统缓存通过apc\_cache\_make\_file\_key函数生成key。通过APC的配置项apc.stat的开关来区别对待不同的方案。在打开的情况下，即

apc.stat= On 时，如果被更新则自动重新编译和缓存编译后的内容。此时的h值是文件的device和inode相加所得的值。在关闭的情况下，即apc.stat=off时，当文件被修改后，如果要使更新的内容生效，则必须重启Web服务器。此时h值是根据文件的路径地址生成，并且这里的路径是绝对路径。即使你是使用的相对路径，也会查找PG(include\_path)定位文件，以取得绝对路径，所以使用绝对路径会跳过检查，可以提高代码的效率。

**常用APC设置**

有时候，$\_ENV会为空，其原因通常是php的配置文件[php.ini](https://www.baidu.com/s?wd=php.ini&from=1012015a&fenlei=mv6quAkxTZn0IZRqIHckPjm4nH00T1dBmWm3uWNbPA7bmycdPW040ZwV5Hcvrjm3rH6sPfKWUMw85HfYnjn4nH6sgvPsT6KdThsqpZwYTjCEQLGCpyw9Uz4Bmy-bIi4WUvYETgN-TLwGUv3EPHm3nHTzP163" \t "_blank)的配置项为：variables\_order = "GPCS"。要想让$\_ENV的值不为空，那么variables\_order的值应该加上一个大写字母“E”，即：variables\_order = "EGPCS"。

上述配置表示了PHP接受的外部变量来源及顺序，EGPCS是Environment、Get、Post、Cookies、Server的缩写。如果variables\_order (变量规则)的配置中缺少E ，则PHP 无法接受环境变量，那么$\_ENV 也就为空了。

SSH Secure Shell Client

[编辑](javascript:;)

SSH是一个用来替代TELNET、FTP以及R命令的工具包，主要是想解决口令在网上[明文传输](https://baike.baidu.com/item/%E6%98%8E%E6%96%87%E4%BC%A0%E8%BE%93" \t "_blank)的问题。为了系统安全和用户自身的权益，推广SSH是必要的。 SSH是英文Secure Shell的简写形式。通过使用SSH，你可以把所有传输的数据进行加密，这样"中间人"这种攻击方式就不可能实现了，而且也能够防止DNS欺骗和IP欺骗。

PHP中$\_SERVER的详细参数与说明  
  
[PHP编程](https://www.baidu.com/s?wd=PHP%E7%BC%96%E7%A8%8B&from=1012015a&fenlei=mv6quAkxTZn0IZRqIHckPjm4nH00T1dWnHwWn1csnv7WPW7BuANW0ZwV5Hcvrjm3rH6sPfKWUMw85HfYnjn4nH6sgvPsT6KdThsqpZwYTjCEQLGCpyw9Uz4Bmy-bIi4WUvYETgN-TLwGUv3EnWDYnWfdP1mY)中经常需要用到一些服务器的一些资料，特把$\_SERVER的详细参数整理下，方便以后使用。  
  
$\_SERVER['PHP\_SELF'] #当前正在执行脚本的文件名，与 document root相关。  
$\_SERVER['argv'] #传递给该脚本的参数。  
$\_SERVER['argc'] #包含传递给程序的命令行参数的个数（如果运行在命令行模式）。  
$\_SERVER['GATEWAY\_INTERFACE'] #服务器使用的 CGI 规范的版本。例如，“CGI/1.1”。  
$\_SERVER['SERVER\_NAME'] #当前运行脚本所在服务器主机的名称。  
$\_SERVER['SERVER\_SOFTWARE'] #服务器标识的字串，在响应请求时的头部中给出。  
$\_SERVER['SERVER\_PROTOCOL'] #请求页面时通信协议的名称和版本。例如，“HTTP/1.0”。  
$\_SERVER['REQUEST\_METHOD'] #访问页面时的请求方法。例如：“GET”、“HEAD”，“POST”，“PUT”。  
$\_SERVER['QUERY\_STRING'] #查询(query)的字符串。  
$\_SERVER['DOCUMENT\_ROOT'] #当前运行脚本所在的文档根目录。在服务器配置文件中定义。  
$\_SERVER['HTTP\_ACCEPT'] #当前请求的 Accept: 头部的内容。  
$\_SERVER['HTTP\_ACCEPT\_CHARSET'] #当前请求的 Accept-Charset: 头部的内容。例如：“iso-8859-1,\*,utf-8”。  
$\_SERVER['HTTP\_ACCEPT\_ENCODING'] #当前请求的 [Accept-Encoding](https://www.baidu.com/s?wd=Accept-Encoding&from=1012015a&fenlei=mv6quAkxTZn0IZRqIHckPjm4nH00T1dWnHwWn1csnv7WPW7BuANW0ZwV5Hcvrjm3rH6sPfKWUMw85HfYnjn4nH6sgvPsT6KdThsqpZwYTjCEQLGCpyw9Uz4Bmy-bIi4WUvYETgN-TLwGUv3EnWDYnWfdP1mY): 头部的内容。例如：“gzip”。  
$\_SERVER['HTTP\_ACCEPT\_LANGUAGE']#当前请求的 Accept-Language: 头部的内容。例如：“en”。  
$\_SERVER['HTTP\_CONNECTION'] #当前请求的 Connection: 头部的内容。例如：“Keep-Alive”。  
$\_SERVER['HTTP\_HOST'] #当前请求的 Host: 头部的内容。  
$\_SERVER['[HTTP\_REFERER](https://www.baidu.com/s?wd=HTTP_REFERER&from=1012015a&fenlei=mv6quAkxTZn0IZRqIHckPjm4nH00T1dWnHwWn1csnv7WPW7BuANW0ZwV5Hcvrjm3rH6sPfKWUMw85HfYnjn4nH6sgvPsT6KdThsqpZwYTjCEQLGCpyw9Uz4Bmy-bIi4WUvYETgN-TLwGUv3EnWDYnWfdP1mY)'] #链接到当前页面的前一页面的 URL 地址。  
$\_SERVER['[HTTP\_USER\_AGENT](https://www.baidu.com/s?wd=HTTP_USER_AGENT&from=1012015a&fenlei=mv6quAkxTZn0IZRqIHckPjm4nH00T1dWnHwWn1csnv7WPW7BuANW0ZwV5Hcvrjm3rH6sPfKWUMw85HfYnjn4nH6sgvPsT6KdThsqpZwYTjCEQLGCpyw9Uz4Bmy-bIi4WUvYETgN-TLwGUv3EnWDYnWfdP1mY)'] #当前请求的 User\_Agent: 头部的内容。  
$\_SERVER['HTTPS'] — 如果通过https访问,则被设为一个非空的值(on)，否则返回off  
$\_SERVER['REMOTE\_ADDR'] #正在浏览当前页面用户的 IP 地址。  
$\_SERVER['REMOTE\_HOST'] #正在浏览当前页面用户的主机名。  
$\_SERVER['REMOTE\_PORT'] #用户连接到服务器时所使用的端口。  
$\_SERVER['SCRIPT\_FILENAME'] #当前执行脚本的[绝对路径](https://www.baidu.com/s?wd=%E7%BB%9D%E5%AF%B9%E8%B7%AF%E5%BE%84&from=1012015a&fenlei=mv6quAkxTZn0IZRqIHckPjm4nH00T1dWnHwWn1csnv7WPW7BuANW0ZwV5Hcvrjm3rH6sPfKWUMw85HfYnjn4nH6sgvPsT6KdThsqpZwYTjCEQLGCpyw9Uz4Bmy-bIi4WUvYETgN-TLwGUv3EnWDYnWfdP1mY" \t "_blank)名。  
$\_SERVER['SERVER\_ADMIN'] #管理员信息  
$\_SERVER['SERVER\_PORT'] #服务器所使用的端口  
$\_SERVER['SERVER\_SIGNATURE'] #包含服务器版本和虚拟主机名的字符串。  
$\_SERVER['PATH\_TRANSLATED'] #当前脚本所在文件系统（不是文档根目录）的基本路径。  
$\_SERVER['SCRIPT\_NAME'] #包含当前脚本的路径。这在页面需要指向自己时非常有用。  
$\_SERVER['REQUEST\_URI'] #访问此页面所需的 URI。例如，“/index.html”。  
$\_SERVER['PHP\_AUTH\_USER'] #当 PHP 运行在 Apache 模块方式下，并且正在使用 HTTP 认证功能，这个变量便是用户输入的用户名。  
$\_SERVER['PHP\_AUTH\_PW'] #当 PHP 运行在 Apache 模块方式下，并且正在使用 HTTP 认证功能，这个变量便是用户输入的密码。  
$\_SERVER['AUTH\_TYPE'] #当 PHP 运行在 Apache 模块方式下，并且正在使用 HTTP 认证功能，这个变量便是认证的类型。

**Note**: 这个变量仅在 [register\_argc\_argv](http://php.net/manual/zh/ini.core.php" \l "ini.register-argc-argv) 打开时可用。

$argc — 传递给脚本的参数数目

$argv — 传递给脚本的参数数组

$php\_errormsg 变量包含由 PHP 生成的最新错误信息。这个变量只在错误发生的作用域内可用，并且要求 [track\_errors](mk:@MSITStore:C:\Users\Administrator\Desktop\php_manual_zh.chm::/res/errorfunc.configuration.html#ini.track-errors) 配置项是开启的（默认是关闭的）。