# [http server原理，nginx与php之间是如何工作的](http://blog.csdn.net/yqbaa/article/details/49280775)

## 一、nginx和http工作的一般流程

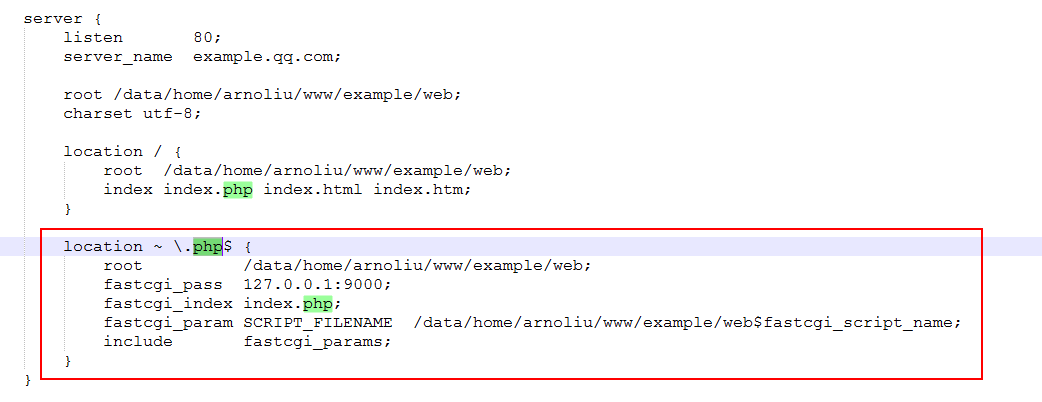
Nginx (“engine x”) 是一个高性能的 HTTP 和 反向代理 服务器。

1、当nginx接收到一个http请求时，通过配置文件找到对应的server。

2、然后匹配server中的所有location，找到最匹配的。

3、在location中的命令会启动不同的模块去完成工作，比如rewrite模块、index模块。因此在nginx中模块可以看作真正的劳动工作者。nginx的模块是被编译到nginx中的，属于静态方式。启动nginx时，模块被自动加载。

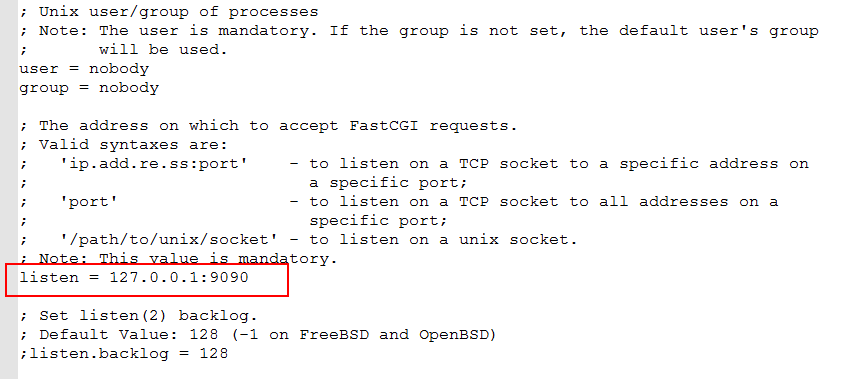
## 二、nginx调用[PHP](http://lib.csdn.net/base/php)进行工作



这个location指令把以php为文件后缀的请求，交给127.0.0.1:9000处理。我想你看到这个应该猜到了，这是一个C/S[**架构**](http://lib.csdn.net/base/architecture)东西。 而这里的IP地址和端口（127.0.0.1:9000）就是fastcgi进程监听的IP地址和端口。

fastcgi是一个可伸缩地、高速地在http server和动态脚本语言间通信的接口。多数流行的http server都支持fastcgi，包括apache、nginx和lighttpd等。同时，fastcgi也被许多脚本语言支持，其中就有php。

## 三、fastcgi的配置IP和端口的来源：在php-fpm.conf中可以看到如下：



fastcgi进程管理器php-fpm自身初始化，启动主进程php-fpm和启动start\_servers个fastcgi子进程。

* 主进程php-fpm主要是管理fastcgi子进程，监听9000端口，fastcgi子进程等待请求。
* 当客户端请求到达nginx时，nginx通过location指令，将所有以php为后缀的文件都交给 127.0.0.1:9000 来处理。
* php-fpm选择并连接到一个fastcgi子进程，并将环境变量和标准输入发送到fastcgi子进程。
* fastcgi子进程完成处理后将标准输出和错误信息返回。
* 当fastcgi子进程关闭连接时，说明处理完成，等待下次处理。

四、普通CGI和FastCGI的区别  
1、普通的cgi模式是进程，每个cgi只能处理一个请求；换言之，每来一个请求就必须要创建一个cgi进程，导致效率缓慢

2、所以才有了 fast cgi fpm就是 fast cgi manager ：fpm的模式也是进程，但他会预创建，就是服务启动的时候，先创建一部分进程等待处理请求，当然，一个FastCGI进程可以处理若干请求。

## 四、认识CGI

不是一种语言，也不是一种技术，而是一种模式

CGI定义：Common Gateway Interface，简称CGI。在物理上是一段程序，运行在服务器上，提供同客户端 HTML页面的接口。

“通用网关接口”：这就是CGI的中文意思。

换句话，只要是提供HTML的服务器端程序都可以叫CGI。APS、PHP、JSP这些都是，你用C语言写一个可以提供HTML的服务器端EXE文件，也叫CGI。

## 五、Php-Fpm的操作

PHP5.3.3的php-fpm 不再支持php-fpm 以前具有的 /usr/local/php/sbin/php-fpm(start|stop|reload)等命令，需要使用信号控制：

master进程可以理解以下信号：

INT, TERM 立刻终止 ;

QUIT  平滑终止 ;

USR1  重新打开日志文件 ;

USR2 平滑重载所有worker进程并重新载入配置和二进制模块 ;

示例：

php-fpm 关闭： kill -INT`cat /usr/local/php/var/run/php-fpm.pid`

php-fpm 重启： kill -USR2`cat /usr/local/php/var/run/php-fpm.pid`

查看php-fpm进程数：

$ps aux | grep -c php-fpm