

欢迎使用fastcgi_cpp

by: dragon jiang < jianlinlong@gmail.com >

fastcgi_cpp 是对fastcgi官方C++库的封装，他即简单易用又功能强大。结合nginx + fastcgi_cpp可快速开发出高效灵活的WEB程序，实在是C++开发WEB应用的必用精品。

0. 体系架构

欲知fastcgi程序的体系统架构，请访问这篇文章(此文画的图实在是太好了！)

[点我访问](#)

1. 安装nginx

[nginx官网下载地址](#)

下载稳定版，而后编译安装。参考如下步骤

1. #./configure
2. #make
3. #make install

2. 安装spawn-fcgi

[下载地址](#)

spawn-fcgi是用来启动fastcgi程序的，我们开发好的可执行程序需要用这个工具来启动。 下载后请编译并安装

3. 安装fastcgi官方库

[下载地址](#)

下载后编译并安装

1. #./configure
2. #make
3. #make install

编译好的动态库位于 libfcgi/.libs ，并且已复制到 /usr/local/lib/目录下。如果某些情况程序找不到动态库，则需要配置LD_LIBRARY_PATH指向此目录。

4. 使用fastcgi_cpp

fastcgi_cpp提供了Http_Application、Http_Request、Http_Response、 Http_Session等几个类，业务开发只需从Http_Handle_Base继承实现handle并注册即可。我们来看一个Hello World:

```
1. //hello.cpp
2. #include "fastcgi_cpp.h"
3. using namespace fastcgi_cpp;
4.
5. class Hello_World : public Http_Handle_Base
6. {
7. public:
8.     virtual void on_request(Http_Request& req, Http_Response& rsp)
9.     {
10.         rsp.set_header_content_type("text/html");
11.         rsp.write_data("hello world<br>");
12.         rsp.write_data("fastcgi_cpp is a modern c++ web framework base on fastcgi<br>");
13.     }
14. };
```

```

15. REGISTRAR_HANDLE_CLASS(Hello_World); //注册已经实现的Handler类名
16.
17. int main(int argc, char* argv[])
18. {
19.     Http_Application app;
20.     app.add_mapping("hello_world.cgi", "Hello_World"); //url访问地址与Handler绑定, 可实现从配置文件中加载
21.     app.run();
22. }

```

短短几行代码一个WEB程序就写好了。

fastcgi_cpp只有头文件, 但编译时需要链接到fastcgi官方库的动态库, Makefile请参考如下:

```
1. g++ -o hello hello.cpp -std=c++11 -lfcgi
```

5. 程序部署

到此, 我们已安装好了nginx和spawn-fcgi, 并且已经有一个hello可执行程序。

- -->首先配置nginx的配置文件, 启动nginx:

```

1. [root@localhost /]# which nginx
2. /usr/local/nginx/sbin/nginx
3. [root@localhost /]# cd /usr/local/nginx/conf

```

编辑 nginx.conf 这个文件, 加入

```

1. location ~ /\.cgi$ {
2.     root          html;
3.     fastcgi_pass   127.0.0.1:9002;
4.     fastcgi_index  index.cgi;
5.     fastcgi_param  SCRIPT_FILENAME  html$fastcgi_script_name;
6.     include        fastcgi_params;
7. }

```

这样, nginx就会把收到的请求转发到 9002 端口。nginx.conf的完整版请参考附件中的nginx.conf。然后启动nginx

```
1. #nginx
```

- -->用spawn-fcgi 启动应用程序

```
1. #spawn-fcgi -a 127.0.0.1 -p 9002 ./hello
```

- -->使用浏览器测试程序

```

1. [root@localhost .libs]# curl 127.0.0.1/hello_world.cgi
2. hello world<br>fastcgi_cpp is a modern c++ web framework base on fastcgi<br>

```

OK, 我们的程序已经正常工作了

6. 更多例子

请参考fastcgi_test.cpp, 共有6个demo