# wsgi规范

## wsgi内容

### 与cgi的区别

1.wsgi 定义了server与app之间的规范，cgi定义的是server与app进程间通信的方式

2.wsgi针对API的，也就是接口参数标准化

3.wsgi实现三个组件,server middleware app

### server

处理请求时调用web应用(app),app会通过配置的形式注册到server,server会去app中寻找一个application的**可调用对象**(函数或实现\_\_call\_\_的类)，并传入两个参数environ、start\_response,这个可调用对象是app的唯一入口，所有请求都通过这个入口进行处理

示例：

一个最简单的

### middleware

中间件，是server与application之间的桥梁，在wsgi协议中，server不是直接与application通信的，server是通过中间件间接调用application的。

### application

示例：

1．最简单的application

#!/usr/bin/env python

# -\*- coding:utf-8 -\*-

from wsgiref.simple\_server import make\_server

def application(environ, start\_response):

state = '200 OK'

headers = [('Content-type', 'text/plain')]

start\_response(state, headers)

return ['body']

server = make\_server('0.0.0.0', 8889, application)

server.serve\_forever()

2.simple\_server.py源码里的测试app

def demo\_app(environ,start\_response):

from StringIO import StringIO

stdout = StringIO()

print >>stdout, "Hello world!"

print >>stdout

h = environ.items(); h.sort()

for k,v in h:

print >>stdout, k,'=', repr(v)

start\_response("200 OK", [('Content-Type','text/plain')])

return [stdout.getvalue()]

## 部署

通过http server 如Apache、nginx 来实现，这些服务器并没有直接实现WSGI协议，Apache使用mod\_wsgi插件实现WSGI，nginx则只是做代理，分发HTTP请求，将请求发送到实现了WSGI的server如uwsgi然后再从实现了WSGI的server中获得相应的结果，返还给client