# C:\Users\sz\Desktop\dfs\zsCat.pngzsCat

开源分布式框架zaCat 群 559182393

# 服务器文件和图片存储

# 1.图片上传

## 图片服务器

### 传统项目中的图片管理

传统项目中，可以在web项目中添加一个文件夹，来存放上传的图片。例如在工程的根目录WebRoot下创建一个images文件夹。把图片存放在此文件夹中就可以直接使用在工程中引用。

优点：引用方便，便于管理

缺点：

1. 如果是分布式环境图片引用会出现问题。
2. 图片的下载会给服务器增加额外的压力

WEB工程

图片文件夹

例如：WebContent/images

传统图片管理方式在分布式环境中的问题：

第一次请求：图片上传

用户请求

Tomcat1

Image（存放图片）

|-image.jpg

第二次请求：

下载图片

负载均衡服务器

Tomcat2

Image（存放图片）

说好的图片呢？？？？？

### 分布式环境的图片管理

Tomcat1

~~Image（存放图片）~~

~~|-image.jpg~~

Tomcat2

~~Image（存放图片）~~

负载均衡服务器

用户请求

http图片上传

图片服务器

Ip或者域名

Image.taotao.com  
nginx实现http服务

/images/2015/07/29/image.jpg

ftp服务接收上传

ftp上传到图片服务器

http请求图片

http图片上传

分布式环境一般都有一个专门的图片服务器存放图片。

我们使用虚拟机搭建一个专门的服务器来存放图片。在此服务器上安装一个nginx来提供http服务，安装一个ftp服务器来提供图片上传服务。

### 搭建图片服务器

第一步：安装vsftpd提供ftp服务

详见：vsftpd安装手册.doc

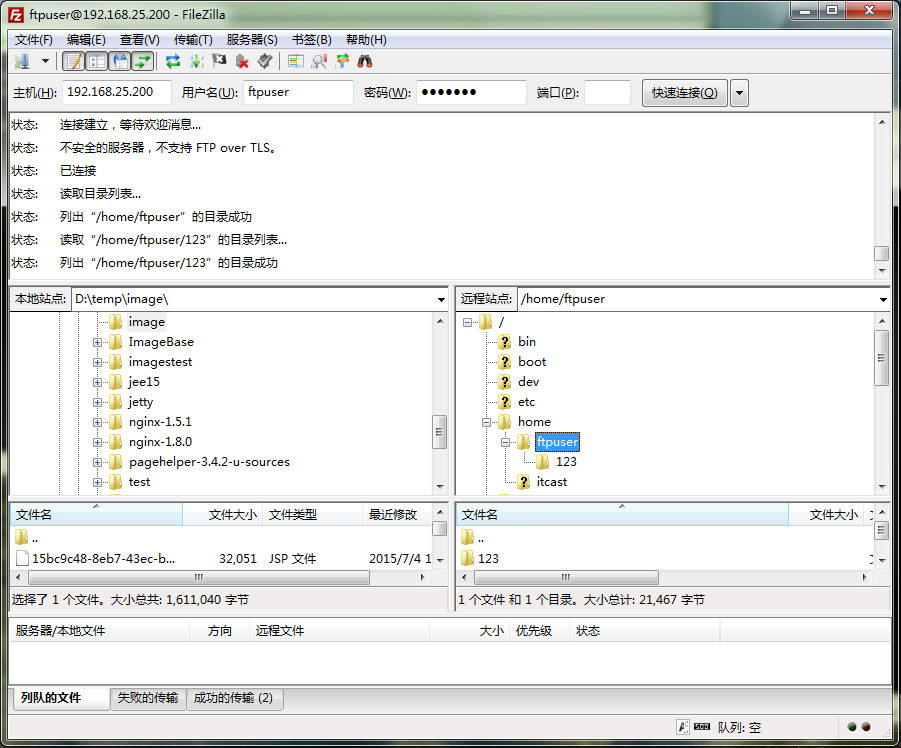
第二步：安装nginx提供http服务

详见：nginx安装手册.doc

### 测试图片服务器

#### ftp服务测试。

1. 使用ftp客户端



1. 使用java程序

ftp可以需要依赖commons-net-3.3.jar包。

|  |
| --- |
| **public** **static** **void** main(String[] args) **throws** Exception {  FTPClient ftpClient = **new** FTPClient();  ftpClient.connect("192.168.25.200");  ftpClient.login("ftpuser", "ftpuser");  FileInputStream inputStream = **new** FileInputStream(**new** File("D:\\Documents\\Pictures\\pics\\21.jpg"));  ftpClient.setFileType(FTP.***BINARY\_FILE\_TYPE***);  ftpClient.storeFile("123.jpg", inputStream);  inputStream.close();    ftpClient.logout();  } |

#### http服务测试

* 1. 浏览器测试

### SpringMVC中实现图片上传

上传思路：

第一步：

导入common-fileupload的依赖

|  |
| --- |
| <!-- 文件上传组件 -->  <dependency>  <groupId>commons-fileupload</groupId>  <artifactId>commons-fileupload</artifactId>  </dependency> |

第二步：

在SpringMVC配置文件中添加文件上传解析器

|  |
| --- |
| <!-- 定义文件上传解析器 -->  <bean id=*"multipartResolver"*  class=*"org.springframework.web.multipart.commons.CommonsMultipartResolver"*>  <!-- 设定默认编码 -->  <property name=*"defaultEncoding"* value=*"UTF-8"*></property>  <!-- 设定文件上传的最大值5MB，5\*1024\*1024 -->  <property name=*"maxUploadSize"* value=*"5242880"*></property>  </bean> |