参看：https://blog.csdn.net/eson\_15/article/details/51736495

# 文件上传—其他服务器

## 导入jar

跨服务器上传文件jar。Io，fileupload



## 需求

上传图片，图片里面回显。Ajax。页面不刷新图片回显。

Ajax能不能提交表单？

<img src=”图片路径”/>

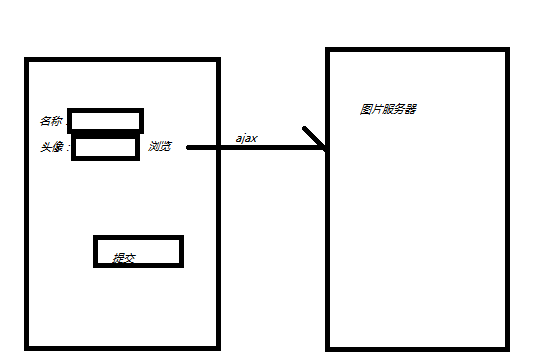
//把文件关联表单

//触发ajax事件

<input type=file onchange=“ajax事件”/>

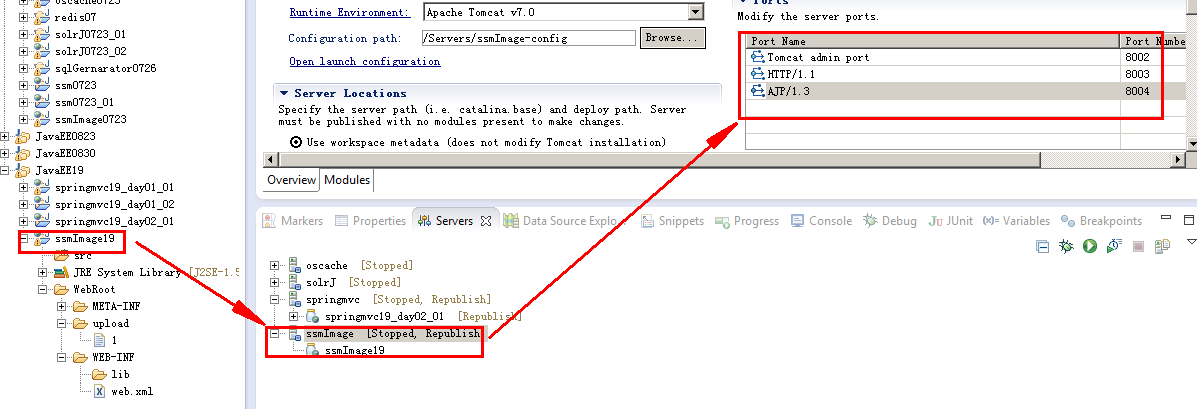
<input type=”hidden” value=”图片相对路径”/>

数据库保存相对路径



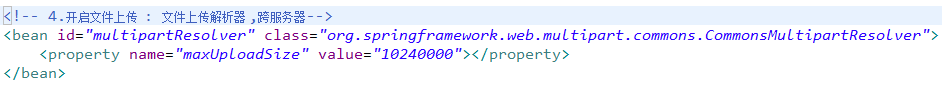
模拟2台服务器：

创建一个项目，图片服务器项目，图片服务器和上传图片的项目端口不一致。



## Springmvc.xml开启文件上传

Springmvc配置文件配置支持文件上传类:

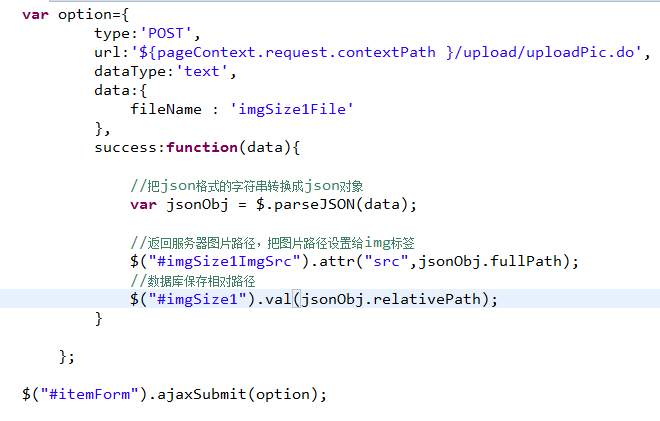


## 页面ajax

发送请求，上传图片：图片被关联表单。提交表单：jquery.form.js

回显：<img src=””>

上传：<input file= > onchange



## uploadController

使用jersy服务器进行跨服务器上传：

@Controller

@RequestMapping("/upload")

**public** **class** UploadController {

@RequestMapping("uploadPic")

**public** **void** uploadPic(HttpServletRequest request,String fileName,PrintWriter out){

//把Request强转成多部件请求对象

MultipartHttpServletRequest mh = (MultipartHttpServletRequest) request;

//根据文件名称获取文件对象

CommonsMultipartFile cm = (CommonsMultipartFile) mh.getFile(fileName);

//获取文件上传流

**byte**[] fbytes = cm.getBytes();

//文件名称在服务器有可能重复？

String newFileName="";

SimpleDateFormat sdf = **new** SimpleDateFormat("yyyyMMddHHmmssSSS");

newFileName = sdf.format(**new** Date());

Random r = **new** Random();

**for**(**int** i =0 ;i<3;i++){

newFileName=newFileName+r.nextInt(10);

}

//获取文件扩展名

String originalFilename = cm.getOriginalFilename();

String suffix = originalFilename.substring(originalFilename.lastIndexOf("."));

//创建jesy服务器，进行跨服务器上传

Client client = Client.*create*();

//把文件关联到远程服务器

WebResource resource = client.resource(Commons.*PIC\_HOST*+"/upload/"+newFileName+suffix);

//上传

resource.put(String.**class**, fbytes);

//ajax回调函数需要会写写什么东西？

//图片需要回显：需要图片完整路径

//数据库保存图片的相对路径.

String fullPath = Commons.*PIC\_HOST*+"/upload/"+newFileName+suffix;

String relativePath="/upload/"+newFileName+suffix;

//{"":"","":""}

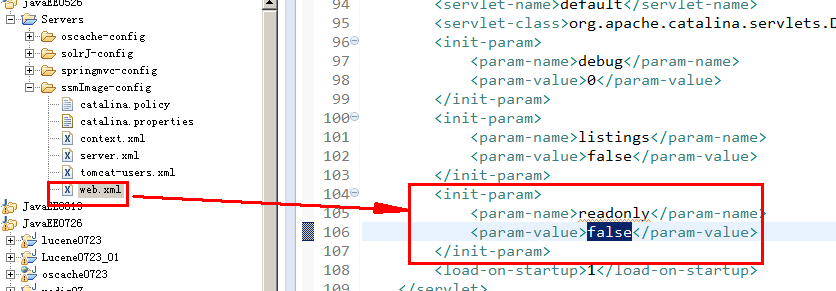
String result="{\"fullPath\":\""+fullPath+"\",\"relativePath\":\""+relativePath+"\"}";

out.print(result);

}

}

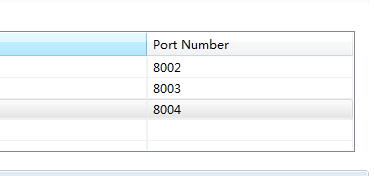
## 修改图片服务器文件上传权限



## 图片上传位置



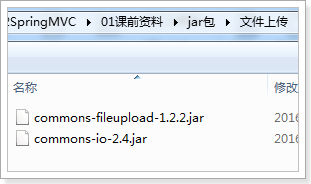
## 重点—我草你大爷



# 上传图片—本地磁盘

## 加入jar包

实现图片上传需要加入的jar包，如下图：



把两个jar包放到工程的lib文件夹中

## 配置上传解析器

在springmvc.xml中配置文件上传解析器

<!-- 文件上传,id必须设置为multipartResolver -->

<bean id=*"multipartResolver"*

class=*"org.springframework.web.multipart.commons.CommonsMultipartResolver"*>

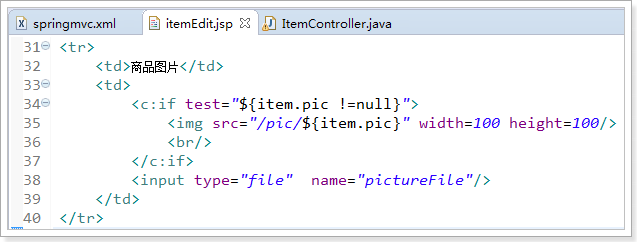
<!-- 设置文件上传大小 -->

<property name=*"maxUploadSize"* value=*"5000000"* />

</bean>

## jsp页面修改

在商品修改页面，打开图片上传功能，如下图：



设置表单可以进行文件上传，如下图：



## 图片上传

在更新商品方法中添加图片上传逻辑

/\*\*

\* 更新商品

\*

\* **@param** item

\* **@return**

\* **@throws** Exception

\*/

@RequestMapping("updateItem")

**public** String updateItemById(Item item, MultipartFile pictureFile) **throws** Exception {

// 图片上传

// 设置图片名称，不能重复，可以使用uuid

String picName = UUID.*randomUUID*().toString();

// 获取文件名

String oriName = pictureFile.getOriginalFilename();

// 获取图片后缀

String extName = oriName.substring(oriName.lastIndexOf("."));

// 开始上传

pictureFile.transferTo(**new** File("C:/upload/image/" + picName + extName));

// 设置图片名到商品中

item.setPic(picName + extName);

// ---------------------------------------------

// 更新商品

**this**.itemService.updateItemById(item);

**return** "forward:/itemEdit.action";

}

效果，如下图：  


# 文件上传—spring mvc

## Controller—单

*//2.图片或视频文件上传 媒体库*@PostMapping(**"/activity/media/upload"**)  
@ResponseBody  
**public** BaseResultVo uploadMedia(  
 MultipartHttpServletRequest multipartRequest,  
 HttpServletRequest request  
) {  
 BaseResultVo brv = **new** BaseResultVo();  
 **try** {  
 List<MultipartFile> fileList = multipartRequest.getFiles(**"file"**); *//获得上传组件* MultipartFile myfile = fileList.get(0);*//获得第一个上传的文件* String FileName = **new** SimpleDateFormat(**"yyyyMMddHHmmss"**).format(**new** Date()) + myfile.getOriginalFilename();*// 获取文件名并改名* String fileFolder = request.getSession().getServletContext().getRealPath(**"/upload"**);*//设置上传路径* String pathNameTemp = **"/upload/"** + FileName; *//完整目录文件* File dir = **new** File(fileFolder);  
 **if** (!dir.exists()) {  
 dir.mkdirs();  
 }  
 File file = **new** File(fileFolder, FileName);  
 file.createNewFile();  
 myfile.transferTo(file);*//保存  
  
 //将上传的文件信息保存到数据库* MediaPo mediaPo = **activityService**.addMedia(FileName, pathNameTemp);  
 brv.setData(mediaPo);  
 **return** brv;  
 } **catch** (Exception e) {  
 e.printStackTrace();  
 brv.setCodeMessage(**"1"**, **"上传失败，请检查格式"**);  
 **return** brv;  
 }  
}

## 多文件

*//3.富文本编辑器图片或视频文件上传 媒体库*@PostMapping(**"/activity/e3media/upload"**)  
@ResponseBody  
**public** E3ResultVo uploadE3Media(  
 @RequestParam(**"multipartFile"**) MultipartFile[] multipartfiles,  
 HttpServletRequest request  
) {  
 **try** {  
 List<MediaPo> mediaes = **new** ArrayList<>();  
 **if**(multipartfiles != **null** && multipartfiles.**length** != 0){  
 **if**(**null** != multipartfiles && multipartfiles.**length** > 0){  
  
 *//遍历并保存文件* **for**(MultipartFile myfile : multipartfiles){  
 String FileName = **new** SimpleDateFormat(**"yyyyMMddHHmmss"**).format(**new** Date()) + myfile.getOriginalFilename();*// 获取文件名并改名* String fileFolder = request.getSession().getServletContext().getRealPath(**"/upload"**);*//设置上传路径* String pathNameTemp = **"/upload/"** + FileName; *//完整目录文件* File dir = **new** File(fileFolder);  
 **if** (!dir.exists()) {  
 dir.mkdirs();  
 }  
 File file = **new** File(fileFolder, FileName);  
 file.createNewFile();  
 myfile.transferTo(file);*//保存  
  
 //将上传的文件信息保存到数据库* MediaPo media = **new** MediaPo();  
 media.setName(FileName);  
 media.setUrl(pathNameTemp);  
 mediaes.add(media);  
 }  
 }  
 }  
 **return activityService**.addMediaes(mediaes);  
 } **catch** (Exception e) {  
 e.printStackTrace();  
 **return new** E3ResultVo();  
 }  
}

# 文件上传目录

## 服务器目录

### Idea

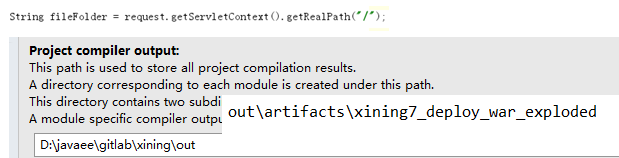
参考：https://blog.csdn.net/jacksonzhou88/article/details/62508188

Intellij IDEA用SpringMVC上传时的路径问题： https://ask.csdn.net/questions/682578?sort=id

注意idea配置tomcat服务器路径



服务器路径：



Idea上传路径不同于eclipse机制，上传的更路径指向的是输出日志artifacts目录下，而不是源文件的webapp下面

### Eclipe

不用修改直接上传到webapp下面

## 本地磁盘

### 手动tomcat配置

在tomcat上配置图片虚拟目录，在tomcat下conf/server.xml中添加：

<Context docBase="D:\develop\upload\temp" path="/pic" reloadable="false"/>

访问http://localhost:8080/pic即可访问D:\develop\upload\temp下的图片。

### Idea

idea设置虚拟路径两种方法：https://blog.csdn.net/cheney550995353/article/details/70185282

### Eclipse

通过eclipse配置，复制一张图片到存放图片的文件夹，使用浏览器访问测试效果，如下图：

