

站内导航

首页 (index.do)

全文检索 | 接口、文章



- 推荐项目
- 产品研发-研发平台
- 吉祥人寿销售支持(吉意宝)
- 产品研发-培训考试系统
- 推荐文章
- 帮助文档
- 技术文档
- 需求规范
- 项目管理
- 接口示例
- 吉祥人寿销售支持(吉意宝)
- 产品研发-产品工厂
- 产品研发-培训考试系统
- 返回码表
- 错误码

数据字典

文章筛选：

帮助文档

技术文档

文件管理模块的使用

描述了文件管理模块的操作流程、项目配置、使用示例。

前言：

本文章是《CentOS配置FastDFS手册》的后续。

第一部分 文件管理的操作流程

该部分主要说明前端浏览器等上传文件的操作流程，角色说明：

- 1、用户：操作前端的客户；
- 2、前端：前端APP、前端浏览器网页；
- 3、业务系统：进行实际业务操作的系统；
- 4、权限系统：进行用户鉴权的系统，也可以和业务系统是一个系统；
- 5、文件系统：文件管理系统，负责接收文件；

一、文件上传操作流程：

- 1、前端请求权限系统下发用户上传文件的令牌token；
- 2、权限系统鉴权通过，生成token下发给前端；
- 3、用户选择上传的文件；
- 4、前端将token和用户选择的文件，上传至文件系统；
- 5、文件系统接收到token和文件后，返回“文件串”和文件预览URL；
- 6\*、前端可以显示文件预览给用户；
- 7、用户提交操作数据至业务系统时，前端将“文件串”一并传递给业务系统；
- 8、业务系统获取“文件串”之后，将“文件串”解析为“文件ID”，保存至DB。

二、文件预览操作流程：

当用户预览、下载一个已经被上传的文件时的操作流程：

- 1、业务系统查询DB，获取文件ID；
  - 2、业务系统调用工具类生成预览URL，返回给前端；
  - 3、前端展示给用户；
- 三、业务系统上传、下载、删除文件的操作流程：

- 1、上传，调用工具类构建token，调用工具类是上传；
- 2、下载，先查询DB获取文件ID，调用工具类下载；
- 3、删除，先查询DB获取文件ID，调用工具类删除；

第二部分 常规Service配置支持文件管理

一、添加Maven依赖：

```
<dependency>
    <groupId>com.ifp</groupId>
    <artifactId>ifp-filemanager-support</artifactId>
    <version>0.0.1-SNAPSHOT</version>
</dependency>
```

二、基础配置：

以下配置非必须，只需要使用特定功能时，配置特定项即可，以下配置建议在@Configuration中初始化方法中配置：

1、业务系统需要生成预览URL

生成预览URL需要配置秘钥（既上一篇文章中配置的秘钥），然后调用如下方法：

```
FileManageSupportConfig.setHttpSecretKey(“秘钥串”);
```

## 2、业务系统想要上传、下载、删除文件

业务系统若想想调用文件系统API，除了使用SpringCloud注册中心调用之外，还可以使用HTTP方式直接调用，需要指明调用的地址，然后配置一个WebClient即可：

```
FileManageSupportConfig.setFileManageUrl("文件系统URL");
FileManageSupportConfig.setWebClient(webClient());
```

需要说明的是，文件系统URL格式为：http://ip (http://ip):port/context/

WebClient简单配置可以如下：

```
@Bean
public WebClient webClient() {
    return WebClient.create();
}
```

依赖如下：

```
<dependency>
    <groupId>org.springframework.boot</groupId>
    <artifactId>spring-boot-starter-webflux</artifactId>
</dependency>
```

文档地址：<https://docs.spring.io/spring/docs/5.0.11.RELEASE/spring-framework-reference/web.html#webmvc-webclient> (<https://docs.spring.io/spring/docs/5.0.11.RELEASE/spring-framework-reference/web.html#webmvc-webclient>)

## 第三部分 调用工具类进行具体操作

一、为前端生成文件上传token：

文件上传token中，各个字段、属性解析：

```
# 以下为常量定义：
DEFAULT_EXPIRE_TIME: 2 * 3600, 缺省过期时间；
DEFAULT_FOREVER: -1, 缺省永久有效时间；
COMMON_FIELD_NAME: *, 缺省通用字段值；
# 以下为业务字段：
1、channel: 用户所属渠道，可选；
2、subject: 用户主标识/用户ID，必选；
3、type: 上传文件种类，可选；
4、busId: 上传文件所关联到的业务ID，可选；
# 以下为配置字段：
5、tokenExpire: 上传文件的票据有效期，单位sec，默认DEFAULT_EXPIRE_TIME；
6、previewExpire: 文件上传成功后，生成的预览URL的有效期，单位sec，默认DEFAULT_EXPIRE_TIME；
7、exts: map格式，本次上传的其他附加参数，可选；NOTE:本附加参数会被自动添加到文件元信息中，有关元信息的概念，可以参考fastdfs中元数据的概念；
8、survivalExpire: 文件存活期，单位sec，默认DEFAULT_FOREVER；文件管理模块会定时清理过期文件以释放磁盘；
9、plugins: map格式，当文件上传到文件管理模块之后，需要调用的插件，可选；插件执行完成后，才将文件上传至后续的文件底层管理组件；
10、expectType: 期望的文件类型，当业务端要求用户上传文件的时候，可能需要限制用户上传的文件类型，如：业务端只希望用户上传图片格式的文件；
```

有关各个字段的更详细解释，请参考FileToken类的Javadoc。

1、简单token获取方式：

```
FileManageSupport.getTokenThin("subject");
FileManageSupport.getToken("channel", "subject", "type", "busId");
FileManageSupport.getToken("channel", "subject", "type", "busId", 3600, 7200, "k1", "v1", "k2", "v2");
```

除了subject之外，其他都可以省略，k1、v1、k2、v2为扩展字段；

2、构造器获取方式：

使用构造器构建token，可以使用token中的所有功能：

```
// 获取token构建器
Builder builder = FileManageSupport.getInstance().getTokenBuilder();
// 设置用户信息
builder.setChannel("channel").setSubject("subject").setType("type").setBusId("busid");
builder.addExt("key1", "val1").addExt("key2", "val2");
// 设置需要执行的插件
builder.addPlugs("image", new String[] { FilePlugs.WATER_MARK_PLUG });
builder.addPlugs("file", new String[] { FilePlugs.ZIP_PLUG });
// 设置上传文件的类型限制
builder.addExpectTypes("image", FileType.IMAGE | FileType.VIDEO);
// 构建token串
FileToken tokenObj = builder.build();
FileManageSupport.getInstance().getTokenStr(tokenObj);
```

设置插件时，第一个参数为表单中字段name，第二个参数为针对该文件执行的插件的名称；目前内置水印和ZIP两个插件，如需自定义插件，请参考《文件管理模块自定义文件插件》；

限制文件上传的类型是，第一个参数为表单中字段name，第二个参数为ContentType类型分类，可以通过“|”设置多个类型；

二、为前端生成预览URL：

生成预览URL，需要完成第二部分中的二中的1，然后调用如下方法：

```
FileManageSupport.getPreviewUrl(prefix, groupName, fileId, expire);
```

prefix: URL前缀，http[s]://ip[:port]/context/

groupName: 组ID

fileId: 文件ID

expire: 预览URL过期时长，sec，默认两个小时，FileManageSupport.URL\_FOREVER代表永久有效；

三、业务系统进行文件上传、下载、删除：

业务系统进行文件上传、下载、删除需要完成第二部分中二中的2，然后调用如下方法：

1、上传：

```
// 首先根据一中步骤，构建token对象...
// 然后调用方法：
Map<String, InputSource> map = new HashMap<>();
map.put("image", () -> FileManageApplicationTests.class.getResourceAsStream("photo.jpg"));
map.put("file", () -> FileManageApplicationTests.class.getResourceAsStream("mark.txt"));
FileUploadResult f = FileManageSupport.getInstance().uploadFile(tokenObj, map, null);
```

uploadFile方法参数说明：

第一个参数：构建的token对象，此处不是字符串，是FileToken对象；

第二个参数：存放文件的map，每一个文件使用一个InputSource表示，需要提供一个文件流，且如果想要限制文件类型，必须实现InputSource的getContentType方法，返回正确的ContentType；并且建议覆盖getFileName方法，返回正确的文件名称；

第三个参数：form中的字符串键值对；

2、下载、删除

```
FileManageSupport.getInstance().downloadFile(channel, subject, fileId, out);
FileManageSupport.getInstance().deleteFile(channel, subject, fileId);
```

下载方法需要提供一个输出流，用来获取文件；

删除方法直接删除。

后续，文件管理模块的配置和部署，请参考《文件管理模块配置、部署》

如果想了解文件管理模块的内部架构，请参考《文件管理模块内幕》

---

## 发表评论

请输入评论内容

图形验证码



发表评论

友情链接： 振利软件 (<http://www.n22.com.cn>)