1. Parent pom修改

<parent>

<groupId>com.ymatou.framework</groupId>

<artifactId>platform-bom</artifactId>

<version>1.0.0</version>

</parent>

1. Disconfclient版本修改，解决refections包的版本冲突

<dependency>

<groupId>com.ymatou.disconf</groupId>

<artifactId>disconf-client</artifactId>

<version>2.7.0</version>

<exclusions>

<exclusion>

<groupId>org.reflections</groupId>

<artifactId>reflections</artifactId>

</exclusion>

</exclusions>

</dependency>

1. Dubbo.xml

<dubbo:protocol name=*"rest"* server=*"servlet"* port=*"${server.port}"* threads=*"400"*/>

1. Servlet配置类修改
2. com.ymatou.demo1.configuration.DubboRestWebConfiguration类中restServlet方法实现DispatchServlet注册，该DispatchServlet用于处理dubbo rest请求

/\*\*

\* 兼容dubbo rest，注册处理dubbo rest协议的servlet(/api/\*)，如果是spring mvc，建议注释本方法

\* **@return**

\*/

@Bean

**public** ServletRegistrationBean restServlet(){

DispatcherServlet dispatcherServlet = **new** DispatcherServlet();

//用ServletRegistrationBean包装servlet

ServletRegistrationBean registrationBean

= **new** ServletRegistrationBean(dispatcherServlet);

registrationBean.setLoadOnStartup(2);

//指定urlmapping

registrationBean.addUrlMappings("/api/\*");

//指定name，如果不指定默认为dispatcherServlet

registrationBean.setName("dubboRest");

**return** registrationBean;

}

}

Swagger文档自动生成

默认的spring-fox仅支持restcontroller类自动生成swagger文档，不支持jax-rs，为了支持jax-rs，需要增加自定义的swagger文档服务，操作如下：

1. ServiceApplication类增加@EnableOpenApi注解
2. com.ymatou.demo1.configuration.service.DubboRestDocService类, conf.setResourcePackage("com.ymatou.demo1.facade")方法设置扫描的包名;

@ResponseBody

@GetMapping("/swagger-doc")

**public** Swagger doc() {

ApiListingResource listing = **new** ApiListingResource();

Response rsp = listing.getListing(**this**, **null**, **null**, **null**, "");

**if** (rsp.getStatus() != Status.***OK***.getStatusCode()) {

Info info = **new** Info();

info.setDescription("swagger doc error.");

Swagger swagger = **new** Swagger();

swagger.setInfo(info);

**return** swagger;

}

**return** (Swagger)rsp.getEntity();

}

@PostConstruct

**public** **void** init() {

BeanConfig conf = **new** BeanConfig();

conf.setTitle("demo API");

conf.setDescription("demo ");

conf.setVersion("1.0.0");

conf.setHost("localhost:8889");

conf.setBasePath("/swagger-doc");

conf.setSchemes(**new** String[] { "http" });

conf.setResourcePackage("com.ymatou.demo1.facade");

conf.setScan(**true**);

}

@Override

**public** Set<Class<?>> getClasses() {

Set<Class<?>> resources = **new** HashSet<>();

resources.add(SwaggerSerializers.**class**);

**return** resources;

}

1. 注册swagger resource

@Primary

@Bean

**public** SwaggerResourcesProvider swaggerResourcesProvider(

InMemorySwaggerResourcesProvider defaultResourcesProvider) {

**return** () -> {

SwaggerResource wsResource = **new** SwaggerResource();

wsResource.setName("demoFacade");

wsResource.setSwaggerVersion("2.0");

wsResource.setLocation("/swagger-doc");

wsResource.setUrl("/swagger-doc");

List<SwaggerResource> resources = **new** ArrayList<>(defaultResourcesProvider.get());

resources.add(wsResource);

**return** resources;

};

}

1. 通过<http://localhost:8889/swagger-doc>可以访问到api doc

通过[http://localhost:8889/doc.html 可以访问ui](http://localhost:8889/doc.html%20%20%20可以访问ui)，注意下拉选择

