**JeecgBoot 单体升级微服务**

1. **参考文档**

|  |  |
| --- | --- |
| 官方文档 | <http://doc.jeecg.com/2043906> |
| JeecgBoot项目 | <https://github.com/jeecgboot/jeecg-boot> |

1. **预先下载**

|  |
| --- |
| git clone <https://github.com/jeecgboot/jeecg-boot>.git  docker pull node:16.8.0  docker pull redis:6.2.1  docker pull mysql:5.7.31  docker pull rabbitmq:3.7.7-management |

1. **配置host**

使用*管理员*权限 在 cmd中 运行 notepad C:\Windows\System32\drivers\etc\HOSTS

添加如下内容

|  |
| --- |
| 127.0.0.1 jeecg-boot-redis  127.0.0.1 jeecg-boot-mysql  127.0.0.1 jeecg-boot-nacos  127.0.0.1 jeecg-boot-gateway  127.0.0.1 jeecg-boot-system  127.0.0.1 jeecg-boot-xxljob  127.0.0.1 jeecg-boot-rabbitmq |

1. **安装 Redis**

|  |
| --- |
| docker rm -f jeecg-boot-redis  docker run -d --name jeecg-boot-redis --privileged=true --restart always -p 6379:6379 redis:6.2.1 |

1. **安装Rabbitmq**
2. 安装Rabbitmq

|  |
| --- |
| docker rm -f jeecg-boot-rabbitmq  docker run -d --restart always --hostname jeecg-boot-rabbitmq --name jeecg-boot-rabbitmq -e RABBITMQ\_DEFAULT\_USER=guest -e RABBITMQ\_DEFAULT\_PASS=guest -p 15672:15672 -p 5672:5672 rabbitmq:3.7.7-management |

1. 访问rabbitmq的管理端

|  |
| --- |
| http://localhost:15672/ |

1. 安装延时队列插件

* 参考:

<https://www.cnblogs.com/geekdc/p/13549613.html>

* 插件下载地址:

<https://github.com/rabbitmq/rabbitmq-delayed-message-exchange/releases/download/v3.8.0/rabbitmq_delayed_message_exchange-3.8.0.ez>

* 命令:

|  |
| --- |
| # 下载插件  wget <https://github.com/rabbitmq/rabbitmq-delayed-message-exchange/releases/download/v3.8.0/rabbitmq_delayed_message_exchange-3.8.0.ez>  #拷贝插件到rabbitmq容器  docker cp rabbitmq\_delayed\_message\_exchange-3.8.0.ez jeecg-boot-rabbitmq:/plugins  #进入容器  docker exec -it jeecg-boot-rabbitmq bash  #进入容器并启用插件  rabbitmq-plugins enable rabbitmq\_delayed\_message\_exchange  #查看  rabbitmq-plugins list  #重新启动容器  docker restart jeecg-boot-rabbitmq |

1. **安装 Mysql**

|  |
| --- |
| docker rm -f jeecg-boot-mysql  docker run --name jeecg-boot-mysql -d --restart always -p 3306:3306 \  -e MYSQL\_ROOT\_HOST='%' \  -e MYSQL\_ROOT\_PASSWORD=root \  mysql:5.7.31 \  --character-set-server=utf8mb4 `# 设置字符编码` \  --collation-server=utf8mb4\_unicode\_ci `# 设置字符编码` \  --lower\_case\_table\_names=1 `# 表名不区分大小写` |

1. **将 JEECG-Boot 的数据库导入**
2. 数据的Sql 文件

|  |
| --- |
| https://github.com/jeecgboot/jeecg-boot/blob/v2.4.6/jeecg-boot/db/jeecgboot-mysql-5.7.sql  https://github.com/jeecgboot/jeecg-boot/raw/v2.4.6/jeecg-boot/db/jeecgboot-mysql-5.7.sql |

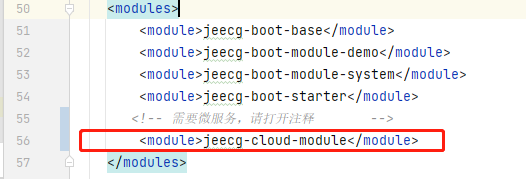
1. Nacos的配置Sql 文件

|  |
| --- |
| https://github.com/jeecgboot/jeecg-boot/blob/v2.4.6/jeecg-boot/db/tables\_nacos.sql  https://github.com/jeecgboot/jeecg-boot/raw/v2.4.6/jeecg-boot/db/tables\_nacos.sql |

1. xxl\_job的配置Sql 文件

|  |
| --- |
| https://github.com/jeecgboot/jeecg-boot/blob/v2.4.6/jeecg-boot/db/tables\_xxl\_job.sql https://raw.githubusercontent.com/jeecgboot/jeecg-boot/v2.4.6/jeecg-boot/db/tables\_xxl\_job.sql |

1. **运行Nacos**
2. 打开 jeecg-boot/jeecg-boot/pom.xml 中的注释



1. 清除开发环境缓存

* Reload jeecg-boot/pom.xml
* 在 IDEA 中运行 File->Invalidate Caches

1. 运行Nacos

|  |
| --- |
| jeecg-boot\jeecg-boot\jeecg-cloud-module\jeecg-cloud-nacos\src\main\java\com\alibaba\nacos\JeecgNacosApplication.java |

1. 访问Nacos

|  |
| --- |
| <http://localhost:8848/nacos> |

1. **启动jeecg-system服务**
2. 删除system项目的pom中demo的依赖

jeecg-boot\jeecg-boot\jeecg-boot-module-system\pom.xml



1. 修改打包配置(修改为true)

jeecg-boot\jeecg-boot-module-system\pom.xml



1. 添加跨域配置

在jeecg-boot\ jeecg-boot-module-system中添加类

|  |
| --- |
| package org.jeecg.config; import org.springframework.context.annotation.Configuration; import org.springframework.web.servlet.config.annotation.CorsRegistry; import org.springframework.web.servlet.config.annotation.WebMvcConfigurerAdapter;  @Configuration public class CorsConfig extends WebMvcConfigurerAdapter {  @Override  public void addCorsMappings(CorsRegistry registry) {  registry.addMapping("/\*\*").allowedOrigins("\*").allowCredentials(true)  .allowedMethods("GET", "POST", "DELETE", "PUT", , "OPTIONS").maxAge(3600);  } } |

1. 启动system微服务 (jeecg-cloud-system-start)

|  |
| --- |
| jeecg-boot\jeecg-cloud-module\jeecg-cloud-system-start\src\main\java\org\jeecg\JeecgSystemCloudApplication.java |

1. 访问后台 api

|  |
| --- |
| http://localhost:7001/doc.html |

1. 验证nacos注册



1. **启动jeecg-demo服务**
2. 添加微服务启动依赖

jeecg-boot\jeecg-boot\jeecg-boot-module-demo\pom.xml

|  |
| --- |
| <!-- 引入jeecg-boot-starter-cloud依赖 -->  <dependency>  <groupId>org.jeecgframework.boot</groupId>  <artifactId>jeecg-boot-starter-cloud</artifactId>  </dependency> |



1. 加入打包插件

jeecg-boot\jeecg-boot\jeecg-boot-module-demo\pom.xml

|  |
| --- |
| <build>  <plugins>  <plugin>  <groupId>org.springframework.boot</groupId>  <artifactId>spring-boot-maven-plugin</artifactId>  </plugin>  </plugins> </build> |



1. 运行

jeecg-boot\jeecg-boot-module-demo\src\main\java\org\jeecg\JeecgDemoCloudApplication.java

1. 访问后台 api

|  |
| --- |
| http://localhost:7002/ |

1. 验证nacos注册



1. **启动后端web**
2. 启动后端web

|  |
| --- |
| cd jeecg-boot\ant-design-vue-jeecg  docker run -it --rm -v $PWD:/jeecgboot -w /jeecgboot -p 3000:3000 node:16.8.0 bash -c "yarn install && yarn serve" |

1. 访问

|  |
| --- |
| http://localhost:3000/ |