

# mall-swarm 微服务电商项目发布重大更新，打造Spring Cloud最佳实践!

原创 梦想de星空 macrozheng 2020-08-19 09:02

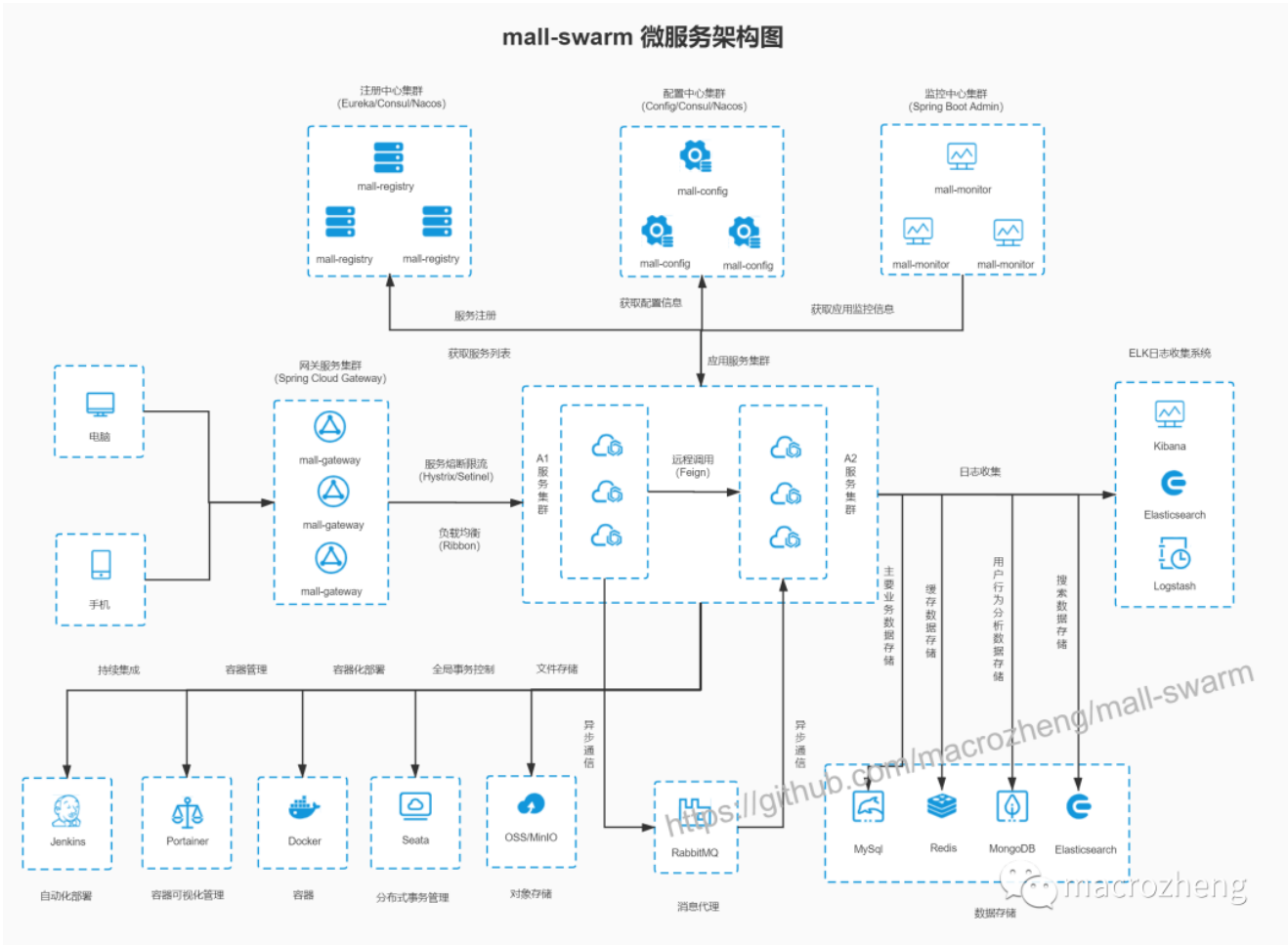
收录于合集

#开源之路

20个

之前把我的 mall 项目更新到了SpringBoot 2.3.0版本，微服务版本 mall-swarm 也已同步更新了。此次更新完善了项目的Spring Cloud技术栈，升级至Spring Cloud Hoxton版本并加入了Spring Cloud Alibaba、Oauth2和Knife4j，致力于打造Spring Cloud 最佳实践项目！

系统架构图



系统架构图

## 项目组织结构

mall

- └─ mall-common -- 工具类及通用代码模块
- └─ mall-mbg -- MyBatisGenerator生成的数据库操作代码模块
- └─ mall-auth -- 基于Spring Security OAuth2的统一的认证中心
- └─ mall-gateway -- 基于Spring Cloud Gateway的微服务API网关服务
- └─ mall-monitor -- 基于Spring Boot Admin的微服务监控中心
- └─ mall-admin -- 后台管理系统服务
- └─ mall-search -- 基于Elasticsearch的商品搜索系统服务
- └─ mall-portal -- 移动端商城系统服务
- └─ mall-demo -- 微服务远程调用测试服务
- └─ config -- 配置中心存储的配置

## 更新内容一览

- 集成Spring Cloud Alibaba，注册中心改用Nacos；
- 权限功能改用OAuth2，实现统一认证和鉴权；
- 集成Knife4j，实现网关聚合API文档；
- 升级Spring Cloud Hoxton.SR5；
- 升级Spring Boot 2.3.0.RELEASE；
- 升级Elasticsearch 7.6.2；
- ELK日志收集功能完善，采用分场景收集日志的方式；
- Window和Linux部署文档更新。

## 更新内容介绍

### Spring Cloud Alibaba

集成了Spring Cloud Alibaba，注册中心和配置中心都改用了Nacos。之前使用的注册中心是Eureka，已经进入维护期不再更新了，之前使用的配置中心是Spring Cloud Config，需要使用消息队列才能实现配置刷新。Nacos既可以当注册中心又可以当配置中心，采用Netty保持TCP长连接实现配置刷新，拥有方便的管理界面，所以就改为使用Nacos了。原来的 `mall-registry` 和 `mall-config` 模块已被移除，在 `config` 文件夹中存放了Nacos中的配置，启动项目时需要导入到Nacos中去。

## Oauth2

- 之前项目中有个 `mall-security` 的模块，我把它称为安全模块。所有需要权限校验的模块都需要依赖它，有点工具包的感觉，在微服务中使用总感觉不伦不类。
- 这次改为使用Oauth2，添加了认证中心 `mall-auth`，实现统一认证和鉴权，更加符合微服务权限控制，具体可以参考[《微服务权限终极解决方案，Spring Cloud Gateway + Oauth2 实现统一认证和鉴权！》](#)。
- 由于项目中存在两套不同的用户体系，后台用户和前台用户，认证中心对多用户体系也有所支持，访问认证中心时使用不同的 `client_id` 和 `client_secret` 即可区分不同用户体系，后台用户使用 `admin-app:123456`，前台用户使用 `portal-app:123456`。
- 对原来的登录接口做了兼容处理，分别会从内部调用认证中心获取Token，依然可以使用。

## Knife4j

之前每个微服务都有自己的API文档地址，需要单独访问。这次把所有微服务的API文档都聚合到了网关上，统一了访问入口，直接访问网关的API文档地址即可，具体可以参考[《微服务聚合Swagger文档，这波操作是真的香！》](#)。

## Spring Cloud Hoxton.SR5

Spring Cloud 和SpringBoot有着版本对应关系，升级到SpringBoot 2.3.0正好对应了该版本。

## Window和Linux部署文档更新

由于部分组件的升级及项目结构的改变，部署文档也更新了，部署有问题的参考最新文档！

- mall-swarm在Windows环境下的部署：  
[http://www.macrozheng.com/#/deploy/mall\\_swarm\\_deploy\\_windows](http://www.macrozheng.com/#/deploy/mall_swarm_deploy_windows)
- mall-swarm在Linux环境下的部署（基于Docker容器）：  
[http://www.macrozheng.com/#/deploy/mall\\_swarm\\_deploy\\_docker](http://www.macrozheng.com/#/deploy/mall_swarm_deploy_docker)

## 其他

其他更新内容和 mall 项目基本相同，具体参考 [《Mall 电商实战项目发布重大更新，全面支持SpringBoot 2.3.0 ! 》](#)。

## 运行效果一览

---

- 查看注册中心服务信息；



- 监控中心应用信息；





- API文档信息；



- 日志收集系统信息;



- 可视化容器管理;





## 项目地址

如此给力的微服务电商实战项目，赶紧点个 **Star** 支持下吧！

<https://github.com/macrozheng/mall-swarm>

## 推荐阅读

- [为什么阿里巴巴禁止使用\*\*Apache BeanUtils\*\*进行属性拷贝？](#)
- [10个解放双手的 \*\*IDEA\*\* 插件，少些冤枉代码！](#)
- [Docker服务开放了这个端口，服务器分分钟变肉机！](#)
- [Mall 电商实战项目发布重大更新，全面支持\*\*SpringBoot 2.3.0\*\*！](#)
- [秒杀商品超卖事故：\*\*Redis\*\*分布式锁请慎用！](#)
- [被我用烂的\*\*DEBUG\*\*调试技巧，专治各种搜索不到的问题！](#)

- 我扒了半天源码，终于找到了**Oauth2**自定义处理结果的最佳方案！
  - 别再**if-else**走天下了，整个注解多优雅！
  - 一个不容错过的**Spring Cloud**实战项目！
  - 我的**Github**开源项目，从**0**到**20000 Star**！
- 



欢迎关注，点个在看

阅读原文

喜欢此内容的人还喜欢

项目中到底该不该用Lombok?

macrozheng

