

# 实验文档

## 项目基本信息

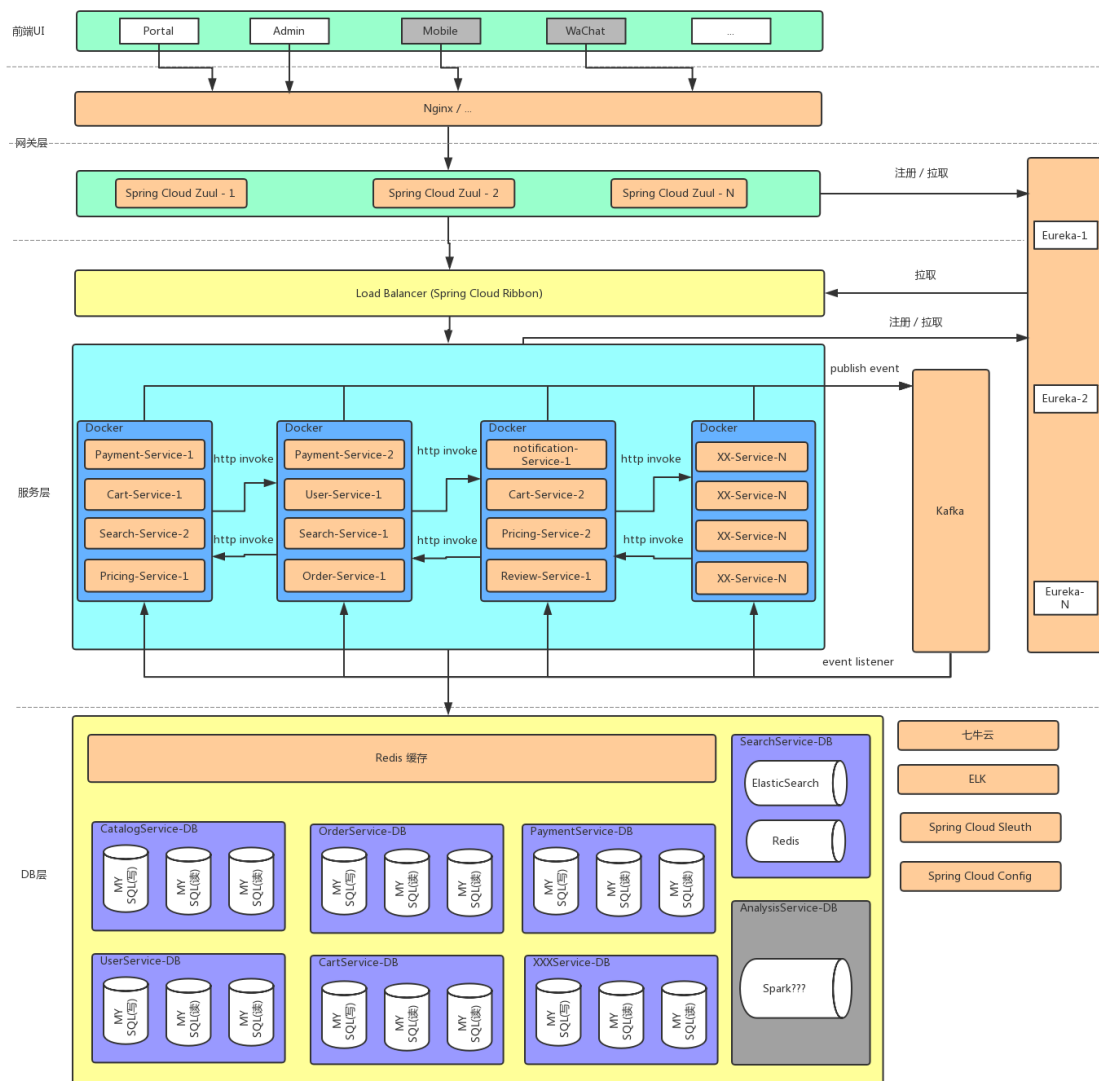
---

### 技术栈

- 服务注册：Spring Cloud Eureka Finchley.M2
- 服务网关：Spring Cloud Zuul Finchley.M2
- 负载均衡：Spring Cloud Ribbon Finchley.M2
- 服务配置：Spring Cloud Config Finchley.M2
- 服务追踪：Spring Cloud Sleuth Finchley.M2
- 持久层框架：Mybatis 3.4.4
- 数据库连接池：Druid
- 数据库：MySQL 5.6
- 全文检索：ElasticSearch 6.2.2
- 消息队列：Kafka 1.0.0
- 缓存和内存数据库：Redis
- 短信服务：阿里大于
- 图片存储：七牛云
- 支付：支付宝当面付沙箱
- API文档：Swagger
- 日志组件：Log4j2, ELK
- 构建工具：Maven
- 部署环境：Docker
- 插件：Lombok

### 项目架构

该项目使用微服务架构，具体架构图如下



## 项目模块

模块	功能
bigfans-cloud-api-gateway	微服务网关,基于SpringCloud-Zuul
bigfans-cloud-config-server	微服务配置中心服务端,基于SpringCloud-Config
bigfans-cloud-discovery-eureka	微服务注册中心,基于SpringCloud-Eureka
bigfans-cloud-service-cart	购物车服务, 提供用户购物车,临时购物车等服务
bigfans-cloud-service-catalog	目录服务,提供商品,分类,品牌,属性,规格,sku库存等服务
bigfans-cloud-service-notification	通知服务,发送短信,邮件等通知
bigfans-cloud-service-order	订单服务,提供下单,结算等服务
bigfans-cloud-service-payment	支付服务,与第三方集成提供支付功能,例如支付宝
bigfans-cloud-service-pricing	计价服务,计算订单,购物车,商品优惠的价格,提供优惠信息等
bigfans-cloud-service-review	评论服务, 提供商品评论,反馈,投诉等服务
bigfans-cloud-service-search	搜索服务, 提供全文检索,相似商品推荐,搜索提示等
bigfans-cloud-service-shipping	配送服务, 目前没实现
bigfans-cloud-service-system	系统服务, 提供系统通用信息
bigfans-cloud-service-user	用户服务, 提供注册,登录,用户地址,信息等

模块	功能
bigfans-cloud- zipkin-server	zipkin请求日志追踪服务端,基于SpringCloud-Sleuth
bigfans-framework	基础框架,提供通用封装,不掺杂任何业务代码,可以直接引入到其他项目使用
bigfans-cloud-base	所有服务的基础模块,提供通用的模型,事件等

## 采集过程

### 项目编译文件的采集过程：

1. 切换到指定分支，并执行 `mvn compile`

```
[INFO] --- maven-resources-plugin:2.6:resources (default-resources) @ bigfans-framework ---
[INFO] Using 'UTF-8' encoding to copy filtered resources.
[INFO] skip non existing resourceDirectory G:\OneDrive\code\java\bigfans-cloud\bigfans-framework\src\main\resources
[INFO]
[INFO] --- maven-compiler-plugin:3.7.0:compile (default-compile) @ bigfans-framework ---
[INFO] Changes detected - recompiling the module!
[INFO] Compiling 158 source files to G:\OneDrive\code\java\bigfans-cloud\bigfans-framework\target\classes
[INFO] G:/OneDrive/code/java/bigfans-cloud/bigfans-framework/src/main/java/com/bigfans/framework/aspect/CacheAspect.java: 某些输入文件使用或覆盖了已过时的 API。
[INFO] G:/OneDrive/code/java/bigfans-cloud/bigfans-framework/src/main/java/com/bigfans/framework/aspect/CacheAspect.java: 有关详细信息，请使用 -Xlint:deprecation 重新编译。
[INFO] G:/OneDrive/code/java/bigfans-cloud/bigfans-framework/src/main/java/com/bigfans/framework/kafka/KafkaConsumerTaskManager.java: 某些输入文件使用了未经检查或不安全的操作。
[INFO] G:/OneDrive/code/java/bigfans-cloud/bigfans-framework/src/main/java/com/bigfans/framework/kafka/KafkaConsumerTaskManager.java: 有关详细信息，请使用 -Xlint:unchecked 重新编译。
```

### 项目包依赖的采集过程：

1. 使用 `IDEA` 进行采集，优化导入之后，导出依赖至XML文档



```
42     jsonData = list()
43     for packageName in packageDict:
44         # print("packageName:{}, dependency:{}".format(packageName, packageDict[packageName]))
45         jsonData.append({
46             "name": packageName,
47             "dependency": list(packageDict[packageName])
48         })
49     print(json.dumps(jsonData))
50     # print(re.search(projectPackagePatter, path))
51
52     # print(L[1])
53
```

问题 43 输出 调试控制台 终端 1: cmd

```
G:\OneDrive\code\python\体系结构>python xml2json.py
[{"name": "com.bigfans.orderservice.api.clients", "dependency": ["com.bigfans.framework.utils", "com.bigfans.model.dto.order", "com.bigfans.orderservice.exception", "com.bigfans.orderservice.model", "com.bigfans.framework.web", "com.bigfans.orderservice.model.entity", "com.bigfans"], {"name": "com.bigfans.orderservice.api.interceptor", "dependency": ["com.bigfans.framework.annotations", "com.bigfans.framework.utils", "com.bigfans.framework", "com.bigfans.framework.web", "com.bigfans.framework.exception", "com.bigfans.framework.model", "com.bigfans"], {"name": "com.bigfans.orderservice.api", "dependency": ["com.bigfans.framework.annotations", "com.bigfans.framework.dao", "com.bigfans.orderservice.api.clients", "com.bigfans.orderservice.exception", "com.bigfans.orderservice.model", "com.bigfans.framework.web", "com.bigfans.orderservice.service", "com.bigfans.framework.kafka", "com.bigfans.framework.model", "com.bigfans.model.dto.order", "com.bigfans.orderservice.model.entity", "com.bigfans.orderservice.api.form"], {"name": "com.bigfans.orderservice", "dependency": ["com.bigfans.orderservice.exception", "com.bigfans.framework", "com.bigfans.framework.web", "com.bigfans.framework.exception", "com.bigfans.framework.kafka"], {"name": "com.bigfans.orderservice.config", "dependency": ["com.bigfans.framework.event", "com.bigfans.orderservice.api.interceptor", "com.bigfans.framework.dao", "com.bigfans.framework.redis", "com.bigfans.framework.kafka"], {"name": "com.bigfans.orderservice.dao", "dependency": ["com.bigfans.framework.dao", "com.bigfans.orderservice.model", "com.bigfans.orderservice.dao.impl", "dependency": ["com.bigfans.framework.dao", "com.bigfans.orderservice.model", "com.bigfans.orderservice.dao", "com.bigfans.framework.model", "com.bigfans.orderservice"]}]}
```

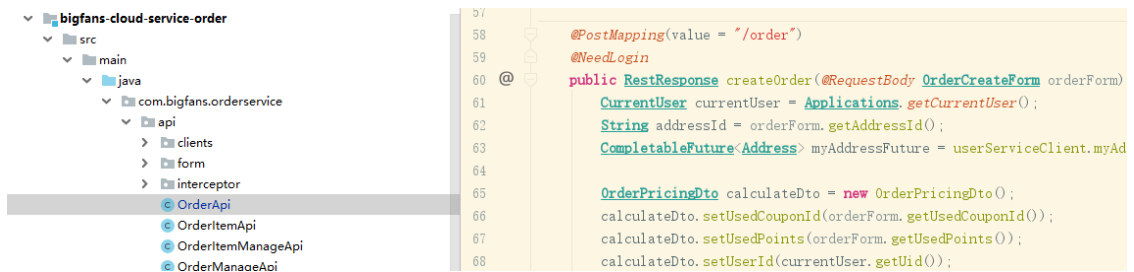
### 3. 加入到最终的 JSON 文件中，并格式化

```
"modules": [ // 代码仓库可能包含多个模块
{
    "path": "bigfans-cloud-service-order", // 模块目录相对于代码仓库根目录的路径
    "language": "Java", // Java
    "packages": [
        {
            "name": "com.bigfans.orderservice.api.clients",
            "dependency": [
                "com.bigfans.framework.utils",
                "com.bigfans.model.dto.order",
                "com.bigfans.orderservice.exception",
                "com.bigfans.orderservice.model",
                "com.bigfans.framework.web",
                "com.bigfans.orderservice.model.entity",
                "com.bigfans"
            ]
        },
        {
            "name": "com.bigfans.orderservice.api.interceptor",
            "dependency": [
                "com.bigfans.framework.annotations",
                "com.bigfans.framework.utils",
                "com.bigfans.framework",
                "com.bigfans.framework.web",
                "com.bigfans.framework.exception",
                "com.bigfans.framework.model",
                "com.bigfans"
            ]
        }
    ]
},
{
    "path": "bigfans-cloud-service-user", // 模块目录相对于代码仓库根目录的路径
    "language": "Java", // Java
    "packages": [
        {
            "name": "com.bigfans.userservice.api.clients",
            "dependency": [
                "com.bigfans.framework.utils",
                "com.bigfans.model.dto.order",
                "com.bigfans.orderservice.exception",
                "com.bigfans.orderservice.model",
                "com.bigfans.framework.web",
                "com.bigfans.orderservice.model.entity",
                "com.bigfans"
            ]
        },
        {
            "name": "com.bigfans.userservice.api.interceptor",
            "dependency": [
                "com.bigfans.framework.annotations",
                "com.bigfans.framework.utils",
                "com.bigfans.framework",
                "com.bigfans.framework.web",
                "com.bigfans.framework.exception",
                "com.bigfans.framework.model",
                "com.bigfans"
            ]
        }
    ]
},
{
    "path": "bigfans-cloud-service-common", // 模块目录相对于代码仓库根目录的路径
    "language": "Java", // Java
    "packages": [
        {
            "name": "com.bigfans.common",
            "dependency": [
                "com.bigfans.framework.utils",
                "com.bigfans.model.dto.order",
                "com.bigfans.orderservice.exception",
                "com.bigfans.orderservice.model",
                "com.bigfans.framework.web",
                "com.bigfans.orderservice.model.entity",
                "com.bigfans"
            ]
        },
        {
            "name": "com.bigfans.common.api.interceptor",
            "dependency": [
                "com.bigfans.framework.annotations",
                "com.bigfans.framework.utils",
                "com.bigfans.framework",
                "com.bigfans.framework.web",
                "com.bigfans.framework.exception",
                "com.bigfans.framework.model",
                "com.bigfans"
            ]
        }
    ]
},
{
    "path": "bigfans-cloud-service-framework", // 模块目录相对于代码仓库根目录的路径
    "language": "Java", // Java
    "packages": [
        {
            "name": "com.bigfans.framework",
            "dependency": [
                "com.bigfans.framework.annotations",
                "com.bigfans.framework.utils",
                "com.bigfans.framework",
                "com.bigfans.framework.web",
                "com.bigfans.framework.exception",
                "com.bigfans.framework.model",
                "com.bigfans"
            ]
        },
        {
            "name": "com.bigfans.framework.web",
            "dependency": [
                "com.bigfans.framework.annotations",
                "com.bigfans.framework.utils",
                "com.bigfans.framework",
                "com.bigfans.framework.web",
                "com.bigfans.framework.exception",
                "com.bigfans.framework.model",
                "com.bigfans"
            ]
        },
        {
            "name": "com.bigfans.framework.exception",
            "dependency": [
                "com.bigfans.framework.annotations",
                "com.bigfans.framework.utils",
                "com.bigfans.framework",
                "com.bigfans.framework.web",
                "com.bigfans.framework.exception",
                "com.bigfans.framework.model",
                "com.bigfans"
            ]
        },
        {
            "name": "com.bigfans.framework.model",
            "dependency": [
                "com.bigfans.framework.annotations",
                "com.bigfans.framework.utils",
                "com.bigfans.framework",
                "com.bigfans.framework.web",
                "com.bigfans.framework.exception",
                "com.bigfans.framework.model",
                "com.bigfans"
            ]
        },
        {
            "name": "com.bigfans.framework.dao",
            "dependency": [
                "com.bigfans.framework.annotations",
                "com.bigfans.framework.utils",
                "com.bigfans.framework",
                "com.bigfans.framework.web",
                "com.bigfans.framework.exception",
                "com.bigfans.framework.model",
                "com.bigfans"
            ]
        },
        {
            "name": "com.bigfans.framework.kafka",
            "dependency": [
                "com.bigfans.framework.annotations",
                "com.bigfans.framework.utils",
                "com.bigfans.framework",
                "com.bigfans.framework.web",
                "com.bigfans.framework.exception",
                "com.bigfans.framework.model",
                "com.bigfans"
            ]
        },
        {
            "name": "com.bigfans.framework.redis",
            "dependency": [
                "com.bigfans.framework.annotations",
                "com.bigfans.framework.utils",
                "com.bigfans.framework",
                "com.bigfans.framework.web",
                "com.bigfans.framework.exception",
                "com.bigfans.framework.model",
                "com.bigfans"
            ]
        }
    ]
},
{
    "path": "bigfans-cloud-service-annotation", // 模块目录相对于代码仓库根目录的路径
    "language": "Java", // Java
    "packages": [
        {
            "name": "com.bigfans.framework.annotations",
            "dependency": [
                "com.bigfans.framework.annotations",
                "com.bigfans.framework.utils",
                "com.bigfans.framework",
                "com.bigfans.framework.web",
                "com.bigfans.framework.exception",
                "com.bigfans.framework.model",
                "com.bigfans"
            ]
        }
    ]
}
]
```

## 微服务数据的采集：

### 方法1：

1. 查看项目目录，找到统一的 API 接口文件目录: xxx/api



## 2. 根据相关信息填写至json文件中

方法2:

查看 **swagger** 文档，进行填写