**实时同步脚本解决方案说明**

**Real-time Synchronization Document**

|  |  |
| --- | --- |
| **项目名称：** | **新官网、租客App、BS、房东PC、房东App** |
| **项目编号：** | cmd-20160720 |
| **作者：** | 雷传盛 |

**版权所有©上海墨果资产管理有限公司**

**Copyright ©Mogoroom Forture Management Co.,Ltd**.

目录

[1. 全局说明 3](#_Toc456878290)

[1.1 系统关系图 3](#_Toc456878291)

[1.1.1 关系总览 3](#_Toc456878292)

[1.2 部署说明 4](#_Toc456878293)

[1.2.1 环境部署 4](#_Toc456878294)

[1.2.2 脚本部署说明 5](#_Toc456878295)

[2. 同步消息说明 5](#_Toc456878296)

[2.1 房源 5](#_Toc456878297)

[2.1.1 消息生产端 5](#_Toc456878298)

[2.1.2 消息消费端 5](#_Toc456878299)

[2.1.3 代码导读 6](#_Toc456878300)

[2.2 房间列表显示价格 6](#_Toc456878301)

[2.2.1 消息生产端 6](#_Toc456878302)

[2.2.2 消息消费端 6](#_Toc456878303)

[2.2.3 代码导读 7](#_Toc456878304)

[2.3 小区 7](#_Toc456878305)

[2.3.1 消息生产端 7](#_Toc456878306)

[2.3.2 消息消费端 7](#_Toc456878307)

[2.3.3 代码导读 7](#_Toc456878308)

[2.4 优选品牌 7](#_Toc456878309)

[2.4.1 消息生产端 7](#_Toc456878310)

[2.4.2 消息消费端 7](#_Toc456878311)

[2.4.3 代码导读 8](#_Toc456878312)

[2.5 普通品牌 8](#_Toc456878313)

[2.5.1 消息生产端 8](#_Toc456878314)

[2.5.2 消息消费端 8](#_Toc456878315)

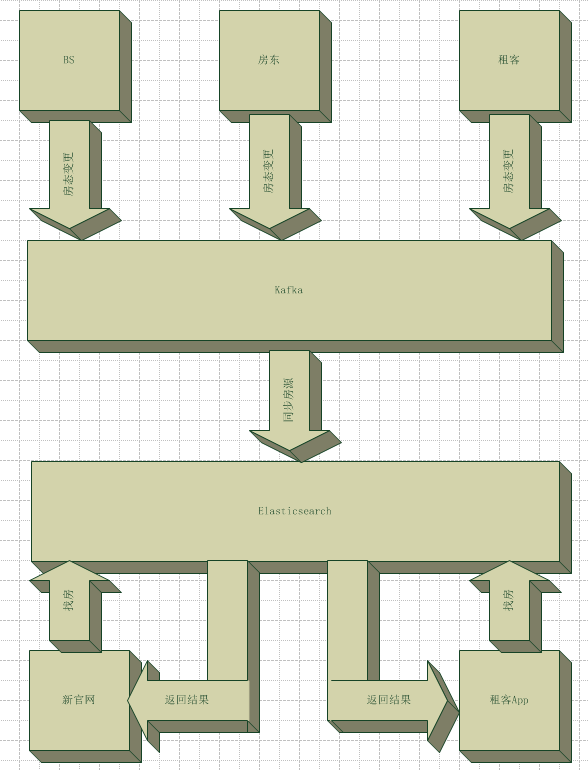
[2.5.3 代码导读 9](#_Toc456878316)

[3. 业务对接人 9](#_Toc456878317)

# 全局说明

## 系统关系图

### 关系总览



## 部署说明

### 环境部署

以下是开发环境的部署说明

1. kafka，

安装地址：/usr/local/data/kafka\_2.9.2-0.8.2.2

broke-list：192.168.30.245:9092

topic：bsmogo

group：bs\_lcs

1. elasticsearch

安装地址：/usr/local/data/elasticsearch-1.7.3

http url：http://192.168.30.245:9200

ip/tcp：192.168.30.245:9300

### 脚本部署说明

1. 脚本地址：/data/script/sync.sh

注：无论是哪个环境该地址不变

1. 该脚本依赖bs的jar包

# 同步消息说明

## 房源

### 消息生产端

凡是涉及到房源数据变更的操作(新增、房态变更、物理删除、逻辑删除等)都需要向kafka发送消息，消息格式：

room-${roomId}

### 消息消费端

1. 检查${roomId}是否为正整型数字，若不是，抛出异常信息：Error when parseInt for keyId[${roomId}]，然后继续接收下一条消息，否则执行下一步；
2. 根据${roomId}获取其对应的房源信息，若找不到，执行以下操作：
3. 抛出异常信息：Error when findRoomById for roomId[${roomId}]；
4. 删除搜索引擎上对应的房源记录(包括rooms和keywords上的索引相关doc)；

若找得到房源信息，则执行下一步；

1. 检查房源是否有效，出现以下三种之一的，即被视为无效房源：
2. 有效状态不为1-有效，即status!=1；
3. 出租状态不为1-未出租，即rentStatus!=1；
4. 上架状态不为1-已上架，即onlineStatus!=1；

若房源无效，执行以下操作：

1. 删除搜索引擎上对应的房源记录(包括rooms和keywords上的索引相关doc)；
2. 抛出警告信息：Invalid room data for roomId[${roomId}], status[${status}], renterStatus[${rentStatus}], onlineStatus[${onlineStatus}]；

若房源有效，执行下一步；

1. 同步小区数据；

根据房间对应的小区Id，获取其对应的小区信息，且将其同步至搜索引擎对应的小区信息中去；

1. 同步房源评分；

根据房间Id，调用房源评分接口，重新计算该房间对应的评分；

1. 同步房源在keywords索引中的doc；
2. 同步小区小面所有的房源，且删除冗余房源，具体如下：
3. 若当前房源是集中式公寓，则获取小区Id下对应的所有的房间信息，然后将其更新至搜索引擎，且消除搜索引擎上冗余的房源doc；
4. 若当前房源是分散式公寓，则将当前房间Id对应的房源信息同步更新至搜索引擎；

### 代码导读

代码入口：com.mogoroom.bs.job.SyncRoomDataJob

消息格式：room-${roomId}

## 房间列表显示价格

### 消息生产端

价格体系中付款方式或付款金额发生任何变化时，都需要向kafka发送消息，消息格式：roomPrice-${roomId}

### 消息消费端

1. 检查${roomId}是否为正整型数字，若不是，抛出异常信息：Error when parseInt for keyId[${roomId}]，然后继续接收下一条消息，否则执行下一步；
2. 根据${roomId}获取其对应的房源信息，若找不到，则爬出异常信息：Not found room for keyId[${roomId}]，然后继续接收下一条消息，否则执行下一步；
3. 计算房间对应的显示价格：
4. 从价格体系统中获取当前房间Id对应的价格信息集(包括付款方式和金额)；
5. 循环遍历该价格信息集，获取其最低价格作为当前房间显示价格，若当前最低价格为蘑菇宝的付款方式且房东对应的蘑菇宝额度为0，则剔除该最低价格，再从该价格以外的集合中获取最低价格；
6. 更新房间对应的显示价格(showPrice)值，且发送room-${roomId}至kafka。

### 代码导读

代码入口：com.mogoroom.bs.job.SyncRoomDataJob

消息格式：room-${roomId}

## 小区

### 消息生产端

凡是小区信息有任何变化(包括增、删、改)，都需要向kafka发送消息，消息格式：community-${communityId}

### 消息消费端

1. 检查${ communityId }是否为正整型数字，若不是，抛出异常信息：Error when parseInt for keyId[${communityId }]，然后继续接收下一条消息，否则执行下一步；
2. 通过${communityId}获取其对应的小区信息，若不存在，则抛出异常信息：Not found community for keyId[${communityId }]，然后继续接收下一条消息，否则执行下一步；
3. 将小区信息同步至搜索引擎的keywords索引。

### 代码导读

代码入口：com.mogoroom.bs.job.SyncRoomDataJob

消息格式：community-${communityId}

## 优选品牌

### 消息生产端

凡是涉及到优选品牌的变化(增、删、改等)都需要向kafka发送消息，消息格式：brand-${brandId}

### 消息消费端

1. 检查${brandId}是否为正整型数字，若不是，抛出异常信息：Error when parseInt for keyId[${brandId}]，然后继续接收下一条消息，否则执行下一步；
2. 根据品牌Id获取其对应的品牌信息，若品牌信息不存在，则执行以下操作：
3. 抛出异常信息：Invalid preferredBrand status for keyId[${brandId}]；
4. 删除品牌在搜索引擎keywords、rooms中相关的索引，且删除其在缓存中的内容；
5. 继续接收下一条消息，否则执行下一步；
6. 检查品牌是否有效，发生以下情况之一，则表示品牌无效：
7. 品牌状态为0-删除，即status=0；
8. 上下线状态为0-下线，即isOnline=0；

若品牌无效，则执行以下操作：

1. 抛出异常信息：Invalid preferredBrand status for keyId[${brandId}]
2. 删除品牌在搜索引擎keywords、rooms中相关的索引，且删除其在缓存中的内容；
3. 继续接收下一条消息

若品牌有效，则执行下一步；

1. 同步优选品牌信息至搜索引擎，即：
2. 同步更新优选品牌信息至搜索引擎的keywords索引相关的doc；
3. 同步更新优选品牌信息至搜索引擎的rooms索引相关的doc；

### 代码导读

代码入口：com.mogoroom.bs.job.SyncRoomDataJob

消息格式：brand-${brandId}

## 普通品牌

### 消息生产端

凡是房东涉及到品牌名称(brandName)的变更，都需要向kafka发送消息，消息格式：landlord-${landlordId}

### 消息消费端

1. 检查${landlordId}是否为正整型数字，若不是，抛出异常信息：Error when parseInt for keyId[${brandId }]，然后继续接收下一条消息，否则执行下一步；
2. 检查房东是否存在，即根据房东Id获取其对应的房东信息，若不存在，则执行以下操作：
3. 抛出警告信息：Not found commonBrand for landlordId[${landlordId}]；
4. 删除该房东Id在搜索引擎keywords索引上对应的doc；
5. 继续接收下一条消息；

若品牌有效，则执行下一步；

1. 检查房东是否存在品牌信息，即检查品牌名称是否为空，若为空，则执行以下操作：
2. 抛出警告信息：Not found commonBrand for brandName is null and landlordId[${landlordId}]
3. 删除该房东Id在搜索引擎keywords索引上对应的doc；
4. 继续接收下一条消息；

若房东存在品牌信息，则执行下一步；

1. 检查房东状态是否有效，若无效，则执行以下操作：
2. 抛出警告信息：invalid commonBrand for landlordId[${landlordId}]
3. 删除该房东Id在搜索引擎keywords索引上对应的doc；
4. 继续接收下一条消息；

若房东状态有效，则执行下一步；

1. 同步普通品牌信息至搜索引擎的keywords索引对应的doc。

### 代码导读

代码入口：com.mogoroom.bs.job.SyncRoomDataJob

消息格式：landlord-${landlordId}

# 业务对接人

|  |  |  |
| --- | --- | --- |
| 业务名称 | 对接人 | 部门 |
| 价格体系 | 蒙先铭 | 架构组 |
| 找房 | 彭晓春、周云 | EQ组 |