**B2B系统数据层向中台迁移方案**

徐良永 2018.5.2

# 第一章 数据层平滑过渡方案

先说结论：数据做从b2b到中台做单向同步。鉴于读多写少，读操作从中台，写操作仍然在b2b。待读操作改造完毕，再改造写操作

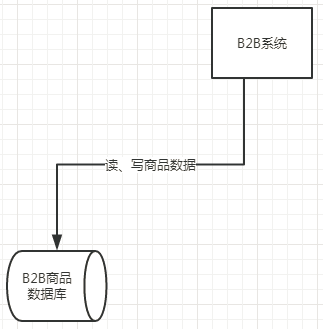
以商品库为例

B2B系统对商品库的操作分为**读操作**和**写操作**

## 1.1 目前架构

如下图

读写都是在B2B数据库，但引入中台后B2B商品数据库会被中台数据库代替，这对代码的影响，尤其是对关联SQL的影响是巨大的；剧粗略估计涉及到商品的sql查询有900+个改造 点。一次性改造完这900个点再上线一是周期长二是风险大。 所以说直接用中台商品库替代B2B商品库是不可行的，我们要使用平滑过渡方案



## 1.2引入中台后架构

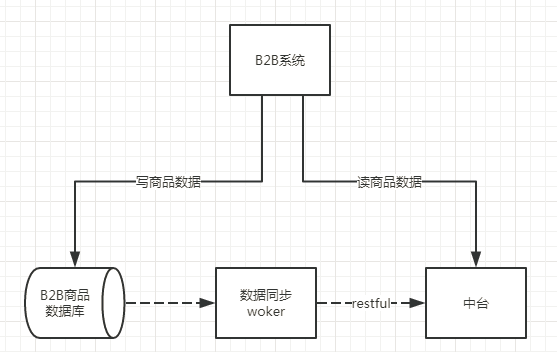
如下图

B2B系统写操作仍然写入B2B数据库，然后数据实时同步到中台数据库

B2B系统读操作则通过中台接口直接获取

这就是平滑过渡方案，其优点如下

* 不需要一次性改完900多个点再上线
* 可以修改一个或多个商品涉及点，在我们可控的周期内上线，循环迭代
* 项目风险大大降低，几乎可做到0风险



### 1.2.1数据同步worker

工作方式有两种

* 同步执行
* 异步执行（MQ）

可能遇到的问题

* 同步执行时如果rest接口超时如何处理？
* 如果有多个**数据同步worker**，如何保证数据的先后顺序？
* 异步执行同样会遇到数据写入的先后顺序问题 ？
* 最终要避免数据不一致的发生

# 第二章 应用层的平滑过渡方案

先说结论：应用的web层不动，即界面不动。Service层变成数据聚合层，其数据来源为中台rest接口和新建的b2b微服务rest接口。

## 目前架构

如下图 b2b是传统巨石型应用

其缺点不再赘述

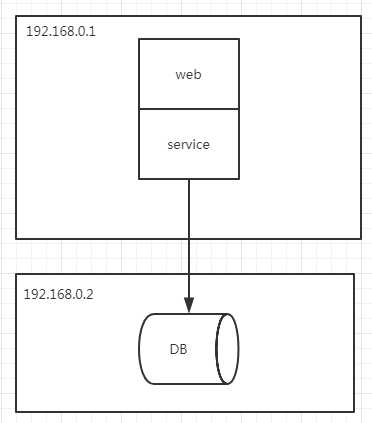


图1

## 2.2 引入中台后架构

如下图 ，Web层即前端页面，不做改动

图1中的**Service层**变成了图2中的**聚合service层。**

### 2.2.1什么是聚合service？

在第一章中已经完成了数据层单向同步，所以我们读服务都可以走中台接口。

这样我们的数据源就有两个：

* B2B数据源
* 中台数据源

对应的服务层也是两个

* B2B服务层 （jvm）
* 中台服务层 （http）

**聚合service层就是把B2B服务层和中台服务层聚合起来**

由于我们的服务读多写少，所以先把读服务平滑过渡到中台；

待读操作过渡完成后，再进行写接口的平滑过渡

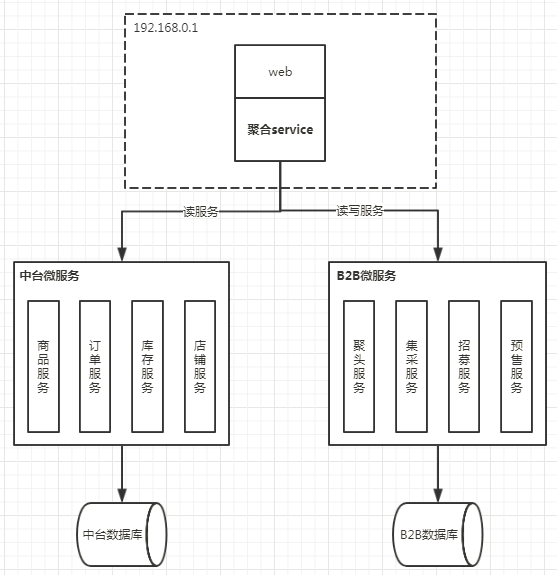


图2

# 总结

如图3，把第一章和第二章的方案整合起来就是B2B项目对接中台的最终方案

整个对接方案的难点有3个：

1. 数据同步worker高可用
2. 数据同步的速度。取决于两个方面：解析binlog数据和写入中台的速度
3. 同步失败的数据处理策略（可采用补偿策略）

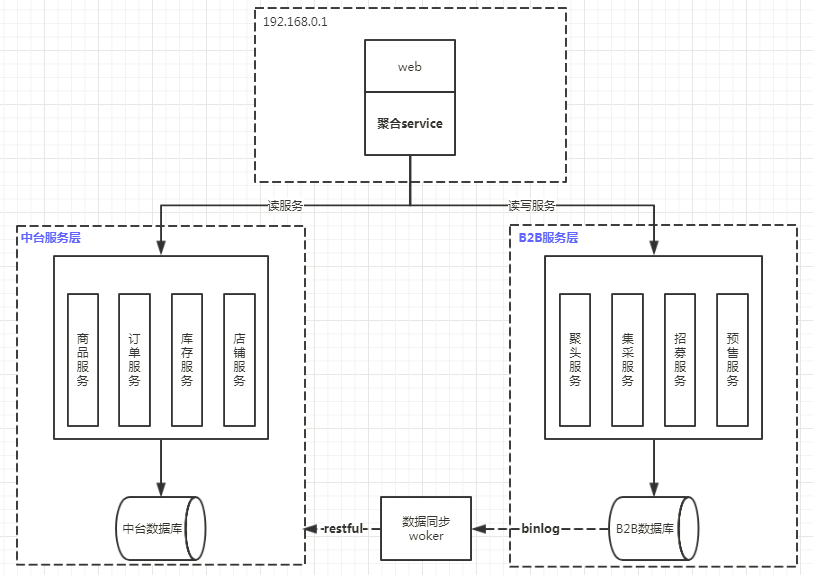


图3