签到表分表

# 分表使用工具

Spring-boot

数据库分表工具：zebra

RPC工具：pigeon

注册中心：zookeeper

# 签到表Dao层接口统计

getCustomerSignInSimpleDetailByMonth

getStoreSigninCountByDays

findAllSigninHistoryPLus

findAllSigninHistory

findSignStatises

findMonthSigninAcount

findRecentSigninAcount

findPeriodFirstSignDay

findCustomerBySignCountAndAddress

findNotSigninDays

findHasSign

findHasSignNotSync

findLastSignDay

countSignCustomers

countSignDays

Mysql 函数：getMyisamID

update

insert

batchInsert

# 签到表涉及业务

## 顾客签到（新/老接口）

接口地址：

1. /scanManager/signin
2. /v1/scanManager/signin

Dao调用：findAllSigninHistory，findHasSign，findHasSignNotSync，countSignCustomers，addSignIn（老接口），insert（老接口），update

## 顾客签到记录查询（新/老接口）

接口地址：

1. /customerSign/signinHistory
2. /1/statistics/signin/history/get
3. /2.3/customerSign/signinHistory

Dao调用：findAllSigninHistory，findAllSigninHistoryPLus

## 礼物领取（新/老接口）

接口地址：

1. /scanManager/collectGiftByCard
2. /scanManager/v2/collectGiftByCard

Dao调用：findAllSigninHistory

## 获取连续签到记录

接口地址：/1/statistics/signin/continue/get

Dao调用：findAllSigninHistory

## 获取签到排名

接口地址：/1/statistics/signin/rank/get

Dao调用：findAllSigninHistory，getCustomerSignInSimpleDetailByMonth，findSignStatises

## 提醒任务：客户住址

客户最近30日到店率超过60%，请记录客户住址信息

Dao调用： findCustomerBySignCountAndAddress

## mq 新增签到记录

类：SigninConsumer

Dao调用：batchInsert，getMyisamID

## 按月获取顾客签到记录

接口地址：/ customerSignManager/getCustomerSignInByMonth

Dao调用：getCustomerSignInSimpleDetailByMonth

## 获取签到排名，以及百分比

接口地址：/1/statistics/signin/rank/get

Dao调用：getCustomerSignInSimpleDetailByMonth

## Xxl-job 存储签到次数crm\_store\_signin\_count

storeSigninJobHandler

syncStoresigninJobHandler

Dao调用：getStoreSigninCountByDays

## 查询当日未签到用户

接口地址：/ customerSignManager/findNotSignedCustomerToday

Dao调用：findMonthSigninAcount（被注释），findNotSigninDays（被注释）

## Xxl-job 客户挽回任务

customerRetrieveJobHandler

Dao调用：findRecentSigninAcount，findNotSigninDays

## 签到率天数基数

无组件调用

Dao调用：findPeriodFirstSignDay

## 请假

接口地址：/1/customer/signin/addSigninLeave

Dao调用：findHasSign，findLastSignDay，getMyisamID，insert，update

## 补签

接口地址：/1/ /customer/signin/addSigninLate

Dao调用：findHasSign，insert，update

## 礼物领取

接口地址：/scanManager/v2/collectGiftByCard

Dao调用：countSignDays

# 维度选取

选择签到表的CUSTOMER\_ID（顾客id）作为维度。

在已有sql语句中，顾客id是在CRUD操作中作为查询目标属性和条件属性中出现次数最多的字段，这样能满足zebra要求中的维度必须出现在sql语句中的限制；同时很多查询，检索的sql都是基于同一顾客进行的，方便业务逻辑变更；此外，顾客id在签到表中与主键直接相关，一条签到记录必然有一个顾客id，根据顾客id能更快的定位签到记录所在范围。

# 路由规则

通过维度和路由规则定位到对应表的过程，在针对签到表的CRUD操作中，通过维度CUSTOMER\_ID字段及其值，经路由规则计算后能定位到对应表。

## 数据库的路由规则

dbRule="#CUSTOMER\_ID# \* 0"

单库，无需定义规则，指出 [维度] 即可。

## 数据库名后缀

dbIndexes="db"

与zebra配置中的ShardDataSource数据源的key相同即可。

## 表的路由规则

参考jdk8 HashMap 原理，路由规则生成思路如下：

1. 主表：sign\_in
2. 维度：key
3. 表下标偏移：offset
4. 表数量：len
5. 散列值：hash = crc32(key)
6. 扰动函数：ha = (hash ^ (hash >>> 16))
7. 取模运算：in = (len - 1) & ha
8. 映射到的表下标：index = offset + in
9. tbRule: offset + ((len - 1) & (crc32(key) ^ (crc32(key) >>> 16)))

如：offset = 3,len = 2,key 为 #customer\_id#

tbRule = "3 + ((2 - 1) &amp; (crc32(#customer\_id#) ^ ((crc32(#customer\_id#) &gt;&gt;&gt; 16)))"

在表sign\_in3，sign\_in4中路由

注意：

* tbRule 格式必须严格相同
* len只能为2的整次幂，否则路由分布不均匀，表现形式之一为插入大量数据，路由到的各个表中新增数据量分别不均匀。
* offset 不能小于 0，即最小为0

说明：

扰动函数：使碰撞（hash冲突）更平均，即 hash 映射到的 index 在可选 index 中被选中的概率趋于相同。

取模运算：高位全部归零，只保留低位，用作下标访问

表数量取2的整数次幂：长度减1，便于取模。

## 表名后缀

tbSuffix="everydb:[0,999]"

everydb:[a,b]：a始终为0，b不小于实际表的最大index即可。

## 主维度

isMaster="true"

只有一个维度

# 代码如何修改

## 原代码修改

通过url请求和dao层调用定位代码，将service层代码修改为RPC调用。

## 抽取RPC服务

### 签到服务

#### Mq 正常签到

将签到的rocketMQ的Consumer转移到签到服务上，将签到服务关联到与签到消息的Producer相同的Name Server。

#### 补签

#### 请假

### 签到历史记录查询服务

#### 按月获取签到记录

#### 连续签到记录查询

### 签到排名服务

### 当日用户签到状态查询服务

#### 当日未签到用户查询

### 签到相关提醒服务

### Xxl-job 相关服务

#### 客户挽回

### 最大连续签到次数查询服务

# 数据过渡（迁移）

分表后的表结构仍采用之前的数据结构，需要老数据迁移到签到服务下的数据库中，按如下步骤进行数据迁移：

1. 确认数据迁移范围：对老数据进行统计，确认需要迁移的数据量，即所有签到数据。
2. 评估数据并准备空间：确认数据量后需要确认迁移到的签到服务中需要准备空间（分表的数量，表的数据容量），在此之前需要确认数据增长速率，表容量。
3. 迁移方式：直接复制数据，md5校验数据完整性。
4. 原系统中移除数据 ：删除签到数据和表。
5. 原系统业务逻辑变更：统计原系统中涉及签到表的业务，替换为RPC调用的方式。
6. 对新的业务逻辑进行测试。

# 几种查询方式的开销

# 数据支撑预测

# 附

## 签到表结构

ID：签到记录 id

CUSTOMER\_ID：签到顾客 id

DATE：签到日期

CURRENT\_SIGN\_STORE\_ID：当前签到门店ID

TYPE：签到类型

CREATE\_EID：补签或者请假时候的操作人

CREATE\_DATE：记录生成时间

## 签到类型

正常签到，补签，请假

## 瓶颈

写操作：签到时大量签到数据，通过rocketMQ将签到数据批量插入签到表中可以缓解。

读操作：业务中有大量的读操作，如统计次数，基于签到记录检索的业务逻辑。

## 参考文章

JDK 源码中 HashMap 的 hash 方法原理是什么：

<https://www.zhihu.com/question/20733617>

java 集合 3 - HashMap

<https://blog.csdn.net/aimeimeits/article/details/79016986>

数据迁移测试实施方案

<https://www.jianshu.com/p/9f6253e6fcc3>