Apache HBase at DiDi

Kang Yuan

滴滴

滴滴一下 美好出行

Agenda

1. About Us

2. DiDi HBase Platform

3. Application and Solution

4. Challenges and future

About Us

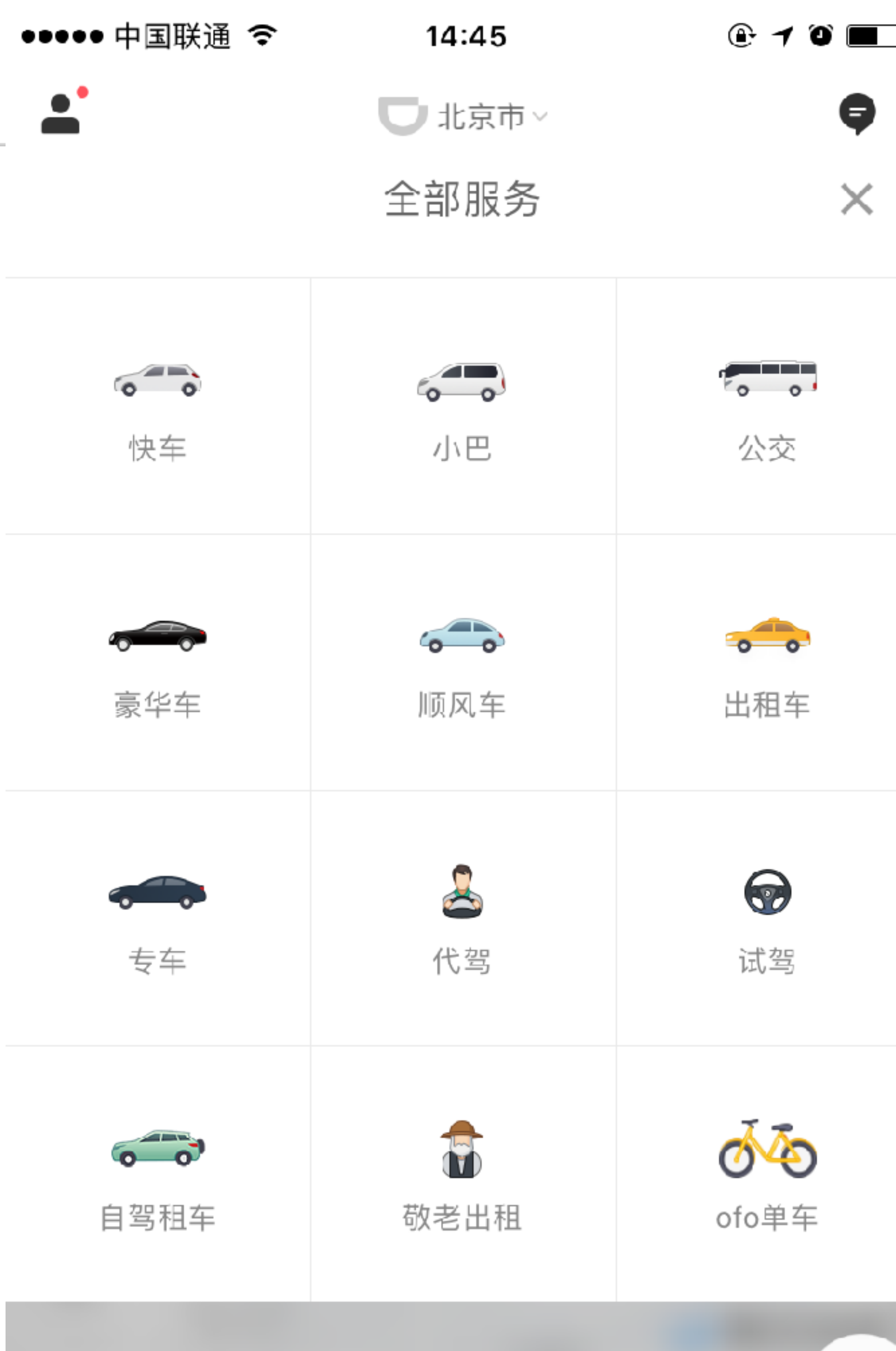
滴滴
滴滴一下 美好出行

- DiDi
- In <Silicon valley>

About Us

- **DiDi**
- the world’s leading mobile transportation platform
- 20 million rides on a daily basis

Express	Mini Bus	Bus
Luxury Car	Car pool	Taxi
Premier	Designated driver	Trial run
Rental Car	Taxi For Aged	ofo sharing Bike



About Us

- **Mission**

  To Redefine the Future of Mobility

- **Vision**

  To become a global leader in smart transportation and automotive technology, the world’s largest operator of vehicle networks and a global leader in smart transportation systems

About Us

- **HBase Team：4 Developers**

- Kang Yuan
- Yang Li
- Hanzhi Zhang
- Jingyi Yao

- Attached to BigData Architecture Department

- Cooperate with Hadoop/Hive/Spark/Flink/Druid Team closely

DiDi HBase Platform

- **Cluster(3)**
  - Storage Cluster – location A
  - Compute Cluster – location B
  - Storage Cluster – location B

  - Location A: the same place with the hadoop Cluster
  - Location B: for online business or streaming
- **Application(50+ business and 160+ tables)**
  - Batching Job result storage
  - Online writing/reading
  - Persistence for Streaming Jobs
- **99.95% available**

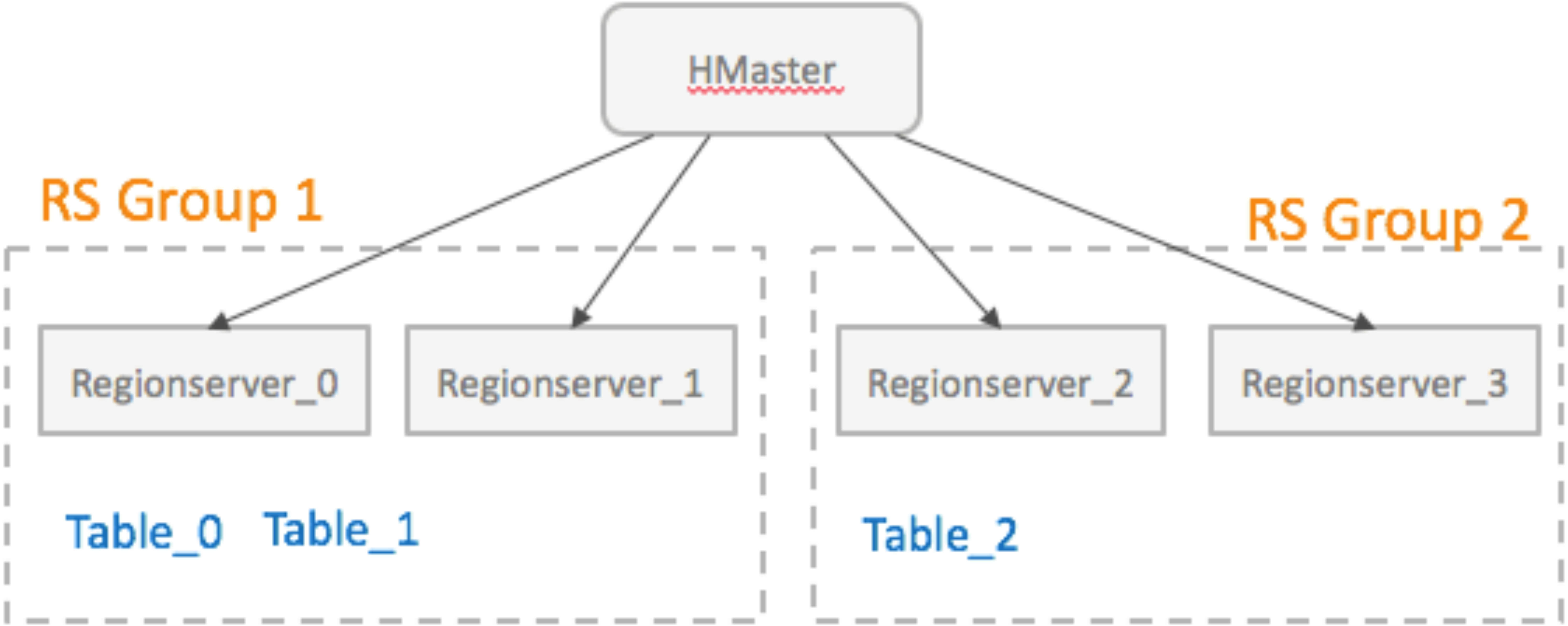
DiDi HBase Platform

- **HBase Version**
  - Based on 0.98.21
  - Region Group patch HBASE-6721
  - Thrift2 patch

- **Multi-tenant Problem**
  - A bad table can put down a cluster
  - We don’t know who the tables belong to

DiDi HBase Platform

- Region Group



DiDi HBase Platform

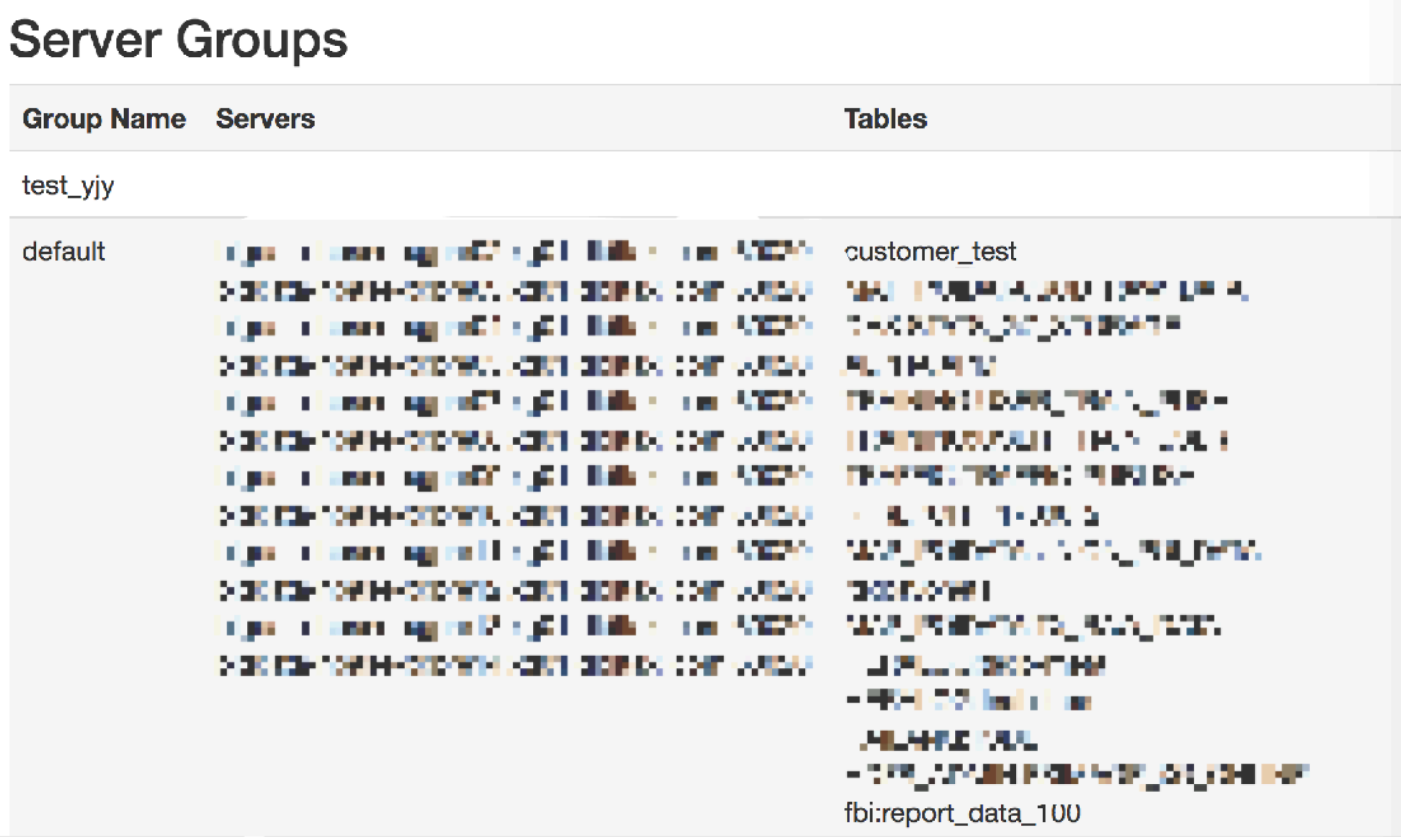
- **Region Group**

  - Isolate important use cases from others
  - Easy to manage(web ui, user group, operation tools)
  - Elastic to assign resources
  - Different Configuration in one Cluster(for different machine types, business, testing etc)
  - Easy to compute the cost of the business
  - Easy to upgrade the regionserver in one Group before do it in the whole cluster

  - Cost(share pool) = TableSize*x x = cost/GB
  - Cost(specific group) = Rscount*y y = cost/RS

DiDi HBase Platform

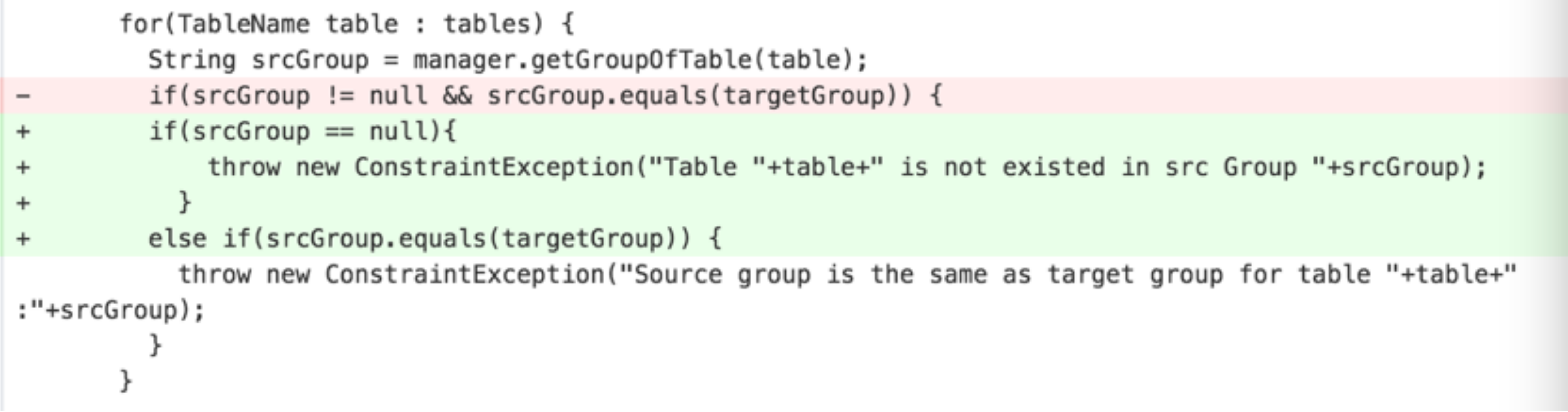
- **Improvement**
  - Web UI to show group
  - MoveTables bug fix
  - CreateTableHandler fix



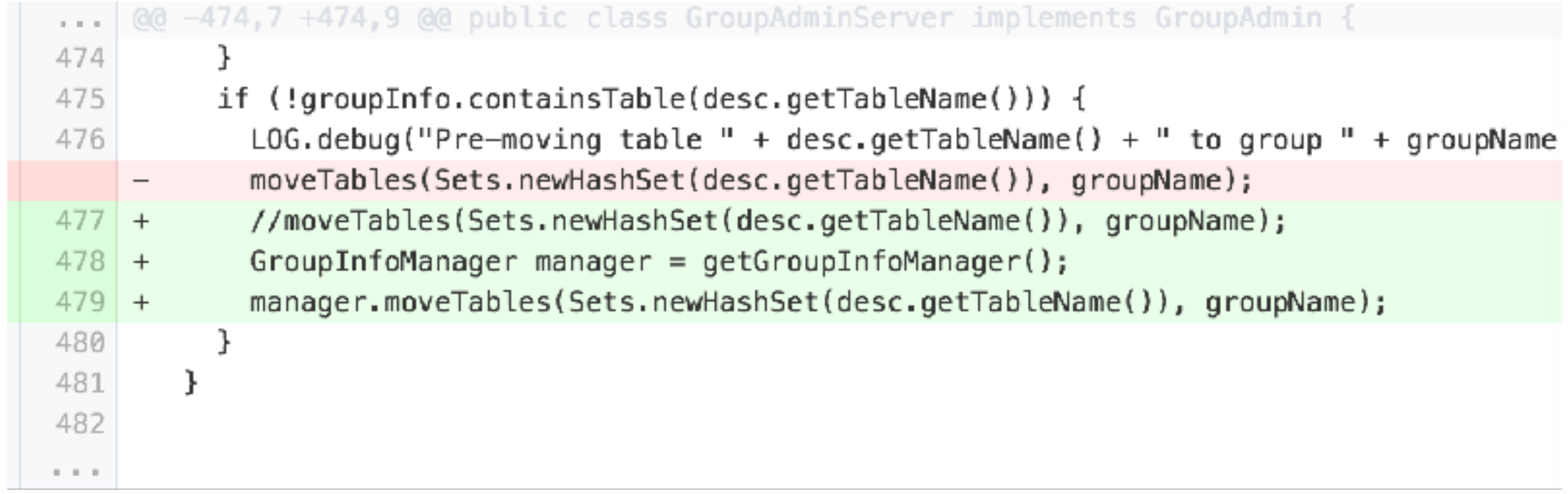
Server Groups

Group Name	Servers	Tables
test_yjy		
default	[redacted]	customer_test [redacted] fbi:report_data_100

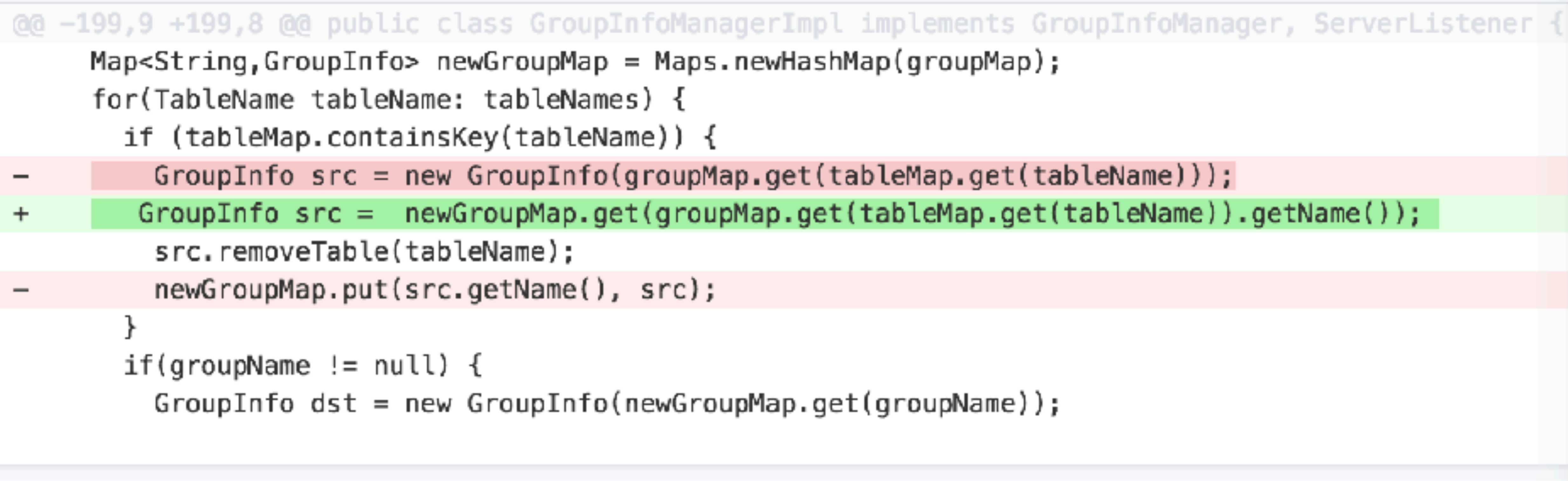
DiDi HBase Platform



```
      for(TableName table : tables) {
        String srcGroup = manager.getGroupOfTable(table);
-       if(srcGroup != null && srcGroup.equals(targetGroup)) {
+       if(srcGroup == null){
+           throw new ConstraintException("Table "+table+" is not existed in src Group "+srcGroup);
+         }
+       else if(srcGroup.equals(targetGroup)) {
          throw new ConstraintException("Source group is the same as target group for table "+table+"
:"+srcGroup);
        }
      }
```



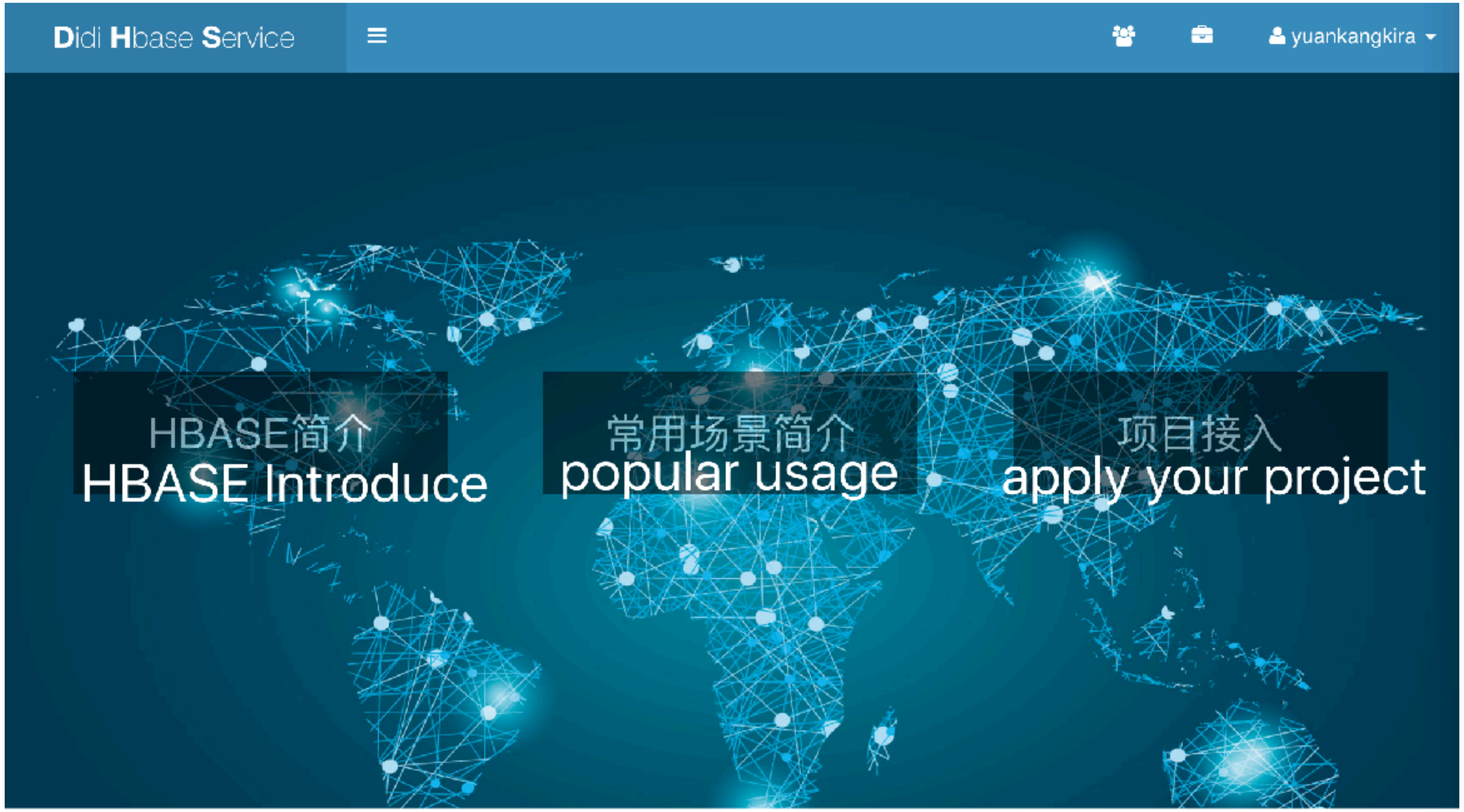
```
@@ -474,7 +474,9 @@ public class GroupAdminServer implements GroupAdmin {
        }
        if (!groupInfo.containsTable(desc.getTableName())) {
          LOG.debug("Pre-moving table " + desc.getTableName() + " to group " + groupName
-         moveTables(Sets.newHashSet(desc.getTableName()), groupName);
+         //moveTables(Sets.newHashSet(desc.getTableName()), groupName);
+         GroupInfoManager manager = getGroupInfoManager();
+         manager.moveTables(Sets.newHashSet(desc.getTableName()), groupName);
        }
      }
```



```
@@ -199,9 +199,8 @@ public class GroupInfoManagerImpl implements GroupInfoManager, ServerListener {
      Map<String,GroupInfo> newGroupMap = Maps.newHashMap(groupMap);
      for(TableName tableName: tableNames) {
        if (tableMap.containsKey(tableName)) {
-         GroupInfo src = new GroupInfo(groupMap.get(tableMap.get(tableName)));
+        GroupInfo src =  newGroupMap.get(groupMap.get(tableMap.get(tableName)).getName());
          src.removeTable(tableName);
-         newGroupMap.put(src.getName(), src);
        }
        if(groupName != null) {
          GroupInfo dst = new GroupInfo(newGroupMap.get(groupName));
```

DiDi HBase Platform

- **DiDi HBase Service**



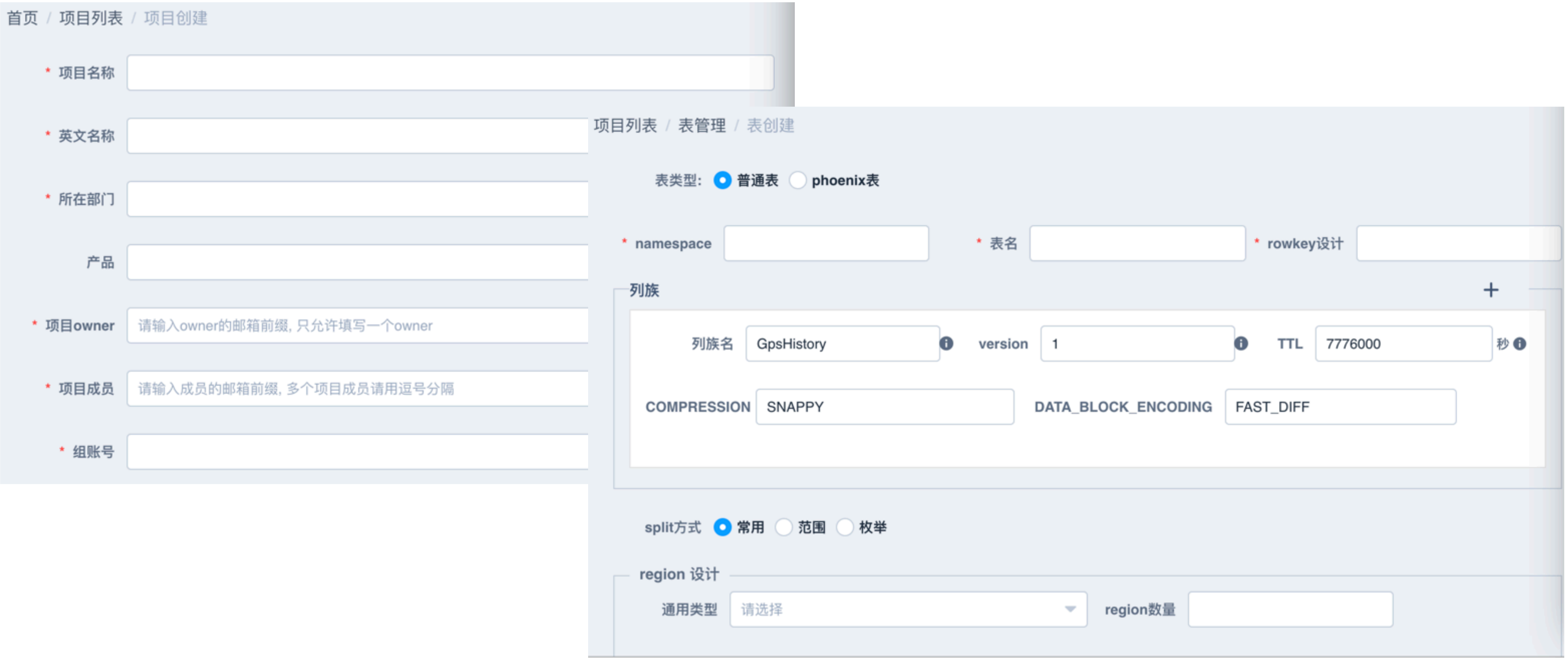
HBASE Introduce

popular usage

apply your project

DiDi HBase Platform

Create Project and Tables



首页 / 项目列表 / 项目创建

- * 项目名称
- * 英文名称
- * 所在部门
- 产品
- * 项目owner: 请输入owner的邮箱前缀，只允许填写一个owner
- * 项目成员: 请输入成员的邮箱前缀，多个项目成员请用逗号分隔
- * 组账号

项目列表 / 表管理 / 表创建

表类型: ● 普通表 ○ phoenix表

* namespace　* 表名　* rowkey设计

列族 +

列族名: GpsHistory　version: 1　TTL: 7776000 秒

COMPRESSION: SNAPPY　DATA_BLOCK_ENCODING: FAST_DIFF

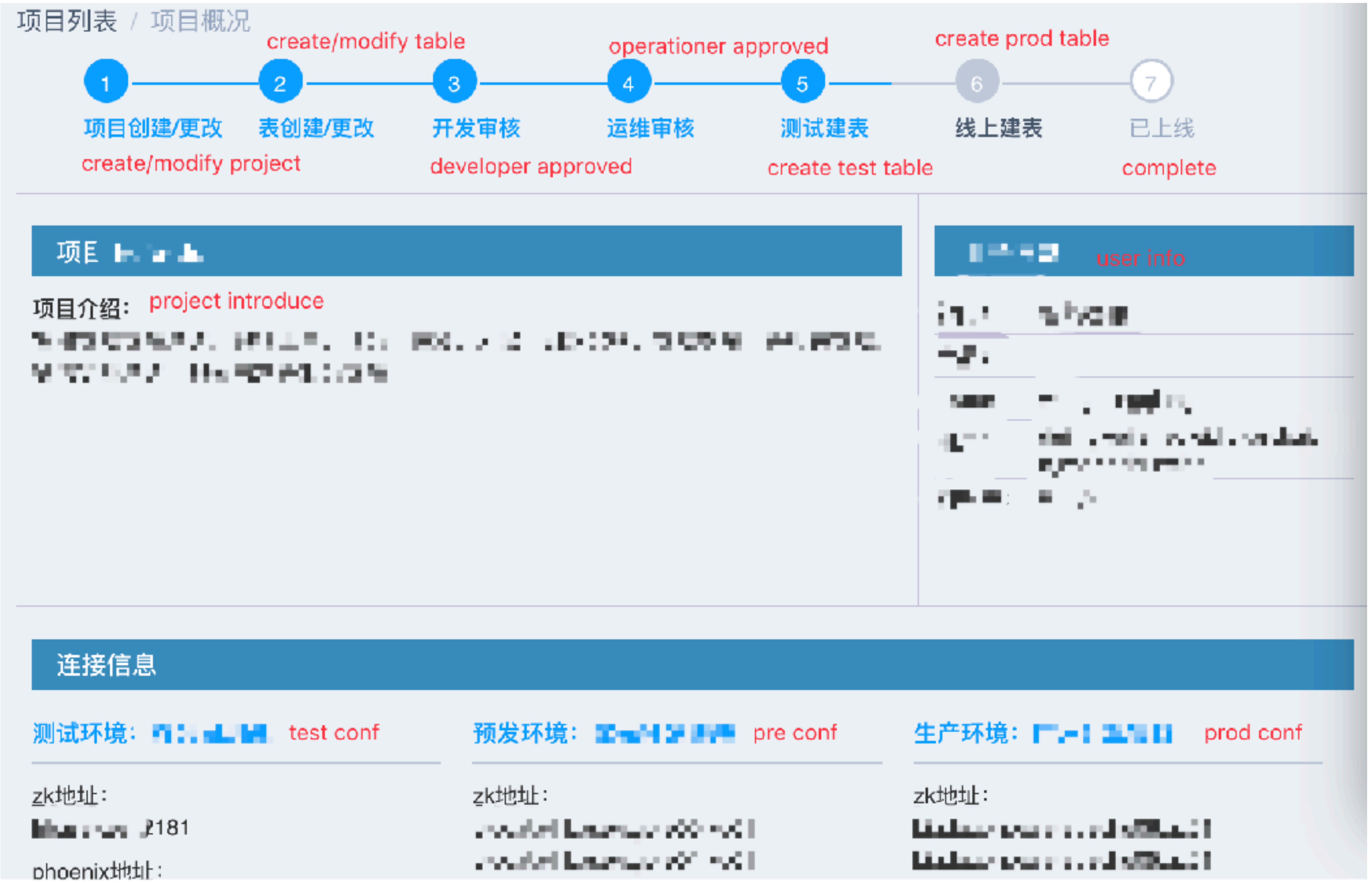
split方式 ● 常用 ○ 范围 ○ 枚举

region 设计

通用类型: 请选择　region数量

DiDi HBase Platform

workflow



项目列表 / 项目概况

1. 项目创建/更改 create/modify project
2. 表创建/更改 create/modify table
3. 开发审核 developer approved
4. 运维审核 operationer approved
5. 测试建表 create test table
6. 线上建表 create prod table
7. 已上线 complete

项E [redacted]

项目介绍： project introduce

[redacted]

[redacted] user info

[redacted]

连接信息

测试环境：[redacted] test conf

zk地址：

[redacted]2181

phoenix地址：

预发环境：[redacted] pre conf

zk地址：

[redacted]

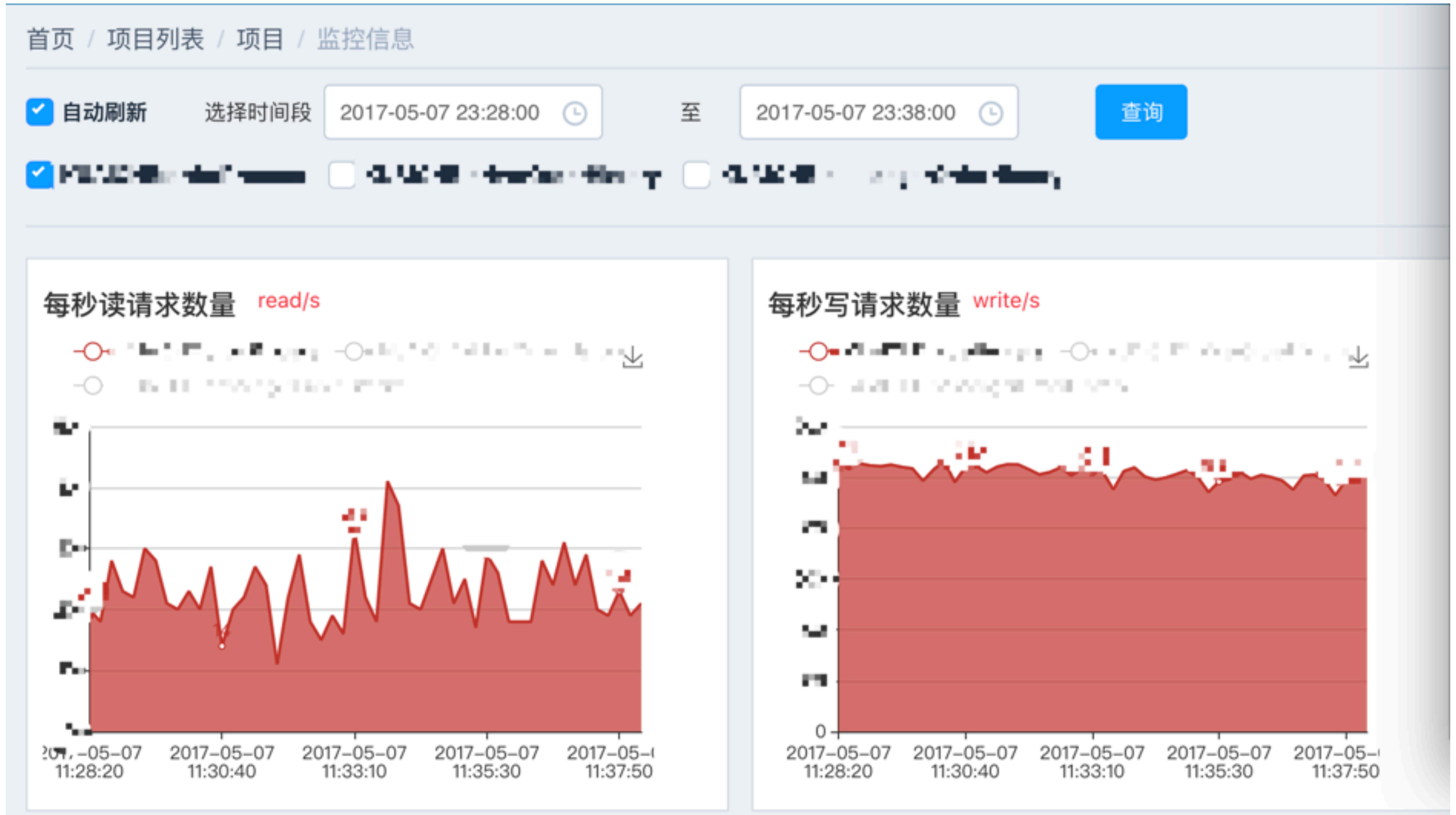
生产环境：[redacted] prod conf

zk地址：

[redacted]

DiDi HBase Platform

Monitor your tables



首页 / 项目列表 / 项目 / 监控信息

☑ 自动刷新　选择时间段 2017-05-07 23:28:00 至 2017-05-07 23:38:00 查询

☑ [redacted] □ [redacted] □ [redacted]

每秒读请求数量 read/s

每秒写请求数量 write/s

Get your bill, user must care for their cost

DiDi HBase Platform

- **Phoenix**
- **Advantage**
  - Easy to use for RDBMS User(jdbc、sql)
  - Auto salting table for performance and hot spot avoiding
  - Like a Big Mysql(One sentence to explain to our users)

- **Disadvantage**
  - Some bug like ordering vector item
  - Unstable statistic info caching
  - No good in Join case
  - So many other hotter system :Presto/Impala/Spark SQL/Kylin

- **Successful Use**
  - Row Timestamp
  - Multidimensional Table Schema

DiDi HBase Platform

- **Phoenix(more customers recently)**
- **Row Timestamp for metrics**
  - Monitoring table write/read/storage
  - Easy to compute avg, max, min for metrics
  - Quick to query recently data

- **Multi-dimension Table Schema**
  - MR/Spark Job to compute BI reporting data
  - Many demission combination result like city, gender, age, business type

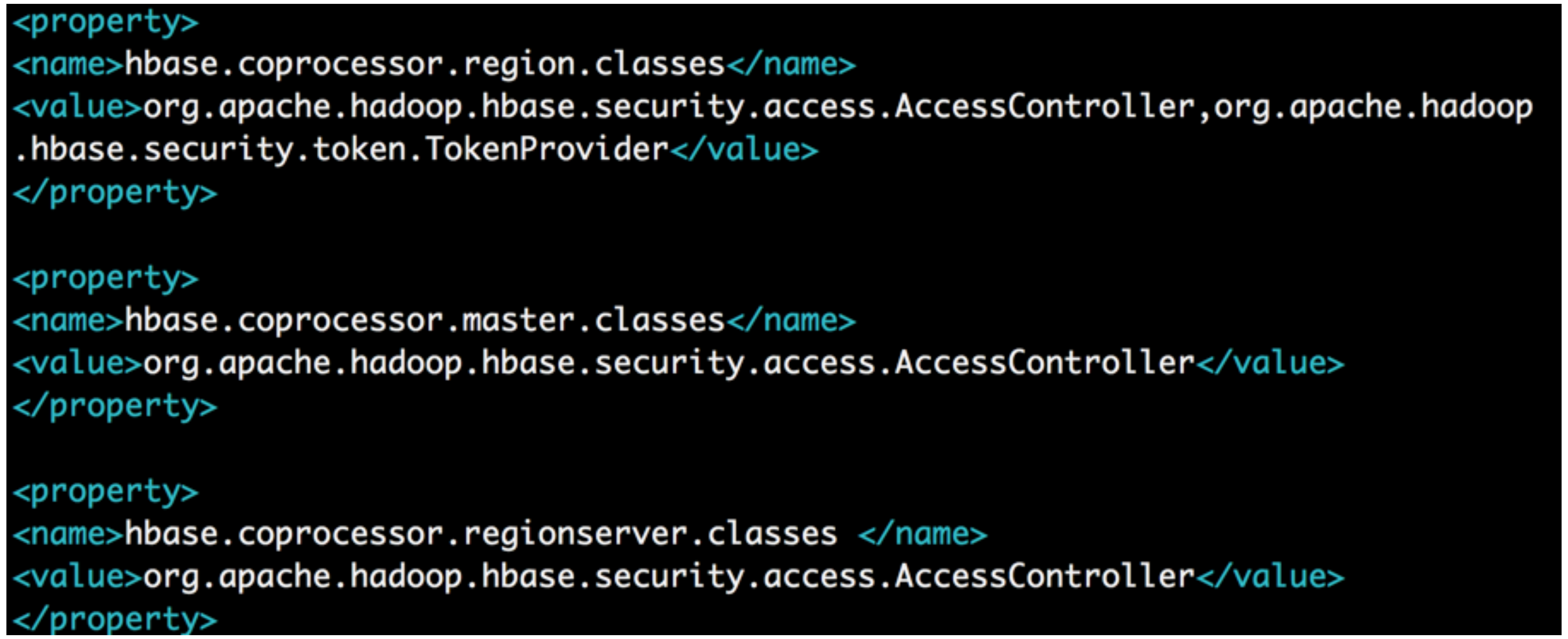
  - Primary Key: JobID, date, dimission1, dimission2, dimission3…
  - Value: dimNameArray, valueArray

  - This can fit nearly all the Multi-dimension reporting business

DiDi HBase Platform

- **Client Access**
  - **Multiple Languages Clients**
  - C++, Go, Python, PHP
  - Thritf2, QueryServer

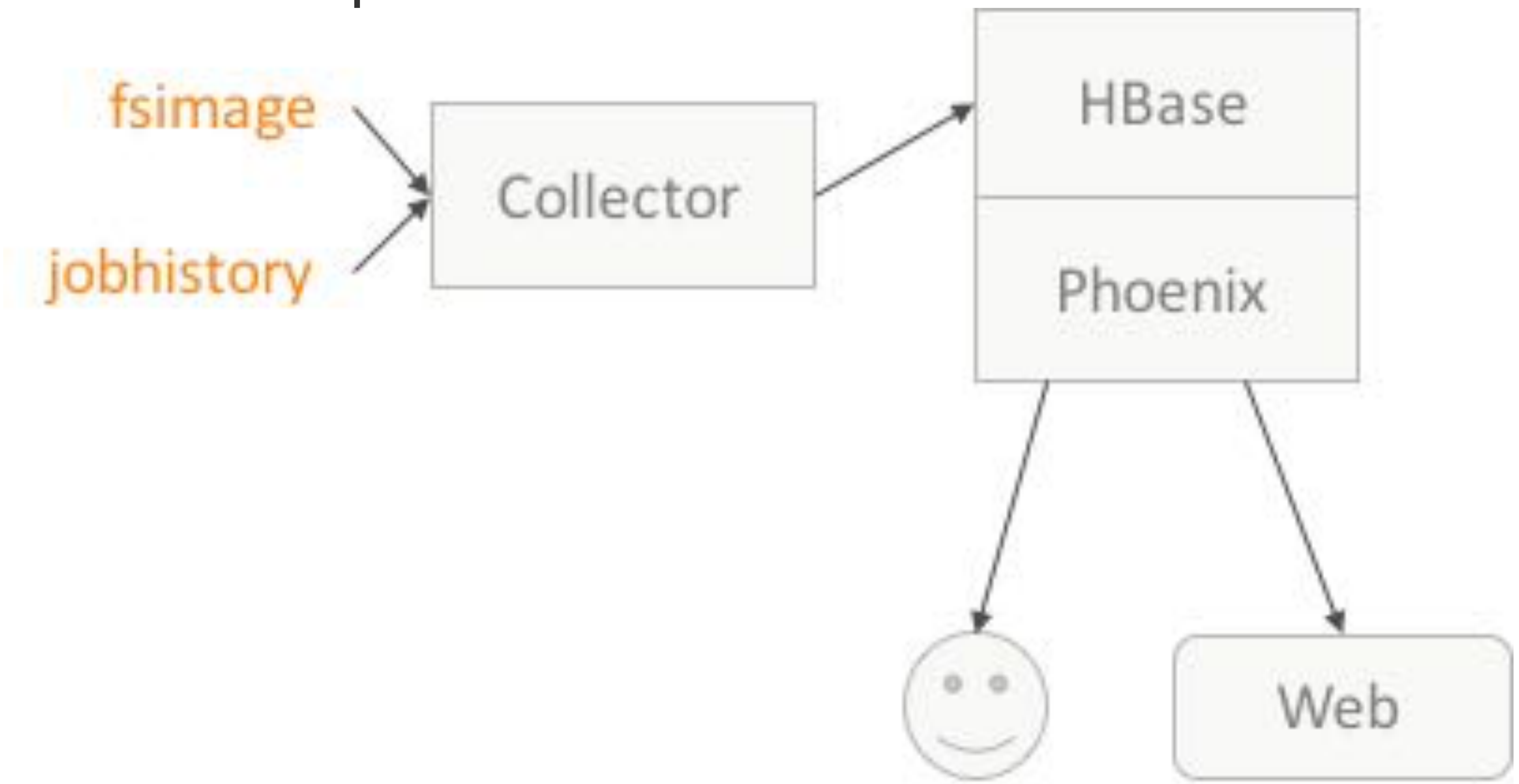
- **Security(ACL)**



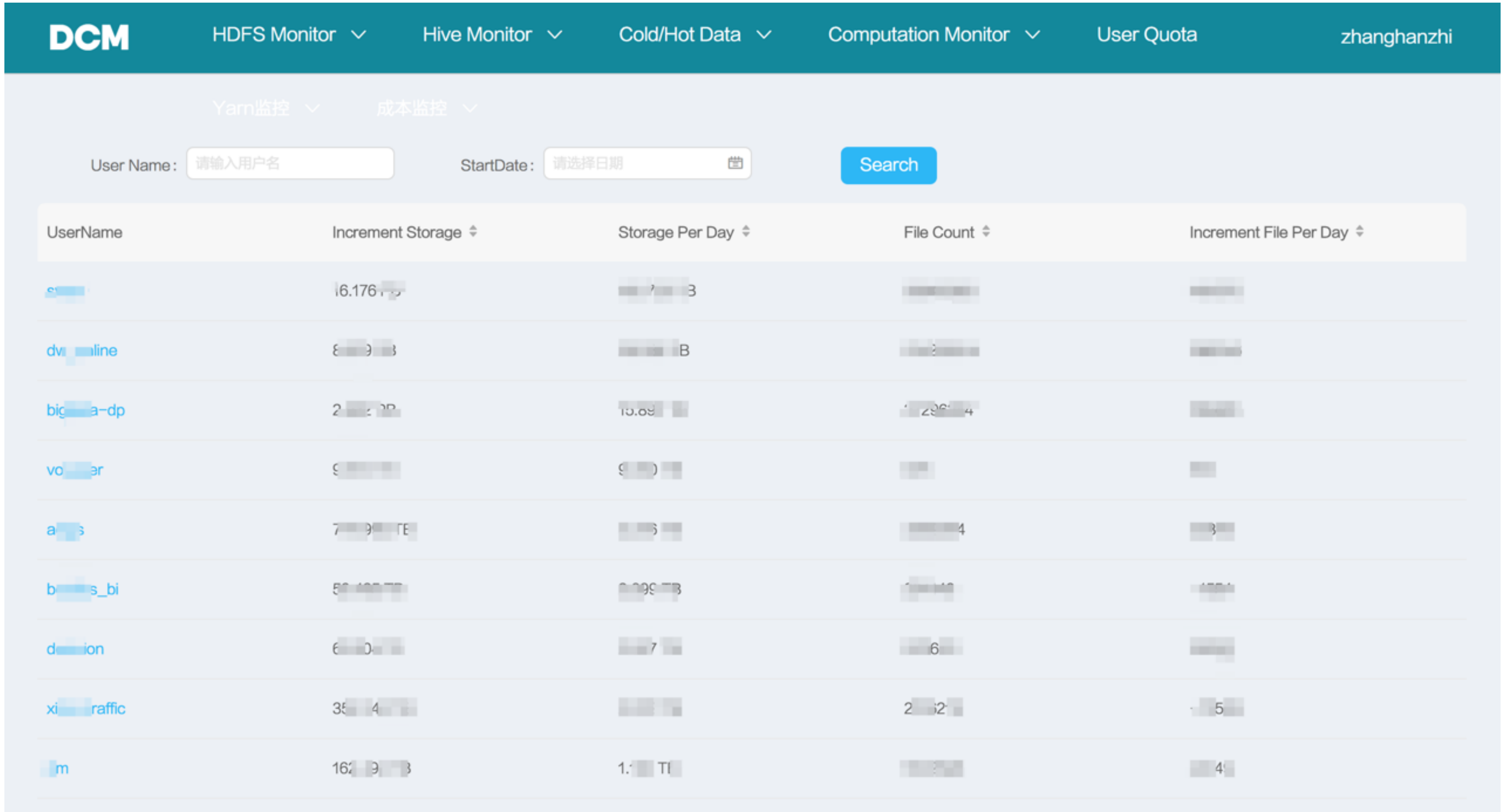
```
<property>
<name>hbase.coprocessor.region.classes</name>
<value>org.apache.hadoop.hbase.security.access.AccessController,org.apache.hadoop
.hbase.security.token.TokenProvider</value>
</property>

<property>
<name>hbase.coprocessor.master.classes</name>
<value>org.apache.hadoop.hbase.security.access.AccessController</value>
</property>

<property>
<name>hbase.coprocessor.regionserver.classes </name>
<value>org.apache.hadoop.hbase.security.access.AccessController</value>
</property>
```

Application and Solution(Hadoop Monitor)

- **Hadoop Monitor**
  - Help hadoop to query their fsimage and jobhistory
  - BI for Hadoop manager
  - Store data in phoenix



Application and Solution



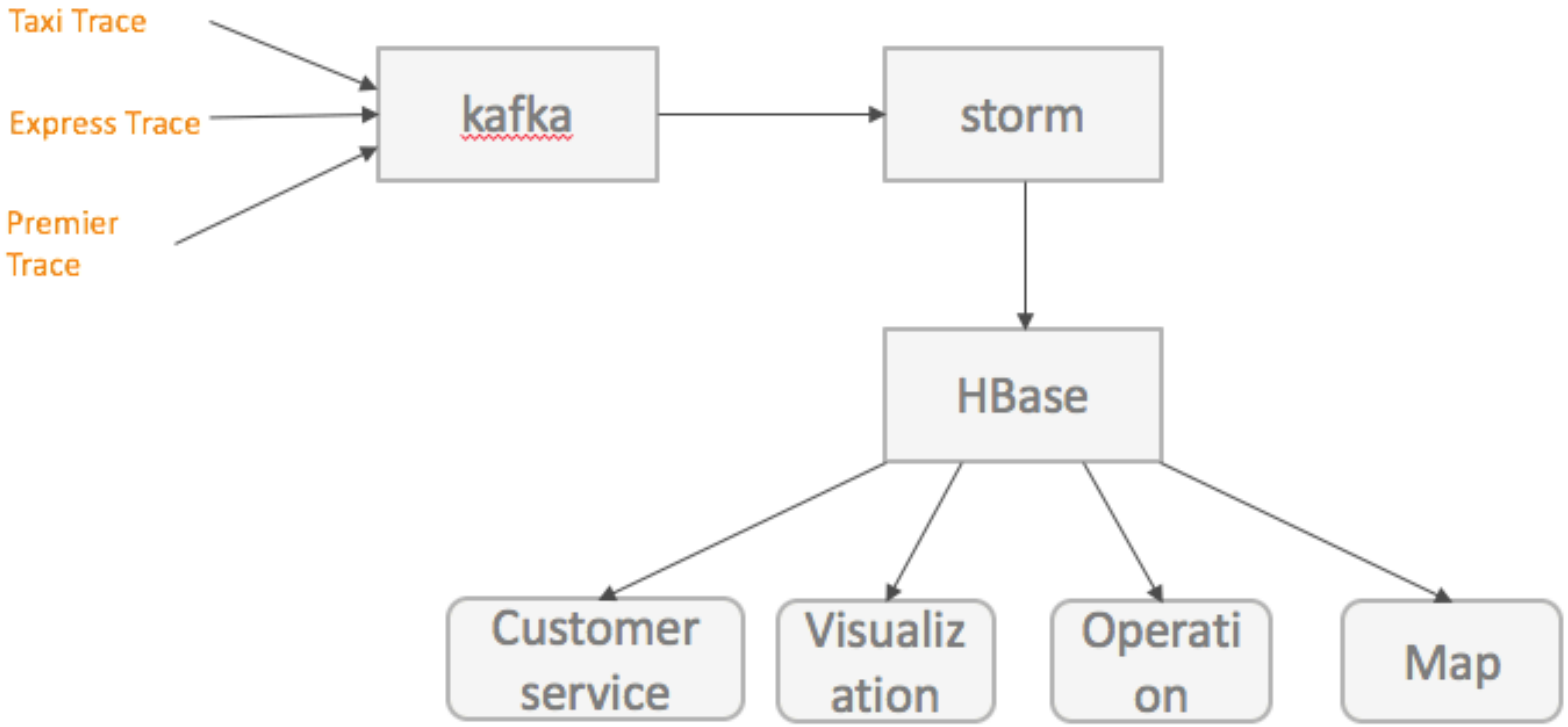
User Name : 请输入用户名

StartDate : 请选择日期

UserName	Increment Storage	Storage Per Day	File Count	Increment File Per Day
[redacted]	[redacted]	[redacted]	[redacted]	[redacted]
[redacted]	[redacted]	[redacted]	[redacted]	[redacted]
[redacted]	[redacted]	[redacted]	[redacted]	[redacted]
[redacted]	[redacted]	[redacted]	[redacted]	[redacted]
[redacted]	[redacted]	[redacted]	[redacted]	[redacted]
[redacted]	[redacted]	[redacted]	[redacted]	[redacted]
[redacted]	[redacted]	[redacted]	[redacted]	[redacted]
[redacted]	[redacted]	[redacted]	[redacted]	[redacted]
[redacted]	[redacted]	[redacted]	[redacted]	[redacted]

Application and Solution(Gis Query)

- **GPS**

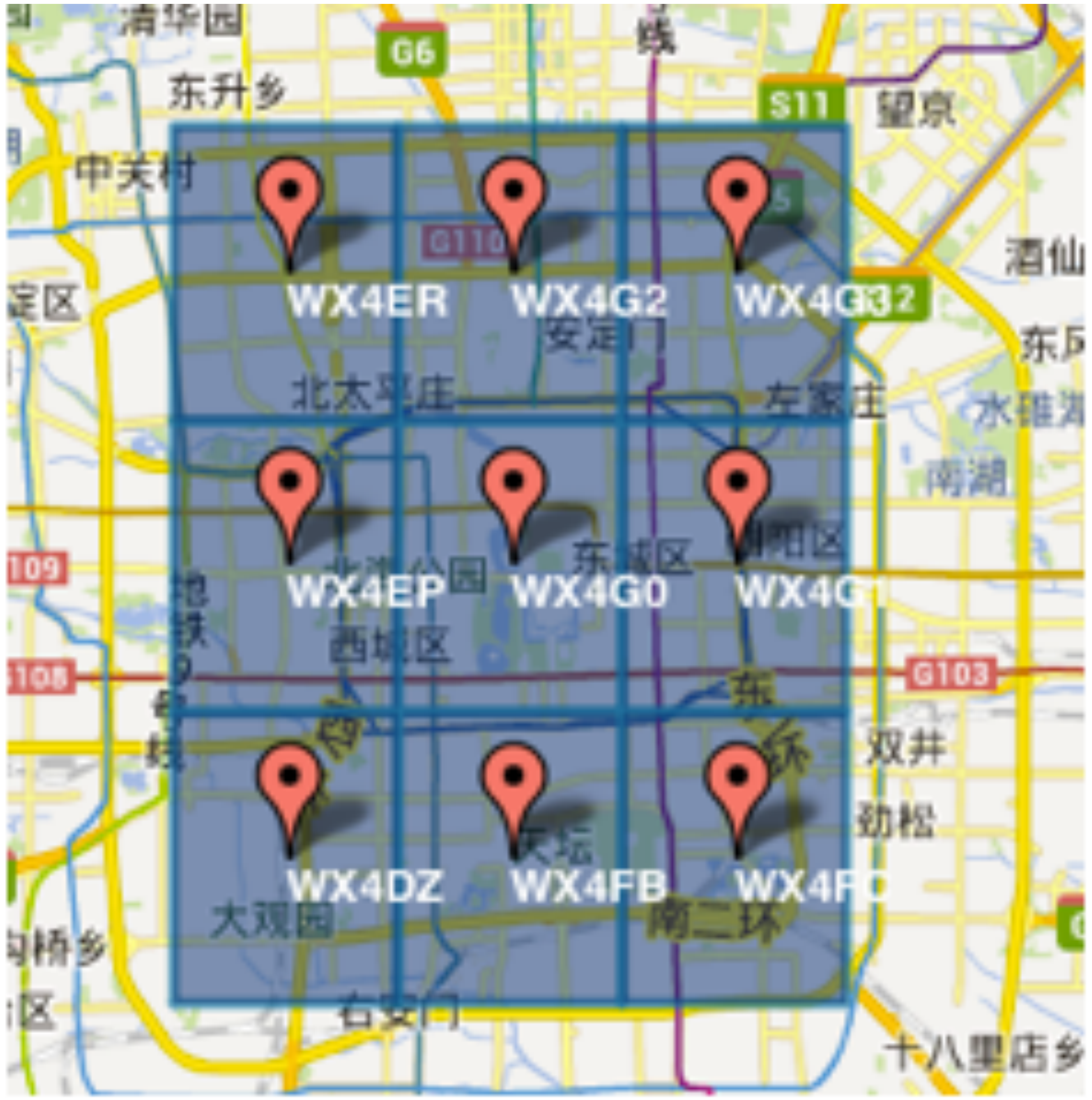


Application and Solution

- **GPS**
  - **Query Model**
  - Rowkey： ID+Timestamp
  - Rowkey： Reversed GeoHash+timestamp+ID
  - **GeoHash**
    - A index in two dimensions
    - Fit HBase rowkey prefect
    - Point 1: same prefix code result in a nearby place
    - Point 2: query rowkey prefix can location a region whose area decided by prefix length

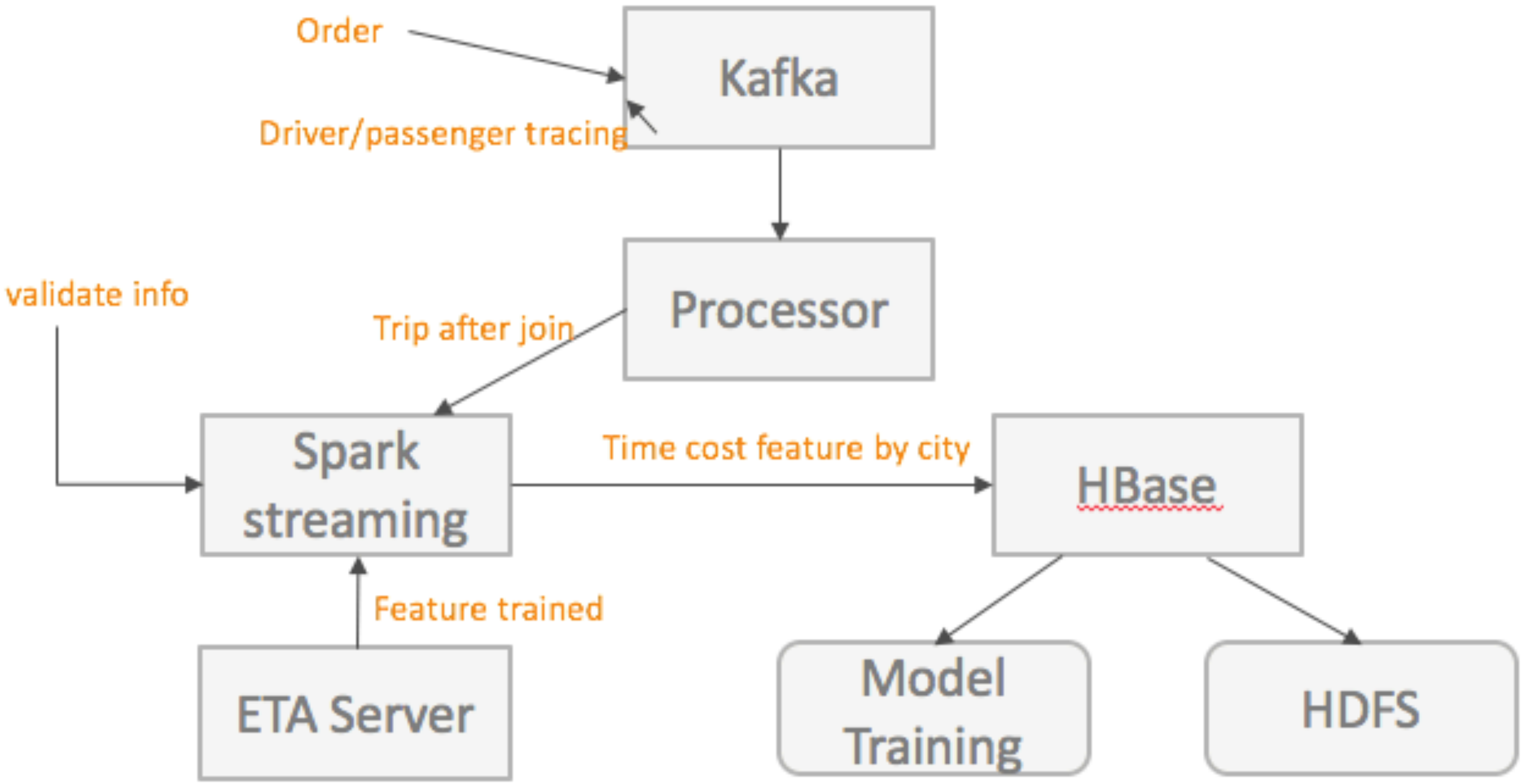
Application and Solution

- GPS
  - GeoHash



Application and Solution(Online Machine Learning)

- **ETA(Estimated Time of Arrival)**
  - Origin data collection
  - ETL
  - Feature extraction
  - Storage
  - Model Training



Application and Solution

- **ETA(Estimated Time of Arrival)**
    - Training data by spark, every 30 minutes
    - Pick up data by city from HBase in 5 minutes
    - Compute ETA in 25 minutes

    - Rowkey: Salting+CityId+Type0+Type1+Type2+Timestamp
    - Columns: Order,Feature

    - Every day HBase data will be dumped into HDFS for offline training

Application and Solution(Image)

- **Traffic in Cloud**
- High Volume Throughput, Little Read

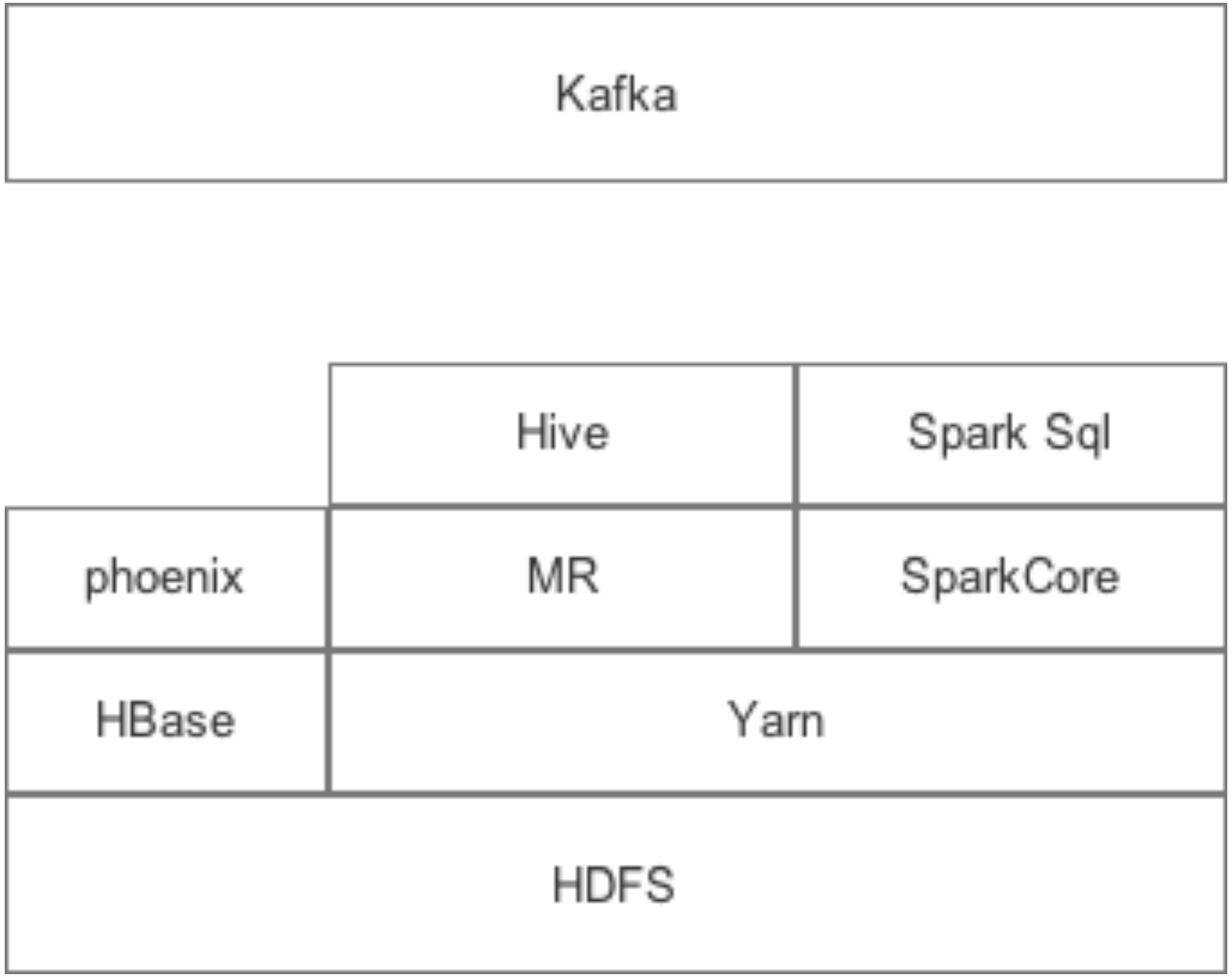
- Read via 8 Thrift nodes

- Road traffic info
- POI data
- Heat-map

- Write with Spark Job

Application and Solution

- **Architecture**



Kafka

Hive

Spark Sql

phoenix

MR

SparkCore

HBase

Yarn

HDFS

Challenges and future

- **More connect with other bigdata framework**

- **Hive Phoenix/HBase Handler**
  - Integration with hive
  - Use Hive sql to query phoenix
  - Easy to load data from hadoop cluster
  - Join with Hive table

- **Spark Phoenix/HBase Handler**

Challenges and future

- More stable
  - hbase1.x upgrade
- OLAP extends
  - Kylin
  - TPC-H Compare
- Thrift load balancer
- Auto Group balancer
- More powerful DHS

Any Question?

滴滴

滴滴一下 美好出行