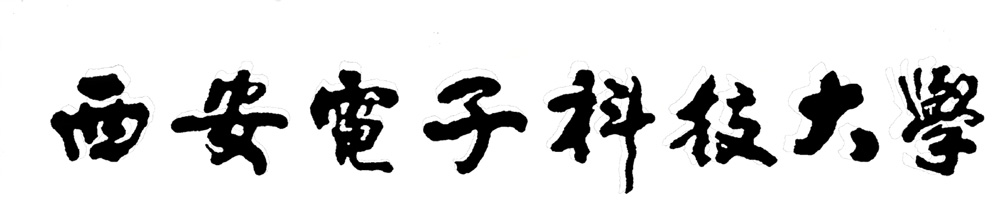
****

**云计算本科大作业**



学 院： 软件工程学院

专 业： 软件工程

**方 向**： **云计算**

小组编号： 第5组

小组名称： 云计算第五组

目录

[一、系统架构 3](#_Toc502333921)

[二、数据流程分析 4](#_Toc502333922)

[1.数据采集过程分析 4](#_Toc502333923)

[2.数据查询和离线处理分析 6](#_Toc502333924)

[三、软件功能分析 7](#_Toc502333925)

[1.系统固有功能分析 7](#_Toc502333926)

[2.系统附加功能分析 8](#_Toc502333927)

[[1] Kafka加分项 8](#_Toc502333928)

[[2] Hbase加分项 9](#_Toc502333929)

[[3] Spark加分项 10](#_Toc502333930)

[[4] Redis加分项 11](#_Toc502333931)

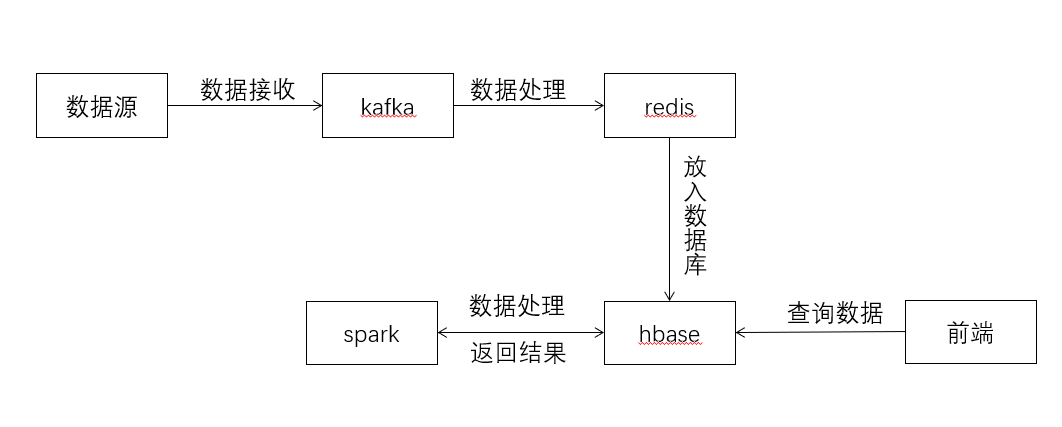
[四、实验感受及收获 11](#_Toc502333932)

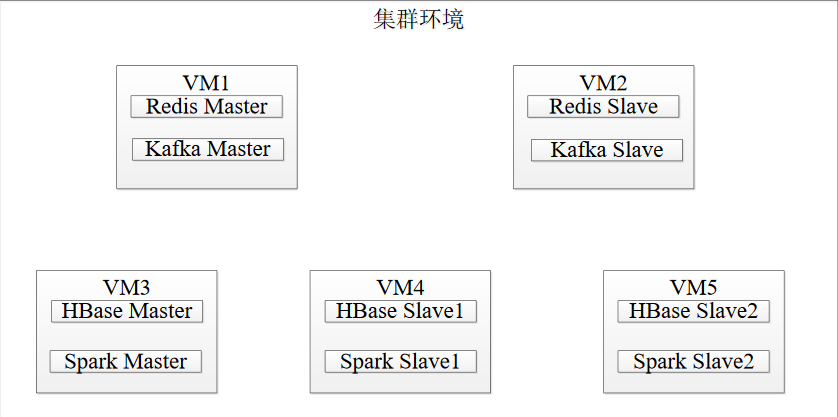
[五、课程建议 14](#_Toc502333933)

# 一、系统架构

该系统用于进行车辆行驶统计。已知有58万条车辆行驶信息记录，记录内容包括车辆的Eid，地点的PlaceID，经纬度，地点以及时间，读取这些信息，并显示行车统计、车辆轨迹以及筛选不符合标准的车辆信息等。

首先，将58万多条record记录发送到kafka中，kafka将数据发送到redis中，redis将数据发送到hbase中进行存储。Spark读取hbase中的数据并进行处理。前端读取hbase中的数据并显示。需要五台电脑，其中两台电脑配置kafka和redis环境，两外三台电脑配置hbase和spark环境。

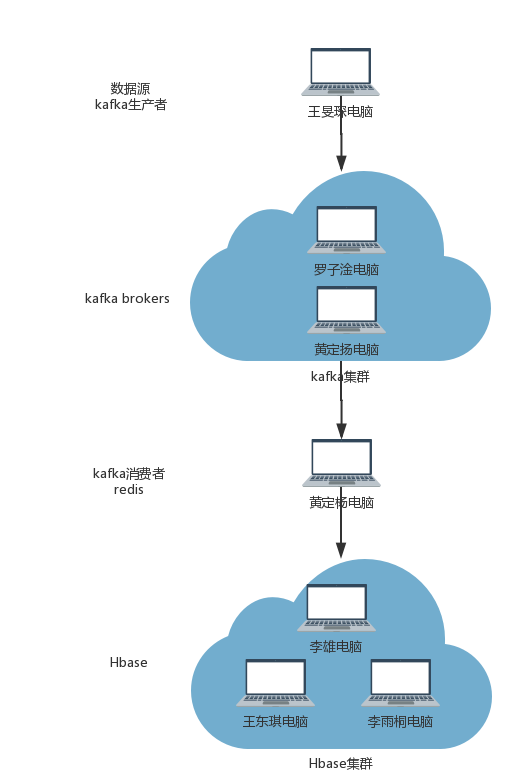
逻辑架构如下:

物理架构如下:

# 二、数据流程分析

## 1.数据采集过程分析

数据采集的过程如流程图所示，其中源数据和数据采集程序源代码位于王旻琛电脑上，而罗子淦与黄定杨的电脑组成了kafka集群作为消息队列，redis位于黄定扬电脑上作为Hbase数据库的缓存，Hbase数据库位于李雄、王东琪和李雨桐电脑组成的集群之上。程序运行时，数据先由jsonToKafka()函数传递到kafka集群上，之后由kafkaToRedis()函数从kafka集群中消费数据并将得到的数据显示出来。接下来调用HBaseCreateOP.main(args)在Hbase中建表准备接收redis数据，最后调用redisToHbase()完成数据入库流程。在数据采集过程中，kafka是一个快速、可扩展的分布式发布-订阅消息系统，其作用是让数据的生产和消费可以异步进行，并且kafka基于zookeeper的分布式应用程序协调服务可以让我们忽略集群部署的细节，将更多精力放在业务逻辑之上。此外，我们的系统中kafka还使用了steams对数据进行了简单过滤，具体实现细节请见本报告加分项说明部分。redis是一个基于内存的Key-Value数据库，其特点是处理速度快，但存储量较小，且因为是基于内存的一般用于保存临时数据。而Hbase是一个基于列的NoSQL数据库，其特点是比redis慢，但存储量大，可靠性高，适合作为最终的大数据存储数据库。因此，在这个系统中我们使用了redis充当Hbase的缓存方式来提高数据采集系统的整体性能。



## 2.数据查询和离线处理分析

运用HBaseSearch类进行数据查询工作。类中searchByPlaceID函数实现通过placeId，扫描全表，获得表中通过该地点所有车辆信息，并返回一个对象为Record的List。

在searchByPlaceID函数中，首先调用HBaseConf类中的getTableByName函数建立与HBase的连接，对数据库操作提供Table连接对象，返回表名为Record的Table。接着建立行过滤器，由于HBase表中行键格式为placeId##time##eid，故运用比较过滤器，匹配子串placeID##。

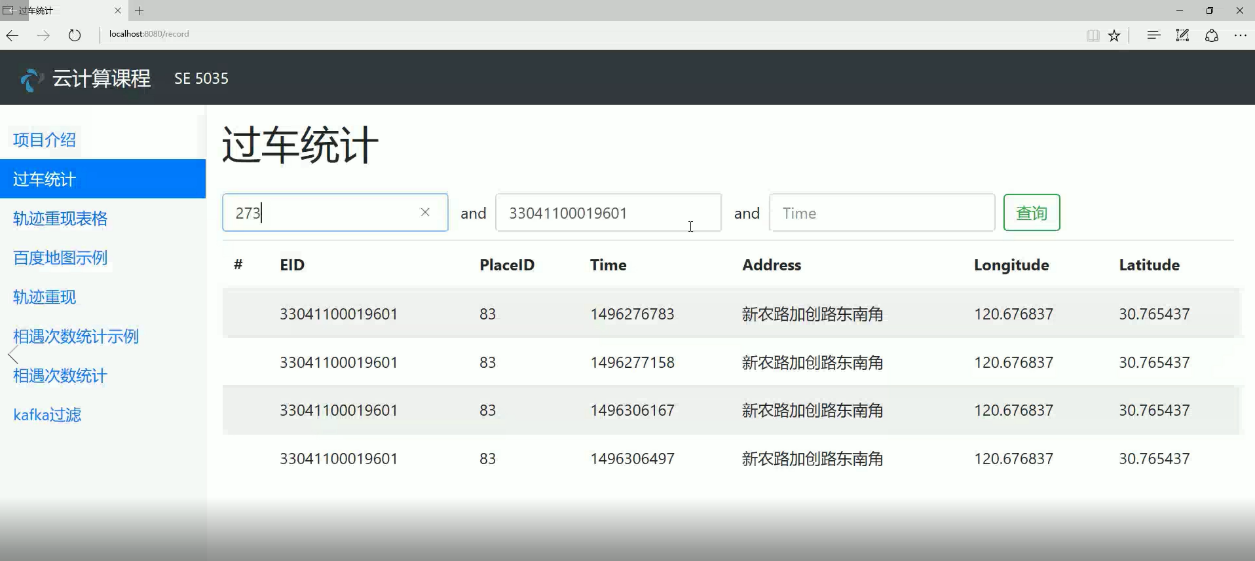
在Table中调用行过滤器，获得行键中placeID与函数传入参数相匹配的行，将匹配成功的每行Result数据通过RecordMap类中的resultMapToRecord函数转换成Record对象：通过调用Result类中的getRow函数获得行键，调用Record类中的setPlaceId、setTime、setEid函数将行键分解，存储到Record类中的placeId、time、eid属性中。再通过调用Result类中的listCells函数获得所有的列信息，调用Record类中的setAddress、setLongitude、setLatitude函数分别将其存储到Record类的address、longitude、latitude属性中。

将转换成Record对象的行数据加入名为resultRecords的List，返回至前端显示。

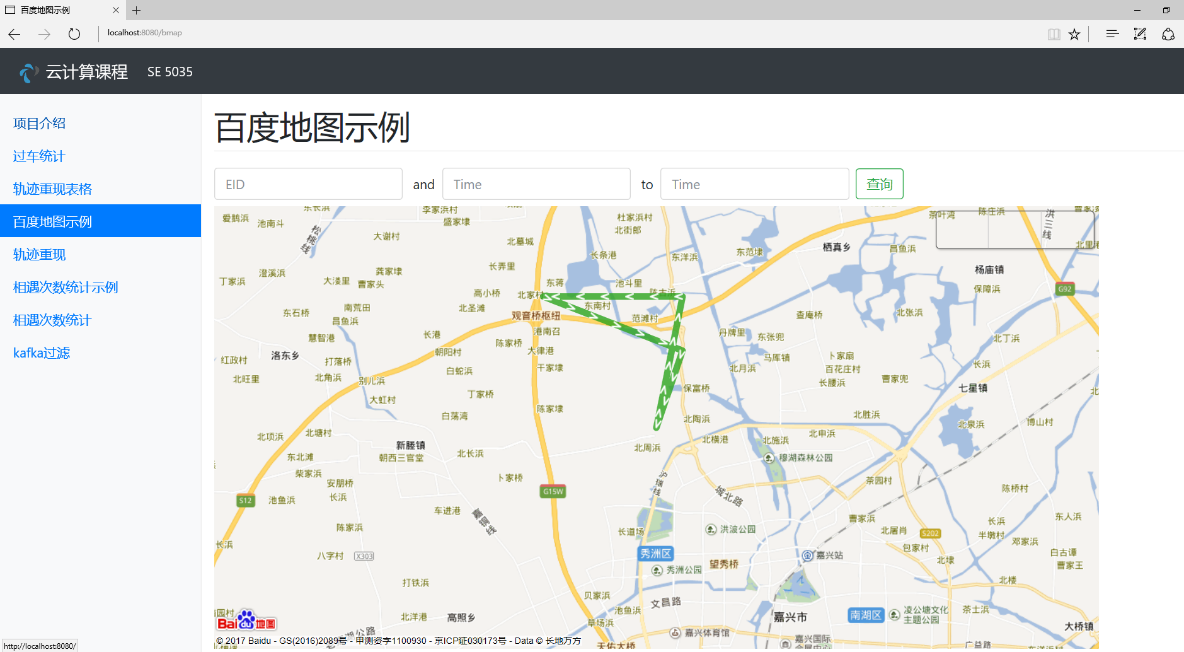
调用HBaseConf类读取对象为Record的List。调用RDD算子mapToPair创建键值对RDD，以(PlaceID, address, latitude, longitude)为键，eid为值。调用键值对聚合函数combineByKey，函数参数中createCombiner将pairRDD中值eid转换为1，mergeValue把元素eid和count合并为count+1，mergeCombiners把countA、countB合并为两者之和。聚合后，pairRDD以过车信息为键，过车次数为值。再次调用mapToPair，返回对象为VehicleCount的List。最终实现以地点信息为键的过车次数统计。  
 为实现相遇次数统计，类似的再次读取对象为Record的List并创建键值对RDD，以palceId为键，(eid, time)为值。调用内连接函数join后pairRDD结构变为以((eid1, time1), (eid2, time2))为值。通过调用filter算子，过滤出eid1与eid2不同，time1与time2相差小于60的行键（即不同车辆接近在同一时间经过同一地点的信息）。再次mapToPair后pairRDD结构变为以(eid1,eid2)为键，1为值。利用reduceByKey函数，将RDD中键相同的多个元素的值reduce为一个值，即相加之和，再与原RDD中的键组成新的键值对。再次调用mapToPair后，返回对象为MeetCount的List。最终实现以两车id为键的相遇次数统计。

# 三、软件功能分析

## 1.系统固有功能分析

固有功能过车统计。整个系统在环境搭好的情况下，能够实现将58万多条的过车数据从kafka经Redis传到Hbase中其，再由前端展示。其中过车统计界面，输入Eid，PlaceID，Time中的任意一、二、三个值都可以查询到相应的数据。界面如下所示：

百度地图实例界面展示了百度地图API如何调用，以及给了百度地图的一个账号密码供前端使用。同时前端带有轨迹重现界面（未完成轨迹实现功能）。

界面如下所示：

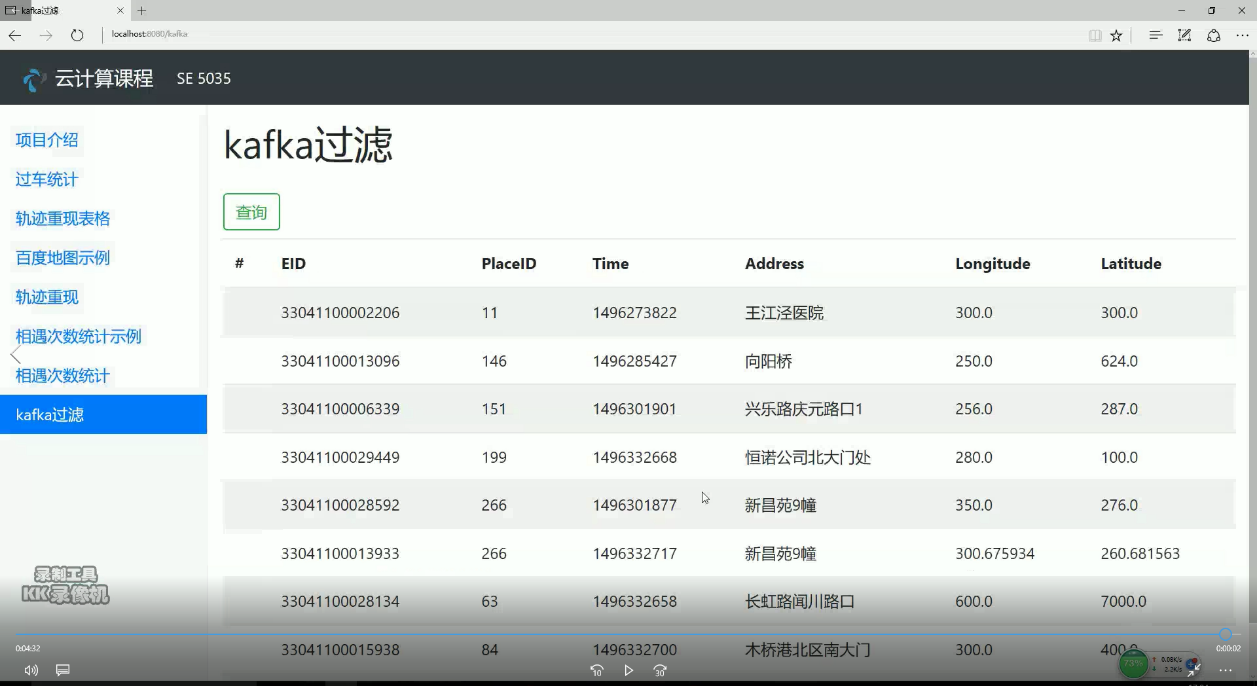
地图上的轨迹是前端mock数据秒点画图呈现出来的。

## 2.系统附加功能分析

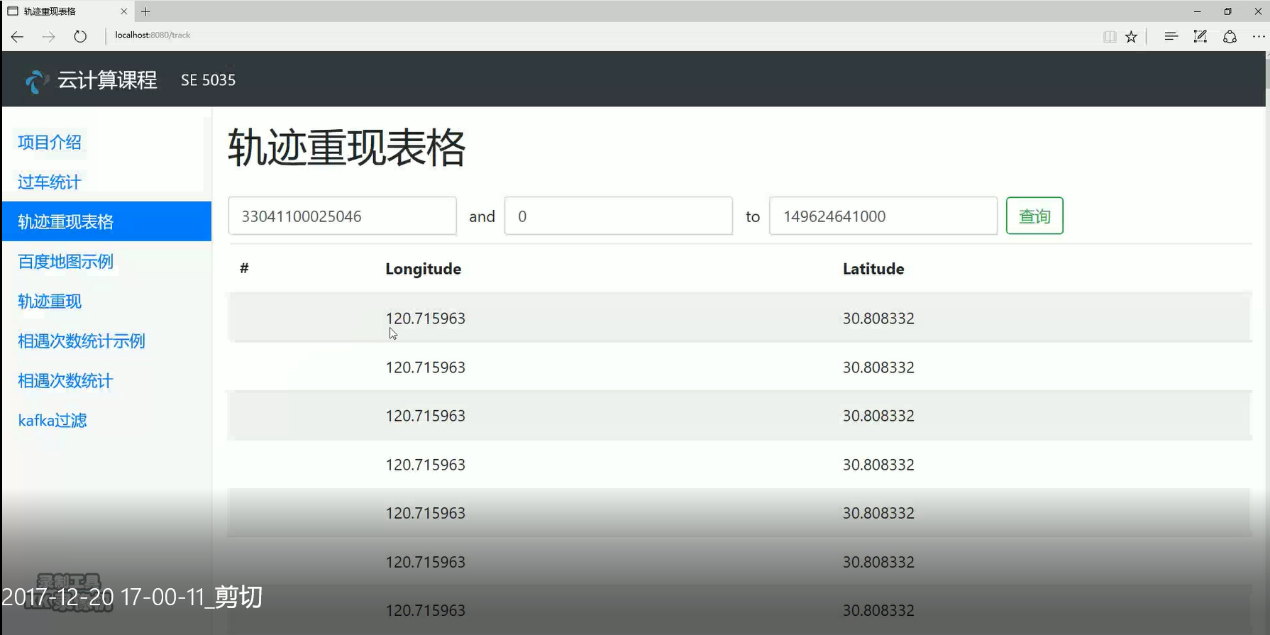
我们组做完了所有的加分项，包括Kafka的过滤，Hbase的查询数据，Spark的相遇次数统计，Redis的发布订阅，以及前端为后端所有加分项提供的界面展示。

1. Kafka加分项

其中系统附加功能有展现Kafka过滤掉的数据（Kafka过滤界面），原始信息中包含着若干条错误记录，错误记录中的经纬度不合法，经度大于130，纬度大于40。编写代码利用Kafka Stream流处理中特有的filter（过滤器）对Kafka中接收到的原始数据进行过滤处理，因为Kafka Stream流处理是针对于topic数据到topic数据的，所以定义两个topic，一个存放原始数据，一个存放正确数据。在处理的过程中定义一个新的rediskey值并在过滤的过程中jedis.sadd(rediskey,value)函数将这些错误的数据传入Redis中以便在之后的前端展示中使用。过滤完成之后将数据传入存放正确数据的topic。之后改写代码，将正确数据传入Redis中，之后在Hbase中建表将错误数据与正确数据分别存放在表中，分别设计界面进行展示。数据由前端读取Hbase中存储的Record2表得到被过滤掉的这九条数据，结果如下所示：

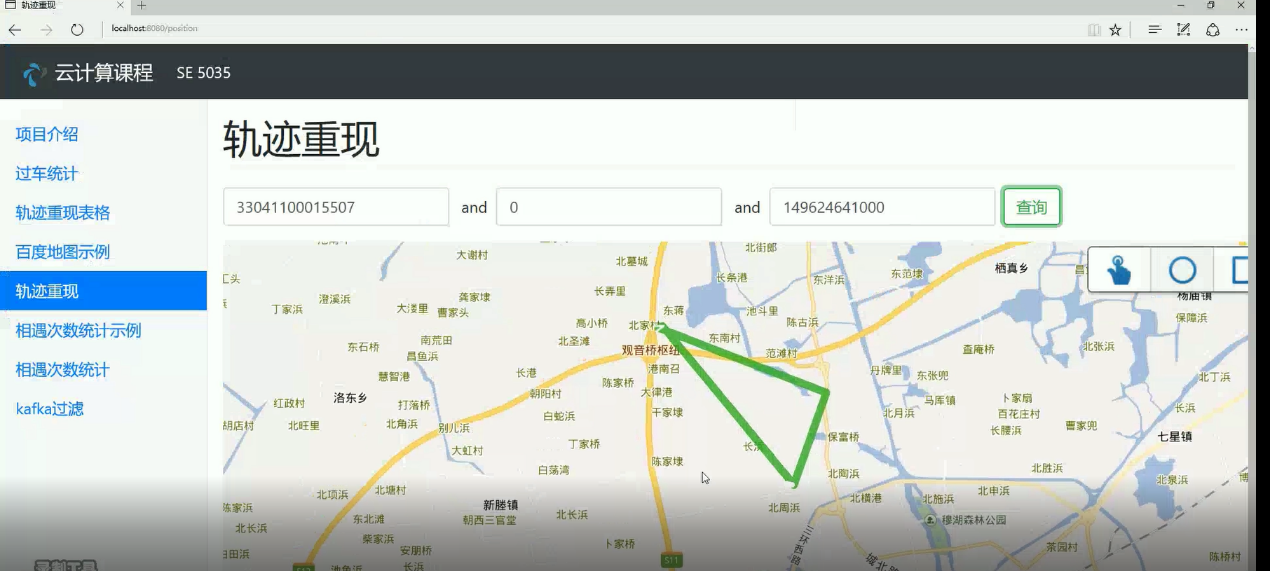
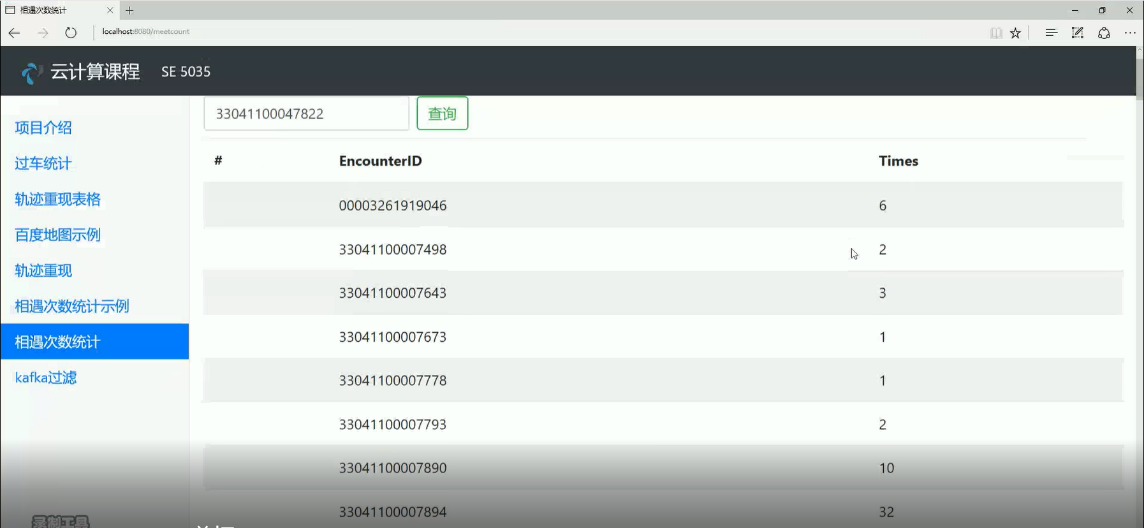


1. Hbase加分项

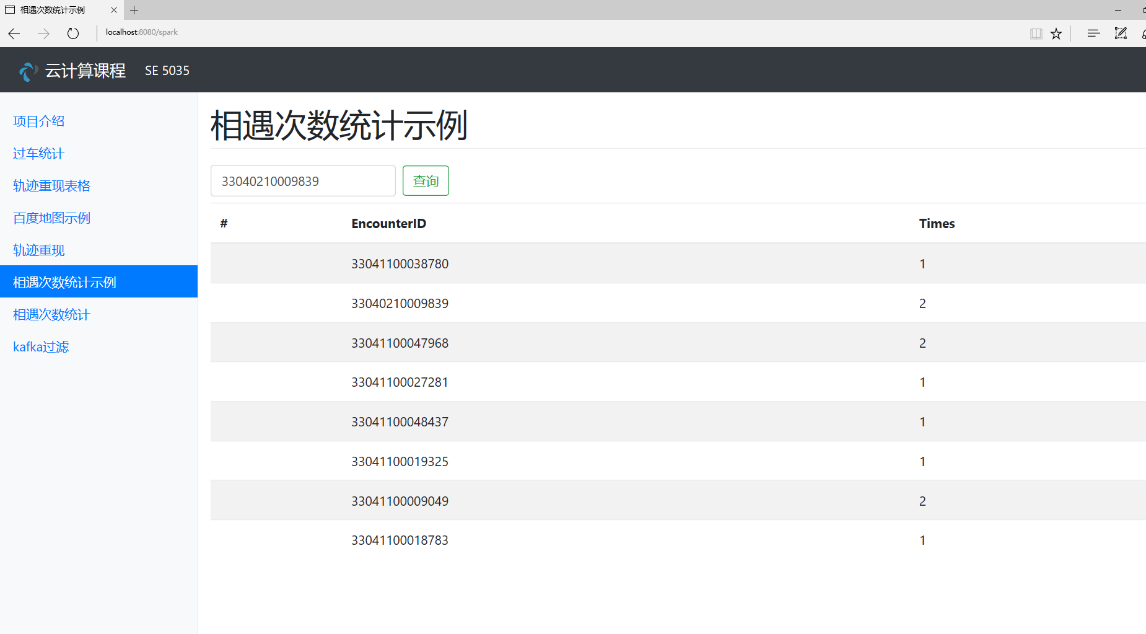
展现Hbase查询结果返回的精度和纬度（轨迹重现表格形式），输入Eid和一个时间段，就可以展示出该车辆按时间排序的经纬度，结果如下所示：

将Hbase查询出的精度和纬度结果用于轨迹重现展示（轨迹重现地图形式），通过调用Hbase中返回的经纬度，调用百度地图覆盖的API就可以将经纬度连城线展示在地图上，获取数据采用定义REStful接口，以ajax的方式动态获得。结果如下所示：

1. Spark加分项

展现Spark加分项——相遇次数统计（相遇次数统计界面），通过调用存储在Hbase中的MeetCount表，用Hbase给出的查询函数将查询结果在前端展现出来，结果如下所示：

模拟Spark相遇次数统计的界面（相遇次数统计示例），前端通过data.java文件mock数据在前端做一个示例性的展示，结果如下所示：



1. Redis加分项

发布订阅(pub/sub)是一种消息通信模式，主要的目的是解耦消息发布者和消息订阅者之间的耦合，这点和设计模式中的观察者模式比较相似。Redis作为一个pub/sub server，在订阅者和发布者之间起到了消息路由的功能。订阅者可以通过subscribe和psubscribe命令向Redis server订阅自己感兴趣的消息类型，redis将消息类型称为频道(channel)。当发布者通过publish命令向Redis server发送特定类型的消息时。订阅该消息类型的全部client都会收到此消息。

具体步骤如下。首先定义一个监听类，其继承JedisPubSub类。在onMessage中将json格式字符串转成Record类，用集合set存储，并取消本次订阅。然后订阅频道channel，遍历Redis中存放的数据，每次取一条数据发布到该频道中。当监听到发布信息后，执行onMessage相关操作。重复上述过程直至取完全部数据。

# 四、实验感受及收获

**罗子淦：**经过这一学期的云计算课程学习，了解了云计算技术在当今社会中的运用，学习了云计算的基本概念、数据中心相关概念、虚拟化技术、软件定义网络/存储等知识。同时通过组成小组完成云计算课程上机大作业让我更加清晰直观对分布式文件系统、云数据库、云计算框架等云计算核心技术有了了解，熟悉了利用云计算技术解决工程问题的方法与技能。在组队完成云计算大作业的过程中，组内分工合作，我与李帧沛共同负责数据采集（Kafka）部分，掌握了Zookeeper集群、Kafka集群在linux环境下的搭建方法与使用方法，明白了Kafka在系统的作用，学会了数据采集的操作。同时在与组内其他同学负责部分相连的过程中，了解了其他部分如Redis、Hbase、Spark在系统中的作用及重要性。最后在实现加分项的过程中使用Kafka Stream流式计算，通过设置条件运用filter过滤掉了错误的数据，成功将正确的数据传到Redis中。总的来说，在这学期的云计算课程学习以及组队完成云计算课程大作业的过程中将课本知识与实际操作相结合，掌握了云计算的相关知识，掌握了运用云计算技术解决实际问题的方法，学会了组成云计算系统的各种部件的搭建方法与使用方法，虽然这些知识仅仅只是云计算知识的冰山一角，但为我打开了云计算世界的大门，让我对这门技术有了浓厚的兴趣与学习的动力！

**李祯沛：**正如鲍亮老师在课程介绍时说的那样，这门课涵盖的信息量很大，因此学习的难度较大。例如在完成大作业的过程中，我负责kafka部分，光环境搭建就十分困难，遇到了不少问题，当时感觉十分焦躁，不过幸好最终我们还是克服了种种麻烦顺利完成了大作业。而且在课程开始的时候，也遇到了很多新的名词和概念，比较迷惑。另外，我认为这门课可能因为课时较少的缘故，所以个人感觉授课效果并不好，很多知识只是泛泛而谈的话可能学完了还是会让人感到困惑。当然，我对这门课的实践部分，也就是大作业的效果评价是很不错的。虽然遇到了不少问题，但正是在不断查找资料解决问题的过程让我逐渐对相关技术的理解变得更加深入了，我认为这才达到了这门课的效果。最后还有一点想说而是建议调整一下课程安排，将这门课置于java课程之后，否则我们在没有系统学习java的情况下调试代码还是比较困难的。总的来说，在本次云计算课程的学习实践中，我受益匪浅。云计算技术是一个较为前沿的概念，因此我十分感谢学院能跟上时代的潮流开设这门课程让我们能接触到最近的知识技术。

**王东琪：**通过云计算课程的学习，我学习了云的基本概念，深入了解了数据中心，云存储等概念，并了解了kafka、redis、hbase、spark等软件的使用，参与了车辆统计的大作业过程。在大作业的实施过程中，我参与配置了每一个软件的环境搭建以及hbase和前端功能的实现，并在一次一次的突破难题中提升了自己的能力。Hbase作为nosql，主要负责大量数据的存储，实现获取车辆轨迹的加分项时，通过获取符合输入的eid的车辆记录，转为Record对象，并将符合时间的对象存入list列表中，前端获取该list，调用百度地图api，获取list中的经纬度并显示。

**李雨桐：**这是我第一次参与并完成10人合作的软件项目。尽管在课程最开始对于自己能否顺利完成这一项目感到不确定和怀疑，但在10人一个学期的讨论和配合之后，还是圆满完成了任务。我觉得自己不仅学到了云计算有关的知识，让自己对于云计算的了解不仅仅停留在理论的表层，而是付诸于实践，亲自动手熟悉了Hbase的相关操作。并且在这个过程中体会到了团队软件开发的不易，每个人以及每个人的工作在这个过程中都是非常重要的一环，大家的密切配合才能成就最后团队项目的顺利完成。

**王旻琛：**通过这次课程，我第一次以团队的形式深度参与到一个项目中。在与团队的交流合作中，我们共同学习之前未曾接触过的新领域，相互讨论解决我们各自遇到的困难，我能感觉到大家都在为不拖累团队而在各自所负责的方面倾尽全力。我作为负责Redis的成员，首次接触到Redis这一强大的开源数据库，了解到其所具有的特性以及能完成的许多功能，并且对云计算的分布式系统搭建和工作方式有了初步的认识，这些都为以后的深入学习打下了坚实的基础。正如老师在第一节课所说，云计算与大数据乍一听很是高大上，但仔细一想却感到十分的空泛，似乎没有一个明确的目标供我们选择。其实，在我看来其所覆盖的知识领域十分广泛，它更像是一个各种学科相互配合协作的有机结合体，我们不可能通过短短一学期的时间凭借简单的一门课就能真正理解掌握这个方向上各方面的知识。我觉得我们应该寻找自身所关注的兴趣点，从兴趣出发，专注一个具体的方向深入学习研究，逐步理解并且能熟练使用各种工具。通过这样的学习，我想我们不但可以准确建立学习的目标，也可以更好的配合团队中其他成员的工作。

**黄定扬：**通过该课程，我和队员们一起探讨云计算的知识、一起参与到云计算环境的搭建，各个部分功能的实现，我们互相配合，互相合作，不懂的问题可以一起交流，我觉得找到了一种团队合作的感觉。虽然刚开始的时候，感觉这个项目难度挺大的，但是在大家的努力下，从搭建环境到跑原代码，从原代码到加分项的实现，我们一步一步走过来，觉得学到很多云计算方面的知识，提高了自己查找资料，自学的能力。我是负责redis这一块的，我了解到Redis作为缓存的强大功能，并且通过动手编程实现发布订阅，感觉收获很大。刚刚接触Redis的时候，感觉入门知识不难，很容易就能理解它的原理，但是结合到整个项目，就不是只掌握一些基础知识能解决的，因为Redis是和其他功能模块紧密联系的。首先要弄懂Redis在kafka和HBase中充当什么角色，Redis有什么作用，原来的代码是什么意思，实现了什么功能，用发布订阅做有什么好处等等，然后开始着手写代码，学习jedis的用法写法才能慢慢写出来。

**李雄：**这学期学习了云计算技术，通过团队的合作以及自己的动手体验我对spark，hbase等的工作原理和工作方式进行了深入了解，知道了云的工作方式，步骤。云计算作为原来IT技术基础上发展起来的一门新技术，必将领引IT技术的发展方向。通过这学期对环境的搭建，代码的运行等步骤我们也会在云计算技术上越走越远。

**汪邦举：**这学期我们通过对一个比较完整的云环境的搭建和工作代码的运行，深入理解了并行计算、分布式计算和网络计算的原理和工作方式。云计算的分布式文件系统，用冗余存储的方式，保证数据存储的高可靠性和高可扩展性。我们通过一系列的实践加深了对理论知识的理解，同时也训练了团队合作的能力。

孙晖：在做整个项目的过程中，组员们都非常的用心和努力，和他们一起做云计算大作业是一件很放心的事情，非常感谢他们的努力。在我做前端的过程中，碰到了许多的问题，非常感谢姚琦宇学长不厌其烦的为我解答，学长耐心的解答给了我很大的帮助。在整个项目中，学到了很多东西，学会了前端的springboot架构，学会了ajax获取数据的方法，收获很大。

# 五、课程建议

1. 这门课程重在参与到云计算环境的开发中，通过这门课程，学生能收获实践经验和提高编写程序的能力。在实际实验中，会遇到各种各样的困难，比如说，对JAVA语言、Linux不熟悉、对开发软件不熟悉、遇到一些难以查找问题的报错等等。单单要把项目运行起来，只要按要求按步骤完成，在辅导老师提供的帮助和相关代码的提供下，实现难度不大。但是，要实现加分项还是有难度的。这个项目最大的弊端是虽然组员人数足够，但每个人分工太明确，导致没有人对整个项目有清晰的理解，很容易造成不知道某一块的工作原理和目的。建议在以后的课程中，老师或辅导老师能在项目开始前，先给我们讲解清楚整个项目，不然做起来没有方向，无从下手。
2. 给出的基础环境搭建的资料里面有些错误，在实际操作中不容易发现，或者在后面调试代码才出现错误。改由于配置环境方面的错误比较耗费精力，建议在以后课程中，给出这些资料之前能检查一下，以免大家在使用这些资料时出错。
3. 这门课虽然是重实践，但是目前我们的理论知识跟不上，建议在课程实验开始前，给同学们发一些基础的知识资料，比如说要用到的JAVA,Linux的基础知识，让大家先熟悉熟悉，因为后续需要在Linux环境下，使用JAVA语言编程，而大三上学期我们只学了C/C++，对Java会不太熟悉。
4. 建议每组都需要一个能力强的队长，因为能力强的队长能对整个项目有一个清晰的理解和认识，能有效地协调不同部分的队员之间的沟通交流。当队员遇到问题时，可能这个问题并不局限在某一个功能模块上，因为每个部分都起着承上启下的衔接的作用，它们是互相影响的，如果只是局限于自己部分的功能的实现，可能导致大家的想法不一样，大家按自己办法实现自己的功能后，最终汇集时可能出现各种问题，导致无法成功。能力强的队长能更好地分配工作，带领能力一般的同学一起学习，反之，如果一个团队中的所有队员的能力都是一般的话，就容易迷茫，不知道该做什么，怎么做。虽然课程中，辅导老师和其他同学们在piazza的提问和回答都挺积极，但是实际遇到的问题太多了，每个问题或大或小，每个问题都上去问貌似不太现实。