1. 卡扣流量分析

全部使用SparkCore实现。

1. 筛选条件内的卡扣信息。

根据使用者传入的指定条件，筛选出指定的一批卡扣信息。

1. 检测卡扣状态

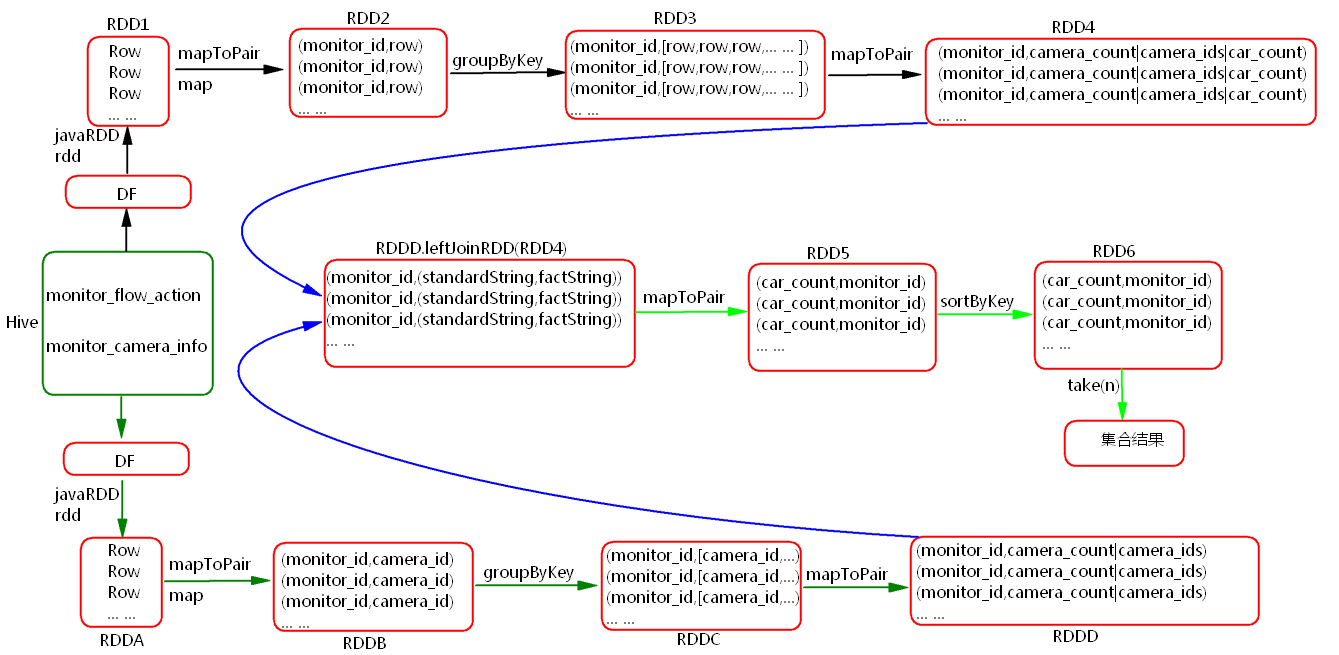
基于第一步查询出来的卡扣信息结果来实现。

检查卡扣（monitor\_id）状态，卡扣状态分为正常，异常状态。

摄像头(camera\_id)的状态分为正常，异常。

异常摄像头的所有信息（monitor\_id：camera\_id）。

业务理解图：



注意：

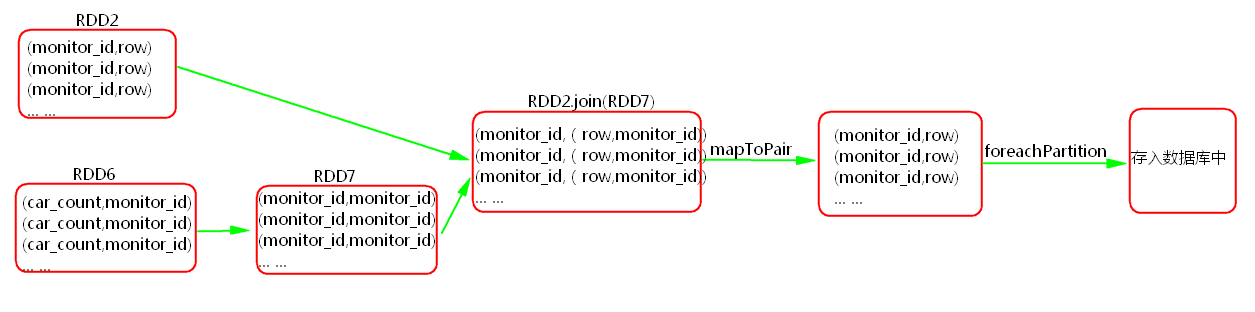
1. .运行代码前，要修改下Mysql中表task中参数的时间。
2. .清空除了task表、areaInfo表其他不相干表中的数据。
3. .如果本地运行要修改本地项目中my.properties文件中运行模式spark.local=true，如果在集群中运行要改为false
4. 提交到集群中运行命令：

|  |
| --- |
| ./spark-submit  --master spark://node1:7077,node2:7077  --class com.bjsxt.spark.skynet.MonitorFlowAnalyze  --driver-class-path ../lib/mysql-connector-java-5.1.6.jar:../lib/fastjson-1.2.11.jar  --jars ../lib/mysql-connector-java-5.1.6.jar,../lib/fastjson-1.2.11.jar  ../lib/ProduceData2Hive.jar  1 |

1. 统计车流量最多的topN卡扣

如上图所示，结果take就是取车流量最多的卡扣。

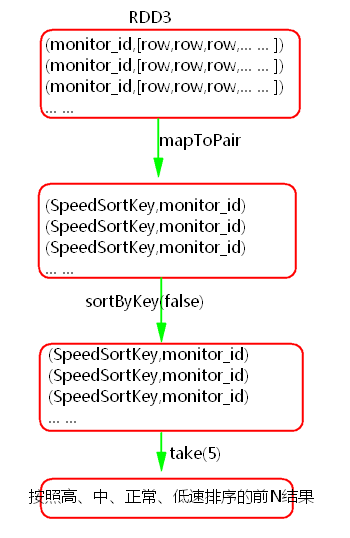
1. 获取topN卡扣所有车辆详细信息并存入数据库



注意：RDD7可以使用广播变量优化。

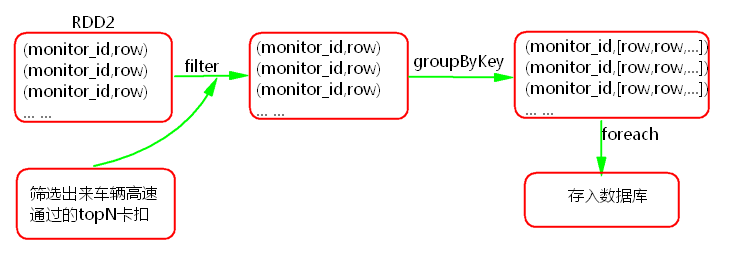
1. 计算经常车辆高速通过的TopN卡扣

思路：要找出车辆高速通过的TopN卡扣不能单单以一个最快或者最慢的速度来衡量，而是要统计每一个卡扣下，高速通过的车辆数，中速通过的车辆数，正常通过的车辆数，低速通过的车辆数。然后统计出每个卡扣下对应的通过车辆高速的总数、中速总数、正常总数、低速总数。然后将这些维度封装到对象中，按照对象来实现排序。



对RDD3使用mapToPair将每条车辆详细信息中速度封装到自定义对象中，然后按照高速通过的车辆数。

1. 获取车辆高速通过的前N卡扣中每个卡扣速度最快的前10车辆，并存入数据库。



1. hive中导入模拟数据
2. 打开hive手动创建database 为traffic 。
3. 生产模拟数据，将模拟数据提交到linux上。
4. 将data2hive代码 打包，提交到客户端运行。
5. 查询hive中数据库为traffic中monitor\_flow\_action和monitor\_camera\_info两张数据库表是否导入数据。
6. mysql数据库乱码的问题

在安装mysql的linux节点路径/etc/my.cnf中加入：

在[client]下添加  
  default-character-set=utf8

在[mysqld]下添加  
  default-character-set=utf8

如图：

