# 课堂笔记

Kafka创建topic：

kafka-topics.sh --zookeeper slave1:2181 --topic order --replication-factor 1 --partitions 2 --create

Kafka消费测试：

kafka-console-consumer.sh --zookeeper slave1:2181 --topic order --from-beginning

## [案例] 按天、地区统计销售额

维度：日期、地区ID

指标：sum(amount)

Select day,areaId,sum(amt)

Group by day,areaId

## 时间窗口

场景描述：统计近N时间的数据。

基于窗口的操作都需要两个参数，分别为窗口时长以及滑动步长，两者都必须是

StreamContext 的批次间隔的整数倍。

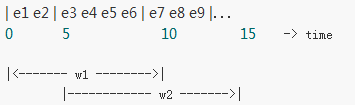
窗口时长：近N时间里的N；

滑动步长：多久计算一次。

窗口时长控制每次计算最近的多少个批次的数据，其实就是最近的windowDuration/batchInterval 个批次。

如果有一个以10 秒为批次间隔的源DStream，要创建一个最近30 秒的时间窗口（即最近3 个批次），就应当把windowDuration设为30 秒。而滑动步长的默认值与批次间隔相等，用来控制对新的DStream 进行计算的间隔。如果源DStream 批次间隔为10 秒，并且我们只希望每两个批次计算一次窗口结果，就应该把滑动步长设置为20 秒。

假如10s一个窗口，滑动5s，效果如下：



可见：窗口数据是有重叠的！属于“去头留尾”的计算。

窗口时长和步长必须是批次时长的整数倍。

代码如：

**val** linerdd = lines.map {row=>  
{  
 **val** arr = row.split(**","**)  
 **val** key = arr(3).substring(0,10)+**"\_"**+arr(0) *//2016-09-04\_Areaid* **val** amt = arr(2).toInt  
 (key,amt)  
}}.reduceByKeyAndWindow(  
 \_ + \_, *//加上新进入窗口的批次中的元素* \_ - \_, *//移除离开窗口的老批次中的元素  
 Minutes*(1), *//窗口时长  
 Seconds*(2), *//滑动步长*

2)

时间窗口只做sum型，不做count(distinct )，场景决定的。

## 按天、地区统计最近1分钟销售额

郑博 20:42:06

\_-\_为啥这样写，这两个不是窗口大小一样的

答：\_ -\_是函数，*移除离开窗口的老批次中的元素*

谢香梅 20:45:02

设置不同的时间窗口和滑动步长有什么好处

答：步长是多久计算一次。

## 去重型计算模板

按天统计购买人数，或者UV（独立访客数）

案例开发：按天统计UV

Select day, count(distinct guid) uv

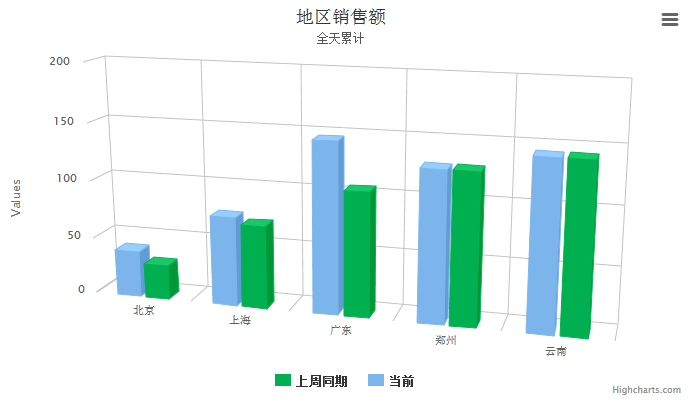
Group by day

实现思路：

把当天全部guid存储到**Set**（通过hashcode对元素进行去重）中。

## 综合项目需求说明

按天、地区（省份）计算实时销售额，5s延迟，图表展示。

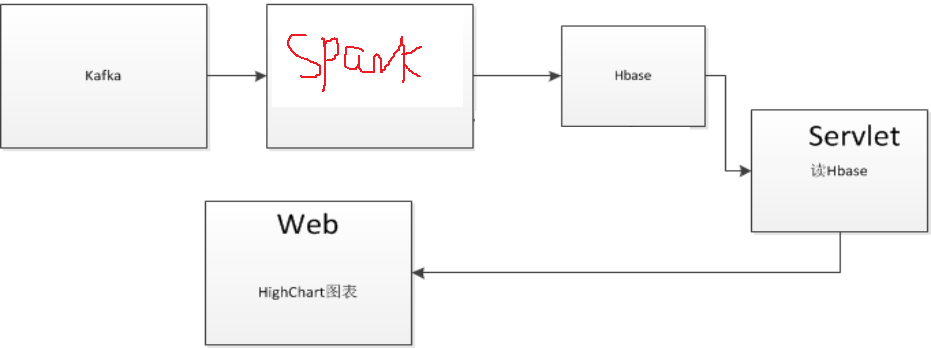


## 架构设计

后台：Spark Steaming进行实时计算

前台：Servlet、Jsp、Highcharts展示

需要做到前后台解耦，互补影响。



优点： 1、前台、后台分离，重启维护互不影响；

2、Spark重启不会影响结果数据，不影响HighCharts图表展示；

3、Tomcat重启不会影响Spark端数据处理；

4、前台后台数据传输通过Hbase，Hbase稳定性强，查询实时性好。

郑振宇 21:47:05

为何写hbase而不写到es里呢

答：HBase 是分布式数据库

ES：搜索引擎，在某些特殊场景中可以当数据库用。

在这个场景中根据你们技术储备和已有架构进行选择。

HBase + ES：HBase存储海量原始数据，ES存储查询条件项。

ES：读写吞吐量、集群稳定性不及HBase。

郑振宇 21:54:11

多少量级以下可以考虑es

答：从数据量和应用场景来综合考虑。

日增量10亿行+，就不建议把ES当数据库用，建议HBase+ES组合。