[第3章 Storm 192](#_Toc515544286)

[3.1 架构 193](#_Toc515544287)

[3.1.1 Nimbus 193](#_Toc515544288)

[3.1.2 Zookeeper 193](#_Toc515544289)

[3.1.3 supervisor 193](#_Toc515544290)

[3.1.4 worker 193](#_Toc515544291)

[3.2 编程模型 194](#_Toc515544292)

[3.2.1 Spout 194](#_Toc515544293)

[3.2.2 Bolt 194](#_Toc515544294)

[3.2.3 并行度 194](#_Toc515544295)

[3.2.4 消息不丢失 195](#_Toc515544296)

[3.3 Storm 物联网wifi项目 195](#_Toc515544297)

[3.3.1 wifi项目背景 195](#_Toc515544298)

[3.3.2 2)数据来源(采集) 196](#_Toc515544299)

[3.3.3 集群规模配置 197](#_Toc515544300)

[3.3.4 数据处理流程 197](#_Toc515544301)

[3.3.5 数据计算流程 199](#_Toc515544302)

[3.4 storm实时看板案例 199](#_Toc515544303)

[3.4.1 需求分析 199](#_Toc515544304)

[3.4.2 确定数据源 200](#_Toc515544305)

[3.4.3 确定采集方案 200](#_Toc515544306)

[3.4.4 确定存储 201](#_Toc515544307)

[3.4.5 数据计算 201](#_Toc515544308)

[3.4.6 展现 201](#_Toc515544309)

[3.5 storm日志监控告警系统 202](#_Toc515544310)

[3.5.1 需求分析 202](#_Toc515544311)

[3.5.2 确定数据源 202](#_Toc515544312)

[3.5.3 确定采集方案 202](#_Toc515544313)

[3.5.4 确定存储 203](#_Toc515544314)

[3.5.5 数据计算 203](#_Toc515544315)

[3.6 Storm的框架 203](#_Toc515544316)

[3.7 面试题 204](#_Toc515544317)

# Storm

storm是twitter公司开源贡献给apache的一款实时流式处理的一个开源软件，主要用于解决数据的实时计算以及实时的处理等方面的问题。Storm的特点：编程模型简单，可扩展，高可靠性，高容错性，支持多种编程语言，支持本地模式，高效。

## 架构

### Nimbus

负责资源分配和任务调度。新版本中的nimbus节点可以有多个，做主备

### Zookeeper

协调集群，公共数据的存放（如心跳数据，集群的状态和配置信息），nimbus将分配给Supervisor的任务写入到Zookeeper

### supervisor

负责接受nimbus分配的任务，启动和停止属于自己管理的worker进程。

### worker

运行具体处理组件逻辑的进程。worker中每一个spout/bolt的线程称为一个task. 在storm0.8之后，task不再与物理线程对应，同一个spout/bolt的task可能会共享一个物理线程，该线程称为executor。最新版本的Jstorm已经废除了task的概念

## C:\Users\admin\Desktop\20161218235943704.jpg编程模型

### Spout

Spout是接受外部数据源的组件，将外部数据源转化成Storm内部的数据，以Tuple为基本的传输单元下发给Bolt。（Tuple是Storm内部中数据传输的基本单元，里面封装了一个List对象，用来保存数据。）

### Bolt

Bolt是接受Spout发送的数据，或上游的bolt的发送的数据。根据业务逻辑进行处理。发送给下一个Bolt或者是存储到某种介质上。介质可以是mongodb或mysql，或者其他。

### 并行度C:\Users\admin\Desktop\fig-parallelism-1.png

Worker：表示一个进程

Executor：表示由worker启动的线程

Task：实际执行数据处理的最小工作单元（注意，task 并不是线程）

并行度的设置：评估上游kafka每秒生产的数据量，分析topic每个partition每秒的数据量，partition的数据量=SpoutTask接受数据量 SpoutTask数量=partition的数量

Worker的设置：如果数据量大，worker的数量等于spouttask的数量

### 消息不丢失

ack机制即， spout发送的每一条消息，

* 在规定的时间内，spout收到Acker的ack响应，即认为该tuple 被后续bolt成功处理
* 在规定的时间内，没有收到Acker的ack响应tuple，就触发fail动作，即认为该tuple处理失败，
* 或者收到Acker发送的fail响应tuple，也认为失败，触发fail动作。

通过Ack机制，spout发送出去的每一条消息，都可以确定是被成功处理或失败处理， 从而可以让开发者采取动作。比如在Meta中，成功被处理，即可更新偏移量，当失败时，重复发送数据。因此，通过Ack机制，很容易做到保证所有数据均被处理，一条都不漏。

## Storm 物联网wifi项目

### wifi项目背景

随着路由器上网的普及，越来越多的人在各个场合选择使用路由器上网，特别是在一些公共场所，例如网吧，酒店，饭店，旅馆，宾馆，洗浴中心等。这些公共场所的网络安全也日益受到各地网安的关注，各种问题也日益凸显。为此特为各地网安推出定制化的路由器，在网安指定的公共地点安装路由器，可以追踪每个人的上网情况，通过路由器或者嗅探设备的mac地址以及经纬度的追踪，可以定位每个人员的上网大致方位，了解每个人的上网内容，做到实时的网页内容监控，地理位置的监控，上网设备的mac地址追踪，通过嗅探设备，实现上网设备的实时路线追踪，为各地网安解决各种定制化的任务。

网安对于公共场所上网的人群，突出关注以下几点：

1. 所有流经路由器人群的mac地址
2. 上网人群的虚拟身份记录
3. 上网上线下记录
4. 搜索关键字记录
5. 网页访问记录
6. 地理位置记录

获取信息的条件:时速小于90KM/小时,并且开启无线网,嗅探设备就能抓到你的MAC地址

### 2)数据来源(采集)

设备铺设场所: 四川遂宁 吉林白山 四川眉山 内蒙赤峰 山东青岛

线上设备数量 8000+台 各个地方的设备往各个网安后台系统上报数据,网安后台使用

ES+mysql分表

上报数据的同时,通过ftp上报一份到我的大数据平台来,使用的是ES+hbase做实时查

询功能,hive整合HBase做批量统计功能

注意:关于物联网的项目,这些数据上报一般有两种方式.第一种手动上报,第二种定时上报,如果要做到实时处理数据,定时的时间肯定不长(几秒钟上传一次,每次上传数据量不大),这些数据要么通过ftp上传,要么通过socket接口传输数据

ftp服务器是外网,大数据集群是内网(内网关闭防火墙)

五种数据类型,FTP服务器开通多个FTP端口,多个FTP目录,用于接收多种设备,多种类

型的数据,减轻FTP服务器的压力,避免数据的堆积

### 集群规模配置

每天FTP接收数据量大概在100G左右

数据存储要求:要求近一年的数据可以随时查询到

服务器集群规划: 12台服务器组成的小型机集群

服务器硬件配置(最低,公司没钱):每台服务器12T硬盘配置

服务器内存配置:32G内存 4 台, 16G内存 8台

网络:外网百兆独享网络,内网千兆交换机

服务器CPU配置: 8核16线程4台 4核8线程8台

注意:服务器如果超过了50台,就形成了大型机集群,20-50台属于中型机集群

20台以内属于小型机集群,服务器内存一般给64G

### 数据处理流程

第一步:使用flume从FTP服务器拿到数据

第二步:flume与kafka整合,数据收集到kafka中

第三步:storm与kafka整合,读取kafka数据经过处理,匹配redis里的黑白名单,将数据存入实时报警监控系统中的mysql表中,并发送实时报警短信.

例:某一个MAC地址,或者手机号是redis黑名单里面的,哪个地区的MAC地址,手机号,

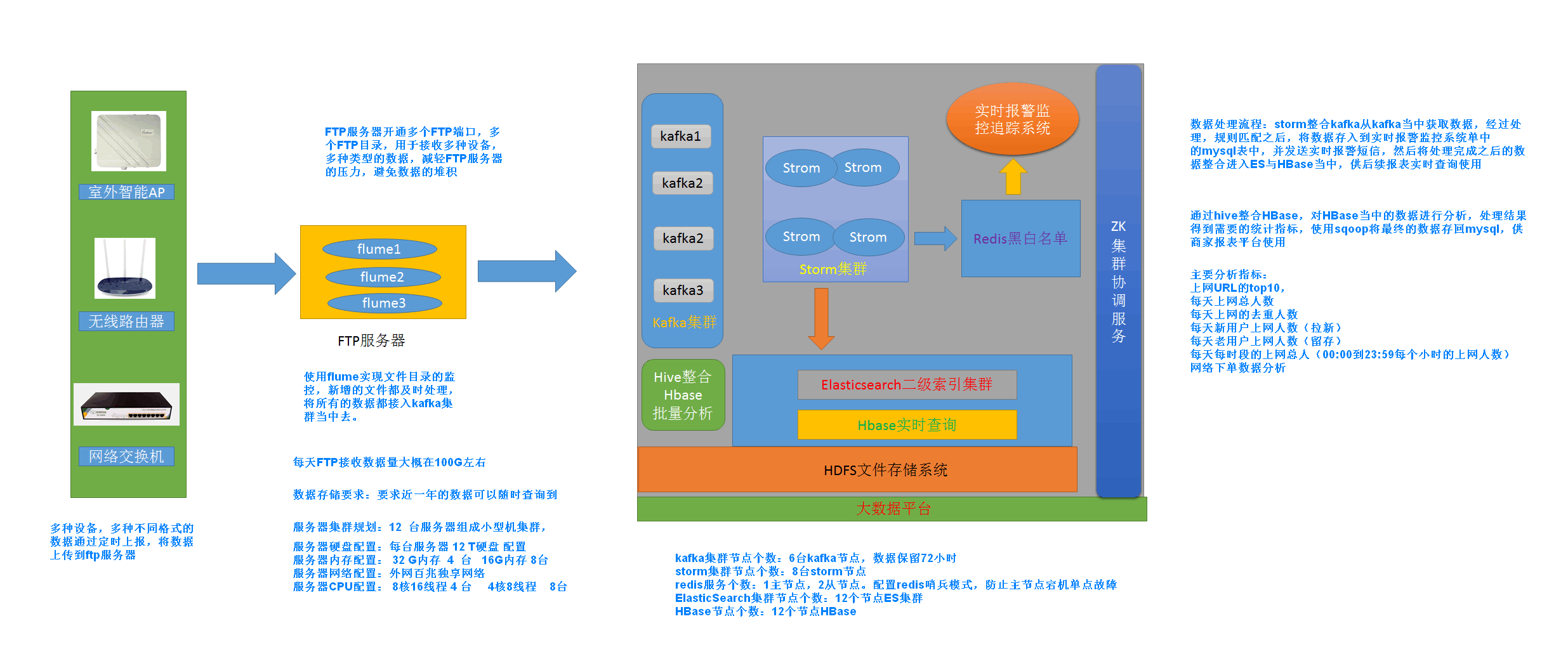
然后通过实时报警系统通知各地的网安

第四步:将处理完的数据整合进入ES与HBase当中,数据的关键信息存入ES集群中,具体的详细信息存入Hbase当中,供后续报表实时查询使用

注意:Hbase条件查询比较弱,如果需要带上各种查询条件,只能使用协处理器,协处理器底层全部是MapReduce.

所以把需要查询的条件字段全部存到ES(空间换时间)

第五步:hive和Hbase整合,实现数据仓库,数据离线分析功能



Kafka集群节点个数: 6台kafkfa节点,数据保留72小时

Storm集群节点个数:8台storm节点

Redis服务个数:1主节点,2从节点.配置redis哨兵模式,防止主节点宕机单点故障

ElasticSecarch集群节点个数:12个节点ES集群

Hbase节点个数:12个节点Hbase

指标分析

由于与网安合作不挣钱,竞争激烈,转到与美团小商家合作

通过hive整合Hbase,对Hbase当中的数据进行分析,处理结果得到需要的统计指标,使用sqoop将最终的数据存回mysql,供商家报表平台使用

主要指标:

上网URL的top10,

每天上网的总人数

每天上网的去重人数

每天新用户上网人数(拉新)

每天老用户上网人数(留存)

每天每时段的上网总人数(00:00到23:59每个小时的上网人数)

网络下单数据分析

### 数据计算流程

kafkaSpout------------------------处理kafka中的数据

WifiErrorBolt--------------------把分隔符去掉,把脏数据过滤掉

WiliWarningBolt--------------------监控黑白名单,告警(没有名单,功能未实现)

WifiTypeBolt--------------------------进行数据存储到HDFS上面去,要求实现五种数据类型按照长度分开存放

注意:数据写不到HDFS不同的目录,可以先写到本地文件系统,根据字段长度的不同,写到本地不同的文件目录里面去.

本地文件系统如何控制:

首先,写到128M的时候上传HDFS

其次,启动定时任务每隔几秒扫描一下昨天的目录,如果昨天的目录里面有文件,上传HDFS对应的文件夹里面去

## storm实时看板案例

### 需求分析

平台在活动促销日（例如双11）要求实时展示当日的一些销售信息

我们从三个维度去统计计算：

平台运维角度统计指标： 平台总销售额 平台下单人数 平台商品销售数量

商品销售角度统计指标： 每个商品总销售额 每个商品购买人数 每个商品销售数量

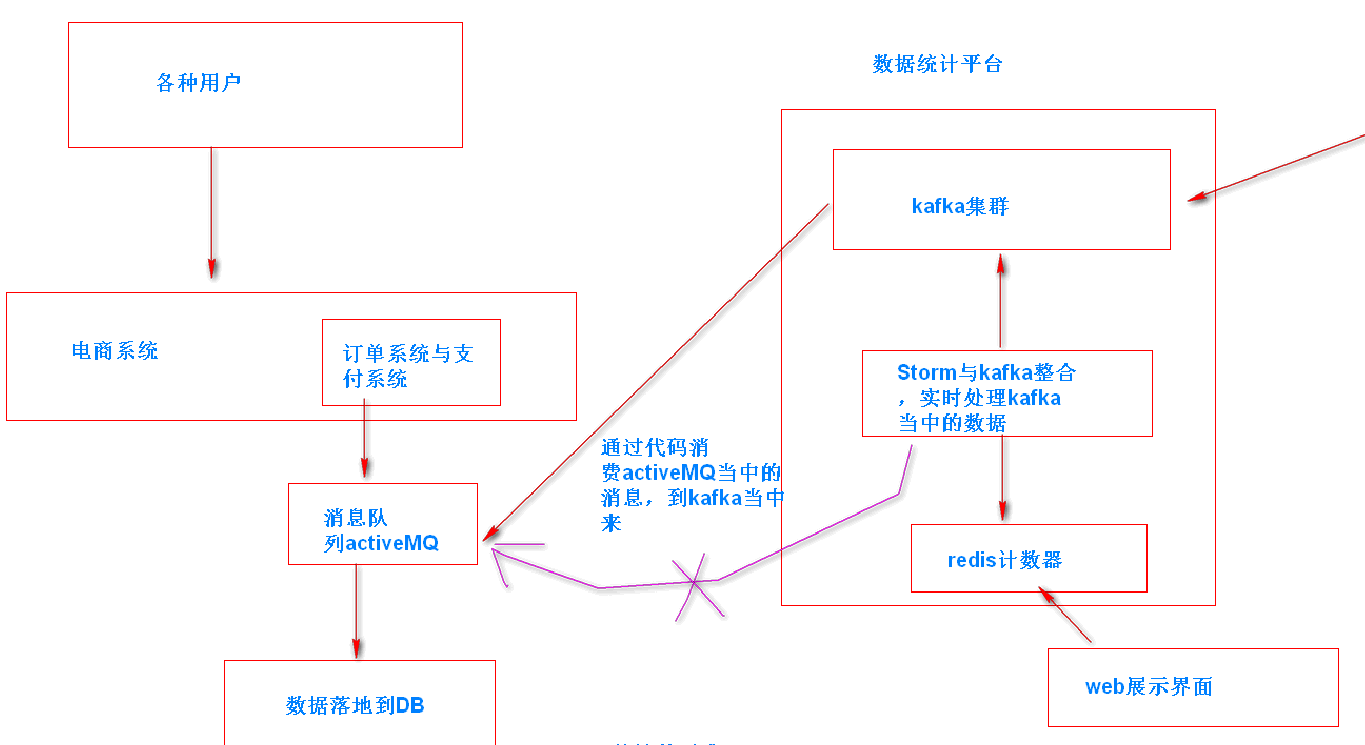
店铺销售角度统计指标： 每个店铺总销售额 每个店铺购买人数 每个店铺销售数量

### 确定数据源

数据源：订单系统与支付系统产生的数据。

### 确定采集方案

订单系统和支付系统产生的数据 通过ActiveMQ 到kafka 中，storm集成kafka获取数据。（**注**：为什么用ActiveMQ?1.电商业务系统当中需要用到事务支持，只能用比较严谨的jms系统来实现，所以考虑到用ActiveMQ。2.kafka直接读取数据库，会影响数据库速度。所以订单系统的数据通过ActiveMQ这个中间件来落地到数据库，kafka从ActiveMQ中间件中取数据，就可以避免直接读数据库的问题。如下图）



### 确定存储

我们将storm处理后的数据，通过Redis来累加计数并存储在Redis中。

### 数据计算

编写storm代码：storm与kafka整合来获取到数据，将每一条数据信息，利用Redis中**incrBy**这个命令来累加计数。并存储在Redis中。

### 展现

可以写一个定时器，定时的去Redis中获取结果数据，展示到web界面上。

## storm日志监控告警系统

### 需求分析

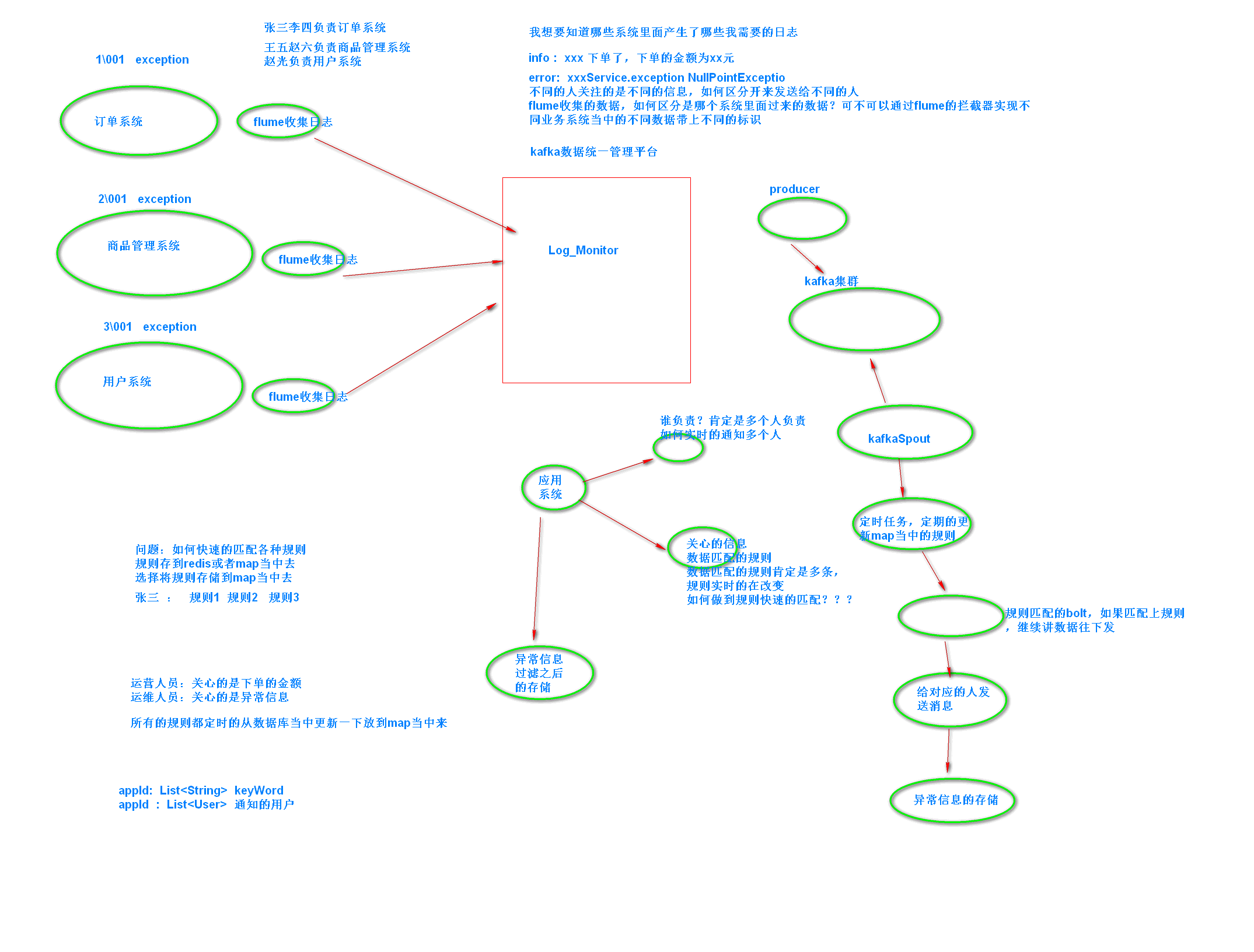
实现项目中日志监控的功能，需要做到日志监控实时告警，例如系统中出现任何异常，触发任何的告警规则，都可以实时通过短信或者邮件告知相关系统负责人。

### 确定数据源

每个业务系统（如: 订单系统，商品管理系统，用户系统）产生的日志信息

### 确定采集方案

将日志数据 通过 flume 采集到 kafka 中 ，storm集成kafka获取数据



### 确定存储

将触发告警规则的数据。存储在MySQL数据库中

### 数据计算

* 通过自定义flume拦截器，给不同系统产生的日志数据前加上一个appId来做唯一标识(不同系统对应的告警规则和负责人不同，所以这里要加一个唯一标识方便我们查找对应的规则和负责人)。
* storm代码编写步骤：
  1. 获取kafka中的数据
  2. 设置定时器，定时读取在MySQL数据库中的告警规则（数据库中的告警规则我们可以随时去修改，所以要采用定时器去读取，保证拿到是最新的规则）
  3. 将获取到的数据，与告警规则进行匹配，得到匹配成功的告警数据信息
  4. 将告警信息 以邮件，或者短信 方式发送给对应的负责人（对应人的信息，从MySQL数据库中获取）。
  5. 最后将警告信息存储到我们的MySQL数据库中。（方便以后查询异常记录）

## Storm的框架

1）storm的架构模型

2）storm的编程模型

3）storm的并行度

4）Storm的分发策略

5）Storm原理

6）storm与kafka集成

7）storm与hdfs的整合使用

8）消息不丢失机制

9）storm的定时器以及与mysql的整合使用

10）日志监控告警系统

## 面试题

##### 1）公司技术选型可能利用storm 进行实时计算,讲解一下storm

描述下storm的设计模式，是基于work、excutor、task的方式运行代码，由spout、bolt组成等等

##### 2）storm 如果碰上了复杂逻辑,需要算很长的时间,你怎么去优化,怎么保证实时性

拆分复杂的业务到多个bolt中，这样可以利用bolt的tree将速度提升

##### 3）Spark Streaming 和 Storm 有何区别？

一个实时毫秒一个准实时亚秒，不过 storm 的吞吐率比较低。