以下四个项目是基于 Storm 框架的小项目，实时计算部分必须使用 Storm，其他部分斟酌自选技术框架。

1. 语音“实时墙” —— 移动互联方向

（1）需求

- 将手机端 APP 的日常访问数据，按照用户访问的省份或者城市实时展示到页面展示系统。
- 数据量每天 1 亿，每秒峰值 20000
- 数据落地到数据展示的延时在 30 秒内

（2）数据

手机端 APP 访问日志数据见文件 1.txt，分隔符\t

字段解释：

序号	字段	类型	描述
0	reportTime	string	时间戳
1	msisdn	string	手机号码
2	apmac	string	AP Mac
3	acmac	string	Ac mac\访问 IP 地址
4	host	string	访问域名
5	siteType	string	网址类型
6	upPackNum	long	上行数据包数量，单位个
7	downPackNum	long	下行数据包数量，单位个
8	upPayLoad	long	上行总流量，单位 Byte
9	downPayLoad	long	下行总流量，单位 Byte
10	httpStatus	string	Http Response 的状态

注意：

请模拟该部分数据，将该部分数据量模拟增加到亿级

（3）IPV4 库

见《ip_area_isp_20131001.txt》和《GetArea.py》两个文件，第一个是 ip 和地址的映射，第二个是解析脚本。其中，第一个文件中的 ip 是换算后的 Long 型值。

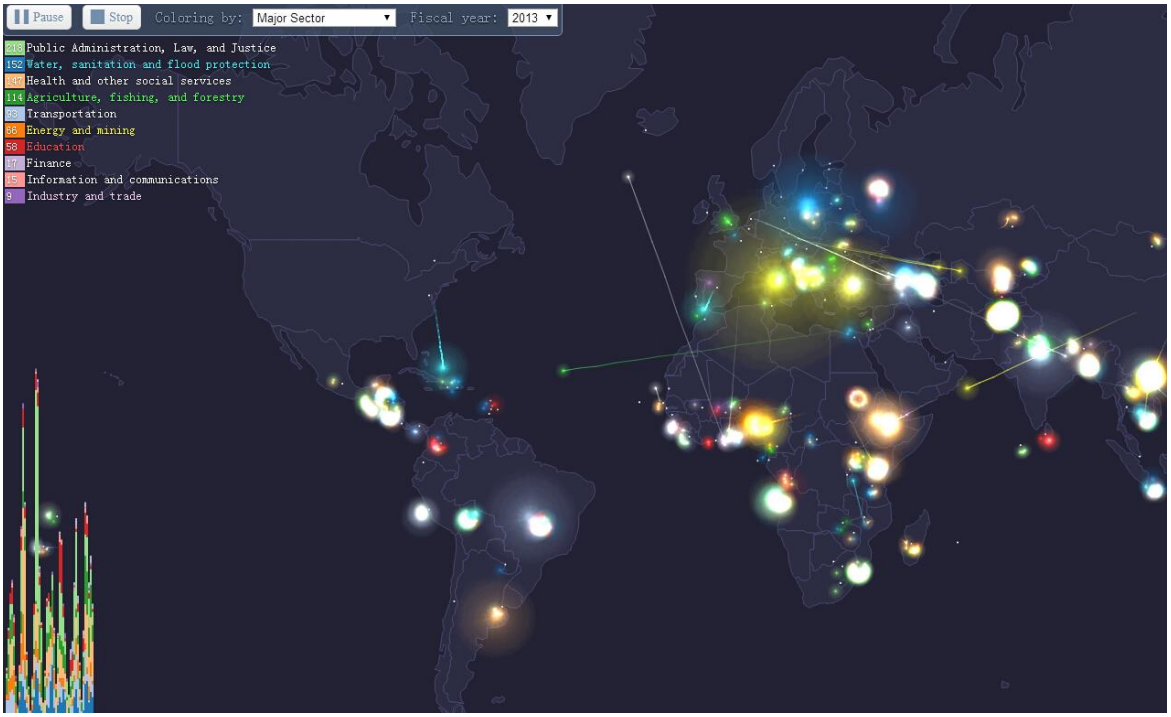
（4）WEB 页面展示

显示哪个城市有用户登录

D3.js

http://d3js.org/

采用 http://d3.artzub.com/wbca/ 这个页面的 js 效果展示数据，如下图所示



（5）产出：

- 概要设计文档：参照《HBase 中间层 v1.0-概要设计文档》中的规范，不少于 20 页
- 详细设计文档：如果项目中涉及到除去 hadoop、hive、hbase、storm、kafka 等之外的框架或者系统，请添加该框架的安装部署文档。（注：该部分只提供安装部署文档即可）
- 部署文档：部署脚本的说明文档。代码使用 github 托管，部署到 x86 服务器，编写部署 shell 脚本，该文档中详细阐述 shell 中的每个操作步骤；需要有运行部署的部分和验证是否部署成功的部分。
- 源码：给出 github 上的 link，必须是严格测试后的代码

2. 实时检测共享账户 —— 通信方向

（1）需求

- 运营商骨干网上采集现网流量流向信息，根据这些原始信息检测账号是否存在异常，如果多个终端使用同一个宽带账号，超过一定阈值则触发报警机制，例如阈值为 5 时，同一个账号同时连接的终端数量不能超过该值，如果超过则报警。
- 数据量每天 1000 亿，每秒峰值 100 000
- 5 分钟是一个周期，每个周期生成一个结果文件，每个周期检测一次共享账号

（2）数据

数据见文件 2.txt

字段解释：

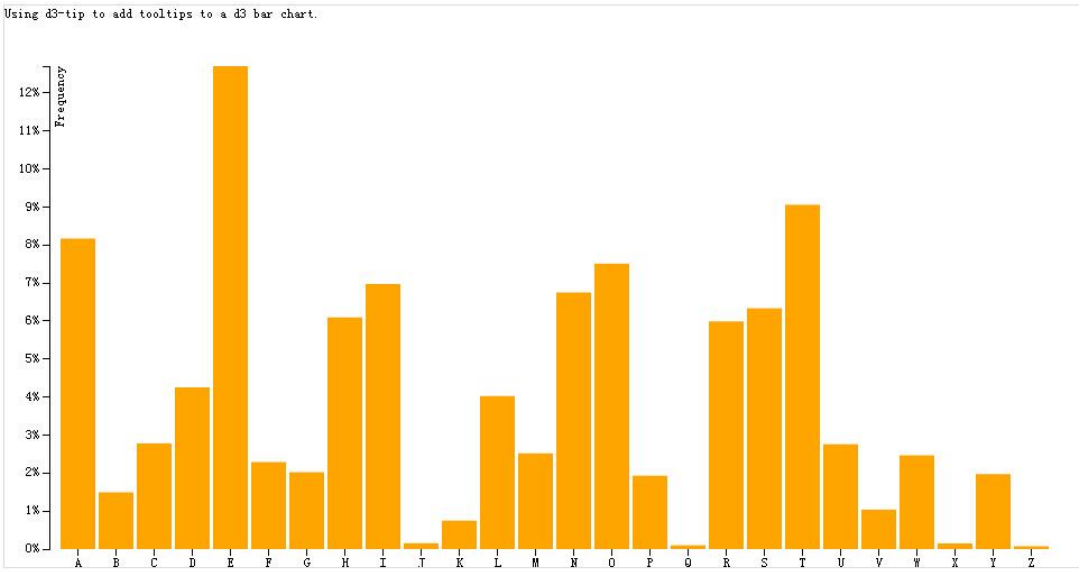
序号	字段	类型	描述
0	stime	Byte	数据统计开始时间
1	etime	Byte	数据统计结束时间
2	userAccount	Byte	宽带账户
3	userIP	Byte	用户以太网 IP
4	qqid	Byte	QQ 号
5	natIP	Byte	内网 IP
6	cookieValue	Byte	Cookie 值
7	devName	Byte	设备名称
8	osName	Byte	操作系统名称

注意：
请模拟该部分数据，将该部分数据量模拟增加到亿级，模拟 1 天数据

（3）检测方法
5 分钟内，同一个账号（UserAccount）下，只要满足下面的任意一个条件，表示出现共享账户问题：

- natIP 去重求和数 > 5
- qqid 去重求和数 > 20
- cookieValue + devName + osName 去重求和数 > 5

（4）统计需求
将 1 天中，每 5 分钟的异常账号总数绘制图表，如下图所示：
http://bl.ocks.org/Caged/6476579



（5）产出：

- 概要设计文档：参照《HBase 中间层 v1.0-概要设计文档》中的规范，不少于 20 页
- 详细设计文档：如果项目中涉及到除去 hadoop、hive、hbase、storm、kafka 等之外的框架或者系统，请添加该框架的安装部署文档。(注：该部分只提供安装部署文档即可)
- 部署文档：部署脚本的说明文档。代码使用 github 托管，部署到 x86 服务器，编写部署 shell 脚本，该文档中详细阐述 shell 中的每个操作步骤；需要有运行部署的部分和验证是否部署成功的部分。
- 源码：给出 github 上的 link，必须是严格测试后的代码

3 基于 GPS 数据的实时路况特征分析 —— GIS 方向

（1）需求

- 实时 GPS 数据客流特征分析系统，数据来源是~某市 20 万辆出租车、公交车的车载 GPS，其目的是要研究出行者的出行特征、实时路况、客流特征等。
- GPS 每 30 秒上报一次数据，需要模拟数据上报过程，可以使用 syslog

（2）数据

数据见文件 3.txt

字段解释：

序号	字段	类型	描述
0	vehicle_number	String	车牌号
1	longitude	String	经度
2	lantitude	String	纬度
3	date_time	String	时间
4	speed	String	速度
5	bearing	String	方向
6	occupied	String	车辆是否使用中，0 表示使用，1 表示未用，其他表示异常

注意：

请模拟该部分数据，将该部分数据量模拟增加到千万量级，模拟 1 小时数据

（3）实时路况分析方法

- 根据数据提取经度和纬度，调用 Sects 类 Sect.fetchSect(GPSrecord)方法，查询本地的地理信息数据库，返回该条 GPS 记录所在的区域标号 districtID，然后将该区域的计算值加 1。

- 注意：Sects 类调用了开源的地理信息系统工具 geotools，感兴趣的朋友可以去 http://www.geotools.org/ 下载安装包，并将相关的 jar 包全部添加到 Eclipse 的 building path 里面，就可以调用 geotools 查询本地的地理信息数据库了。

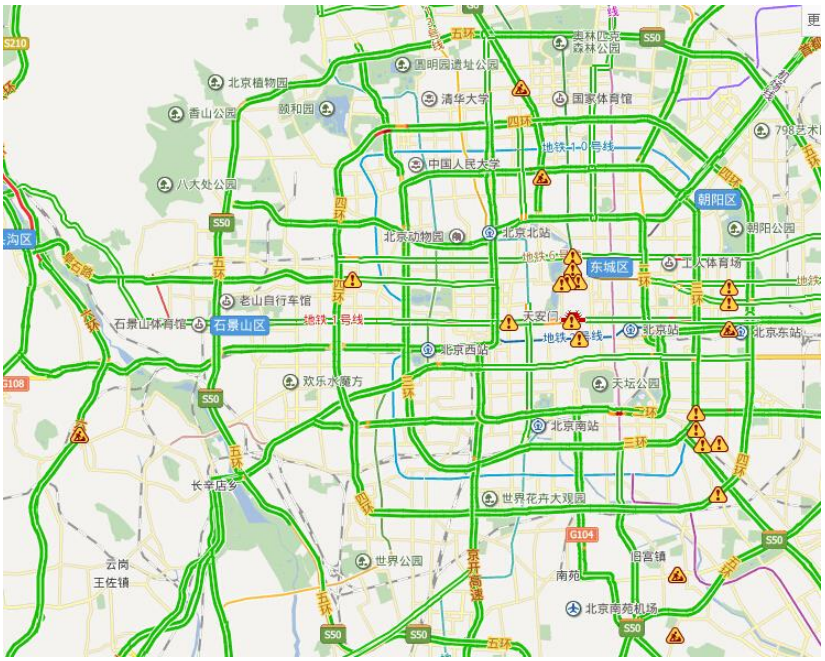
- 每分钟统计所有区域的车辆情况，汇总完成后，整个区域值清零

（4）统计需求

展示当前整个城市的道路实时路况图，每分钟刷新一次页面，js 框架在下面列表选择其一即可：

- http://jquerygeo.com/
- http://leafletjs.com/examples.html
- http://echarts.baidu.com/doc/example/map3.html

下图是一个百度地图实时路况的示例，并不要求达到该效果，只要按照区域编号将颜色作出标识即可。



（5）产出：

- 概要设计文档：参照《HBase 中间层 v1.0-概要设计文档》中的规范，不少于 20 页
- 详细设计文档：如果项目中涉及到除去 hadoop、hive、hbase、storm、kafka 等之外的框架或者系统，请添加该框架的安装部署文档。（注：该部分只提供安装部署文档即可）
- 部署文档：部署脚本的说明文档。代码使用 github 托管，部署到 x86 服务器，编写部署 shell 脚本，该文档中详细阐述 shell 中的每个操作步骤；需要有运行部署的部分和验

证是否部署成功的部分。

- 源码：给出 github 上的 link，必须是严格测试后的代码

4. 数据质量监控系统 —— 互联网方向

（1）需求

- 将手机端 APP 的日常访问数据，监控接收数据的总条数、手机号为空条数两个参数。具体的以每周为一个周期，使用 2 周数据相互对比（即同比上周）
- 数据量每天 1 亿，每秒峰值 20000
- 数据落地到数据展示的延时在 30 秒内

（2）数据

手机端 APP 访问日志数据见文件 1.txt，分隔符\t

字段解释：

序号	字段	类型	描述
0	reportTime	string	时间戳
1	msisdn	string	手机号码
2	apmac	string	AP Mac
3	acmac	string	Ac mac\访问 IP 地址
4	host	string	访问域名
5	siteType	string	网址类型
6	upPackNum	long	上行数据包数量，单位个
7	downPackNum	long	下行数据包数量，单位个
8	upPayLoad	long	上行总流量，单位 Byte
9	downPayLoad	long	下行总流量，单位 Byte
10	httpStatus	string	Http Response 的状态

注意：

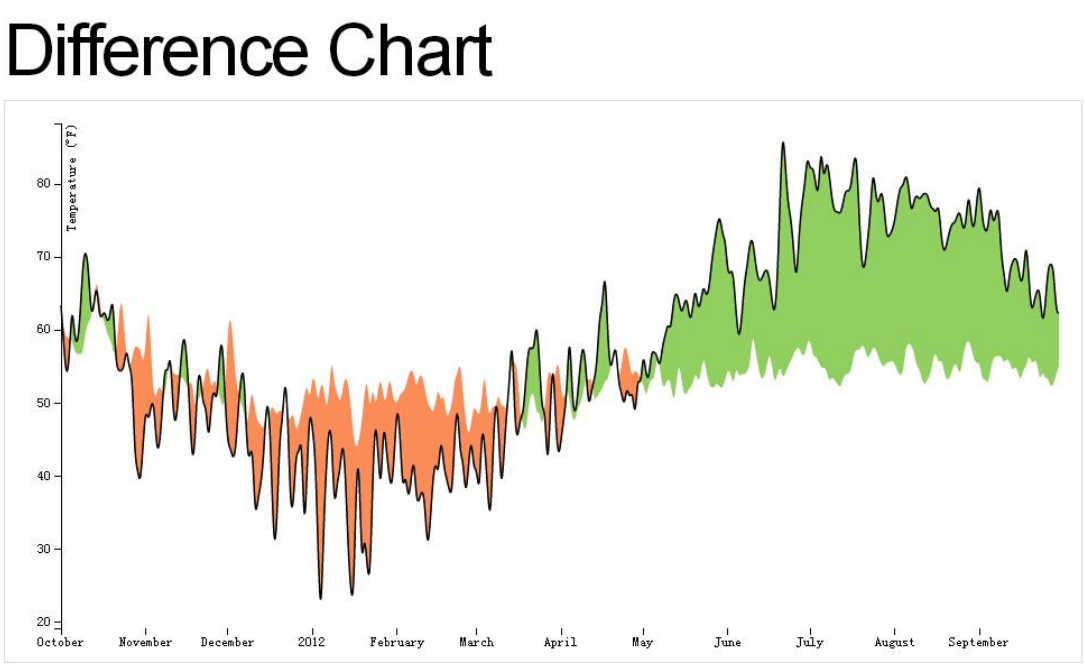
请模拟该部分数据，将该部分数据量模拟增加到亿级，要模拟连续 14 天的均匀数据，每天必须包含部分手机号为空的数据

（3）WEB 页面展示

D3.js

http://d3js.org/

采用 http://bl.ocks.org/mbostock/3894205 这个页面的 js 效果展示数据，如下图所示



要求：可以按天展示数据，连续 7 天的数据

（4）产出：

- 概要设计文档：参照《HBase 中间层 v1.0-概要设计文档》中的规范，不少于 20 页
- 详细设计文档：如果项目中涉及到除去 hadoop、hive、hbase、storm、kafka 等之外的框架或者系统，请添加该框架的安装部署文档。（注：该部分只提供安装部署文档即可）
- 部署文档：部署脚本的说明文档。代码使用 github 托管，部署到 x86 服务器，编写部署 shell 脚本，该文档中详细阐述 shell 中的每个操作步骤；需要有运行部署的部分和验证是否部署成功的部分。
- 源码：给出 github 上的 link，必须是严格测试后的代码