# 1. 项目背景

网站、app的运营者需要知道自己的产品或服务的运营状况，就需要对使用自己产品的用户进行各种角度的数据分析，比如：

用户数量

新增用户

留存用户

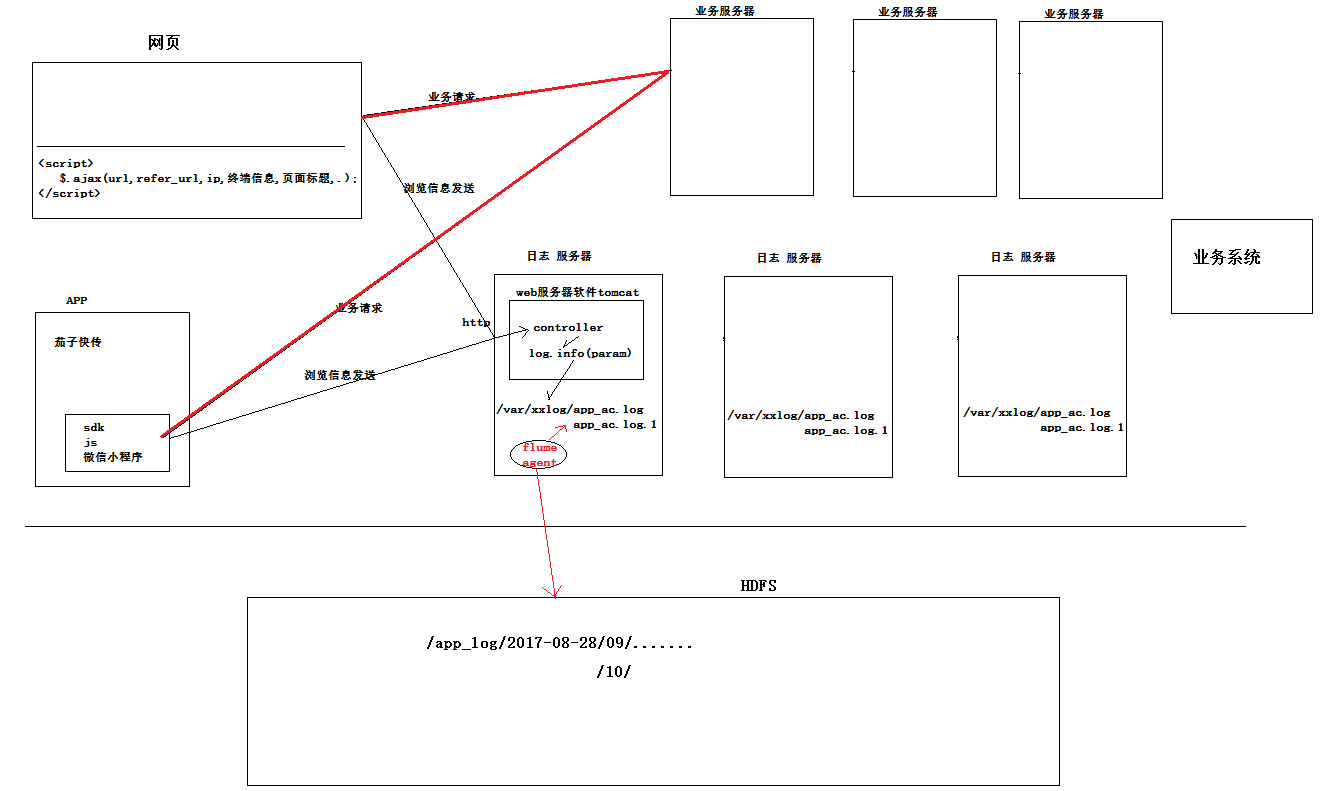
活跃用户

地域分析

渠道分析

.......

要做这样的分析，数据来源应该是用户的产品使用的行为日志，行为日志是由app或者网站的页面获取用户相关信息后，发送给后台服务器记录下来的：



# 项目开发--模块1（数据预处理）

**需求：**

1. 检查每条日志的必选字段是否完整，不完整的日志应该滤除（分号标记表示必选字段）

(

cid\_sn ;

mobile\_data\_type

os\_ver

mac ;

resolution

commit\_time ;

sdk\_ver ;

device\_id\_type ;

city ;

device\_model ;

android\_id

carrier

promotion\_channel

app\_ver\_name ;

imei ;

app\_ver\_code ;

pid

net\_type

device\_id ;

app\_device\_id

release\_channel ; ## 用户下载该app时所用的app应用市场：360，安智市场，

country ;

time\_zone ;

os\_name ; ## 操作系统名称

manufacture ## 手机生产厂家

commit\_id ; ## 提交请求的序号

account ## 账号

app\_token ; ## app名称：

app\_id ; ## app的id标识(所属的公司事业部)

language ; ## 用户的操作系统语言（）

build\_num

)

2、为每条日志添加一个用户唯一标识字段：user\_id

user\_id的取值逻辑：

如果是ios设备，user\_id=device\_id

如果是android设备， user\_id = android\_id

如果android\_id为空，则user\_id = device\_id

1. 将event字段抛弃，将header中的各字段解析成普通文本行

主要技术点：json解析 gson/fastjson/jackson/......

一个非主流需求：

需要将清洗后的结果数据，分ios和android和其他 三种类别，输出到3个不同的文件夹；

/app\_clean\_log/2017-09-20/ios/part-r-00000

/ios/part-r-00000

/ios/part-r-00001

/android/part-r-00000

/android/part-r-00001

/other/part-r-00000

/other/part-r-00001

实现方法：multipleOutputs

# 数据入库

提示：

1. 在hive中建表ods\_app\_log（外部表、分区表）映射预处理阶段生成的清洗数据
2. 每天定时将预处理之后的天数据 导入到 ods\_app\_log的天分区 中

# 新增用户分析

新增用户的定义：

比如，在2017-08-28日出现了一些以前从没出现过的用户，则这些用户就是2017-08-28日的新增用户

需求：

1. 将每日的新增用户从ods\_app\_log表中抽取出来，存入一个新用户信息表：

dw\_new\_user\_day的日分区中

2、统计如下报表：

某日 城市 渠道 版本 新增用户数

2017-08-28 all all all ?

2017-08-28 具体城市 all all ?

2017-08-28 all 具体渠道 all ?

2017-08-28 all all 具体版本 ?

2017-08-28 all 具体渠道 具体版本 ?

2017-08-28 具体城市 all 具体版本 ?

2017-08-28 具体城市 具体渠道 all ?

2017-08-28 具体城市 具体渠道 具体版本 ?

# 5. 活跃用户（日活）

概念：某一天使用过app的用户就是活跃用户

计算过程：

* 源表——ods\_app\_log
* 目标表：日活表

create table etl\_user\_active\_day like ods\_app\_log;

计算逻辑：

# 6. 留存用户统计

概念：

比如，15号的新增用户，在16号又活跃了，这些用户就是次日留存用户;

比如，12号的新增用户，在15号又活跃了，这些用户就是3日留存用户;

做次日留存的分析：

思路

——ETL：先抽取出次日留存用户，存入一个次日留存用户信息表；

——维度分析：统计各种维度下的留存用户数、留存用户比例

# 8. 沉默用户分析

概念：一段时间内（连续7天）没有使用过app的用户

思路：

比如，现在运算的是20号的报表，

用13号的新增用户 left join 活跃用户表的（14-20号分区）

取右表join后结果为null的用户

# 9. 版本升级轨迹

需求：每天统计出如下报表

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 日期 | user\_id | app\_token | channel | city | source\_ver | curr\_ver |

*示例：*

*2017-08-14,许老师,共享女友,360应用,北京,v1.0*

*2017-08-14,赵老师,共享女友,安智市场,北京,v1.2*

*2017-08-14,许老师,共享女友,360应用,天津,v1.2*

*2017-08-14,许老师,共享女友,小米应用,天津,v2.0*

*2017-08-15,许老师,共享女友,360应用,北京,v2.0*

*2017-08-15,赵老师,共享女友,安智市场,北京,v1.2*

*2017-08-15,赵老师,共享女友,安智市场,北京,v1.5*

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| *2017-08-14* | *许老师* | *共享女友* | *360应用* | *天津* | *v1.0* | *v1.2* |
| *2017-08-14* | *许老师* | *共享女友* | *小米应用* | *天津* | *v1.2* | *v2.0* |

# 补充：hive的窗口分析函数

上述例子中的需求，就可以按如下步骤解决：

|  |
| --- |
| 2017-08-14,许老师,共享女友,360应用,北京,v1.0  2017-08-14,赵老师,共享女友,安智市场,北京,v1.2  2017-08-14,许老师,共享女友,360应用,天津,v1.2  2017-08-14,许老师,共享女友,小米应用,天津,v2.0  2017-08-15,许老师,共享女友,360应用,北京,v2.0  2017-08-15,赵老师,共享女友,安智市场,北京,v1.2  2017-08-15,赵老师,共享女友,安智市场,北京,v1.5  create table t\_ver(day string,user\_id string,app\_token string,channel string,city string,version string)  row format delimited  fields terminated by ',';  select tmp.\*  from  (select  '2017-08-14',  user\_id,  version,  lag(version,1,null) over(partition by user\_id,app\_token order by version) as pre\_version  from t\_ver) tmp  where tmp.pre\_version is not null; |

更多窗口分析函数使用方法，请参考《HIVE窗口分析函数.sql》