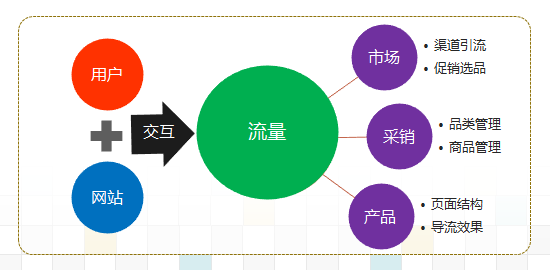
**网站流量分析介绍**

# 一、流量介绍

流量是用户与网站交互产生的点击流数据的反应，是业务与用户的营销接触点，是业务红利的入口 ，流量数据通过用户行为反应营销漏斗过程。通过对网站各种流量指标的监控分析，可以对整个网站、不同业务、网站页面的不同区域的用户活跃情况进行精细化的统计，从而检查目前运营状况，进而对运营规划、产品优化、市场分析作出决策。

[](http://biwiki.bdp.jd.com/w/index.php?title=%E6%96%87%E4%BB%B6:Pc1.png)

# 二、流量基本指标

数量指标(页面浏览量、访次、独立访客数、独立页面访问数)

**页面浏览量（pv）**

**定义**：页面被浏览次数的总和，统计一个网站时，指的是网站所有页面被点击的总次数。 从技术层面讲，指浏览器加载网页的次数总和。   
**公式：**pv=count(1)

**访问次数(visits)**

**定义：**从来到网站到离开，算一次统计。离开后，再次重新访问，访问量按2次算。如果一直没有关掉网站，访问网站的不同页面，算一次访问。一天内不同时间访问网站N次，算N次访问。   
**公式：**visits=count(distinct session\_id)

**独立访客数(uv)**

**定义：**指不同的、通过互联网访问这个网页的自然人。 从技术层面上讲，用Cookie 来确定独立访客数，在浏览器cookie数据不清除的情况下，即使用多个IP切换来登录一个网站，也会只记为一个访客数。 独立访客很接近但并不完全就是真实独立的人。   
**公式：**uv=count(distinct browser\_uniq\_id)

**独立页面访问数(upv)**

**定义：**UPV是针对页面这个维度而言的访问次数（一般访问次数都是针对访客的），即有多少个访问次数（Visits）经过这个页面，就是这个页面的UPV，Upv主要是避免页面的重复加载和刷新导致Pv虚高的情况，所以在同一个Visit当中重复打开同一个页面，该页面的upv始终只被记为1次。   
**公式：**upv=sum( case when url\_request\_seq\_num = 1 then 1 else 0 end )

质量指标(跳出率、退出率、平均页面停留时间、平均网站停留时间)

**跳出率(Bounce Rate)**

**定义：**只浏览了一个页面便离开网站的访问次数占总访问次数的百分比。   
跳出次数:Bounces，指进入网站后，不点击页面上的任何链接，就关闭它，那么对该网站而言就是一个Bounce（跳出） bounces=sum( case when first\_request\_flag = 1 and last\_request\_flag = 1 then 1 else 0 end )   
**公式：**跳出率=跳出次数/访问次数.

**退出率(Exits Rate)**

**定义：**访客在一个页面离开网站的次数占该页面浏览量的百分比。   
**公式：**退出率=退出次数/PV sum(case when last\_request\_flag = 1 then 1 else 0 end) as exits

**平均页面停留时间(Avg\_Page\_Rt)**

**定义：**特定网页上的停留时间总和除以该页面的浏览量   
**公式：** Avg\_Page\_Rt =avg(case when stm\_rt > 0 and stm\_rt < 1800 then stm\_rt end)

**平均网站停留时间(Avg\_Site\_Rt)**

**定义：**全站所有访次的停留时间总和除以全站总访次   
**公式：** Avg\_Site\_Rt =avg(case when last\_request\_flag = 1 and session\_rt >= 1 then session\_rt end)

转化指标(转化率)

不同业务的定义可能不一样，目前较常用的定义有：订单量/访次，订单量/uv

# 三、埋点方法

埋点，即在需要监控的站点页面植入专用js代码，当刷新页面和点击埋点区域时，会触发js代码中监控模块异步发送get请求至点击流系统，实现对流量指标信息的记录。

因此，埋点可分为两种：浏览埋点和点击埋点。如果想统计页面的pv、uv、访次等指标，请使用浏览埋点；如果想统计页面某些区域（如广告、按钮）的点击量，请使用点击埋点。浏览埋点只需要在页面植入专用js即可，点击埋点的流量指标统计则需要在植入js的基础上，还需在相应html对象上加clstag（自己定义）属性，用于点击触发时自动识别该标签，记录固定格式日志。

示例：

<"div clstag="pageclick|keycount|home2013|08a">

参数说明:   
clstag标签格式:pageclick|keycount|clstag\_log\_id1|clstag\_log\_id2   
其中pageclick|keycount 为固定值，clstag\_log\_id1 和clstag\_log\_id2 clstag标签以| 分隔，后两个字段确定该clstag日志的类型（如示例中的home2013 和08a ），由点击流系统生成。

# 四、架构

接着上节提到触发埋点后发送get请求至点击流系统，点击流系统接收到请求后，对请求进行解析并补充诸如服务器时间、访问者IP等信息，然后将信息按固定格式发送至消息队列Kafka，并同时写入磁盘备份。日志会在Kafka上保留一天，本地硬盘会保留最近两天日志，更早期的日志会放入专用服务器。

整体架构如下：

# 五、流量分析展示

前面做了大量的工作，最后还得形成具有一定意义的报表展示出来，以供分析从而更好的反哺业务和产品。从技术层面讲，由于流量从用户触发，到点击流系统，再到大数据平台再到数据集市层，要经过多次的分析和加工处理，而对于京东这样的网站流量是相当巨大的，所以要做到对当日数据实时分析是相对比较困难的。所以从架构中可以看出，当日数据是直接从Kafka 队列中获取，经storm计算后加工成我们最终想要的结果存入mysql中。

**离线数据：**大数据平台会定时爬取日志，经过分析加工后推到数据集市层，数据需求者从集市按需抽取数据用于离线数据分析和报表展示。

**实时数据：**实时数据的通过storm接入Kafka消息队列，经过加工后形成报表展示。