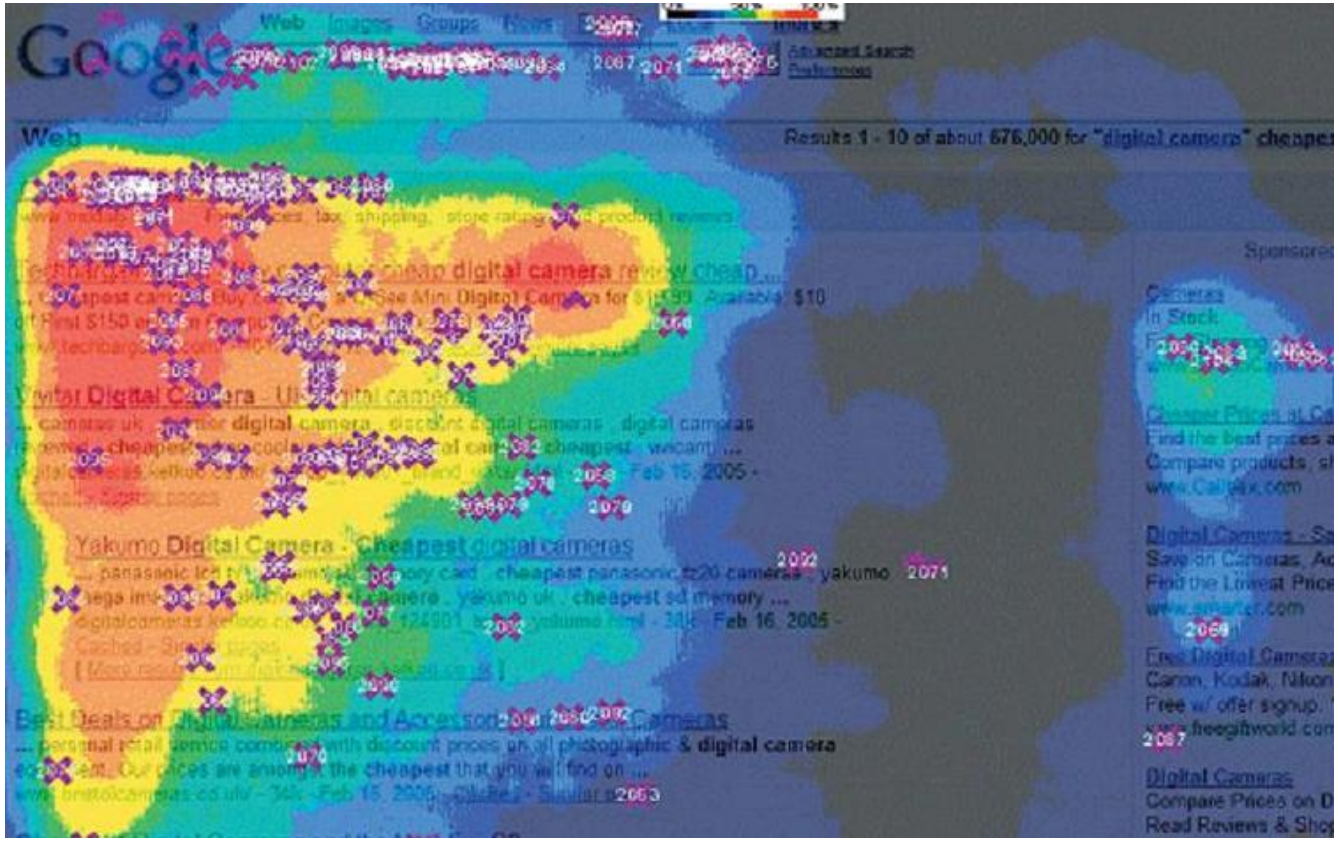
DATAGURU 炼数成金



Hadoop应用开发实战案例 第2周

法律声明

【声明】本视频和幻灯片为炼数成金网络课程的教学资料，所有资料只能在课程内使用，不得在课程以外范围散播，违者将可能被追究法律和经济责任。

课程详情访问炼数成金培训网站

http://edu.dataguru.cn

课程内容

- 项目背景：Web日志分析
- 需求分析：KPI指标
- 架构设计：日志分析系统架构
- 算法模型：Map-Reduce并行算法
- 程序开发：
  - 用Maven构建Hadoop项目
  - MapReduce程序实现

项目背景: Web日志分析概述

- Web日志由Web服务器产生，可能是Nginx, Apache, Tomcat等。从Web日志中，我们可以获取网站每个页面的PV值（PageView，页面访问量）、独立IP数；
- 稍微复杂一些的，可以计算得出用户所检索的关键词排行榜、用户停留时间最高的页面等；
- 更复杂的，构建广告点击模型、分析用户行为特征等等。

项目背景: Nginx日志

- Web日志中，每条日志通常代表着用户的一次访问行为

- 例如: 下面就是一条nginx日志
- 222.68.172.190 - - [18/Sep/2013:06:49:57 +0000] "GET /images/my.jpg HTTP/1.1" 200 19939 "http://www.angularjs.cn/A00n" "Mozilla/5.0 (Windows NT 6.1) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/29.0.1547.66 Safari/537.36"

Apache日志样式

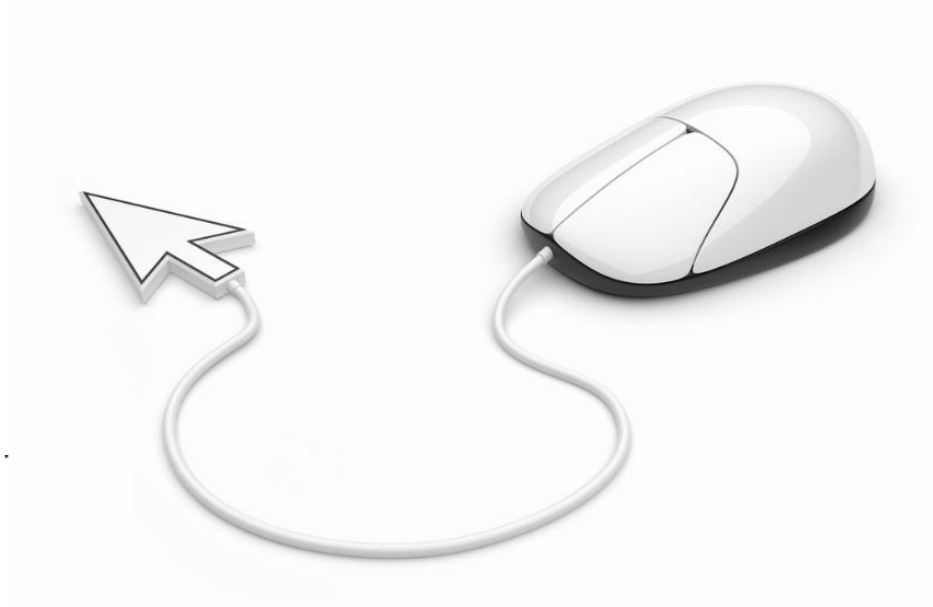
```
xmenu=1&inajax=1" "Mozilla/4.0 (compatible; MSIE 7.0; Windows NT 5.1; .NET CLR 1.1.4322; .NET CLR 2.0.50727; .NET CLR 3.0.045
06.30)"
183.3.51.76 - - [29/Nov/2013:01:27:25 +0800] "GET /member.php?mod=logging&action=login HTTP/1.1" 200 17707 "http://r.dataguru
.cn/member.php?mod=logging&action=login" "Mozilla/4.0 (compatible; MSIE 7.0; Windows NT 5.1; .NET CLR 1.1.4322; .NET CLR 2.0.
50727; .NET CLR 3.0.04506.30)"
183.3.51.76 - - [29/Nov/2013:01:27:26 +0800] "GET /member.php?mod=logging&action=login HTTP/1.1" 200 17707 "http://r.dataguru
.cn/member.php?mod=logging&action=login" "Mozilla/4.0 (compatible; MSIE 7.0; Windows NT 5.1; .NET CLR 1.1.4322; .NET CLR 2.0.
50727; .NET CLR 3.0.04506.30)"
183.3.51.76 - - [29/Nov/2013:01:27:26 +0800] "POST /member.php?mod=logging&action=login&loginsubmit=yes&inajax=1&ajaxmenu=1 H
TTP/1.1" 200 297 "http://r.dataguru.cn/member.php?mod=logging&action=login&loginsubmit=yes&inajax=1&ajaxmenu=1" "Mozilla/4.0
(compatible; MSIE 7.0; Windows NT 5.1; .NET CLR 1.1.4322; .NET CLR 2.0.50727; .NET CLR 3.0.04506.30)"
66.249.64.1 - - [29/Nov/2013:01:30:19 +0800] "GET /home.php?mod=space&uid=50144&do=home&view=me&from=space HTTP/1.1" 200 5769
 "-" "Mozilla/5.0 (iPhone; CPU iPhone OS 6_0 like Mac OS X) AppleWebKit/536.26 (KHTML, like Gecko) Version/6.0 Mobile/10A5376
e Safari/8536.25 (compatible; Googlebot-Mobile/2.1; +http://www.google.com/bot.html)"
66.249.64.8 - - [29/Nov/2013:01:30:44 +0800] "GET /space-uid-73446.html HTTP/1.1" 200 4782 "-" "Mozilla/5.0 (compatible; Goog
lebot/2.1; +http://www.google.com/bot.html)"
210.51.177.136 - - [29/Nov/2013:01:35:28 +0800] "GET / HTTP/1.0" 200 46531 "-" "User-Agent: Mozilla/5.0 (compatible; MSIE 6.0
;Windows XP)"
66.249.64.1 - - [29/Nov/2013:01:36:52 +0800] "GET /space-uid-73384.html HTTP/1.1" 200 4776 "-" "Mozilla/5.0 (compatible; Goog
lebot/2.1; +http://www.google.com/bot.html)"
66.249.64.1 - - [29/Nov/2013:01:38:25 +0800] "GET /space-uid-73345.html HTTP/1.1" 200 4434 "-" "Mozilla/5.0 (compatible; Goog
lebot/2.1; +http://www.google.com/bot.html)"
183.3.20.129 - - [29/Nov/2013:01:38:45 +0800] "GET /member.php?mod=logging&action=login HTTP/1.1" 200 17707 "http://r.datagur
u.cn/member.php?mod=logging&action=login" "Mozilla/4.0 (compatible; MSIE 7.0; Windows NT 5.1; .NET CLR 1.1.4322; .NET CLR 2.0
.50727; .NET CLR 3.0.04506.30)"
183.3.20.129 - - [29/Nov/2013:01:38:49 +0800] "GET /member.php?mod=logging&action=login HTTP/1.1" 200 17707 "http://r.datagur
u.cn/member.php?mod=logging&action=login" "Mozilla/4.0 (compatible; MSIE 7.0; Windows NT 5.1; .NET CLR 1.1.4322; .NET CLR 2.0
.50727; .NET CLR 3.0.04506.30)"
183.3.20.129 - - [29/Nov/2013:01:38:49 +0800] "POST /member.php?mod=logging&action=login&loginsubmit=yes&inajax=1&ajaxmenu=1
HTTP/1.1" 200 297 "http://r.dataguru.cn/member.php?mod=logging&action=login&loginsubmit=yes&inajax=1&ajaxmenu=1" "Mozilla/4.0
 (compatible; MSIE 7.0; Windows NT 5.1; .NET CLR 1.1.4322; .NET CLR 2.0.50727; .NET CLR 3.0.04506.30)"
[root@class2room web_logs]#
```

探针设计

```
<script type="text/javascript">

  var _gaq = _gaq || [];
  _gaq.push(['_setAccount', 'UA-20237423-4']);
  _gaq.push(['_setDomainName', '.itpub.net']);
  _gaq.push(['_trackPageview']);

  (function() {
    var ga = document.createElement('script'); ga.type = 'text/javascript'; ga.async = true;
    ga.src = ('https:' == document.location.protocol ? 'https://ssl' : 'http://www') + '.google-ar
    var s = document.getElementsByTagName('script')[0]; s.parentNode.insertBefore(ga, s);
  })();

</script>
<div style="display:none">
<script type="text/javascript">
var _bdhmProtocol = (("https:" == document.location.protocol) ? " https://" : " http://");
document.write(unescape("%3Cscript src='" + _bdhmProtocol + "hm.baidu.com/h.js%3F5016281862f595e7
</script></div>
<!-- END STAT PV --></body>
</html>
```

排除爬虫和程序点击，对抗作弊

- 用鼠标测动对抗爬虫
- 常用流量作弊手段
- 跟踪用户

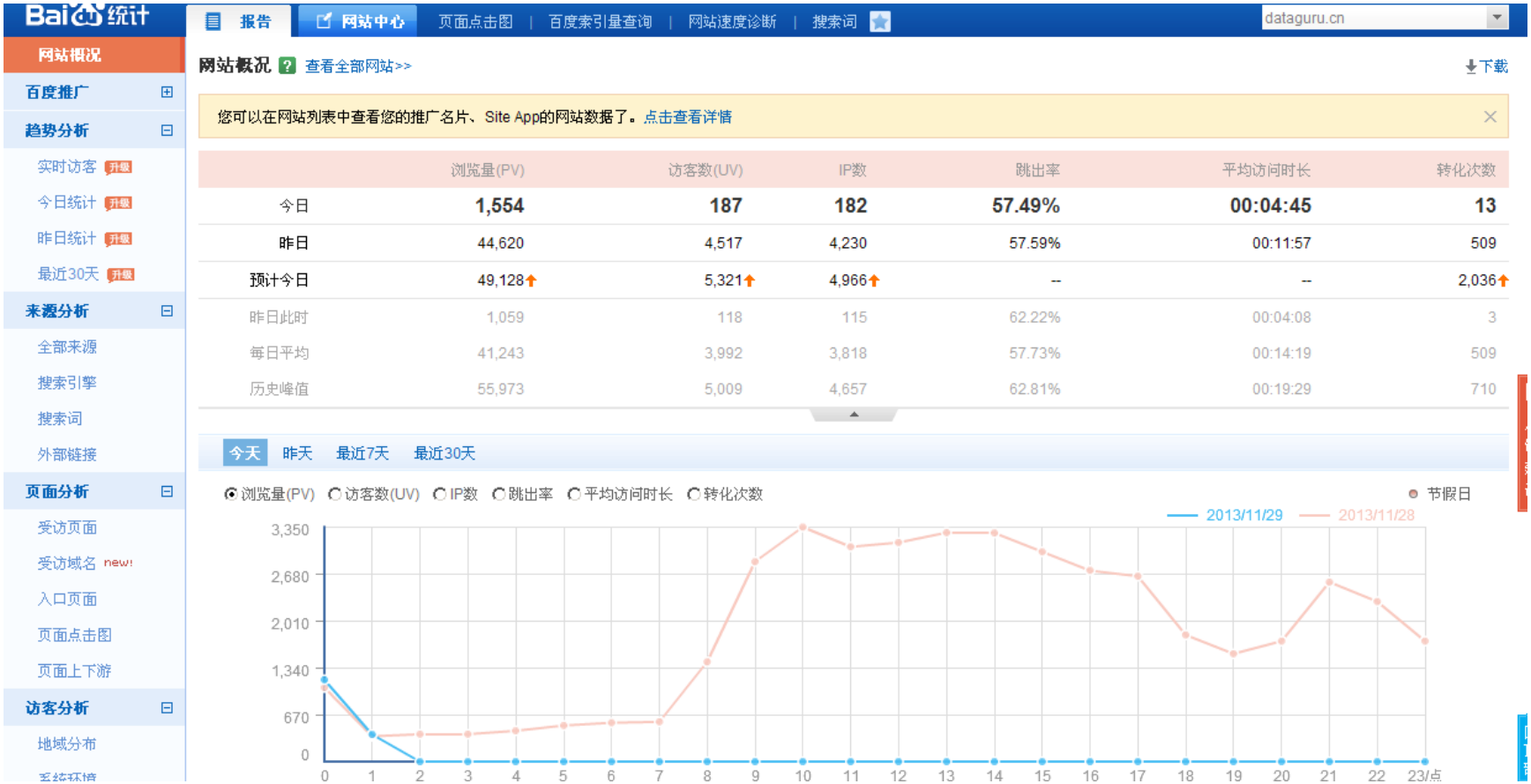


```
t="sso_username_utf8";
break}})();
A(e,"mouseover",F,false);
A(e,"mousemove",F,false);
z=function(){var a,c;
if(e.body){a=e.body.clientWidth|e.documentElement.clientWidth;
c=e.body.clientHeight|e.documentElement.clientHeight}else{a=e.docu
```

流行网站分析工具

- Webtrends
- Google分析
- 百度统计

百度统计：常见分析指标


Bai统计
报告
网站中心
页面点击图
百度索引量查询
网站速度诊断
搜索词
dataguru.cn
网站概况
百度推广
趋势分析
实时访客
今日统计
昨日统计
最近30天
来源分析
全部来源
搜索引擎
搜索词
外部链接
页面分析
受访页面
受访域名
入口页面
页面点击图
页面上下游
访客分析
地域分布
网站概况 查看全部网站>>
下载
您可以在网站列表中查看您的推广名片、Site App的网站数据了。点击查看详情
浏览量(PV) 访客数(UV) IP数 跳出率 平均访问时长 转化次数
今日 1,554 187 182 57.49% 00:04:45 13
昨日 44,620 4,517 4,230 57.59% 00:11:57 509
预计今日 49,128 5,321 4,966 -- -- 2,036
昨日此时 1,059 118 115 62.22% 00:04:08 3
每日平均 41,243 3,992 3,818 57.73% 00:14:19 509
历史峰值 55,973 5,009 4,657 62.81% 00:19:29 710
今天 昨天 最近7天 最近30天
浏览量(PV) 访客数(UV) IP数 跳出率 平均访问时长 转化次数
节假日
2013/11/29
2013/11/28
3,350
2,680
2,010
1,340
670
0
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23/点

项目背景:少量数据的情况

- 少量数据的情况(10Mb,100Mb,10G)，在单机处理尚能忍受的时候，我可以直接利用各种Unix/Linux工具，awk、grep、sort、join等都是日志分析的利器，再配合perl, python，正则表达式，基本就可以解决所有的问题。

- 例如，从nginx日志中得到访问量最高前5个IP，实现很简单：

```
conan@master:~/datafiles$ cat access.log.10 | awk '{a[$1]++} END {for(b in a) print b"\t"a[b]}' | sort -k2
-r | head -n 10
163.177.71.12   972
101.226.68.137  972
183.195.232.138 971
50.116.27.194   97
14.17.29.86     96
61.135.216.104  94
61.135.216.105  91
61.186.190.41   9
59.39.192.108   9
220.181.51.212  9
```

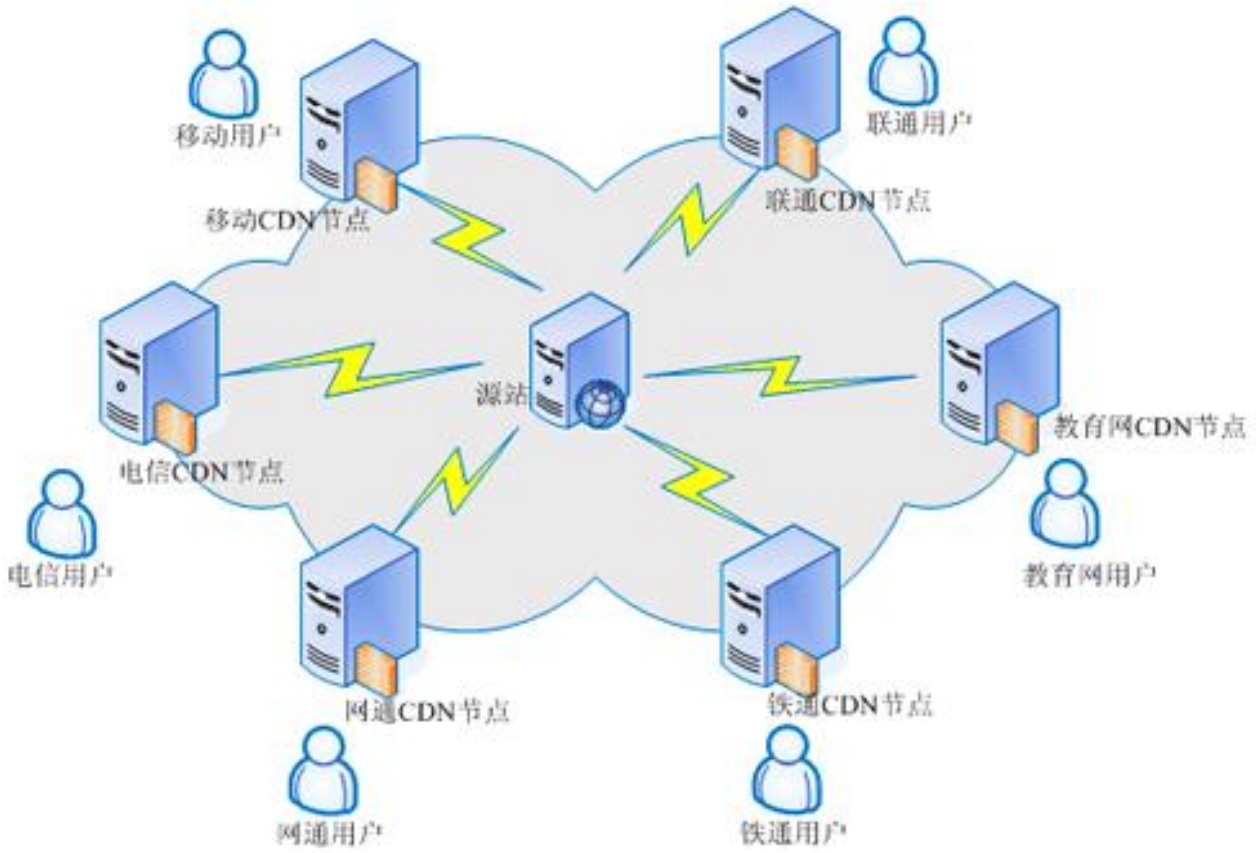
- ~ cat access.log.10 | awk '{a[$1]++} END {for(b in a) print b"\t"a[b]}' | sort -k2 -r | head -n 5

项目背景: 海量数据的情况

- 当数据量每天以10G、100G增长的时候，单机处理能力已经不能满足需求。我们就需要增加系统的复杂性，用计算机集群，存储阵列来解决。在Hadoop出现之前，海量数据存储，和海量日志分析都是非常困难的。只有少数一些公司，掌握着高效的并行计算，分步式计算，分步式存储的核心技术。

- Hadoop的出现，大幅度的降低了海量数据处理的门槛，让小公司甚至是个人都能力，搞定海量数据。并且，Hadoop非常适用于日志分析系统。

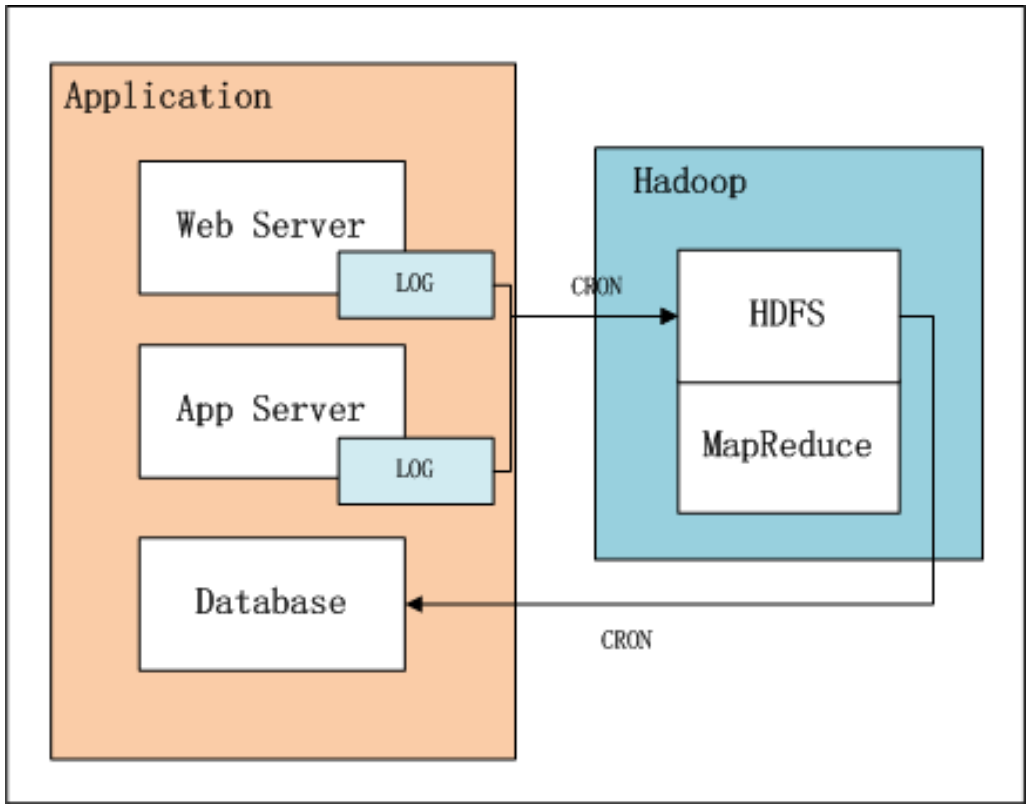
CDN


移动用户
移动CDN节点
联通用户
联通CDN节点
电信CDN节点
源站
教育网CDN节点
电信用户
教育网用户
网通CDN节点
铁通CDN节点
网通用户
铁通用户

分布式日志的收集

- 脚本方案
- Flume
- Chukwa

架构设计：应用系统及日志系统架构

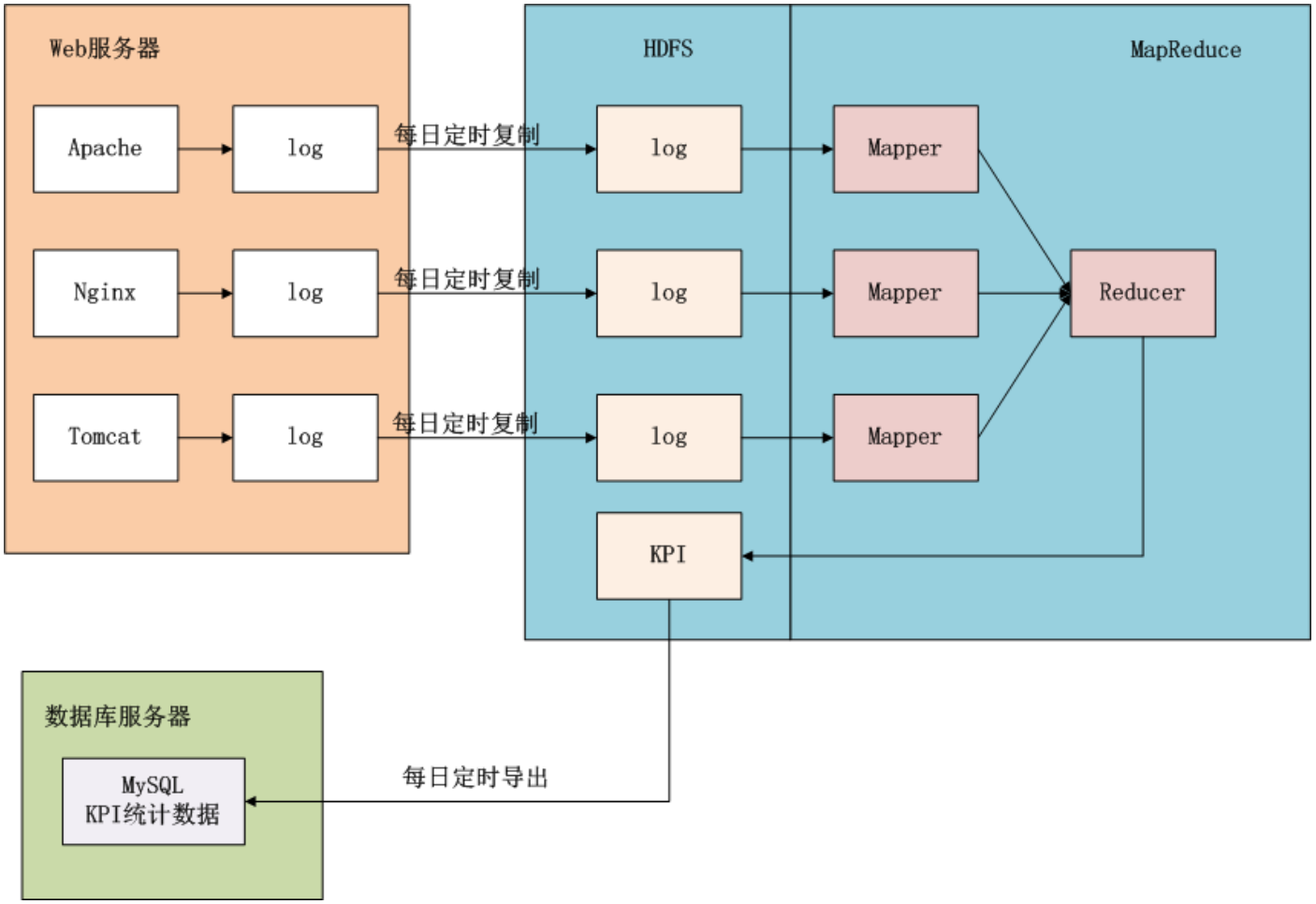


- 左边: Application业务系统
- 右边: Hadoop的HDFS, MapReduce。

架构设计：应用系统及日志系统架构

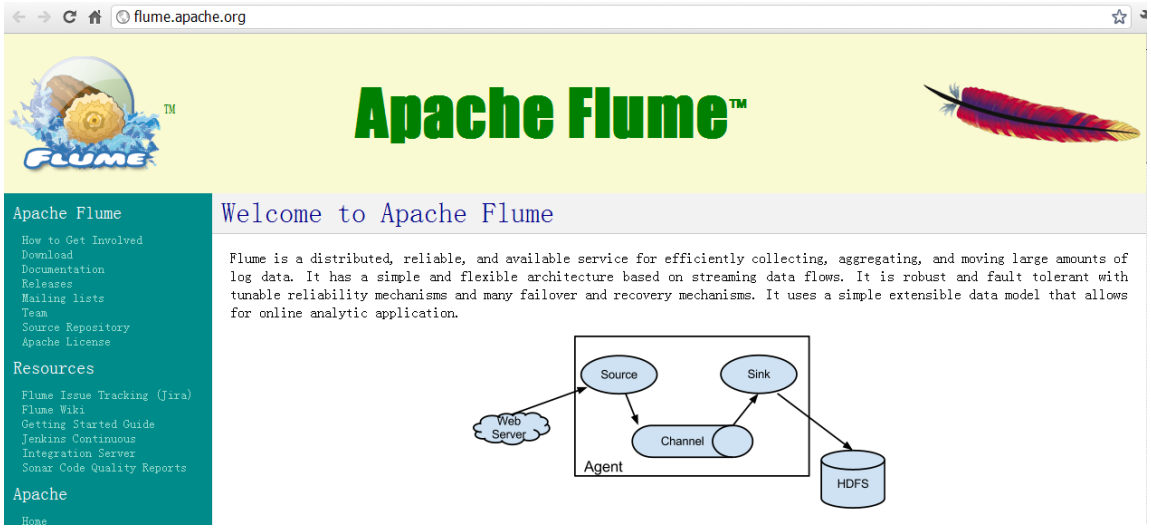
- 日志是由业务系统产生的，我们可以设置web服务器每天产生一个新的目录，目录下面会产生多个日志文件，每个日志文件64M。
- 设置系统定时器CRON，夜间在0点后，向HDFS导入昨天的日志文件。
- 完成导入后，设置系统定时器，启动MapReduce程序，提取并计算统计指标。
- 完成计算后，设置系统定时器，从HDFS导出统计指标数据到数据库，方便以后的即使查询。

架构设计：数据流

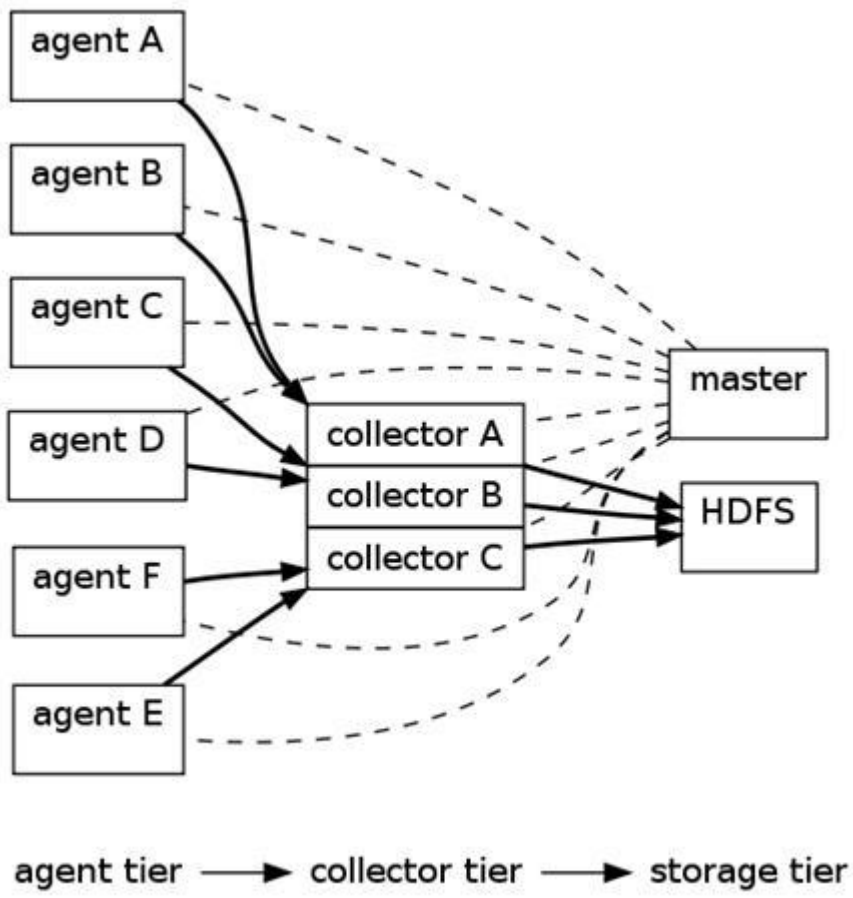

Web服务器
Apache
log
Nginx
log
Tomcat
log
每日定时复制
每日定时复制
每日定时复制
HDFS
log
log
log
KPI
MapReduce
Mapper
Mapper
Mapper
Reducer
数据库服务器
MySQL
KPI统计数据
每日定时导出

Flume

- Cloudera提供的分布式、可靠、和高可用的海量日志采集、聚合和传输的系统
- Flume提供了从console（控制台）、RPC（Thrift-RPC）、text（文件）、tail（UNIX tail）、syslog（syslog日志系统，支持TCP和UDP等2种模式），exec（命令执行）等数据源上收集数据的能力。同时，Flume的数据接受方，可以是console（控制台）、text（文件）、dfs（HDFS文件）、RPC（Thrift-RPC）和syslogTCP（TCP syslog日志系统）等。



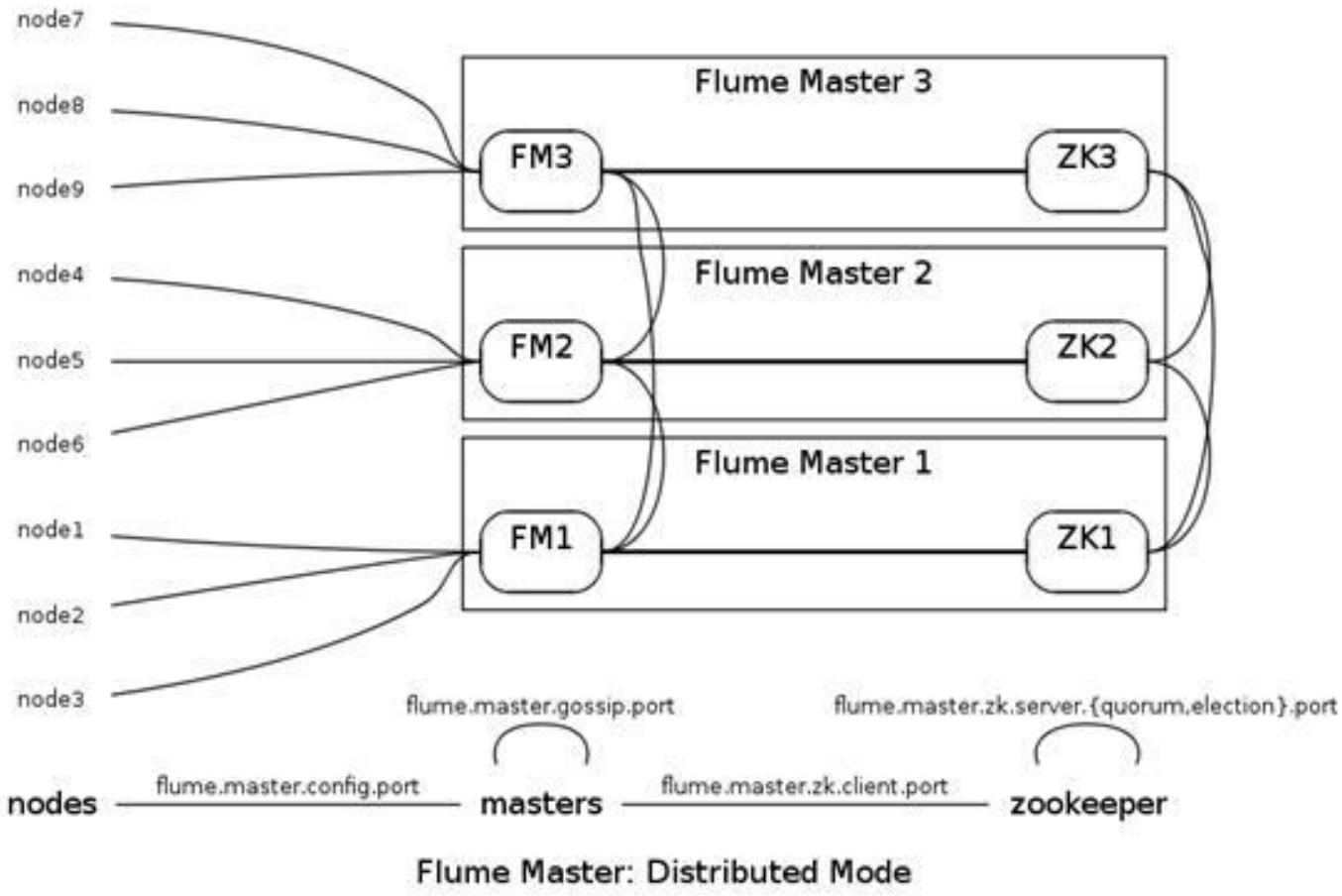
体系架构



- data flow描述了数据从产生，传输、处理并最终写入目标的一条路径（图中的实线）。
- Agent用于采集数据，是flume中产生数据流的地方，将产生的数据流传输到collector。
- collector用于对数据进行聚合，往往会产生一个更大的流。
- 收集数据有2种主要工作模式，如下：
  Push Sources：外部系统会主动地将数据推送到Flume中。
  Polling Sources：Flume到外部系统中获取数据。

Flume Master

- 用于管理数据流的配置



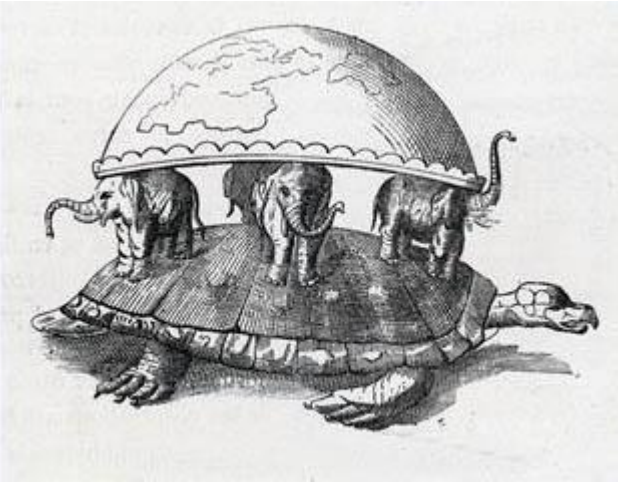
Flume Master: Distributed Mode

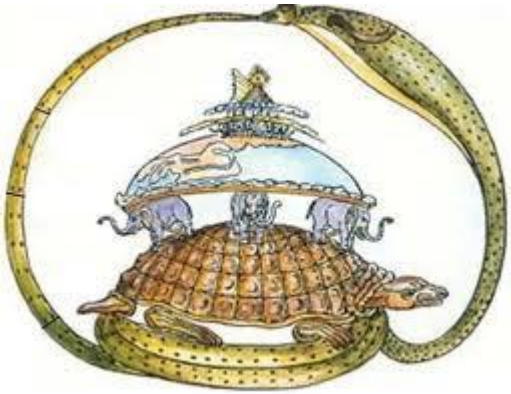
部署

- http://blog.csdn.net/zhouleilei/article/details/8568147

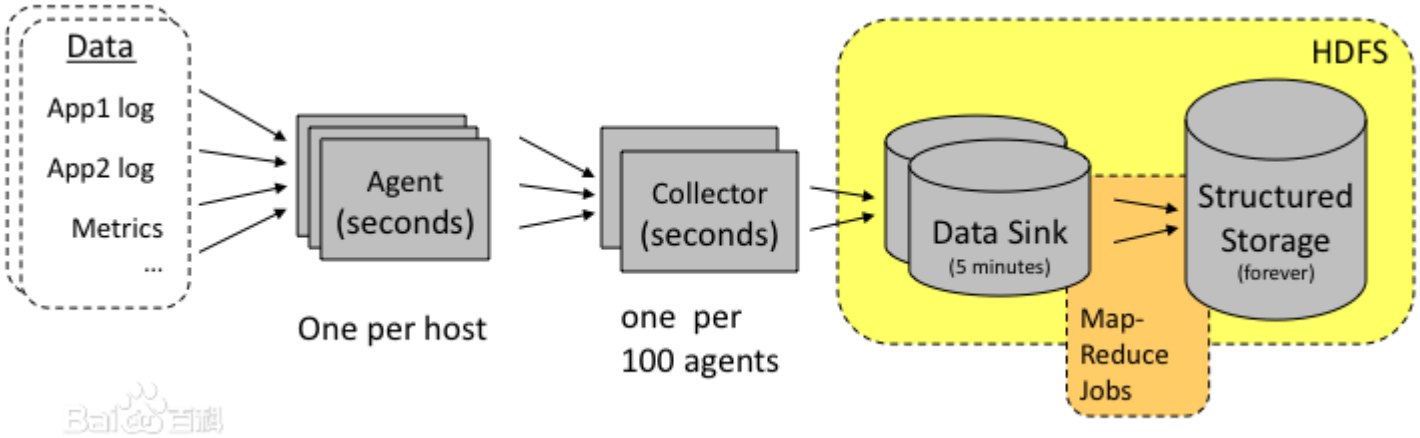
Chukwa

- 在印度神话中Chukwa是一只最古老的龟。它支撑着世界。在它的背上还支撑着一种叫做Maha-Pudma的大象，在大象的背上顶着这个地球。呵呵，大 象？Hadoop？不难理解为什么在Hadoop中的这个子项目叫做Chukwa了，或许Chukwa的其中一位开发者是印度人？呵呵，我瞎猜的，神话中 的Chukwa的，貌似是这样





架构图


Data
App1 log
App2 log
Metrics
...
Agent
(seconds)
One per host
Collector
(seconds)
one per
100 agents
HDFS
Data Sink
(5 minutes)
Structured
Storage
(forever)
Map-
Reduce
Jobs

部署Chukwa

- 陆嘉恒书第415页

需求分析：案例介绍

- 某电子商务网站，在线团购业务。每日PV数100w，独立IP数5w。用户通常在工作日上午10:00-12:00和下午15:00-18:00访问量最大。日间主要是通过PC端浏览器访问，休息日及夜间通过移动设备访问较多。网站搜索浏量占整个网站的80%，PC用户不足1%的用户会消费，移动用户有5%会消费。

- 通过简短的描述，我们可以粗略地看出，这家电商网站的经营状况，并认识到愿意消费的用户从哪里来，有哪些潜在的用户可以挖掘，网站是否存在倒闭风险等。

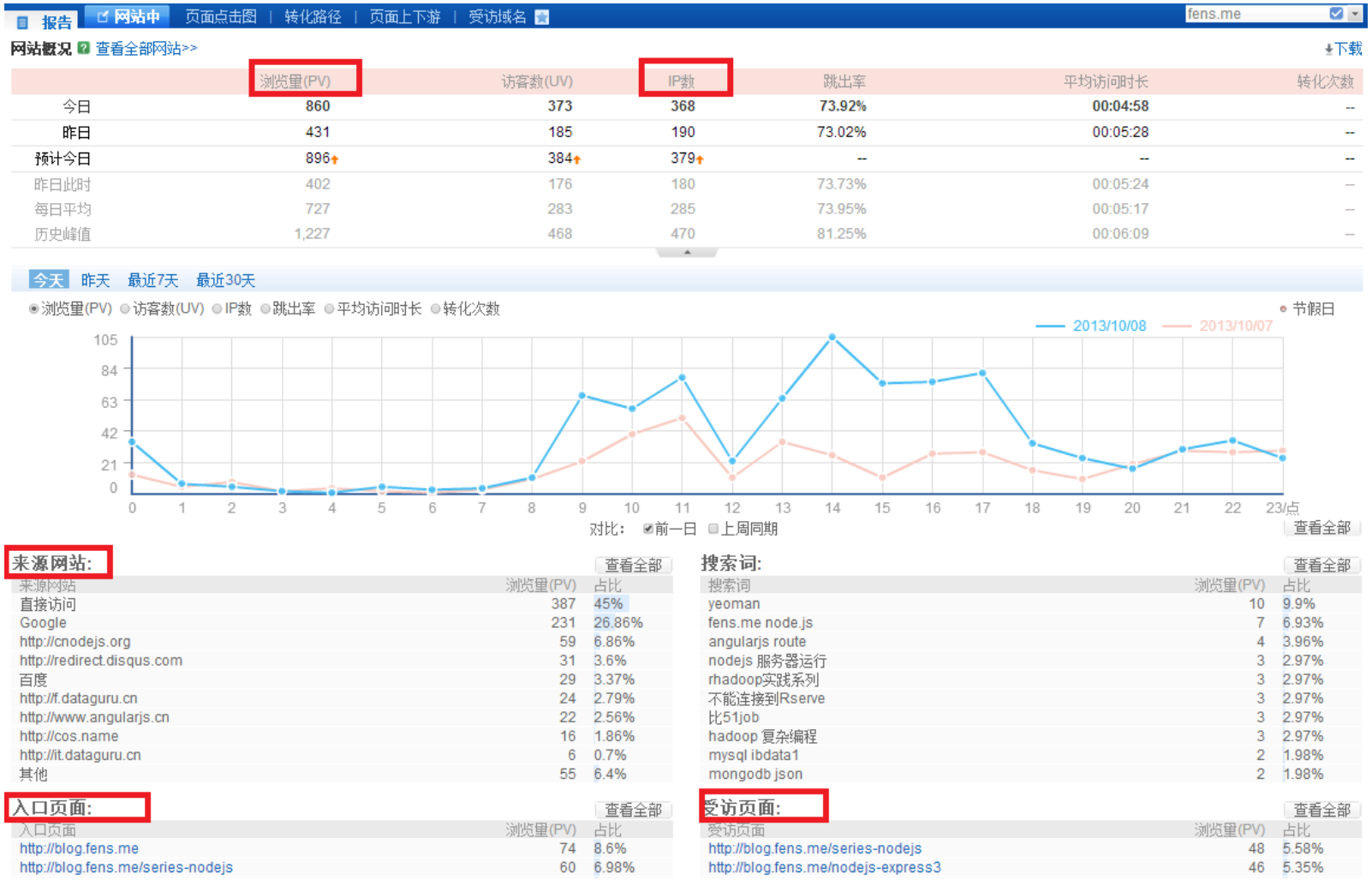
需求分析： KPI指标设计

- PV(PageView): 页面访问量统计
- IP: 页面独立IP的访问量统计
- Time: 用户每小时PV的统计
- Source: 用户来源域名的统计
- Browser: 用户的访问设备统计

- **注：商业保密限制，无法提供电商网站的日志。**

  下面的内容，将以我的个人网站为例提取数据进行分析, http://www.fens.me

需求分析: 基本统计指标


报告
网站中
页面点击图
转化路径
页面上下游
受访域名
fens.me
网站概况
查看全部网站>>
下载
浏览量(PV)
访客数(UV)
IP数
跳出率
平均访问时长
转化次数
今日 860 373 368 73.92% 00:04:58 --
昨日 431 185 190 73.02% 00:05:28 --
预计今日 896 384 379 -- -- --
昨日此时 402 176 180 73.73% 00:05:24 --
每日平均 727 283 285 73.95% 00:05:17 --
历史峰值 1,227 468 470 81.25% 00:06:09 --
今天 昨天 最近7天 最近30天
浏览量(PV) 访客数(UV) IP数 跳出率 平均访问时长 转化次数
节假日
2013/10/08
2013/10/07
105
84
63
42
21
0
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23/点
对比: 前一日 上周同期
查看全部
来源网站:
查看全部
来源网站 浏览量(PV) 占比
直接访问 387 45%
Google 231 26.86%
http://cnodejs.org 59 6.86%
http://redirect.disqus.com 31 3.6%
百度 29 3.37%
http://f.dataguru.cn 24 2.79%
http://www.angularjs.cn 22 2.56%
http://cos.name 16 1.86%
http://it.dataguru.cn 6 0.7%
其他 55 6.4%
搜索词:
查看全部
搜索词 浏览量(PV) 占比
yeoman 10 9.9%
fens.me node.js 7 6.93%
angularjs route 4 3.96%
nodejs 服务器运行 3 2.97%
rhadoop实践系列 3 2.97%
不能连接到Rserve 3 2.97%
比51job 3 2.97%
hadoop 复杂编程 3 2.97%
mysql ibdata1 2 1.98%
mongodb json 2 1.98%
入口页面:
查看全部
入口页面 浏览量(PV) 占比
http://blog.fens.me 74 8.6%
http://blog.fens.me/series-nodejs 60 6.98%
受访页面:
查看全部
受访页面 浏览量(PV) 占比
http://blog.fens.me/series-nodejs 48 5.58%
http://blog.fens.me/nodejs-express3 46 5.35%

需求分析: 访问设备指标

★ 自定义指标

转化目标 全部页面目标

	浏览器	浏览里(PV)	访客数(UV)	IP数	跳出率	平均访问时长
⊟ 1	计算机端浏览器	814	362	363	74.11%	00:04:58
	Google Chrome	545	246	248	74.85%	00:05:21
	Firefox	85	51	52	81.25%	00:02:32
	IE 9.0	53	7	7	36.36%	00:14:40
	IE 8.0	36	14	12	71.43%	00:02:57
	IE 10.0	29	11	11	64.29%	00:07:19
	搜狗高速	20	10	9	50%	00:04:03
	Safari	16	9	10	84.62%	00:03:00
	百度浏览器	15	2	2	33.33%	00:01:04
	猎豹浏览器	7	5	5	83.33%	00:02:43
	IE 6.0	3	2	2	50%	00:08:55
	枫树浏览器	1	1	1	100%	00:03:19
	世界之窗	1	1	1	100%	00:02:00
	Opera	1	1	1	100%	00:02:00
	其他	1	1	1	100%	00:05:27
	QQ浏览器	1	1	1	100%	00:02:00
⊟ 2	移动端浏览器	49	9	11	71.43%	00:08:05
	Safari移动版	23	2	2	50%	00:01:32
	Android Webkit Browser	18	2	2	33.33%	00:31:07
	Chrome移动版	6	4	5	100%	00:01:55
	IE移动版	2	1	2	100%	00:01:59
	当页汇总	**863**	**371**	**374**	**74.04%**	**00:05:04**

需求分析: 电商网站VS个人网站

- 从商业的角度，个人网站的特征与电商网站不太一样，没有转化率，同时跳出率也比较高。
- 从技术的角度，同样都关注KPI指标设计。
  - PV, IP, 转化率, 跳出率, 在线时长, 来源网站，来源域名，外部链接
  - 搜索流量，搜索关键词
  - 入口页面，跳出页面，受访页面
  - 访客男女，访客年龄，访客位置
  - 使用设置，操作系统，浏览器，爬虫，RSS阅读器
  - ....

拆解为8个变量

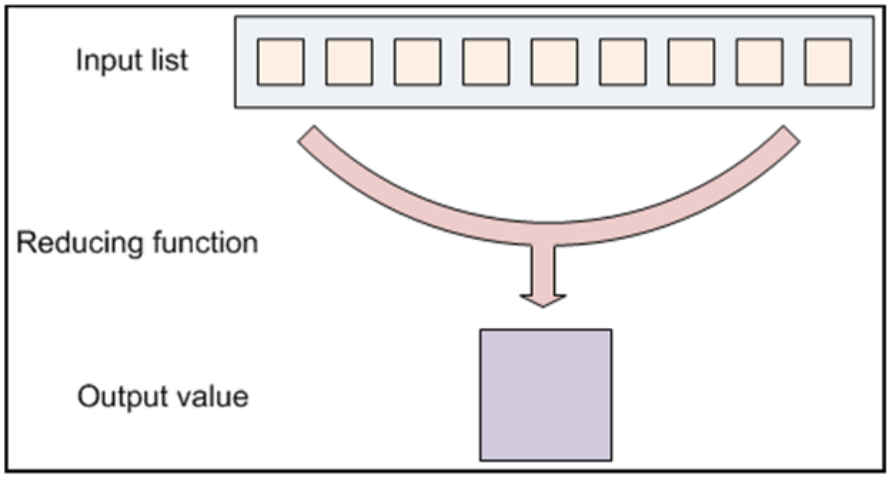
- remote_addr: 记录客户端的ip地址, 222.68.172.190
- remote_user: 记录客户端用户名称, –
- time_local: 记录访问时间与时区, [18/Sep/2013:06:49:57 +0000]
- request: 记录请求的url与http协议, “GET /images/my.jpg HTTP/1.1”
- status: 记录请求状态,成功是200, 200
- body_bytes_sent: 记录发送给客户端文件主体内容大小, 19939
- http_referer: 用来记录从那个页面链接访问过来的, “http://www.angularjs.cn/A00n”
- http_user_agent: 记录客户浏览器的相关信息, “Mozilla/5.0 (Windows NT 6.1) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/29.0.1547.66 Safari/537.36”

算法模型：变量解析

- remote_addr
- remote_user
- time_local
- Request
- Status
- body_bytes_sent
- http_referer
- http_user_agent

124.42.13.230 - - [18/Sep/2013:06:57:51 +0000] "GET /wp-content/themes/silesia/js/jquery.cycle.all.min.js HTTP/1.1" 200 31539 "http://blog.fens.me/mongodb-replica-set/" "Mozilla/4.0 (compatible; MSIE 8.0; Windows NT 5.1; Trident/4.0; BTRS101170; InfoPath.2; .NET4.0C; .NET4.0E; .NET CLR 2.0.50727)"

222.68.172.190 - - [18/Sep/2013:06:49:57 +0000] "GET /images/my.jpg HTTP/1.1" 200 19939 "http://www.angularjs.cn/A00n" "Mozilla/5.0 (Windows NT 6.1) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/29.0.1547.66 Safari/537.36"



算法模型: 并行算法 1

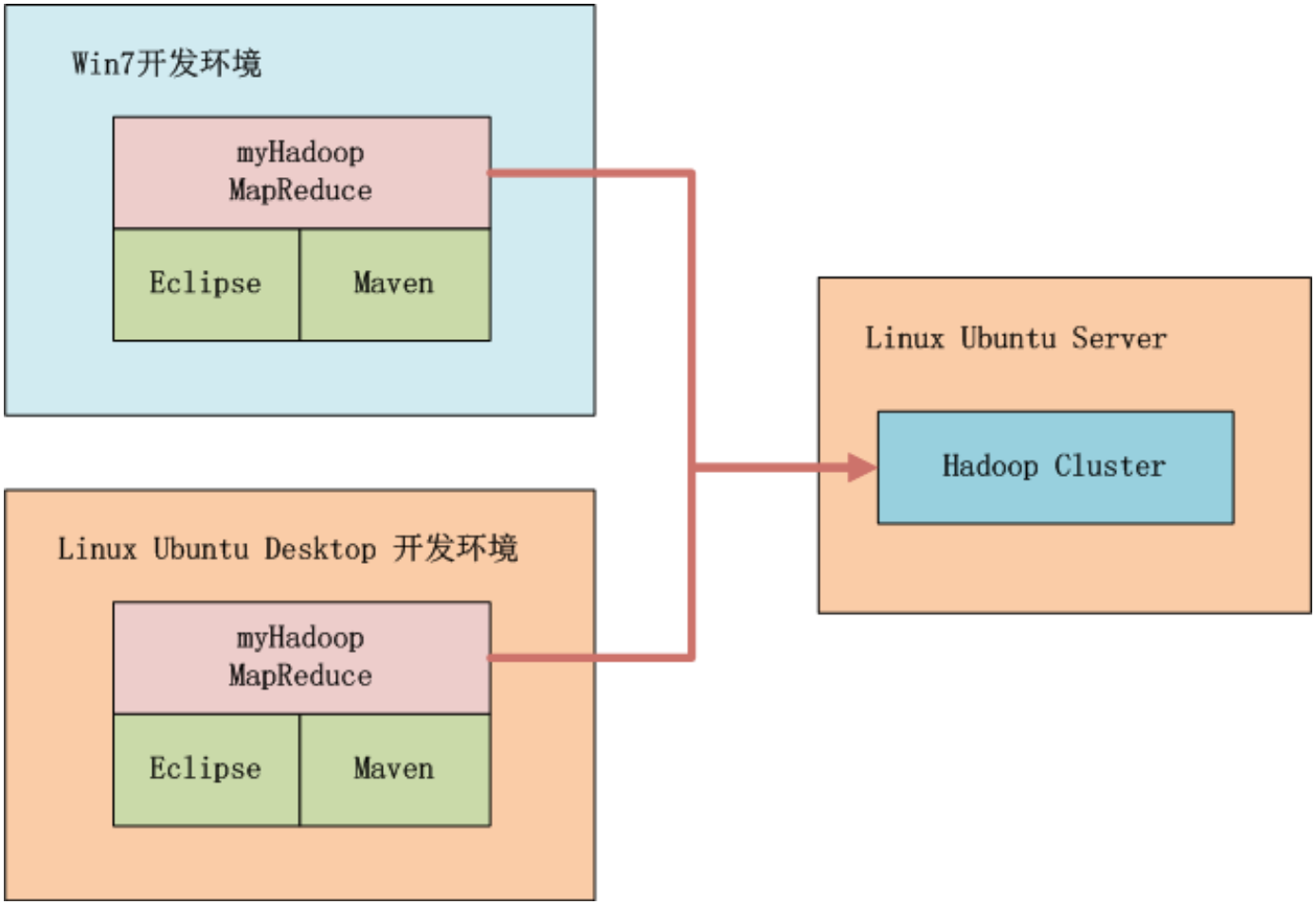
- **PV(PageView): 页面访问量统计**
  - Map: {key:$request,value:1}
  - Reduce: {key:$request,value:求和(sum)}
- **IP: 页面独立IP的访问量统计**
  - Map: {key:$request,value:$remote_addr}
  - Reduce: {key:$request,value:去重再求和(sum(unique))}
- **Time: 用户每小时PV的统计**
  - Map: {key:$time_local,value:1}
  - Reduce: {key:$time_local,value:求和(sum)}

算法模型: 并行算法 2

- **Source: 用户来源域名的统计**
  - Map: {key:$http_referer,value:1}
  - Reduce: {key:$http_referer,value:求和(sum)}

- **Browser: 用户的访问设备统计**
  - Map: {key:$http_user_agent,value:1}
  - Reduce: {key:$http_user_agent,value:求和(sum)}

程序开发: 用Maven构建Hadoop项目



程序开发: 用Maven构建Hadoop项目

- 开发环境
  - Win7 64bit
  - Java 1.6.0_45
  - Maven3
  - Eclipse Juno Service Release 2

- Hadoop集群系统环境：
  - Linux: Ubuntu 12.04.2 LTS 64bit Server
  - Java: 1.6.0_29
  - Hadoop: hadoop-1.0.3，单节点，IP:192.168.1.210

- 请参考文章：用Maven构建Hadoop项目

程序开发: 向HDFS导入日志文件

- 我们需要把日志文件，上传的HDFS里/user/hdfs/log_kpi/目录

- 参考下面的命令操作

~ hadoop fs -mkdir /user/hdfs/log_kpi

~ hadoop fs -copyFromLocal /home/conan/datafiles/access.log.10 /user/hdfs/log_kpi/

程序开发: MapReduce开发流程

- 对”日志行”的解析
- Map函数实现
- Reduce函数实现
- 启动程序实现

程序源代码下载：

https://github.com/bsspirit/maven_hadoop_template/releases/tag/kpi_v1

程序开发: 对日志行的解析 1

- 新建文件：org.conan.myhadoop.mr.kpi.KPI.java

```
    public static void main(String args[]) {
        String line = "222.68.172.190 - - [18/Sep/2013:06:49:57 +0000] \"GET /images/my.jpg
        System.out.println(line);
        KPI kpi = new KPI();
        String[] arr = line.split(" ");

        kpi.setRemote_addr(arr[0]);
        kpi.setRemote_user(arr[1]);
        kpi.setTime_local(arr[3].substring(1));
        kpi.setRequest(arr[6]);
        kpi.setStatus(arr[8]);
        kpi.setBody_bytes_sent(arr[9]);
        kpi.setHttp_referer(arr[10]);
        kpi.setHttp_user_agent(arr[11] + " " + arr[12]);
        System.out.println(kpi);

        try {
            SimpleDateFormat df = new SimpleDateFormat("yyyy.MM.dd:HH:mm:ss", Locale.US);
            System.out.println(df.format(kpi.getTime_local_Date()));
            System.out.println(kpi.getTime_local_Date_hour());
            System.out.println(kpi.getHttp_referer_domain());
        } catch (ParseException e) {
            e.printStackTrace();
        }
    }

}
```

程序开发: 对日志行的解析 2

- 控制台输出

```
222.68.172.190 - - [18/Sep/2013:06:49:57 +0000] "GET /images/my.jpg HTTP/1.1" 200 19939 "ht
valid:true
remote_addr:222.68.172.190
remote_user:-
time_local:18/Sep/2013:06:49:57
request:/images/my.jpg
status:200
body_bytes_sent:19939
http_referer:"http://www.angularjs.cn/A00n"
http_user_agent:"Mozilla/5.0 (Windows
2013.09.18:06:49:57
2013091806
www.angularjs.cn
```

- 我们看到日志行，被正确的解析成了kpi对象的属性

程序开发: map，reduce

- 对map方法，reduce方法，启动方法，我们单独写一个类来实现

- 下面将分别介绍MapReduce的实现类：
  - PV: org.conan.myhadoop.mr.kpi.KPIPV.java
  - IP: org.conan.myhadoop.mr.kpi.KPIIP.java
  - Time: org.conan.myhadoop.mr.kpi.KPITime.java
  - Browser: org.conan.myhadoop.mr.kpi.KPIBrowser.java

程序开发: map方法

```
public static class KPIPVMapper extends MapReduceBase implements Mapper {
    private IntWritable one = new IntWritable(1);
    private Text word = new Text();

    @Override
    public void map(Object key, Text value, OutputCollector output, Reporter reporter) {
        KPI kpi = KPI.filterPVs(value.toString());
        if (kpi.isValid()) {
            word.set(kpi.getRequest());
            output.collect(word, one);
        }
    }
}
```

程序开发: reduce方法

```
public static class KPIPVReducer extends MapReduceBase implements Reducer {
    private IntWritable result = new IntWritable();

    @Override
    public void reduce(Text key, Iterator values, OutputCollector output, Reporter repor
        int sum = 0;
        while (values.hasNext()) {
            sum += values.next().get();
        }
        result.set(sum);
        output.collect(key, result);
    }
}
```

程序开发: 启动方法

```
public static void main(String[] args) throws Exception {
    String input = "hdfs://192.168.1.210:9000/user/hdfs/log_kpi/";
    String output = "hdfs://192.168.1.210:9000/user/hdfs/log_kpi/pv";

    JobConf conf = new JobConf(KPIPV.class);
    conf.setJobName("KPIPV");
    conf.addResource("classpath:/hadoop/core-site.xml");
    conf.addResource("classpath:/hadoop/hdfs-site.xml");
    conf.addResource("classpath:/hadoop/mapred-site.xml");

    conf.setMapOutputKeyClass(Text.class);
    conf.setMapOutputValueClass(IntWritable.class);

    conf.setOutputKeyClass(Text.class);
    conf.setOutputValueClass(IntWritable.class);

    conf.setMapperClass(KPIPVMapper.class);
    conf.setCombinerClass(KPIPVReducer.class);
    conf.setReducerClass(KPIPVReducer.class);

    conf.setInputFormat(TextInputFormat.class);
    conf.setOutputFormat(TextOutputFormat.class);

    FileInputFormat.setInputPaths(conf, new Path(input));
    FileOutputFormat.setOutputPath(conf, new Path(output));

    JobClient.runJob(conf);
    System.exit(0);
}
```

程序开发: 对指定页面的过滤

- 在map方法中，程序会调用KPI类的方法

  KPI.filterPVs(value.toString());

- 在filterPVs方法，我们定义了一个pages的过滤，就是只对这个页面进行PV统计。

```
/**
 * 按page的pv分类
 */
public static KPI filterPVs(String line) {
    KPI kpi = parser(line);
    Set pages = new HashSet();
    pages.add("/about");
    pages.add("/black-ip-list/");
    pages.add("/cassandra-clustor/");
    pages.add("/finance-rhive-repurchase/");
    pages.add("/hadoop-family-roadmap/");
    pages.add("/hadoop-hive-intro/");
    pages.add("/hadoop-zookeeper-intro/");
    pages.add("/hadoop-mahout-roadmap/");

    if (!pages.contains(kpi.getRequest())) {
        kpi.setValid(false);
    }
    return kpi;
}
```

程序开发: 运行 KPIPV.java

- 用hadoop命令查看HDFS文件

```
~ hadoop fs -cat /user/hdfs/log_kpi/pv/part-00000

/about  5
/black-ip-list/ 2
/cassandra-clustor/     3
/finance-rhive-repurchase/      13
/hadoop-family-roadmap/ 13
/hadoop-hive-intro/     14
/hadoop-mahout-roadmap/ 20
/hadoop-zookeeper-intro/        6
```

程序开发: 独立IP统计 KPIIP.java

```
22  public class KPIIP {
23
24      public static class KPIIPMapper extends MapReduceBase implements Mapper<Object, Te
25          private Text word = new Text();
26          private Text ips = new Text();
27
28          @Override
29          public void map(Object key, Text value, OutputCollector<Text, Text> output, Rep
30              KPI kpi = KPI.filterIPs(value.toString());
31              if (kpi.isValid()) {
32                  word.set(kpi.getRequest());
33                  ips.set(kpi.getRemote_addr());
34                  output.collect(word, ips);
35              }
36          }
37      }
38
39      public static class KPIIPReducer extends MapReduceBase implements Reducer<Text, Te
40          private Text result = new Text();
41          private Set<String> count = new HashSet<String>();
42
43          @Override
44          public void reduce(Text key, Iterator<Text> values, OutputCollector<Text, Text
45              while (values.hasNext()) {
46                  count.add(values.next().toString());
47              }
48              result.set(String.valueOf(count.size()));
49              output.collect(key, result);
50          }
51      }
```

程序开发: 运行 KPIIP.java

- 用hadoop命令查看HDFS文件

```
conan@master:~$ hadoop fs -cat /user/hdfs/log_kpi/ip/part-00000
Warning: $HADOOP_HOME is deprecated.

/about  1
/black-ip-list/ 2
/cassandra-clustor/     3
/finance-rhive-repurchase/      4
/hadoop-family-roadmap/ 5
/hadoop-hive-intro/     6
/hadoop-mahout-roadmap/ 7
/hadoop-zookeeper-intro/        8
conan@master:~$
```

程序开发: 访问设备统计KPIBrowser.java

```
public class KPIBrowser {

    public static class KPIBrowserMapper extends MapReduceBase implements Mapper<Object
        private IntWritable one = new IntWritable(1);
        private Text word = new Text();

        @Override
        public void map(Object key, Text value, OutputCollector<Text, IntWritable> outp
            KPI kpi = KPI.filterBroswer(value.toString());
            if (kpi.isValid()) {
                word.set(kpi.getHttp_user_agent());
                output.collect(word, one);
            }
        }
    }

    public static class KPIBrowserReducer extends MapReduceBase implements Reducer<Text
        private IntWritable result = new IntWritable();

        @Override
        public void reduce(Text key, Iterator<IntWritable> values, OutputCollector<Text
            int sum = 0;
            while (values.hasNext()) {
                sum += values.next().get();
            }
            result.set(sum);
            output.collect(key, result);
        }
    }
```

程序开发: 运行 KPIBrowser.java

- 用hadoop命令查看HDFS文件

```
conan@master:~$ hadoop fs -cat /user/hdfs/log_kpi/browser/part-00000
Warning: $HADOOP_HOME is deprecated.

""      2
"-"     21
"360spider(http://webscan.360.cn)"      2
"AolReader/0.1.18 +http://reader.aol.com/"      30
"BOT/0.1 (BOT   5
"Baiduspider+(+http://www.baidu.com/search/spider.htm)" 2
"Baiduspider-image+(+http://www.baidu.com/search/spider.htm)"   5
"DNSPod-Monitor/1.0"    2915
"Digg Feed      3
"Disqus/1.0"    1
"DoCoMo/2.0 N905i(c100;TB;W24H16)       5
"Embedly +support@embed.ly"     1
"Feedly/1.0 (+http://www.feedly.com/fetcher.html;       11
"FreeWebMonitoring SiteChecker/0.2      15
"Googlebot-Image/1.0"   6
"HTTP_Request2/2.1.1 (http://pear.php.net/package/http_request2)        4
"Instapaper/5.0 CFNetwork/672.0.2       3
"InternetSeer.com"      4
"Jakarta Commons-HttpClient/3.1"        3
"Java/1.6.0_24" 3
"MQBrowser/5.0 (iPhone; 27
"Mozilla/4.0 (compatible;       1231
"Mozilla/4.0 (compatible;)"     34
"Mozilla/4.0"   321
```

程序开发: 按时间段统计KPITime.java

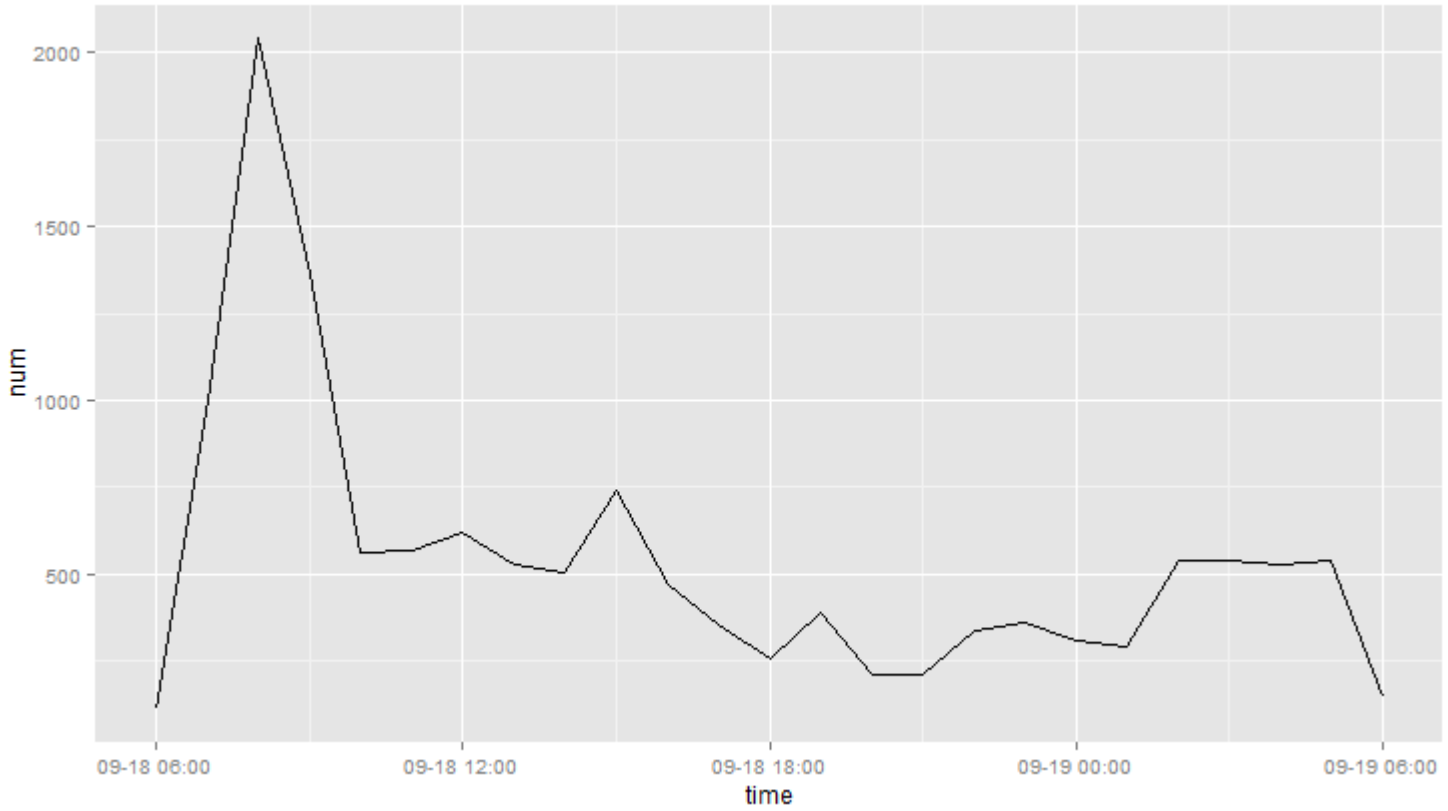
```
public class KPITime {

    public static class KPITimeMapper extends MapReduceBase implements Mapper<Object, T
        private IntWritable one = new IntWritable(1);
        private Text word = new Text();

        @Override
        public void map(Object key, Text value, OutputCollector<Text, IntWritable> outp
            KPI kpi = KPI.filterBroswer(value.toString());
            if (kpi.isValid()) {
                try {
                    word.set(kpi.getTime_local_Date_hour());
                    output.collect(word, one);
                } catch (ParseException e) {
                    e.printStackTrace();
                }
            }
        }
    }

    public static class KPITimeReducer extends MapReduceBase implements Reducer<Text, I
        private IntWritable result = new IntWritable();

        @Override
        public void reduce(Text key, Iterator<IntWritable> values, OutputCollector<Text
            int sum = 0;
            while (values.hasNext()) {
                sum += values.next().get();
            }
            result.set(sum);
            output.collect(key, result);
        }
    }
```

程序开发: 运行 KPITime.java

- 用hadoop命令查看HDFS文件

```
conan@master:~$ hadoop fs -cat /user/hdfs/log_kpi/time/part-00000
Warning: $HADOOP_HOME is deprecated.

2013091806      111
2013091807      1003
2013091808      2040
2013091809      1363
2013091810      564
2013091811      570
2013091812      618
2013091813      527
2013091814      503
2013091815      743
2013091816      471
2013091817      355
2013091818      259
2013091819      388
2013091820      210
2013091821      210
2013091822      338
2013091823      361
2013091900      307
```

程序开发: Time的时间序列图



程序开发: 下载Time的统计结果

- 可以参考下面的命令，从HDFS下载到Linux文件系统

~ hadoop fs -copyToLocal /user/hdfs/log_kpi/time /home/conan/datafiles

~ ls /home/conan/datafiles/time/part-00000

- R语言程序

```
1  library(lubridate)
2  library(scales)
3  library(ggplot2)
4
5  data<-read.table(file="part-00000",header=FALSE)
6  names(data)<-c("time","num")
7  data$time<-parse_date_time(data$time, "%y%m%d%H")
8  g<-ggplot(data,aes(x=time,y=num))
9  g<-g+geom_line();
10 g<-g+scale_x_datetime(labels = date_format("%m-%d %H:%M"))
11 g
12
13
```

关于张丹

- 骨灰级程序员, 大数据创业者
- DataguruID: bsspirit
- Weibo: @Conan_Z
- Blog：http://blog.fens.me
- Email: bsspirit@gmail.com

炼数成金逆向收费式网络课程

- Dataguru（炼数成金）是专业数据分析网站，提供教育，媒体，内容，社区，出版，数据分析业务等服务。我们的课程采用新兴的互联网教育形式，独创地发展了逆向收费式网络培训课程模式。既继承传统教育重学习氛围，重竞争压力的特点，同时又发挥互联网的威力打破时空限制，把天南地北志同道合的朋友组织在一起交流学习，使到原先孤立的学习个体组合成有组织的探索力量。并且把原先动辄成千上万的学习成本，直线下降至百元范围，造福大众。我们的目标是：低成本传播高价值知识，构架中国第一的网上知识流转阵地。
- 关于逆向收费式网络的详情，请看我们的培训网站 http://edu.dataguru.cn

Thanks

FAQ时间