**Pig**

**1 Pig的简介**

（1）Pig是一个基于Hadoop的大规模数据分析平台，它提供的SQL-LIKE语言叫Pig Latin，该语言的编译器会把类SQL的数据分析请求转换为一系列经过优化处理的MapReduce运算。Pig为复杂的海量数据并行计算提供了一个简单的操作和编程接口。Pig内部，每个操作或变换是对输入进行数据处理，然后产生输出结果，这些变换操作被转换成一系列MapReduce作业，Pig让程序员不需要知道这些转换具体是如何进行的，这样工程师可以将精力集中在数据上，而非执行的细节上。

（2）Compare：相比Java的MapReduce API，Pig为大型数据集的处理提供了更高层次的抽象，与MapReduce相比，Pig提供了更丰富的数据结构，一般都是多值和嵌套的数据结构。Pig还提供了一套更强大的数据变换操作，包括在MapReduce中被忽视的连接Join操作。

**（3）Pig包括两部分：**

@用于描述数据流的语言，称为Pig Latin。

@用于执行Pig Latin程序的执行环境，当前有两个环境：单JVM中的本地执行环境和Hadoop集群上的分布式执行环境。

**2 Pig的特点**

（1）专注于于大量数据集分析；

（2）运行在集群的计算架构上，Yahoo Pig提供了多层抽象，简化并行计算让普通用户使用；这些抽象完成自动把用户请求queries翻译成有效的并行评估计划，然后在物理集群上执行这些计划；

（3）提供类似 SQL 的操作语法；

（4）开放源代码；

**3 Pig与Hive的区别**

对于开发人员，直接使用Java APIs可能是乏味或容易出错的，同时也限制了Java程序员在Hadoop上编程的运用灵活性。于是Hadoop提供了两个解决方案，使得Hadoop编程变得更加容易。

　　•Pig是一种编程语言，它简化了Hadoop常见的工作任务。Pig可加载数据、表达转换数据以及存储最终结果。Pig内置的操作使得半结构化数据变得有意义（如日志文件）。同时Pig可扩展使用Java中添加的自定义数据类型并支持数据转换。

　　•Hive在Hadoop中扮演数据仓库的角色。Hive添加数据的结构在HDFS，并允许使用类似于SQL语法进行数据查询。与Pig一样，Hive的核心功能是可扩展的。

　　Pig和Hive总是令人困惑的。Hive更适合于数据仓库的任务，Hive主要用于静态的结构以及需要经常分析的工作。Hive与SQL相似促使 其成为Hadoop与其他BI工具结合的理想交集。Pig赋予开发人员在大数据集领域更多的灵活性，并允许开发简洁的脚本用于转换数据流以便嵌入到较大的应用程序。Pig相比Hive相对轻量，它主要的优势是相比于直接使用Hadoop Java APIs可大幅削减代码量。正因为如此，Pig仍然是吸引大量的软件开发人员。

**4 Pig和数据库的比较**

1）Pig是数据流编程语言，而SQL是一种描述型编程语言。Pig是相对于输入的一步步操作，其中每一步都是对数据的一个简单的变换；而SQL语句是一个约束的集合，这些约束结合在一起定义了输出。Pig更像RDBMS中的查询规划器。

2）RDBMS把数据存储在严格定义了模式的表内，但pig对数据的要求更宽松，可以在运行时定义模式，而且是可选的。

3）pig对复杂、嵌套数据结构的支持更强；

4）Pig不支持事务和索引，也不支持随机读和几十毫秒级别的查询，它是针对数据批量处理的。

5）Hive是介于Pig和RDBMS之间的系统。hive以HDFS为存储，但是查询语言是基于SQL的，而且Hive要求所有数据必须存储在表中，表必须有模式，而模式由Hive管理。但Hive允许为预先存在HDFS中。

**5 Pig的安装配置**

　　下载pig的压缩包，这里使用的是pig-0.11.1版本，已经上传至了百度网盘中（URL：http://pan.baidu.com/s/1o6IDfhK）

　　（1）通过FTP工具上传到虚拟机中，可以选择XFtp、CuteFTP等工具

　　（2）解压缩

tar -zvxf pig-0.11.1.tar.gz

　　（3）重命名

mv pig-0.11.1 pig

　　（4）修改/etc/profile，增加内容如下，最后重新生效配置文件source /etc/profile

export PIG\_HOME=/usr/local/pig

export PATH=.:$HADOOP\_HOME/bin:$PIG\_HOME/bin:$HBASE\_HOME/bin:$ZOOKEEPER\_HOME/bin:$JAVA\_HOME/bin:$PATH

（5）设置Pig与Hadoop关联

　　 进入$PIG\_HOME/conf中，编辑pig.properties文件，加入以下两行内容：

fs.default.name=hdfs://hadoop-master:9000

mapred.job.tracker=hadoop-master:9001

**6 Pig的使用实例**

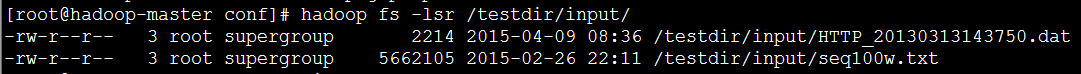
**（1）文件背景**

结合本笔记第五篇《自定义类型处理手机上网日志》的手机上网日志为背景，我们要做的就是通过Pig Latin对该日志进行流量的统计。该日志的数据结构定义如下图所示：（该文件的下载地址为：<http://pan.baidu.com/s/1dDzqHWX>）



　　PS：在使用Pig之前先将该文件上传至HDFS中，这里上传到了/testdir/input目录中

hadoop fs -put HTTP\_20130313143750.dat /testdir/input



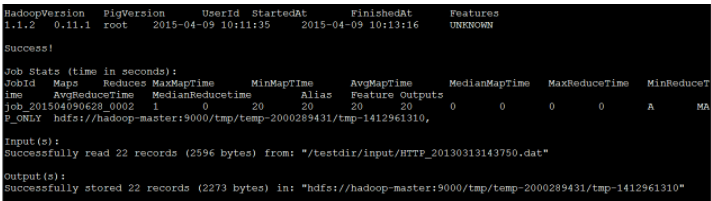
**（2） Load:把HDFS中的数据转换为Pig可以处理的模式**

1）首先通过**输入Pig进入grunt**，然后使用Load命令将原始文件转换为Pig可以处理的模式：

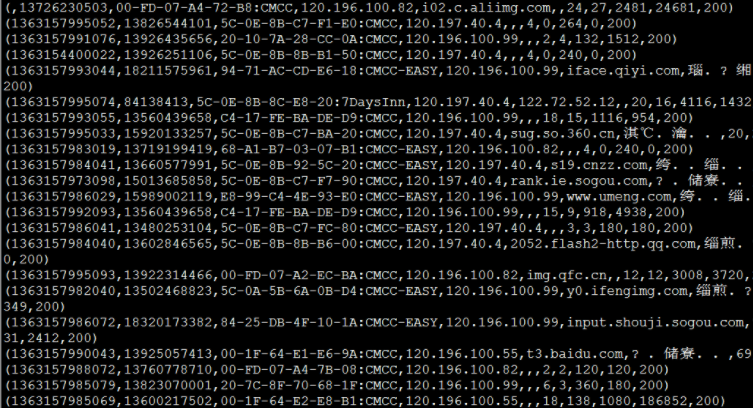
grunt>A = LOAD '/testdir/input/HTTP\_20130313143750.dat' AS (t0:long,

msisdn:chararray, t2:chararray, t3:chararray, t4:chararray, t5:chararray, t6:long, t7:long, t8:long, t9:long, t10:chararray);

2）通过Pig对指令的解析，帮我们转换成为了MapReduce任务：



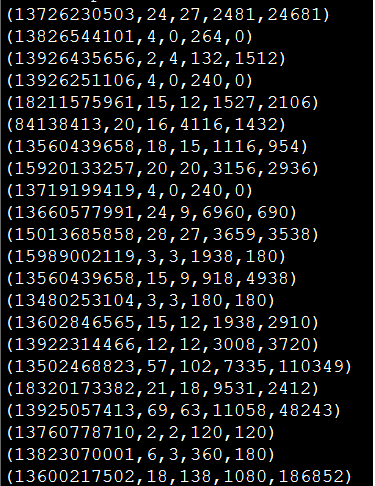
3）通过以下命令可以查看结果： grunt>DUMP A;



**（3）FOREACH:把A中有用的字段抽取出来**

1）这里我们需要统计的只是手机号以及四个流量数据，因此我们通过遍历将A中的部分字段抽取出来存入B中：grunt> B = FOREACH A GENERATE msisdn, t6, t7, t8, t9;

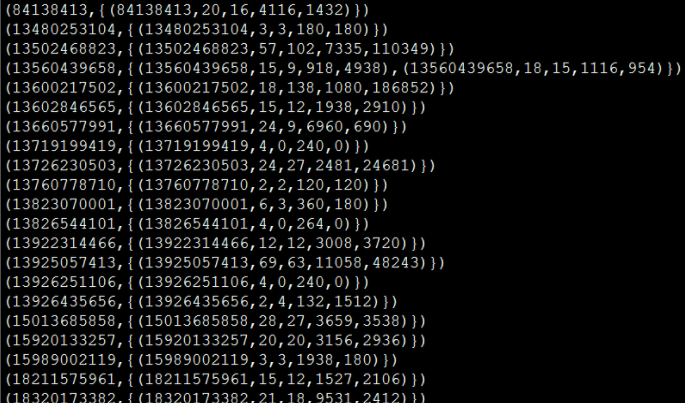
2）通过以下命令可以查看结果：grunt>DUMP B;



**（4） GROUP:分组数据**

1）有用信息抽取出来后，看到结果中一个手机号可能有多条记录，因此这里通过手机号进行分组：grunt> C = GROUP B BY msisdn;

2）通过以下命令可以查看结果： grunt>DUMP C;



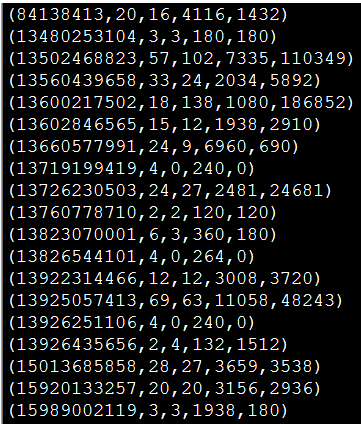
**（5）GENERATE:流量汇总**

1）在对手机号进行分组之后，我们可以看到某个手机号对应着多条流量记录数据，因此继续使用FOREACH遍历分组数据，然后对四个流量数据进行汇总，这里使用了聚合函数SUM()：

grunt> D = FOREACH C GENERATE group, SUM(B.t6), SUM(B.t7), SUM(B.t8), SUM(B.t9);

2）通过以下命令可以查看结果：

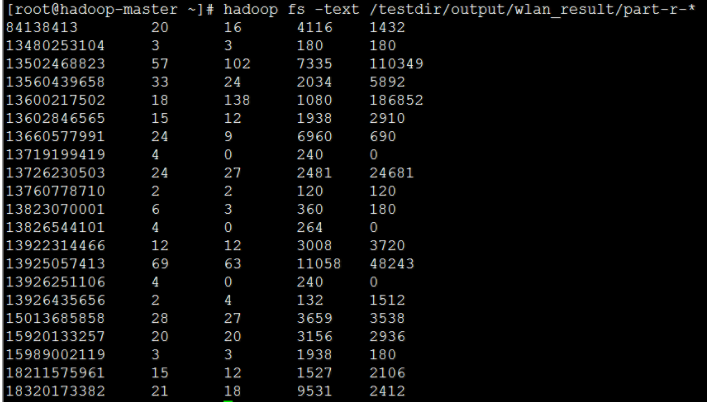
grunt>DUMP D;



**（6）STORE:将统计结果存储到HDFS中进行持久化**

1）在对流量统计完毕之后，结果仍然是在Pig中，这里就需要对其进行持久化操作，即将结果存储到HDFS中：grunt> STORE D INTO '/testdir/output/wlan\_result';

2）通过HDFS Shell查看存储结果：hadoop fs -text /testdir/output/wlan\_result/part-r-\*



**7 Pig简单示例**

我们以查找最高气温为例，演示如何利用Pig统计每年的最高气温。假设数据文件内容如下（每行一个记录，tab分割）：

1990 21

1990 18

1991 21

1992 30

1992 999

1990 23

以local模式进入pig，依次输入以下命令（注意以分号结束语句）：

records = load ‘/home/user/input/temperature1.txt’ as (year: chararray,temperature: int);

dump records;

describe records;

valid\_records = filter records by temperature!=999;

grouped\_records = group valid\_records by year;

dump grouped\_records;

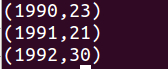
describe grouped\_records;

max\_temperature = foreach grouped\_records generate group,MAX(valid\_records.temperature);

--备注：valid\_records是字段名，在上一语句的describe命令结果中可以查看到group\_records 的具体结构。

dump max\_temperature;

最终结果为：



注意：

1）如果你运行Pig命令后报错，且错误消息中包含如下信息：

WARN org.apache.pig.backend.hadoop20.PigJobControl- falling back to default JobControl (not using hadoop 0.20 ?)

Java.lang.NoSuchFieldException:runnerState

则可能你的Pig版本和Hadoop版本不兼容。此时可重新针对特定Hadoop版本进行编辑。下载源代码后，进入源代码根目录，执行以下命令：

ant clean jar-withouthadoop-Dhadoopversion=23

注意：版本号是根据具体Hadoop而定，此处23可用于Hadoop2.2.0。

也可到以下网址下载：

http://download.csdn.net/detail/zythy/6843681

（因为文件太大，分成了3个压缩包）

2）Pig同一时间只能工作在一种模式下，比如以MapReduce模式进入后，只能读取HDFS文件，如果此时你用load 读取本地文件，将会报错。

**7 本质上说说Pig与Hive。**

经过Pig Latin的转换后变成了一道MapReduce的作业，通过MapReduce多个线程，进程或者独立系统并行执行处理的结果集进行分类和归纳。Map() 和 Reduce() 两个函数会并行运行，即使不是在同一的系统的同一时刻也在同时运行一套任务，当所有的处理都完成之后，结果将被排序，格式化，并且保存到一个文件。Pig利用MapReduce将计算分成两个阶段，第一个阶段分解成为小块并且分布到每一个存储数据的节点上进行执行，对计算的压力进行分散，第二个阶段聚合第一个阶段执行的这些结果，这样可以达到非常高的吞吐量，通过不多的代码和工作量就能够驱动上千台机器并行计算，充分的利用计算机的资源，打消运行中的瓶颈。

**也就是说，Pig最大的作用就是对mapreduce算法(框架)实现了一套shell脚本 ，类似我们通常熟悉的SQL语句，在Pig中称之为Pig Latin，在这套脚本中我们可以对加载出来的数据进行排序、过滤、求和、分组(group by)、关联(Joining)，**Pig也可以由用户自定义一些函数对数据集进行操作，也就是传说中的**UDF(user-defined functions)**。

**我们知道hive、pig都提供了一种脚本语法，便于我们快速构建mapreduce任务，不同的是hive的语法类似sql，而pig的语法类似于shell**

我感觉Hive和Pig本质上应该是没有区别的，都是建立在hadoop之上对数据进行处理和分析得到想要结果，区别：

1、hive 是一个类似sql语法的数据语言，pig是一个高级抽像的数据流语言，编程语言，简化hadoop常见的工作任务。

2、hive可以做为数据仓库，可以把数据按照hive的压缩格式放到指定目录从而对数据分析，pig只是抽像了数据存储的模型，到分析时按数据模析去分析。

3、hive中可以执行插入、删除等，但Pig我没有发现可以插入数据的方法。

4、hive有schemas和分区的概念，pig就只知道要加载数据的分隔格式

5、pig 无jdbc,hive有的。

6、hive给数据库管理员用，pig给运维人员用。