Pig实验

Pig Latin

Type	Operation	Illustrator
Load and Store	load	Load data from storage systems
	store	Store data to storage systems
	dump	Output data to the screen
Operation	foreach	Take a set of expressions to every line in the data
	flatten	Un-nest tuples as well as bags
Filtering	filter	Select which records will be retained in data
	distinct	Remove duplicate records
Group and join	join	Select records from one input to put together with records from another input
	group	Collect together records with the same key

Pig实例（1）- 源数据

- /data/log.txt

IP	Host	SP	Traffic
125.39.127.20	b8.photo.store.qq.com	qq	10
125.39.127.21	b5.photo.store.qq.com	qq	15
125.39.127.30	b25.photo.store.qq.com	qq	25
125.39.127.21	b40.photo.store.qq.com	qq	15
125.39.127.30	b32.photo.store.qq.com	qq	10
125.39.127.30	s6.photo.store.qq.com	qq	15
122.228.243.250	q.i02.wimg.taobao.com	taobao	100

- 分析目标：每个IP访问qq.com的不同host的流量占总流量百分比

Pig实例（2）- 加载（LOAD）

将数据加载到pig中

```
grunt> records = LOAD '/data/log.txt' AS (ip:chararray, host:chararray, sp:chararray, traffic:int);

grunt> describe records;
records:{ip: chararray,host: chararray,sp: chararray, traffic: int}

grunt> dump records;
```

```
(125.39.127.20,b8.photo.store.qq.com,qq,10)
(125.39.127.21,b5.photo.store.qq.com,qq,15)
(125.39.127.30,b25.photo.store.qq.com,qq,25)
(125.39.127.21,b40.photo.store.qq.com,qq,15)
(125.39.127.30,b32.photo.store.qq.com,qq,10)
(125.39.127.30,s6.photo.store.qq.com,qq,15)
(122.228.243.250,q.i02.wimg.taobao.com,taobao,100)
```

records

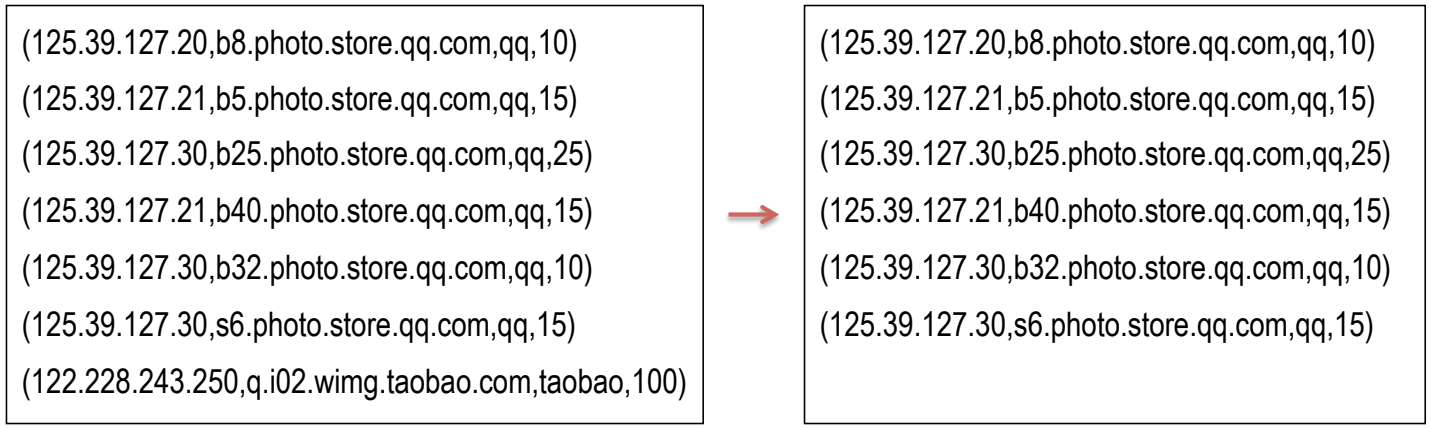
Pig实例（3）- 过滤（FILTER）

只选取host包含qq的记录

```
grunt> filter_records = FILTER records BY sp MATCHES '.*qq.*';

grunt> describe filter_records;
filter_records: {ip: chararray,host: chararray,sp: chararray, traffic: int}

grunt> dump filter_records;
```



records　　filter_records

Pig实例（4）- 排序（ORDER）

按IP排序

```
grunt> order_records = ORDER filter_records BY ip;

grunt> describe order_records;
order_records: {ip: chararray,host: chararray,sp: chararray, traffic: int}

grunt> dump order_records;
```

filter_records
(125.39.127.20,b8.photo.store.qq.com,qq,10)
(125.39.127.21,b5.photo.store.qq.com,qq,15)
(125.39.127.30,b25.photo.store.qq.com,qq,25)
(125.39.127.21,b40.photo.store.qq.com,qq,15)
(125.39.127.30,b32.photo.store.qq.com,qq,10)
(125.39.127.30,s6.photo.store.qq.com,qq,15)

order_records
(125.39.127.20,b8.photo.store.qq.com,qq,10)
(125.39.127.21,b5.photo.store.qq.com,qq,15)
(125.39.127.21,b40.photo.store.qq.com,qq,15)
(125.39.127.30,b25.photo.store.qq.com,qq,25)
(125.39.127.30,b32.photo.store.qq.com,qq,10)
(125.39.127.30,s6.photo.store.qq.com,qq,15)

Pig实例（5）- 分组（GROUP）

按IP分组

```
grunt> group_records = GROUP order_records BY ip;

grunt> describe group_records;
group_records: {group: chararray, order_records: {(ip: chararray, host: chararray,
sp: chararray, traffic: int)}}

grunt> dump group_records;
```

```
(125.39.127.20,b8.photo.store.qq.com,qq,10)
(125.39.127.21,b5.photo.store.qq.com,qq,15)
(125.39.127.21,b40.photo.store.qq.com,qq,15)
(125.39.127.30,b25.photo.store.qq.com,qq,25)
(125.39.127.30,b32.photo.store.qq.com,qq,10)
(125.39.127.30,s6.photo.store.qq.com,qq,15)
```

order_records

```
(125.39.127.20,{(125.39.127.20,b8.photo.store.qq.com,qq,10)})
(125.39.127.21,{(125.39.127.21,b5.photo.store.qq.com,qq,15),
   (125.39.127.21,b40.photo.store.qq.com,qq,15)})
(125.39.127.30,{(125.39.127.30,b25.photo.store.qq.com,qq,
   25),(125.39.127.30,b32.photo.store.qq.com,qq,10),
   (125.39.127.30,s6.photo.store.qq.com,qq,15)})
```

group_records

Pig实例（6）- 循环处理（FOREACH）

对每个IP计算总流量

```
grunt> count_records = FOREACH group_records GENERATE group, COUNT(order_records.ip) as count,
 SUM(order_records.traffic) as sumTraffic;

grunt> describe count_records;
count_records: {group: chararray, count: long, sumTraffic: long}

grunt> dump count_records;
```

```
(125.39.127.20,{(125.39.127.20,b8.photo.store.qq.com,qq,10)})
(125.39.127.21,{(125.39.127.21,b5.photo.store.qq.com,qq,15),
   (125.39.127.21,b40.photo.store.qq.com,qq,15)})
(125.39.127.30,{(125.39.127.30,b25.photo.store.qq.com,qq,25),
   (125.39.127.30,b32.photo.store.qq.com,qq,10),
   (125.39.127.30,s6.photo.store.qq.com,qq,15)})
```

group_records

```
(125.39.127.20,1,10)
(125.39.127.21,2,30)
(125.39.127.30,3,50)
```

count_records

Pig实例（7）- 联结（JOIN）

将order_records和count_records按照ip和group进行联结

```
grunt> join_records = JOIN order_records BY ip, count_records BY group;
grunt> describe join_records;
join_records: {order_records::ip: chararray, order_records ::host: chararray,
order_records ::sp: chararray, order_records::traffic:int, count_records::group:chararray,
count_records::count: long, count_records::sumTraffic: long}
grunt> dump join_records;
```

(125.39.127.20,b8.photo.store.qq.com,qq,10)
(125.39.127.21,b5.photo.store.qq.com,qq,15)
(125.39.127.21,b40.photo.store.qq.com,qq,15)
(125.39.127.30,b25.photo.store.qq.com,qq,25)
(125.39.127.30,b32.photo.store.qq.com,qq,10)
(125.39.127.30,s6.photo.store.qq.com,qq,15)

(125.39.127.20,1,10)
(125.39.127.21,2,30)
(125.39.127.30,3,50)

→

(125.39.127.20,b8.photo.store.qq.com,qq,10,125.39.127.20,1,10)
(125.39.127.21,b5.photo.store.qq.com,qq,15,125.39.127.21,2,30)
(125.39.127.21,b40.photo.store.qq.com,qq,15,125.39.127.21,2,30)
(125.39.127.30,b25.photo.store.qq.com,qq,25,125.39.127.30,3,50)
(125.39.127.30,b32.photo.store.qq.com,qq,10,125.39.127.30,3,50)
(125.39.127.30,s6.photo.store.qq.com,qq,15,125.39.127.30,3,50)

join_records

Pig实例（8）- 结果

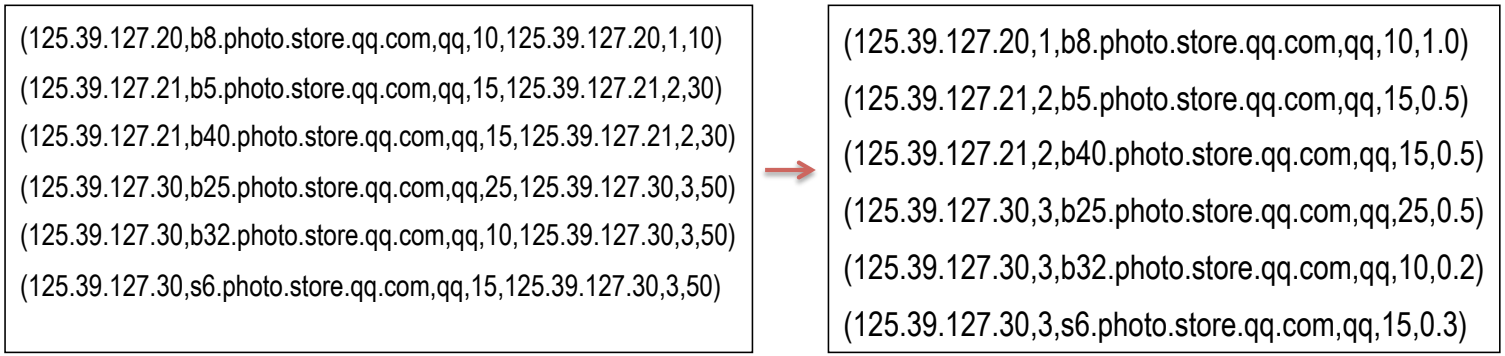
每个IP访问qq.com的host的流量占总流量比例

```
grunt> end_records = FOREACH join_records GENERATE order_records::ip, count_records::count, order_records::host,
 order_records::sp, order_records::traffic, (double)order_records::traffic/(double)count_records::sumTraffic as percent:double;

grunt> describe end_records;

end_records：{order_records::ip: chararray,count_records::count: long, order_records ::host: chararray, order_records::sp:
 chararray, percent: double}

grunt> dump end_records;
```



join_records　　end_records