Introduction to Pig programming

Yahoo Search Engineering

陳奕瑋 (**Yiwei Chen**)

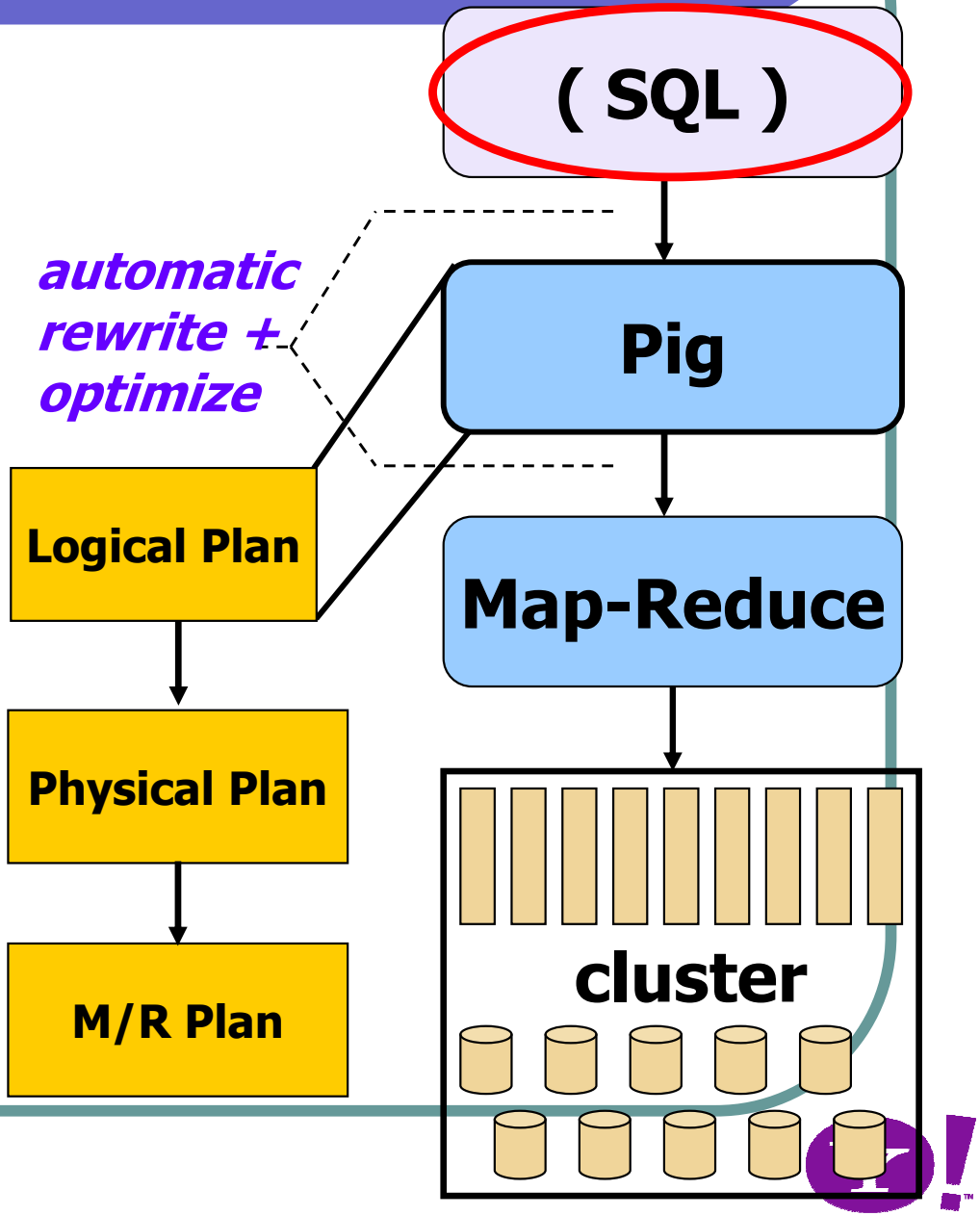
Pig Script Example

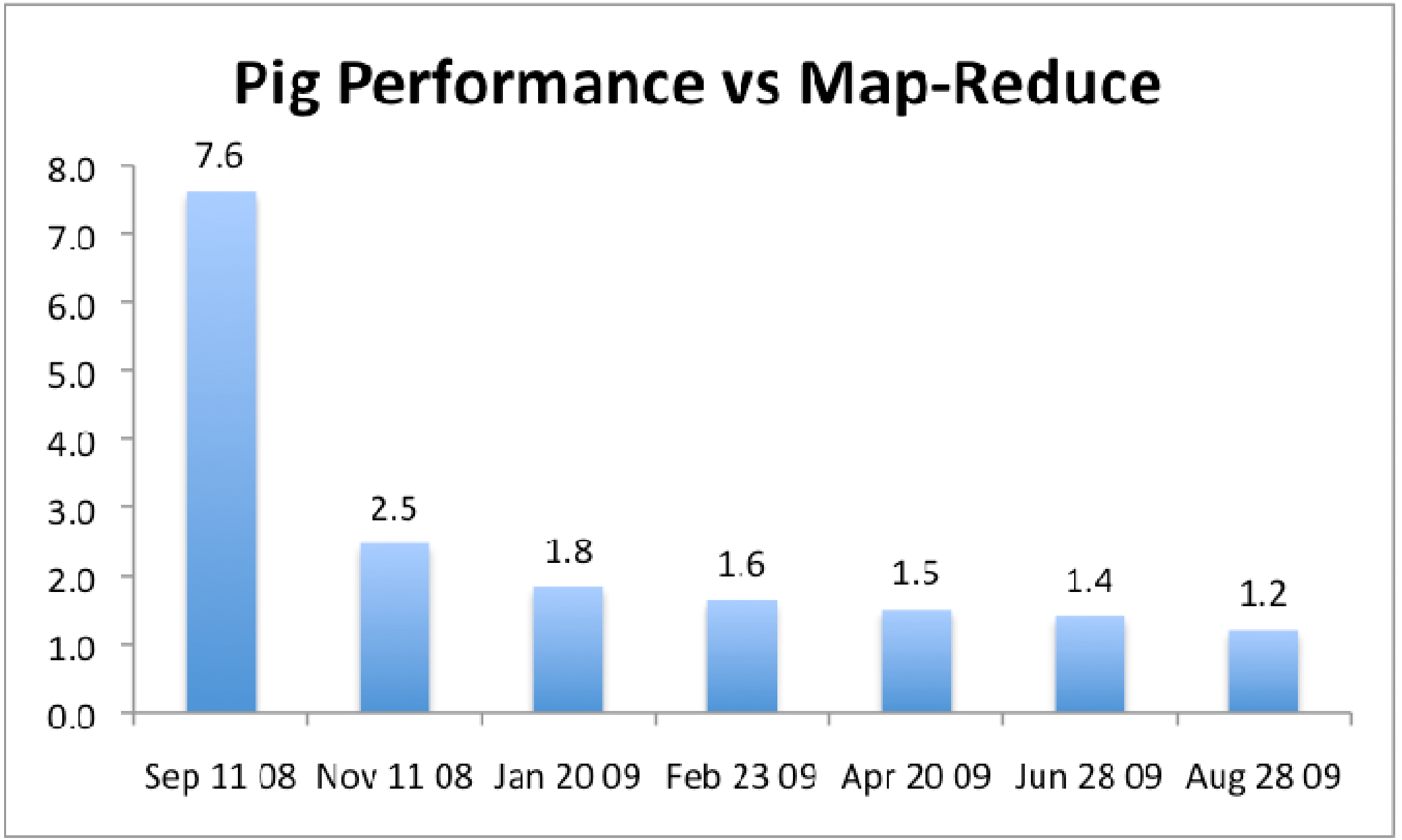
- Top sites visited by users aged 18 to 25

```
Users = LOAD ‘users.in’ AS (name, age);
Fltrd = FILTER Users by age >= 18 and age <= 25;

Pages = LOAD ‘pages.in’ AS (user, url);

Jnd   = JOIN Fltrd BY name, Pages BY user;
Grpd  = GROUP Jnd by url;
Smmd  = FOREACH Grpd GENERATE group, COUNT(Jnd) AS
        clicks;

Srtd  = ORDER Smmd BY clicks;
Top100 = LIMIT Srtd 100;

STORE Top100 INTO ‘top100sites.out’;
```

Pig script → Map/Reduce

- 不需懂底下 Map-Reduce 運作
- Pig 幫忙翻譯




Pig Performance vs Map-Reduce
8.0
7.0
6.0
5.0
4.0
3.0
2.0
1.0
0.0
7.6
2.5
1.8
1.6
1.5
1.4
1.2
Sep 11 08
Nov 11 08
Jan 20 09
Feb 23 09
Apr 20 09
Jun 28 09
Aug 28 09

How to execute

- Local:
  - `pig -x local foo.pig`
- Hadoop (HDFS):
  - `pig foo.pig`
  - `pig -Dmapred.job.queue.name=xxx foo.pig`
    - `hadoop queue -showacls`

How to execute

- Interactive pig shell
  - `$ pig`
  - grunt> _

Load Data

```
Users = LOAD 'users.txt'
        USING PigStorage(',') AS (name, age);
```

- LOAD … AS …
- PigStorage(‘,’) to specify separator

```
John,18
Mary,20
Bob,30
```

name	age
John	18
Mary	20
Bob	30

Filter

```
Fltrd = FILTER Users
        BY age >= 18 AND age <= 25;
```

- FILTER … BY …
  - constraints can be composite

name	age
John	18
Mary	20
Bob	30

name	age
John	18
Mary	20

Generate / Project

```
Names = FOREACH Fltrd GENERATE name;
```

- FOREACH … GENERATE

name	age
John	18
Mary	20

name
John
Mary

Store Data

```
STORE Names INTO 'names.out';
```

- STORE … INTO …
  - PigStorage(‘,’) to specify separator if multiple fields

Command - JOIN

```
Users = LOAD 'users' AS (name, age);
Pages = LOAD 'pages' AS (user, url);
Jnd   = JOIN Users BY name, Pages BY user;
```

name	age
John	18
Mary	20
Bob	30

user	url
John	yaho
Mary	goog
Bob	bing

name	age	user	url
John	18	John	yaho
Mary	20	Mary	goog
Bob	30	Bob	bing

Command - GROUP

```
Grpd  = GROUP Jnd by url;
describe Grpd;
```

name	age	url
John	18	yhoo
Mary	20	goog
Dee	25	yhoo
Kim	40	bing
Bob	30	bing

yhoo	(John, 18, yhoo) (Dee, 25, yhoo)
goog	(Mary, 20, goog)
bing	(Kim, 40, bing) (Bob, 30, bing)

Other Commands

- `PARALLEL` – controls #reducer
- `ORDER` – sort by a field
- `COUNT` – eval: count #elements
- `COGROUP` – structured JOIN
- More at http://hadoop.apache.org/pig/docs/r0.5.0/piglatin_reference.html

UDF

- http://hadoop.apache.org/pig/docs/r0.3.0/udf.html
- http://hadoop.apache.org/pig/javadoc/docs/api/
- PiggyBank
  - Pig users UDF repo
  - http://wiki.apache.org/pig/PiggyBank

References

- FAQ
  - http://wiki.apache.org/pig/FAQ
- Documentation
  - http://hadoop.apache.org/pig/docs/r0.5.0/
- Talks & papers
  - http://wiki.apache.org/pig/PigTalksPapers
  - http://www.cloudera.com/hadoop-training-pig-introduction