Interoperability in the Apache Hive Ecosystem

Sushanth Sowmyan (Hortonworks)
Mithun Radhakrishnan (Yahoo)
201404081645

Ecosystem

"An ecosystem is a community of living organisms in conjunction with the nonliving components of their environment, interacting as a system."

Hive Ecosystem

- Part of the larger Hadoop Ecosystem
- Very much a living part of it.
- Sharing data and interoperation between large scale data processing frameworks

Hadoop Ecosystem

- HDFS
- M/R framework

Problem : Anagrams

- Consider the following words:
  - Dictionary : `ACT, BAG, CAT, DOG, GAB, GOOD, MART, TRAM`
- Desired Result :
  - `[ [ACT, CAT], [BAG, GAB], [DOG], [GOOD], [MART,TRAM]]`

HashMap!

- Where the key is something uniquely common for a group of anagrams, and the value is a list of all the words that hash to that key.
- The trick is in coming up with what's appropriate for that key.

HashMap! (contd…)

```
{
        “ACT”  => [ “ACT” , “CAT” ],
        “ABG”  => [ “BAG” , “GAB” ],
        “DGO”  => [ “DOG” ],
        “DGOO” => [ “GOOD” ],
        “AMRT” => [ “MART” , “TRAM” ],
}
```

Memory constraints

- What if above hashmap is too large to fit into memory?
- We can flatten it so it's a list of key-values, and constantly emit them:

  -
    ```
    {"ACT" => "ACT"}, {"ABG" => "BAG"},
    {"ACT" => "CAT"}, ["DGO" => "DOG"},
    {"ABG" => "GAB"}, …
    ```
- Then we can sort by the first key, and group.

Key-generate-Group?

- Map , Classify, Transform
- Group, Reduce, Aggregate, FoldLeft

Hadoop

- Moving computation to data – data locality
- Generally for jobs where the notion of fitting datasets in memory without some manner of massaging is infeasible.

What is Hive?

- Database Engine that works on Hadoop
- Data processing/Data warehouse infrastructure on Hadoop, but with the following notions from the get-go:
    - Tabular data
    - Storage engine abstractions
    - Metadata-driven
    - based on SQL
    - Query optimizations, indexes
    - Query evaluation and M/R job generation engine

Why Hive? (Why not Pig?)

- Pig comes from an ETL world and appeals to the scripters/programmers. Hive appeals to analysts more familiar with SQL.
- BI tools written against SQL
- Pig was written to be able to "eat anything." Hive considers schema and structure to be central, and dines with a knife and fork.
- Ultimately, stylistic differences in communities.

Why Hive (What about HBase?)

- Excellent for random r/w
- Excellent for Billions of rows, millions of columns sparse tables.
- These are not common use-cases for relational operations, and speak to a very different mindset and optimization requirement.
- NoSQL ?

What is HCatalog?

- Formerly Howl, which was formerly Owl.
- Apache Incubator project that's now part of hive
- Metadata-backed storage abstraction
- Allows for interoperability between various tools – Pig or a custom M/R program can now read/write Hive tables.
- Allows for data migrations, and format-independence.

What is HiveServer2?

- Next step in evolution of Hive usage:
  - Lone fat-client with all the logic, mostly a way to use SQL on top of HDFS data
  - Need for sharing metadata across a team - common database server for metadata
  - Need for protecting kinds of access to the database server, and not deploying database login credentials across the board – metastore server
  - Need for access control on metastore server, protected HDFS access – Hive-as-a-service
  - JDBC/ODBC connection modes, SQLLine(Beeline) support - HS2

Project Starling

Hadoop Introspection, for make benefit Audit and Performance Tuning

Going “Meta”

- “How do we store, catalogue and analyze the logs generated from the thousands of Hadoop jobs run across the dozens of Hadoop clusters across the company?”
- What insights do we gather from analyzing Hadoop logs?

Example Insights

- Metering:
  - Which user/group uses the most compute-time?
  - Who's hogging HDFS space?
- Job analysis:
  - Weapon of Choice: M/R, Pig, Hive
  - Pig 0.10 vs. 0.11
  - Job-failures: OOM, Auto-tune mapreduce.map.memory.mb.
- Data analysis:
  - Which dataset is most popular?
  - Last-access-time:
    - Tune data-retention periods, Replication factor, Cold Storage
- Analyze using SQL, Graph with Excel/Tableau/MicroStrategy

Hive

- Query large datasets on the Hadoop DFS (and elsewhere) using SQL-ish syntax
  - E.g. `SELECT page_url FROM page_views WHERE user_id = ‘mithunr’ AND dt=‘20140101’;`
- Represent large datasets as databases, tables and partitions
  - HCatalog: Metadata storage
  - Wrappers for Map Reduce and Pig
- JDBC/ODBC connectivity:
  - HiveServer2
  - Business Intelligence tools: Tableau, MicroStrategy

Example Code

- CREATE TABLE job_info(job_id STRING, job_name STRING, user STRING, queue STRING, priority STRING, status STRING, run_time BIGINT) partitioned by (cluster_name STRING, dt STRING) STORED AS RCFILE;

- Hive query:

```
SELECT user_id, sum(run_time) AS total_runtime
FROM job_info
WHERE cluster_name='staging3' and dt BETWEEN '20140101' AND '20140131'
GROUP BY user_id ORDER BY total_runtime DESC LIMIT 10;
```

- Pig:

```
job_info = LOAD 'starling.job_info' using org.apache.hive.hcatalog.pig.HCatLoader();
january_staging_job_info = FILTER job_info BY dt >= '20140101' AND dt <= '20140131' AND
cluster_name='staging3';
grouped_job_info = GROUP january_staging_job_info by user_id;
agg_runtimes = FOREACH grouped_job_info GENERATE group, SUM(run_time) total_runtime;
sorted_agg_runtimes = ORDER agg_runtimes BY total_runtime DESC;
top_users = LIMIT sorted_agg_runtimes LIMIT 10;
DUMP top_users;
```

- HiveServer2:
  - jdbc:hive2://hive-server2.myth.net:10000/starling

DistCp

- Uses MapReduce to copy files between clusters.
- E.g.

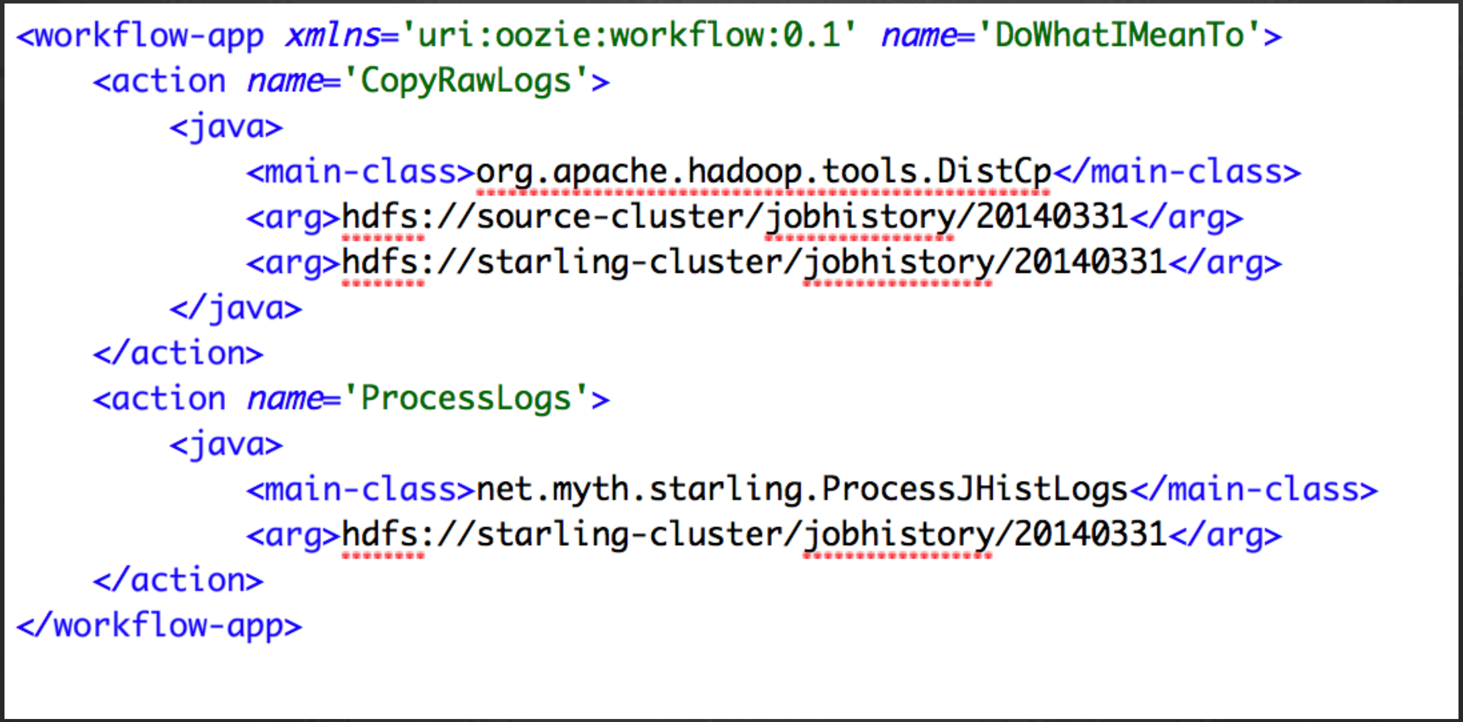
```
hadoop distcp -m 20 \

      hftp://source-cluster:50070/jobhistory/20140331/* \

      hdfs://starling-cluster:8020/jobhistory/20140331
```

Parser Reference

- org.apache.hadoop.tools.rumen.JobHistoryParser
- Hadoop File System Offline Image View (HDFS OIV)
- Convert from Raw log-lines to HiveRecords.

Oozie

- cron for Hadoop



```xml
<workflow-app xmlns='uri:oozie:workflow:0.1' name='DoWhatIMeanTo'>
    <action name='CopyRawLogs'>
        <java>
            <main-class>org.apache.hadoop.tools.DistCp</main-class>
            <arg>hdfs://source-cluster/jobhistory/20140331</arg>
            <arg>hdfs://starling-cluster/jobhistory/20140331</arg>
        </java>
    </action>
    <action name='ProcessLogs'>
        <java>
            <main-class>net.myth.starling.ProcessJHistLogs</main-class>
            <arg>hdfs://starling-cluster/jobhistory/20140331</arg>
    </action>
</workflow-app>
```

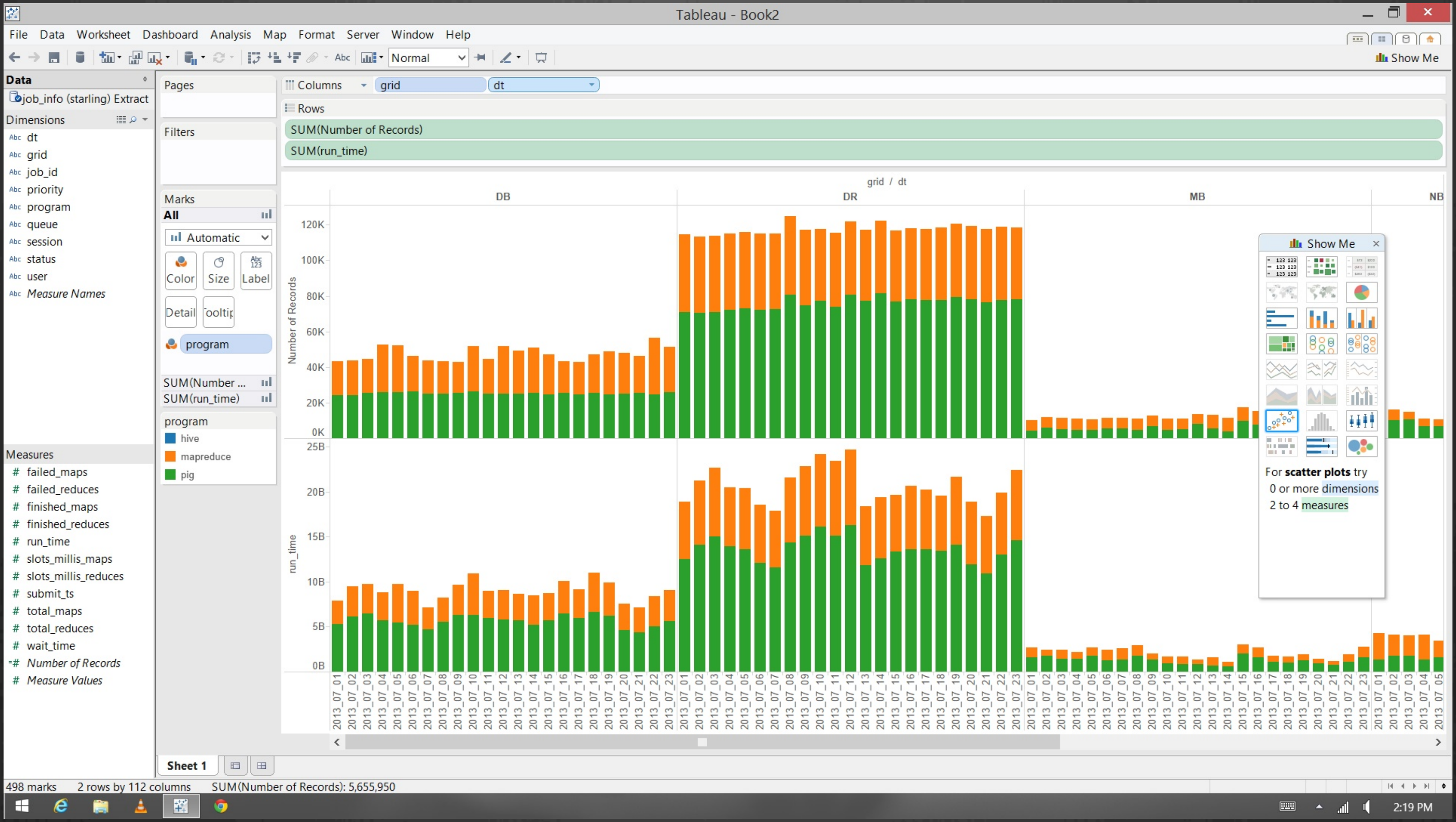


Tableau - Book2
File Data Worksheet Dashboard Analysis Map Format Server Window Help
Normal
Show Me
Data
job_info (starling) Extract
Dimensions
dt
grid
job_id
priority
program
queue
session
status
user
Measure Names
Measures
failed_maps
failed_reduces
finished_maps
finished_reduces
run_time
slots_millis_maps
slots_millis_reduces
submit_ts
total_maps
total_reduces
wait_time
Number of Records
Measure Values
Pages
Filters
Marks
All
Automatic
Color
Size
Label
Detail
Tooltip
program
SUM(Number ...
SUM(run_time)
program
hive
mapreduce
pig
Columns
grid
dt
Rows
SUM(Number of Records)
SUM(run_time)
grid / dt
DB
DR
MB
NB
Number of Records
run_time
Show Me
For scatter plots try
0 or more dimensions
2 to 4 measures
Sheet 1
498 marks
2 rows by 112 columns
SUM(Number of Records): 5,655,950
2:19 PM

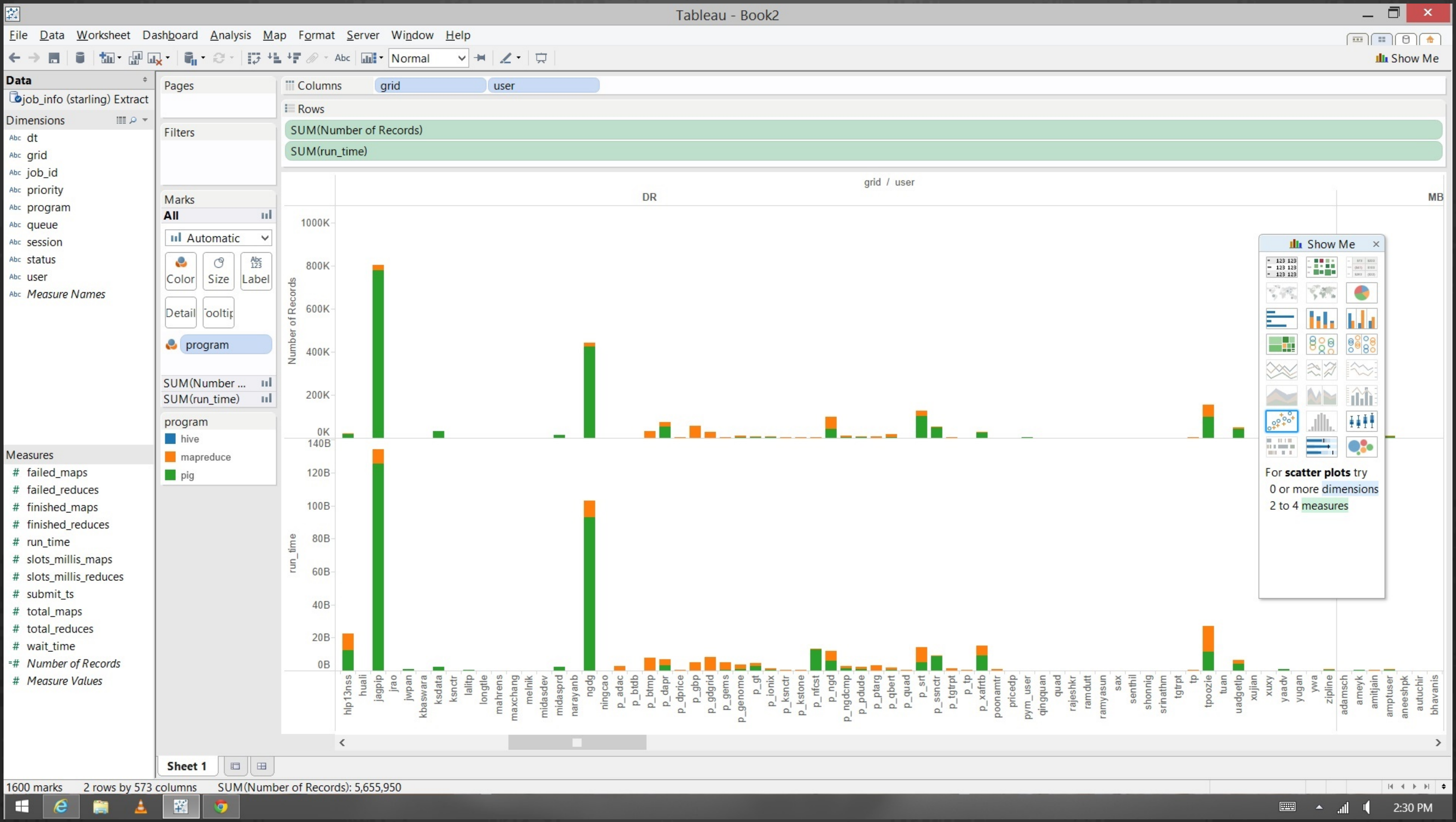


Tableau - Book2
File Data Worksheet Dashboard Analysis Map Format Server Window Help
Normal
Show Me
Data
job_info (starling) Extract
Dimensions
dt
grid
job_id
priority
program
queue
session
status
user
Measure Names
Measures
failed_maps
failed_reduces
finished_maps
finished_reduces
run_time
slots_millis_maps
slots_millis_reduces
submit_ts
total_maps
total_reduces
wait_time
Number of Records
Measure Values
Pages
Filters
Marks
All
Automatic
Color
Size
Label
Detail
Tooltip
program
SUM(Number ...
SUM(run_time)
program
hive
mapreduce
pig
grid / user
DR
MB
Number of Records
1000K
800K
600K
400K
200K
0K
run_time
140B
120B
100B
80B
60B
40B
20B
0B
hlp13nss huali jagpip jrao jwpan kbaswara ksdata ksnctr lalitp longtie mahrens maxchang melnik midasdev midasprd narayanb ngdg ningcao p_adac p_btdb p_btmp p_dapr p_dprice p_gbp p_gdgrid p_gems p_genome p_gt p_ionix p_ksnctr p_kstone p_nfcst p_ngd p_ngdcmp p_pdude p_ptarg p_qbert p_quad p_srt p_ssnctr p_tgtrpt p_tp p_xafrtb poonamtr pricedp pym_user qingquan quad rajeshkr ramdutt ramyasun sax senthil shonnig srinathm tgtrpt tp tpoozie tuan uadgetlp xujian xuxy yaadv yugan ywa zipline adamsch ameyk amitjain amptuser aneeshpk autuchir bhavanis
Show Me
Sheet 1
2:30 PM

Columns: grid | user

Rows: SUM(Number of Records) | SUM(run_time)

For **scatter plots** try
0 or more dimensions
2 to 4 measures

1600 marks 2 rows by 573 columns SUM(Number of Records): 5,655,950

Questions?

EOF