Applied Storage Performance For Big Analytics

Hubbert Smith
LSI
It’s NOT THIS SIMPLE !!!
Theoretical vs Real World

Theoretical & Lab
Storage Workloads

Real World Server
Storage Workload

Real World Cluster
Storage Workload

Ingest
MapReduce (transform)
Query
Hadoop Software Stack

- **Ingest**: “get the Data”
- **Store, Replicate**
- **Batch Reduce Store result**
- **Do something with the result on another cluster**

MisConfiguration is big 35% of Cloudera support tickets Mostly RAM and Storage related
Hadoop Data Flow
Latency and Throughput

<table>
<thead>
<tr>
<th>Throughput  (MB/s)</th>
<th>Latency  (ms)</th>
<th>Excessive Latency</th>
</tr>
</thead>
<tbody>
<tr>
<td>What you are Sold</td>
<td>What you actually Get</td>
<td>Drive Crash</td>
</tr>
</tbody>
</table>

[Images of speed limit sign, traffic lights, and a crashed car representing latency and throughput concepts.]
Big Data Origins

- Mega-Data Center Hardware Engineering

- Data-Source (internet)
  - Ingest-Transform
  - LOAD
  - Interactive Query
  - Users (search, social)

- Cluster A “Get the data”
- Cluster B “Use the Data”
Measures of Success

Mega-Data Center
Performance Engineering

- Terasort
- Teragen

Ingest-Transform

LOAD

Cluster A
“Get the data”

Cluster B
“Use the Data”

Users (search, social)

Interactive Query

About 6,350,000 results (0.36 seconds)
Analytics for Business and Industry (not web-scale): 
TIME SENSITIVE, HIGH VALUE, MIXED USE

- Fraud Risk
  - Currency Fraud
  - Credit card fraud
  - Insurance fraud

- Operations
  - Preventative Maintenance
  - Moving inventory to demand
  - Customer insight, customer retention

- Scientific
  - Genome project
  - Clinical treatment: complex symptoms matched to complex treatments
Measures of success
Analytics for Business and Industry

❖ One Cluster
Simultaneous workloads:
“GET the Data”
“USE the Data”

❖ Terasort, Teragen
❖ HiBench
❖ FIO, TestDFSIO
❖ YCSB
Benchmark tool

- Java application
  - Many systems have Java APIs
  - Other systems via HTTP/REST, JNI or some other solution

GOOD
- “Use the data” focus
- Extensible workloads

NOT GOOD
- Not easily used in real world sizing (#Users)
- Cryptic output
YCSB understanding the output

I search in the site and found another question about that, but there's no answers.

I'm executing YCSB tool on a cassandra cluster, and the output of YCSB is:

```
[OVERALL], RunTime(ms), 302016.0 -&gt; 05 mins 02 secs
[OVERALL], Throughput(ops/sec), 3311.0828565374018
[UPDATE], Operations, 499411
[UPDATE], AverageLatency(us), 2257.980987603397
[UPDATE], MinLatency(us), 389
[UPDATE], MaxLatency(us), 169380
[UPDATE], 95thPercentileLatency(ms), 4
[UPDATE], 99thPercentileLatency(ms), 8
[UPDATE], Return=0, 499411
[UPDATE], 0, 50039
[UPDATE], 1, 222610
[UPDATE], 2, 138349
[UPDATE], 3, 49465
```

How much hardware??
How many users??

http://stackoverflow.com/questions/19998009/ycsb-understanding-the-output
YCSB Output (Graphing)
from 2010 whitepaper, your mileage will vary

- 95/5 Read/update

- 50/50 Read/update

How much hardware??
How many users??

https://github.com/brianfrankcooper/YCSB/wiki/core-workloads
Elasticity

- Run a read-heavy workload on 2 servers; add a 4th, then 5th, then 6th server.

Comment: HBase shows a small latency bump as the cluster reconfigures. But data is not moved to the new server until a compaction is performed (not shown in the graph).

SLA Events
- Adding Server
- Fail/Rebuild Disk
- Meaningful in High Value, Time Sensitive use cases

https://github.com/brianfrankcooper/YCSB/wiki/core-workloads
SLA? NOT Replication and Rebuild

- SLA?
- Size hardware for best case? for likely case? for worst case?
- Sharding or RAID
Just add servers – key ratios: IOPs/$, Usable TB/$, #Users/$

<table>
<thead>
<tr>
<th>Storage Arrays</th>
<th>Config1</th>
<th>Config2</th>
</tr>
</thead>
<tbody>
<tr>
<td>IOP</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Usable TB</td>
<td></td>
<td></td>
</tr>
<tr>
<td>$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Watts</td>
<td></td>
<td></td>
</tr>
<tr>
<td>#Users</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Key Ratios</th>
<th>Config1</th>
<th>Config2</th>
</tr>
</thead>
<tbody>
<tr>
<td>IOP/$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>IOP/Watt</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Usable TB/$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Usable TB/watt</td>
<td></td>
<td></td>
</tr>
<tr>
<td>#Users/$</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Avoid throwing HW & Money at the problem
Analytics for Business and Industry

Today's assessment:

- Business and industry is different than Webscale
  - High Value, Time Sensitive -> SLA
  - “Just add servers” approach is NOT feasible
  - Same cluster -> shared workload: get data, use data

- Performance tools are disconnected, befuddled, fail to measure relevant stuff

- The $400M dollar man

- Confront “Fast-Cheap-Safe”
Hot Data (fast)
Cool Data (cheap, safe)

- Apache.org JIRA “Caching”
  - Couchdb-1668
  - HDFS-6151
  - HDFS-6122
  - HDFS-6109
  - HDFS-4949
  - HDFS-5149
  - Hbase bucket cache
  - Federated data
  - Hadoop 2.4 datanode cache
“get the data”, “use the data” and caching

- Top graph is ETL “get the data”
- Blue is a cache hit (good)
- Orange is a cache fill (not good)
- Bottom graph is Analysis “use the data”

Caching Works!!!
Hot Data (fast)  
Cool Data (cheap, safe)

“It’s better to move the compute than move the data”  
Disaggregation Works!!!
LOAD SIMULATORS

- YARN Scheduler Load Simulator (SLS)
- Jmeter (apache)
- LoadSim (sourceforge)
- HP LoadRunner (HP)
Future of Analytics for Business and Industry

- Business and Industry: High Value, Time Sensitive
- Relevant Measures
  - Load Simulator, #Users per Cluster
  - Not MB/s, not Latency, not Terasort, not Teragen
  - Fast, Cheap, Safe (learn from the past “fast only”)
- Hot data, Cold data, Federated data
  - The cheapest IO is one that doesn’t happen federated data, sharding, not replication
- Multi-Tenant, Cloud Hosting, SLA
- Rack-level Reference designs, runbook baseline
Resources

- Katie Ting, Cloudera - Hadoop World Troubleshooting [VIDEO]
- Greg Rahn, Cloudera Strata – performance analysis and tuning for Cloudera Impala [SLIDES]
- Kim Leyenaar, LSI – Accelerate MongoDB with LSI Nytro MegaRAID [PAPER]
- Yahoo Cloud Services Benchmark, YCSB [PAPER]
- Yahoo-inc. Brian Cooler [YCSB Cloud serving benchmark]
- Cisco Common Platform Architecture with Cloudera [Cisco CPA]
- Univ. SanDiego Super ComputerSDSC [Hadoop Performance]
- Argil Data Mailing List [info@argildata.com]
- Whitepaper – [the datacenter as the computer]

Hubbert.smith@lsi.com, hubbert@hubbertsmith.com
Conclusion and Call to Action

- The value is in the data, unlock it.
- Relevant measures – Users/$
- Fast-cheap-safe, SLAs, Cloud services/Hosting
- Beyond MB/s and Latency and beyond ingest (terasort/teragen)
- The cheapest I/O is one that doesn’t happen (consolidated, less replication, federated)
- “Better to move the compute than move the data”

- YCSB
- Load Simulators
- “Use the data” performance focus with Flash and Cache
- Get involved in the community
Running the Write Job

hadoop jar /usr/lib/gphd/hadoop-mapreduce/hadoop-mapreduce-client-jobclient-2.0.2-alpha-gphd-2.0.1.0-tests.jar TestDFSIO -write -nrFiles 64 -fileSize 16GB -resFile /tmp/TestDFSIOwrite.txt

13/08/21 10:56:45 INFO fs.TestDFSIO: ----- TestDFSIO ----- : write
13/08/21 10:56:45 INFO fs.TestDFSIO: Number of files: 64
13/08/21 10:56:45 INFO fs.TestDFSIO: Total MBytes processed: 1,048,576
13/08/21 10:56:45 INFO fs.TestDFSIO: Throughput mb/sec: 23.0
13/08/21 10:56:45 INFO fs.TestDFSIO: Average IO rate mb/sec: 23.1
13/08/21 10:56:45 INFO fs.TestDFSIO: IO rate std deviation: 1.5
13/08/21 10:56:45 INFO fs.TestDFSIO: Test exec time sec: 797

[gpadmin@hdm3 ~]$ hdfs dfs -cat /benchmarks/TestDFSIO/io_write/part*
  f:rate 1481181.8
  f:sqrate 3.4433252E7
  l:size 1099511627776
  l:tasks 64