Data Warehousing and Analytics Infrastructure at Facebook

SIGMOD 2010 paper

6.S897 Presentation

Manasi Vartak

Data Applications

Site features e.g. recommendations, ad insights

Ad-hoc analysis e.g. A/B testing, engagement

Business intelligence e.g. dashboards

Diverse Query Workload

- 10K jobs/day
- Variety of query complexity
- Mix of ad hoc queries, periodic batch queries
- Different SLAs, resource-needs, data delivery deadlines

Data Growing Rapidly

- More users, more instrumentation (logging)
- >2.5 PB in warehouse, loads > 10TB/data.
- Growing 2X/6 months

Strong Scalability Requirements: horizontal scaling, cheap commodity hardware

Core Technologies

- Hadoop
  - Distributed File System + MapReduce
- Hive (developed at FB)
  - SQL, warehousing tools on top of Hadoop
- Scribe (developed at FB)
  - Log collection from thousands of servers

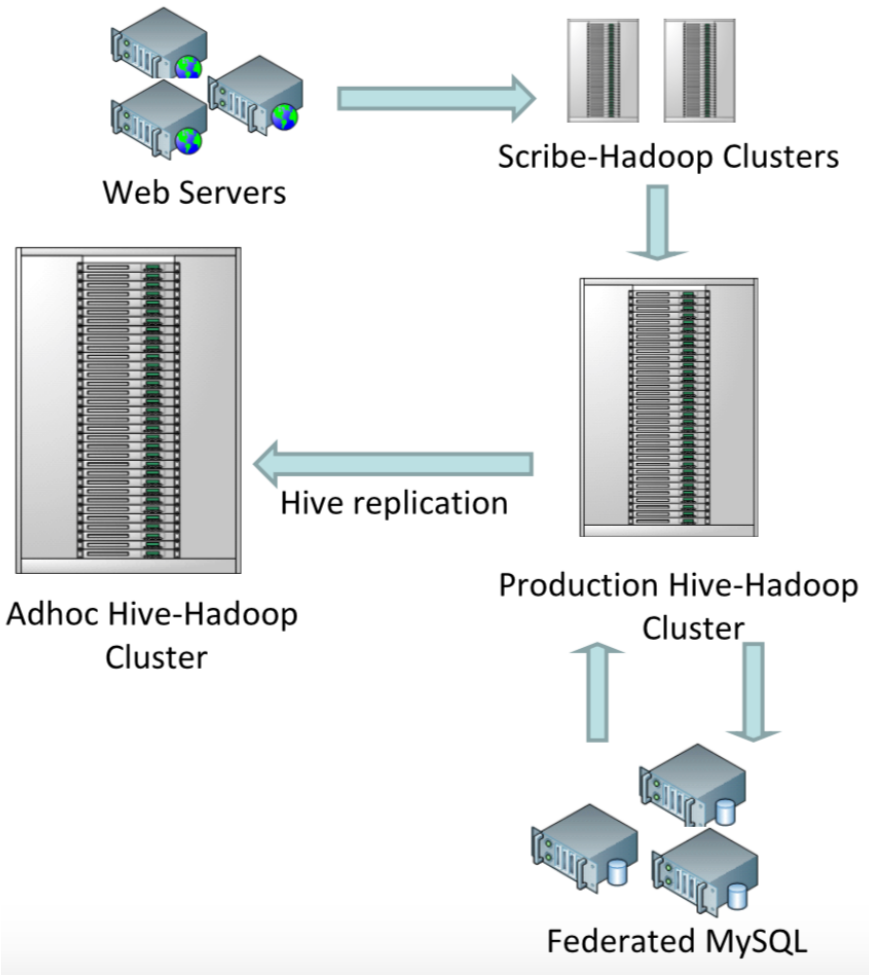
DataFlow Architecture

2 data sources:

- MySQL tier (FB site data)
- web tier (log data)

DataFlow

- MySQL tier scraped *daily* —> Hive-Hadoop cluster
- Web servers push logs to scribe (15 mins) servers —> Hive-Hadoop cluster
- 2 replicated Hive-Hadoop clusters (command logging)
- Queries run on Hive-Hadoop clusters



Storage

Raw Data

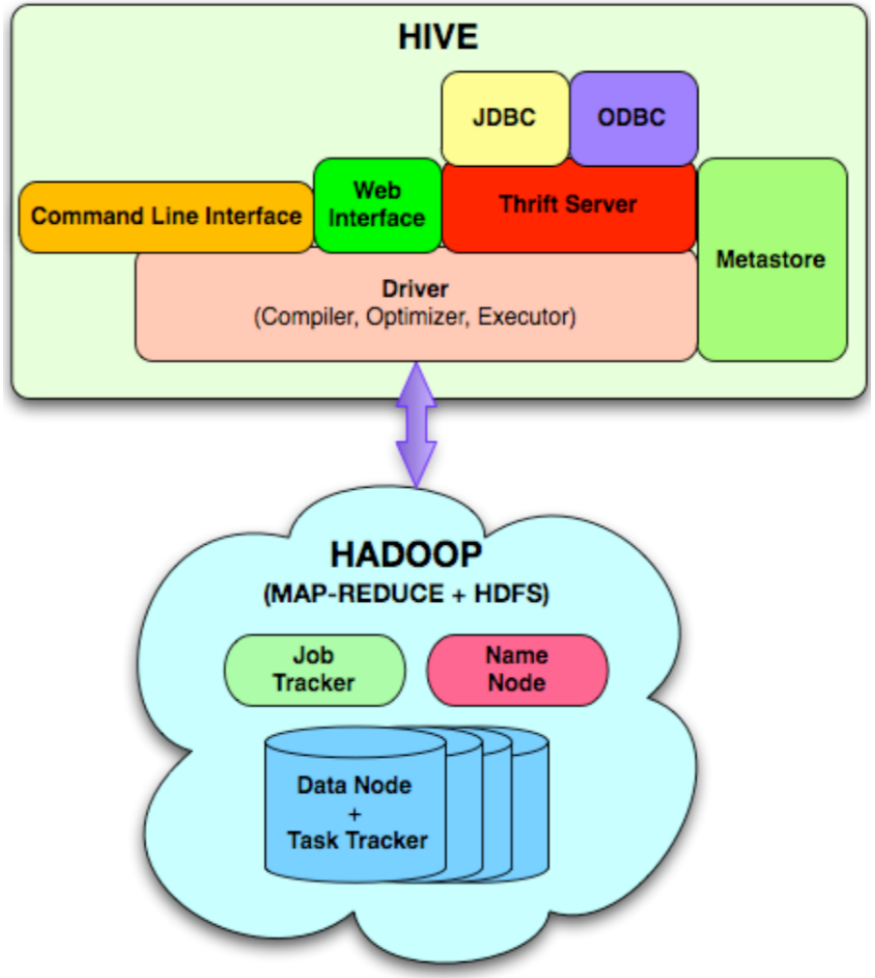
- Space a constraint w/high ingest rates and need for historical data
- Prod cluster: 1 month's worth of data; ad-hoc: all
- Data compressed (gzip); Hive - row columnar compression
- HDFS stores 3 copies of a file —> 2 copies of file, 2 copies of error codes (2.2 copies)

Metadata in NameNode (master server w/file —> block mappings)

- 100M blocks, files; heap size on NameNode ~ 48GB
- More efficient data structures; concatenating small files

Hive

- Data warehousing framework on top of Hadoop
- Predominantly used for querying and analysis
- Increased analyst productivity
- Components:
  - Tables, partitions in HDFS
  - Metadata in Metastore
  - Driver: parses HiveQL query and writes MR jobs
  - Optimizations: predicate push-down, join re-ordering



Tools on Top of Hive

- Hive accessed through HiPal (web interface for making querying graphically), Hive CLI
- Data discovery: wiki-like tool to keep track of datasets
- Tools to extract lineage information, identify experts from query logs
- DataBee — framework for specifying periodic jobs

Resource Sharing

- Ad-hoc queries and batch queries on same cluster
  - Ad-hoc queries — minimal response time
  - Periodic batch queries — predictable time, available before a deadline
- Jobs in Hadoop cannot be pre-empted (* long jobs can hog resources and starve ad-hoc queries)
- Hadoop Fair Share Scheduler: users divided into pools and resources shared equally between pools
  - Additions to scheduler to monitor memory and CPU consumption and kill tasks if usage above threshold
- Prod cluster vs. ad-hoc cluster

Operations

- Need to keep these systems running, all the time!
- Monitoring:
  - System: CPU usage, I/O activity, memory usage
  - Hadoop: JobTracker (e.g. job submissions), NameNode (e.g. space usage)
- Alerts, dashboards, figure out provisioning needs

Conclusion & Highlights

- Sheer size of data (10TB/day ingest in 2010)
- Diverse query workload (resource sharing)
- Importance of horizontal scaling/commodity H/W
- Open source
- Monitoring and ops