Apache Hadoop Goes Realtime at Facebook

Guide -
Dr. Sunny S. Chung

Presented By-
Anand K Singh
Himanshu Sharma
Index

- Problem with Current Stack
- Apache Hadoop and Hbase
- Zookeeper
- Applications of HBase at Facebook
- Future of Hbase at Facebook
- Conclusion
Problems with Current stack

- MySQL is stable, but...
  - Not inherently distributed
  - Table size limitation
  - Inflexible schema

- Hadoop is scalable, but...
  - MapReduce is slow and difficult
  - Does not support random writes
  - Poor support for random reads
Optimal solutions

- High-throughput, persistent key-value
  - Tokyo Cabinet
- Large scale data warehousing
  - Hive/Hadoop

- Solution: A near realtime Hadoop/HBase that is modified to provide scalability, consistency, availability and a compatible data model.
Need of data store

- Requirements for Facebook Messages
  - Massive datasets, with large subsets of cold data
  - Elasticity and high availability
  - Strong consistency within a datacenter
  - Fault isolation
Introduction to HBase

- Hbase: A NoSQL database that utilizes an on-disk column storage format.
- Hbase USP: Provides fast key-based access to a specific cell or data or a range of cells.
- Based on Google's BigTable but extends it
- Has Row atomicity and read-modify-write consistency
- Simplifies a lot of tasks related to distributed databases.
- Tagline: Random access to web-scale data
Why Hbase?

- In early 2010, engineers at FB compared DBs
  - Apache Cassandra, Apache HBase, Sharded MySQL
- Compared performance, scalability, and features
  - HBase gave excellent write performance, good reads
  - HBase already included many good features
    - Atomic read-modify-write operations
    - Multiple shards per server
    - Bulk importing
Apache HBase

- Originally part of Hadoop
  - HBase adds random read/write access to HDFS

- Required some Hadoop changes for FB usage
  - File appends
  - HA NameNode
  - Read optimizations
HBase System Overview
Introduction to Zookeeper

- Zookeeper: A software service for a distributed environment that coordinates and configures different machines in a centralized way.
- A change is not considered successful until it has been written to a quorum
- A leader is elected within the ensemble for conflicts
- In HBase, ZooKeeper coordinates and shares state between the Masters and RegionServers.
- Tagline: Enables highly reliable distributed coordination
HDFS – Hadoop Distributed File System

- HDFS was originally designed to be a file system to support offline MapReduce application.
- In HDFS scalability and streaming performance are most critical.
- HDFS linear scalability and fault tolerance results in huge cost savings across the enterprise.

**PROBLEM:**

- The design of HDFS has a single master – the NameNode. Whenever the master is down, the HDFS cluster is unusable until the NameNode is back up.
Realtime HDFS - AvatarNode
Realtime HDFS – Logging

- Enhancements to Transaction logging:
  - Conventional HDFS
  - Change: Let the StandbyNode always know about block ids.
  - Avoidance of partial reads between Active and Standby node
Applications of Hbase at Facebook

- Titan
- Facebook Messages
Facebook Messaging

- High write throughput for every message, like instant message, SMS, and e-mail Search indexes for all.
- Denormalized Schema

- A product at massive scale on day one
  - 6k messages a second
  - 50k instant messages a second
  - 300TB data growth/month compressed
Puma

- Introduction
- Realtime MapReduce
- Facebook Insights
Puma

- **Realtime Data Pipeline**
  - Utilize existing log aggregation pipeline (Scribe-HDFS)
  - Extend low-latency capabilities of HDFS (Sync+PTail)
  - High-throughput writes (HBase)

- **Support for Realtime Aggregation**
  - Utilize HBase atomic increments to maintain roll-ups
  - Store checkpoint information directly in HBase
Puma as Realtime MapReduce

- Map phase with PTail
  - Divide the input log stream into N shards
  - First version only supported random bucketing
  - Now supports application-level bucketing

- Reduce phase with HBase
  - Every row+column in HBase is an output key
  - Aggregate key counts using atomic counters
  - Can also maintain per-key lists or other structures
Puma for Facebook Insights

- Realtime URL/Domain Insights
  - Domain owners can see deep analytics for their site
  - Clicks, Likes, Shares, Comments, Impressions
  - Detailed demographic breakdowns
  - Top URLs calculated per-domain and globally

- Massive Throughput
  - Billions of URLs
  - > 1 Million counter increments per second
ODS
Operational Data Store

- System metrics (CPU, Memory, IO, Network)
- Application metrics (Web, DB, Caches)
- Facebook metrics (Usage, Revenue)
  - Easily graph this data over time
  - Supports complex aggregation, transformations, etc.

- Difficult to scale with MySQL
  - Millions of unique time-series with billions of points
  - Irregular data growth patterns
Future of Hbase at Facebook

- User and Graph Data
Hbase at Facebook

- Looking at Hbase to augment MySQL
  - Only single row ACID from MySQL is used
  - DBs are always fronted by an in-memory cache
  - HBase is great at storing dictionaries and lists

- Database tier size determined by IOPS
  - HBase does only sequential writes
  - Lower IOPs translate to lower cost
  - Larger tables on denser, cheaper, commodity nodes
Conclusion

- Facebook investing in Realtime Hadoop/HBase
  - Work of a large team of Facebook engineers
  - Close collaboration with open source developers
  - One addition was YARN
    - A powerful cluster resource management
    - Added the High Availability feature to NameNode by introducing the Hot/Standyby NameNode.