HBase Internals and Operations

SRECon19 Asia/Pacific
June 13, 2019

Biju Nair
Software Engineer
bnair10@bloomberg.net

TechAtBloomberg.com
Agenda

• Introduction to HBase
• Operating HBase
• Questions
Bloomberg in a nutshell

The *Bloomberg Terminal* delivers a diverse array of information on a single platform to facilitate financial decision-making.
Bloomberg technology by the numbers

• 5,000+ software engineers
• 150+ technologists and data scientists devoted to machine learning
• One of the largest private networks in the world
• 120 billion pieces of data from the financial markets each day, with a peak of more than 10 million messages/second
• 2 million news stories ingested / published each day (500+ news stories ingested/second)
• News content from 125K+ sources
• Over 1 billion messages and Instant Bloomberg (IB) chats handled daily

TechAtBloomberg.com
HBase at Bloomberg

• Started with v0.94.6

• >2 billion reads per day

• >1 billion writes per day

• 51+ TB of compressed data stored in HBase
HBase Principles

• Ordered Key Value Store

• Distributed shared nothing
<table>
<thead>
<tr>
<th>Key</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Key-9999</td>
<td>Value-a</td>
</tr>
<tr>
<td>Key-9998</td>
<td>Value-b</td>
</tr>
<tr>
<td>Key-9997</td>
<td>Value-c</td>
</tr>
<tr>
<td>Key-9996</td>
<td>Value-d</td>
</tr>
<tr>
<td>Key-9995</td>
<td>Value-e</td>
</tr>
<tr>
<td>Key-9994</td>
<td>Value-f</td>
</tr>
</tbody>
</table>
## Ordered Key Value

<table>
<thead>
<tr>
<th>Key</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Key-9999</td>
<td>Value-a</td>
</tr>
<tr>
<td>Key-9998</td>
<td>Value-b</td>
</tr>
<tr>
<td>Key-9997</td>
<td>Value-c</td>
</tr>
<tr>
<td>Key-9996</td>
<td>Value-d</td>
</tr>
<tr>
<td>Key-9995</td>
<td>Value-e</td>
</tr>
<tr>
<td>Key-9994</td>
<td>Value-a</td>
</tr>
<tr>
<td>Key-9993</td>
<td>Value-g</td>
</tr>
</tbody>
</table>

Lexicographic order
Ordered Key Value

Lexicographic order

Region

<table>
<thead>
<tr>
<th>Key</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Key-9999</td>
<td>Value-a</td>
</tr>
<tr>
<td>Key-9998</td>
<td>Value-b</td>
</tr>
<tr>
<td>Key-9997</td>
<td>Value-c</td>
</tr>
<tr>
<td>Key-9996</td>
<td>Value-d</td>
</tr>
<tr>
<td>Key-9995</td>
<td>Value-e</td>
</tr>
<tr>
<td>Key-9994</td>
<td>Value-a</td>
</tr>
<tr>
<td>Key-9993</td>
<td>Value-g</td>
</tr>
</tbody>
</table>
## Distributed Ordered Key Value

<table>
<thead>
<tr>
<th>Key</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Key-199</td>
<td></td>
</tr>
<tr>
<td>Key-198</td>
<td></td>
</tr>
<tr>
<td>Key-197</td>
<td></td>
</tr>
</tbody>
</table>

... ordered...

<table>
<thead>
<tr>
<th>Key</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Key-499</td>
<td></td>
</tr>
<tr>
<td>Key-498</td>
<td></td>
</tr>
<tr>
<td>Key-497</td>
<td></td>
</tr>
</tbody>
</table>

... ordered...

<table>
<thead>
<tr>
<th>Key</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Key-999</td>
<td></td>
</tr>
<tr>
<td>Key-998</td>
<td></td>
</tr>
<tr>
<td>Key-997</td>
<td></td>
</tr>
</tbody>
</table>

... ordered...
Distributed Ordered Key Value

- **RegionServer**: Key-199, Value
  - Key-198, Value
  - Key-197, Value
  - ... (continues)

- **RegionServer**: Key-299, Value
  - Key-298, Value
  - Key-297, Value
  - ... (continues)

- **RegionServer**: Key-399, Value
  - Key-398, Value
  - Key-397, Value
  - ... (continues)

- **RegionServer**: Key-499, Value
  - Key-498, Value
  - Key-497, Value
  - ... (continues)

- **RegionServer**: Key-599, Value
  - Key-598, Value
  - Key-597, Value
  - ... (continues)

- **RegionServer**: Key-999, Value
  - Key-998, Value
  - Key-997, Value
  - ... (continues)
Table Row View

<table>
<thead>
<tr>
<th>Row Id</th>
<th>Column Id</th>
<th>Timestamp</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Key</td>
<td>Value</td>
<td></td>
<td></td>
</tr>
<tr>
<td>R11</td>
<td>col1</td>
<td>1234567</td>
<td>Value-A</td>
</tr>
<tr>
<td>R11</td>
<td>col2</td>
<td>1234567</td>
<td>Value-B</td>
</tr>
<tr>
<td>R11</td>
<td>col3</td>
<td>1234567</td>
<td>Value-C</td>
</tr>
<tr>
<td>R11</td>
<td>col4</td>
<td>1234567</td>
<td>Value-D</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>R11</th>
<th>Col1</th>
<th>Col2</th>
<th>Col3</th>
<th>Col4</th>
</tr>
</thead>
<tbody>
<tr>
<td>Value-A</td>
<td>Value-B</td>
<td>Value-C</td>
<td>Value-D</td>
<td></td>
</tr>
</tbody>
</table>
Versioning

Descending order
Column Family

<table>
<thead>
<tr>
<th></th>
<th>cf1</th>
<th>cf2</th>
</tr>
</thead>
<tbody>
<tr>
<td>R11</td>
<td>col1:1234567 Value-A</td>
<td>R11</td>
</tr>
</tbody>
</table>
ACIDity

• Atomic at row level

• Consistent to a point in time before the request

• Isolation through MVCC (reads) and row locks (mutations)

• Durability is guaranteed for all successful mutations
Namespace

Namespace: Blue

Namespace: Yellow

ACL

ACL

ACL

ACL

ACL

ACL

ACL
HBase Write

- HBase Region Server
- WAL
- File System

Write

1
HBase Write

1. Write to WAL
2. Memstore
3. HBase RS
4. File System
HBase Write

Write

HBase RS

Memstore

File System

WAL

Block

Block

Block

Data

Idx

Blm

Store files

© 2019 Bloomberg Finance L.P. All rights reserved.
HBase Write

HBase RS

Memstore

WAL

DFSClient

Store files

1

2

3

Write
HBase Write

```
HBase RS

Memstore

1

WAL
DFSClient

2

WAL

3

HDFS DataNode

Store files

HDFS DataNode
```
Compaction

File System

K-x  K-x  K-x  K-x  K-x  D-1

Store files
Compaction

File System

Store files

Compaction

File System

Store files
HBase Read

HBase RS

Memstore

Block Cache

Block

Read

1

WAL

File System

Store files

© 2019 Bloomberg Finance L.P. All rights reserved.
HBase Read

- HBase RS
  - Memstore
  - Block Cache
  - Block
  - WAL
- File System
  - Store files

1. File System
2. Block Cache
HDFS Read

HBase/DFSClient → TCP → HDFS

HDFS

File System

Data
HDFS Short-Circuit Read
Large Read Cache

HBase

Memstore

Block Cache

File System

WAL

Store files

Read

1

2
Large Read Cache
HBase Complete

HMaster

RS1  RS2  RS3  RS4  RS5  RS6
HBase Complete

- HMaster
- RS1
- RS2
- RS3
- RS4
- RS5
- RS6
HBase Complete
HBase Complete

HMaster

RS1
RS2
RS3
RS4
RS5
RS6
ZK
HBase Complete
Region Server Failure

HMaster

RS1
RS2
RS3
RS4
RS5
RS6

ZK

HBase Client

system
Region Server Failure
Region Replication

HMastr

RS1

RS2

RS3

RS4

RS5

RS6

HBase

Client

ZK

system

T1 r1

T1 r1
Region Replication

HMaster

RS1
RS2
RS3
RS4
RS5
RS6

ZK

HBase Client
Region Replication

https://www.youtube.com/watch?v=l6S-Vbs9WsU
Load Balancing

- HMaster
- RS1
- RS2
- RS3
- RS4
- RS5
- RS6
- ZK
- HBase Client

TechAtBloomberg.com

© 2019 Bloomberg Finance L.P. All rights reserved.
Load Balancing

HMaster
Balancer/AM

RS1
RS2
RS3
RS4
RS5
RS6

ZK
system

HBase Client

TechAtBloomberg.com
© 2019 Bloomberg Finance L.P. All rights reserved.
Balancer

- Region Count Cost
- Primary Region Count Cost
- Table Skew Cost
- Locality Cost
- Rack Locality Cost
- Region Replica Host Cost
- Region Replica Rack Cost
- Read Request Cost
- Write Request Cost
- Memstore Size Cost
- Storefile Size Cost
- Move Cost
Other Features

• HBase Replication

• HBase multi-tenancy support
  — https://www.youtube.com/watch?v=bZjz2G38Ju0

• HBase Co-processors and Filters
  — https://www.slideshare.net/Hadoop_Summit/hbase-coprocessors-uses-abuses-solutions
ZooKeeper Availability

• ZK Quorum

• One leader and remaining followers
  ― stat
  ― ruok
  ― mntr

• Test for availability
  ― e.g., List children of a znode
HBase Availability

• Master Availability - http://hmaster-node:16010/jmx
  — "name" : "Hadoop:service=HBase,name=Master,sub=Server","tag.isActiveMaster" : "true"

• Dead RegionServers
  — "name" : "Hadoop:service=HBase,name=Master,sub=Server", "numDeadRegionServers" : 0

• Region In Transition
  — "name" : "Hadoop:service=HBase,name=Master,sub=AssignmentManger","ritCount" : 0

• Test for availability
  — e.g., Query system table by listing tables
HDFS Availability

- **Namenode Availability** - [http://namenode-host:50070/jmx](http://namenode-host:50070/jmx)
  — "name" : "Hadoop:service=NameNode,name=FSNamesystem", "tag.HAState" : "active"

- **Dead Datanodes**
  — "name" : "Hadoop:service=NameNode,name=NameNodeInfo", "DeadNodes" : "{}"

- **Missing Blocks**
  — "name" : "Hadoop:service=NameNode,name=NameNodeInfo", "NumberOfMissingBlocks" : 0

- **Percentage Used**
  — "name" : "Hadoop:service=NameNode,name=NameNodeInfo", "PercentUsed" : 59

- **Under replicated blocks**
  — "name" : "Hadoop:service=NameNode,name=FSNamesystemState","UnderReplicatedBlocks":0

- **Test for availability**
  — e.g., Append data to a test file
HBase Performance

- RegionServer JMX metrics - [http://rs-node:60300/jmx](http://rs-node:60300/jmx)
  - "name" : "Hadoop:service=HBase,name=RegionServer,sub=Server"
    - Blockcache hit ratio
    - Request counts
    - Request response time
    - Compaction related metrics
    - Region count
    - Flush related metrics
    - Percentage of files local
    - Split related metrics
  - "name" : "Hadoop:service=HBase,name=RegionServer,sub=Tables",
    - Table level metrics

JVM

• GC – JMX Metrics
  — "name" : "java.lang:type=GarbageCollector,name=ParNew",
  — "name" : "java.lang:type=GarbageCollector,name=ConcurrentMarkSweep",

• GC Logging
  — -verbose:gc
  — -XX:+PrintHeapAtGC
  — -XX:+PrintGCDetails
  — -XX:+PrintGCTimeStamps
  — -XX:+PrintGCDateStamps
  — -XX:+PrintGCApplicationStoppedTime
  — -XX:+PrintClassHistogram
  — -XX:+PrintGCApplicationConcurrentTime
  — -XX:+PrintTenuringDistribution
  — -Xloggc:
OS/HW

- Memory
- CPU
- Disk
- Networking
Logs

- ZooKeeper Log

- HDFS
  - Namenode log
  - Datanode log

- HBase
  - Master log
  - RegionServer log

- OS
  - Syslog
Interacting with HBase

- HBase shell
  - DDL: create namespace/table, alter
  - Security: grant, revoke
  - DML: get, put, scan
  - Tools: assign, compact, balance
  - General: status

- HBase admin API

- HBase client API
Data Backup / Restore

• Snapshot
  — hbase shell > snapshot 'table', 'table_mmddyy'

• Restore from snapshot
  — hbase shell > restore_snapshot 'table_mmddyy'

• Export Snapshot
  — $ hbase org.apache.hadoop.hbase.snapshot.ExportSnapshot

• CopyTable
  — hbase org.apache.hadoop.hbase.mapreduce.CopyTable
Thank You!

Acknowledgement: Apache HBase Community
Reference: http://hbase.apache.org
Connect with Hadoop Team: hadoop@bloomberg.net
We are hiring!

https://www.bloomberg.com/careers

Questions?

TechAtBloomberg.com