hbase-2.0.0

Michael Stack
Release Manager
stack@apache.org

LONG TIME COMING

Not Yet Released!

- 2.0.0-alpha1 released June 8th, 2017
  - Preview Release, ‘rough cut’.
- Plan: Alphas => Betas => Final

EOY 2017

#GOALS

Goals: Compatibility

- Double-down on Semantic Versioning, semver
  - Adopted in hbase-1.0.0
  - MAJOR.MINOR.PATCH[-IDENTIFIER]
    - E.g. 2.0.0-alpha1

But….

- Semantic Versioning is about API only
    - ***What about….***
        - Internal/External Interfaces
            - Where is Client Interface when Spark/MapReduce

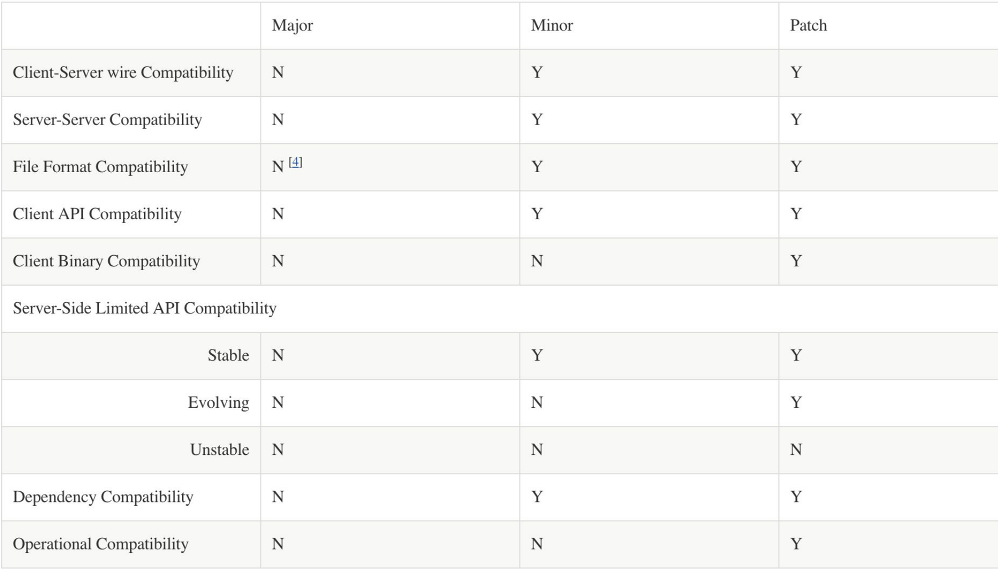
API Annotations

- From Hadoop…
  - `InterfaceAudience.Public`
    - Get/Put/Scan/Connection
  - `InterfaceAudience.LimitedPublic`
    - Coprocessors, Replication, etc.
  - `InterfaceAudience.Private`
    - Internal only

But (Again)...

- What about…
  - Source/Binary compatibility
  - Serializations
    - Wire
    - Formats in HDFS/Zookeeper
  - Dependencies
- See refguide semver section

Table 3. Compatibility Matrix [3]



	Major	Minor	Patch
Client-Server wire Compatibility	N	Y	Y
Server-Server Compatibility	N	Y	Y
File Format Compatibility	N [4]	Y	Y
Client API Compatibility	N	Y	Y
Client Binary Compatibility	N	N	Y
Server-Side Limited API Compatibility			
Stable	N	Y	Y
Evolving	N	N	Y
Unstable	N	N	N
Dependency Compatibility	N	Y	Y
Operational Compatibility	N	N	Y

But still….

- Grey areas…
    - Protobufs
        - Protobufs in API
        - Protobufs as Interface (c++/go clients)

Goal: Rolling Upgrade 1.x to 2.x

- **SemVer for DML at least in 2.x**
  - No admin of hbase-2.x with hbase-1.x client
  - Coprocessors will need to be upgraded
- hbase-1.x client can work against hbase-2.x cluster
  - Even 1.x Coprocessor Endpoints will work with an hbase-2.x cluster
- No Singularity! No downtime.

Other Goals

- Scale
  - More Regions, bigger clusters
- Performance
  - Inline read/write but also macro restart, assign, etc.
  - Better resource utilization
    - I/O, RAM
- Fix primary root of operational woes/bugs
  - Master Region Assignment
- Cleanup
  - Spark narrative
  - Interfaces

2.0.0

What’s inside hbase-2.0.0?

- Currently >4400 issues resolved
    - ~500 exclusive to 2.0.0

Inside: Prerequisites

- JDK8 only
- Hadoop-2.7.1 minimum
    - Will work against the coming Hadoop-3.x

Inside: Main Features

- New Master Core (A.K.A AMv2)
    - Prompt assign of millions of Regions, faster startup, larger scale!
    - Many small Regions instead of a few big ones
    - Promise: new degree of ***Resilience***
- Offheap Read/Write path
    - Less JVM
- Accordion
    - In-memory compaction
    - Less write amplification
- New Async Client
- etc.

Assignment Manager VERSI2N

- Assignment Manager v1 root of many operational headaches
- Redo based on custom “ProcedureV2”-based State Machine
  - Scale/Performance
  - All Master ops recast as Pv2 procedures
    - E.g. Move is a Procedure composed of an Unassign and Assign subprocedure...
- Operation aggregation
- Entity locking mechanism
  - Isolate operations on tables, regions, servers
  - Parallelization
- One *hbase:meta* writer, the Master only

Assignment Manager VERSI2N

- No more intermediate state in ZK
  - At other end of an RPC...
  - Record intent in local Master ops WAL, persisted up in HDFS
  - Only final state published to *hbase:meta*
- No more HBCK, no more RIT
  - No more distributed state
- Faster Assign
  - PE AMv2
- Standalone Testable

Offheap

- Smaller JVM heaps, less copying
  - More accounting!
- Offheap Read Path
  - L1 on Heap for Indexes & Blooms
  - L2 off Heap via BucketCache for Data
    - Big
  - Better latency
    - Cache more
    - Less GC, less erratic
- Offheap Write Path
  - RPC=>HDFS data kept offheap (+Async WAL client)
- Follow-ons:
  - BucketCache always-on

THE NIGHTMARE WHO COULD NOT DIE!
SKYWALD COMICS
THE HEAP
43704
NO.1
SEPT.
25¢
THE BEASTS OF BEDLAM BATTLE FOR A SLEEPING BEAUTY!
THE HEAP CONFRONTS MAN'S MOST GHOULISH FOE... SINISTER SCYTHE!

Accordion

- In-memory LSM
  - In-memory flush from ConcurrentSkipListMap to read-only pipeline of ‘segments’
    - Memory ‘breathes’ like an accordion’s bellows...
    - Optional in-memory compaction of Segments
      - Prune deletes/versions early
      - Benefit depends on # of versions
    - Less flushing, write amplification
- Flavors:
  - NONE
  - BASIC (Default)
  - ENHANCED
- Symbiosis w/ Offheaping
  - MemStore SLABs offheap
  - Compact in-memory layout (CellChunkMap)

Miscellaneous: Complete

- Medium Object Blobs (MOB)
  - Documents, Images, etc.
  - 100k => 10MB
- RegionServer Groups (rsgroups)
  - Coarse Isolation
  - Tables vs RegionServers
- WALs and HFiles in different Filesystems
  - AWS EMR HBase on S3 offering (Amazon S3 Storage Mode)
- New Netty Server/Client chassis
- Filesystem Quotas on Table/Namespace sizes
  - Per Table, per Namespace -- not per User
  - Violation Policy
  - Via Shell

Miscellaneous: In progress, **BLOCKERs**

- Spark module
  - Cleanup of our HBase Spark story
- AsyncDFSClient
  - Done but for the testing
- Update+Relocation of core dependencies
  - `hbase-thirdparty`
    - Guava 0.12 => 0.22
    - Protobuf 2.5 => 3.3
    - Netty
  - Etc.

Miscellaneous: In progress, **Nice-to-have**

- C++Client
- Backup/Restore
- Hybrid Logical Clock
    - Leap Seconds or Errant clock
    - Overload timestamp to do time and sequence id duty
    - Metadata only for 2.x

TODOs: **BLOCKERs**

- Edit of default configs
    - Stale
- API compatibility review
    - And that it wholesome!
- Testing
    - Each new feature/configuration
    - Rolling upgrade
        - Different data provenance/Starting point
- Operation & Scan Timeout Narrative Cleanup
- Sequence Identifier narrative
- Tie-it-off
    - We keep adding to 2.0.0… it'll never release.

Moving to 2.x

- API Cleanup
  - Interfaces are returned rather than Implementations
    - TableDescriptor instead of HTableDescriptor, etc.
  - Purged Protobuf and Guava classes from API
    - Purged from CLASSPATH too….(relocated)
- Evangelism:
  - Use ***shaded*** hbase-client and hbase-server instead going forward

hbase-3.0.0

- Split hbase:meta
- Refactor filesystem Interface and Implementation
  - Less dependency on HDFS
  - Scale
- HLC everywhere all the time on all tables
- Replication v2
- Dynamic Online Configuration

That’s it!

- Thank you to our host Huawei
- Running 2.0 Status: hbase-2.0.0 Status Doc
- References:
  - Matteo Bertozzi on hbase2: https://speakerdeck.com/matteobertozzi/hbase-2-dot-0-sf-meetup-oct-15
  - Enis Soztutar on hbase2: https://www.slideshare.net/enissoz/meet-hbase-20
- Images:
  - Accordion: https://static.roland.com/assets/images/products/gallery/fr-1x_top_open_gal.jpg
  - Version2: http://logofaves.com/wp-content/uploads/2009/06/version_m.jpg?9cf02b
  - Long Time: http://www.junodownload.com/products/one-night-only-long-time-coming/1957247-02/
  - Comic: http://4.bp.blogspot.com/-upwza0_ILn4/TmXB4IKkPKI/AAAAAAAAAHY/9lA7VYCmSkI/s1600/heap_0001.jpg
  - Stork: http://phicenter.org/wp-content/uploads/2012/08/stork_baby_delivery_720x540.jpg
  - Apache Helicopter: https://i.ytimg.com/vi/SQp-3fUBefA/maxresdefault.jpg
  - Goals: http://az616578.vo.msecnd.net/files/2016/02/29/635923548833557721545302495_GOALS600.png