Hadoop 2 : New and Noteworthy

Sujee Maniyam, ElephantScale

SNIA Legal Notice

- The material contained in this tutorial is copyrighted by the SNIA unless otherwise noted.
- Member companies and individual members may use this material in presentations and literature under the following conditions:
  - Any slide or slides used must be reproduced in their entirety without modification
  - The SNIA must be acknowledged as the source of any material used in the body of any document containing material from these presentations.
- This presentation is a project of the SNIA Education Committee.
- Neither the author nor the presenter is an attorney and nothing in this presentation is intended to be, or should be construed as legal advice or an opinion of counsel. If you need legal advice or a legal opinion please contact your attorney.
- The information presented herein represents the author's personal opinion and current understanding of the relevant issues involved. The author, the presenter, and the SNIA do not assume any responsibility or liability for damages arising out of any reliance on or use of this information.

  NO WARRANTIES, EXPRESS OR IMPLIED. USE AT YOUR OWN RISK.

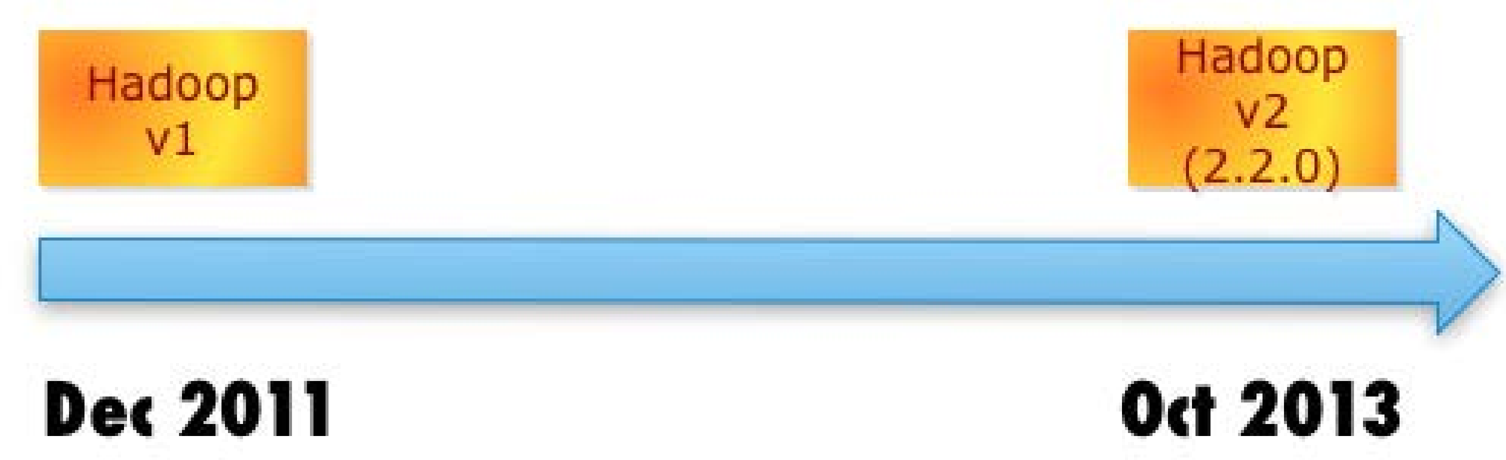
Abstract

- Hadoop 2 : New And Noteworthy Features
  - This session will appeal to Data Center Managers, Development Managers, and those that are looking for an overview of ‘whats new’ in Hadoop 2 platform. The session will highlight some of the notable features in Hadoop 2.

Quick Poll

- How many of you are NEW to Hadoop?
- How many of you are USING Hadoop?

Hadoop Timeline



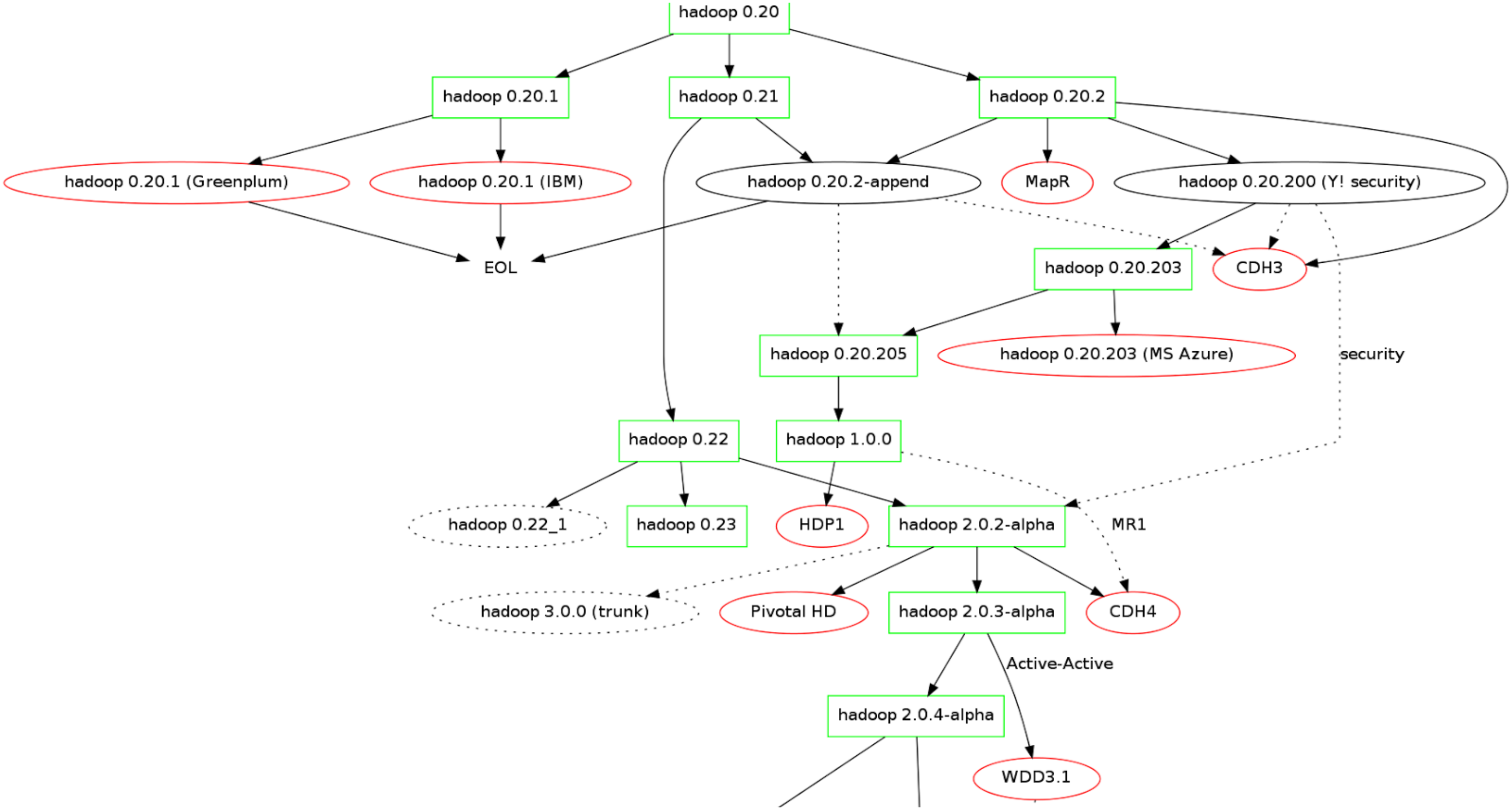
Hadoop v1

Hadoop v2 (2.2.0)

Dec 2011

Oct 2013

Hadoop Versions – ☺



Hadoop Versions – Simplified

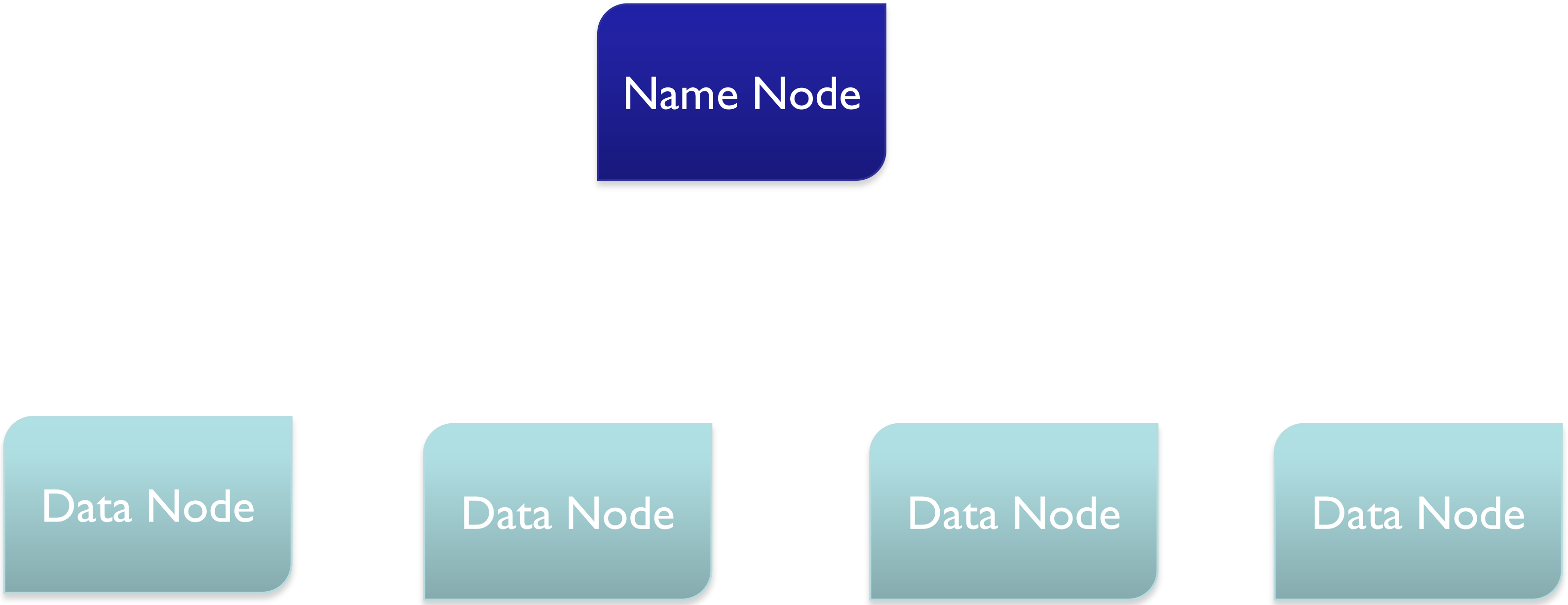
Hadoop 1	Hadooop 2
1.2.1 (aug 2013)	2.2.0 : (oct 2013)

Feature Matrix

Component	Feature	V1	v2
HDFS	NameNode High Availability		X
	Namenode federation		X
	Snapshots		X
	NFS v3 access to HDFS		X
	Improved IO		X
Processing	MapReduce v1	X	
	YARN (MapReduce v2)		X
Other	Kerberos security	X	X

Next : HDFS High Availability

HDFS Architecture (V1)



Name Node High Availability

- HDFS has (had) a ONE NameNode/ many Datanode design
- This leads to ‘Single Point of Failure’ (SPOF) for Name Node

Namenode Is Very Important In A Cluster

Is Hadoop NN Failure A Big Deal?

- At Yahoo study
  - 18 month study
  - 22 failure on 25 clusters
  - 0.58 failures per cluster per year
  - Only half of them would have benefited from HA
  - → 0.23 failure / year / cluster
- http://www.slideshare.net/Hadoop_Summit/hdfs-namenode-high-availability

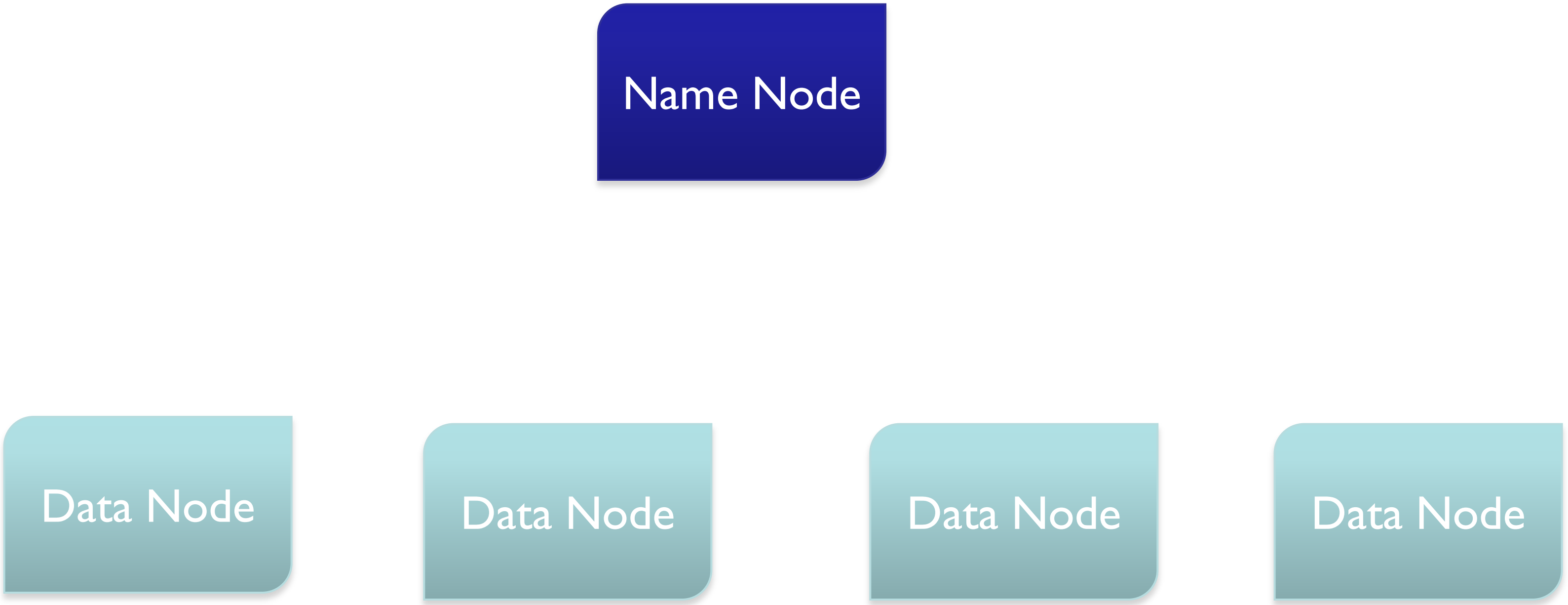
Still Needs To Be Fixed

- Downtime may be acceptable for batch workloads
- But not acceptable for running real time workloads like HBase that depend on HDFS
  - Downtime (even minutes) is not acceptable
- Make Hadoop more Enterprise friendly

How Do We Fix A Single Namenode Failure?

- Have two Namenodes !
- One ACTIVE and another PASSIVE
- When Active NN fails, Passive one will take over
- Fail over can be automated

HDFS Architecture (v1)



Name Node

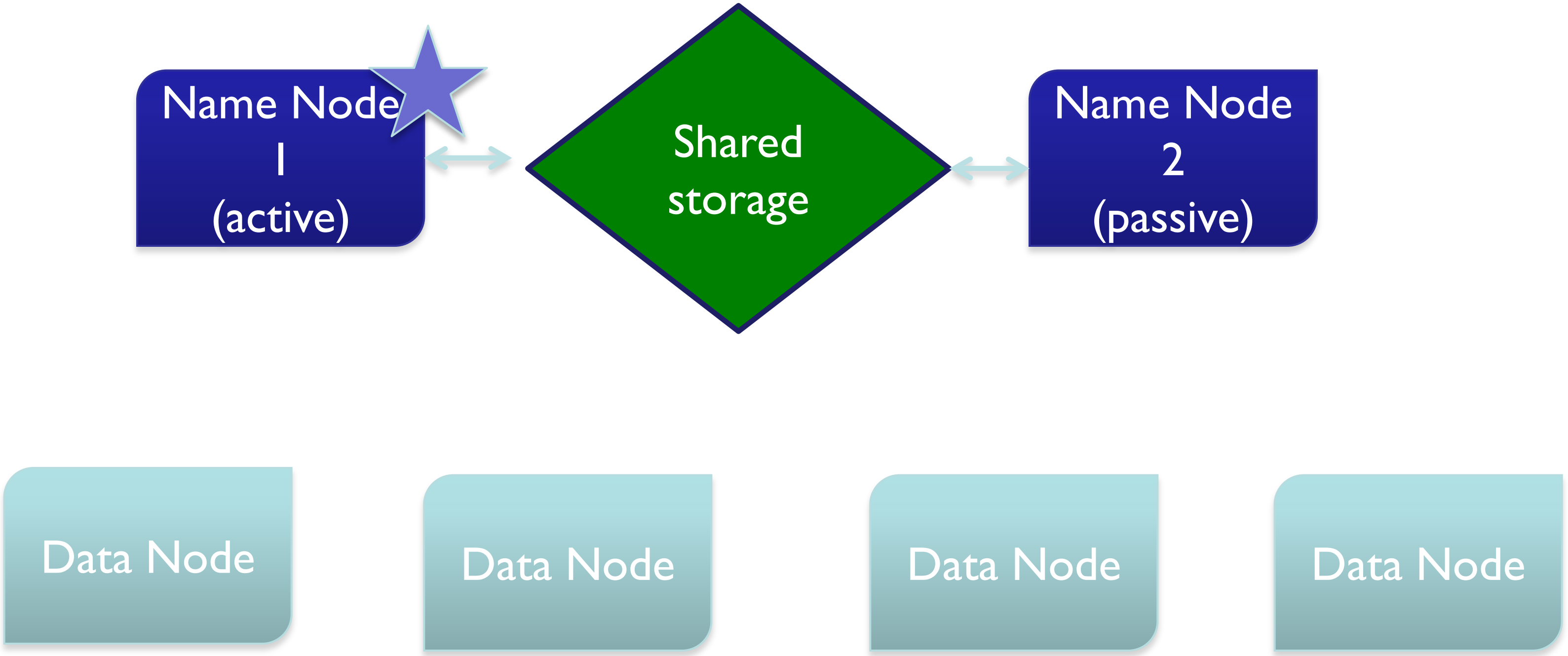
Data Node

Data Node

Data Node

Data Node

NameNode HA (V2)



Name Node 1 (active)

Shared storage

Name Node 2 (passive)

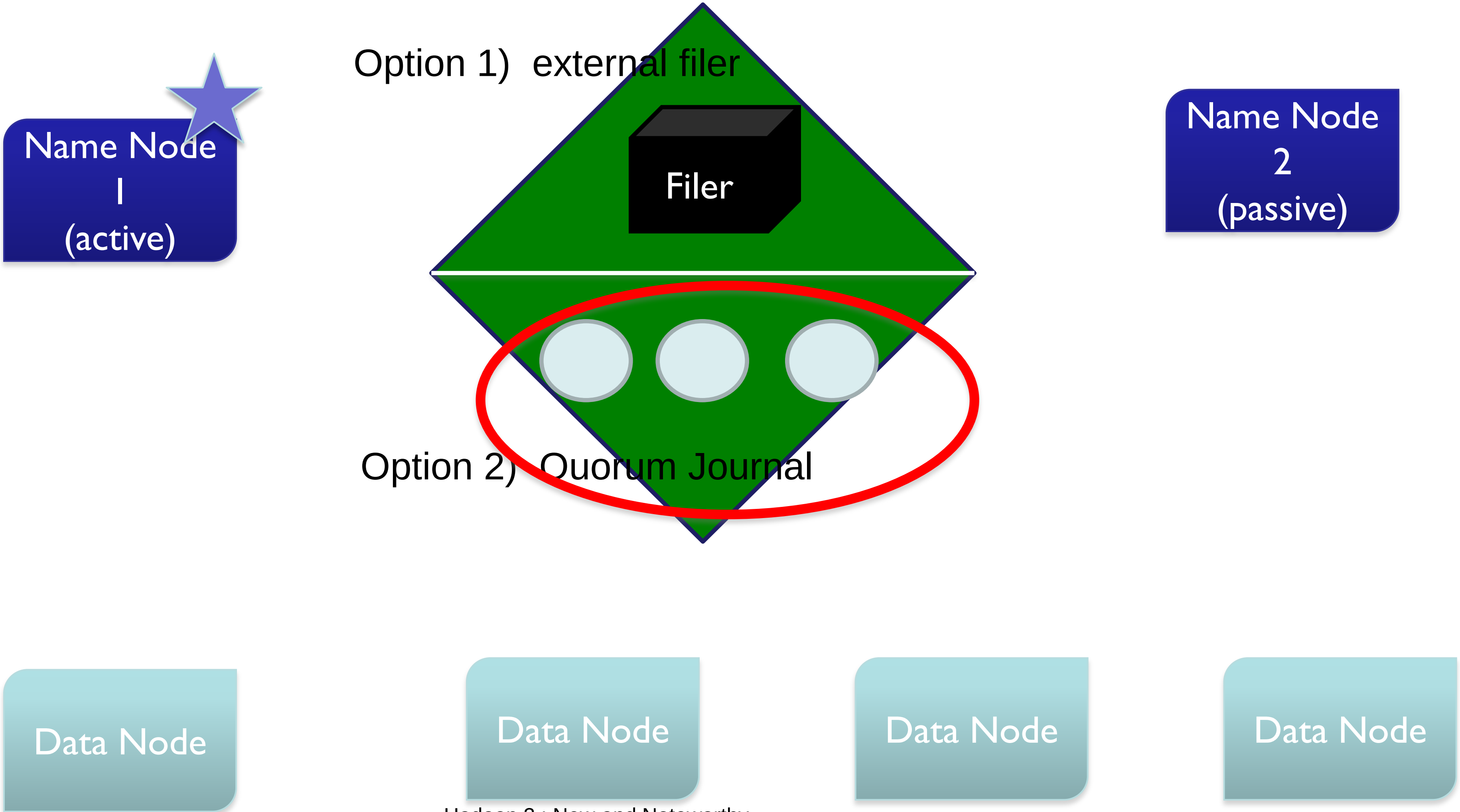
Data Node

Data Node

Data Node

Data Node

NameNode HA : Shared Storage



Namenode HA

- Namenode meta data is written to a shared storage (external filer or Quorum Journal Manager)
- Only ONE active NN can write to shared storage
- Passive NN reads and replays meta data from shared storage
- When Active NN fails, passive NN is promoted to active
  - Can be manual or automatic

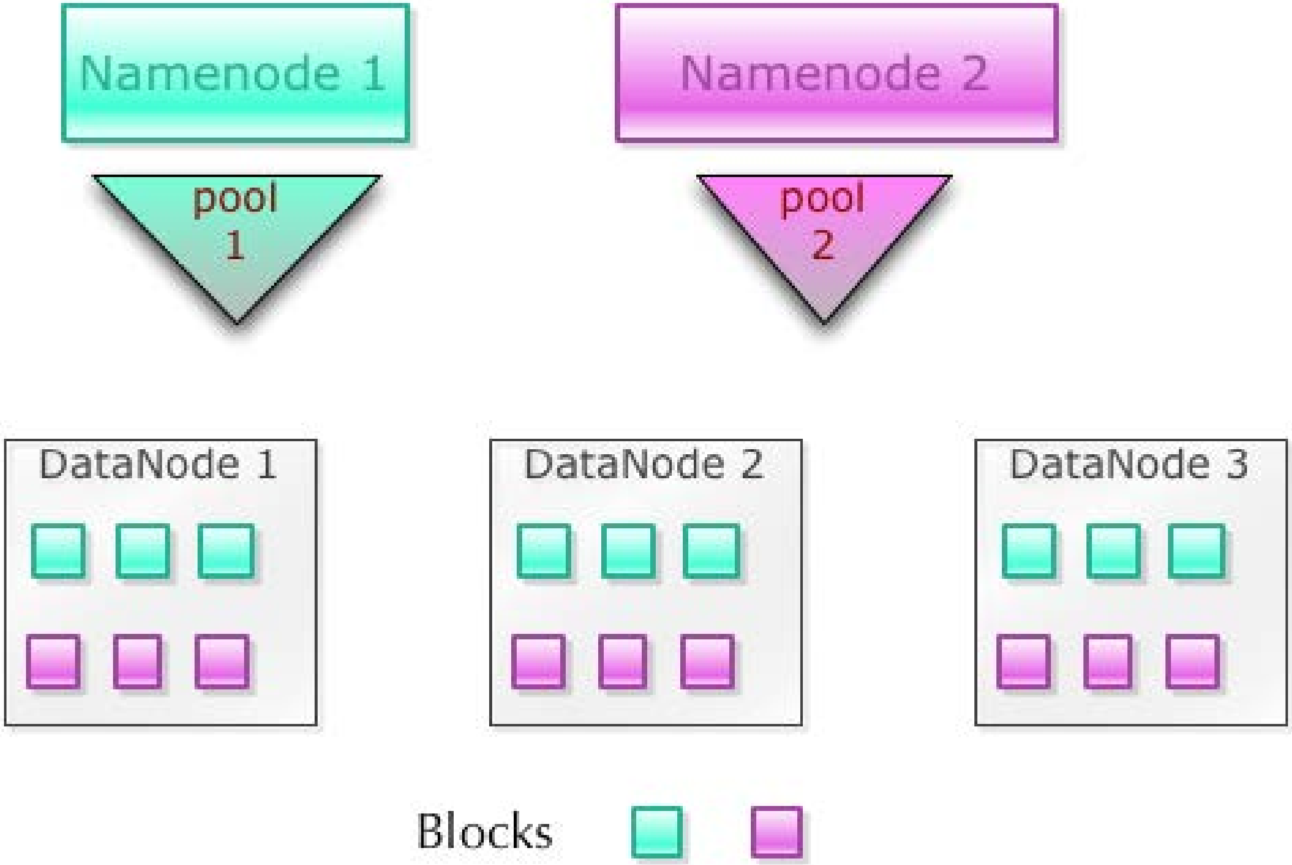
V2 Features

- HDFS
  - ~~Namenode HA~~
  - Namenode federation

Namenode Federation

- Namenode stores meta data in memory
- For large (very large) clusters, NN could exhaust memory
- Spread meta-data over mulitiple namenodes

HDFS Federation



HDFS Federation

- Now the namespace is divided
- /hbase → NN1
- /user → NN2
- /hive → NN3

HDFS Federation

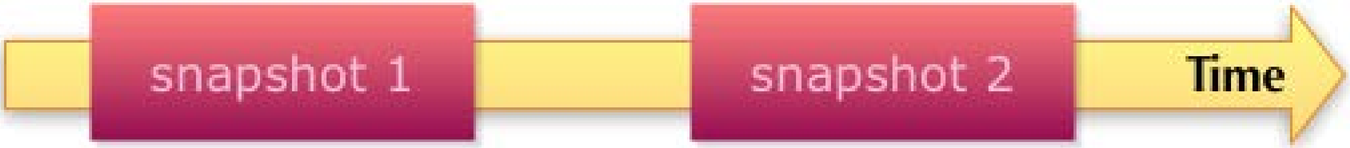
- Namespace is partitioned into ‘block pools’
- Datanodes are shared across cluster
  - They store blocks for different pools
- Datanodes send heart-beats to all NNs

V2 Features

- HDFS
  - ~~Namenode HA~~
  - ~~Namenode federati0n~~
  - Snapshots

HDFS Snapshots

- Wait, doesn’t HDFS makes replicas?
  - Yes
- But it doesn’t save you from :
  hdfs dfs –rm –r /data
- ‘Trash’ feature only works for CLI utilities
  - You can delete files using API.. Poof gone



HDFS Snapshots

- Recover from user errors, other disasters
- Peroidic snapshots
  - E.g : daily backups… keep them for 15 days
- Snapshotting is
  - Efficient (no data duplication, copy on write)
  - Fast
  - snapshot part of file system (not the whole thing)
- http://cdn.oreillystatic.com/en/assets/1/event/100/HDFS%20Snapshots%20and%20Beyond%20Presentation.pdf

V2 Features

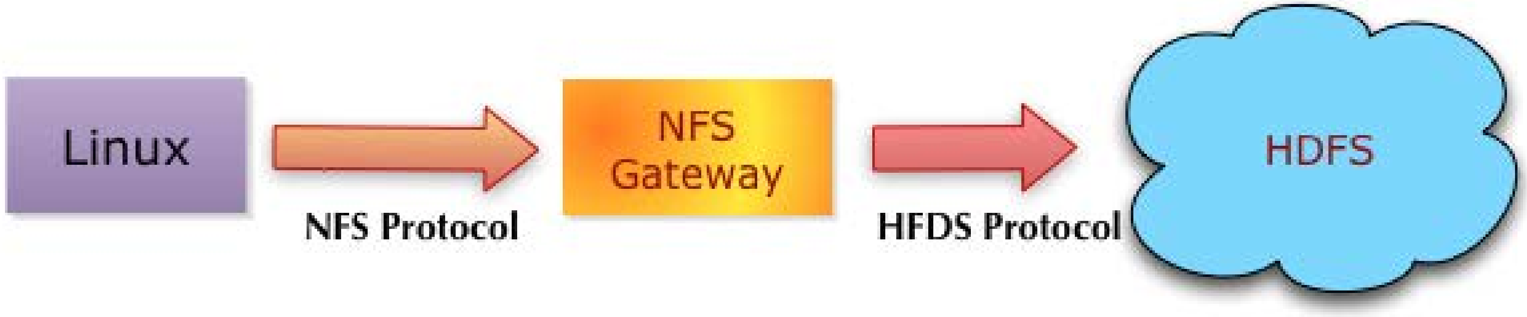
- HDFS
  - ~~Namenode HA~~
  - ~~Namenode federati0n~~
  - ~~Snapshots~~
  - NFSv3 access to HDFS

NFS Access to HDFS

- HDFS is a userland file system
  - Not a kernel file system
- So most linux programs can not read/write data to HDFS
  - We use ‘hdfs’ command line utils

NFS Access to HDFS

- HDFS supports NFS protocol starting with v2
- NFS is done via gateway machine



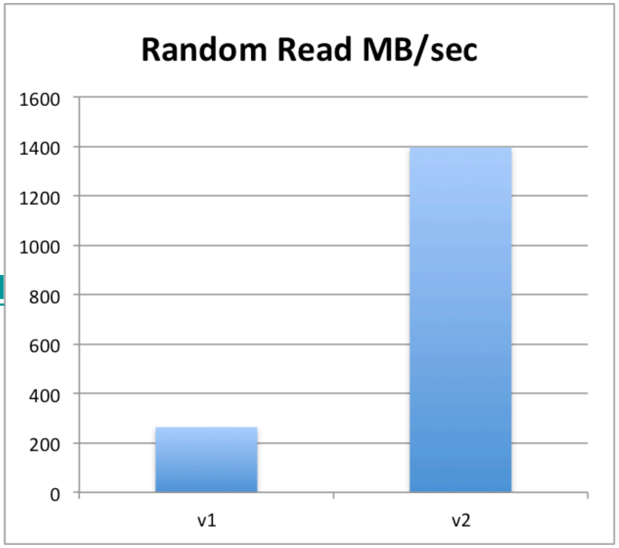
V2 Features

- HDFS
  - ~~Namenode HA~~
  - ~~Namenode federati0n~~
  - ~~Snapshots~~
  - ~~NFSv3 access to HDFS~~
  - Improved performance

HDFS Improved IO

- Lots of performance fixes from v1 → v2
- Quick comparison
  - Multi threaded random-read
  - HDFS v1 : 264 MB/sec
  - HDFS v2 : 1395 MB /sec (5x !)

Source : http://www.slideshare.net/cloudera/hdfs-u apache-hadoop-forum



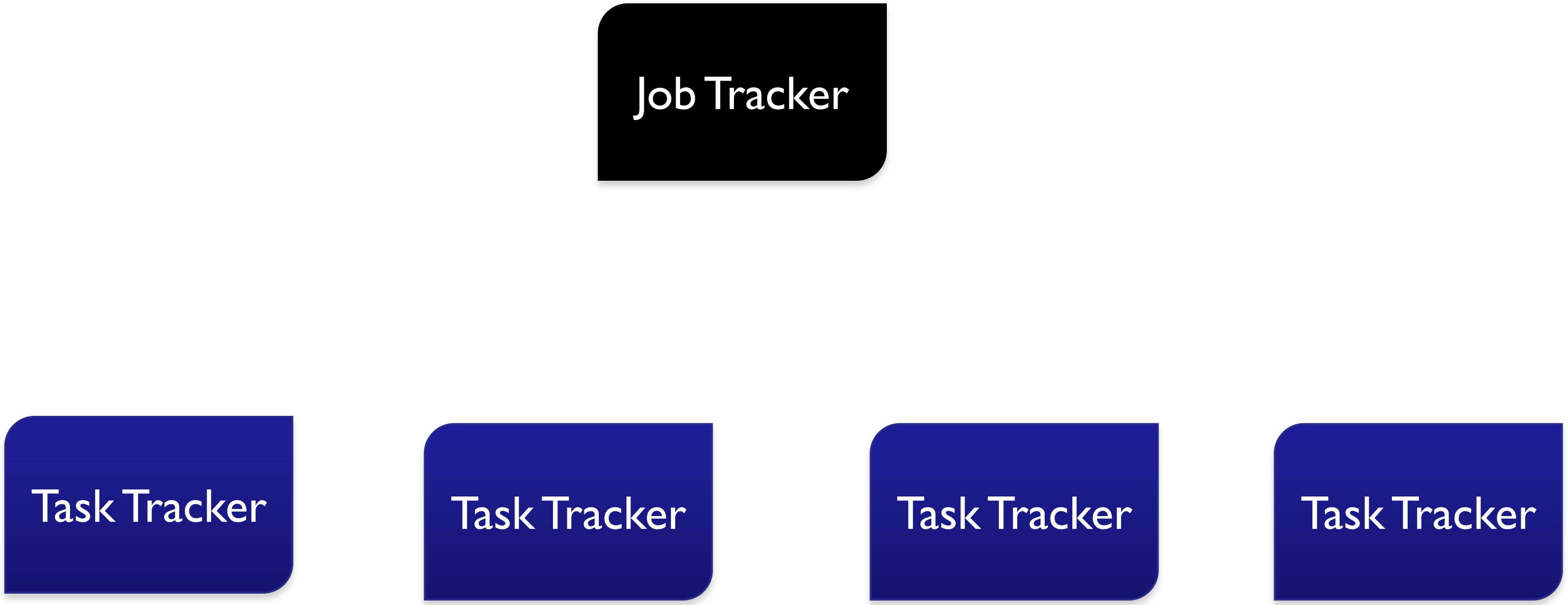
V2 Features

- ~~HDFS~~
- Processing
  - YARN

MapReduce V1

- MRV1 proved itself as a reliable batch processing framework!
- One Job Tracker (master) and many task tracker (workers)

MapReduce Architecture



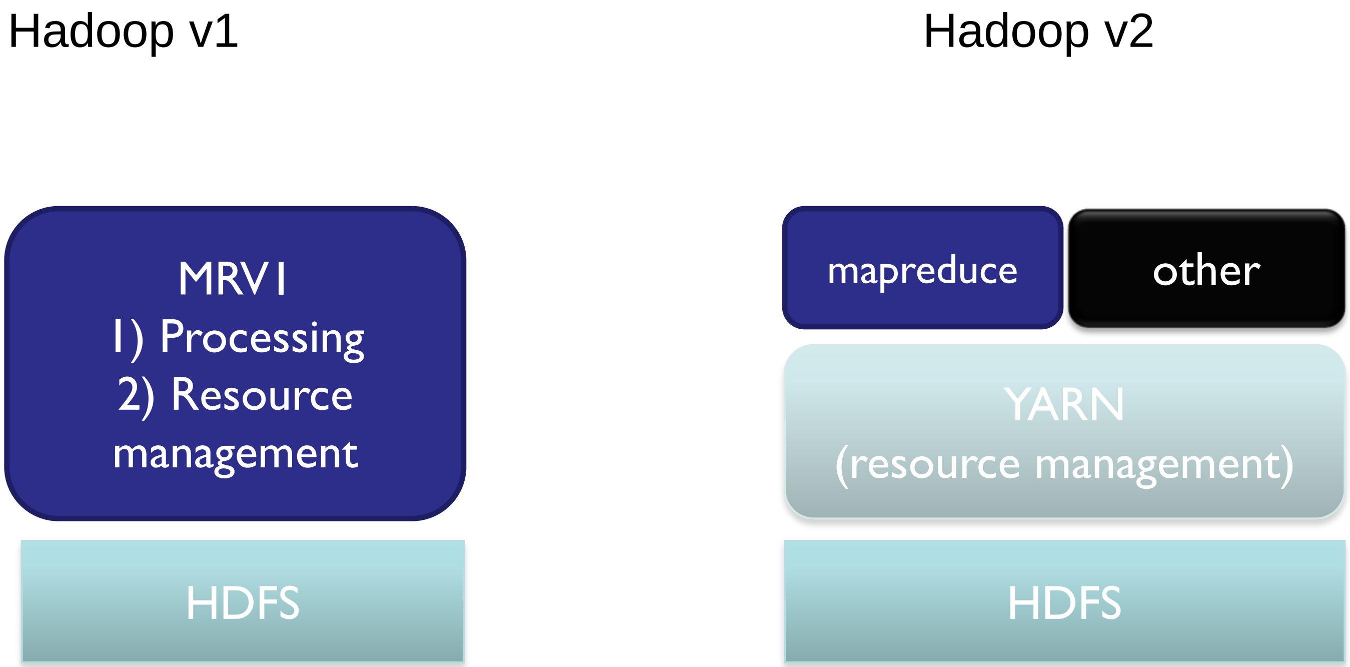
MRV1 Limitations

- Only supports one programming paradigm
  - Batch processing
- Alternate processing is hard to (or not possible) implement on top of MRV1
  - Real time processing
  - In-memory data

MRV1 Limitations

- Single Job Tracker (JT) → single point of failure
- JT Failure kills all running jobs (and queued jobs)
- JT started hit scalability limitations for very large clusters
  - 4,000 nodes

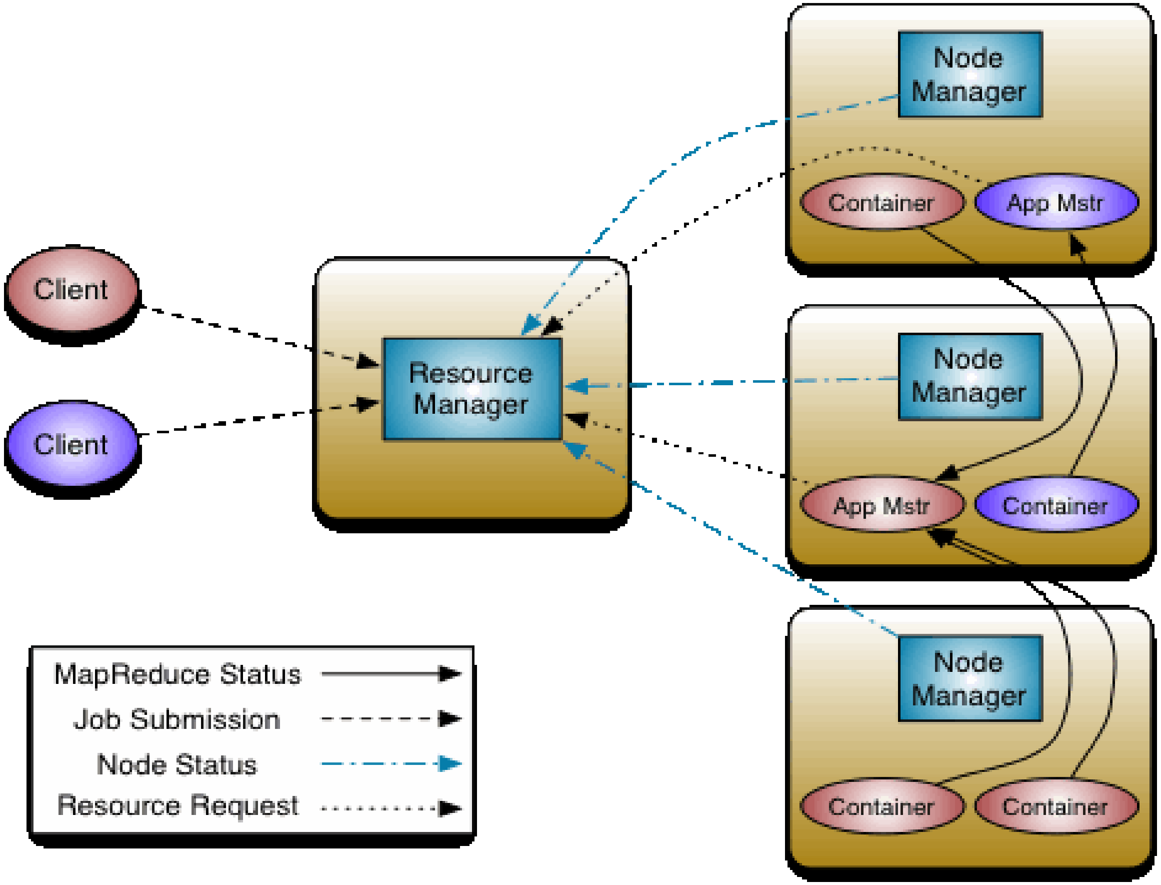
Looking Ahead



Yarn

- MRV1 did
  - Resource Management
  - And Processing
- Separate both out
- Yarn for resource management
- Mapreduce / other frameworks for processing
  - Now mapreduce is ‘just another app’

Yarn Architecture



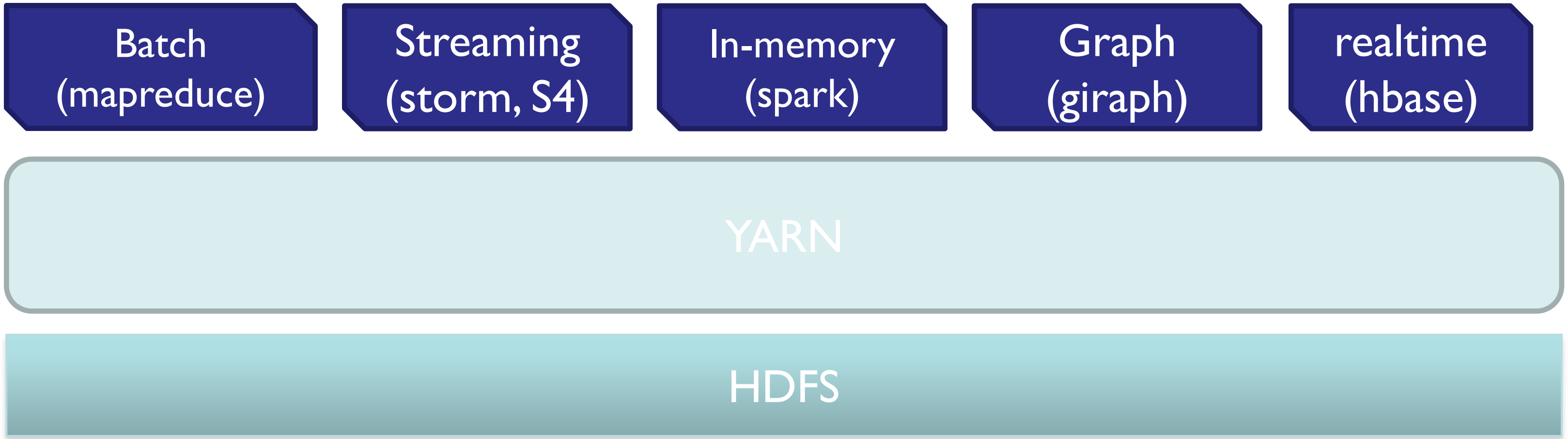
YARN Architecture

- resource manager : manages the resource for entire cluster
- node manager : manages resources a single node
- Containers : resource buckets (2 cpu + 8 G RAM)
- application masters : one for each application
  - batch mapreduce, storm …etc
  - Manages application scheduling and execution

Adoption of YARN

- Standard on Hadoop v2
- Already running at Yahoo at scale
- Lot of applications are already moving to YARN architecture

Apps on Yarn


Batch (mapreduce)
Streaming (storm, S4)
In-memory (spark)
Graph (giraph)
realtime (hbase)
YARN
HDFS

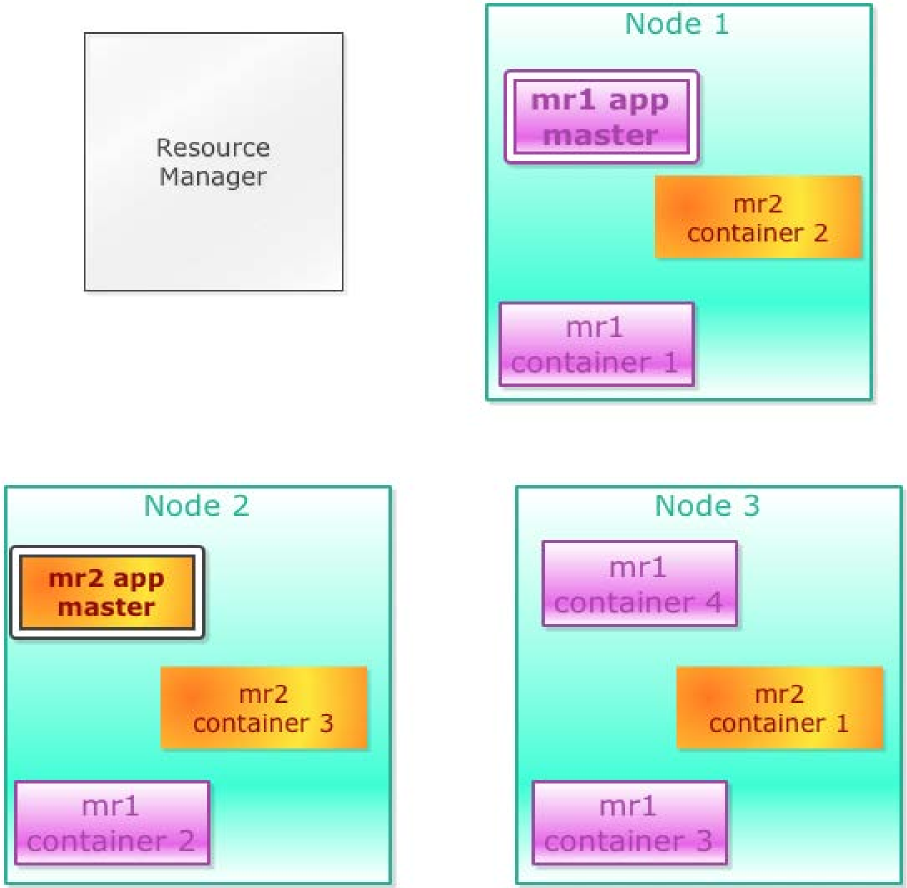
Apps on YARN

- Storm : real time event processing
- Giraph : graph processing (in memory)
- Spark : in-memory, iterative processing
- Hbase !

MapReduce on YARN

- MapReduce is NOT going anywhere
  - Works very well for batch processing
  - Proven
  - Lots of code out there
- No more single JobTracker
- Each MapReduce job runs an Application
- So failure one AppMaster only causes that job to fail
  - Other jobs are insulated

MapReduce on YARN



Writing A YARN Application

- http://hadoop.apache.org/docs/stable/hadoop-yarn/hadoop-yarn-site/WritingYarnApplications.html

V2 Features

- HDFS
  - ~~Namenode HA~~
  - ~~Namenode federati0n~~
- Processing
  - ~~YARN~~

So Which Hadoop Should I Use?

- Hadoop v1
  - Field-tested
  - Compatible with lots of other components
- Hadoop v2 – new, shiny

Hadoop Distributions

Distribution	Hadoop v1	Hadoop v2
Cloudera	CDH 3.x / CDH 4.x	CDH 5.x
Horton Works	HDP 1.x	HDP 2.x
Intel	Intel Hadoop	
Pivotal	HD	

Hadoop v2 + MRV1 ?

- You like to get all HDFS improvements
- But not ready to move from MRV1 to YARN yet…
- → Cloudera 4.x

Future…

- HDFS
  - Mirroring across data centers
  - Work well with SSD (solid state drives / flash drives)

If These Happen…

- I will be here to tell you about it ☺

Thanks & Questions?

Attribution & Feedback

The SNIA Education Committee thanks the following individuals for their contributions to this Tutorial.

Authorship History

Sujee Maniyam (Sept 2014)

Additional Contributors

Joseph White : Review & Feedback

Please send any questions or comments regarding this SNIA Tutorial to ***tracktutorials@snia.org***

Backup Slides