cloudera

cloudera

Hadoop and Cloudera

Managing Petabytes with Open Source

Jeff Hammerbacher
Chief Scientist and Vice President of Products, Cloudera
August 21, 2009

Why You Should Care

Quotes from HEP Researchers

- “The shift from dCache to Hadoop has been a pleasant transition”
  - “Easier management and much more stable performance”
- “We believe that [HDFS] is superior ... because of”
  - “Manageability”
  - “Reliability”
  - “Usability”
  - “Scalability”
- “Administration tools as well as performance particularly appreciated”
  - Alternatives: “maintenance and stability of code a big issue”

My Background

Thanks for Asking

- hammer@cloudera.com
- Studied Mathematics at Harvard
- Worked as a Quant on Wall Street
- Conceived, built, and led Data team at Facebook
  - Nearly 30 amazing engineers and data scientists
  - Several open source projects and research papers
- Founder of Cloudera
  - Vice President of Products and Chief Scientist (other titles)

Presentation Outline

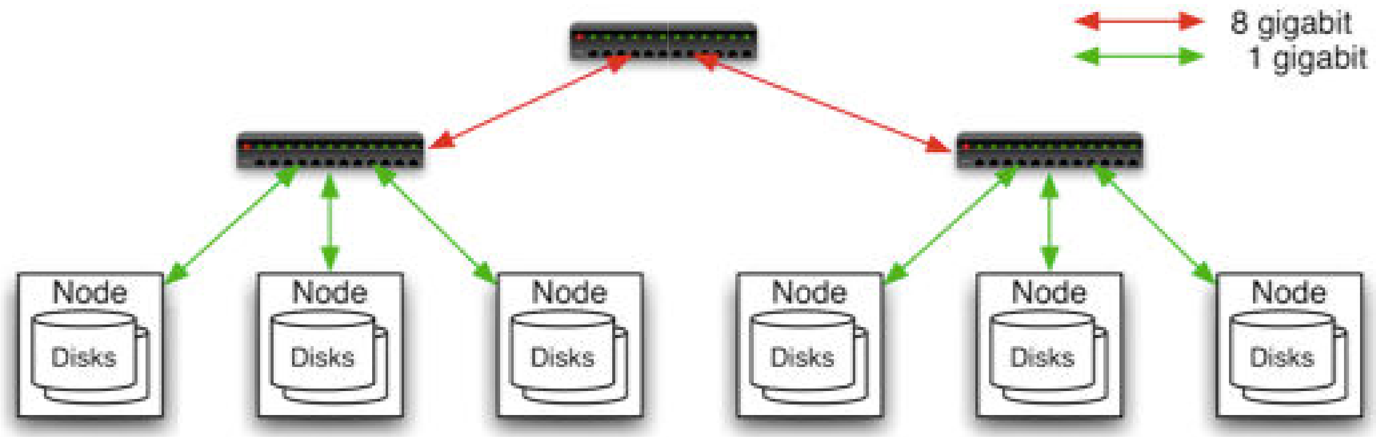
- What is Hadoop?
- Solving big data problems with Hadoop at Facebook and Yahoo!
  - Short history of Facebook's Data team
  - Hadoop applications at Yahoo!, Facebook, and Cloudera
- HDFS in more detail
  - Utilities and common problems
  - Future work
- Hadoop and HEP
- Questions and Discussion

What is Hadoop?

- Apache Software Foundation project, mostly written in Java
- Inspired by Google infrastructure
- Software for programming warehouse-scale computers (WSCs)
- Hundreds of production deployments
- Project structure
  - Hadoop Distributed File System (HDFS)
  - Hadoop MapReduce
  - Hadoop Common
  - Other subprojects
    - Avro, HBase, Hive, Pig, Zookeeper

Anatomy of a Hadoop Cluster

- Commodity servers
  - 1 RU, 2 x 4 core CPU, 8 GB RAM, 4 x 1 TB SATA, 2 x 1 gE NIC
- Typically arranged in 2 level architecture
  - 40 nodes per rack
- Inexpensive to acquire and maintain

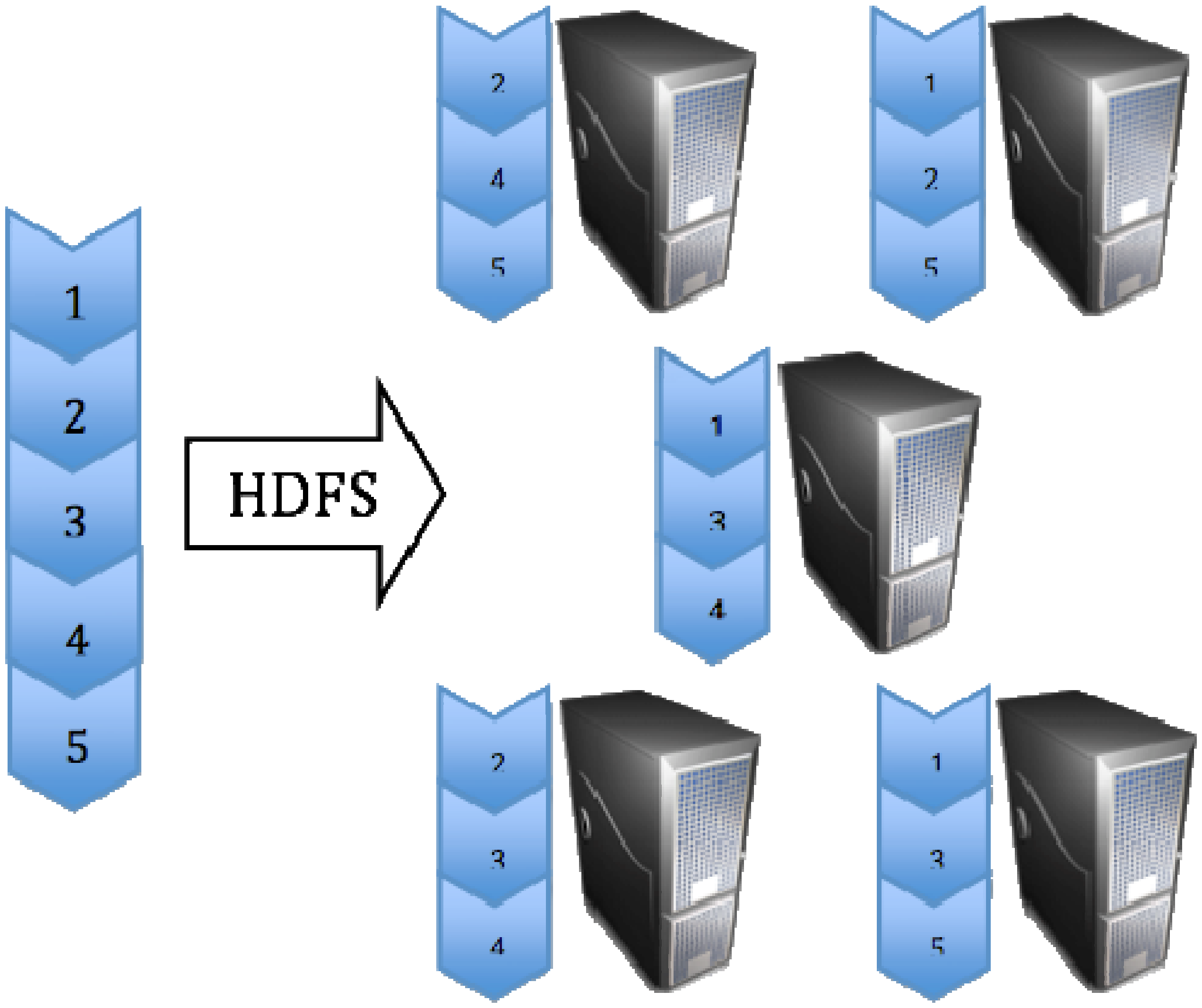


HDFS

- Pool commodity servers into a single hierarchical namespace
- Break files into 128 MB blocks and replicate blocks
- Designed for large files written once but read many times
  - Files are append-only
- Two major daemons: NameNode and DataNode
  - NameNode manages file system metadata
  - DataNode manages data using local filesystem
- HDFS manages checksumming, replication, and compression
- Throughput scales nearly linearly with node cluster size
- Access from Java, C, command line, FUSE, or Thrift

HDFS

HDFS distributes file blocks among servers

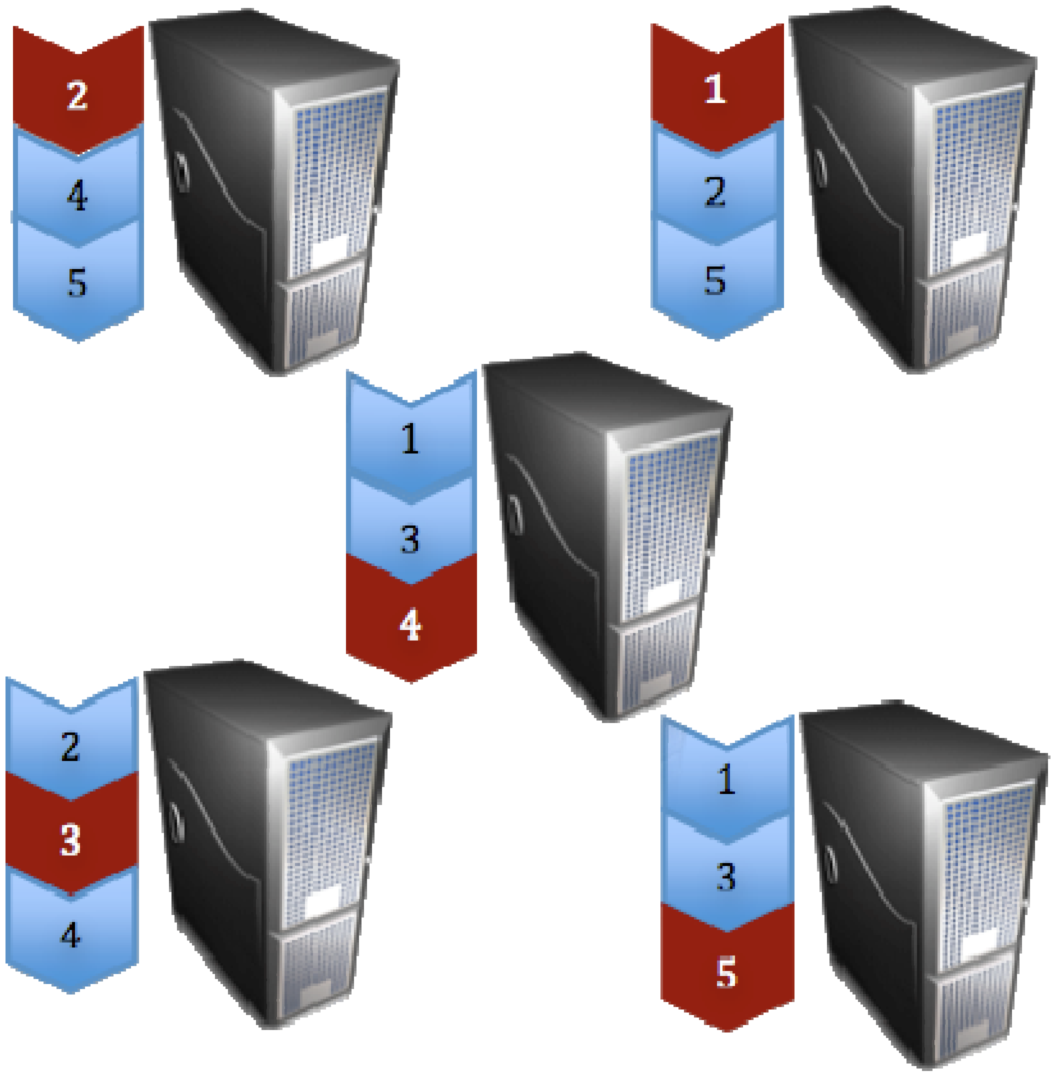


Hadoop MapReduce

- Fault tolerant execution layer and API for parallel data processing
- Can target multiple storage systems
- Key/value data model
- Two major daemons: JobTracker and TaskTracker
- Many client interfaces
  - Java
  - C++
  - Streaming
  - Pig
  - SQL (Hive)

MapReduce

MapReduce pushes work out to the data



Hadoop Subprojects

- Avro
  - Cross-language framework for RPC and serialization
- HBase
  - Table storage on top of HDFS, modeled after Google's BigTable
- Hive
  - SQL interface to structured data stored in HDFS
- Pig
  - Language for data flow programming; also Owl, Zebra, SQL
- Zookeeper
  - Coordination service for distributed systems

Hadoop Community Support

- 185 total contributors to the open source code base
  - Yahoo!, Facebook, and Cloudera are major contributors
- Over 750 (paid!) attendees at Hadoop Summit West
  - Expect similar numbers for upcoming Hadoop World NYC
- Three books (O’Reilly, Apress, Manning)
- Training videos free online
- Regular user group meetups in many cities
- University courses across the world
- Growing consultant and systems integrator expertise
- Commercial training, certification, and support from Cloudera

Hadoop Project Mechanics

- Trademark owned by ASF; Apache 2.0 license for code
- Rigorous unit, smoke, performance, and system tests
- Release cycle of 3 months (-ish)
  - Last major release: 0.20.0 on April 22, 2009
  - 0.21.0 will be last release before 1.0; feature freeze on 9/18
  - Subprojects on different release cycles
- Releases put to a vote according to Apache guidelines
- Releases made available as tarballs on Apache and mirrors
- Cloudera packages own release for many platforms
  - RPM and Debian packages; AMI for Amazon's EC2

Hadoop at Facebook

Early 2006: The First Research Scientist

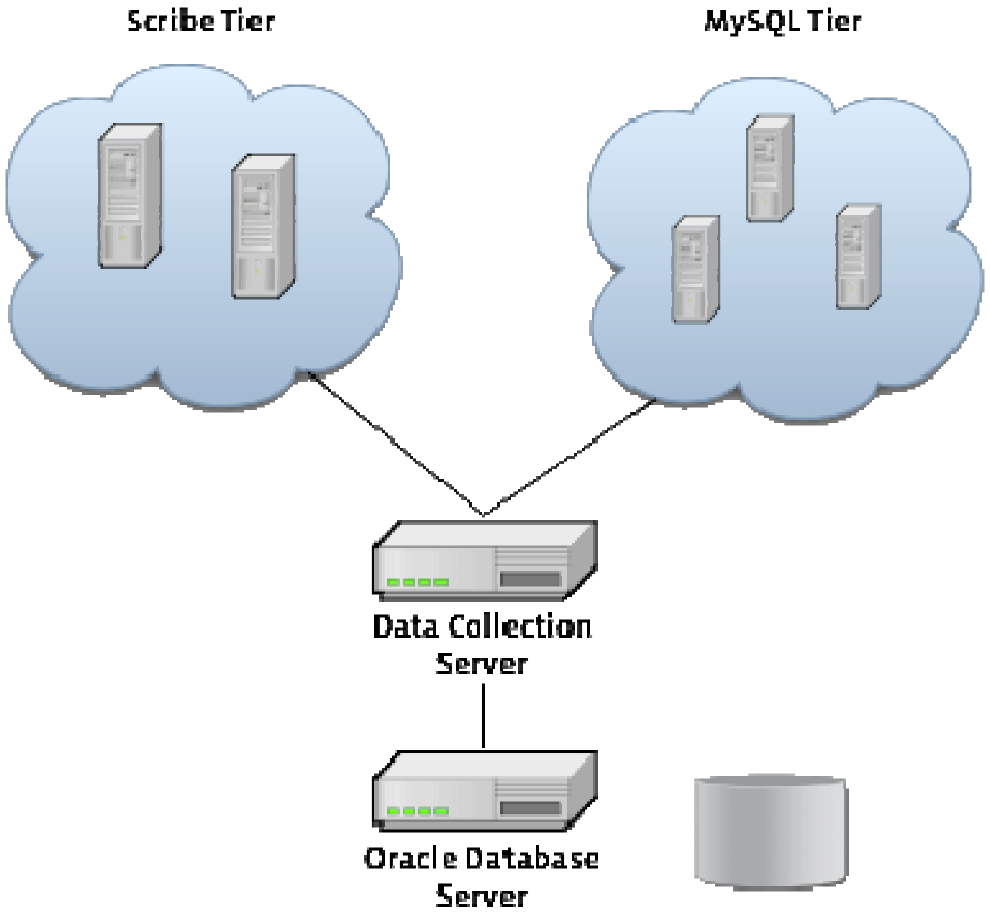
- Source data living on horizontally partitioned MySQL tier
- Intensive historical analysis difficult
- No way to assess impact of changes to the site

- First try: Python scripts pull data into MySQL
- Second try: Python scripts pull data into Oracle

- ...and then we turned on impression logging

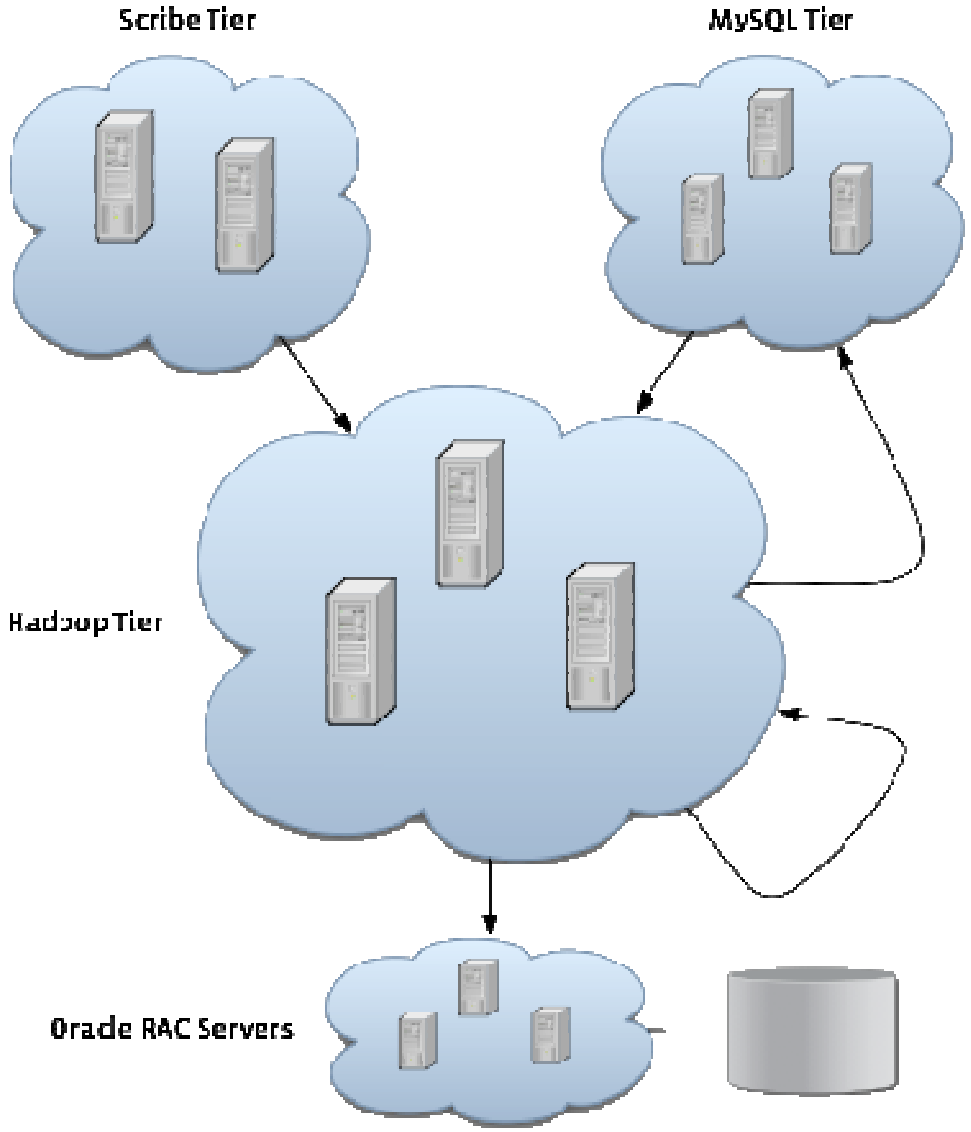
Facebook Data Infrastructure

2007



Facebook Data Infrastructure

2008



Major Data Team Workloads

- Data collection
  - server logs
  - application databases
  - web crawls
- Thousands of multi-stage processing pipelines
  - Summaries consumed by external users
  - Summaries for internal reporting
  - Ad optimization pipeline
  - Experimentation platform pipeline
- Ad hoc analyses

Workload Statistics

Facebook 2009

- 1,000 servers running Hadoop and Hive
- 2.4 PB of data (uncompressed)
- 15 TB added per day
- Over 200 users of the cluster, over half non-engineers

- Data from http://bit.ly/bKGcz and http://bit.ly/bKGcz

Hadoop at Yahoo!

- Jan 2006: Hired Doug Cutting
- Apr 2006: Sorted 1.9 TB on 188 nodes in 47 hours
- Apr 2008: Sorted 1 TB on 910 nodes in 209 seconds
- Aug 2008: Deployed 4,000 node Hadoop cluster
- May 2009: Sorted 1 TB on 1,460 nodes in 62 seconds
- Data Points
  - Over 25,000 nodes running Hadoop across 17 clusters
  - Hundreds of thousands of jobs per day
  - Typical HDFS cluster: 1,400 nodes, 2 PB capacity
  - Sorted 1 PB on 3,658 nodes in 16.25 hours

Example Hadoop Applications

- Yahoo!
  - Yahoo! Search Webmap
  - Content and ad targeting optimization
- Facebook
  - Fraud and abuse detection
  - Lexicon (text mining)
- Cloudera
  - Facial recognition for automatic tagging
  - Genome sequence analysis
  - Financial services, government, and of course: HEP!

Cluster Facilities and Hardware

- Data center: run Hadoop in a single data center, please
- Servers
  - Clusters are often either capacity bound or CPU bound
  - The 1U configuration specified previously is mostly standard
  - Many organizations now testing 2U, 12 drive configurations
  - Use ECC RAM and cheap hard drives: 7200 RPM SATA
  - Start with standard 64-bit box for masters and workers
- Network
  - Gigabit ethernet, 2 level tree, 5:1 oversubscription to core
  - May want redundancy at top of rack and core

System Software

- Operating system: Linux, CentOS mildly preferred
- Local file system
  - ext3 versus xfs
  - Mount with noatime for performance improvements
- RAID configuration: RAID0 versus JBOD
- Java 6, update 14 or later (compressed ordinary object pointers)
- Useful unix utilities
  - sar, iostat, iftop, vmstat, nfsstat, strace, dmesg, and friends
- Useful java utilities
  - jps, jstack, jconsole

HDFS

Operator Utilities

- Safe mode
- Filesystem check (fsck)
- dfsadmin
- Block scanner
- balancer
- archive
- distcp
- quotas: name space and disk space

HDFS

More Operator Utilities

- Users, groups, and permissions
- Audit logs
- Topology
- Web UIs
- Trash
- HDFS Proxy and Thriftfs
- Benchmarks and load testing
- Coming soon: symlinks

HDFS

Common Problems

- Disk capacity!
  - Especially due to log file sizes
  - Crank up dfs.datanode.du.reserved
- Slow, but not dead, disks
- Checkpointing and backing up metadata
- Losing a write pipeline for long-lived writes
- Upgrades
- Many small files

HDFS

Feature Requests from HEP

- Authentication
- High availability
- Better support for random reads
- Split events reasonably over the cluster
- Proxy layer to buffer random writes (since HDFS is append-only)

Hadoop and HEP

How Can We Help?

- HDFS is already storing over 700 TB of HEP data
  - 2 US CMS T2 sites
  - 2 US CMS T3 sites
  - 1 non-LHC grid site
- One site is requesting official US CMS approval to run only HDFS
- Experimentation at UK CMS site as well
- How to best integrate with existing tools?
  - dCache, ROOT, PROOF, Reflex, PanDA, etc.

cloudera