BIG DATA ANALYTICS WITH HADOOP

Philippe Julio

Open for Business...

WHO AM I

- Big Data / Analytics / BI & Cloud Solutions Specialist
- http://www.linkedin.com/in/JulioPhilippe
- Skills

BIG DATA MANAGEMENT INSIGHT

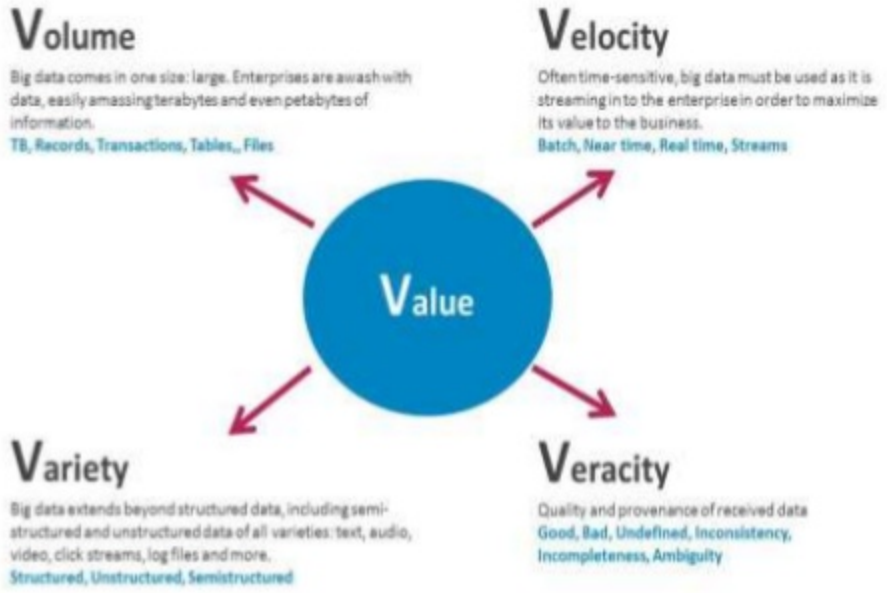
« Data don’t spring relevant, they become though ! »

DATA-DRIVEN ON-LINE WEBSITES

- To run the apps : messages, posts, blog entries, video clips, maps, web graph...
- To give the data context : friends networks, social networks, collaborative filtering...
- To keep the applications running : web logs, system logs, system metrics, database query logs...

BIG DATA – NOT ONLY DATA VOLUME

- Improve analytics and statistics models
- Extract business value by analyzing large volumes of multi-structured data from various sources such as databases, websites, blogs, social media, smart sensors...
- Have efficient architectures, massively parallel, highly scalable and available to handle very large data volumes up to several petabytes



Thematics

- Web Technologies
- Database Scale-out
- Relational Data Analytics
- Distributed Data Analytics
- Distributed File Systems
- Real Time Analytics

BIG DATA APPLICATIONS DOMAINS

- **Digital marketing optimization** (e.g., web analytics, attribution, golden path analysis)
- **Data exploration and discovery** (e.g., identifying new data-driven products, new markets)
- **Fraud detection and prevention** (e.g., revenue protection, site integrity & uptime)
- **Social network and relationship analysis** (e.g., influencer marketing, outsourcing, attrition prediction)
- **Machine-generated data analytics** (e.g., remote device insight, remote sensing, location-based intelligence)
- **Data retention** (e.g. long term retention of data, data archiving)

SOME BIG DATA USE CASES BY INDUSTRY

Energy

- Smart meter analytics
- Distribution load forecasting & scheduling
- Condition-based maintenance

Telecommunications

- Network performance
- New products & services creation
- Call Detail Records (CDRs) analysis
- Customer relationship management

Retail

- Dynamic price optimization
- Localized assortment
- Supply-chain management
- Customer relationship management

Manufacturing

- Supply chain management
- Customer Care Call Centers
- Preventive Maintenance and Repairs
- Customer relationship management

Banking

- Fraud detection
- Trade surveillance
- Compliance and regulatory
- Customer relationship management

Insurance

- Catastrophe modeling
- Claims fraud
- Reputation management
- Customer relationship management

Public

- Fraud detection
- Fighting criminality
- Threats detection
- Cyber security

Media

- Large-scale clickstream analytics
- Abuse and click-fraud prevention
- Social graph analysis and profile segmentation
- Campaign management and loyalty programs

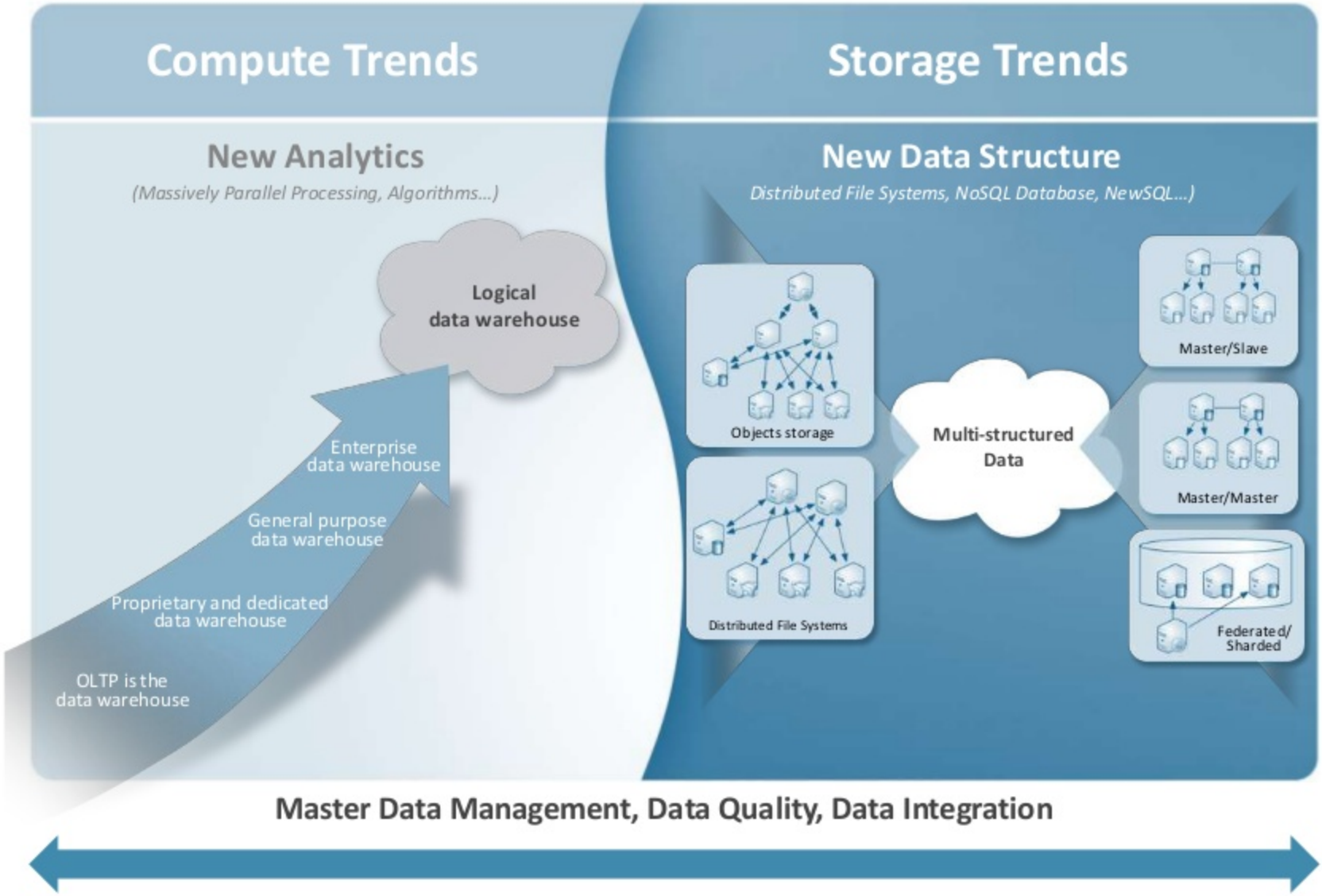
Healthcare

- Clinical trials data analysis
- Patient care quality and program analysis
- Supply chain management
- Drug discovery and development analysis

TOP 10 BIG DATA SOURCES

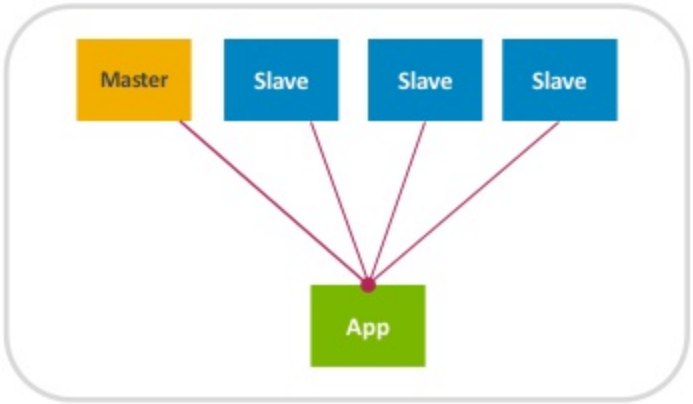
1. Social network profiles
2. Social influencers
3. Activity-generated data
4. SaaS & Cloud Apps
5. Public web information
6. MapReduce results
7. Data Warehouse appliances
8. NoSQL databases
9. Network and in-stream monitoring technologies
10. Legacy documents

NEW DATA AND MANAGEMENT ECONOMICS



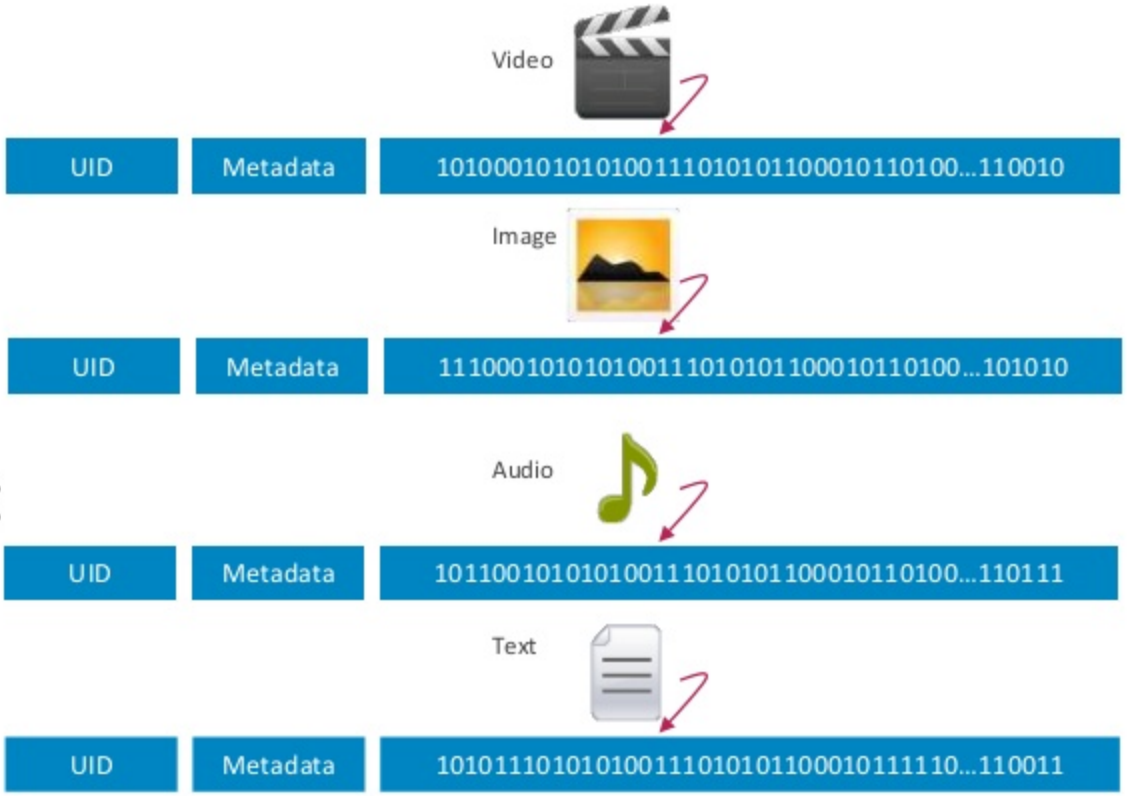
DISTRIBUTED FILE SYSTEMS

- System that permanently store data
- Divided into logical units (files, shards, chunks, blocks…)
- A file path joins file and directory names into a relative or absolute address to identify a file
- Support access to file and remote servers
- Support concurrency
- Support distribution
- Support replication
- NFS, GPFS, Hadoop DFS, GlusterFS, MogileFS, MooseFS….



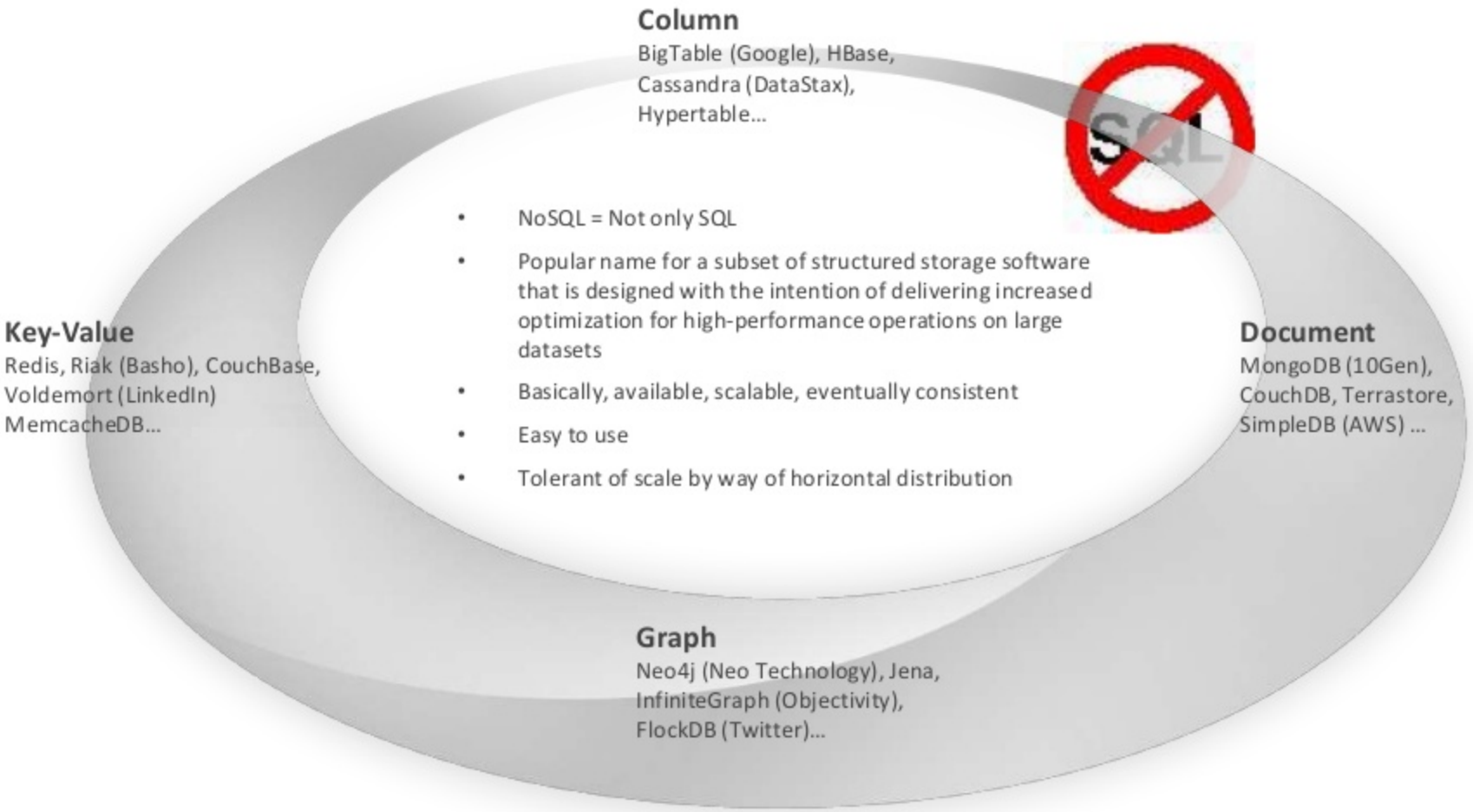
UNSTRUCTURED DATA AND OBJECT STORAGE

- Metadata values are specific to each individual type
- Enables automated management of content
- Ensure integrity, retention and authenticity



Unstructured Data + Metadata = Object Storage

NOSQL DATABASES CATEGORIES



WHAT IS HADOOP ?

"Flexible and available architecture for large scale computation and data processing on a network of commodity hardware"

Open Source Software + Hardware Commodity
= IT Costs Reduction

WHAT IS HADOOP USED FOR ?

- Searching
- Log processing
- Recommendation systems
- Analytics
- Video and Image analysis
- Data Retention

WHO USED HADOOP ?

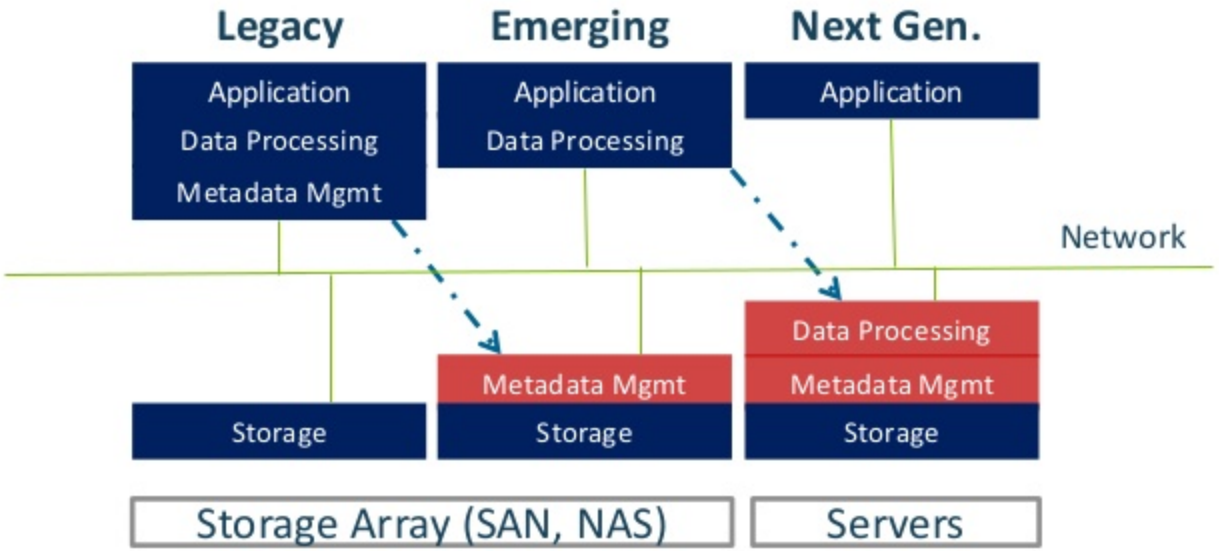
- Top level Apache Foundation project
- Large, active user base, mailing lists, user groups
- Very active development, strong development team
- http://wiki.apache.org/hadoop/PoweredBy#L

MOVING COMPUTATION TO STORAGE

General Purpose Storage Servers

- Combine server with disks & networking for reducing latency
- Specialized software enables general purpose systems designs to provide high performance data services

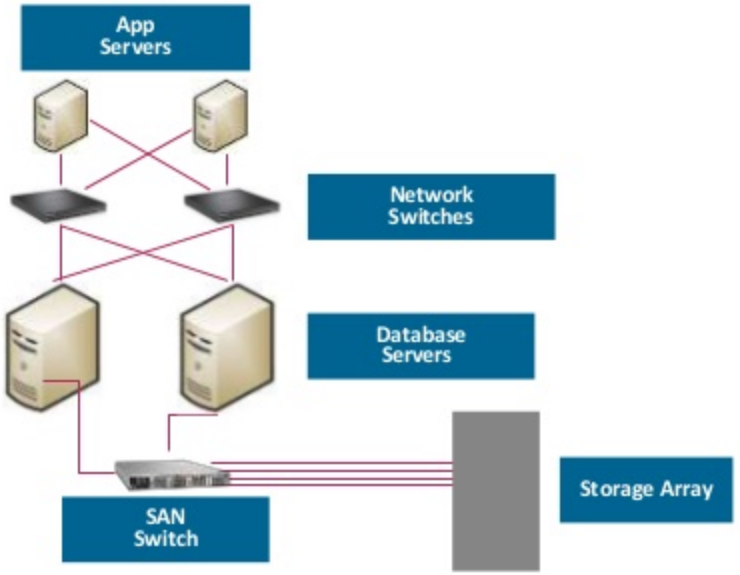
Moving Data processing to Storage



BIG DATA ARCHITECTURE

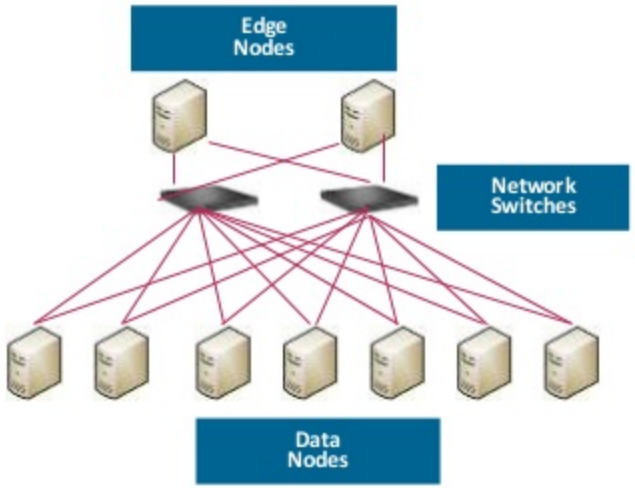
BI & DWH Architecture - Conventional

- SQL based
- High availability
- Enterprise database
- Right design for structured data
- Current storage hardware (SAN, NAS, DAS)

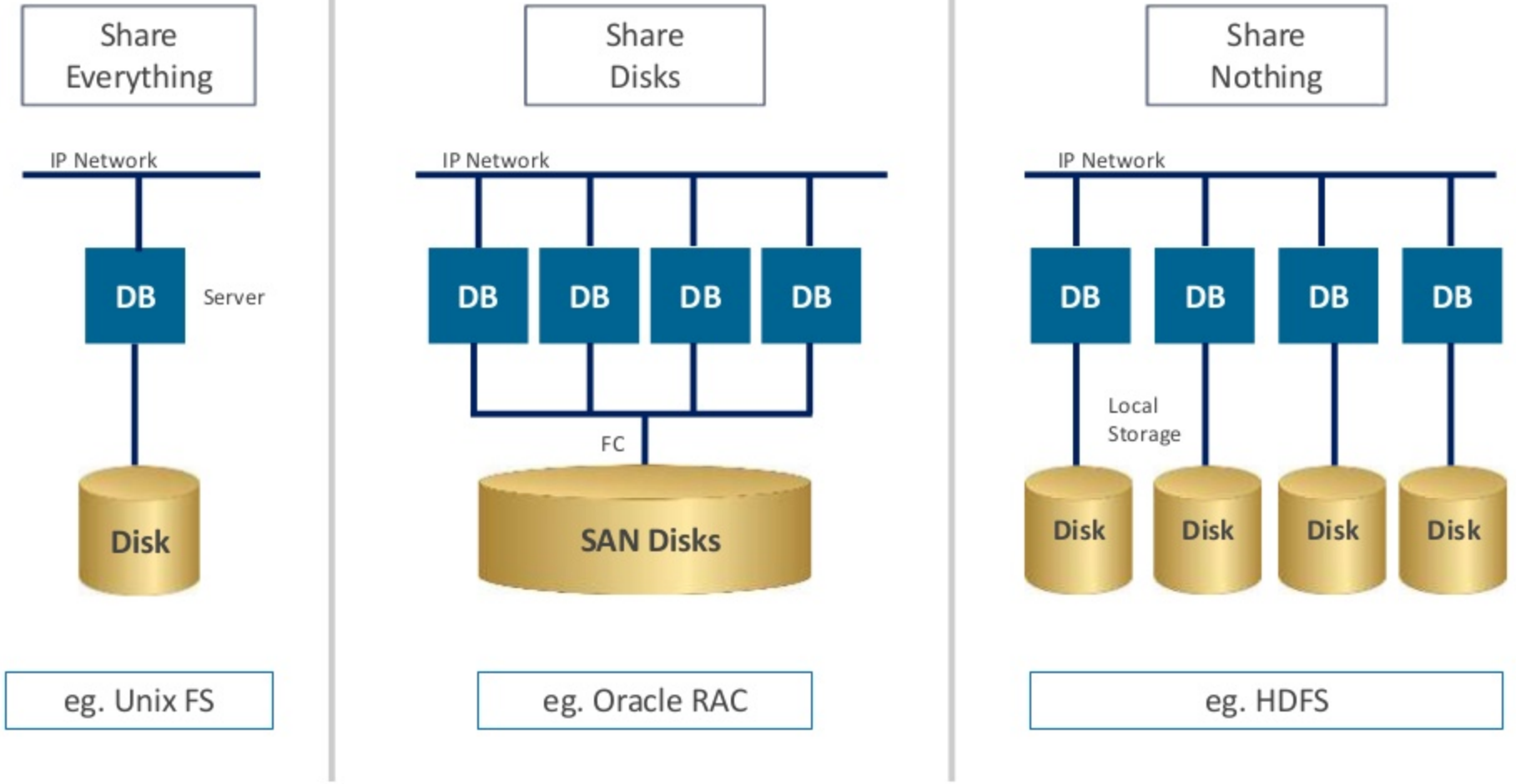


Analytics Architecture – Next Generation

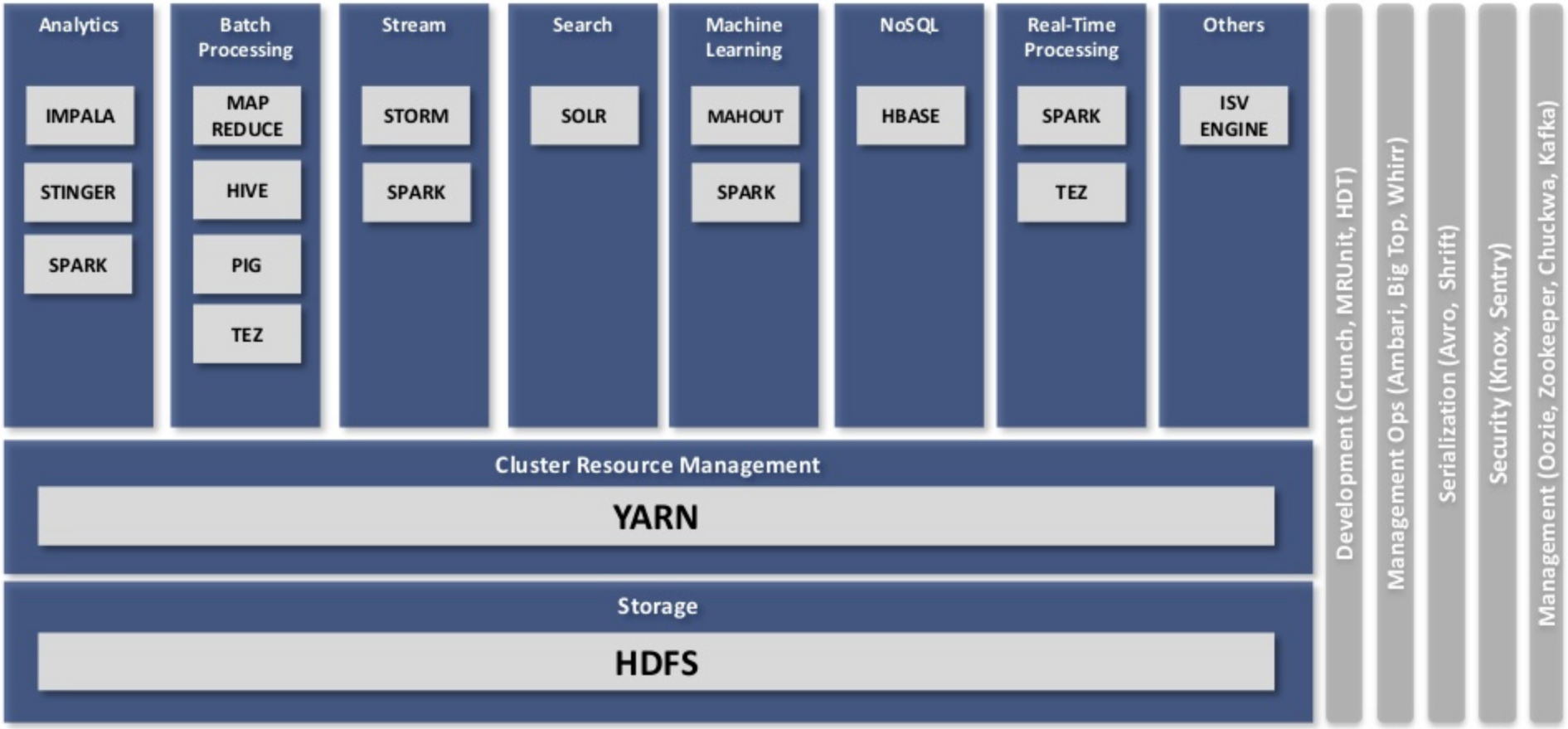
- Not only SQL based
- High scalability, availability and flexibility
- Compute and storage in the same box for reducing the network latency
- Right design for semi-structured and unstructured data



SHARE NOTHING ARCHITECTURE



APACHE HADOOP 2.0 ECOSYSTEM



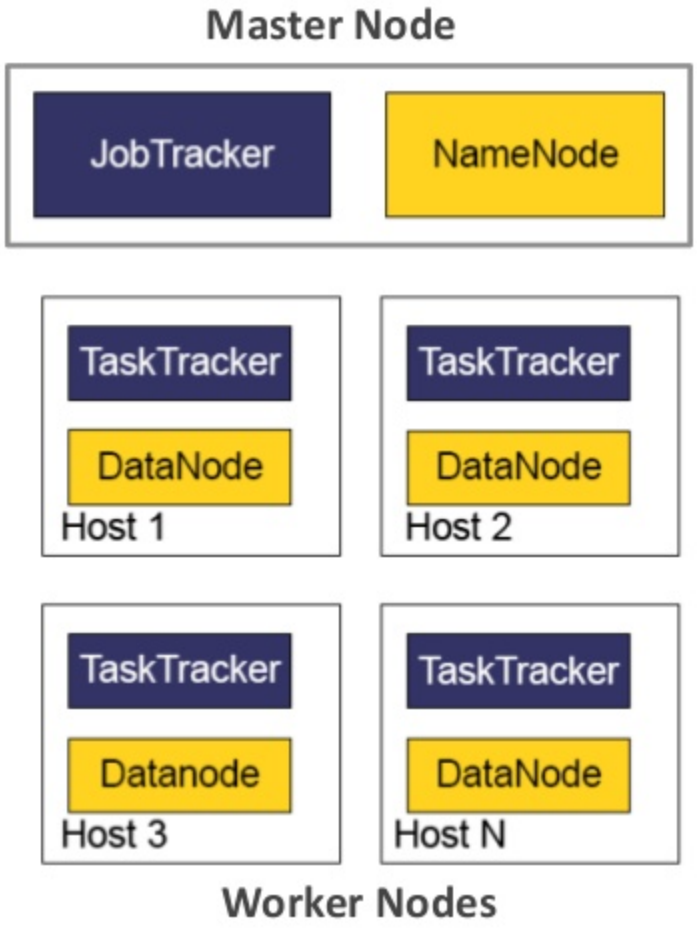
http://incubator.apache.org/projects/

HADOOP COMMON

- Hadoop Common is a set of utilities that support the Hadoop subprojects.
- Hadoop Common includes Filesystem, RPC, and Serialization libraries.

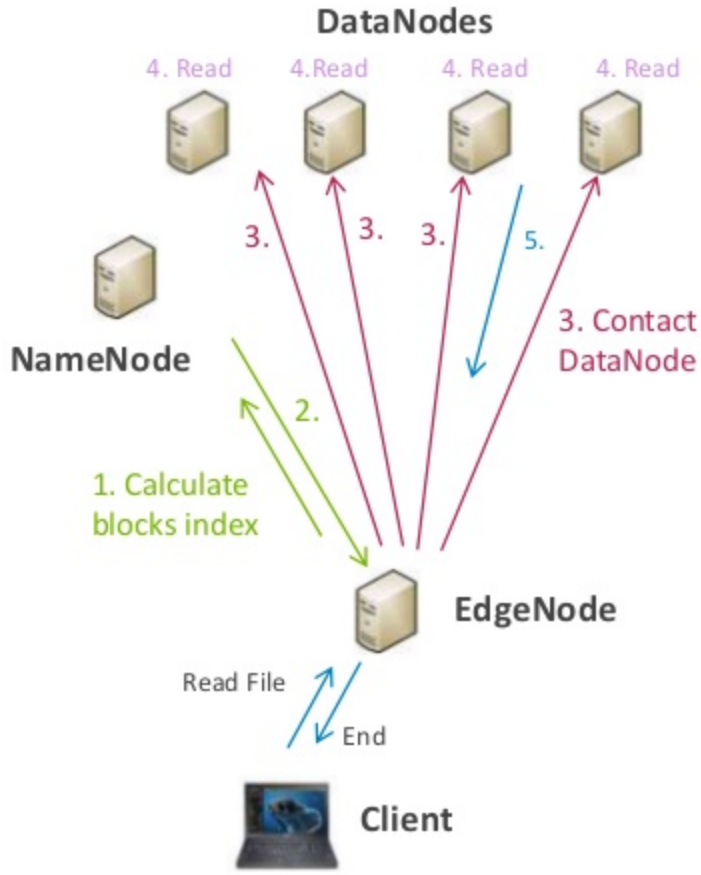
HDFS & MAPREDUCE

- **Hadoop Distributed File System**
  - A scalable, Fault tolerant, High performance distributed file system
  - Asynchronous replication
  - Write-once and read-many (WORM)
  - Hadoop cluster with 3 DataNodes minimum
  - Data divided into 64MB (default) or 128MB blocks, each block replicated 3 times (default)
  - No RAID required for DataNode
  - Interfaces: Java, Thrift, C Library, FUSE, WebDAV, HTTP, FTP
  - **NameNode** holds filesystem metadata
  - Files are broken up and spread over the **DataNodes**
- **Hadoop Map Reduce**
  - Software framework for distributed computation
  - Input | Map() | Copy/Sort | Reduce() | Output
  - **JobTracker** schedules and manages jobs
  - **TaskTracker** executes individual map() and reduce() tasks on each cluster node



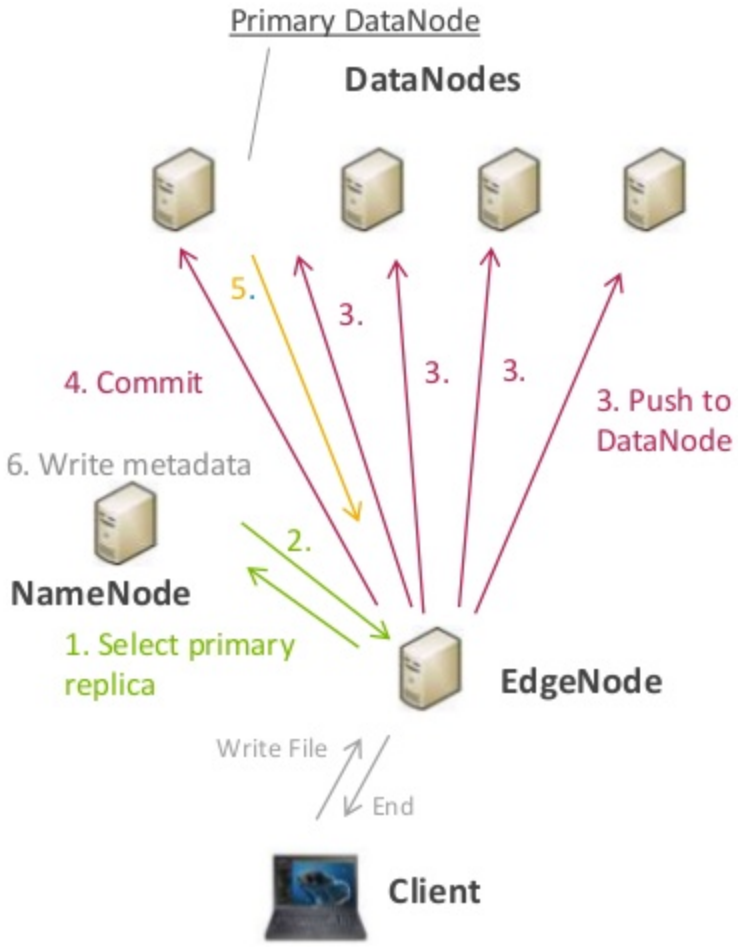
HDFS - READ FILE

1. The client API calculates the blocks index based on the offset of the file pointer and make a request to the NameNode
2. The NameNode replies which DataNodes has a copy of that block
3. The client contacts the DataNodes directly without going through the NameNode
4. The DataNodes read the blocks
5. The DataNodes response to the client about the success



HDFS - WRITE FILE

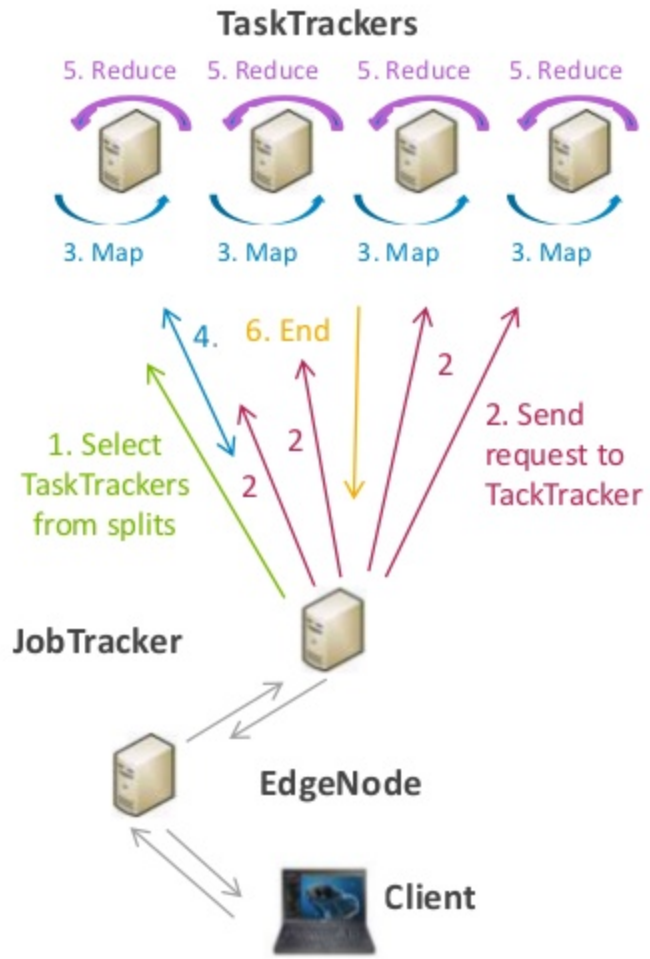
1. Client contacts the NameNode who designates one of the replica as the primary
2. The response of the NameNode contains who is the primary and who are the secondary replicas
3. The client pushes its changes to all DataNodes in any order, but this change is stored in a buffer of each DataNode
4. The client sends a “commit” request to the primary, which determines an order to update and then push this order to all other secondaries
5. After all secondaries complete the commit, the primary response to the client about the success
6. All changes of blocks distribution and metadata changes be written to an operation log file at the NameNode



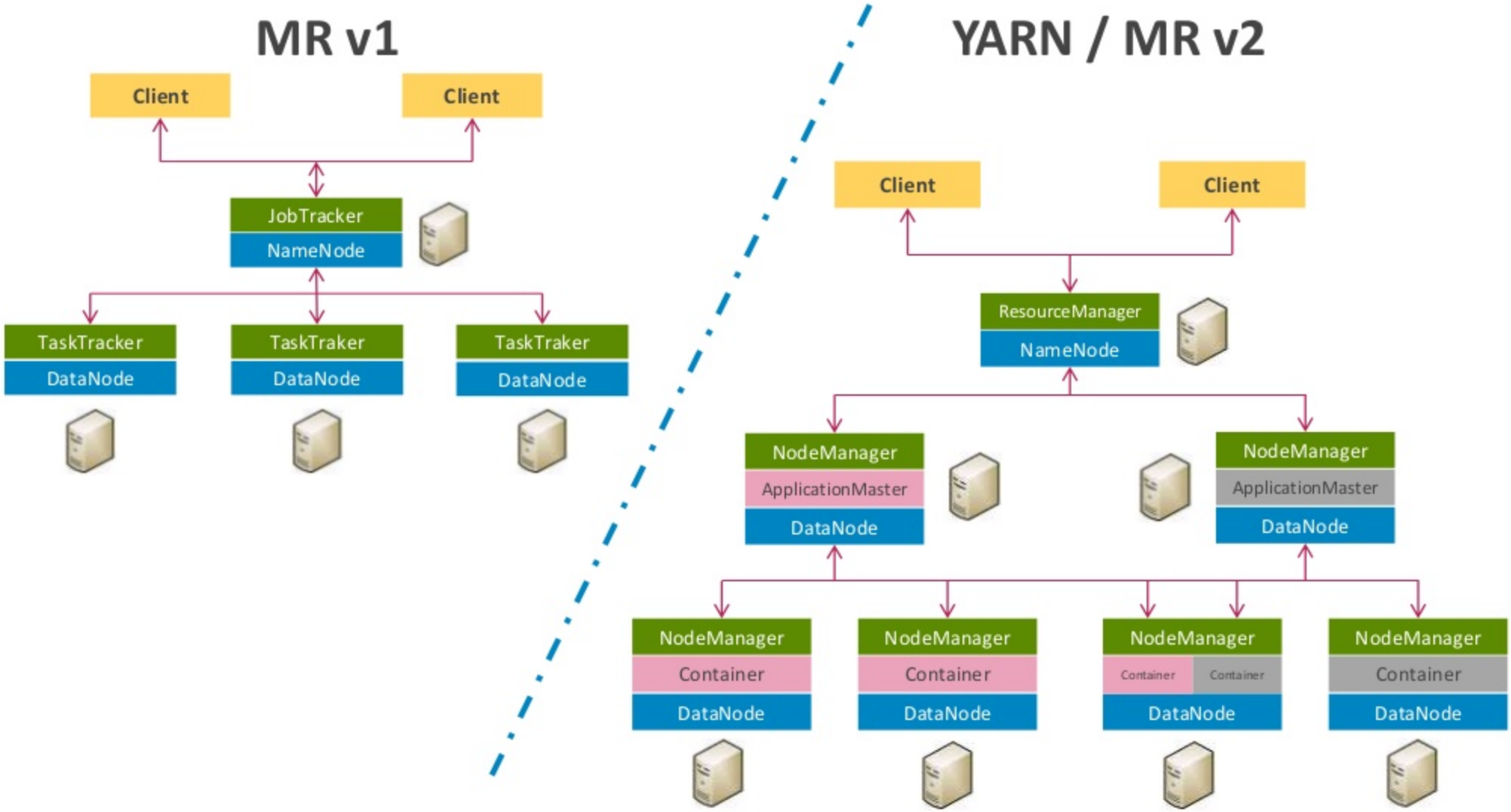
MAPREDUCE - EXEC FILE

Le client program is copied on each node

1. The JobTracker determines the number of splits from the input path, and select some TaskTrackers based on their network proximity to the data sources
2. JobTracker sends the task requests to those selected TaskTrackers
3. Each TaskTracker starts the map phase processing by extracting the input data from the splits
4. When the map task completes, the TaskTracker notifies the JobTracker. When all the TaskTrackers are done, the JobTracker notifies the selected TaskTrackers for the reduce phase
5. Each TaskTracker reads the region files remotely and invokes the reduce function, which collects the key/aggregated value into the output file (one per reducer node)
6. After both phase completes, the JobTracker unblocks the client program



MR VS. YARN ARCHITECTURE



- **YARN** : **Y**et **A**nother **R**esource **N**egotiator
- **MR** : MapReduce

HBASE

- Clone of Big Table (Google)
- Implemented in Java (Clients : Java, C++, Ruby...)
- Data is stored "Column-oriented"
- Distributed over many servers
- Tolerant of machine failure
- Layered over HDFS
- Strong consistency

- It's not a relational database (No joins)
- Sparse data – nulls are stored for free
- Semi-structured or unstructured data
- Data changes through time
- Versioned data
- Scalable – Goal of billions of rows x millions of columns

Table - Example

Row	Timestamp	Animal		Repair
		Type	Size	Cost
Enclosure1	12	Zebra	Medium	1000€
	11	Lion	Big	
Enclosure2	13	Monkey	Small	1500€

Region

Key

Column

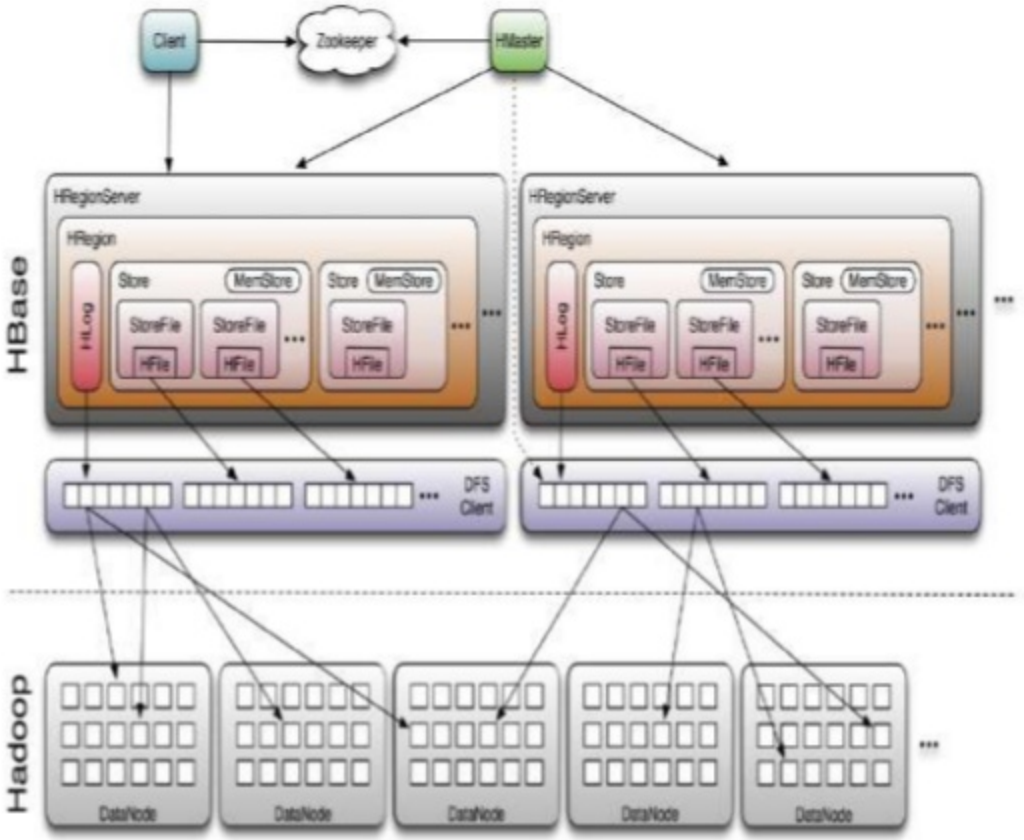
Family

Cell

(Table, Row_Key, Family, Column, Timestamp) = Cell (Value)

HBASE

- Table
  - Regions for scalability, defined by row [start_key, end_key)
  - Store for efficiency, 1 per Family
    - 1..n StoreFiles (HFile format on HDFS)
- Everything is byte
- Rows are ordered sequentially by key
- Special tables -ROOT- , .META.
  - Tell clients where to find user data

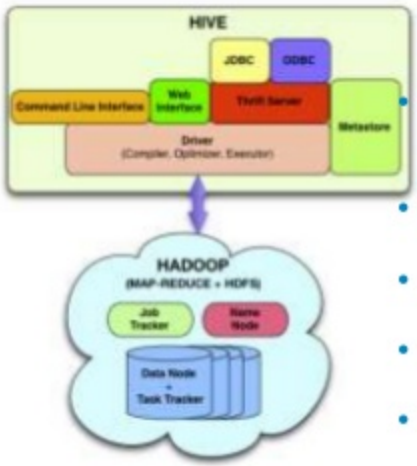


Source: http://www.larsgeorge.com/2009/10/hbase-architecture-101-storage.html

DATA ACCESS

HIVE

- Data Warehouse infrastructure that provides data summarization and ad hoc querying on top of Hadoop
  - MapReduce for execution
  - HDFS for storage
- MetaStore
  - Table/Partitions properties
  - Thrift API : Current clients in PHP (Web Interface), Python interface to Hive, Java (Query Engine and CLI)
  - Metadata stored in any SQL backend
- Hive Query Language
  - Basic SQL : Select, From, Join, Group By
  - Equi-Join, Multi-Table Insert, Multi-Group-By
  - Batch query



PIG

- A high-level data-flow language and execution framework for parallel computation
- Pig Latin
  - Data processing language
  - Compiler to translate to MapReduce
- Simple to write MapReduce program
- Abstracts you from specific detail
- Focus on data processing
- Data flow
- Data manipulation

HCatalog

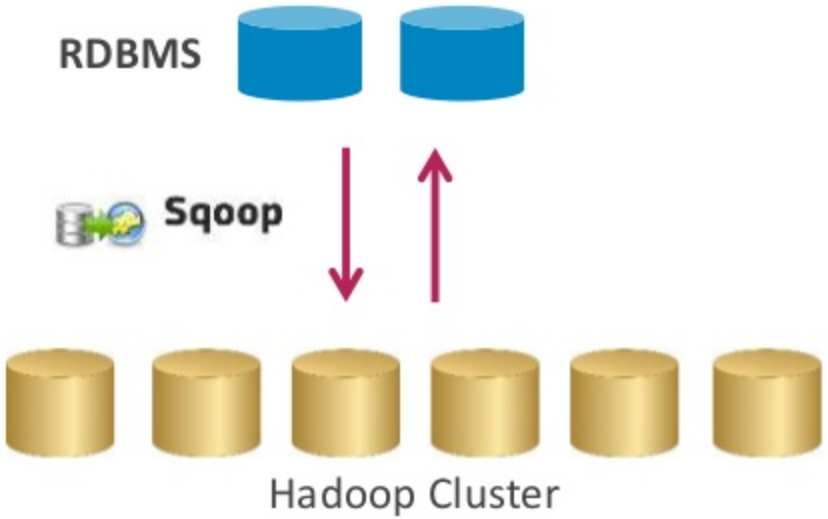
- Table and storage management service for data created using Apache Hadoop
- Providing a shared schema and data type mechanism
- Providing a table abstraction so that users need not be concerned with where or how their data is stored.
- Providing interoperability across data processing tools such as Pig, Map Reduce and Hive
- HCatalog DDL (Data Definition Language)
- HCatalog CLI (Common

HCatalog

DATA TRANSFER

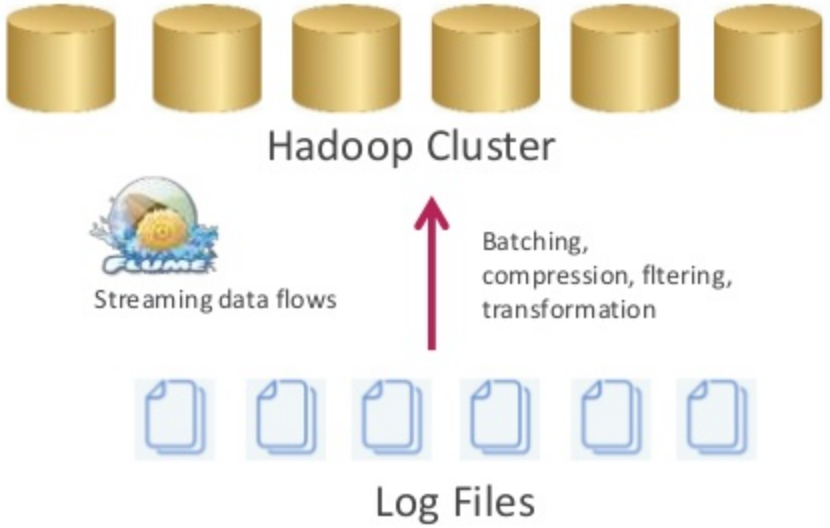
SQOOP

- Data import/export
- Sqoop is a tool designed to help users of large data import existing relational databases into their Hadoop clusters
- Automatic data import
- Easy import data from many databases to Hadoop
- Generates code for use in MapReduce applications



FLUME

- Apache Flume is a distributed, reliable, and available service for efficiently collecting, aggregating, and moving large amounts of log data
- Simple and flexible architecture based on streaming data flows
- Robust fault tolerant with tunable reliability mechanisms and many failover and recovery mechanisms
- The system is centrally managed and allows for intelligent dynamic management



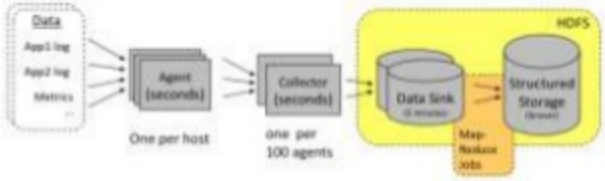
MANAGEMENT

OOZIE

- Oozie is a server based *Bundle Engine* that provides a higher-level Oozie abstraction that will batch a set of coordinator applications. The user will be able to start/stop/suspend/resume/rerun a set coordinator jobs in the bundle level resulting a better and easy operational control
- Oozie is a server based *Coordinator Engine* specialized in running workflows based on time and data triggers
- Oozie is a server based *Workflow Engine* specialized in running workflow jobs with actions that execute Hadoop Map/Reduce and Pig jobs

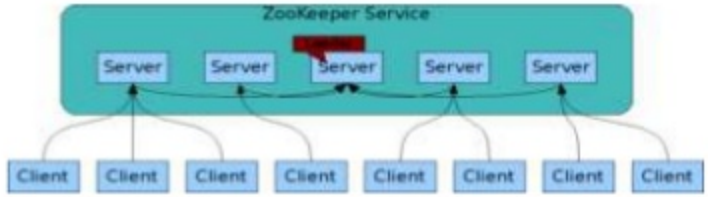
CHUKWA

- A data collection system for managing large distributed systems
- Build on HDFS and MapReduce
- Tools kit for displaying, monitoring and analyzing the log files



ZOOKEEPER

- A high-performance coordination service for distributed applications
- Zookeeper is a centralized service for maintaining configuration information, naming, providing distributed synchronization, and providing group services



MACHINE LEARNING

- Apache Mahout is an Apache project to produce free implementations of distributed or otherwise scalable machine learning algorithms on the Hadoop platform
- Mahout machine learning algorithms:
  - **Recommendation mining**, takes users' behavior and find items said specified user might like
  - **Clustering**, takes e.g. text documents and groups them based on related document topics
  - **Classification**, learns from existing categorized documents what specific category documents look like and is able to assign unlabeled documents to the appropriate category
  - **Frequent item set mining**, takes a set of item groups (e.g. terms in a query session, shopping cart content) and identifies, which individual items typically appear together

SERIALIZATION

- A data serialization system that provides dynamic integration with scripting languages
- Avro Data
  - Expressive
  - Smaller and Faster
  - Dynamic
    - Schema store with data
    - APIs permit reading and creating
  - Include a file format and a textual encoding
- Avro RPC
  - Leverage versioning support
  - For Hadoop service provide cross-language access

MANAGEMENT OPS

WHIRR

- Apache Whirr is a set of libraries for running cloud services
- A common service API
- Provision, Install, Configure and Manage
- Deploy clusters on demand for processing or for testing
- Command line for deploying clusters

AMBARI

- Ambari is a web-based set of tools for
  - Deploying
  - Administering
  - Monitoring

  Apache Hadoop clusters

Ambari

OTHERS APACHE HADOOP PROJECTS

Hadoop Project	Description
TEZ	Tez is an effort to develop a generic application framework which can be used to process arbitrarily complex data-processing tasks and also a re-usable set of data-processing primitives which can be used by other projects.
GORA	Gora is an ORM framework for column stores such as Apache HBase and Apache Cassandra with a specific focus on Hadoop.
DRILL	Drill is a distributed system for interactive analysis of large-scale datasets, inspired by Google's Dremel.
LUCENE	Lucene.NET is a source code, class-per-class, API-per-API and algorithmatic port of the Java Lucene search engine to the C# and .NET platform utilizing Microsoft .NET Framework.
BLUR	Blur is a search platform capable of searching massive amounts of data in a cloud computing environment.
GIRAPH	Giraph is a large-scale, fault-tolerant, Bulk Synchronous Parallel (BSP)-based graph processing framework.
HAMA	Hama is a distributed computing framework based on BSP (Bulk Synchronous Parallel) computing techniques for massive scientific computations, e.g., matrix, graph and network algorithms.
ACCUMULO	Accumulo is a distributed key/value store that provides expressive, cell-level access labels.
CRUNCH	Crunch is a Java library for writing, testing, and running pipelines of MapReduce jobs on Apache Hadoop.
MRUNIT	MRUnit is a library to support unit testing of Hadoop MapReduce jobs.
HADOOP DEVELOPMENT TOOLS	Eclipse based tools for developing applications on the Hadoop platform.
BIGTOP	Bigtop is a project for the development of packaging and tests of the Hadoop ecosystem.
THRIFT	Cross-language serialization and RPC framework.
KNOX	Knox Gateway is a system that provides a single point of secure access for Apache Hadoop clusters.
KAFKA	Kafka is a distributed publish-subscribe system for processing large amounts of streaming data.
CASSANDRA	Cassandra is columnar NoSQL store with scalability, availability and performance capabilities.
FALCON	A data processing and management solution for Hadoop designed for data motion, coordination of data pipelines, lifecycle management, and data discovery.
SENTRY	Sentry is a highly modular system for providing fine grained role based authorization to both data and metadata stored on an Apache Hadoop cluster.
STORM	Storm is a distributed, fault-tolerant, and high-performance real time computation system that provides strong guarantees on the processing of data.
S4	S4 (Simple Scalable Streaming System) is a general-purpose, distributed, scalable, partially fault-tolerant, pluggable platform that allows programmers to easily develop applications for processing continuous, unbounded streams of data.
SPARK	Apache Spark is an open source, parallel data processing both in-memory and on disk, combining batch, streaming, and interactive analytics

https://incubator.apache.org/projects/

SOME HADOOP SERVICES PROVIDERS

Cloudera

MapR

VMware

DataStax

Hortonworks

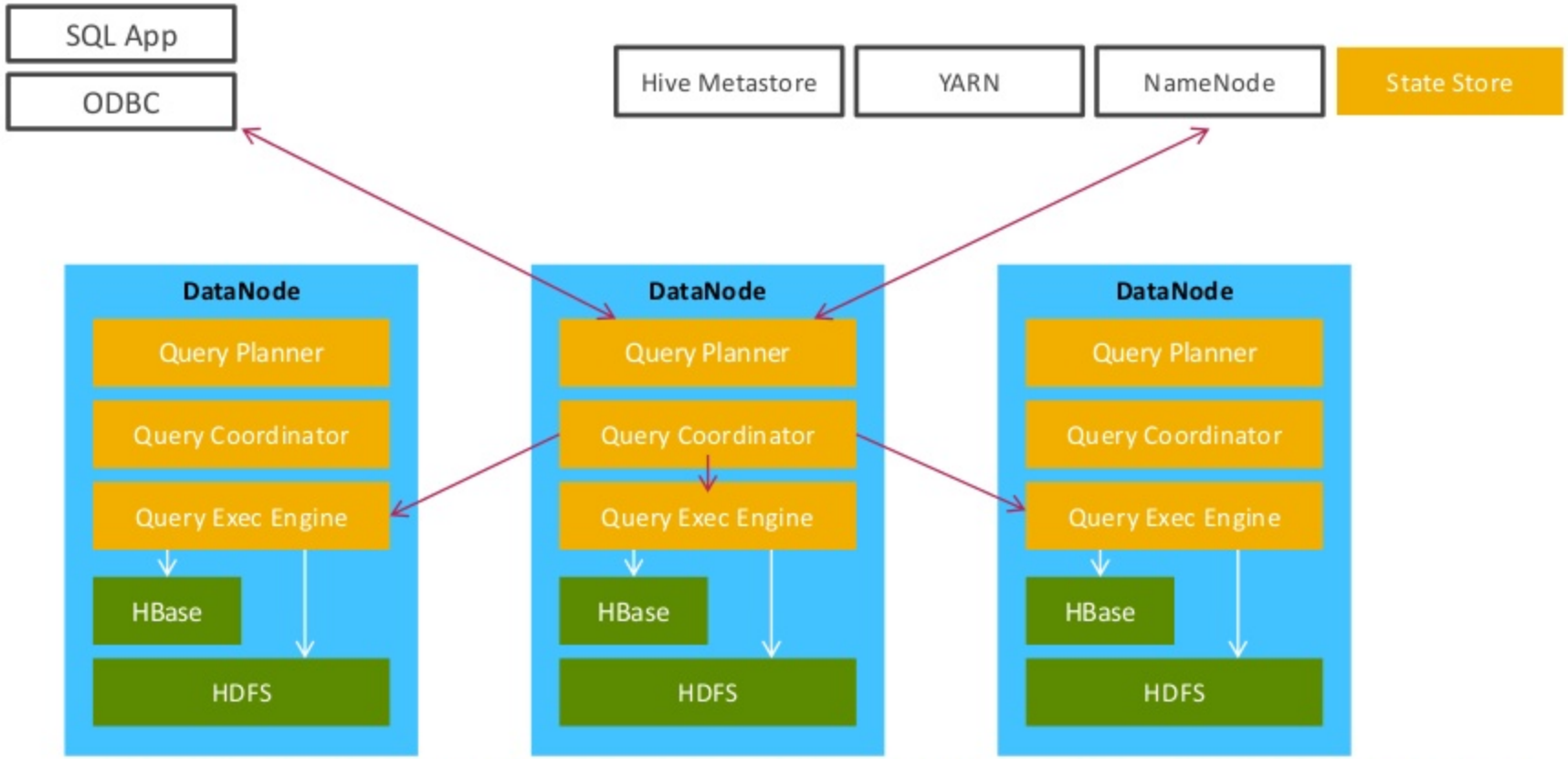
Microsoft

http://wiki.apache.org/hadoop/Distributions%20and%20Commercial%20Support

CLOUDERA IMPALA

- Unified storage : supports HDFS and HBase, Flexible file formats
- Unified Metastore
- Unified security
- Unified client interface : ODBC, SQL syntax, Hue, Beeswax

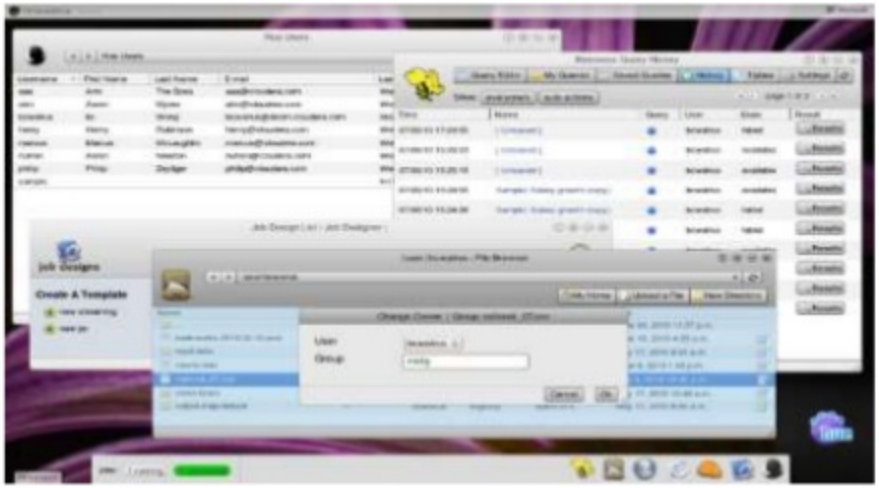
- Impala: real time SQL queries, native distributive query engine, optimized for low-latency
- Answers as fast as you can ask
- Everyone to ask questions for all data, Big Data storage and analytics together



Source : http://cloudera.com/content/cloudera/en/products-and-services/cdh/impala.html

CLOUDERA HUE

- HUE (aka. Hadoop User Experience)
- Open source project started as Cloudera
- HUE is a web UI for Hadoop
- Platform for building custom applications with a nice UI library



- **User Admin**: Account management for HUE users.
- **File Browser**: Browse HDFS; change permissions and ownership; upload, download, view and edit files.
- **Job Designer**: Create MapReduce jobs, which can be templates that prompt for parameters when they are submitted.
- **Job Browser**: View jobs, tasks, counters, logs, etc.
- **Beeswax**: Wizards to help create Hive tables, load data, run and manage Hive queries, and download results in Excel format.
- **Help**: Documentation and help

Source : http://blog.cloudera.com/blog/category/hue/

HORTONWORKS STINGER INITIATIVE

Interactive query for Apache Hive

- The Stinger Initiative is a broad, community-based effort to drive the future of Apache Hive
- Stinger delivers 100x performance improvements at petabyte scale with familiar SQL semantics

Source: http://hortonworks.com/labs/stinger/

MAPR HADOOP

MapR Distribution for Apache Hadoop Advantages

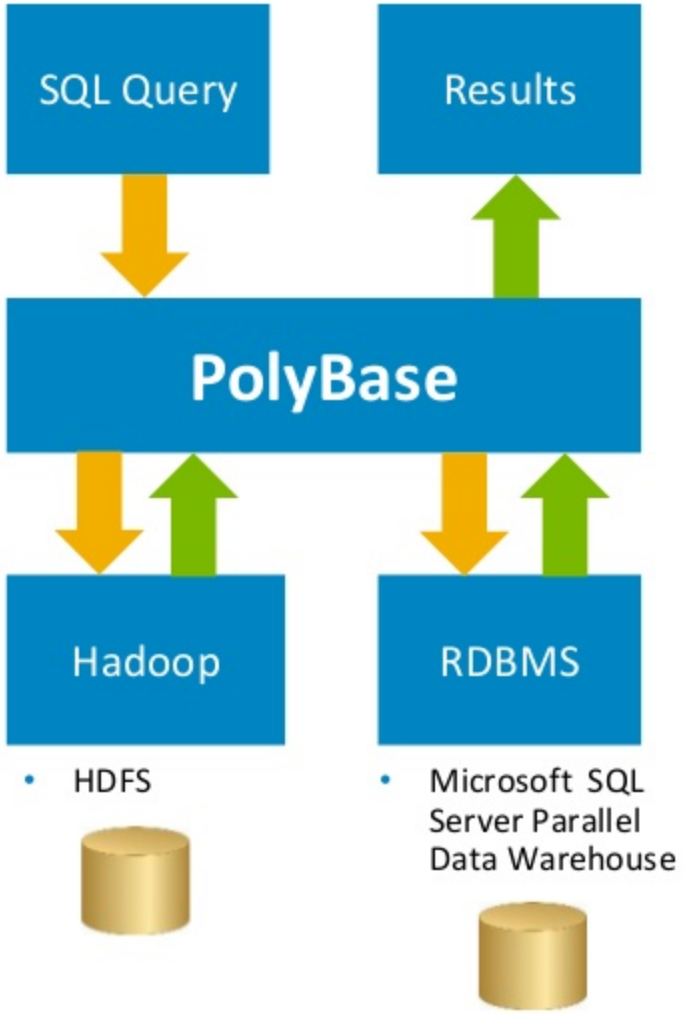
Source : http://www.mapr.com/products/why-mapr

MapR Distribution Editions

Source : http://www.mapr.com/products/mapr-editions

MICROSOFT POLYBASE

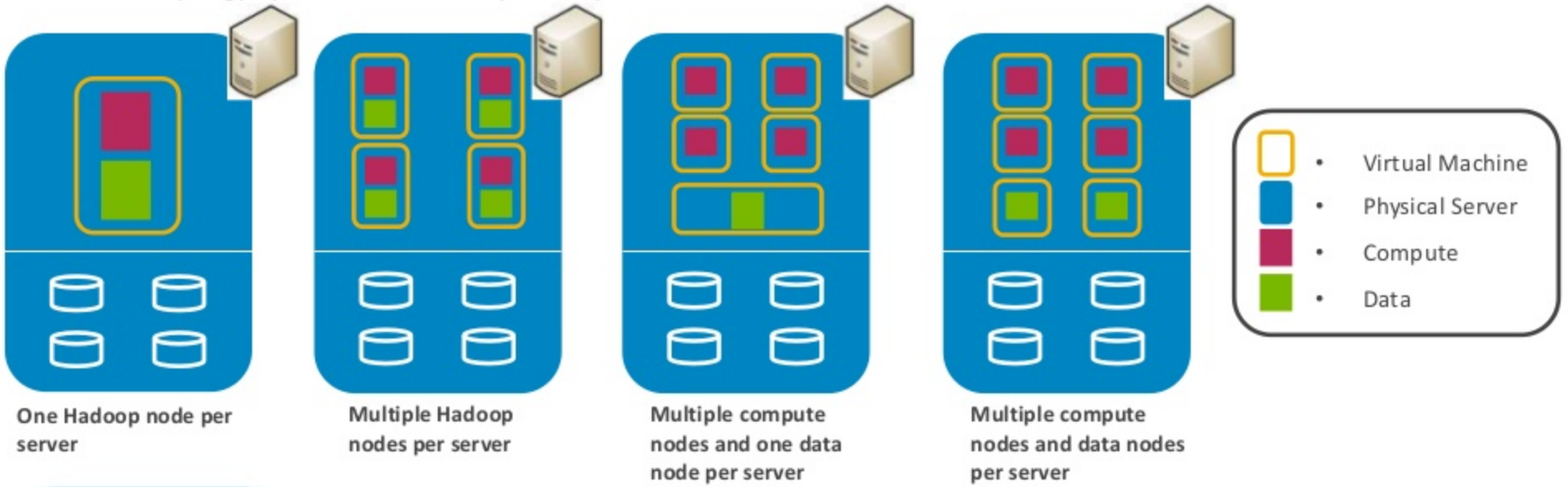
- PolyBase is a technology on the data processing engine in SQL Server Parallel Data Warehouse (PDW) designed as the simplest way to combine non-relational data and traditional relational data in your analysis
- PolyBase provides the easiest and broadest way to access Hadoop with the standard SQL query language without needing to learn MapReduce
- PolyBase moves data in parallel to and from Hadoop and PDW allowing end users to perform their analysis without the help of IT

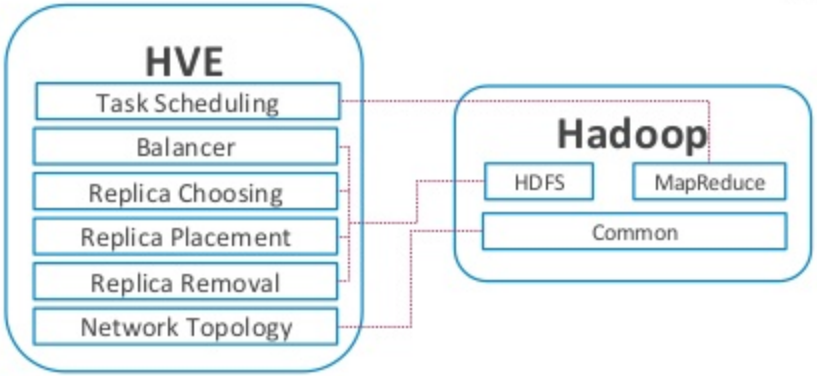


Source : http://www.microsoft.com/en-us/sqlserver/solutions-technologies/data-warehousing/polybase.aspx

VMWARE HADOOP VIRTUALIZATION EXTENSION

- **HADOOP VIRTUALIZATION EXTENSION (HVE)** is designed to enhance the reliability and performance of virtualized Hadoop clusters with extended topology layer and refined locality related policies

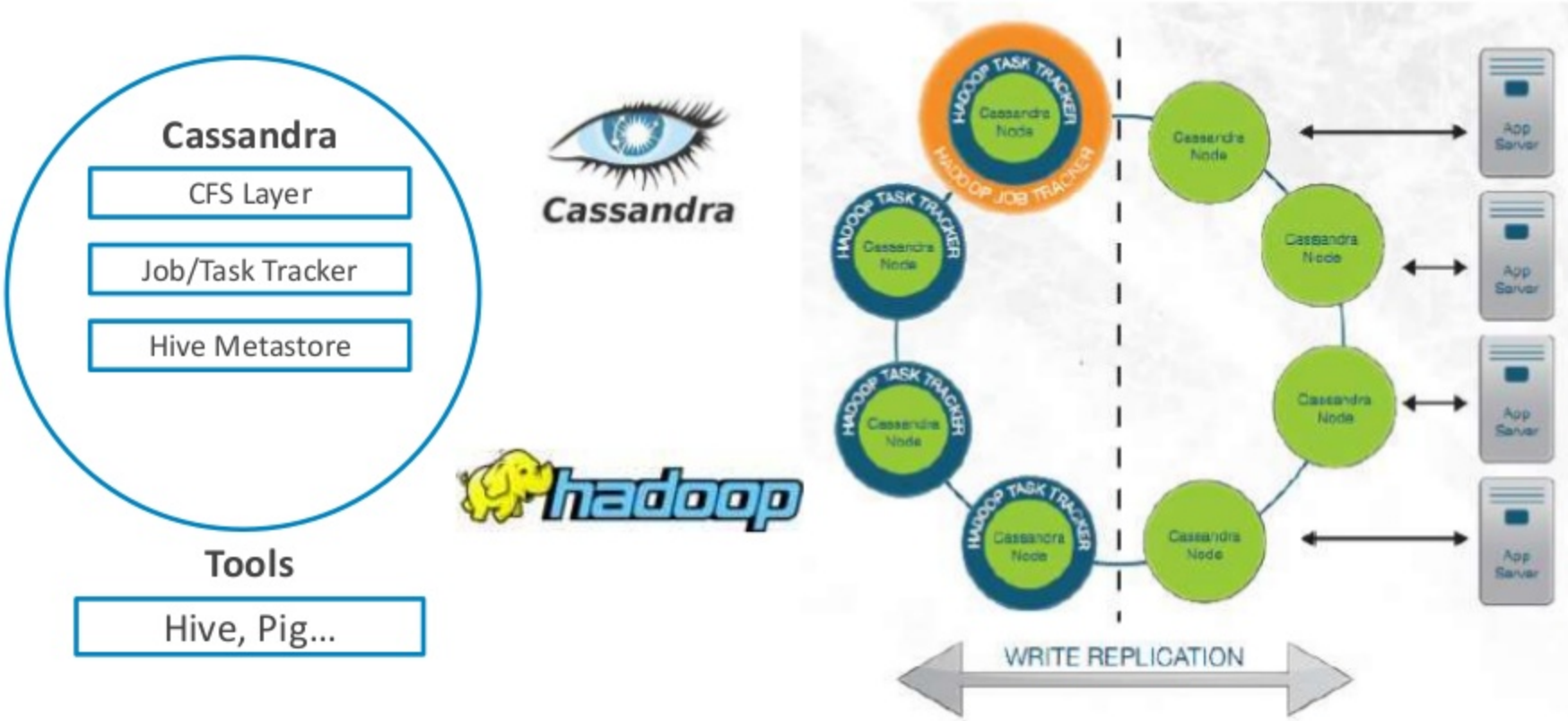




- **HVE** consists of hooks and extensions to data-locality-related components of Hadoop, including: network topology, replica, balancer and task scheduling. It touches all sub projects of Hadoop: Common, HDFS and MapReduce

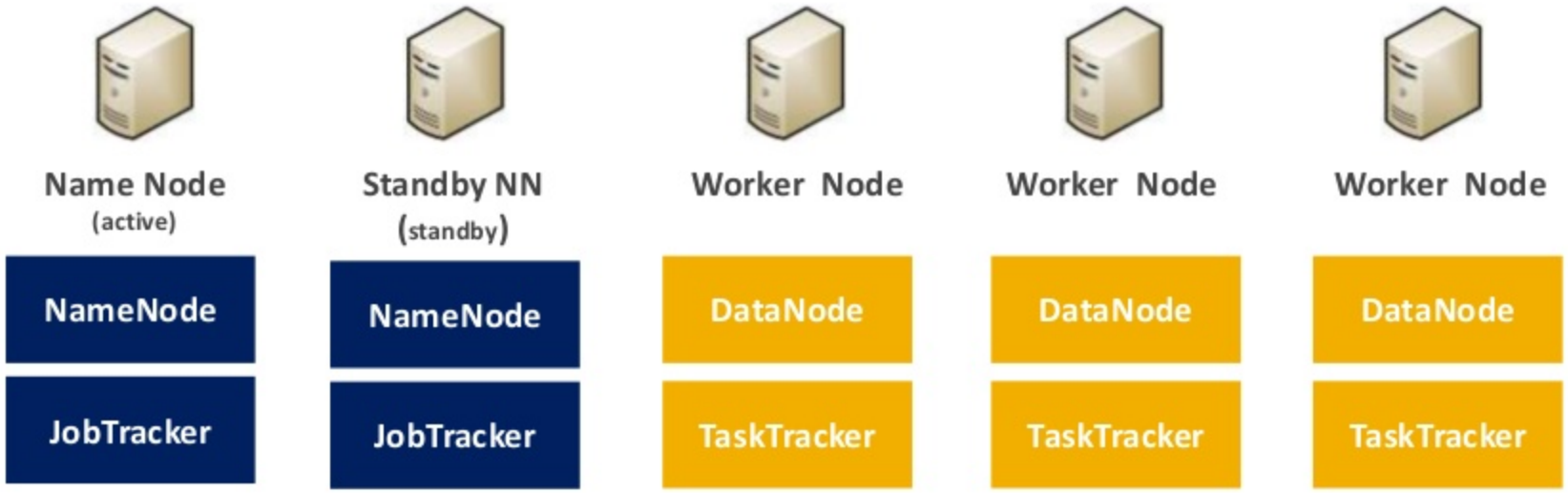
Source: http://www.vmware.com/products/big-data-extensions

HADOOP AND CASSANDRA INTEGRATION



- Brisk solution from DataStax combines the real time capabilities of Cassandra with the analytical power of Hadoop
- The Hadoop Distributed File System (HDFS) is replace by the Apache Cassandra File System (CFS) - Data store in CFS
- Blocks are compressed with Snappy
- Hive Metastore in Cassandra – Automatically maps Cassandra Column Families to Hive tables

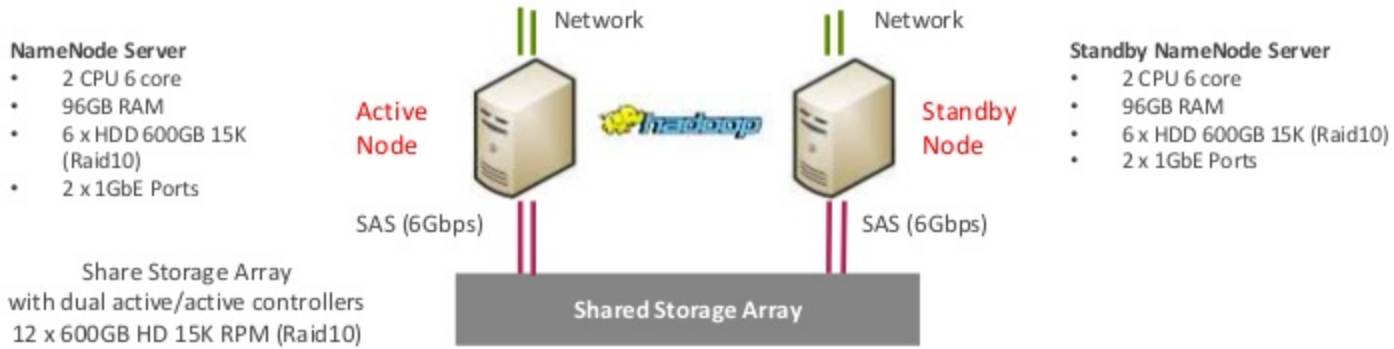
HIGH AVAILABILITY SOLUTIONS



- Automatic blocks replication on 3 DataNodes – Rack awareness
- NameNode and Standby NameNode
- Quorum Journal Manager and Zookeeper
- Disaster Recovery by replication
- Hive and NameNode Metastores backup
- HDFS Snapshots

HA* WITH CONVENTIONAL SHARED STORAGE

- NameNode and Standby NameNode
- Shared storage array

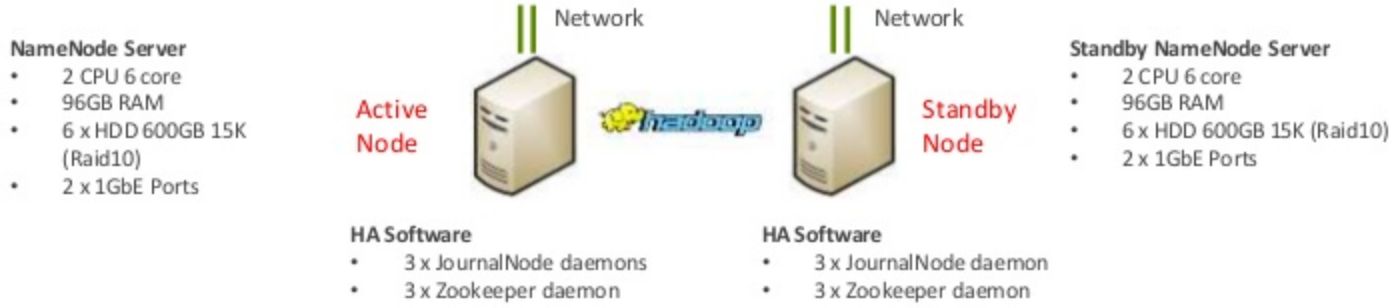


- **NameNode and Standby NameNode Servers:** the Active and Standby nodes should have equivalent hardware
- **Shared Storage Array**: shared directory which both Active node and Standby node servers can have read/write access to. The share storage Array supports NFS and is mounted on each node. Currently only a single shared edits directory is supported. The availability of the system is implemented by the redundancy of the shared edits directory with multiple network paths to the storage, and redundancy in the storage itself (disk, network, and power). It is recommended that the shared storage array be a high-quality dedicated storage.

* High Availability

HA* WITH JOURNAL MANAGER AND ZOOKEEPER

- NameNode and Standby NameNode
- Automatic Failover with Zookeeper : Quorum, ZKFC (ZKFailoverController)
- Quorum Journal Manager for reliable edit log storage

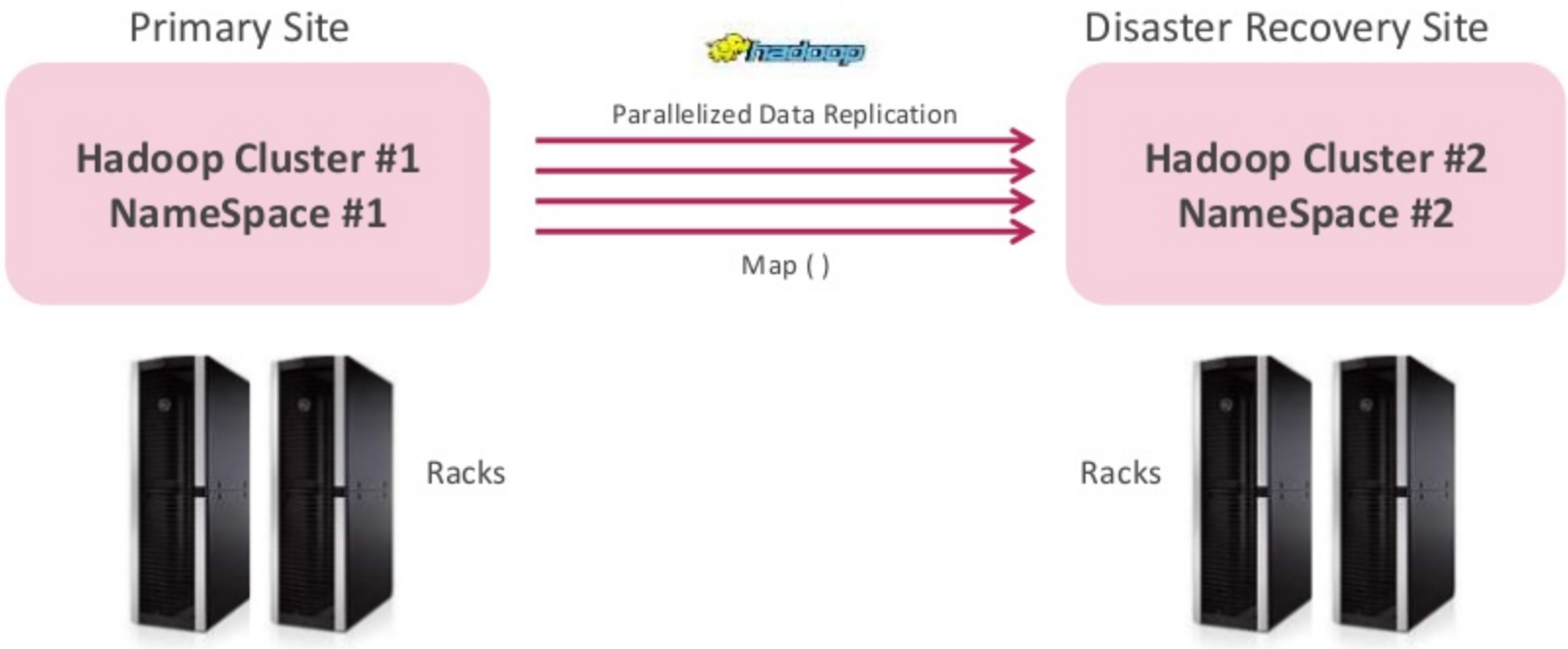


- **NameNode and Standby NameNode Servers:** the Active and Standby nodes should have equivalent hardware
- **Quorum Journal Manager:** In order for the Standby Node to keep its state synchronized with the Active Node, both nodes communicate with a group of separate daemons called "JournalNodes"
- **Zookeeper** : a highly available service for maintaining small amounts of coordination data, notifying clients of changes in that data, and monitoring clients for failures. Zookeeper detects the Active node server failure and actives automatically the Standby node server.

* High Availability

HA* WITH DISASTER RECOVERY

- **DistCp**: tool for parallelized copying of large amounts of data
- Large inter-cluster copy
- Based on MapReduce



Moving an Elephant: Large Scale Hadoop Data Migration at Facebook
http://www.facebook.com/notes/paul-yang/moving-an-elephant-large-scale-hadoop-data-migration-at-facebook/10150246275318920

* High Availability

SECURITY

HDFS / KERBEROS / AD

- Files permissions
  - Files permissions like Unix (owner, group, mode)
- User identity
  - Simple
  - Super-user
- Kerberos connectivity
  - Users authenticate to the edge of the cluster with Kerberos
  - Users and group access is maintained in cluster specific access control lists
- Microsoft Active Directory connectivity

KNOX

- Knox is a system that provides a single point of authentication and access for Apache Hadoop services in a cluster
- Knox simplifies Hadoop security for users who access the cluster data and execute jobs, and for operators who control access and manage the cluster
- Knox runs as a server or cluster of servers that serve one or more Hadoop clusters

GAZZANG

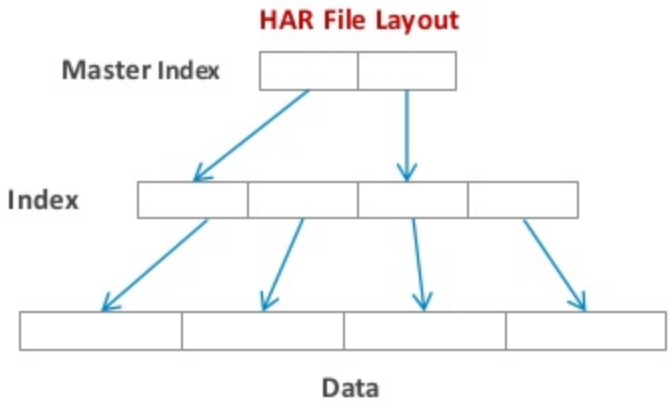
- Advanced Key Management - Stores keys separate from the encrypted data
- Transparent Data Encryption - Protects data at rest resulting in minimal performance impact
- Process Based Access Controls - Restricts access to specific processes rather than by OS user
- Encrypt and Decrypt Unstructured Data - Secures sensitive data that could be considered damaging if exposed outside the business
- Automation Tools - Rapid distributed deployment from ten to thousands of data nodes

BACKUP

- Replication blocks is not a form of backup
- HDFS Snapshots
- When a file is deleted by a user or an application, it is not immediately removed from HDFS
  - The file is moved in the /trash directory
  - The file can be restored quickly as long as it remains in /trash
  - A file remains in /trash for a configurable amount of time
- When a file is corrupted restore from backup is necessary
  - Create incremental backup HDFS files
  - Use a time-stamp of the file
  - Use a staging area to store the backup files
  - Move the backup files to tape if necessary
- Backing up with DistCp or copyToLocal commands

ARCHIVING

- Hadoop Archives, or HAR files, are a file archiving facility that packs files into HDFS blocks more efficiently
- Reduce the NameNode memory usage while still allowing transparent access to files.
- An effective solution for the small files problem
  - http://developer.yahoo.com/blogs/hadoop/posts/2010/07/hadoop_archive_file_compaction/
  - http://www.cloudera.com/blog/2009/02/the-small-files-problem/
- Archives are immutable
- Rename, deletes and create return an error
- Hadoop Archives is exposed as a file system MapReduce will be able to use all the logical input files in Hadoop Archives as input



DATA COMPRESSION

LZO

- https://github.com/toddlipcon/hadoop-lzo
- By enabling compression, the store file uses a compression algorithm on blocks as they are written (during flushes and compactions) and thus must be decompressed when reading
- Compression reduces the number of bytes written/read to/from HDFS
- Compression effectively improves the efficiency of network bandwidth and disk space
- Compression reduces the size of data needed to be read when issuing a read

SNAPPY

- Hadoop Snappy is a project for Hadoop that provide access to the snappy compression. http://code.google.com/p/snappy/
- Hadoop-Snappy can be used as an add-on for recent (released) versions of Hadoop that do not provide Snappy Codec support yet
- Hadoop-Snappy is being kept in synch with Hadoop Common
- Snappy is a compression/decompression library. It does not aim for maximum compression, or compatibility with any other compression library

HDFS OVER NFS & CIFS PROTOCOLS

- Make a HDFS filesystem available across networks as a exported share
- **Over NFS**
  - Install FUSE server
  - Mount HDFS filesystem over FUSE
  - Export HDFS filesystem
- **Over CIFS**
  - Install FUSE server
  - Install SAMBA server on FUSE server
  - Mount HDFS filesystem over SAMBA
  - Export HDFS filesystem

FUSE

FUSE is a framework which makes it possible to implement a filesystem in a userspace program.
Features include:

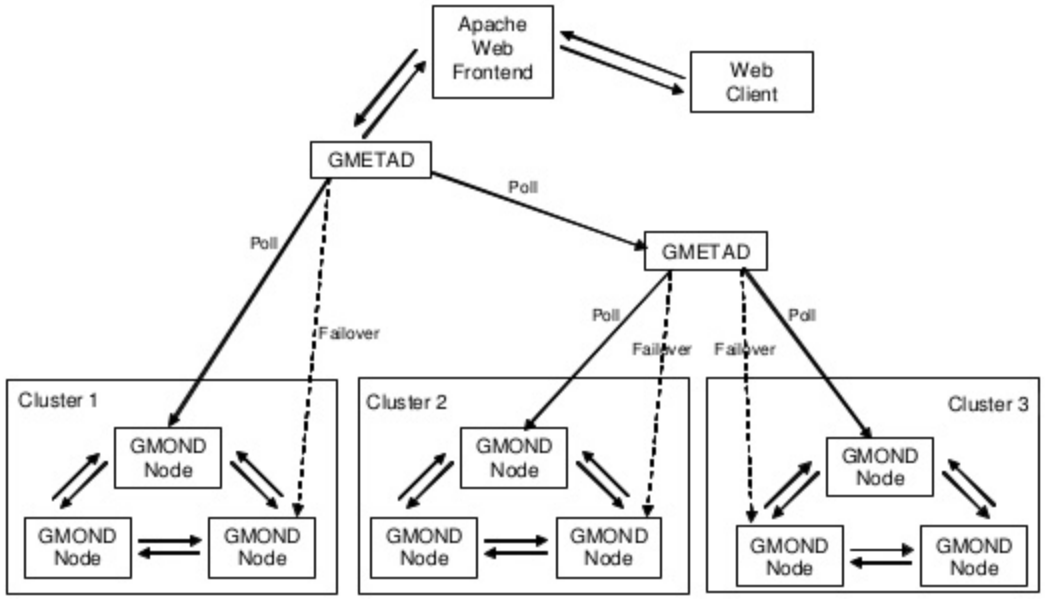
- Simple yet comprehensive API
- Secure mounting by non-root user
- Multi-threaded operation

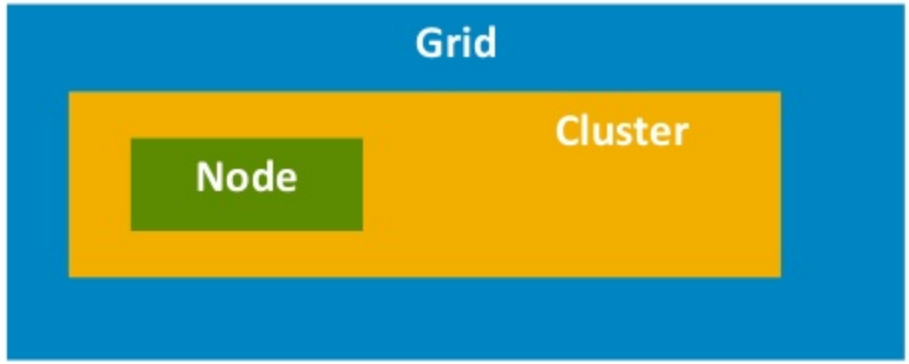
SAMBA

SAMBA is a suite of Linux applications that speak the SMB (Server Message Block) protocol. Many operating systems, including Windows use SMB to perform client-server networking.

GANGLIA MONITORING

- Ganglia by itself is a highly scalable cluster monitoring tool, and provides visual information on the state of individual machines in a cluster or summary information for a cluster or sets of clusters. Ganglia provides the ability to view different time windows
- 2 daemons: GMON & GMETAD
- GMOND collects or received metric data on each DataNode
- 1 GMETAD/grid
- Polls 1 GMOND per cluster for data
- A node belongs to a cluster
- A cluster belongs to a grid



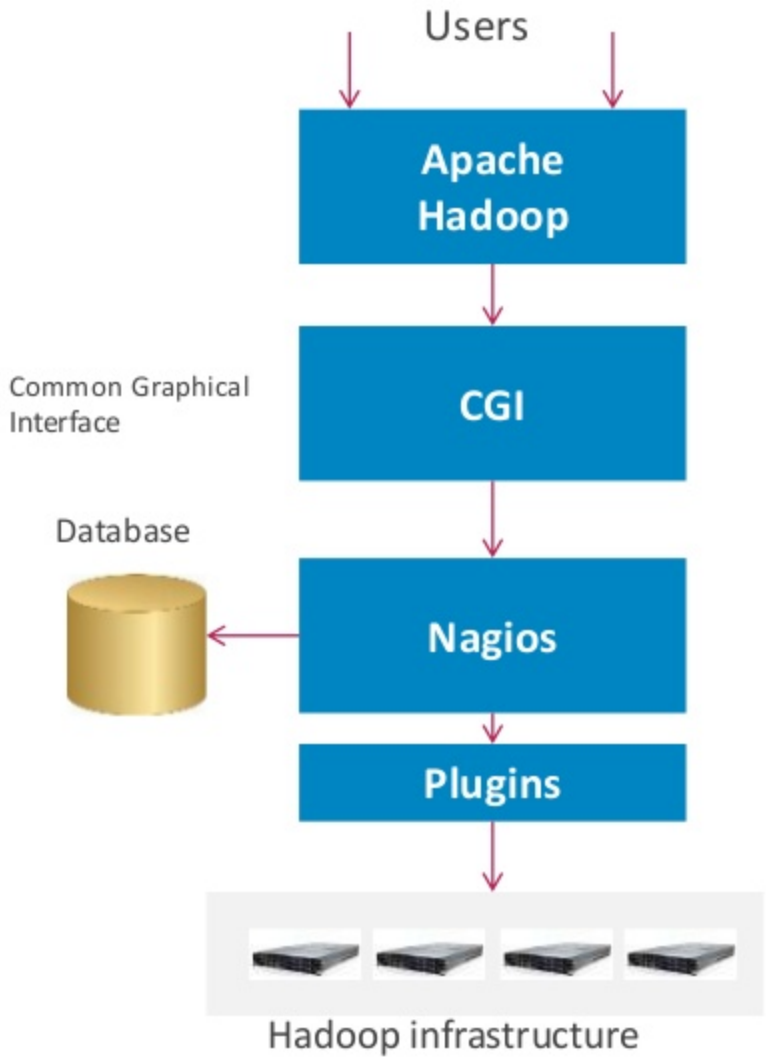


Source : http://ganglia.sourceforge.net/

NAGIOS MONITORING

- A system and network monitoring application
- Nagios is an open source tool especially developed to monitor hosts and services and designed to inform the network incidents before end-users, clients do
- Nagios watches hosts and services which we specify and alerts when things go bad and when things get recovered
- Initially developed for servers and applications monitoring, it is widely used to monitor and networks availability

Source : http://www.nagios.org/



CROWBAR SOFTWARE FOR HADOOP

CROWBAR
The cloud *un*boxer.

Solutions

A modular, open source framework that accelerates multi-node deployments, simplifies maintenance, and streamlines ongoing updates

- Deploy Hadoop cluster in hours instead of days
- Use or build barclamps to install and configure software modules
- Supports a cloud operations model to interact, modify, and build based on changing needs

Componants

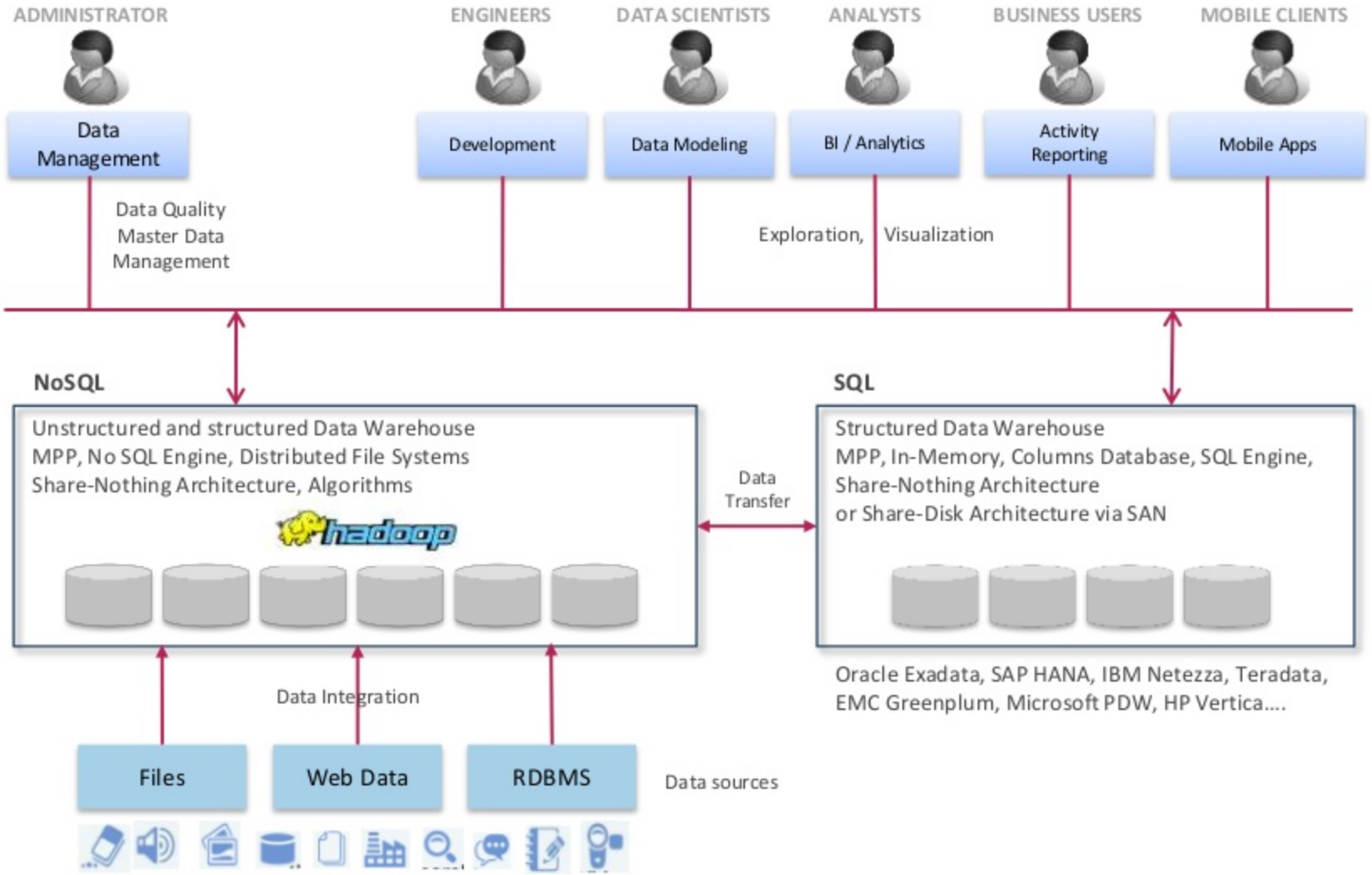
Opscode Chef Server Capabilities

Download the open source software:
https://github.com/dellcloudedge/crowbar

Active community
http://lists.us.dell.com/mailman/listinfo/crowbar

Resources on the Wiki:
https://github.com/dellcloudedge/crowbar/wiki

LOGICAL DATA WAREHOUSE WITH HADOOP



INFRASTRUCTURE RECOMMENDATIONS

General

- Determine the volume of data that needs to be stored and processed
- Determine the rate at which the data is expected to grow
- Determine when the cluster needs to grow, and whether there will be a need for additional processing and storage
- For greater power efficiency and higher ROI over time, choose machines with more capacity. This helps to reduce the frequency of new machines being added
- For high availability 2 Admin servers in different racks are recommended
- For high availability 2 EdgeNodes mini in different racks are recommended

Storage

- The number of disks should be based on the amount of raw data required
- Check on the rate of growth of data and try reducing the requirement of adding new machines every year. For example, depending on the net new data per year, it may be worthwhile using 12 hard drives per server than using 6 hard drives per server, to accommodate larger amount of new data in the existing cluster

Network

- Use 10GbE switch per rack according the performance of the Hadoop cluster
- Use low-latency 10GbE switch across multiple racks
- For high availability 2 network switches on top of each rack are recommended

DataNode

- Each DataNode runs a TaskTracker daemons
- 2 CPU 6-core is recommended for each DataNode for mostly cases
- For increase the I/O performance use the SAS 15K RPM disk, otherwise the SATA/SAS NL 7.2K RPM at better price is sufficient
- Using RAID is not recommended
- JBOD configuration is required. HDFS provides built-in redundancy by replicating blocks across multiple nodes. The x3 replication factor is recommended
- 48GB RAM per server is recommended for mostly cases
- For tmp, log , etc, add 20% to usable disk space
- The ratio between useable data and raw data is 3.6

NameNode

- The NameNode runs a JobTracker daemons
- A copy of the NameNode metadata is stored on a separate machine
- Losing the NameNode metadata would mean all data in HDFS lost. Use the Standby NameNode for high availability
- The NameNode is not commodity hardware, and needs to have sufficient RAM and disk performance
- The amount of RAM allocated to the NameNode limits the size of the cluster
- Having plenty of extra NameNode memory space is highly recommended, so that the cluster can grow without having to add more memory to the NameNode, requiring a restart.
- 96GB of RAM per server is recommended for mostly cases in the large cluster
- Using RAID10 and 15K RPM disks is highly recommended
- The NameNode and the secondary NameNode are the same server configuration

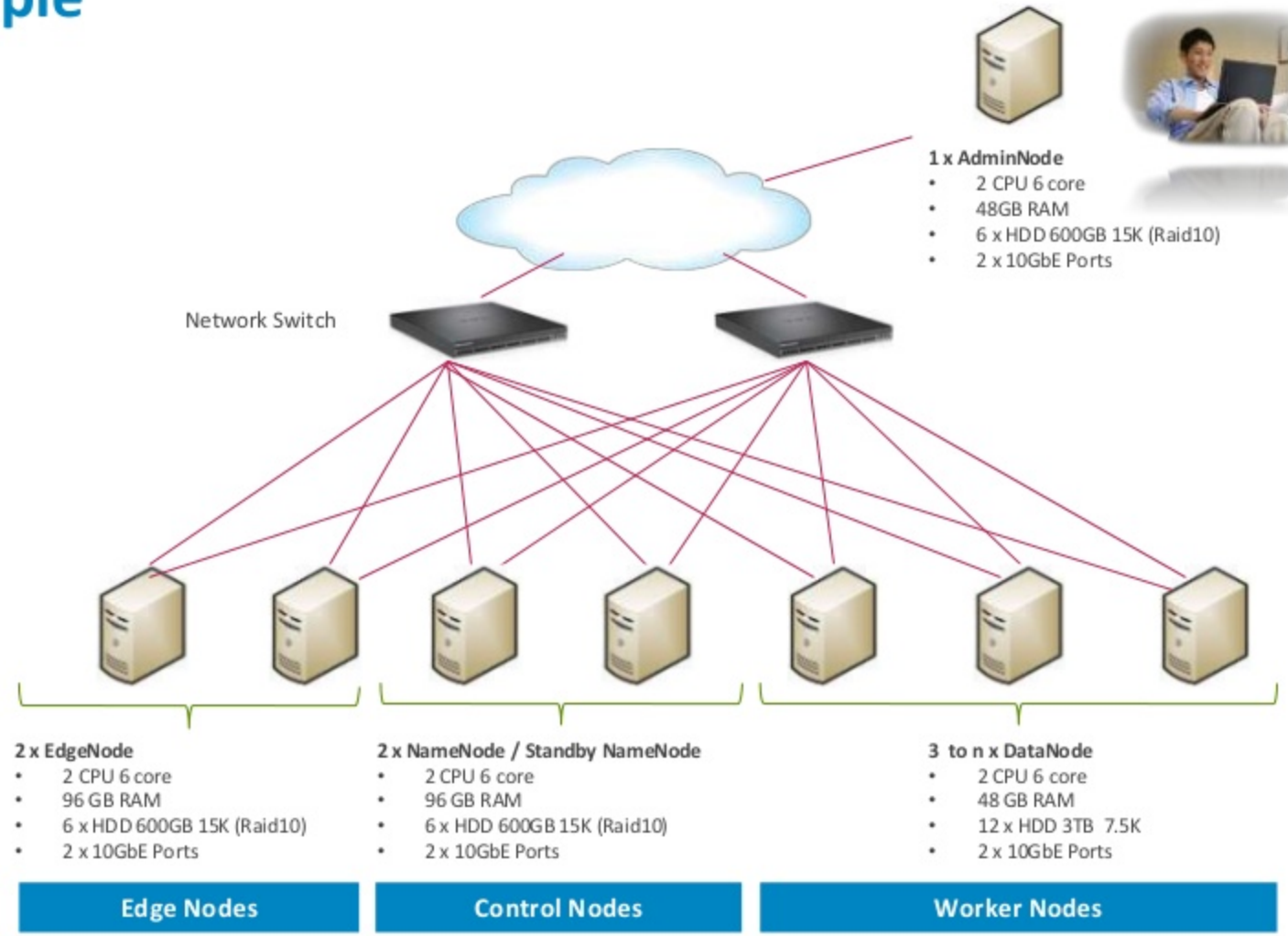
INFRASTRUCTURE SIZING

Feature	Description & Formula	Example
Replication_blocks	• Number of replication blocks • 3 recommended	• 3
Usable_data_volume	• Data source volume (business data)	• 400TB
Temp_data_volume_ratio	• tmp, log… • 20% of Usable_data_volume	• 1,2
Data_compression_ratio	• Data compression ratio	• 0,6
Raw_data_volume	• (Usable_data_volume x Replication_blocks x Temp_data_volume_ratio x Data_compression_ratio	• 400 x 3 x 1,2 x 0,6 = 864TB
Rack_units_per_rack	• Number of rack units in one rack	• 42RU
Rack_units_switch	• Number of rack units for a network switches in one rack	• 2RU
Rack_units_per_DataNode	• Number of rack units for one DataNode	• 2RU
DataNode_volume	• Raw data volume in one DataNode	• 12 x Disks 2TB = 24TB
DataNodes	• Number of DataNodes • Raw_data_volume / DataNode_volume	• 864/24 = 36 DataNodes
Data Racks	• Number of racks • (DataNodes x Rack_units_per_DataNode) / (Rack_units_per_rack – Rack_units_switch)	• (36 x 2) / (42 – 2) = 2 Data Racks

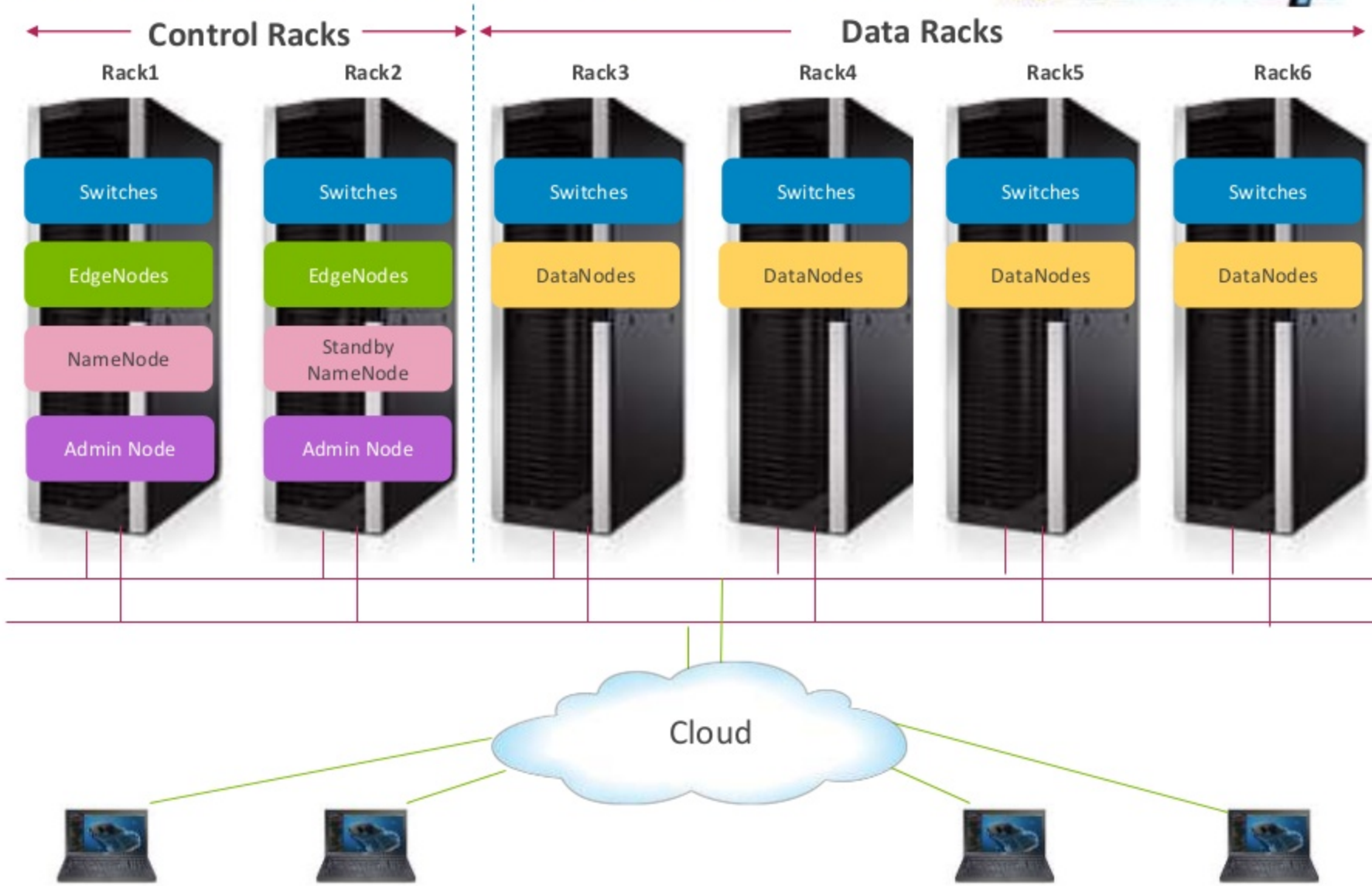
- Add 1 or 2 x Control Rack 24RU integrating NameNode, Secondary NameNode, EdgeNodes and AdminNodes depending on availability of the Hadoop cluster

HADOOP ARCHITECTURE

Example



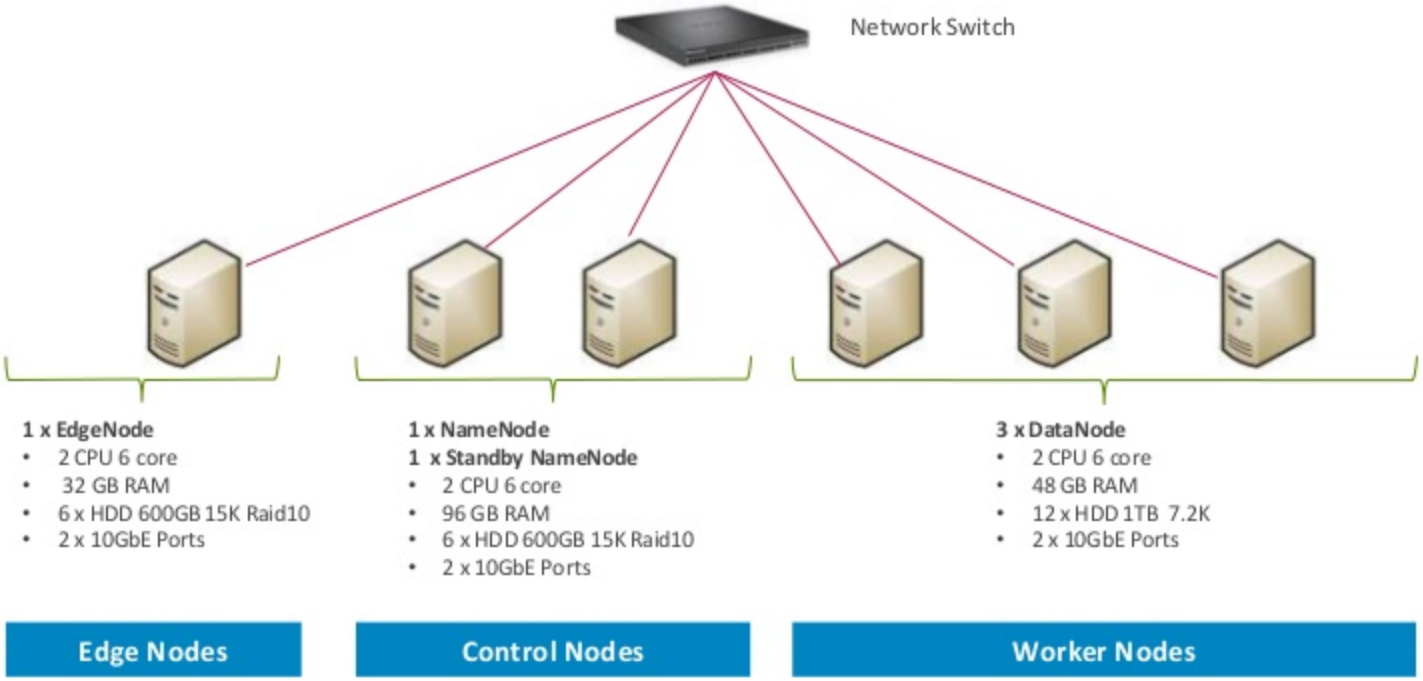
RACKS CONFIGURATION OVERVIEW



POC CONFIGURATION

Example

- Architecture example
- The exact configuration and sizing is designed depending on the customer's needs
- AdminNode in on Standby NameNode server
- Zookeeper processes are on NameNode and Standby NameNode servers



HADOOP BENCHMARKS

- Designing appropriate hardware for a Hadoop cluster requires benchmarking or POC and careful planning to fully understand the workload. However, Hadoop clusters are commonly heterogeneous and it is recommended deploying initial hardware with balanced specifications when getting started
- HiBench, a Hadoop benchmark suite constructed by Intel, is used intensively for Hadoop benchmarking, tuning & optimizations
- A set of representative Hadoop programs including both micro-benchmarks and more "real world" applications such as: search, machine learning and Hive queries

Source: Intel Cloud Builder Guide to Cloud Design and Deployment on Intel Platform – Apache Hadoop - February 2012

HIBENCH

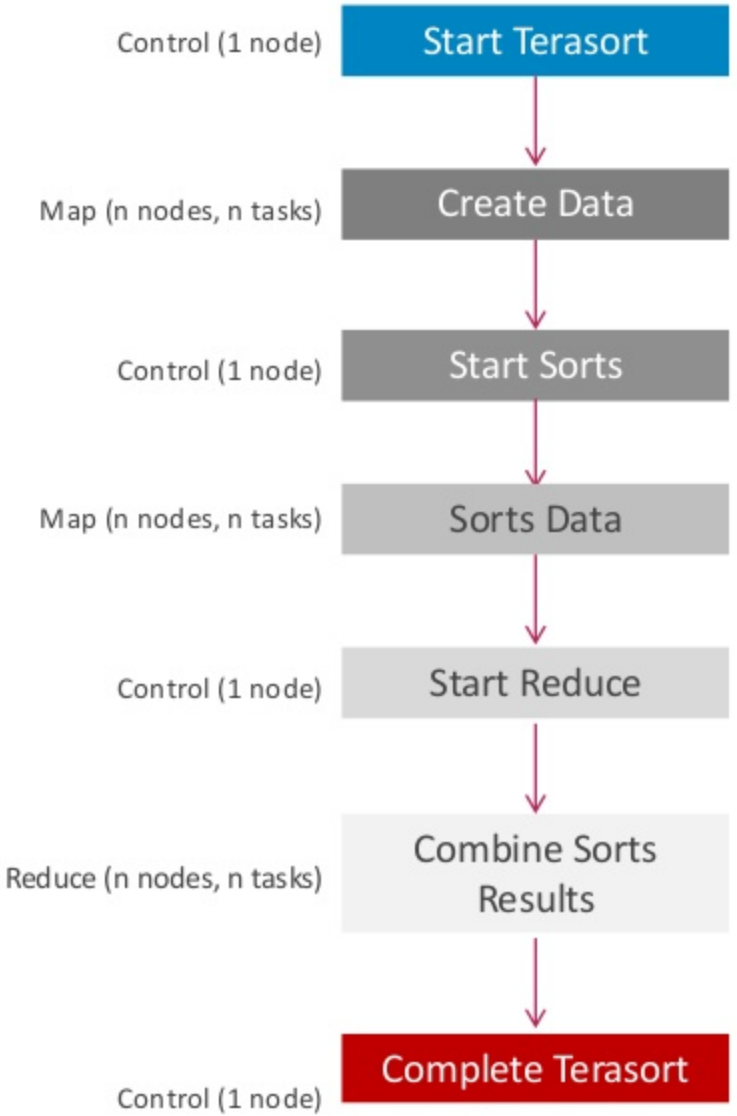
Category	Workload	Description
Microbenchmarks	Sort	This workload sorts its binary input data, which is generated using the Hadoop* RandomTextWriter example.
	WordCount	This workload counts the occurrence of each words in the input data which is generated using Hadoop RamdomTextWrter
	TeraSort	A standard benchmark for large-size data sorting that is generated by the TeraGen program
	DFSIO	Computes the aggregated bandwidth by sampling the number of bytes read/written at fixed time intervals in each map task
Web Search	Nutch Indexing	This workload tests the indexing subsystem in Nutch, a popular Apache* open-source search engine. The crawler subsystem in Nutch is used to crawl an in-house Wikipedia* mirror and generates 8.4 GB of compressed data (for about 2.4 million web pages) total as workload input
	Page Rank	This workload is an open-source implementation of the page-rank algorithm, a link analysis algorithm used widely in Web search engines
Machine Learning	K-Means Clustering	Typical application area of MapReduce for large-scale data mining and machine learning
	Bayesian Classification	This workload tests the naive Bayesian (a well-known classification algorithm for knowledge discovery and data mining) trainer in Mahout*, which is an Apache open-source machine-learning library
Analytical Query	Hive Join	This workload models complex analytic queries of structured (relational) tables by computing the sum of each group over a single read-only table
	Hive Aggregation	This workload models complex analytic queries of structured (relational) tables by computing both the average and sum for each group by joining two different tables

HADOOP TERASORT WORKFLOW

- **Teragen** is a utility included with Hadoop for use when creating data sets that will be used by Terasort. Teragen utilizes the parallel framework within Hadoop to quickly create large data sets that can be manipulated. The time to create a given data set is an important point when tracking performance of a Hadoop environment

- **Terasort** benchmark tests HDFS and MapReduce functions in the Hadoop cluster. Terasort is a compute-intensive operation that utilizes the Teragen output as the Terasort input. Terasort will read the data created by Teragen into the system's physical memory and then sort it and write it back out to the HDFS. Terasort will exercise all portions of the Hadoop environment during these operations

- **Teravalidate** is used to ensure the data produced by Terasort is accurate. It will run across the Terasort output data and verify all data is properly sorted, with no errors produced, and let the user know the status of the results



THANK YOU