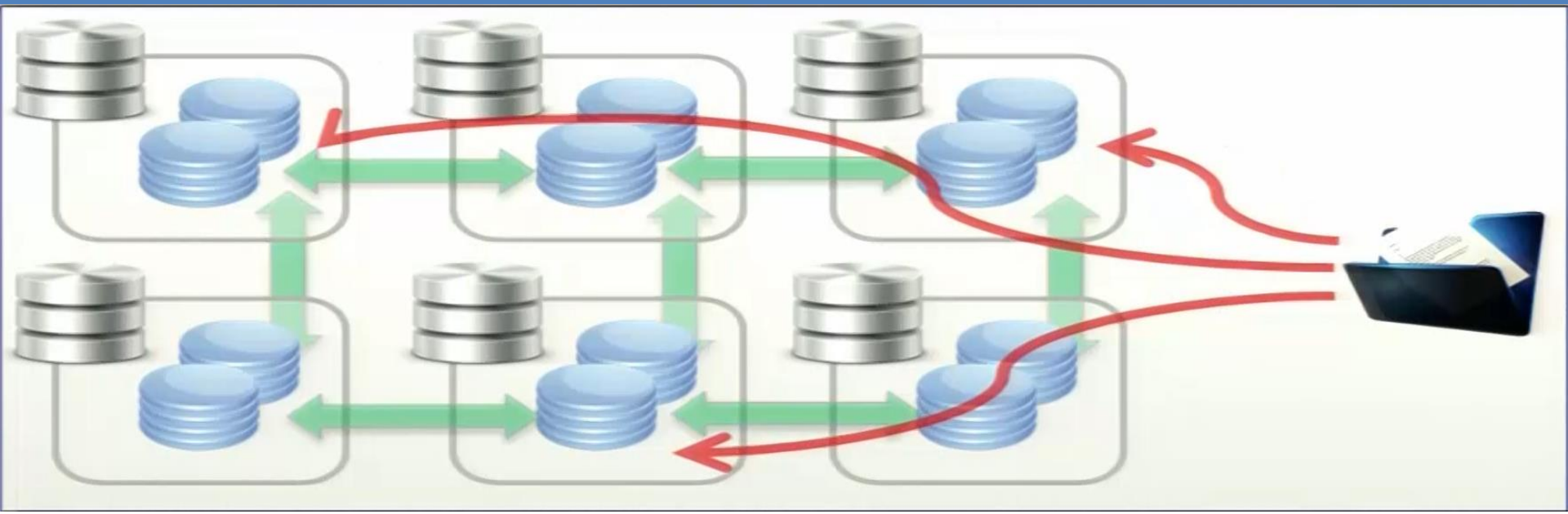
# *Big Data Engineering and Analytics*

# *(DS6132)*


## Hadoop Stack for Big Data

- Hadoop Technologies Opportunities
- Hadoop Technologies Challenges
- Hadoop Stack Applications
- Hadoop Stack Technologies

# Moving Computation to Data



- Hadoop started out as a simple batch processing framework.
- The idea behind Hadoop is that instead of moving data to computation, we move computation to data.

# Scalability

- Scalability's at it's core of a Hadoop system.
- We have cheap computing storage.
- We can distribute and scale across very easily in a very cost effective manner.

# Reliability

- Hardware Failures Handles Automatically!


- If we think about an individual machine or rack of machines, or a large cluster or super computer, they all fail at some point of time or some of their components will fail. These failures are so common that we have to account for them ahead of the time.
- And all of these are actually handled within the Hadoop framework system. So the Apache's Hadoop MapReduce and HDFS components were originally derived from the Google's MapReduce and Google's file system. Another very interesting thing that Hadoop brings is a new approach to data.

# New Approach to Data: Keep all data

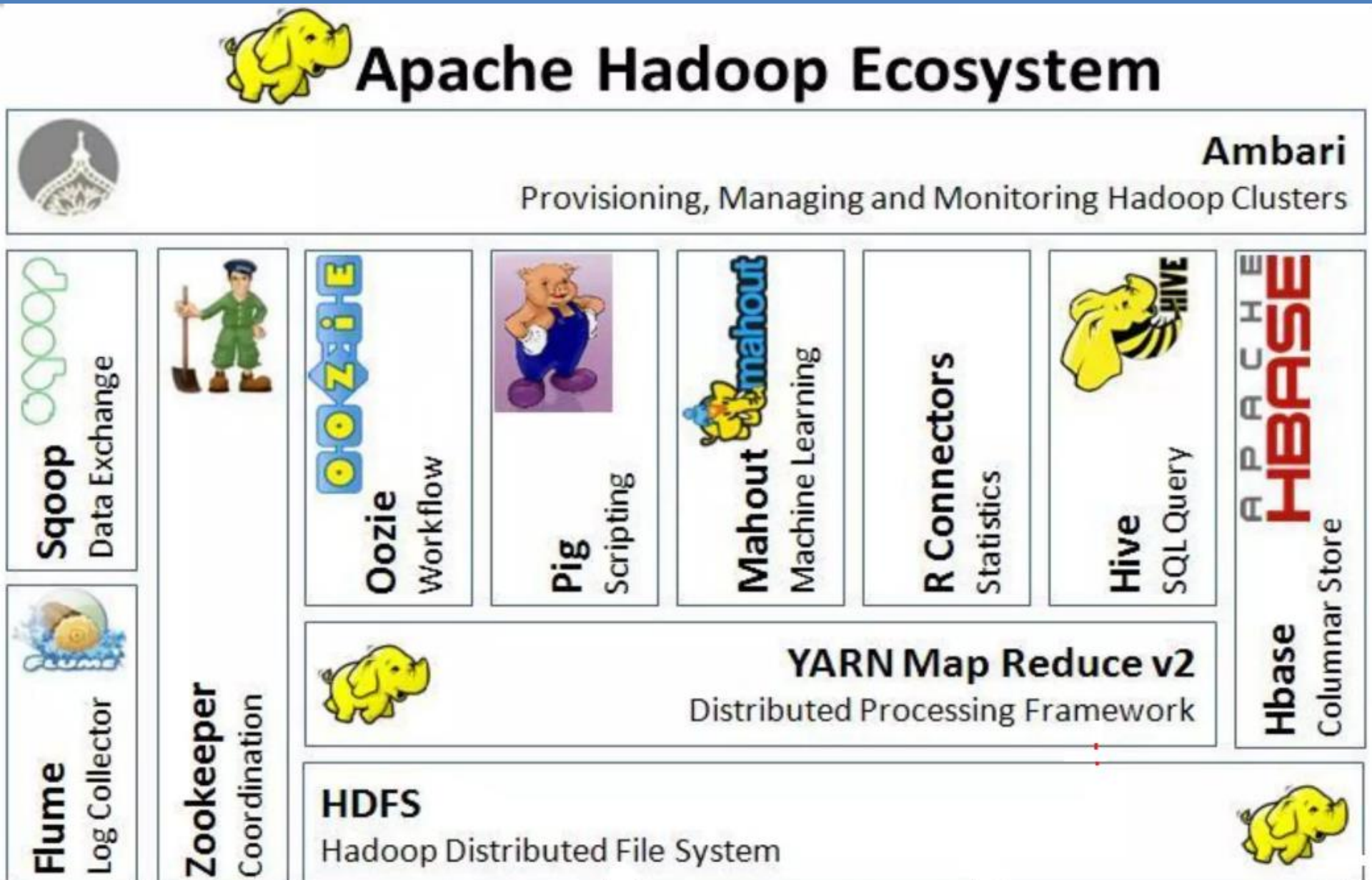


- A new approach is, we can keep all the data that we have, and we can take that data and analyze it in new interesting ways. We can do something that's called schema and read style.
- And we can actually allow new analysis. We can bring more data into simple algorithms, which has shown that with more granularity, you can actually achieve often better results in taking a small amount of data and then some really complex analytics on it.

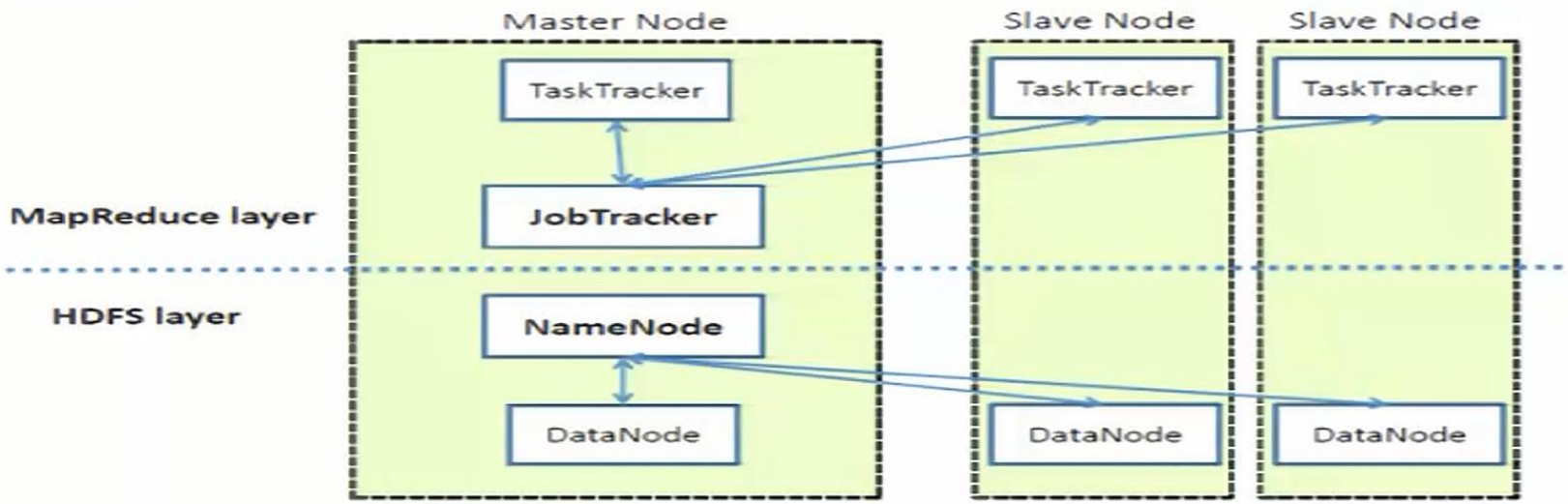
# Apache Framework Basic Modules

- **Hadoop Common:** It contains libraries and utilities needed by other Hadoop modules.
- **Hadoop Distributed File System (HDFS):** It is a distributed file system that stores data on a commodity machine. Providing very high aggregate bandwidth across the entire cluster.
- **Hadoop YARN:** It is a resource management platform responsible for managing compute resources in the cluster and using them in order to schedule users and applications.
- **Hadoop MapReduce:** It is a programming model that scales data across a lot of different processes.

# Apache Framework Basic Modules



# High Level Architecture of Hadoop




- Two major pieces of Hadoop are: Hadoop Distribute the File System and the MapReduce, a parallel processing framework that will map and reduce data. These are both open source and inspired by the technologies developed at Google.

- If we talk about this high level infrastructure, we start talking about things like TaskTrackers and JobTrackers, the NameNodes and DataNodes.

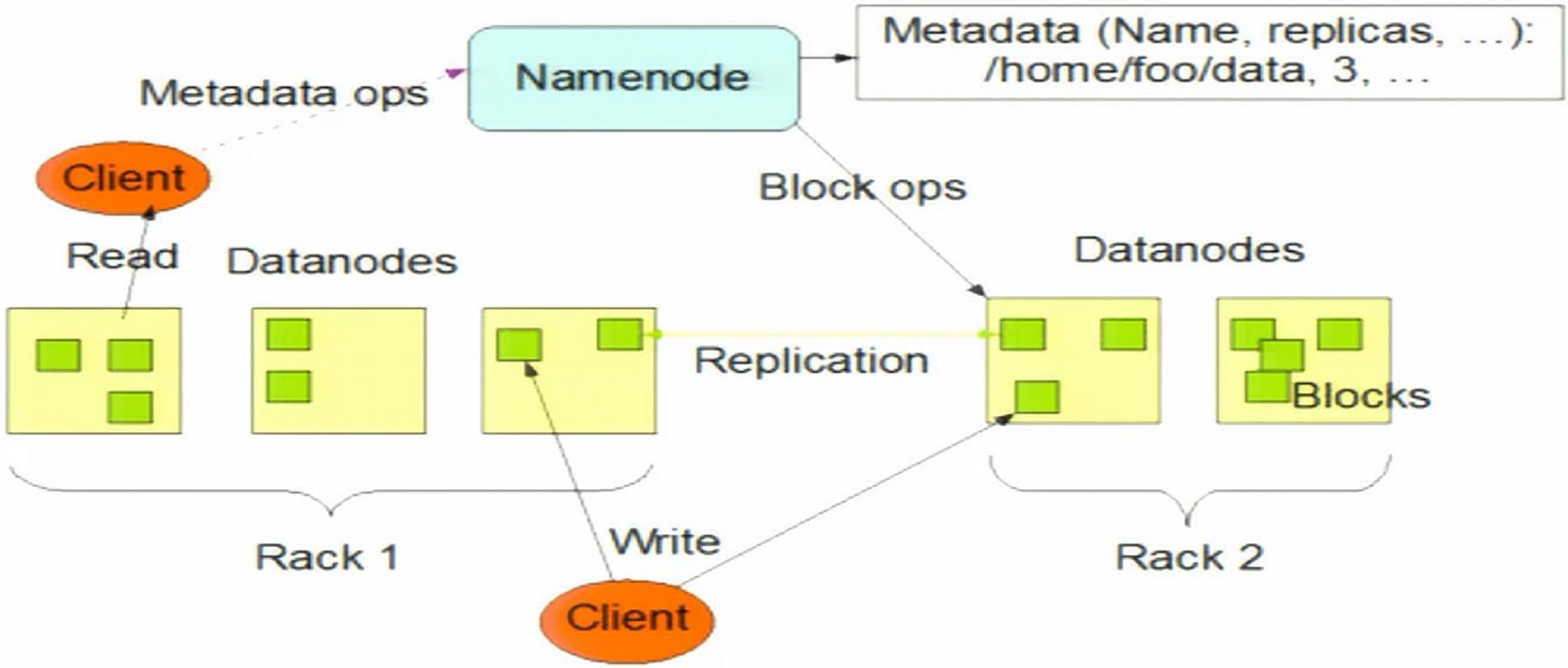
# HDFS: Hadoop Distributed File System

- Distributed, scalable, and portable file-system written in Java for the Hadoop framework.

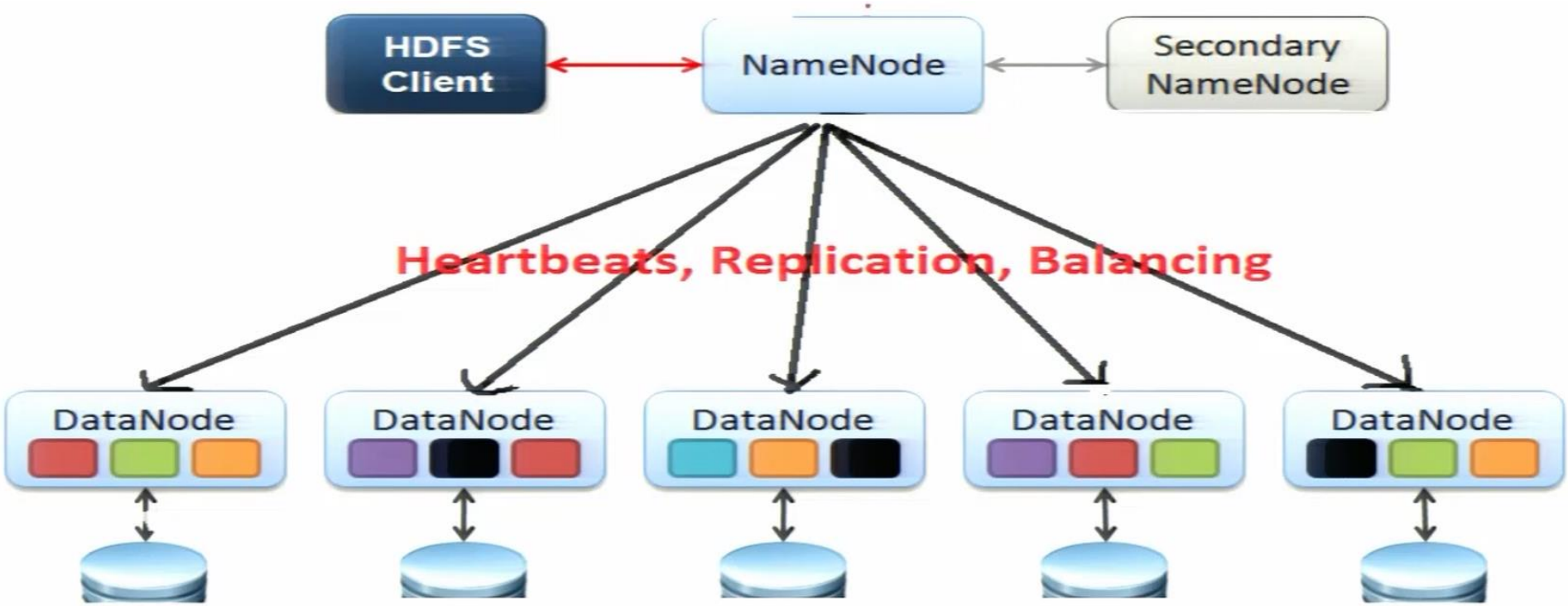
- Each node in Hadoop instance typically has a single name node. and a cluster of data nodes that formed this HDFS cluster.

- Each HDFS stores large files, typically in ranges of gigabytes to terabytes, and now petabytes, across multiple machines. And it can achieve reliability by replicating the cross multiple hosts, and therefore does not require any range storage on hosts.

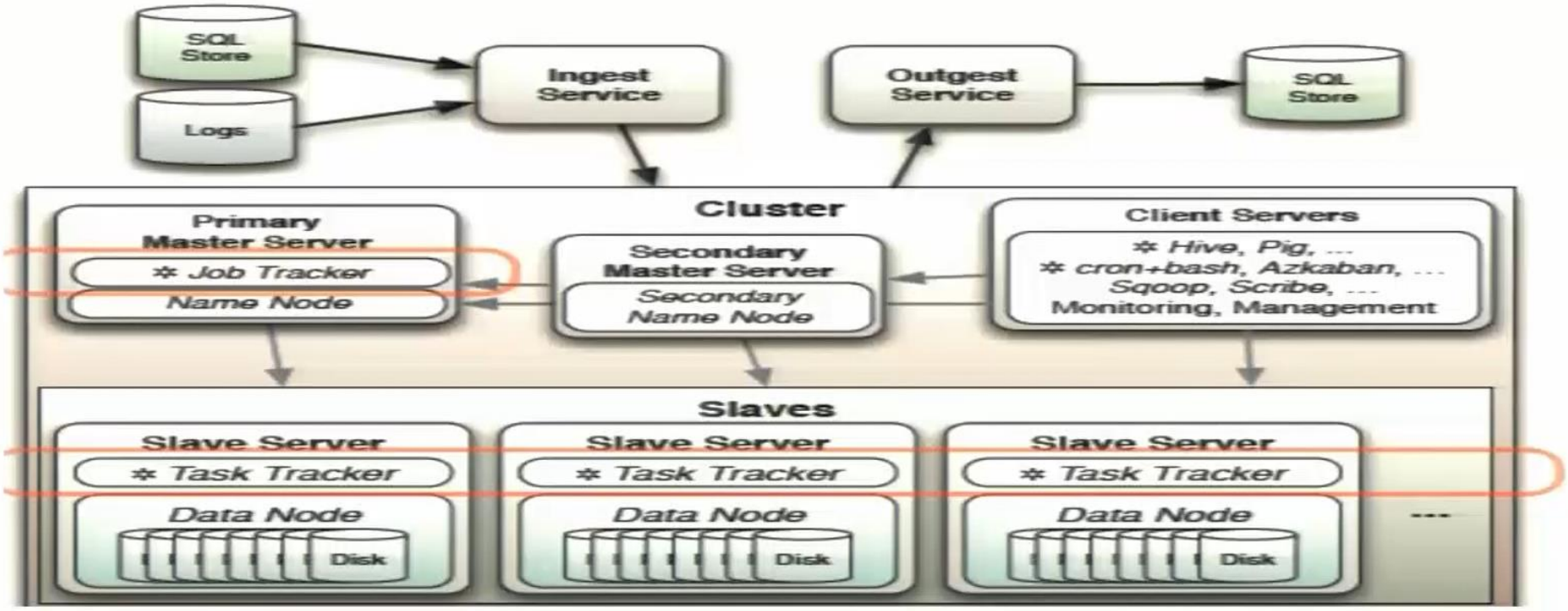
# HDFS


Metadata ops
Namenode
Metadata (Name, replicas, …):
/home/foo/data, 3, …
Client
Block ops
Read
Datanodes
Datanodes
Replication
Blocks
Rack 1
Write
Rack 2
Client

# HDFS


HDFS Client
NameNode
Secondary NameNode
Heartbeats, Replication, Balancing
DataNode
DataNode
DataNode
DataNode
DataNode

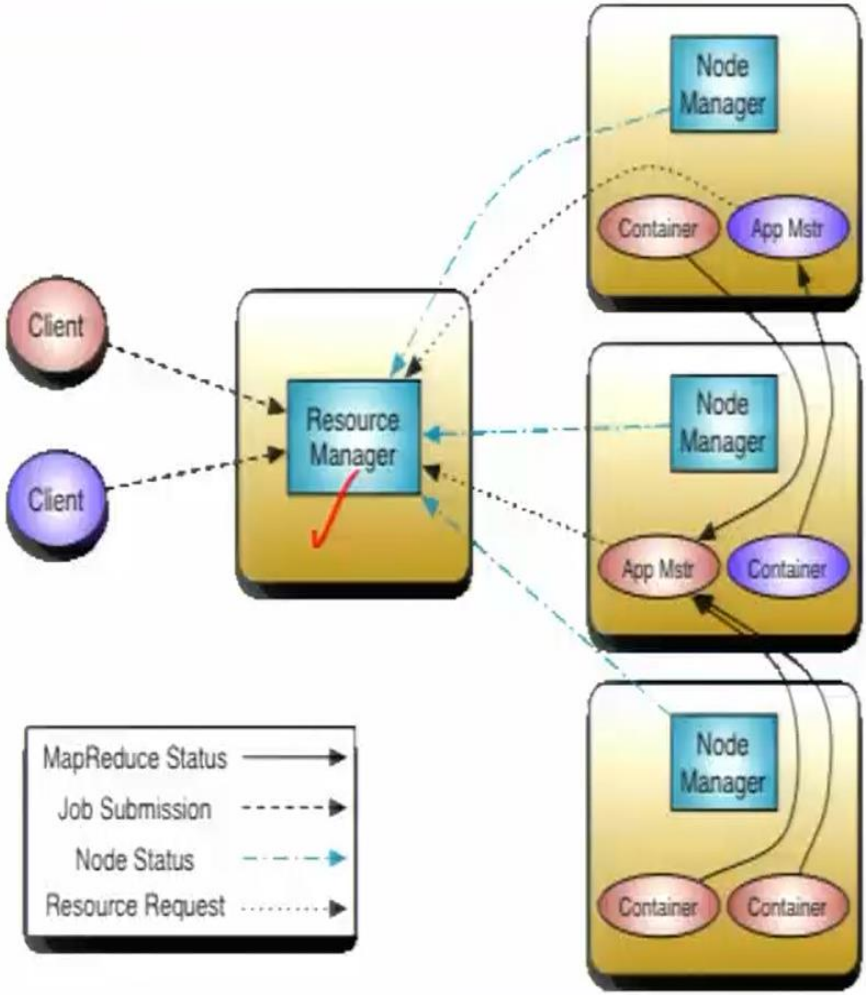
# MapReduce Engine




- The typical MapReduce engine will consist of a job tracker, to which client applications can submit MapReduce jobs, and this job tracker typically pushes work out to all the available task trackers, now it's in the cluster. Struggling to keep the word as close to the data as possible, as balanced as possible.

# Apache Hadoop MapReduce Next Gen(YARN)

- Yarn enhances the power of the Hadoop compute cluster, without being limited by the map produce kind of framework.
- It's scalability's great. The processing power and data centers continue to grow quickly, because the YARN resource manager focuses exclusively on scheduling. It can manage those very large clusters quite quickly and easily.
- YARN is completely compatible with the MapReduce. Existing MapReduce application end users can run on top of the Yarn without disrupting any of their existing processes.



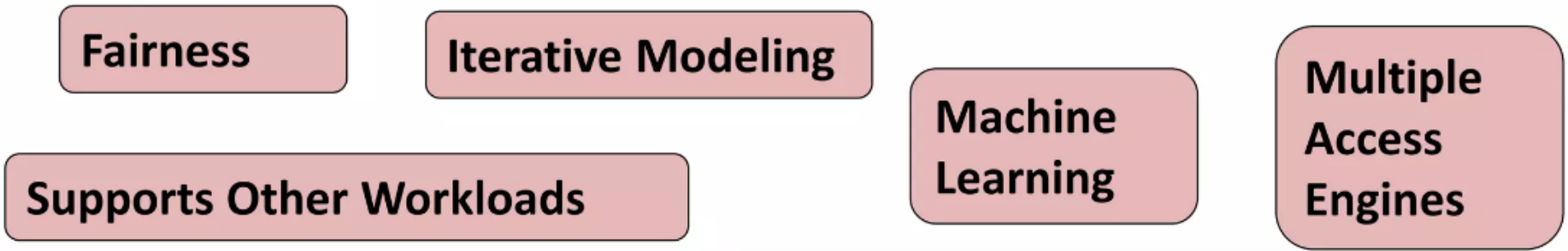
# What is YARN?

- Yarn enhances the power of the Hadoop compute cluster, without being limited by the map produce kind of framework.
- It's scalability's great. The processing power and data centers continue to grow quickly, because the YARN research manager focuses exclusively on scheduling. It can manage those very large clusters quite quickly and easily.
- YARN is completely compatible with the MapReduce. Existing MapReduce application end users can run on top of the Yarn without disrupting any of their existing processes.
- It does have a Improved cluster utilization as well. The resource manager is a pure schedule or they just optimize this cluster utilization according to the criteria such as capacity, guarantees, fairness, how to be fair, maybe different SLA's or service level agreements.

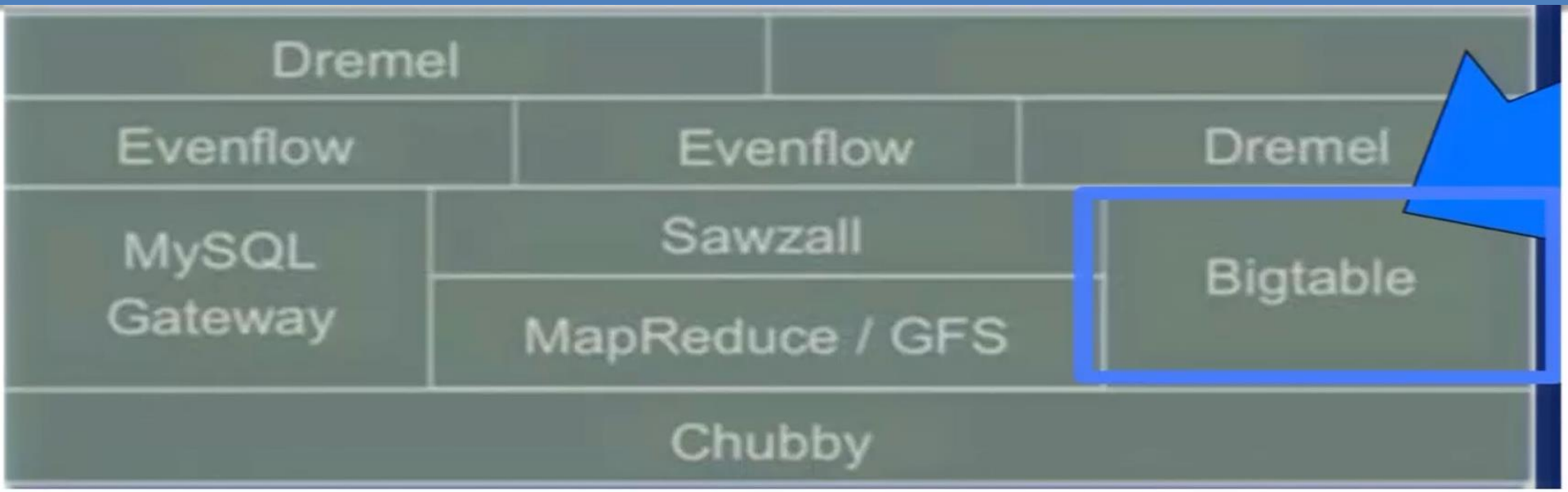
**Scalability** | **MapReduce Compatibility** | **Improved cluster utilization**

# What is YARN?

- It supports other work flows other than just map reduce.
- Now we can bring in additional programming models, such as graph process or iterative modeling, and now it's possible to process the data in your base. This is especially useful when we talk about machine learning applications.
- Yarn allows multiple access engines, either open source or proprietary, to use Hadoop as a common standard for either batch or interactive processing, and even real time engines that can simultaneous acts as a lot of different data, so you can put streaming kind of applications on top of YARN inside a Hadoop architecture, and seamlessly work and communicate between these environments.

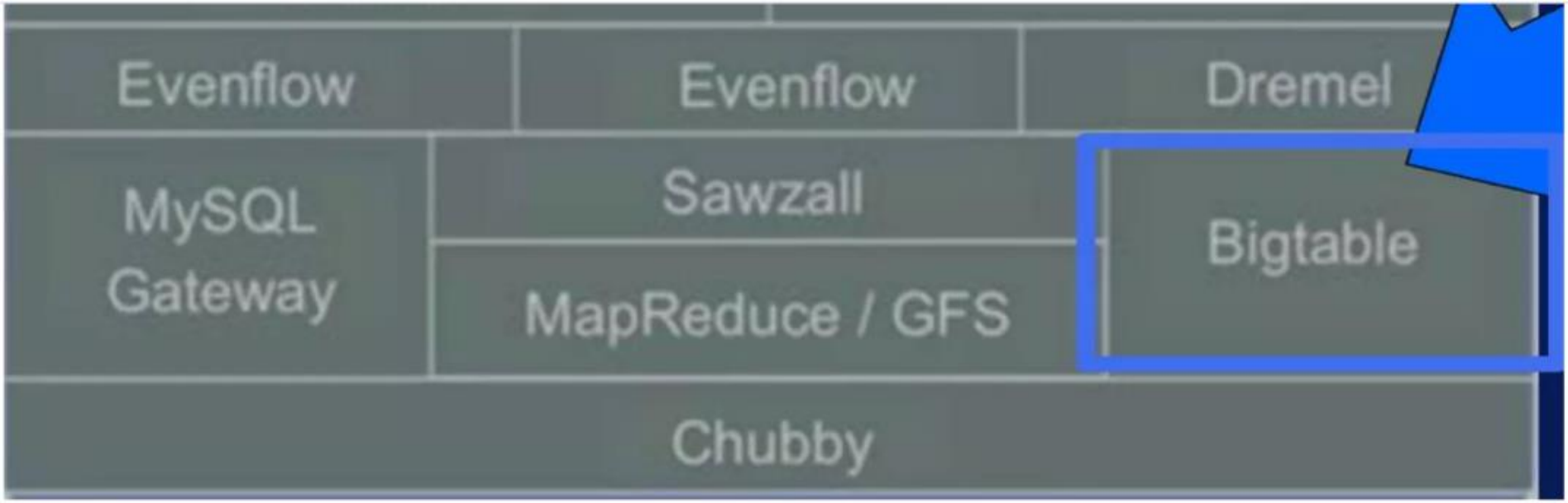


# Original Google Stack



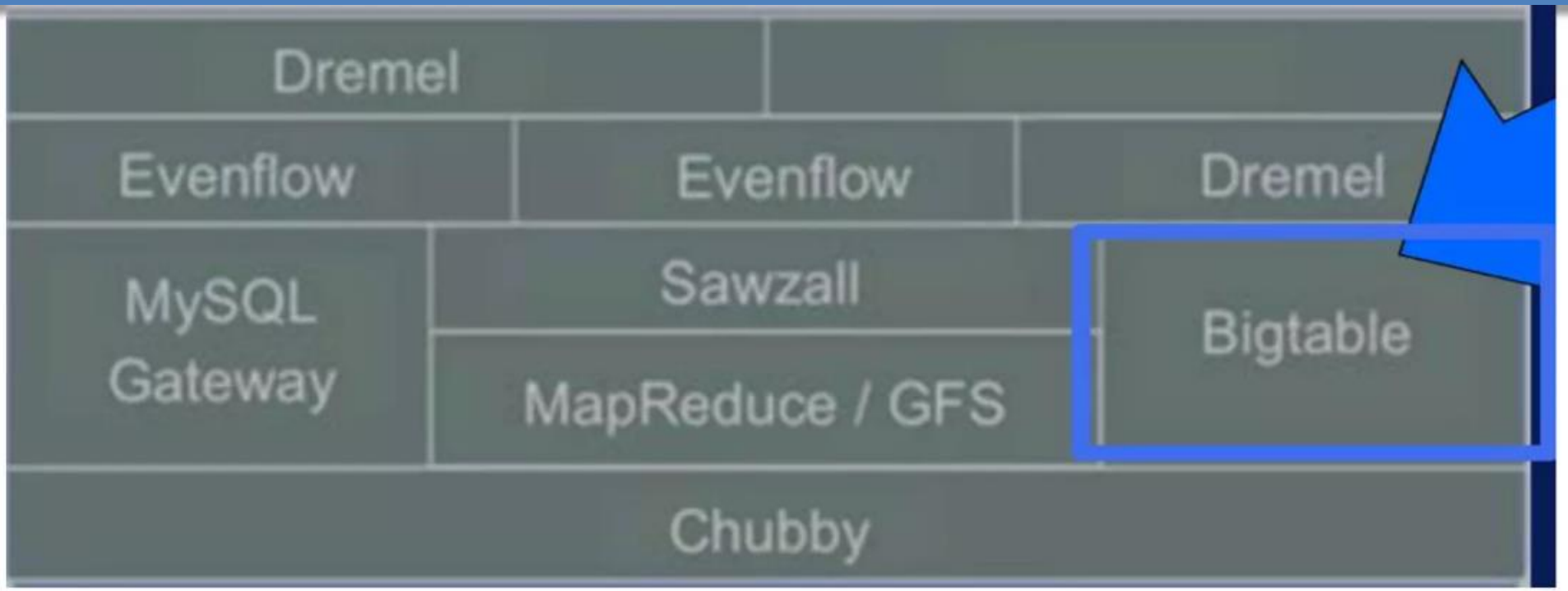

- Had their original MapReduce, and they were storing and processing large amounts of data.
- Like to be able to access that data and access it in a SQL like language. So they built the SQL gateway to adjust the data into the MapReduce cluster and be able to query some of that data as well.

# Original Google Stack



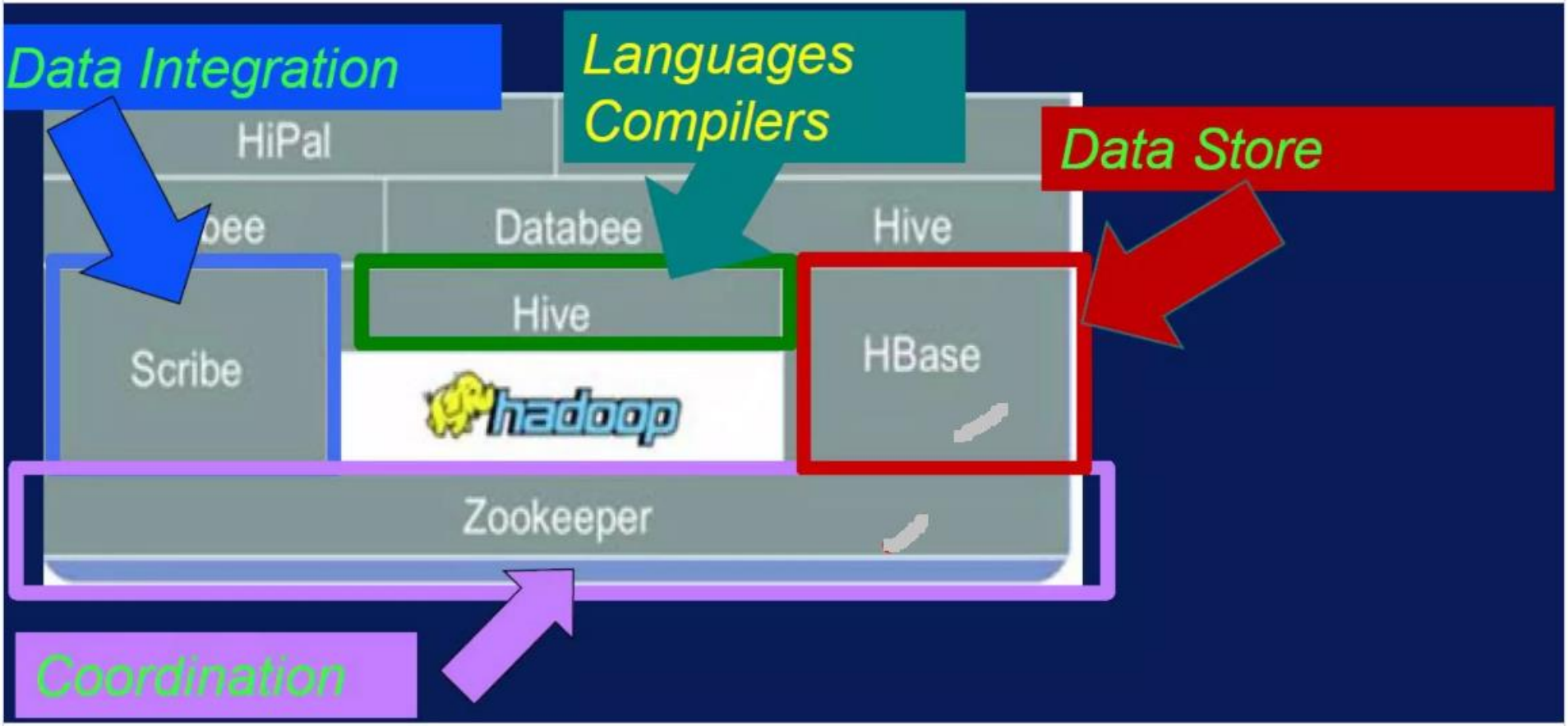

- Then, they realized they needed a high-level specific language to access MapReduce in the cluster and submit some of those jobs. So Sawzall came along.
- Then, Evenflow came along and allowed to chain together complex work codes and coordinate events and service across this kind of a framework or the specific cluster they had at the time.

# Original Google Stack

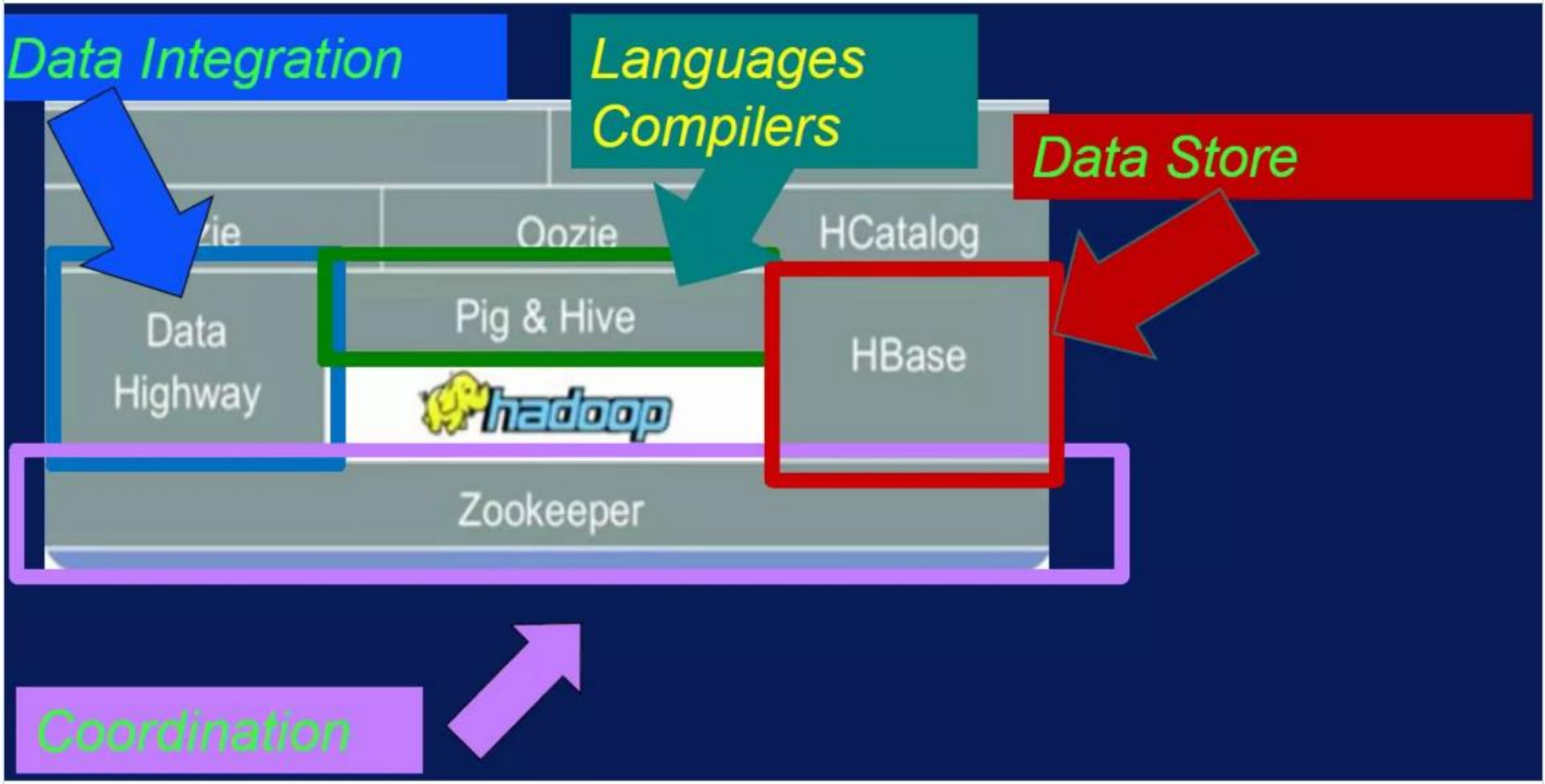



- Then, Dremel came along. Dremel was a columnar storage in the metadata manager that allows us to manage the data and is able to process a very large amount of unstructured data.
- Then Chubby came along as a coordination system that would manage all of the products in this one unit or one ecosystem that could process all these large amounts of structured data seamlessly.

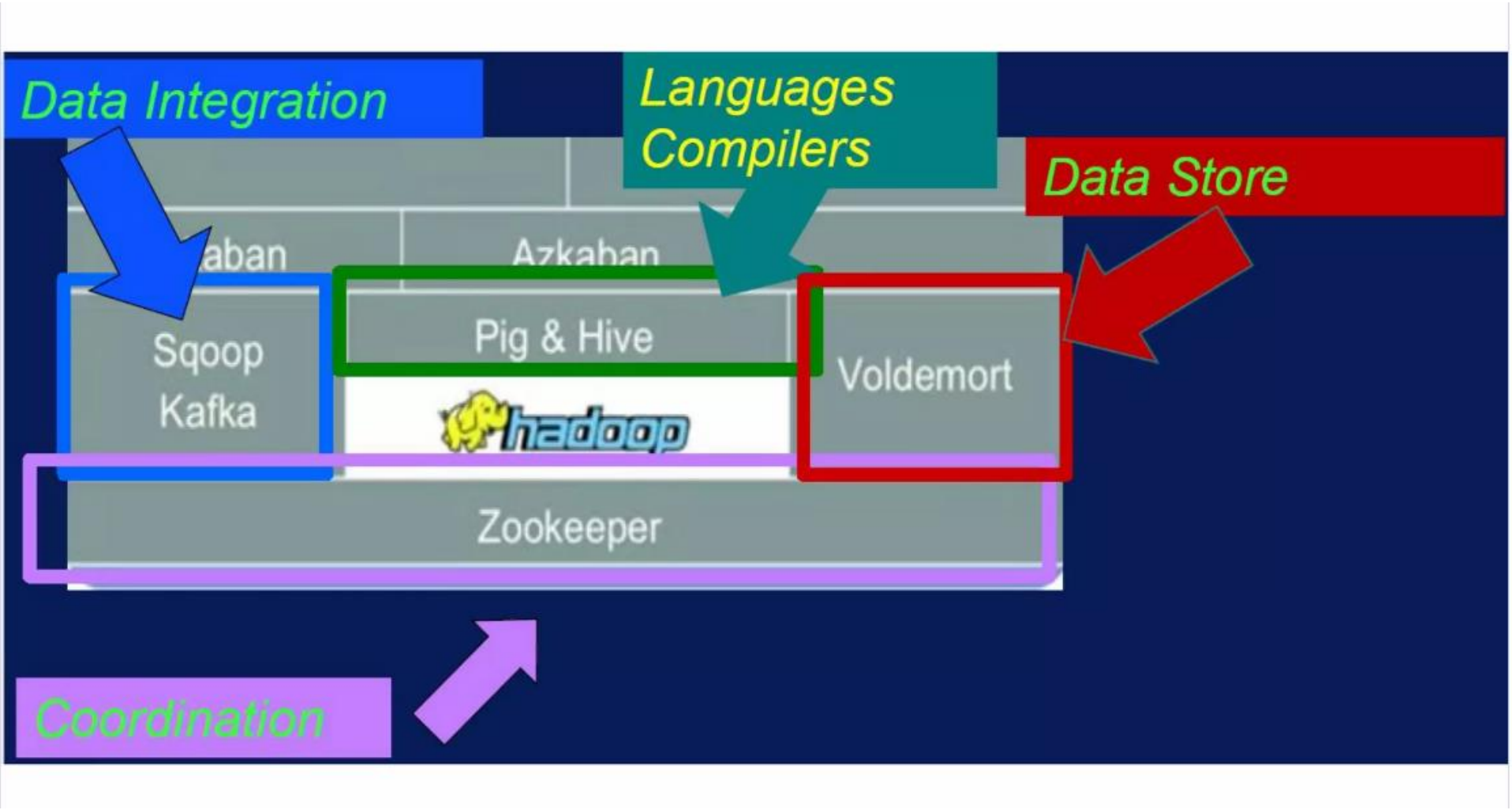
# Facebook's Version of Stack



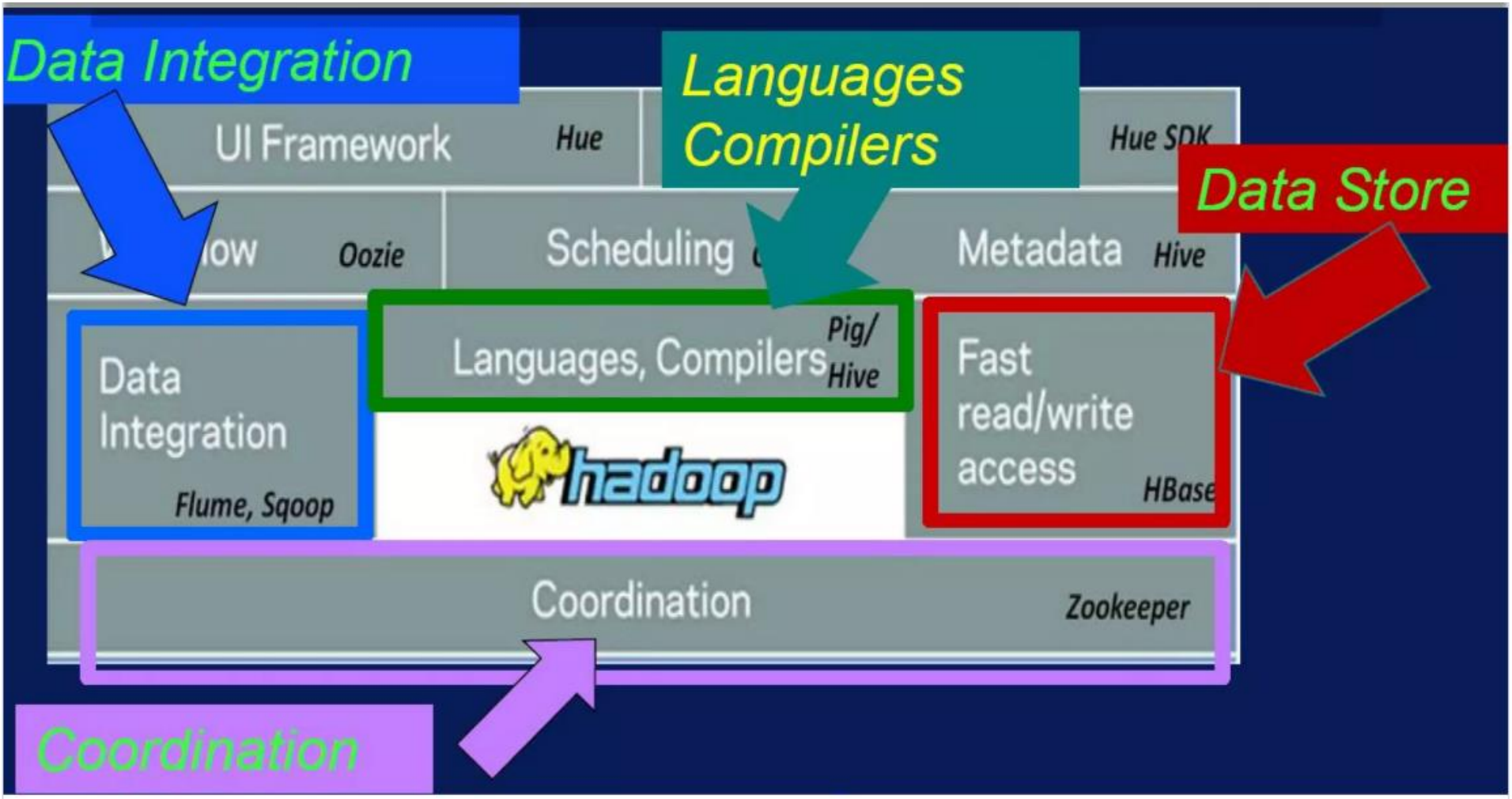
# Yahoo's Version of Stack



# Linkedin's Version of Stack



# Cloudera's Version of Stack



# Apache Hadoop Ecosystem

**Ambari**
Provisioning, Managing and Monitoring Hadoop Clusters

**Sqoop**
Data Exchange

**Flume**
Log Collector

**Zookeeper**
Coordination

**Oozie**
Workflow

**Pig**
Scripting

**Mahout**
Machine Learning

**R Connectors**
Statistics

**Hive**
SQL Query

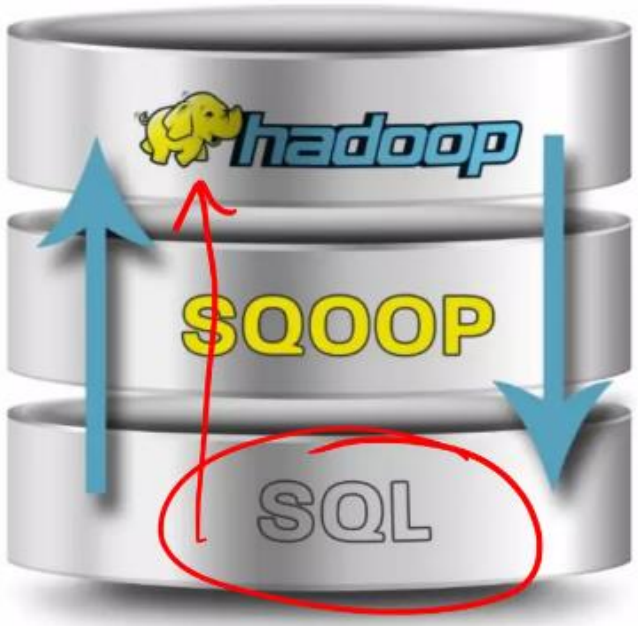
**Hbase**
Columnar Store

**YARN Map Reduce v2**
Distributed Processing Framework

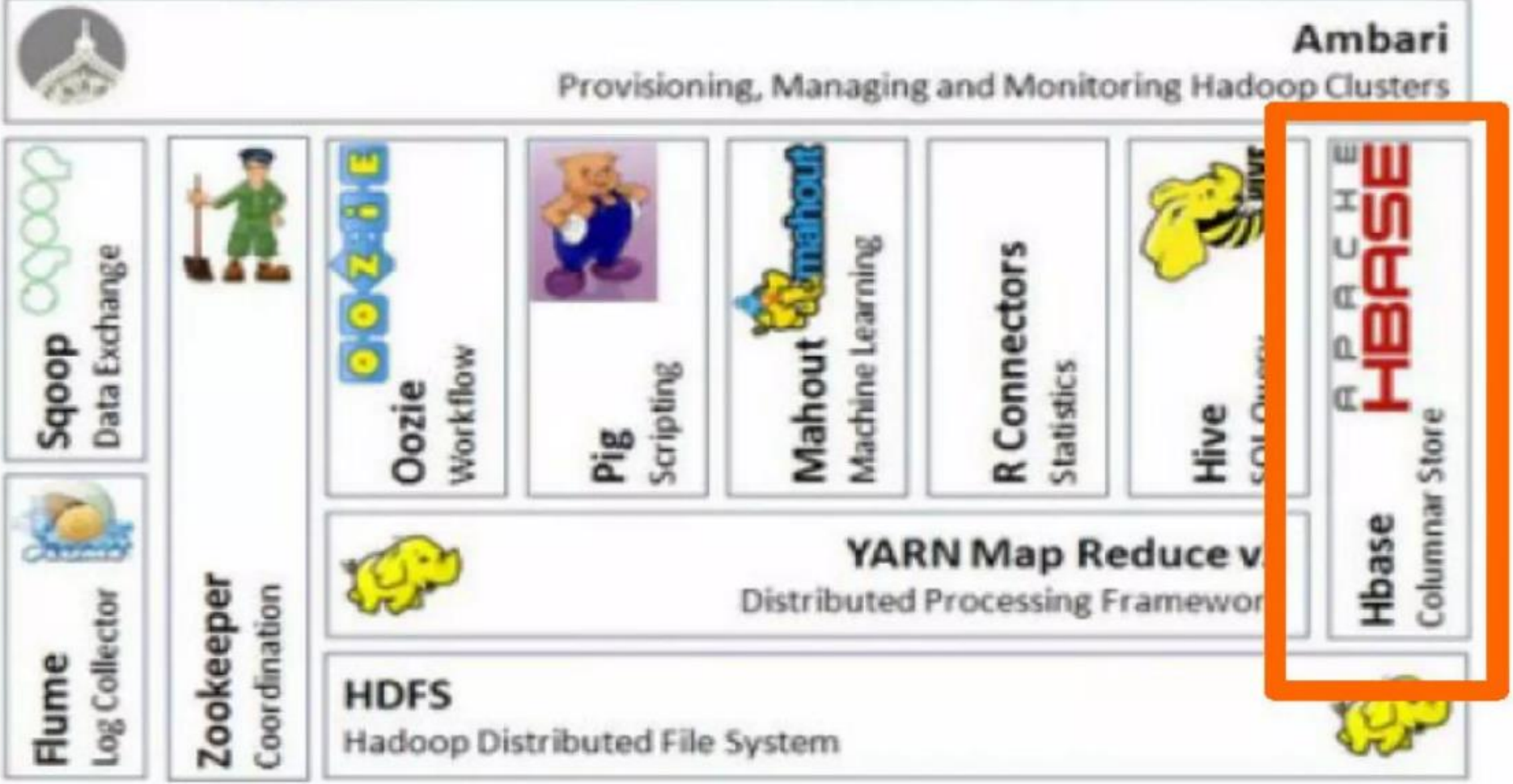
**HDFS**
Hadoop Distributed File System

# Apache Sqoop

- Tool designed for efficiently transferring bulk data between Apache Hadoop and structured datastores such as relational databases



# Apache Hadoop Ecosystem



# HBASE

- Hbase is a key component of the Hadoop stack, as its design caters to applications that require really fast random access to significant data set.

- Column-oriented database management system
- Key-value store
- Based on Google Big Table
- Can hold extremely large data
- Dynamic data model
- Not a Relational DBMS

# Apache Hadoop Ecosystem

**Ambari**
Provisioning, Managing and Monitoring Hadoop Clusters

**Sqoop**
Data Exchange

**Flume**
Log Collector

**Zookeeper**
Coordination

**Oozie**
Workflow

**Pig**
Scripting

**Mahout**
Machine Learning

**R Connectors**
Statistics

**Hive**
SQL Query

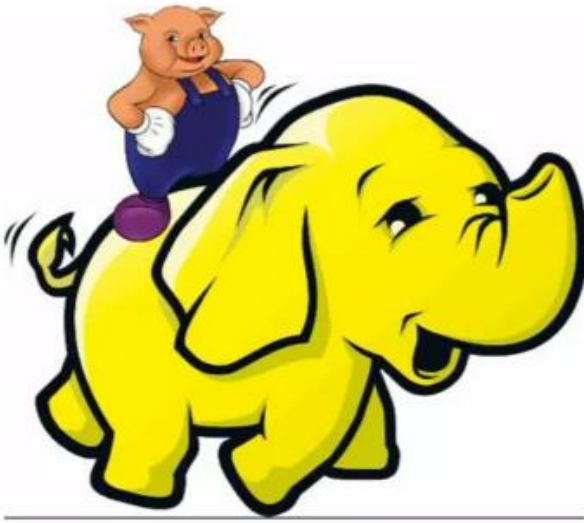
**Hbase**
Columnar Store

**YARN Map Reduce v2**
Distributed Processing Framework

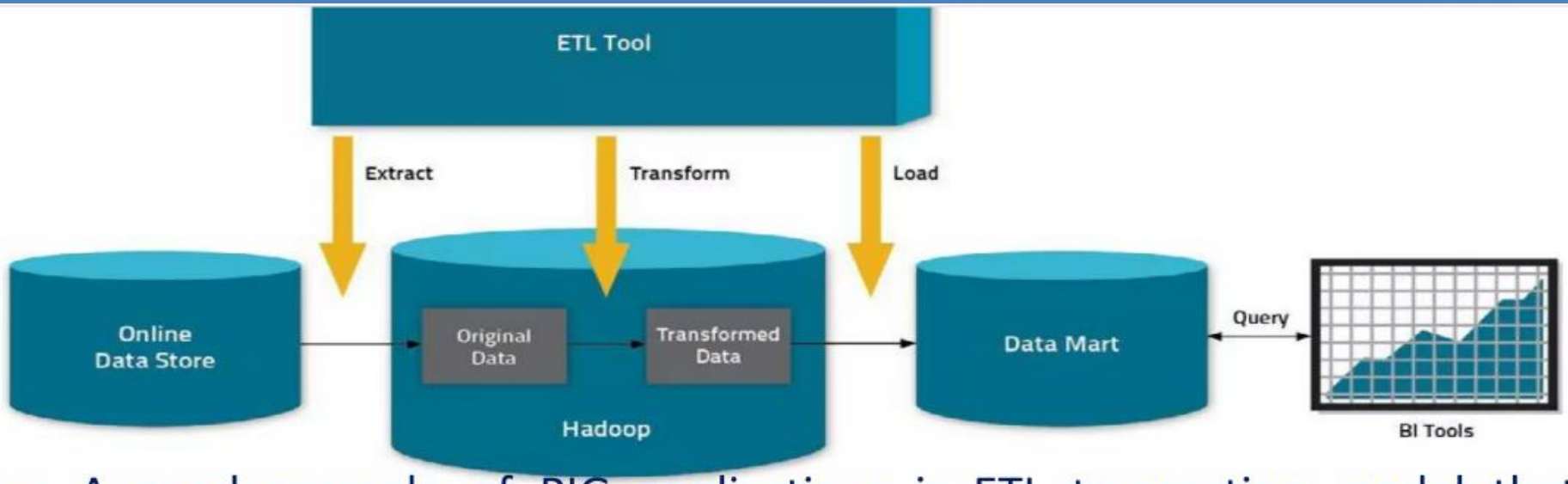
**HDFS**
Hadoop Distributed File System

# PIG

- High level programming on top of Hadoop MapReduce
- The language: Pig Latin
- Data analysis problems as data flows
- Originally developed at Yahoo 2006



# PIG for ETL




- A good example of PIG applications is ETL transaction model that describes how a process will extract data from a source, transporting according to the rules set that we specify, and then load it into a data store.
- PIG can ingest data from files, streams, or any other sources using the UDF: a user-defined functions that we can write ourselves.
- When it has all the data it can perform, select, iterate and do kinds of transformations.

# Apache Hadoop Ecosystem

**Ambari**
Provisioning, Managing and Monitoring Hadoop Clusters

**Sqoop**
Data Exchange

**Flume**
Log Collector

**Zookeeper**
Coordination

**Oozie**
Workflow

**Pig**
Scripting

**Mahout**
Machine Learning

**R Connectors**
Statistics

**Hive**
SQL Query

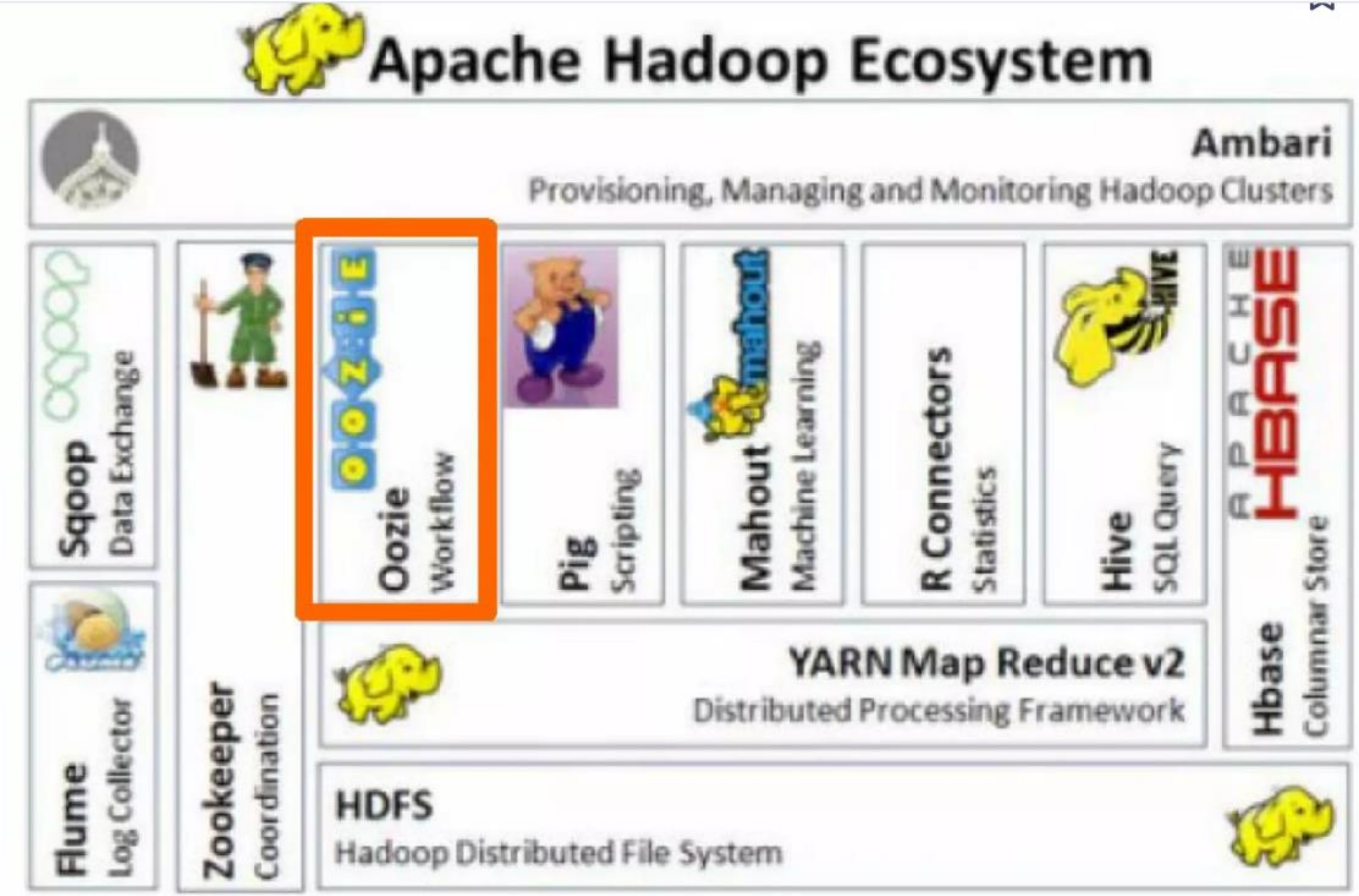
**Hbase**
Columnar Store

**YARN Map Reduce v2**
Distributed Processing Framework

**HDFS**
Hadoop Distributed File System

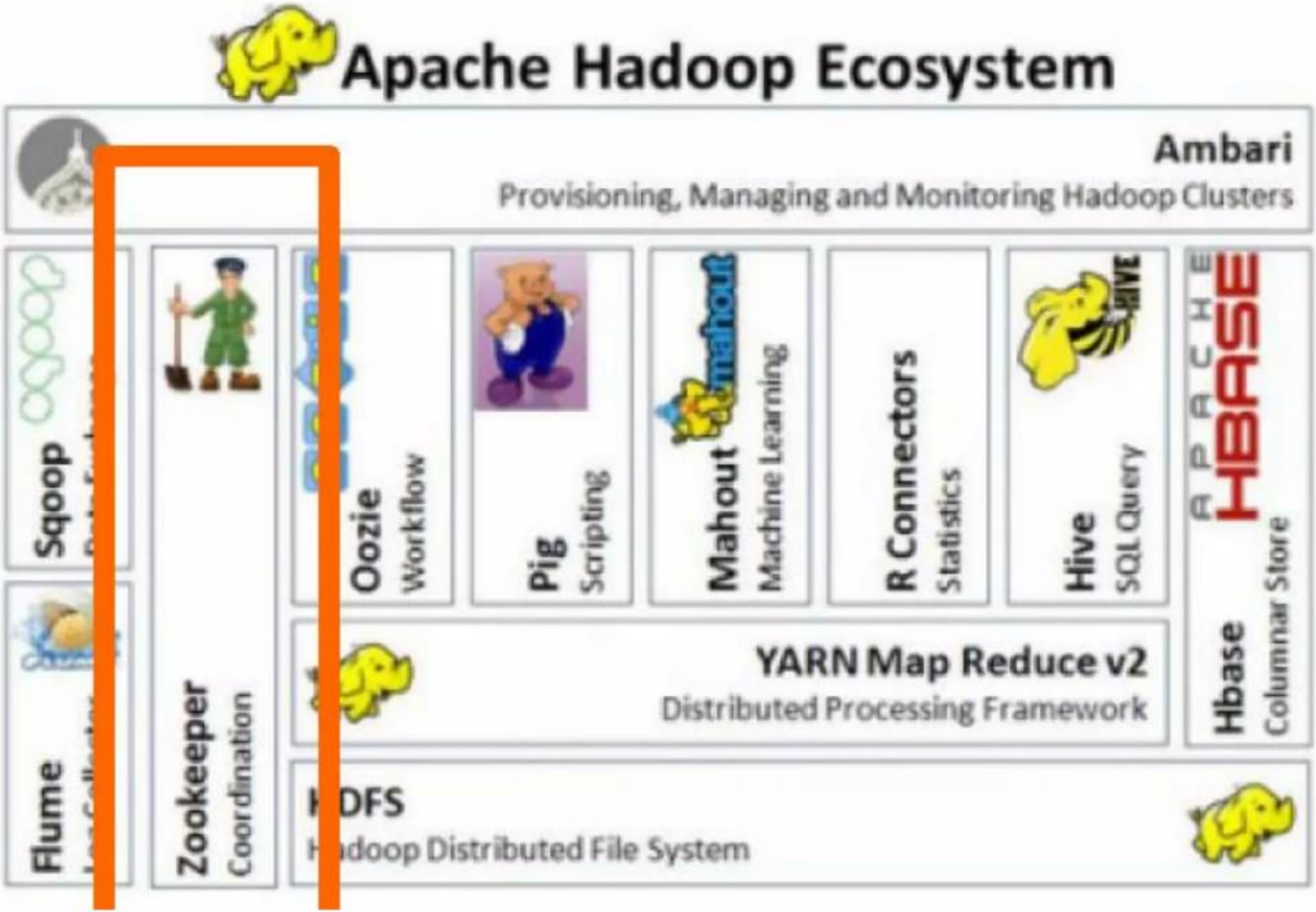
# Apache HIV

- Data warehouse software facilitates querying and managing large datasets residing in distributed storage
- SQL-like language!
- Facilitates querying and managing large datasets in HDFS
- Mechanism to project structure onto this data and query the data using a SQL-like language called HiveQL



Apache Hadoop Ecosystem
Ambari
Provisioning, Managing and Monitoring Hadoop Clusters
Sqoop
Data Exchange
Flume
Log Collector
Zookeeper
Coordination
Oozie
Workflow
Pig
Scripting
Mahout
Machine Learning
R Connectors
Statistics
Hive
SQL Query
Hbase
Columnar Store
YARN Map Reduce v2
Distributed Processing Framework
HDFS
Hadoop Distributed File System

# Oozie


- Workflow scheduler system to manage Apache Hadoop jobs
- Oozie Coordinator jobs!
- Supports MapReduce, Pig, Apache Hive, and Sqoop, etc.

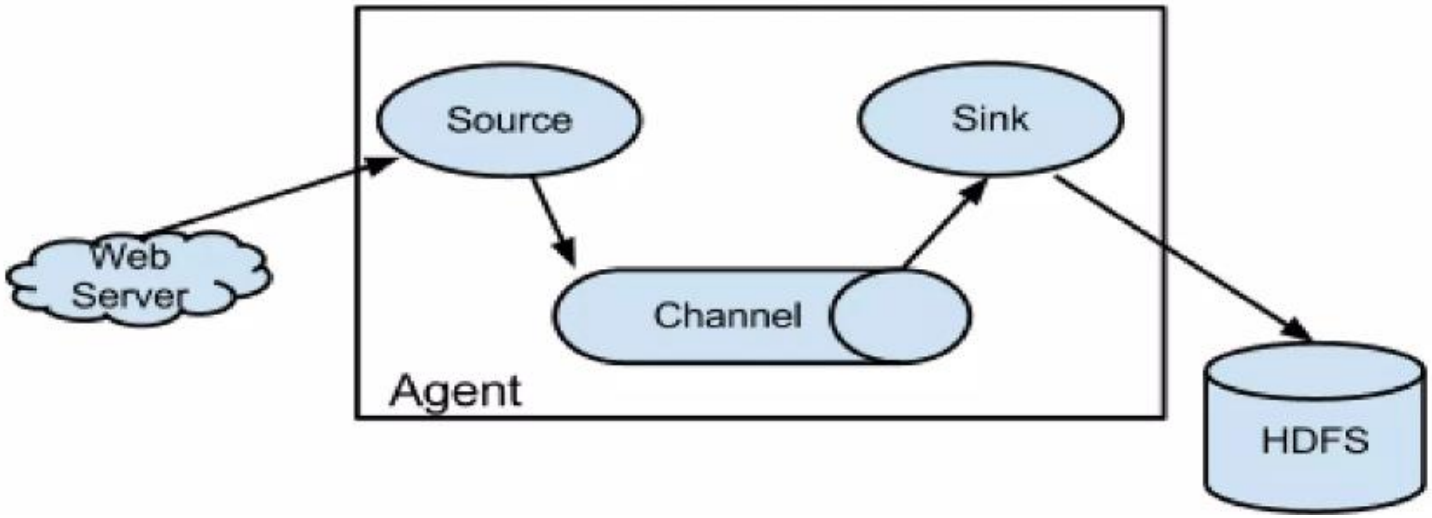

Apache Hadoop Ecosystem
Ambari
Provisioning, Managing and Monitoring Hadoop Clusters
Sqoop
Zookeeper
Coordination
Oozie
Workflow
Pig
Scripting
Mahout
Machine Learning
R Connectors
Statistics
Hive
SQL Query
APACHE HBASE
Hbase
Columnar Store
Flume
YARN Map Reduce v2
Distributed Processing Framework
HDFS
Hadoop Distributed File System

# Zookeeper

- Provides operational services for a Hadoop cluster group services
- Centralized service for: maintaining configuration information naming services
- Providing distributed synchronization and providing group services

# Flume

- Distributed, reliable, and available service for efficiently collecting, aggregating, and moving large amounts of log data
- It has a simple and very flexible architecture based on streaming data flows. It's quite robust and fall tolerant, and it's really tunable to enhance the reliability mechanisms, fail over, recovery, and all the other mechanisms that keep the cluster safe and reliable.
- It uses simple extensible data model that allows us to apply all kinds of online analytic applications.

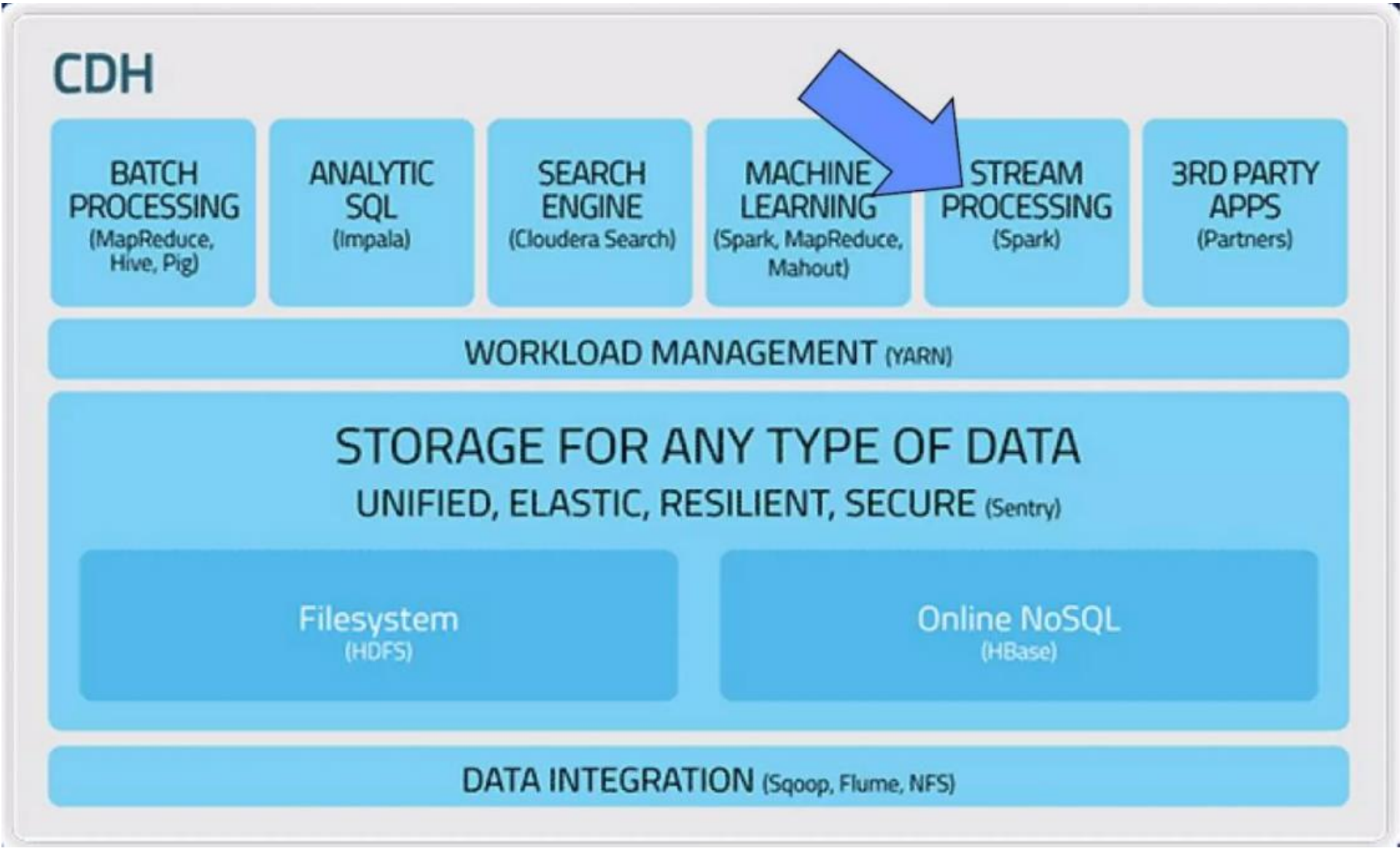



# Additional Cloudera Hadoop Components; Impala

# Impala

- Cloudera, Impala was designed specifically at Cloudera, and it's a query engine that runs on top of the Apache Hadoop. The project was officially announced at the end of 2012, and became a publicly available, open source distribution.
- Impala brings scalable parallel database technology to Hadoop and allows users to submit low latencies queries to the data that's stored within the HDFS or the Hbase without acquiring a ton of data movement and manipulation.
- Impala is integrated with Hadoop, and it works within the same power system, within the same format metadata, all the security and reliability resources and management workflows.
- It brings that scalable parallel database technology on top of the Hadoop. It actually allows us to submit SQL like queries at much faster speeds with a lot less latency.

# Additional Cloudera Hadoop Components Spark



# Spark

- Apache Spark™ is a fast and general engine for large-scale data processing
- Spark is a scalable data analytics platform that incorporates primitives for in-memory computing and therefore, is allowing to exercise some different performance advantages over traditional Hadoop's cluster storage system approach. And it's implemented and supports something called Scala language, and provides unique environment for data processing.
- Spark is really great for more complex kinds of analytics, and it's great at supporting machine learning libraries.
- It is yet again another open source computing frame work and it was originally developed at MP labs at the University of California Berkeley and it was later donated to the Apache software foundation where it remains today as well.

# Spark Benifits

- In contrast to Hadoop's two stage disk based MapReduce paradigm Multi-stage in-memory primitives provides performance up to 100 times faster for certain applications.
- Allows user programs to load data into a cluster's memory and query it repeatedly
- Spark is really well suited for these machined learning kinds of applications that often times have iterative sorting in memory kinds of computation.
- Spark requires a cluster management and a distributed storage system. So for the cluster management, Spark supports standalone native Spark clusters, or you can actually run Spark on top of a Hadoop yarn, or via patching mesas.
- For distributor storage, Spark can interface with any of the variety of storage systems, including the HDFS, Amazon S3.