White Paper

Big Data
Executive Overview

WP-BD-10312014-01

By

Jafar Shunnar & Dan Raver

ITG Software Engineering

Big Data

Table of Contents

Section 01	Big Data Facts	Page 3-4
Section 02	What is Big Data?	Page 5-6
Section 03	What is Driving Big Data?	Page 7-8
Section 04	Big Data History	Page 9-9
Section 05	Apache Open Source Project (Hadoop)	Page 10-11
Section 06	Big Data Design Principles	Page 12-14

Section 01

Big Data Facts

The amount of data created in the modern digital world is truly astounding.

Here are some facts and figures that illustrate the point:

The volume of data created by U.S. companies alone each year is enough to fill ten thousand Libraries of Congress.

According to McKinsey – a retailer using big data to the full could increase its operating margin by more than 60%.

Zuckerberg noted that 1 billion pieces of content are shared via Facebook's Open Graph daily.

Google's Eric Schmidt claims that every two days now we create as much information as we did from the dawn of civilization up until 2003.

According to Gartner Big data will drive $232 billion in spending through 2016.

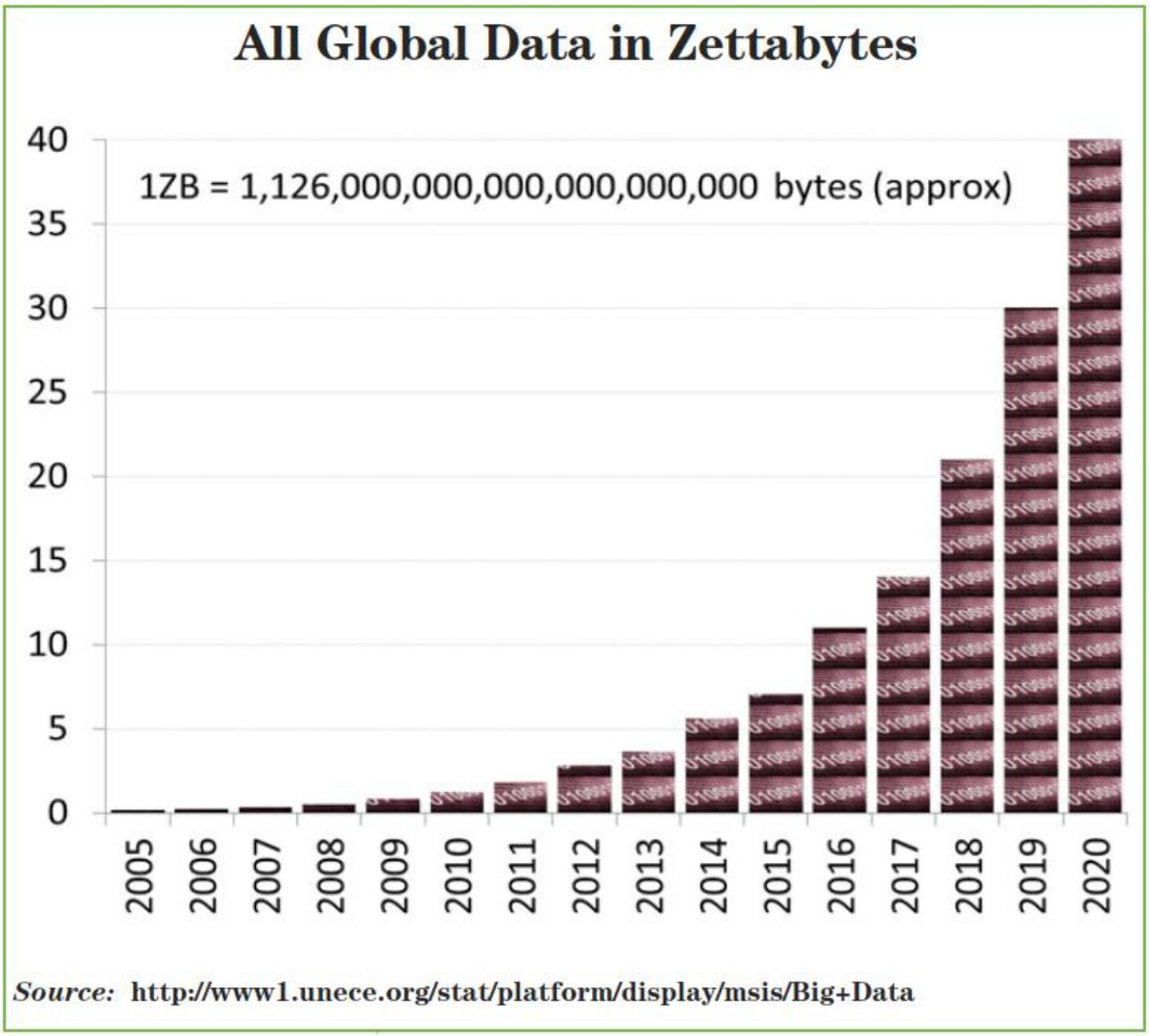
70% of data is created by individuals – but enterprises are responsible for storing and managing 80% of it.

By 2020 one third of all data will be stored, or will have passed through the cloud, and we will have created 35 zetabytes worth of data.

Big Data is set to create 1.9M IT Jobs in U.S. By 2015

Section 01

Big Data Facts



Global Data Graph

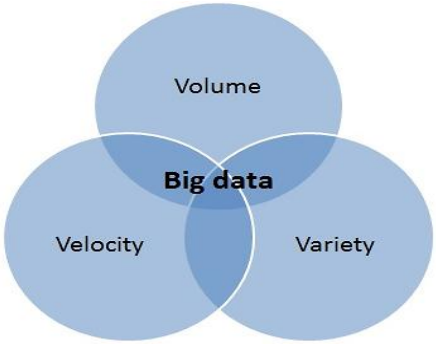
The chart above says it all, from 1 Zettabyte to 40 in 15 years. That is a staggering amount of data, this is truly changing the face of information technology.

Section 02

What is Big Data?

Big data is an all-encompassing term for any collection of data sets so large and complex that it becomes difficult to process them using traditional data processing applications. Big data is defined as any kind of data source that has at least three shared characteristics:

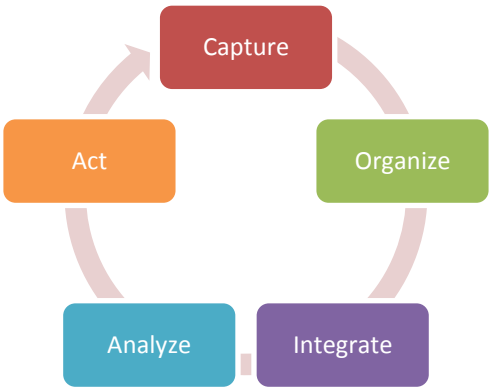
- Volume: How much data.
- Velocity: How fast that data is processed.
- Variety: The various types of data.



The 3v's of Big data

Big data enables organizations to gather, store, manage, and manipulate vast amounts of data at the right speed, and the right time, to gain insights. Big data is an evolution born out of necessity to try and solve specific data management challenges.

The life cycle of Big data



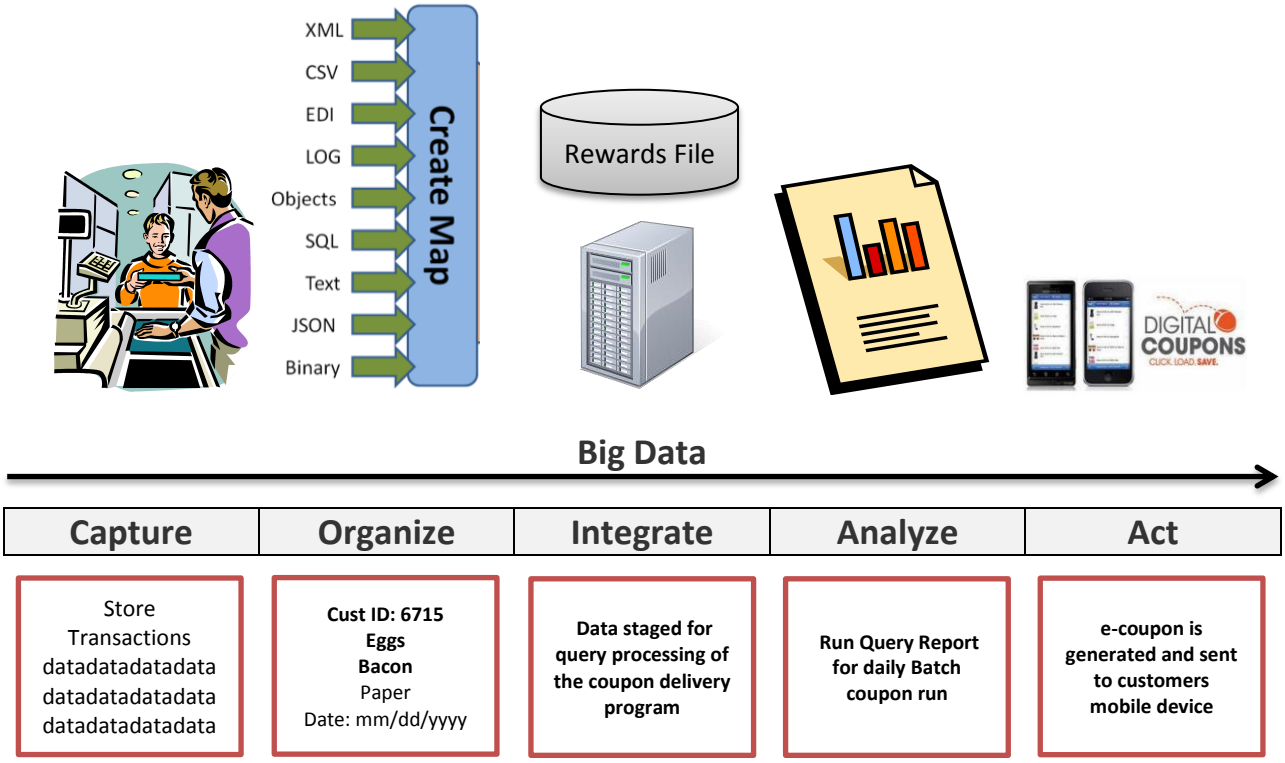
The cycle of Big data management

Section 02

What is Big data?

The life Cycle of Big data; data must first be captured, organized and integrated. Once these phases are completed, data can be analyzed to help solve a specific problem . For example, you may shop at a large grocery chain that has a rewards program. The grocery store captures, organizes, and integrates your purchasing history, and emails coupons to your mobile device.

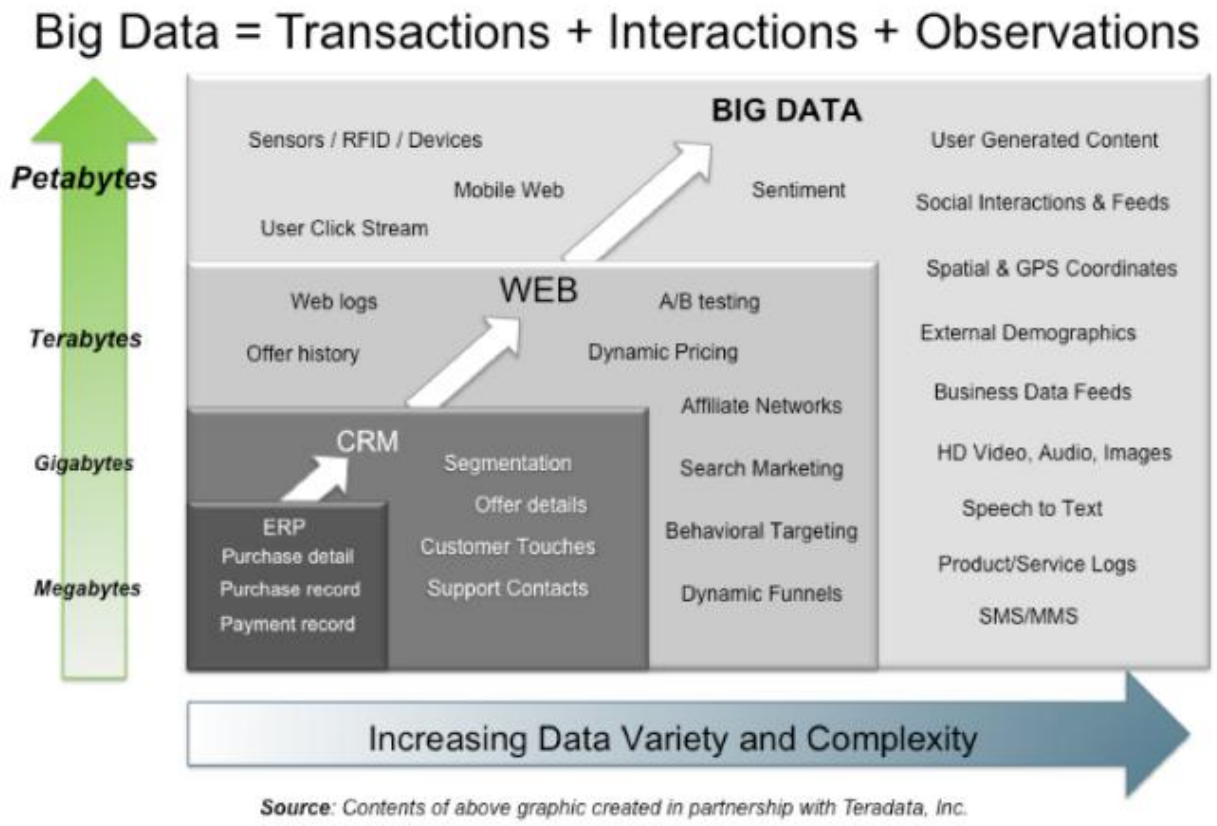
- **Capture:** Your purchasing transactions are captured.
- **Organize:** Information is processed into meaningful data.
- **Integrate:** Information is integrated into the rewards programs.
- **Analyze:** Query programs analyze data.
- **Act:** Reward coupons are emailed to customer.



Big data store loyalty processing

Section 03

What is driving Big data?



***Source**: Contents of above graphic created in partnership with Teradata, Inc.*

The data we are now capturing requires many times the data storage than that of ordinary txt files, combine that with social media and those using mobile devices, this creates an environment for generating very large amounts of data at an astounding rate.

- Social Media
- Videos
- Spatial & GPS Coordinates
- Business Data Feeds
- Product Service logs

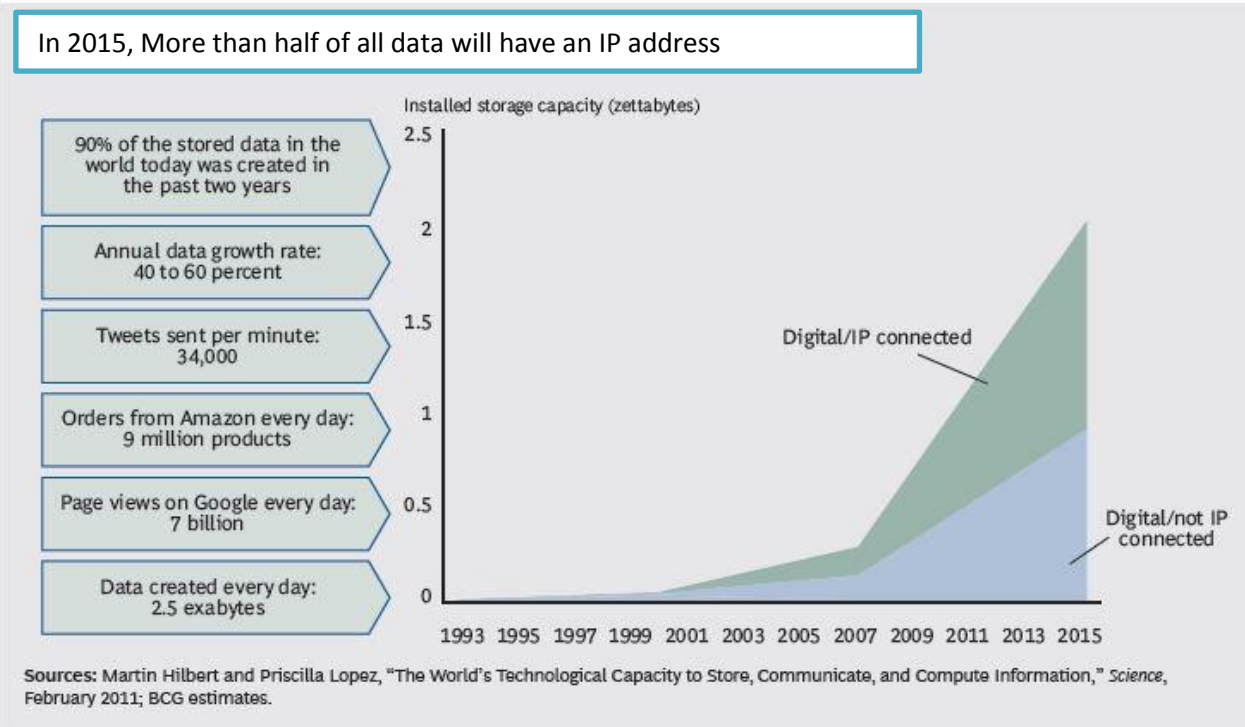
Section 03

What is driving Big Data?

Key drivers behind the Big data market:

- Opportunity to enable new innovative business models.
- Potential for new insight that drive competitive advantage.
- Data collected and stored continues to grow at an astounding rate.
- Data is increasingly everywhere and in many formats.
- Traditional solutions are not keeping up with data demand.
- Cost of data systems as a percentage of IT continues to grow
- Cost of commodity hardware and open source software are enabling new approaches.

These is a new generation of data management technologies that are providing cost effective and innovative approaches for emerging Big Data technologies, such as Apache Hadoop.

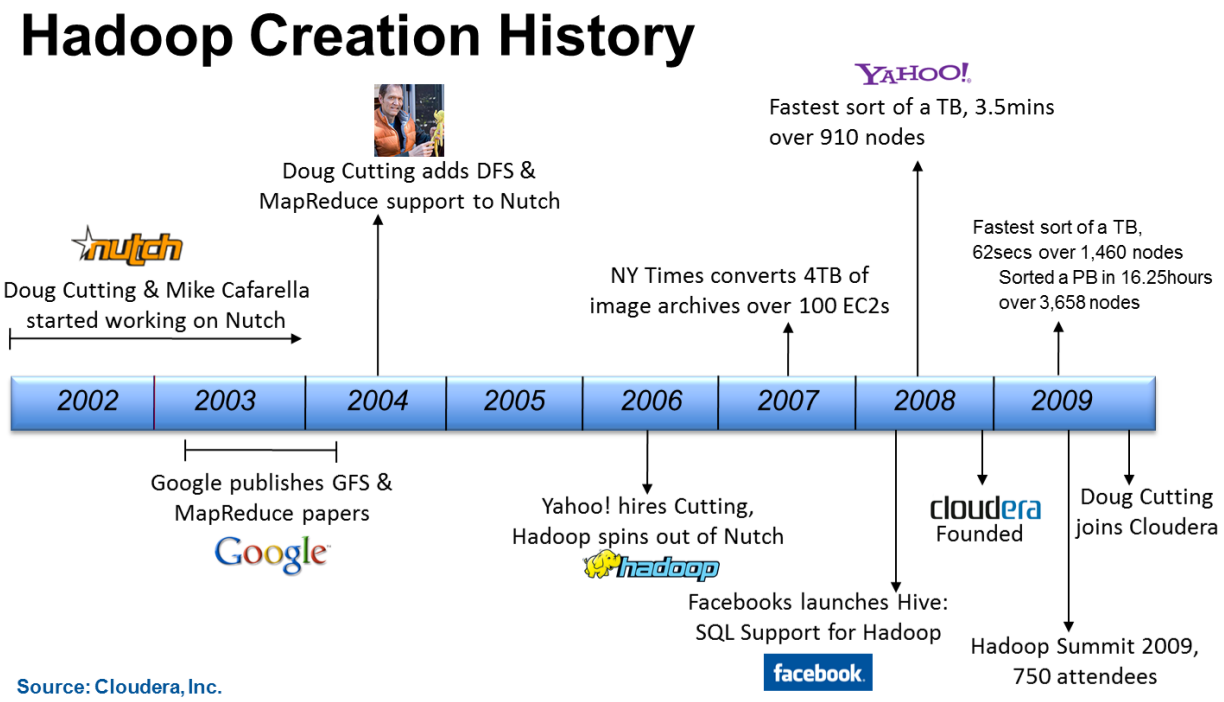


Section 04

Big Data History

In 2000, Seisint Inc. develops C++ based distributed file sharing framework for data storage and querying. Structured, semi-structured and/or unstructured data is stored and distributed across multiple servers. Querying of data is done by modified C++ called ECL which uses apply scheme on read method to create structure of stored data during time of query. In 2004 LexisNexis acquired Seisint Inc. and 2008 acquired ChoicePoint, Inc– and their high speed parallel processing platform. The two platforms were merged into HPCC Systems and in 2011 was open sourced under Apache v2.0 License. Currently HPCC and Quantcast File System–are the only publicly available platforms capable of analyzing multiple exabytes of data.

In 2004, Google published a paper on a process called MapReduce that used such an architecture. The MapReduce framework provides a parallel processing model and associated implementation to process huge amounts of data. With MapReduce, queries are split and distributed across parallel nodes and processed in parallel (the Map step). The results are then gathered and delivered (the Reduce step). The framework was very successful, so others wanted to replicate the algorithm. Therefore, an implementation of the MapReduce framework was adopted by an Apache open source project named Hadoop.



Section 05

Apache Open Source Project (Hadoop)

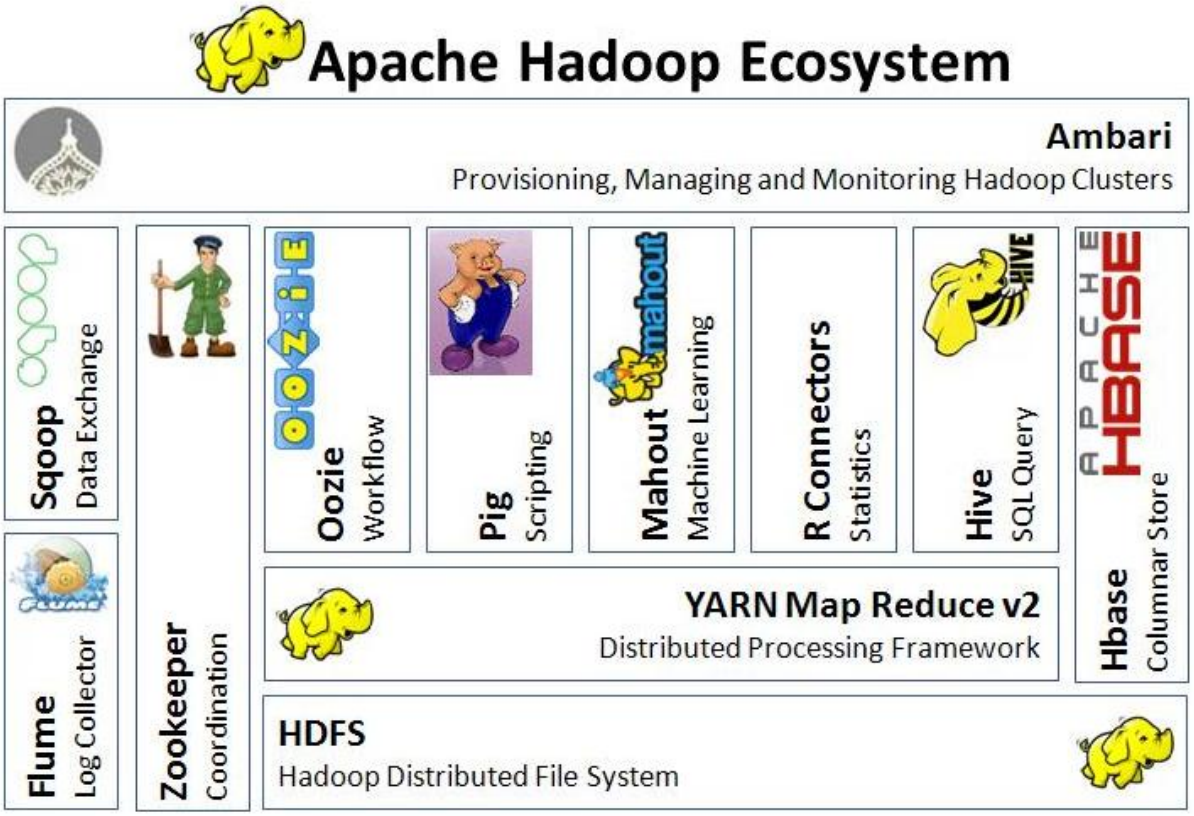
The Apache Hadoop software library is a framework that allows for the distributed processing of large data sets across clusters of computers using simple programming models. It is designed to scale up from single servers to thousands of machines, each offering local computation and storage. Rather than rely on hardware to deliver high-availability, the library itself is designed to detect and handle failures at the application layer, so delivering a highly-available service on top of a cluster of computers, each of which may be prone to failures.

The project includes these modules:

- **Hadoop Common**:
- **Hadoop Distributed File System (HDFS™)**
- **Hadoop YARN**
- **Hadoop MapReduce**

Section 05

Apache Open Source Project (Hadoop)



Although Hadoop is best known for MapReduce and its distributed file system (HDFS, renamed from NDFS), the term is also used for a family of related projects that fall under the umbrella of infrastructure for distributed computing and large-scale data processing.

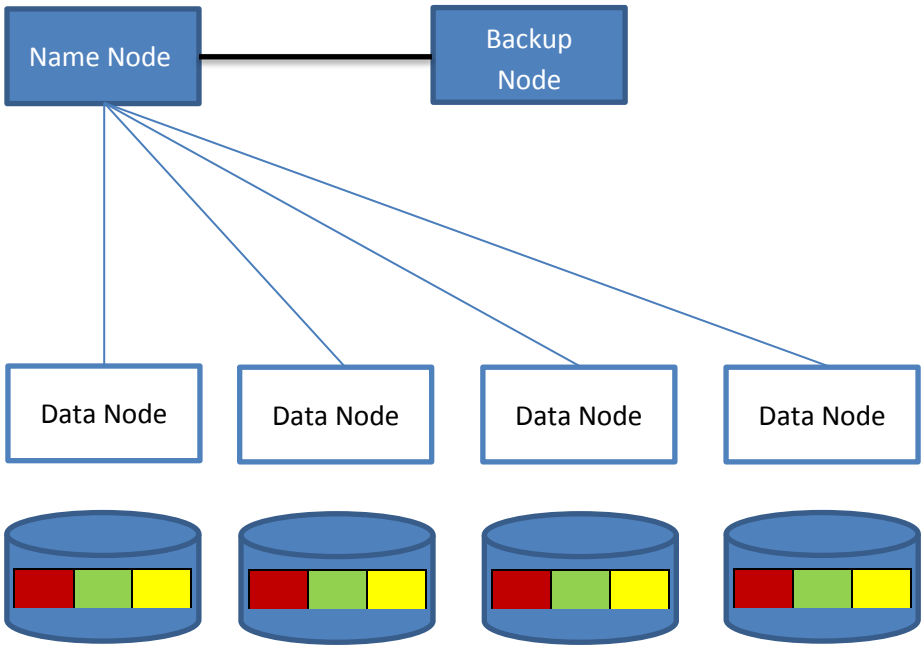
Most of the core projects are hosted by the Apache Software Foundation, which provides support for a community of open source software projects, including the original HTTP Server from which it gets its name.

Section 06

Big Data Design Principles

Hadoop is:

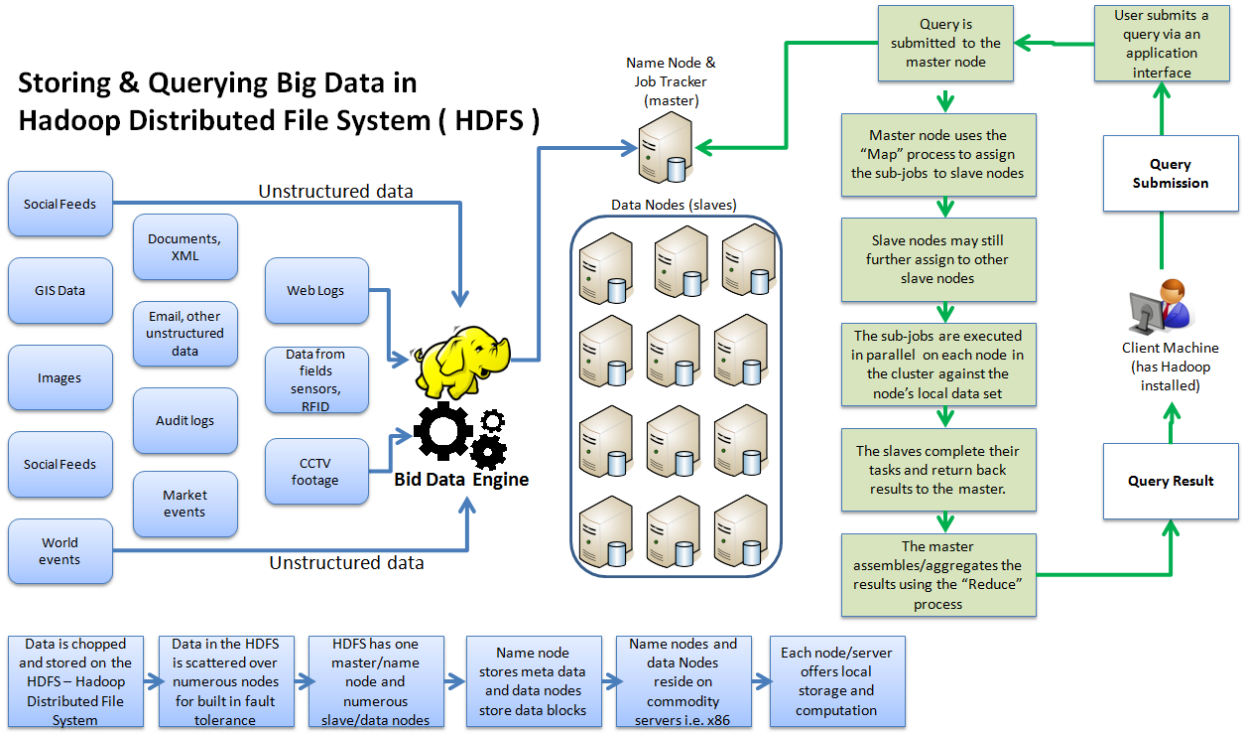
- Scalable linear
- Distributed
- Modular
- Extensible
- Move compute to data
- Each Node holds data and processes work
- Designed with hardware failure in mind
- System manages and heals itself
    - Automatically and transparently routes around failure
    - Automatically execute redundant tasks if certain nodes are detected to be slow



HDFS Architecture

Section 06

Big Data Design Principles



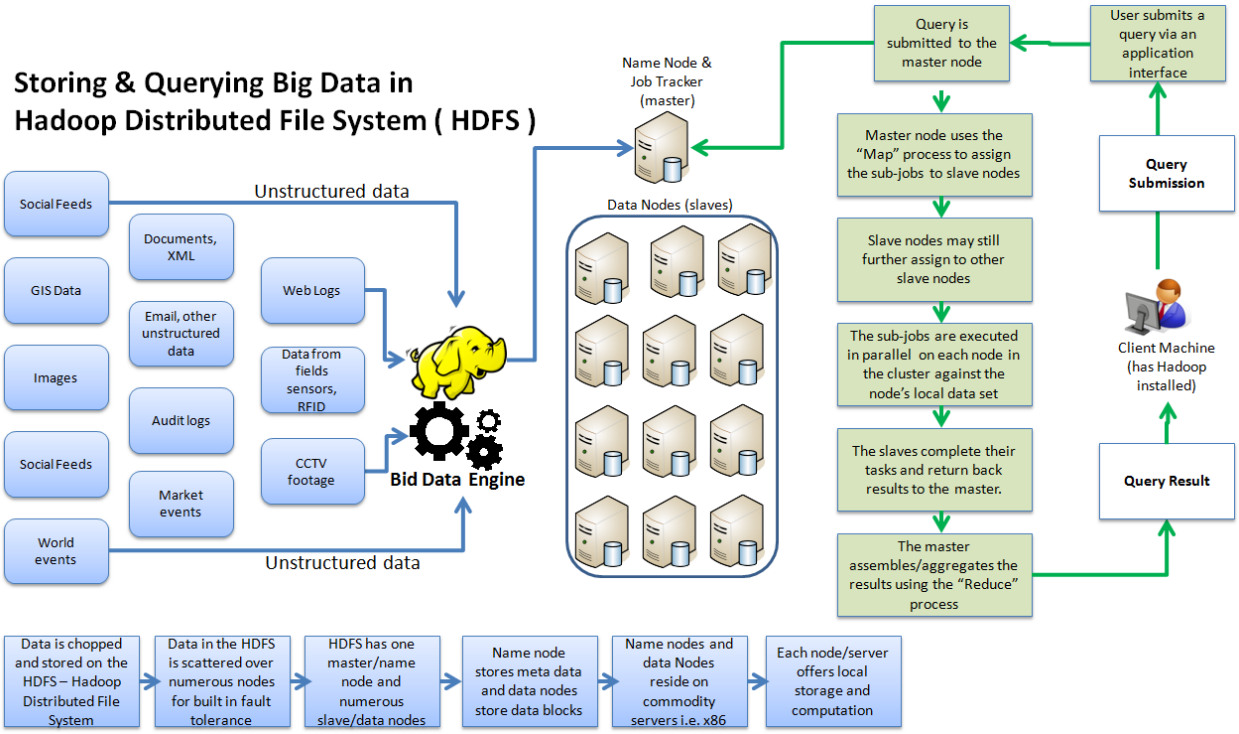
This is the Hadoop architecture for storing and querying data within HDFS.

Data Ingestion & Storage:

- Data ingestion (unstructured data).
- Data is chopped into data blocks 128-256 MBs.
- Data is stored over multiple nodes for performance and fault tolerance.
- Name node stores meta data.
- Data nodes stores block data.
- Name nodes and data nodes reside on commodity servers.
- Each node has local storage and each node does computation on local data sets.

Section 06

Big Data Design Principles



This is the Hadoop architecture for storing and querying data within HDFS.

Data Query:

- User submits query to the API.
- Query is submitted to the master node.
- Master node uses the Map process to assign sub-jobs to slave nodes.
- Sub-jobs are executed in parallel on each node in the cluster (each nodes local data set).
- Slave completes task and returns result back to the master.
- The master assembles the result using the reduce process.
- Query returned to the requester.