**INTRODUCTION TO**



Big Data Success Stories:

1. EBAY activates the power of big data- eBay leverages data to drive the business forward in all domains, from more accurate reporting to personalizing the customer experience. The company sees data not only as an essential asset, but as its key competitive edge.

## 2. “ONE RED CROSS” USES DATA TO BE POWERFUL AND BREATHTAKING- The American Red Cross uses data from multiple sources to better understand its donor base and deliver “powerful, breathtaking experience.”

3. Most of us heard the news about TESLA, the new car of the future that can drive by itself and lets the driver take off the hands from the wheel and relax. But how has Tesla achieved this and how it decides what to put and what not to put in the car, and if it puts something, how. The secret behind all these is the extensive use of data analytics. Tesla engineers are continuously monitoring the driver, what he/she does, how he/she reacts in various situations, what is comfortable and what is not, how the various situations are affecting the driver, etc. This is achieved through an extensive network of sensors that collect a huge volume of data. This data is then analyzed to provide valuable insight that is used to improve the car overall and offer an extensive, secure, comfortable and pleasant experience to the driver.

4. Another example in the retail sector is Wal-Mart that records the thousands of transactions that the customers do in their individual locations and then analyse these results to understand the popularity of products, associations between them and their preferences and habits.

5. IBM was another success story recently with the IBM Watson Jeopardy competition, where a computer was called to answer knowledge questions like those that humans are typically answering.

6. Sports is a sector that has benefited a lot from the use of big data, financial companies like American Express are running extensive analytics and business intelligence techniques on the transactions performed by their customers and try to understand their intentions, airlines are running extensive flight data analytics to better understand flight patterns and how the different planes behave in different situations.

7. As a final example in recent days can be the winning of Trump in US presidential elections where big data analytics are used.

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.

A definition of Big Data could be "Big Data is not about size, but it's about granularity". The ability of software systems to identify individuals and personalized data is the ironic implication of "Big". It is the ability to focus on the minutiae of the individual, in real time.

Extremely large data sets that may be analyzed computationally to reveal patterns, trends, and associations, especially relating to human behavior and interactions.



Characteristics of Big Data:-

1. **Volume**- scale of data

2. **Velocity**- analytics of streaming of data

3. **Variety**- different forms of data

4. **Veracity**- uncertainty of data

5. **Value**- what we get are researching

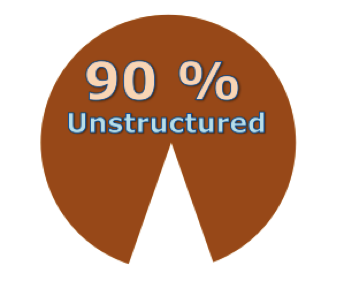
1. Volume:-

We need to think for a little how the situation has changed. Nowadays, it is estimated that we are producing 2.5 quintillions of bytes of data per day. To realize how much this is, think that one quintillion is one million millions.

The rate in which data is produced is constantly increasing. It is again estimated that 90% of the data that we have generated and collected so far has been produced in the last 2 years. You can imagine what is expected to happen in the years to come.



Another interesting fact of the data that we have nowadays is that approximately 90% is unstructured, i.e., coming from sources that are not well curated, or they are by non-conventional databases.



But the most important characteristic of the data is the fact that that they are produced at rates that would have never seen before. For instance, in only 60 seconds there are more than 80 thousand wall posts on Facebook, 370 thousands of Skype calls, around 100 thousands of tweets and more than 170 millions of emails sent.

The data of interest that are produced are not only of electronic or social network nature but can be from every aspect of human activity, like the phones that are purchased, the money transfers, the passing of people from a point of the city, or the fluctuations of a temperature in some neighborhoods.

2. Velocity:-

Another characteristic of the data we collect nowadays is that in many cases the data is arriving to the collection and processing system at very high rates. This is not an issue as long as the only task that needs to be done is to store the data. However, this is not always the case. Very often we need to be able to analyze and detect a situation that is modeled in the data at run time and take a corrective decision or any other type of action immediately.

In the past, city officials or emergency servicemen were not aware of a problem before someone was actually calling to report the problem. In the modern era, a problem can be reported at real time. For instance, cameras in various parts of the city allow the monitoring employees to see an incident the moment it happens. The fall of an electric pole is detected the moment it falls and the emergency workers can be notified and respond immediately, without waiting for someone to call and report the problem. A number of credit card transactions should immediately trigger a suspicious activity alarm and block the card pending further verification.

This kind of data are typically called streams. Streams need special management since one does not have the luxury to store the data and make multiple passes over it. One can take only one look of the data as it arrives and then the data is lost, or even if it is store, the speed in which the data arrives does not allow a second or third pass over it.

When the rates of the arriving data is relatively slow, there are already multiple techniques that have been developed over the past decades that allow one to make inferences and answer queries over streaming data, even with a single look over their values. However, nowadays that the streams have become so large in terms of speed and volume, and the number of streams has also significantly multiply, the processing of data streams is becoming a real challenge.

3. Variety:-

It refers to the different types of data we can now use. In the past we focused on structured data that neatly fits into tables or relational databases, such as financial data (e.g. sales by product or region). In fact, 80% of the world’s data is now unstructured, and therefore can’t easily be put into tables (think of photos, video sequences or social media updates). With big data technology we can now harness differed types of data (structured and unstructured) including messages, social media conversations, photos, sensor data, video or voice recordings and bring them together with more traditional, structured data.

4. Veracity:-

By **Veracity** we mean the amount of trust that one puts on the data. Since the data is coming from sources that may not be of our control, may have many data quality issues, and may not be coming from specific authorities, there is a need for measuring the trust that we put on the data, which in turn means the trust we put on the conductions, and insights that we have obtained by analyzing the data.

5. Value:-

Value refers to our ability turn our data into value. It is important that businesses make a case for any attempt to collect and leverage big data. It is easy to fall into the buzz trap and embark on big data initiatives without a clear understanding of the business value it will bring.

Big data can deliver value in almost any area of business or society:

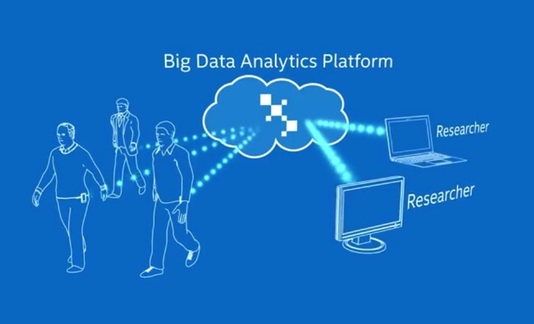
* It helps companies to better understand and serve customers: Examples include the recommendations made by Amazon or Netflix.
* It allows companies to optimize their processes: Uber is able to predict demand, dynamically price journeys and send the closest driver to the customers.
* It improves our health care: Government agencies can now predict flu outbreaks and track them in real time and pharmaceutical companies are able to use big data analytics to fast-track drug development.
* It helps us to improve security: Government and law enforcement agencies use big data to foil terrorist attacks and detect cyber crime.
* It allows sport stars to boost their performance: Sensors in balls, cameras on the pitch and GPS trackers on their clothes allow athletes to analyze and improve upon what they do.

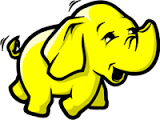
We must be sure to never forget the fifth V: Value. How will big data benefit you and your organization? Without that starting point organizations will drown in their data while thirsting for the benefits.



Big Data Analytics:-

It is the process of examining large data sets containing a variety of data types- i.e., big data – to uncover hidden patterns, unknown correlations, market trends, customer preferences and other useful business information.



 HADOOP

History-

We all know about the Google search engine which processes the request of millions of users daily. For executing and displaying the result of queries of those many users daily would be a heavy task. In 1990, problem got raised for storing data as it have to store a huge amount of data to satisfy the needs of millions of users from all over the world. It took 13years to find a solution to that problem as it can’t be solved by using traditional databases. In 2003, GFS(Google File System) and in 2004, MapReduce was introduced by Google for storing and processing huge quantities of data. This solution was a discovery not an implementation method. It is a theoretical paper which describes how to solve that problem and what should be the measures to be taken. Yahoo took this solution as base and modified it by applying its own methodologies and introduced HDFS(Hadoop Distributed File System) in 2006-07 and MapReduce in 2007-08.

The inventor and founder of hadoop is **Doug Cutting**. During the project, everyone was thinking about the logo for the new discovery. One fine day Doug observed his son playing with an elephant toy to which he gave the name as hadoop. At that moment he decided to name their inventory as hadoop.

**Hadoop** is an open-source framework given by Apache Software Foundation for storing huge datasets and processing huge datasets with cluster of commodity hardware. It has two main components:

**1**. **HDFS**- Hadoop distributed file-system is a framework used for storing our data.

**2. MapReduce-** MapReduce is a framework which is used to process the data which we are storing in HDFS.

1. Hadoop Distributed File System(HDFS):-

Hadoop distributed file system is a specially designed file-system for storing large datasets with cluster of commodity hardware and with streaming access pattern. Here cluster is a group of data nodes or network of machines or processors where each of them having their own primary storage(ex-RAM) and secondary storage(ex-harddisk). The hardware used in hadoop file system will be at low cost i.e., we don’t require costly machines for storing and processing our data. Hadoop need not to be worked on highly reliable hardware. It can be worked on simple cheap hardware.

Suppose if we have 500TB of data. It will take 4-5crores of money to maintain that much data on servers whereas 1-1.5crores will be sufficient to maintain on normal machines which we use daily like PCs, laptops etc.

Let us see an example how the files are stored in normal file system and HDFS.

We have a harddisk capacity 500GB where each block consists of 4Kb size. If we have a file size of 2Kb then remaining 2Kb space will be wasted. It is left blank or as free space. This is done in our normal file system.

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  | Each block 4kb in size |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |

But in our HDFS, by default the block is given 128Mb. We install hadoop(HDFS) on top our hard-disk where the memory will not wasted.

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  | Each block 128Mb in size |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |

If we have a file size of 35Mb, it is first divided into blocks and stored in one of the 128Mb block where after storing 93Mb will be remaining as free space which will be released for some other file storing. As hadoop stores large data sets that’s the reason why the block size is default 128Mb.

Features of HDFS:-

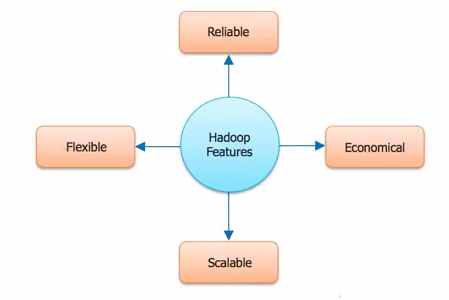
Hadoop distributed file system has four important features which makes it idle to use for big data analytics.

1. Scalable

2. Reliable

3. Economical

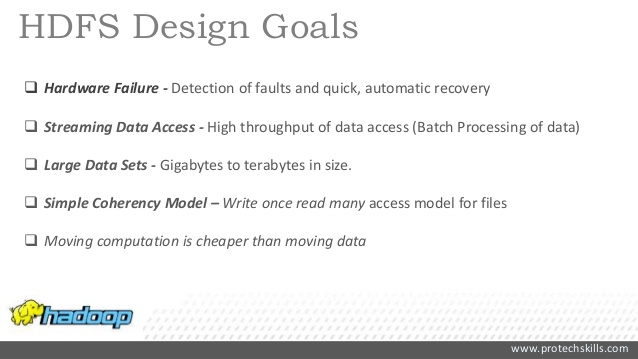
4. Flexible



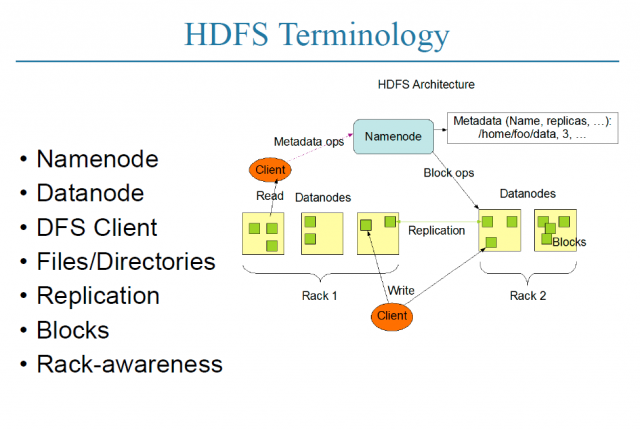
* Reliable:
* In a large cluster, something is always broken.
* Nodes fail every day. Failure is expected, rather than exception.
* Engineering reliability into every app is expensive.
* Fault tolerance and easy management:
* Every block is replicated.
* Disk corruption and node failures are internally handled.
* Node addition/removal is done automatically.
* Every 3K nodes managed by one operator.
* Economical:
* Use commodity components when possible.
* Thousands of these put into an effective compute and storage platform.
* Minimizes bandwidth consumption and latency:
* Satisfies a read request from replica closest to the reader(on the same rack).
* If the cluster has multiple data centers then local replica is preferred over remote replica.
* Optimized for batch processing



Goals of HDFS:



HDFS Terminology:



Master-Slave Architecture:-

HDFS has two core components:-

1. NameNode

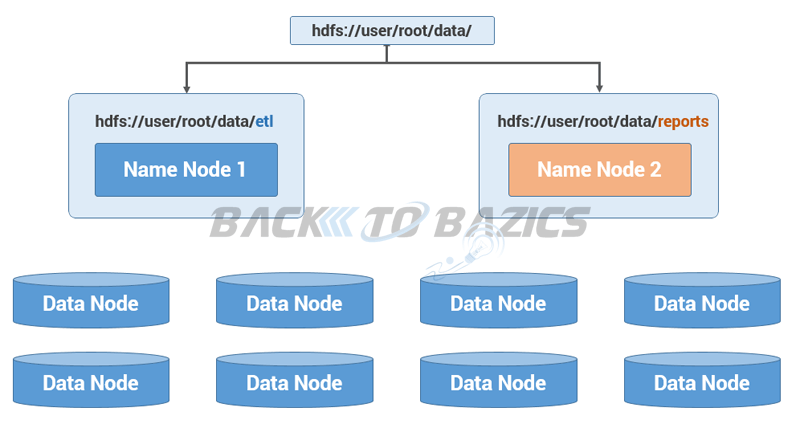
2. DataNode

NameNode is a master service/master node/master demon which holds the metadata about the files stored in the data nodes.

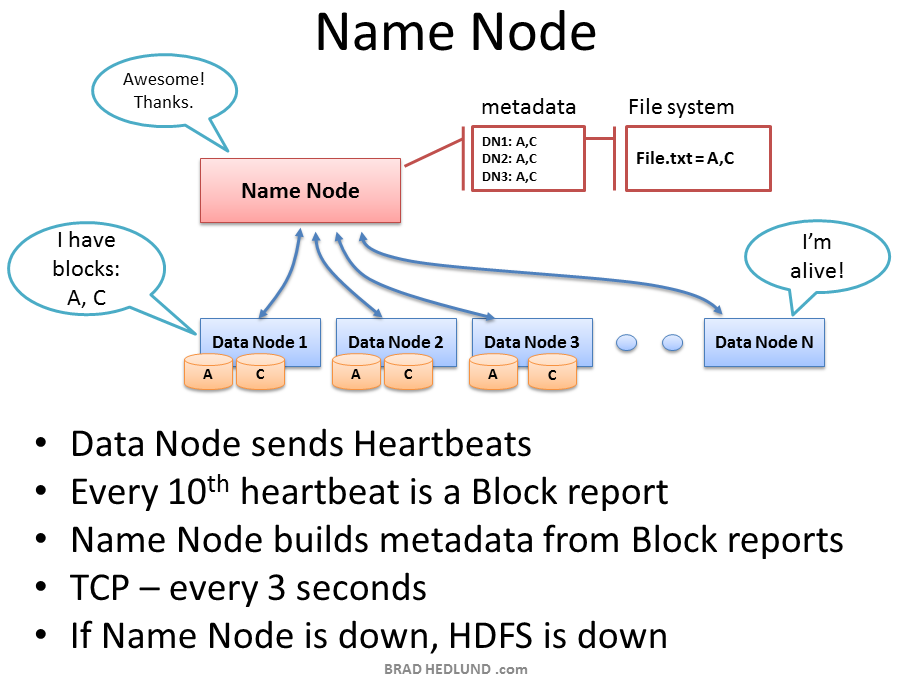
DataNode is a slave service/slave node/slave demon which holds the actual data of the file in the form of blocks.

The cluster will never hold all the data in a single node. It always follows distributed manner for storing a file. Master node through the help of metadata keeps the whole track of a file and it controls the entire cluster.

Reading and writing operations are done sequentially and processing is done parallely.



NameNode’s Responsibilities:

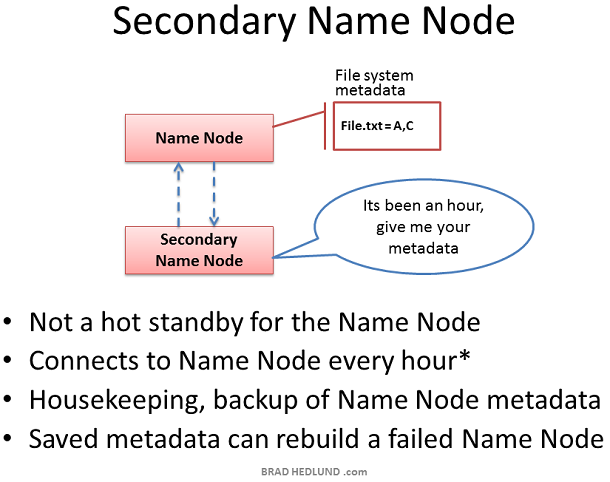


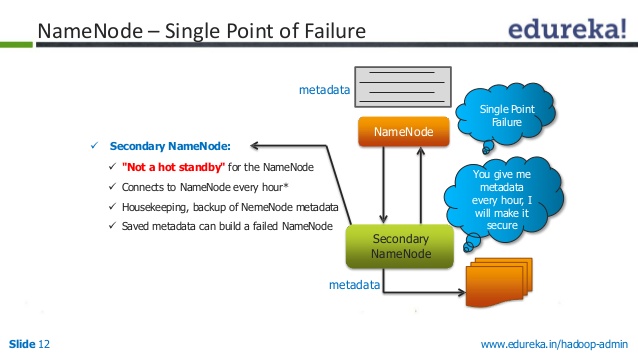
* Splitting the given file into blocks
* Datanode availability and bandwidth check
* Block storage
* Fetching data from blocks and giving to user
* Master to all slaves
* Replication operations
* Maintains the heartbeat of all datanodes

NameNode: Persistence of Metadata

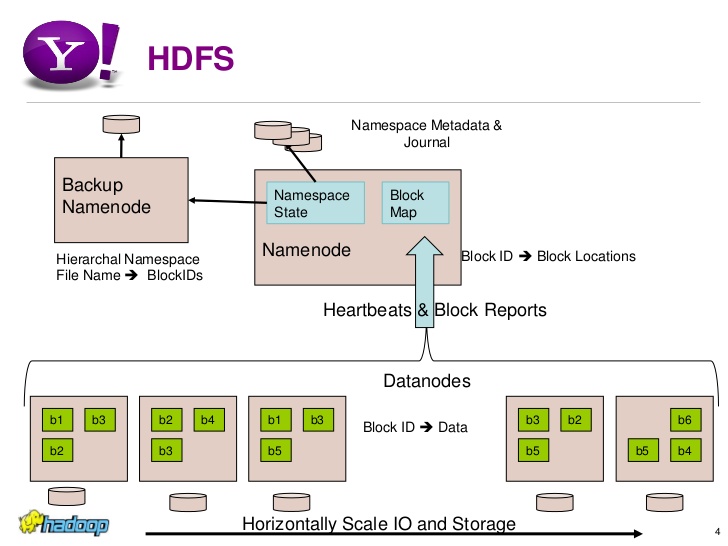
* Metadata-EditLog(external client details) and FSImage(metadata about the blocks)
* Checkpoint creation
* Block Reports and heartbeats
* Handling transactions performance
* Future additions

Secondary Name Node:

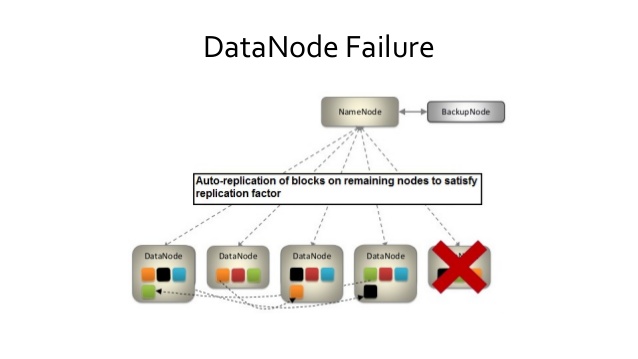




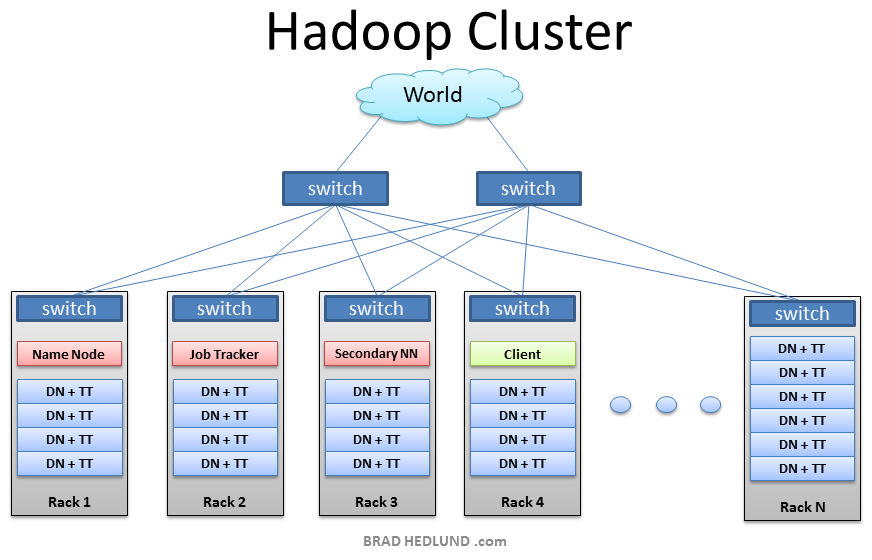
DataNode:



* Slaves to the NameNode
* Block contents
* Handshaking on startup
* Node registration
* Block Reports
* Heartbeat communication(status of the datanodes)



HDFS Cluster Setup:

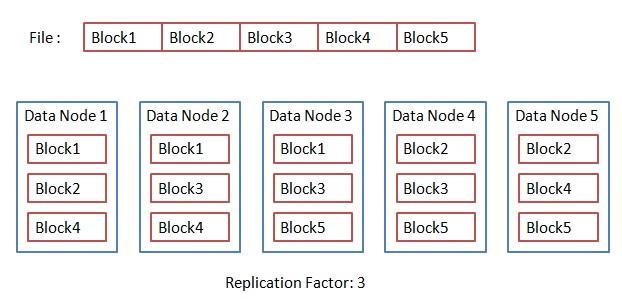


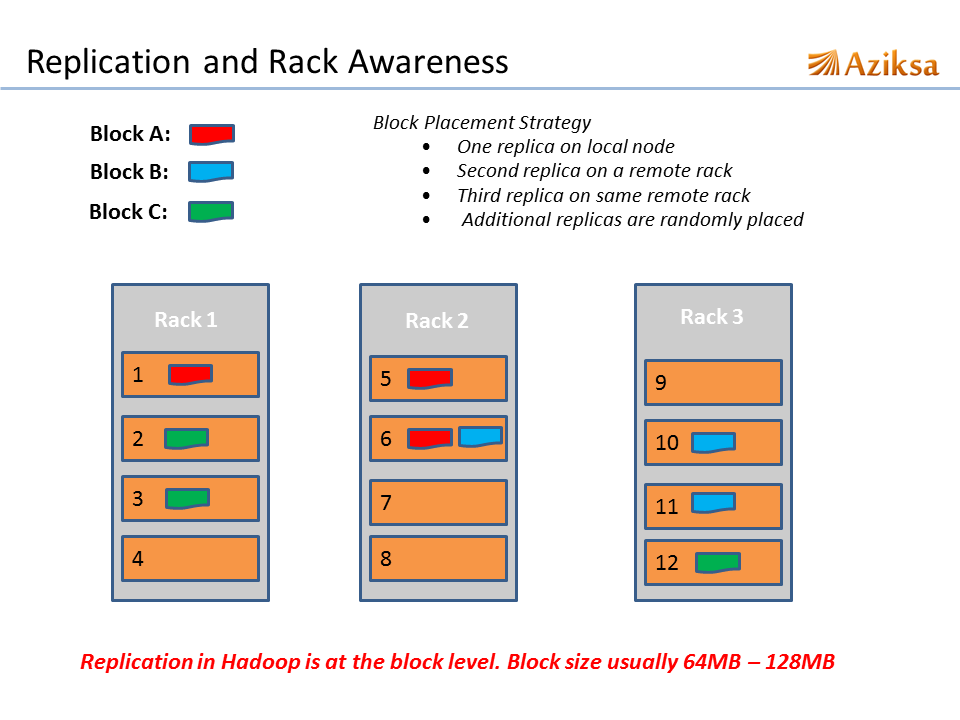
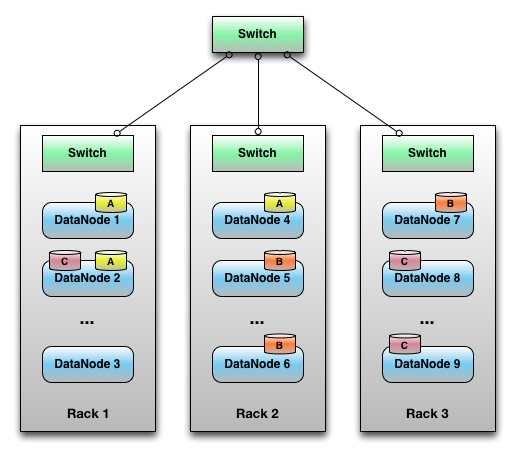
HDFS-Replication:-

Every block present in the data nodes are replicated over the network. This process is being done to make the data available even fault tolerance occurs and to keep the data safe from unauthorized accessing.

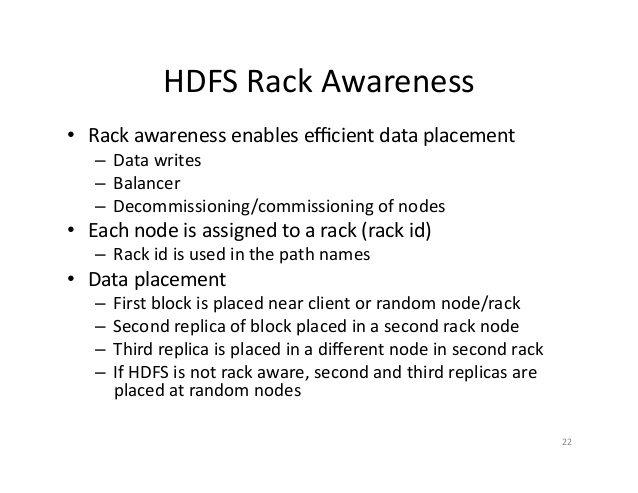
All of the copies are readable, process-able, updatable and accessible by all the users depending upon the location of the user and bandwidth. The main advantage of replication is dividing the load factor.

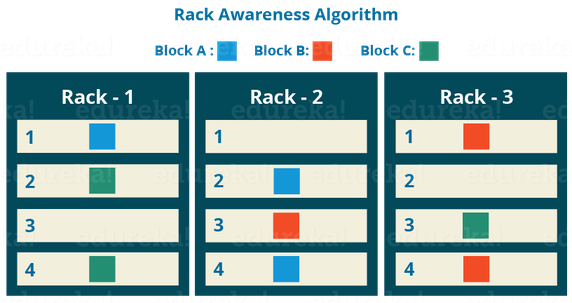
The default replica factor=3 i.e., minimum 3 copies are made of each block and they are stored in different data nodes so that if any node gets crashed or any network failure occurs we can fetch the data from other copies of that file. All the replicas are original copies and gets updated in all of its copies simultaneously if any change occur in any of the copy.

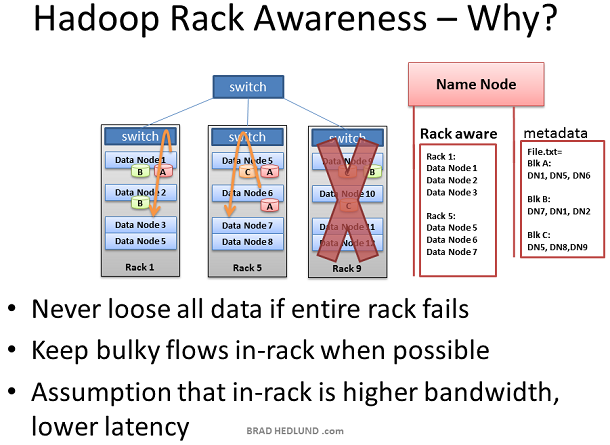


HDFS-Reliability

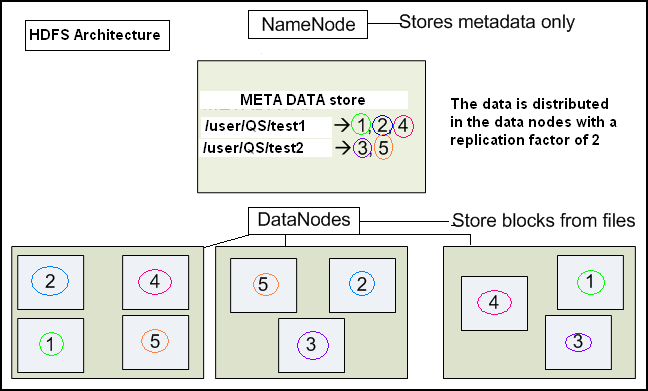
HDFS-Rack Awareness:



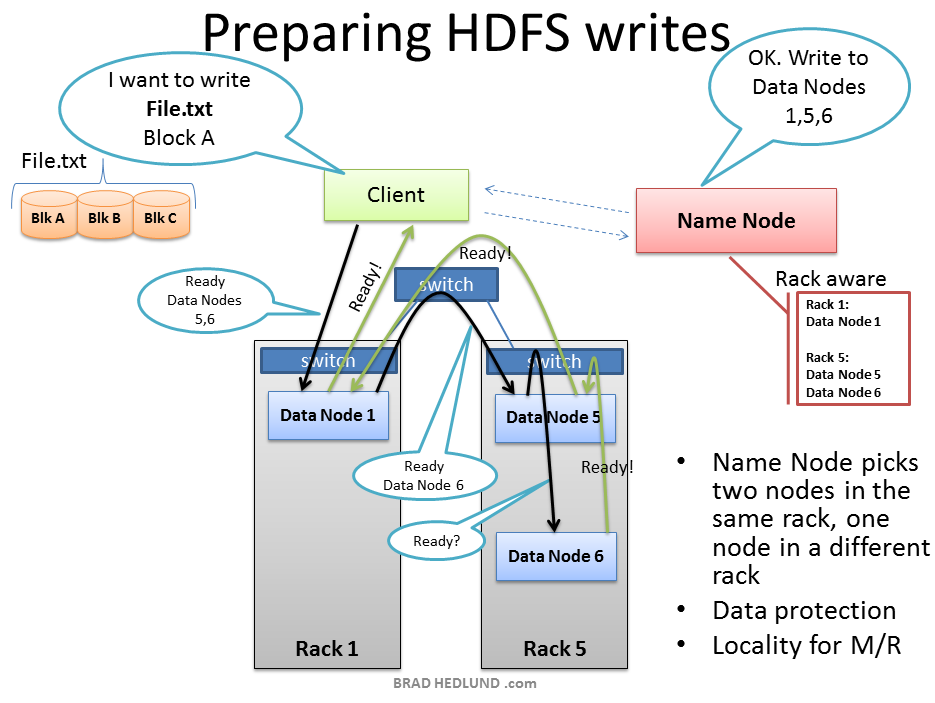


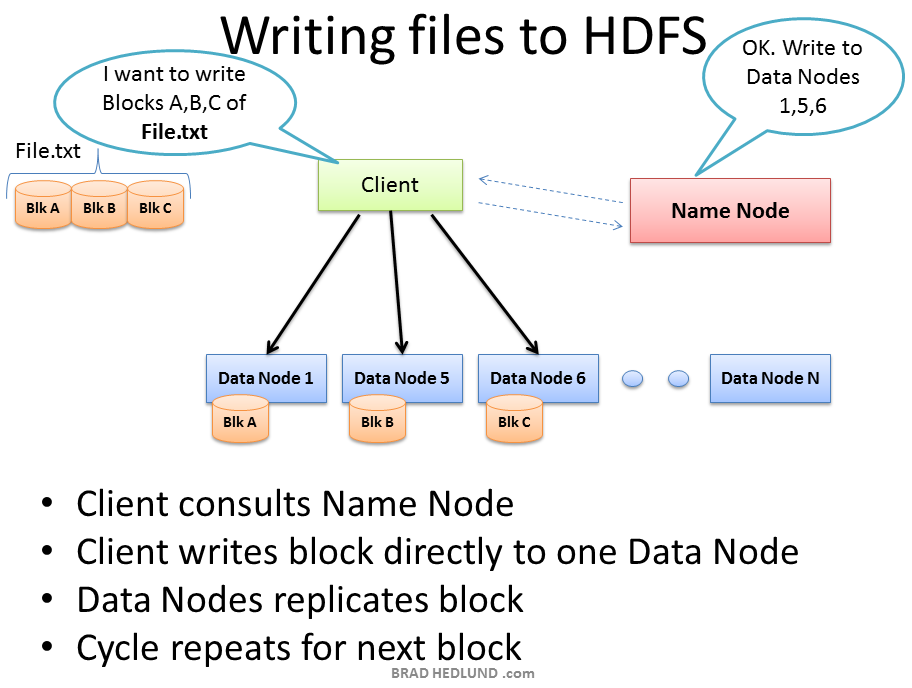


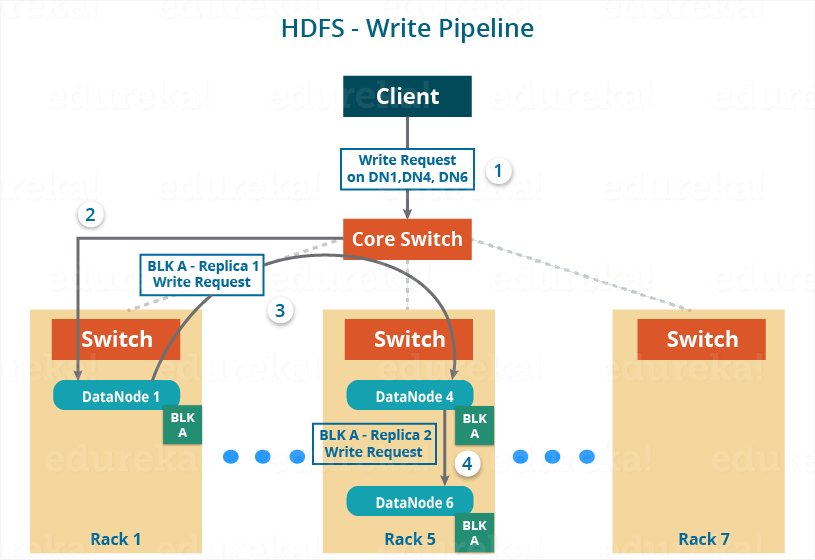
HDFS-Data Organization:

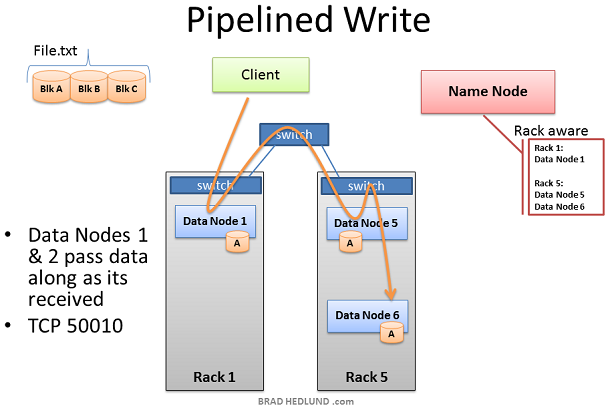


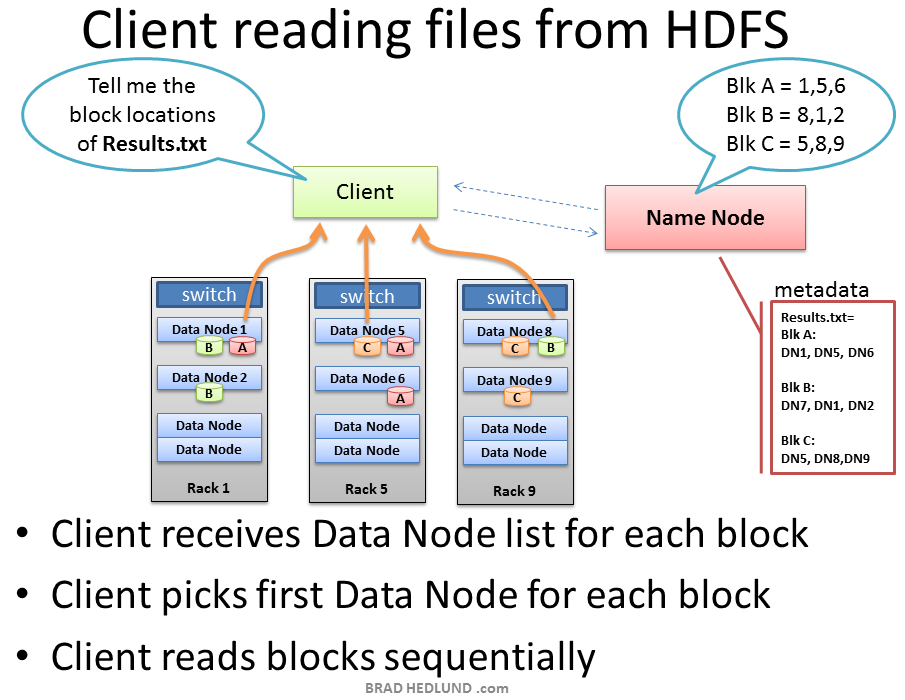
HDFS: Writes and Reads

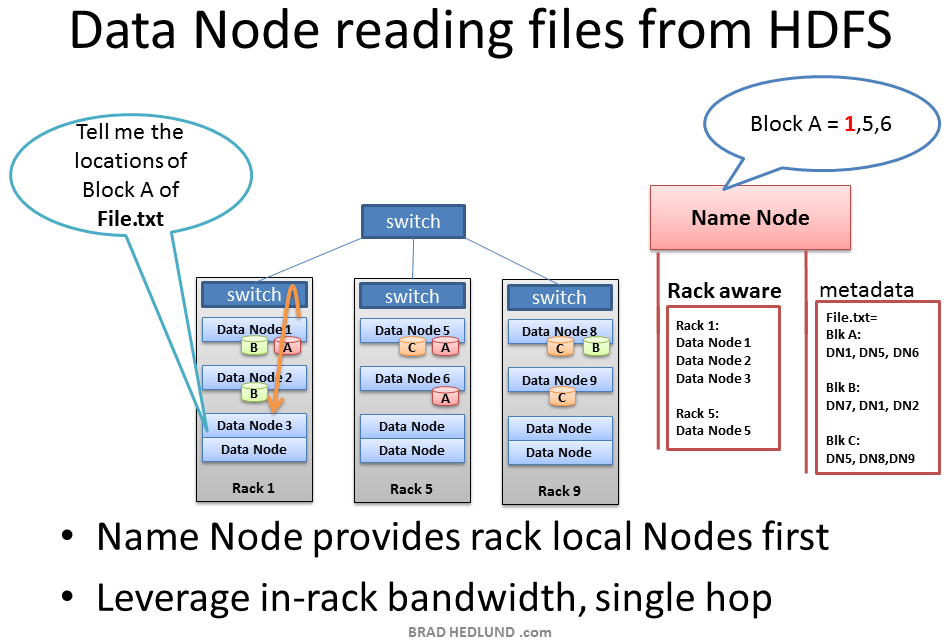




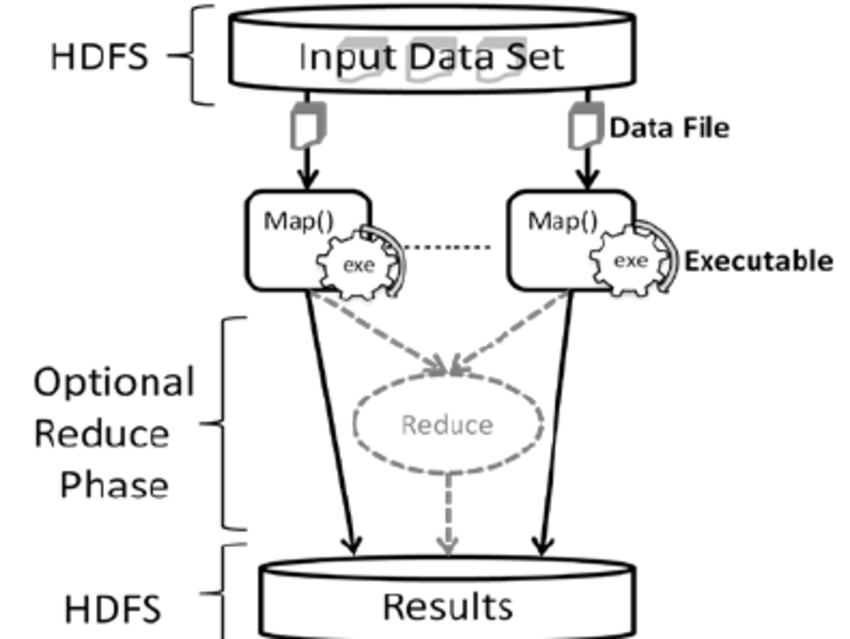


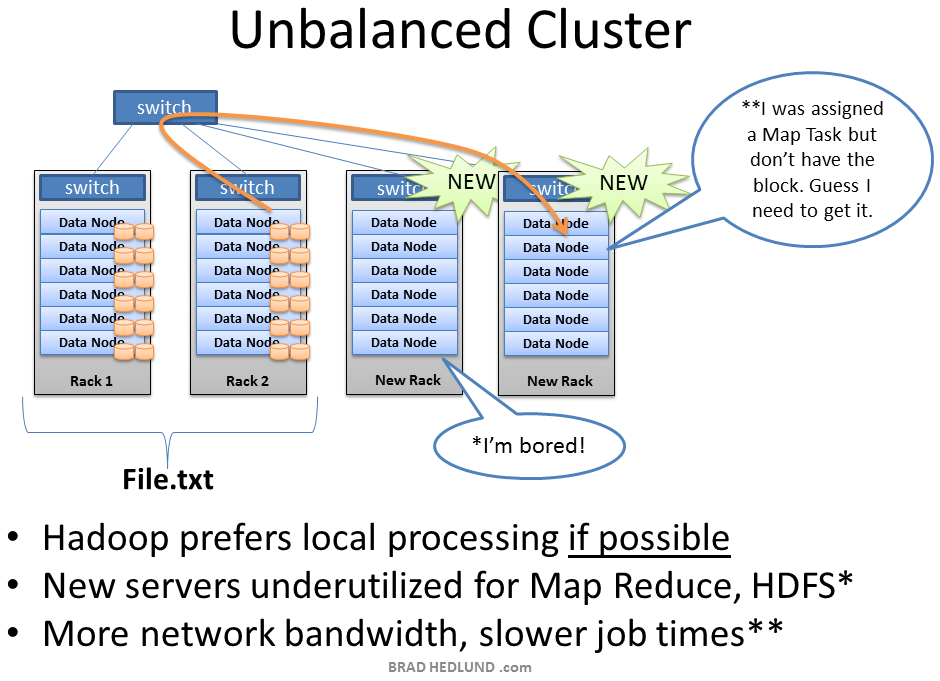






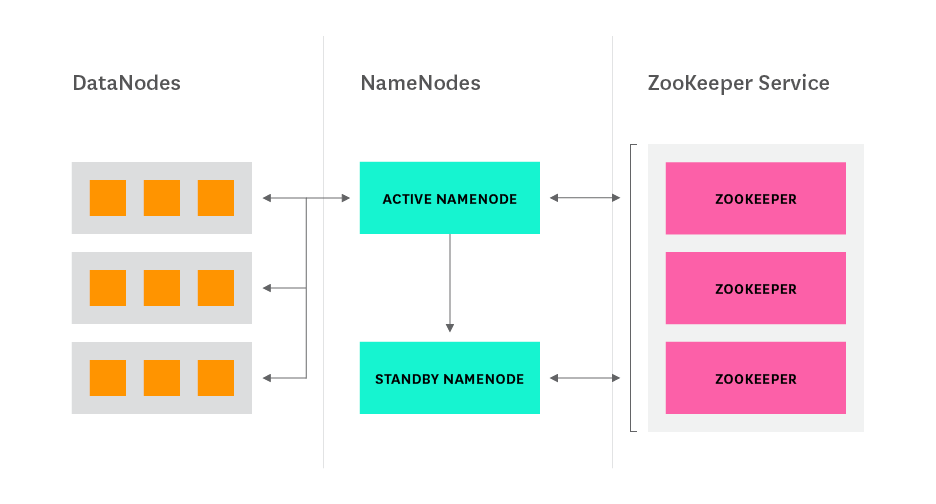
HDFS: Data Locality

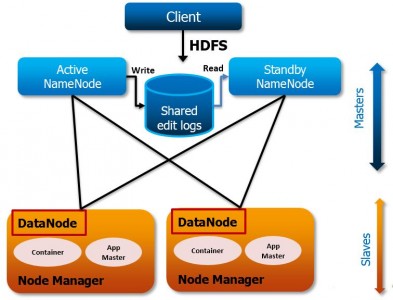


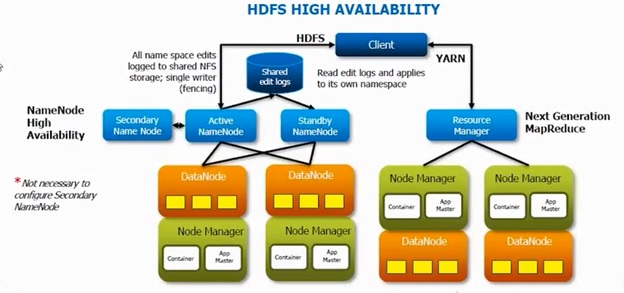
HDFS:Rebalancer

In HDFS, there is an in-built software called re-balancer present in namenode which is used to balance the load present on the datanodes so that the data is made available to the user when requested. Depending upon the bandwidth of the datanodes, blocks are read.

The Latest Hadoop(with HA):







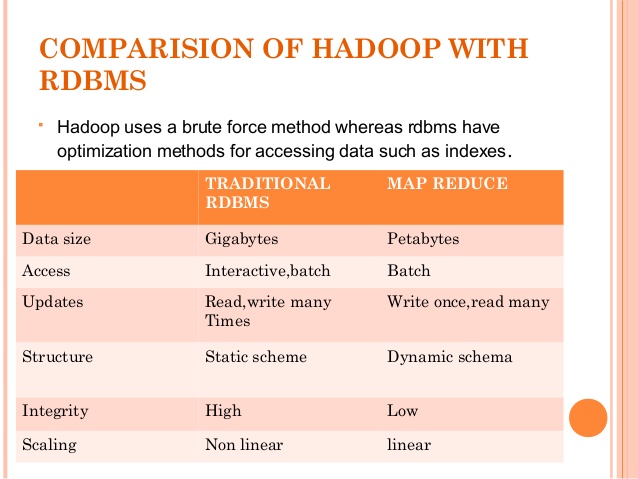
Till hadoop 1.x version we have secondary namenode which acts like an external data which stores data on hourly basis. The last hour amount of data will not be stored. It gets lost.

To overcome this problem activenamenode and standbynamenode are introduced from hadoop 2.x in the place of secondary namenode.

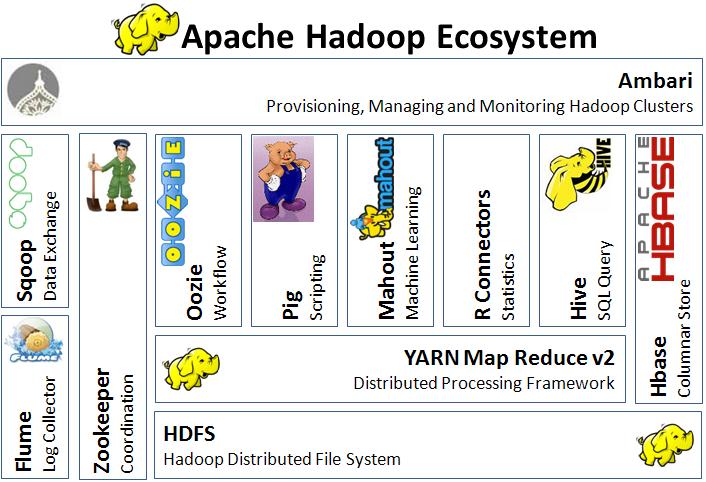
ActiveNameNode is nothing but the namenode which acts a master node and standby namenode in which data is stored as soon as it arrives or get updates. It is an online storage system where the storage process is parallely done. We can have multiple namenodes and standby namenodes in this version of hadoop 2.x. Standbynamenode is called HotStandby and SecondaryNameNode is called ColdStandby.

In the case of namenode getting down, we can make standbynamenode as activenamenode by copying its contents to other namenode and it will act as activenamenode as the standbynamenode is updated through online and contains the same information as activenamenode does. Whereas the secondary namenode cannot be converted to namenode as the last amount of data is not stored in it and it’s back-up process is done on hourly basis. The whole process of this conversion and managed by zookeeper which maintains the heartbeat of the namenode.

HADOOP VS RDBMS:-



Hadoop Ecosystem:-



HDFS- framework for storing our files

YARN- framework used for processing the files stored in our HDFS

Flume, Sqoop- tools helpful to import/export the data(data loading tools)

Zookeeper- tool used for synchronizing the various services present in the hadoop framework

Oozie- tool used to keep the track of the workflow

Hive, Pig- frameworks used for processing the stored data

Mahout- framework provides an environment for creating machine learning applications

R Connectors- tool used for predictive analysis

HBase- nosql database

2. MapReduce:

Hadoop MapReduce is a software framework for easily writing applications which process vast amounts of data (multi-terabyte data-sets) in-parallel on large clusters (thousands of nodes) of commodity hardware in a reliable, fault-tolerant manner. MapReduce idea is inspired from similar paradigm used by functional programming languages like LISP. Their programs are easy to run in distributed environment.

In 2003, Google has introduced the MapReduce concept but later in December 2004, it published a paper explaining architecture of MapReduce.

A MapReduce job usually splits the input data-set into independent chunks which are processed by the map tasks in a completely parallel manner. The framework sorts the outputs of the maps, which are then input to the reduce tasks. Typically both the input and the output of the job are stored in a file-system. The framework takes care of scheduling tasks, monitoring them and re-executes the failed tasks.

Typically the compute nodes and the storage nodes are the same, that is, the MapReduce framework and the Hadoop Distributed File System (see [HDFS Architecture Guide](https://hadoop.apache.org/docs/r1.2.1/hdfs_design.html)) are running on the same set of nodes. This configuration allows the framework to effectively schedule tasks on the nodes where data is already present, resulting in very high aggregate bandwidth across the cluster.

MapReduce has two core functions:

1. Mapper

2. Reducer

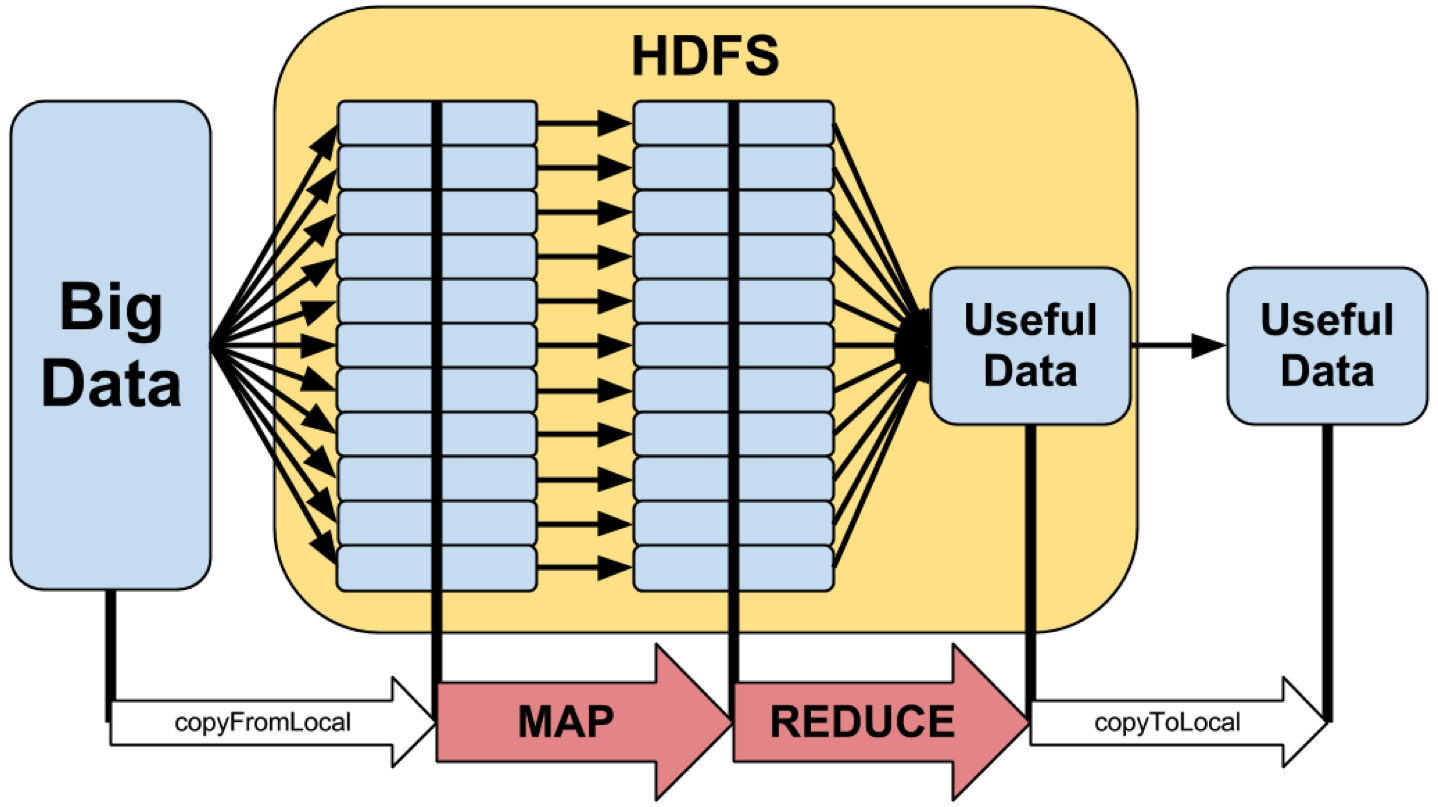
1. Mapper:

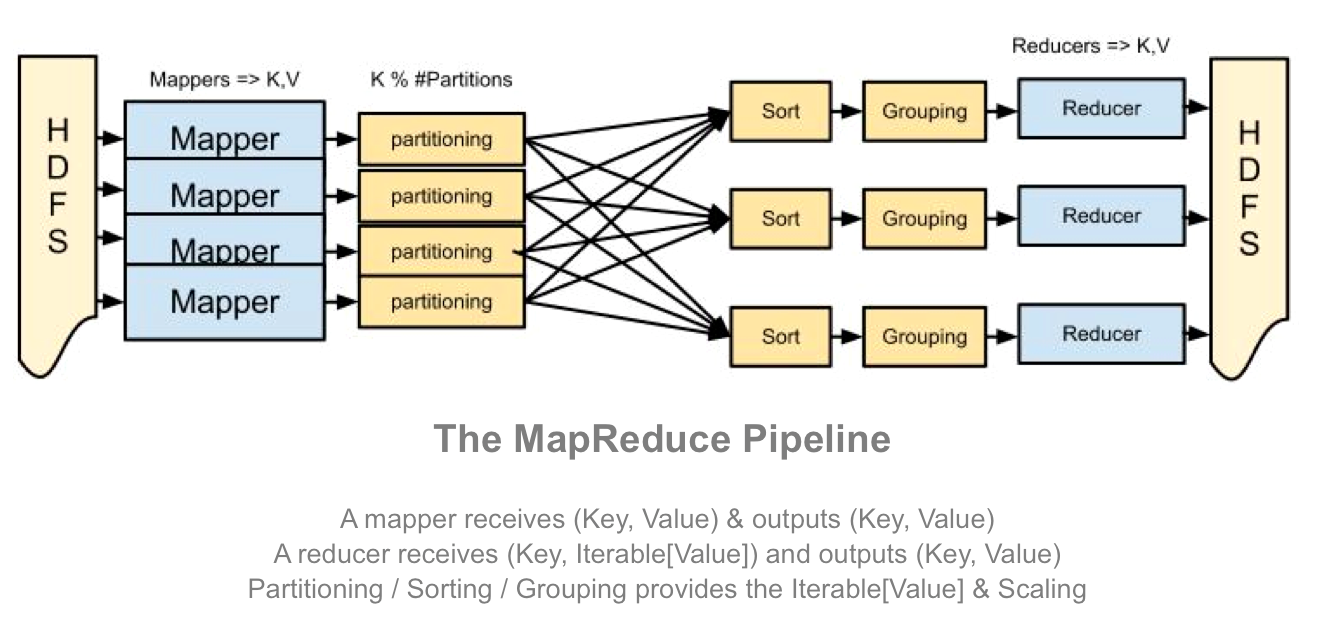
Mapper is a simple program which filters the data and transforms to the lowest logical level. It extracts the required data from the blocks and does processing on it. It takes input as key & value pair and generates output also in key & value.

2. Reducer:

Reducer is a function where it is used for aggregating the values of the mapper output. It applies business logic to the output of the mapper which is taken as an input to the reducer function. It takes map function output key and corresponding value list. After aggregating all the values it generates output key & value.

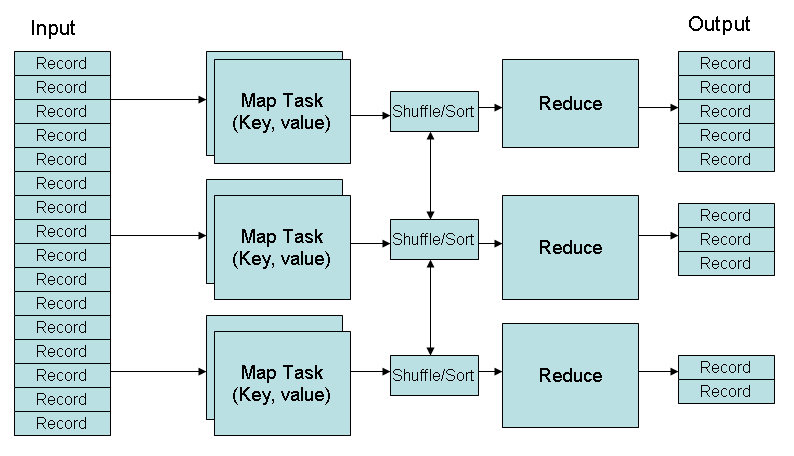
These functions makes it easy to run programs in parallel mode. These tasks are performed on a particular block.

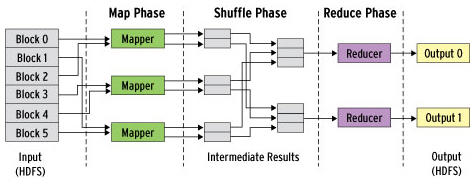


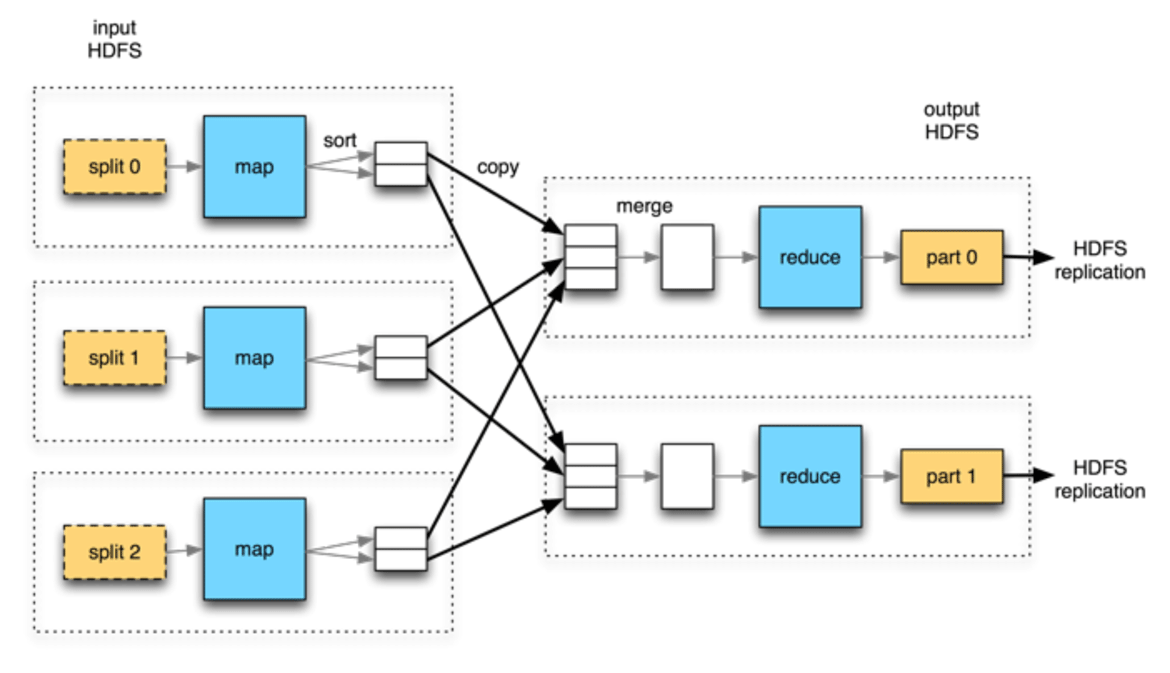


Maps are the individual tasks that transform input records into intermediate records. The transformed intermediate records do not need to be of the same type as the input records. A given input pair may map to zero or many output pairs. The number of maps is usually driven by the total size of the inputs, that is, the total number of blocks of the input files.

The transformed records are shuffled and sorted based on the key & values given by the output by the mapper and the sorted records are given as input to the reducer where aggregation operations are performed based on that key & value.







Let us understand with help of an example:



