Big Data Course by Trending Tech(Sumit sir)
Week-1 Introduction to Big Data
Introduction to Big Data

Def by IBM - any data can be classified as 3 V's

1. Volume - huge vol of data
2. Variety -
    a) Structured data - RDBMS DB [oracle & MySQL]
    b) Semi-Structured data - CSV, XML, JSON
    c) Unstructured data - Audio, Video, Image, LOG files
3. Velocity - Speed at which data is coming
4. Veracity - nature of data -- Poor quality data, unclean data

Why Big data ?
- To process huge amount of data which traditional systems are not capable of processing.
- Big Data system Requirements -- ***Storage***(store massive amount of data), ***Process***(process it in a timely manner), ***Scalability*** (scale easily as data grows)

Two ways to build a system-

✓ **Monolithic** - One powerful big system with lots of resources.
- Hard to add resources after a certain limit.
- not scalable | 2xResources != 2xPerformance

✓ **Distributed** - Many smaller systems come together, each small system is called a node, together it is called a cluster.
- 2xResources = 2xSpeed
- Adding nodes is easy

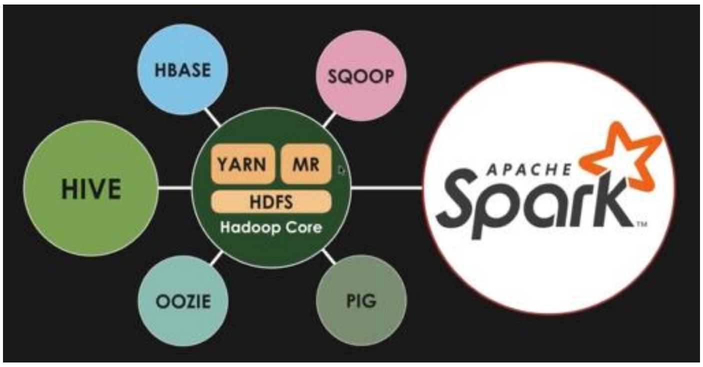
Resources -- RAM-Hard Disk - CPU | Memory-Storage-Compute

Vertical Scaling[Monolithic | Not true scaling] - keep on adding resources to a single machine, after a certain point there is a dip in performance.

Horizontal Scaling [Distributed | True scaling] - adding more machines - better than monolithic for scalability.

HADOOP

Hadoop is a framework to solve Big Data problems.
- Hadoop 1.0 --
- HDFS is for distributed storage.
- **Map Reduce** is for distributed processing.
- **YARN** [Yet Another Resource Negotiator] - Resource management
- Hadoop 2.0 -- HDFS + Map Reduce + YARN



Hadoop provides High throughput and high latency
HIVE - Data Warehouse tool built on top of Apache Hadoop for providing data query and analysis.
HQL - High Query Language.
SQOOP - A command- line interface application for transferring data between relational dbs and Hadoop or vice-versa. Sqoop is for *data ingestion* and *data migration*.
HBASE - A column-oriented NoSQL database that runs on top of HDFS
PIG - Pig Latin | A scripting language for data manipulation.

1. To clean the data.
2. Convert unstructured data into structured format.

Pig is obsolete nowadays, so both these things are done by Apache Spark now.
OOZIE - A workflow scheduler system to manage Apache Hadoop jobs.

== Sqoop, Hive and Pig internally both are Mapreduce.
== Sqoop is a special mapreduce job where only mappers work not reducers.

Que. what does it mean by sqoop/hive/pig are an internal MR job?
-- Sqoop transfers data from RDBMS table to hdfs directory using Sqoop commands.
Hive is used for analyzing stored data in HDFS or in hive tables using SQL language.
So whatever operation we do on these frameworks, they are executed as a MapReduce job. ie writing a simple SQL statement like insert into table in Hive is converted into a MapReduce job internally and then executed.

APACHE SPARK

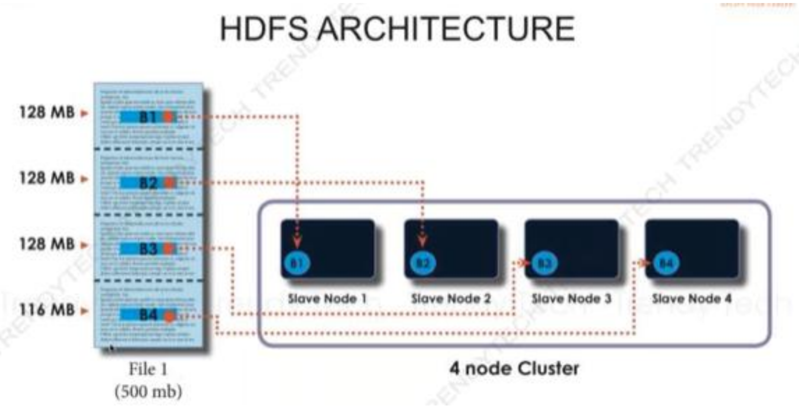
Spark -A distributed general purpose in-memory computing engine.
- Spark in place of MapReduce for computation
- **Apache Spark = HDFS + Spark + YARN**

Spark Cluster needs two things to work with
a) Storage -- Local Storage / HDFS / Amazon S3
b) Resource Manager - YARN / MESOS / Kubernetes
c) Compute - Spark

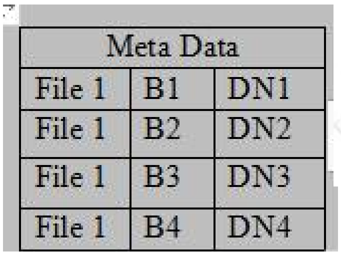
== Spark on top of Hadoop means storage manager is HDFS and Resource manager is YARN
== Spark can run without Hadoop also ex. Amazon S3 + Kubernetes.
== Spark supports Java, Scala, Python(PySpark) and R

HDFS Architecture Part-1

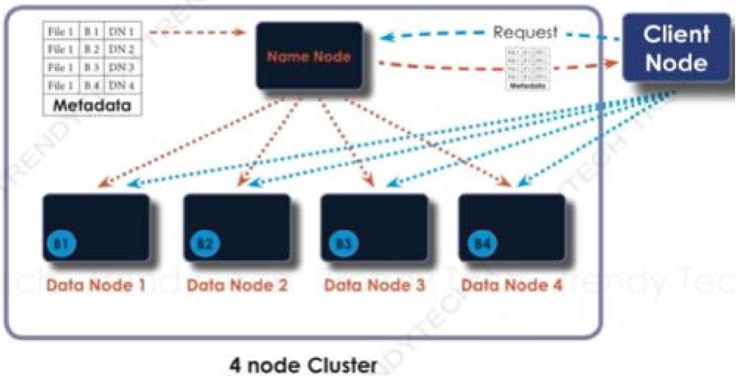
Block Size in hadoop 2.0 = 128 mb
Master Node | Name Node -- which stored meta data on what is kept where.
Slave Node | Data Node -- where actual data is kept in form of blocks



This metadata table is stored in name node in memory andNot in disk to save time.



Meta Data		
File 1	B1	DN1
File 1	B2	DN2
File 1	B3	DN3
File 1	B4	DN4

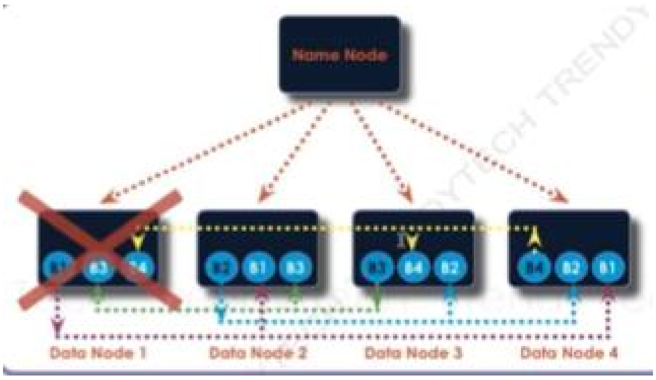


Client Node -- you sitting on a laptop nd u made a request for reading file. So this request will goto the name node and name node will give you location where file is kept. Ie this table.

== only client can read the file, name node will not read the data present in file.

Failures wrt Name Node and Data Node

Replication Factor (Default Hadoop) = 3
All the replicas should not be on same machine. Ie if one goes down all the replicas will go down.
Replication info^n should also be contained in the meta data table.

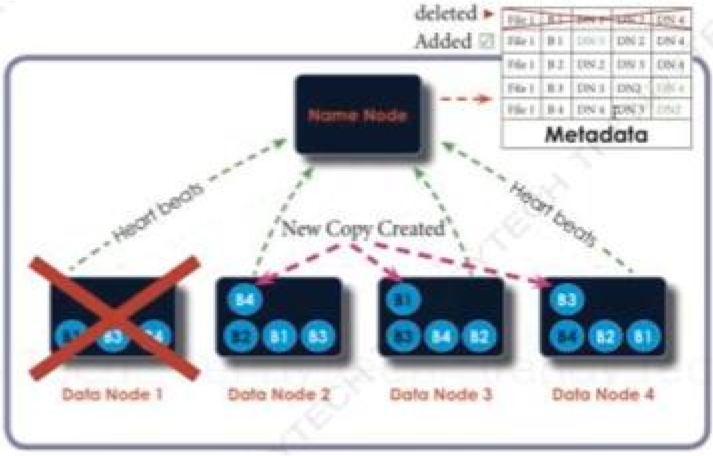


Heart Beat Mechanism

Each Data Node sends heartbeats to Name Node in every 3 seconds.
If a Name Node doesn't receive 10 consecutive heart beats, it assumes that the Data Node is dead or running very slow.

Fault Tolerance

If a Data Node goes down the replication factor came down to < 3
Name Node will create one more copy to maintain the replication factor



HDFS Architecture Part-2

Failure Management of a Name Node

== In Hadoop V1 the Name node was a single point of failure.
== in Hadoop V2 Name node is no longer a single point of failure. That is if name node goes down there's is no failure involved.

== Name node fails means no access to metadata that means no access to cluster that is data in data node.

-- Problem if Name Node is not there?
-- We will be losing the block mapping information.
-- what we want is that if we have the latest block mapping info^{n} (meta data) then we can make sure there is no downtime involved.

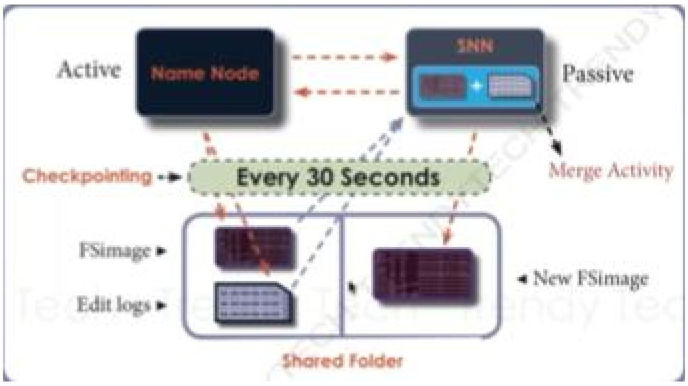
2 things that help us to make sure there is no downtime involved.
<u>2 Important metadata files</u>--

- ✓ FSimage --> snapshot of in-memory file system at a given moment. {scorecard during a cricket match.}
- ✓ Edit logs (edits) --> all the new changes or transactions that happen after the snapshot is taken will come to the edit logs file.

latest FSimage = FSimage + edit logs
== merging of FSimage + edit logs is a compute heavy process.
== Name node should not take the activity of merging these 2 files as NN is already busy doing lot of other things.

Secondary Name Node (Passive) -
Performs FSimage and Edit logs merging to get new updated FSimage

Shared Folder is a shared place on some other machine
Where name node keeps on writing Fsimage and Edit logs and Secondary NN will keep on reading these 2 and will merge it and will overwrite the new fsimage in place of old fsimage.



Once the merger activity completed the Fsimage is replaced with new FSimage and Edit logs is reset to empty.

Checkpointing - the SNN performs FSimage and Edit logs merging to get new updated Fsimage.
This process repeats after <u>every 30 seconds</u>

== Primary NN failure | SNN will become as Active NN as SNN will load new fsimge in memory.
== Responsibility of Hadoop Admin to introduce a new SNN.

1. What is the default block size in hadoop 2? --128mb
2. Can it be changed? -- Yes. Mostly hadoop admins changes.
3. what will happen if you increase the block size? -- we will have less no. of blocks. If no. are less than we will use less number of nodes, so some nodes will be sitting idle.
   Conclusion -- if we inc the block size, it can lead to under utilization of cluster.
4. If we dec the block size to very small, we will have more number of blocks. We will have huge amount of metadata which NN has to handle, and this will overburden your NN.

== <u>The block size should neither be too high nor too low.</u>

HDFS Architecture Part-3

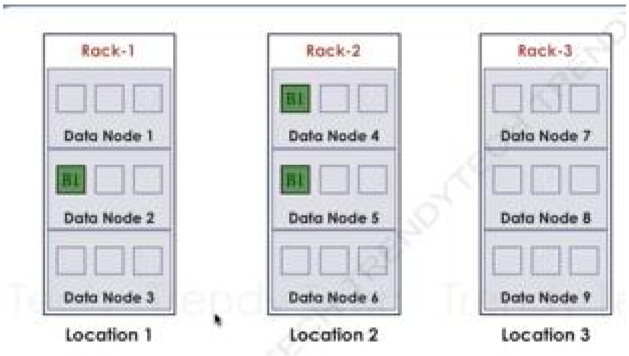
Rack Awareness Mechanism - Understanding Replication factor

-- Rack is a group of systems/servers kept in a geographical locations.
-- Name Node stores data block in a data node of a rack.

Scenario 1 [one rack 3 copies]
Replica Pipeline - Replicas of data block are forwarded to new locations. Forwarding data blocksrequires large amount of network bandwidth. It involves more input-output operations.
Rack Failure - If we place all replicas in a single rack, there are high chances of data loss if the rack goes down.

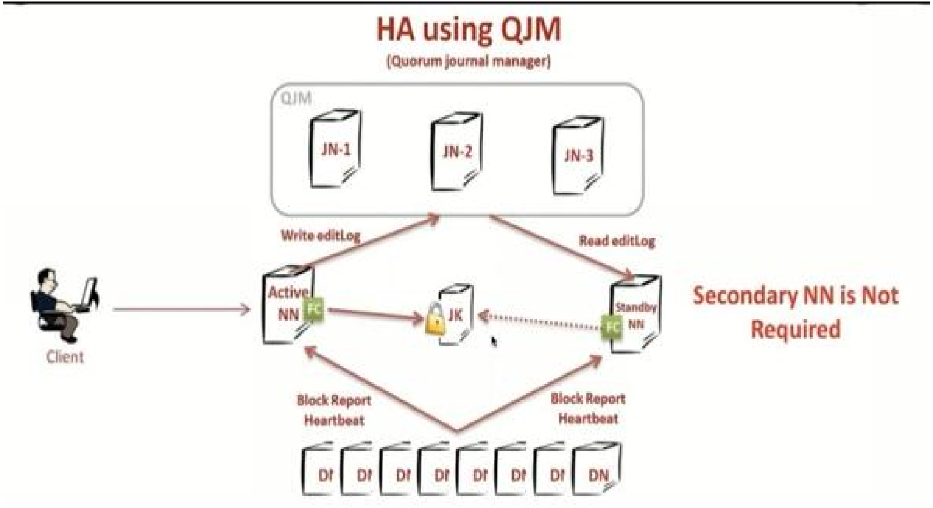
Scenario 2 [3 racks 3 copies]
Choosing Multiple Racks - Each Replica placed in different rack is also not a ideal solution. Because all three are on different geographic locations, it will take lot of bandwidth to go from one location to another.

Scenario 3 **[Ideal Choice - 2 racks 3 copies]**
Rack Awareness Mechanism - The balanced approach is to place replicas in two different racks.
One replica in one rack and other two in a different rack or vice versa. Mixed approach for best of both scenarios.



Block Report - each data node sends a block report to the name node at a fixed frequency indication if any blocks are corrupted. Then NN will try to delete this block and will add new record in metadata.

Two ways to achieve high availability of name node -
-- using shared location - Checkpointing
-- using **QJM (Quorum journal manager)** -- Quorum means group



JN - journal node (normal computers holding metadata)
Instead of shared location we have quorum(group) of journal nodes

Active NN ->JN1 ->JN2 ->JN3 -> Standby NN-> Jk(Joo Keeper)

JK is used when Active NN becomes dead and standby NN gets the lock and it will become active NN.

Hadoop Installation modes

3 Modes-

- ✓ Local Mode - Install only mapreduce related things on our pc. Not hdfs or yarn related.. not a good mode to use.
- ✓ Pseudo distributed mode - all 3 core components on a single machine [we will use this in this course]
- ✓ Fully distributed mode - more than one node/machine -- used in companies.

Important configuration files
Cd /etc/hadoop/conf
Ls -ltr
- hdfs-site.xml, mapred-site.xml, yarn-site.xml, core-site.xml

What is gateway/edge node?

- ✧ Edge node is a totally a separate machine.
- ✧ This node is configured in a way that it has the connectivity to hadoop cluster.
- ✧ This node provides you an interface to talk to the hadoop cluster.
- ✧ If in your company u req for hadoop cluster access. They willgive you acess to edge node.
- ✧ They will give IP address of edge node and you need to login to that with the right username and password.

HDFS commands

Help	Hadoop fs -help ls
Sort by name	Hadoop fs -ls /
By time in reverse order	Hadoop fs -ls -t -r /
By size	Hadoop fs -ls -S /
By size in human readable form	Hadoop fs -ls -S -h /
Home dir in local	/home/cloudera
Home dir in hdfs	/user/cloudera
List recursively	Hadoop fs -ls -R /
Search in the list	Hadoop fs -ls /user | grep cloudera
New dir	Hadoop fs -mkdir /user/cloudera/dir1
Hierarchy of dir	Hadoop fs -mkdir -p /user/cloudera/folder1/folder2
To remove files	Hadop fs -rm /user/cloudera/file1.txt
To remove a dir	Hadoop fs -rm -R /user/cloudera/dir1
To remove empty dir	Hadoop fs -rmdir /user/cloudera/dir1
Copy file/folder from local to hdfs	Hadoop fs -copyFromLocal /sourcedir /targetdir Hadoop fs -put /sourcedir /targetdir
Copy from hdfs to local	Hadoop fs -copyToLocal Hadoop fs -get
To view last 10 lines	Hadoop fs -tail <hdfs file path>
First few lines	Hadoop fs -cat <hdfs file path> | more
Copy file/folder form one hdfs loc to another	Hadoop fs -cp <hdfs file path> <hdfs loc 2>
Cut paste from one hdfs to another hdfs	Hadoop fs -mv <hdfs file path> <hdfs loc 2>
To check free disk space	Hadoop fs -df -h /user/cloudera
Check disk usage	Hadoop fs -du -h /user/cloudera
To summarize result	Hadoop fs -du -s -h /user/cloudera
Dynamically change replication factor	Hadoop fs dfs.replication=5
To see metadata in hdfs	Hdfs fsck
Check how many times a word is present in a file	Cat file.txt | grep hello | wc -l
To create a file in hdfs	hadoop fs –touchz folder_name/filename
Shared folder	Mount -t vboxsf shared shared