**Cloud Computing**

**Cloud computing**🡪 ability to access and use computer system resources (such as data storage and recovery, software, analytics, databases and networks) using remote servers instead of local ones or personal computers.

**Three main Branches of Cloud Computing**

|  |  |  |
| --- | --- | --- |
| **IaaS (Infrastructure as a service)** | **PaaS (Platform as a Service)** | **SaaS (Software as a Service)** |
| Data  Application  Services  Network  Database  OS  Storage/Memory  Hardware | Data  Application  Services  Network  Database  OS  Storage/Memory  Hardware | Data  Application  Services  Network  Database  OS  Storage/Memory  Hardware |

Made by us. Taken care for us by the cloud.

* IaaS🡪 Enables us to construct and control our own servers, data storage facilities, networks, OS, etc. We can buy the services from a provider in the cloud(cloud servers acts as a host that provides service through a dashboard or an application programming interface (API).

Examples of companies that offer IaaS= Microsoft Azure, Digital Ocean (slack, Gitlab, Hashicorp are partners), Amazon Web Services(AWS).

* PaaS🡪 Provides a cloud platform to develop and create software. Leases hard and software tools. It is basically for the programmers as it makes it easier for developers to construct custom applications without having to outsource for data storage and management. Base level platform that we can dump our code into and our code will run. We do not worry about the Ram, processor, updates to server, etc. Has Hadoop.

Examples of companies that offer PaaS= Microsoft Azure, AWS, Salesforce, Software AG, IBM Cloud, FutureFuel.io.

* SaaS🡪 Distributes software hosted by a third party provider to a user through the internet. Might require subscription and paying provider to gain access to this software. The software itself is running on servers in the cloud (not residing on our computer) then we just connect to it. Most popular among businesses. Easy to scale as business grows.

Examples of companies that offer SaaS= Squibler, Lumen5

**Online Processing Systems**

**OLTP –** Online Transaction Processing **OLAP** – Online Analytical Processing

|  |  |
| --- | --- |
| **OLTP** | **OLAP** |
| **-**Supports transaction-oriented applications in a 3-tier architecture.  -Administers day to day transaction of an organization.  -Primary objective = data processing  -Characterized by large number of short online transactions.  -Uses traditional DBMS(RDBMS-Oracle, MySQL, SQL server, PostgreSQL  -Transactional superiority instead of data analysis.  - | -Provides analysis of data for business decisions.  **-**Allows db information analysis from multiple db systems at one time.  -Primary objective = data analysis  -Characterized by a large volume of data.  -Data warehouse is created uniquely to integrate different data sources for building a consolidated database. |
| Example = ATM center, online banking, online airline ticket booking, sending a text message, order entry-update-delete-add an item. | Example = Any data warehouse system.  Company comparing sales, company analyzing purchases by customers. |
| Benefits = Administers daily transactions of an organization, widens the customer base of an organization by simplifying individual processes. | Benefits = Consistent information and calculation, security restrictions on users and objects ease, creates single platform for business analytical needs such as planning-budgeting-forecasting-analysis. |
| Downsides = Hardware failures severely affects online transactions, creates unprecedented situation when parallel accessing of data occurs. | Downside = Tools require a complicated modeling procedure, requires cooperation between various departments. |

**Scalability**

Scalability 🡪The capability of a system, network or a process to handle the growing amount of work in a capable manner. Can be applicable to both hardware and software.

Two types of scalability = Horizontal (Sharding) and Vertical

|  |  |
| --- | --- |
| **Horizontal Scaling** | **Vertical Scaling** |
| - Adding more machines to an existing stack. Adding multiple computing devices or nodes to the system to improve performance.  - In respect to data, it’s separating one table’s rows to multiple different tables (Partitions.) Each partition has same schema and columns but different rows, each data is unique.  - Ex. 12GB can be distributed throughout 3 machines with 10GB(Distributed storage)  -Used in distributed systems  -Cost-effective | -Adding more power(CPU, RAM) to an existing machine. Adding more resources to a single computing device to improve performance.  - Entire columns are separated and put into new distinct tables, Distinct rows and columns. Has limitations.  - Ex. On the same machine from 156GB to 500GB  - Used in virtualization  -Not as cost-effective |

Schema 🡪 structure or logical description that describes a database. It’s visual representation of different table relationships.

ETL 🡪 a type of data integration that refers to 3 steps. Extract, Transform, Load. It blends data from multiple sources. It copies data from one or more sources into a destination system that represents the data differently from the source or in different context than the source. Often used to build data warehouse.

Ex. Organization

HR team IT team Finance }each different data types

**ETL**

**S**torage

**Hadoop**

* A platform that is solution to handle Big Data.
* Distributes computing framework (DCF)
* Default application factor in Hadoop is 3.
* It’s an open source framework that is used to efficiently store and process large datasets.
* Instead of one large computer to store and process the data, it allows clustering multiple computers to analyze massive datasets in parallel more quickly.
* Various goals of Hadoop design = Fault tolerance, high availability, handling of large datasets, data locality, portability across heterogeneous hardware and software platforms, etc.

|  |  |
| --- | --- |
| **Processing** | **Storage** |
| -Map reduce 🡪framework that helps programs do the parallel computation on data. Map task converts input data into dataset that can be computed in key value pairs. The output is consumed by reduce tasks to aggregate output and provide desired result.  -YARN (Yet Another Resource Negotiator) 🡪 Manages and monitors cluster nodes and resource usage. Schedules jobs and tasks. | HDFS (Hadoop Distributed File System) 🡪provides better data throughput than traditional file systems. Has high fault tolerance and native support of large datasets. Runs on standard or low-end hardware.  🡪splits the data unit into smaller units called blocks and stores them in a distributed manner. |

Storage type in windows = NTFS. Previously it was FAT32x but it is outdated.

Storage type in UNIX = ext2, ext3, ext4

Block 🡪 the smallest unit of storage on a computer system/ the smallest contiguous storage allocated to a file. Default block size in Hadoop2.x = 128MB or 256MB.

Hadoop 1.x =64MB

Ex.

|  |
| --- |
| File.txt(700MB) |
| |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | | A(128MB) | B(128MB) | C(128MB) | D(128MB) | E(128MB) | F(60MB) | |

|  |  |
| --- | --- |
| 4kb | 4kb |

In unix 8kb 🡪

**Hadoop DAEMONS/PROCESSES**

Daemon 🡪 is a process.

Hadoop daemons 🡪 Hadoop processes.

All Hadoop daemons are java processes (b/c Hadoop is built using Java.)

To check the list of java processes running in our system we use = **jps** command.

Execute jps command – Hadoop daemons will run – Hadoop cluster is also running.

**Basic Hadoop Daemons:-**

1. Name Node – Metadata (Data about data)
2. Data Node – Physical Data
3. Secondary Name Node – Check pointing process. (Merging of Edit log file (In disk) and fs image file(in memory only)).

* Pushes whatever is in file to NN

1. Resource Manager – Resource Allocation
2. Node Manager – task execution. (responsible for executing your task on the slave machine)

We can find the above daemons in the ***sbin*** directory of Hadoop. Then we to start all the Hadoop daemons we use ***start-all.sh.*** command. To stop all the daemons we use the command ***stop-all.s.***

**Hadoop Architecture**

Hadoop has Master-Slave architecture.

Resource manager = cluster manager, communicates b/n master and slave

Master

N.N runs on master server= Namespace mgnt & regulates file access

RM

D.N runs on slave nodes= Stores actual business data

Slave1

Cluster

Slave2

Slave3

NN, SNN and RM run on Master server.

DN, NM run on slave nodes.

NN manages modifications to file system namespace. (opening, closing and renaming files or directories.)

NN has meta data. It is very important.

In HDP 1.x – N manager=task tracker and Resource manager= job tracker

N manager = components of YARN

Active status NN2 is copy of NN1. High availability backing up the NN1

NN1 Zoo keeper NN2 🡪 stand by status

|  |
| --- |
| S1 |
| S2 |
| S3 |

Zoo keeper is a coordinating system that connects the two NN  
 A backup node for metadata definition. Should have at least 3 nodes.

DN3

DN2

DN1

NN2

Zoo keeper

lock

NN1

Fail over control

Zoo keeper place a lock on NN1. If NN1 fails it removes the lock and makes NN2 active by locking it.

KEKBEROS – in Greek means a dog with 3 head

TGS- Ticket generating server

AS- Authentication Server

Tools available in Hadoop:-

SQOOP – Ingesting data to data link (RDBMS) -HDFS

HIVE – Data Warehousing in Hadoop OLAP. We use SQL commands.

FLUME – Near real time data integration tool

PIG-PIGLATIN – Scripting language used for transformation language

HBASE – OLTP (ACID)

**Types of Data**

|  |  |  |
| --- | --- | --- |
| **Structural data** | **Semi Structural data** | **Unstructured data** |
| -Has schema  -Ex. Sql tables | -Some structured some unstructured  -Ex. Csv, Jason, xml files | -No structure at all  -Ex. Text, video, audio files |

All this data are sent and stored in HDFS(File system storage in Hadoop).

HDFS doesn’t know what type of file. It just stores data.