Cloud Computing (Class)

- **Cloud computing** -

Cloud computing is a service that lets you use any computing service such as a server, storage, networking, database & intelligence, right through your browser without owning anything.

Essential characteristics -

1) On-demand self service
2) Broad network access
3) Resource Pooling
4) Measured Service
5) Rapid Elasticity

Cloud Service Models

1) **Iaas** : Infrastructure as a service,
   - deliver computing infrastructure on demand
   - used by network architects
   - just provides server, networking and storage. O/s, app and middleware of user.

   eg Google computing Engine | Microsoft Azure ↑ Paas

2) **Paas** : Platform as a service
   - delivers tools used in development of apps
   - used by developer.
   - eg: Heroku, Google App Engine

3) **Saas** : Software as a service
   - host software make available for client
   - used by end users
   - eg: Facebook, gmail, MS office web.

Xaas : anything as a service

Cloud deployment models

- Private cloud
- Public cloud
- Hybrid cloud
- Community cloud

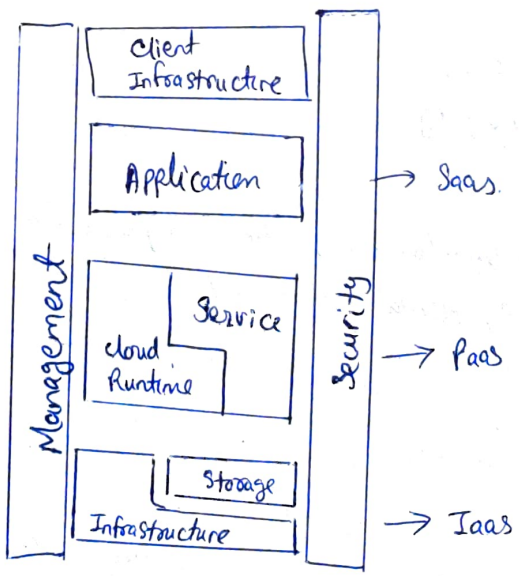
Virtualization

It is creation of a virtual (rather than actual) version of something, such as hardware platform, operating system, a storage device or network resources.

- Multiprogramming - Each process thinks it has complete control on all the resources.
  - Virtual Memory
  - CPU sharing

here CPU shared among processes while in virtualization it is shared among O/s.

Cloud Computing Architecture (Xaas)



CLOUD SECURITY

3 components

1) Confidentiality :
Keeping data and resources hidden

2) Integrity :
data or information in your system is maintained so that it is not modified or deleted by unauthorized parties
- Data integrity (integrity)
- Origin integrity (authentication)

3) Availability :
Enabling access to data & resources.

4 types of security attacks :

-> Interruption
-> Interception
-> Modification
-> fabrication

Classes of threats :

- Disclosure
  Snooping
- Deception
  Modification, spoofing, repudiation of origin, denial of receipt
- Disruption
  Modification
- Usurpation
  Modification, spoofing, delay, denial of service

Virtualization

- Components :
  -> Virtual Machine (VM)
  -> Virtual Machine manager (VMM) or hypervisor
- Two types :
  -> Full virtualization : VMs run on hypervisor that interacts with hardware.
  -> Para virtualization : VMs interact with host OS
- Major functionality :
  resource isolation

SLA

- A service Level Agreement (SLA) is the bond for performance negotiated between the cloud service provider & client.
- A formal contract between a service Provider (SP) and service consumer (SC)

SLA : foundation of consumer's trust in the provider.

Purpose : to define a formal basis for the performance & availability SP guarentees to deliever.

- SLA contains Service Level Objectives (SLOs)
  -> Objectively measurable conditions for the service
  -> SLA & SLO : basis of selection of cloud provider.

-> Contains :
- set of services provider will deliever.
- Responsibilities of provider & consumer.
- set of metrics to measure whether provider is offering services as guaranteed.

Service Level Objectives (SLO)

- Objectively measurable conditions for the services
- Encompasses multiple QoS parameters viz availability, serviceability, billing, penalties, throughput, response time or quality.

Broker for cloud Market Place

1) SMICloud : A framework for comparing and Ranking cloud services

High level attributes :
- Accountability
- Agility
- Cost
- Performance
- Assurance
- Security & Privacy
- Usability

Virtualization

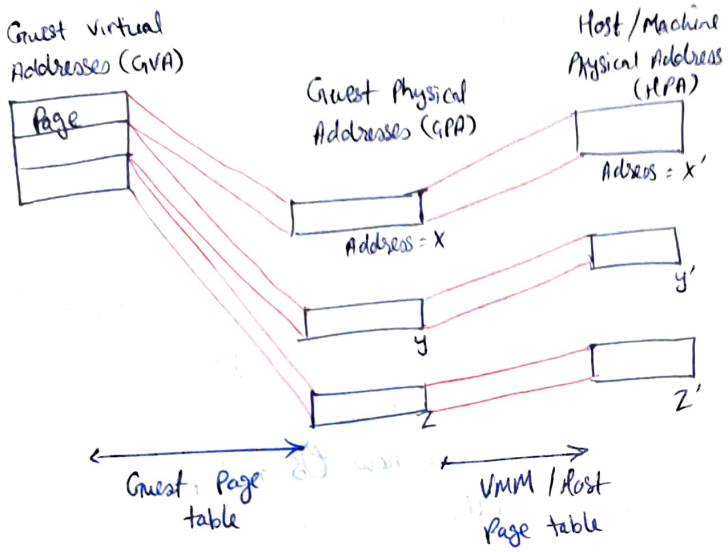
- VMM :- stands for Virtual Machine monitor. It is software program which allows management, governance and creation of VM (virtual machines) and also manages virtualized environment operation. It is also called Hypervisor & virtual machine manager.

→ Similar to how OS multiplexes processes on CPU.

Techniques to virtualize x86

- Paravirtualization :
  - Guest OS is not completely isolated but it is partially isolated by virtual machine from the virtualization layer and hardware.
  - category of CPU virtualization which uses hypercalls for operations to handle instruction at compile time.
  - Needs OS source code changes, cannot work with unmodified OS
    eg Xen hypervisor
- Full virtualization :
  - guest OS is completely isolated by VM from virtualization layer & hardware.
  - Dynamic (on the fly) binary translation, so works with unmodified OS.
    eg VMWare workstation

Memory Virtualization



- Guest Page table has GVA → GPA mapping
  - Each guest OS thinks it has access to all RAM starting at address 0.
- VMM / Host OS has GPA → HPA mapping
  - Guest "RAM" pages are distributed across host memory.

→ MMU : memory management unit is a computer hardware component that handles all memory & caching operations associated with processor.

- Shadow Paging : VMM creates a combined mapping GVA → HPA & MMU is given a pointer to this page table.
  VMM tracks changes in guest page table & updates shadow page table.
- Extended page tables (EPT) : MMU hardware is aware of virtualization, takes pointers to two seperate page tables.

EPT is more efficient but requires hardware support

I/O virtualization

- Guest OS needs to access I/O devices, but cannot give full control of I/O to any one guest OS.
- Two main techniques for I/O virtualization
  - Emulation : guest OS I/O operations trap to VMM, emulated by doing I/O in VMM / host OS
  - Direct I/O or passthrough : assign a slice of device directly to each VM.

Container

Container is a standard unit of software that packages up code and all its dependencies so that application runs quickly and reliably from one computing environment to another.

- Can be used to run anything from small microservice to larger application.
- Inside a container are all necessary executables, binary code, libraries and configuration files.
- Compared to server or machine virtualization approaches, however, containers do not contain operating system image. This makes them more light weight & portable, with significantly less overhead.
- For larger application, multiple containers may be deployed as one or more container clusters. Such cluster might be managed by a container orcheotrator such as Kubernetes.

Benefits :-

- Less overhead
- Increased portability
- More consistent operation
- Greater efficiency
- Better application development

Docker

It is popular runtime environment used to create & build software inside the containers. It uses Docker images (copy-on-write snapshots) to deploy containerized application or software in multiple environments, from development to test and production.

Kubernetes

Kubernetes orchestrates the operation of multiple containers in harmony together. It manages areas like use of underlying infrastructure resources for containerized applications such as amount of compute, network & storage resources required. orchestration tools like Kubernetes make it easier to automate & scale container based workloads for live production environment.

④ GREEN CLOUD COMPUTING

- Green cloud computing : Making the practices and approaches of using technological advances like computing and IT resources sustainable for potential environment benefits.

Green computing approaches :

1. Virtualization & use of terminal Servers
   Virtualization is a process wherein multiple operating systems run on computer system at same time. The application running appear as if they have their machine. Use of common servers and sharing the terminals found to save 80% energy.
2. Power supply and power managment
   Electricity transmission wasted 40% power & utilized only 60%. The power managment in cloud computing using green algorithm is reducing power consumption by computers.
3. Reducing e-waste generation

Applications in green cloud computing

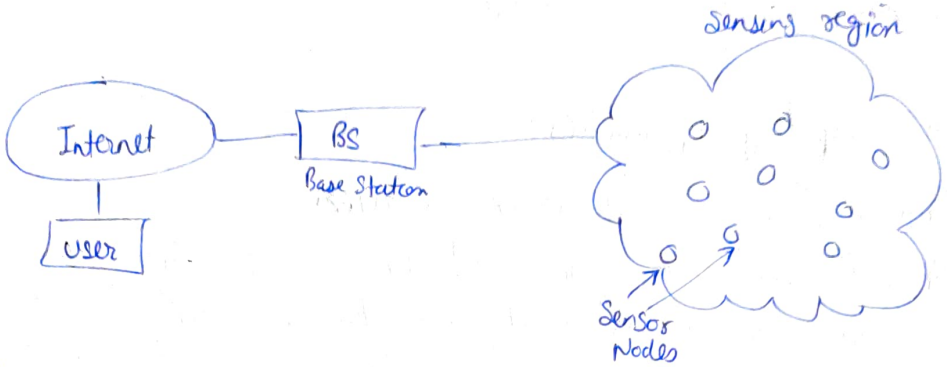
used in

- Managment of energy in Data centers
- Green wireless Network
- Green parallel computing with Big Data Network
- Green computing with an algorithm

- **Disadvantages**
  - Implementation cost is high
  - Evolving Technology will be challenging to adapt to
  - Green computers may be considered underpowered

Wireless Sensor Network (WSN)

It can be defined as network of devices that can communicate the information gathered from monitored field through wireless network.



WSN can be used for processing, analysis, storage and mining of data.

Application of WSN

- Internet of things (IOT)
- Surveillance and Monitoring for security, threat detection
- Environmental temperature, humidity & air pressure
- Landslide detection
- Medical applications like patient monitoring

Sensor Cloud Computing

Sensor cloud is new paradigm for cloud computing that uses physical sensors to accumulate its data and transmit all sensor data into cloud computing infrastructure.

virtual Sensors

Virtual sensor is a pure software sensor which autonomously produces signals by combining & aggregating signals that it receives from other physical sensors.

Virtual sensors mimic physical sensors by processing data originally gathered by physical sensors.

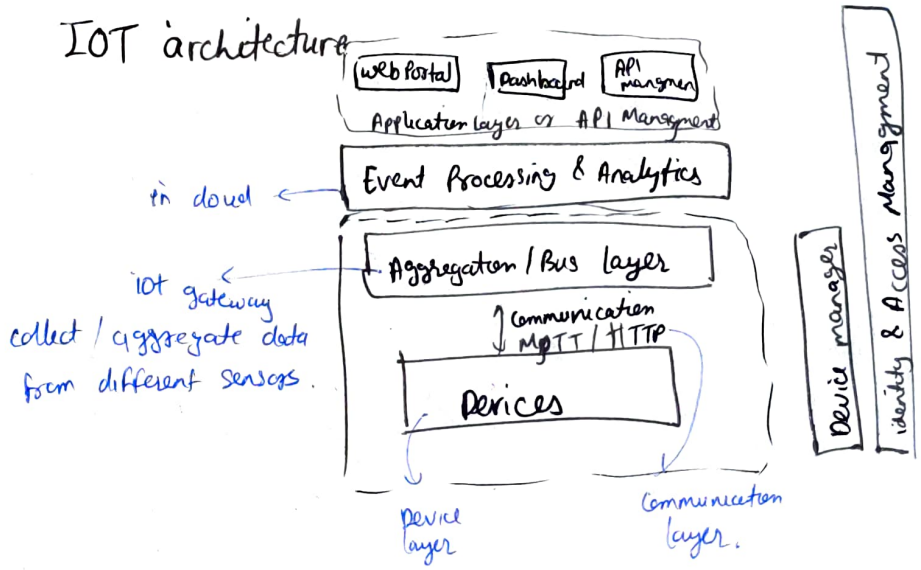
IOT cloud

IOT describes network of physical objects - 'things' - that are embedded with sensors, software and other technologies for the purpose of connecting and exchanging data with other devices and systems over the internet

IOT devices include wireless sensors, softwares, actuators, computer devices and more. They are attached to particular object that operates through internet, enabling transfer of data among objects or people without human intervention.

eg i) Smart AC sends its condition data to customer service center which help customer service to recommend you to fix AC issues when it has some problem.
2) Location tracking (GPS)

IOT architecture



Basic building Blocks
1) devices (Things)
2) Gateway
3) Network and cloud

IOT cloud platform

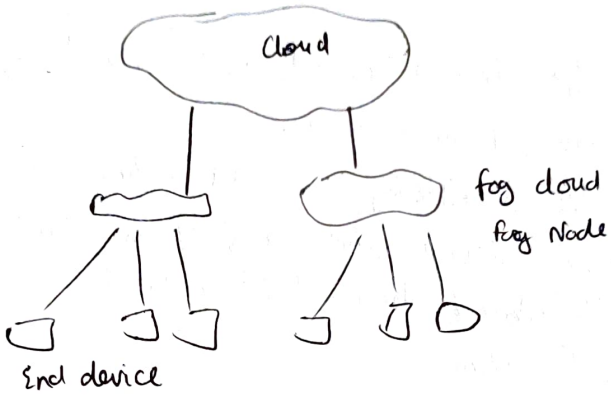
IOT cloud platforms bring together capabilities of IOT devices and cloud computing delivered as a end-to-end service. Typical features include connectivity & network management, device managment, data acquisition, processing analysis and visualization, application enablement, storage & integration.

→ they vary from normal cloud as they have better capability to work with real time data generated by different devices and to handle big data.

FOG COMPUTING

Fog computing, also called fog networking or fogging, describes a decentralized computing structure located bet cloud & devices that produce data

This data in fog cloud is analyzed locally, filtered and then sent to cloud for long-term storage if necessary.



When to use fog computing

1. It is used when only selected data is required to send to cloud. This selected data is chosen for long-term storage and is less freqeently accessed by host.
2. It is used when data should be analyzed within fraction of seconds. Time-sensitive data like alarms, fault warnings, & device status greatly benefits from speed of fog computing.
3. Devices that are subjected to rigorous computations and processings must use fog computing.

fog computing vs edge computing vs cloud

- edge computing helps devices to get faster results by processing data simultaneously received from devices.
- fog computing helps in filtering imp. information from massive amount of data collected from device and saves it in cloud by sending filtered data. Fog is distributed decentralized network.
- Cloud is centralized system

Advantages of fog computing

- This approach reduce the amount of data that needs to be sent to cloud.
- Since distance to be travelled by data is reduced, it results in saving network bandwidth.
- Reduce response time of system
- It improves overall security of system as data resides close to host
- It provides better privacy as industories can perform analysis on their data locally.

Disadvantages

- Congestion may occur betn host & fog node due to increased traffic
- Power consumption increased as one more layer of devices added
- Scheduling tasks betn host & fog nodes along with fog nodes & cloud is difficult.
- Data managment become tedious as along with data stored & computed, transmission of data involves encryption - decryption too which in turn release data.

5. Hadoop

BIG Data

Big data is a collection of data that is huge in volume, yet growing exponentially with time

5V's of big data :-

1) Volume : Large volume of data
2) Velocity : velocity refers to high speed of accumulation of data.
3) Variety : Refers to various variety of data
    1) Structured : eg excel
    2) Semi-structured : eg Log files, json file
    3) Unstructured : Text, picture, videos
4) Veracity : it refers to inconsistencies & uncertainity in data, that is data which is available can sometimes get messy & quality and accuracy are difficult to control
5) Value : Data itself is of no use or importance but it needs to be converted into something valuable to extract information.

Big Data Challenges

1) Scalability : processing should scale with increase in data
2) Fault tolerance : fn in presence of hardware failure
3) Cost Effective : Should run on commodity hardware
4) Ease of use : program shalld be small
5) Flexibility : able to process unstructured data

Big data → frameworks → Cassandra
→ Hadoop
→ Apache spark

Hadoop

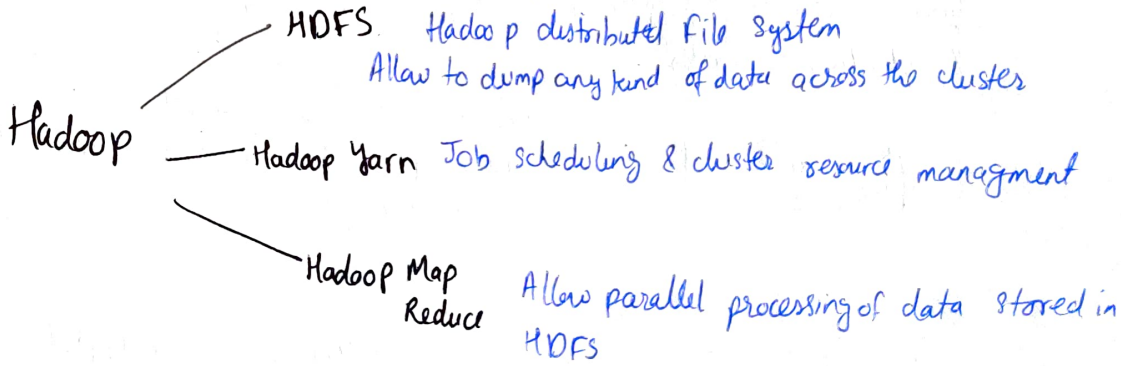
Hadoop is a framework that uses distributed storage and parallel processing to store & manage big data.

Problems with traditional approach :-

1) Storing huge & exponentially growing datasets
2) Processing data having complex structure
3) Bringing huge amount of data to computation unit becomes a bottleneck

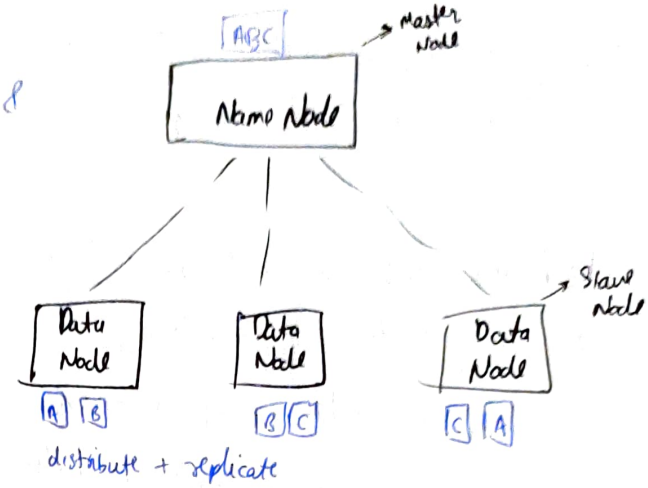
Hadoop as a soln :-

1) HDFS provides a distributed way to store Big data
2) HDFS can store all kinds of data whether it is structured, semi-structured or unstructured
3) Provides parallel processing of data present in HDFS. Allows to process data locally ie each node works with a part of data which is stored in it.



HDFS (Hadoop distributed file system)

- Storage unit of Hadoop
- Divide file into smaller chunks & stores it across cluster
- Store any kind of data
- Horizontal scaling as per requirement (to add more data add new data node)
- Replica of data is created to deal with faults occuring

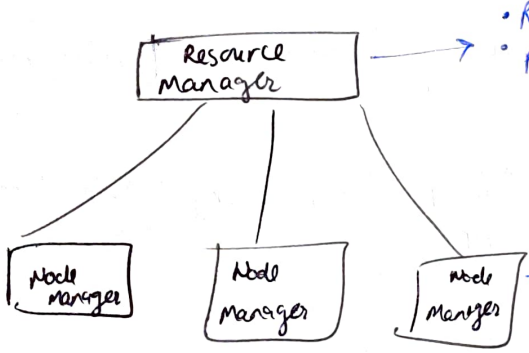


Name Node :- is heart of an HDFS filesystem, it maintains & manages the file system metadata eg: what blocks make up a file, and on which datanodes those blocks are stored, size etc.

Data Node :- Where HDFS stores the actual data.
- Serves read & write requests

YARN (Yet Another Resource Negotiater)

Yarn performs all your processing activities by allocating resources & scheduling tasks

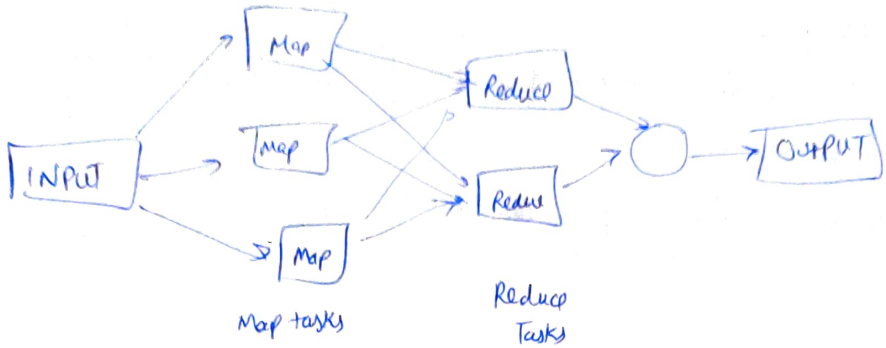


- Receives processing requests
- Passes parts of requests to corresponding Node manager

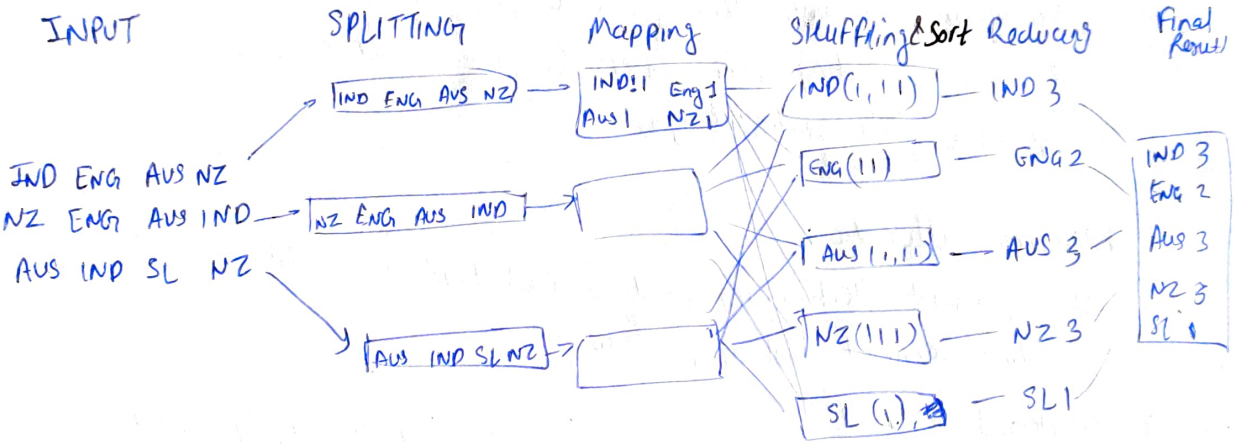
- installed on every Data Node
- Responsible for execution of task on every single Data Node.

MapReduce

MapReduce is a software framework which helps in writing applications that process large data sets using distributed & parallel algorithm using Hadoop environment



eg



1) Cloud Resource Managment

Resources types

1) Physical Resource
   - Computer, disk, database, network, scientific instruments
2) Logical resource
   - Execution, monitoring, communicate application

How to conserve energy with a cloud?

- Schedule VMs to conserve energy
- Managment of both VMs & underlying infrastructure
- Minimize operating inefficiencies for non-essential tasks
- Optimize data center design.

1. Resource allocation
   It represent techniques that is used to optimize utilization of resources & reduce required cost of processing.
2. Workload Balance
   it is used to manage energy effectiveness and also avoid congestion, low-load resource managment & overload.
3. Resource Provisioning
   It represents an approach that shows how to administer requests for tasks & data among Nodes.
4. Task Scheduling:
   To manage large set of tasks that are working together & are dependen on certain set of resources.

⑥ Mobile Cloud Computing (MCC)

MCC is defined as a combination of mobile computing, cloud computing & wireless network that come up together purpose such as rich computational resources to mobile users, network operators as well as to cloud computing providers.

In this technology, data processing & data storage happen outside of mobile devices

Advantages

1. Extended battery life
2. Improvement in data storage capacity & processing power
3. Improved synchronization of data due to 'store in one place, accessible from anywhere' platform theme.
4. Improved reliability & scalability
5. Ease of Integration.

Dynamic Runtime Offloading

It basically means the module may be transferred for operation & execution in the cloud when the application is running. It occurs when the program is partitioned at runtime because of network latency, it is refered as dynamic offloading

It involves the issue of :-

- Continuous synchronization for entire duration of runtime execution platforms
- dynamic application profiling & solver on SMP
- runtime application partitioning
- migration of intensive component

MCC challanges

1) Security & Privacy for Mobile users
2) Security for mobile users
3) Content-aware mobile cloud services
4) Network access managment
5) Quality of services
6) Pricing
7) Standard interface
8) Service Convergence

Cloud Storage Systems:

1) Dynamo

- High-performance No-SQL key-value store from Amazon.
- Map a key to blob of unstructure data, stored across multiple nodes
- Highly available
  high performance
  highly scalable
- Weak consistency: a get may not always return latest value put in past
  - No ACID property
  - A get may also return multiple conflicting values

2) Bigtable: semi-structured data store used at google.

Built over 2 other systems from Google: Chubby (a distributed locking/consensus system) and Google file system (distributed file system)

- Tables of rows, columns (string)
  Each cell also has timestamp.
  Maps row key, column key & timestamp to a value (array of bytes)
  (row: string, column: string, time: int64) → string
  - row: stored alphabetically
  - time: timestamp act as version number
- group of adjacent rows is tablet
- SSTable → file format to store Bigtable data internally
  Immutable map of key-value pairs, stored on sequence of blocks on disk. Block stored on GFS

Chubby : distributed lock service / distributed directory service

- Exposes a namespace of directories & files
- Users of chubby can acquire a lock on screen directory / file & read / write its content

Haystack

- cloud storage system for efficiently storing & retriving photos at facebook

→ minimal metadata per photo, so that all metadata in memory, only one disk access even for unpopular photos

3 components : Store , Cache , Directory

Store : has actual photoes

Cache : caches popular content that is not already cached in CDNs

Directory : maintains location mapping (which CDN / cache / store / logical volume may have photos)

Memcache :

Distributed cache used at facebook to cache queries from database servers.

Cache sits betn web/app servers & backend databases.

- Cache results of queries to backend databases

Front end cluster
- web server (memcache clients)
- Memcache (memcached servers)

Backend storage cluster
- MySQL databases