Big Data Challenges

Key Concept: Resource requirements for Big data

As data complexity, size & variety increases more resources are required

Resources

(1) Storage : HDD, SDD, CD
(2) Memory : RAM for processing tasks
(3) Performance : CPU cores for computation

System Architecture for Big Data

Monolithic Systems

* Single System

↑ RAM Storage
64GB , 2TB

Limitations

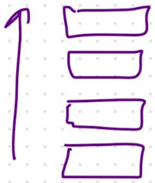
* Hardware constraints (max Ram or Storage capacity)

efficiently

Vertical Scale

32GB

8X	4P	— 6
16x	5P	— 5
32x	6P	— 4
	10P	
64X	6P	

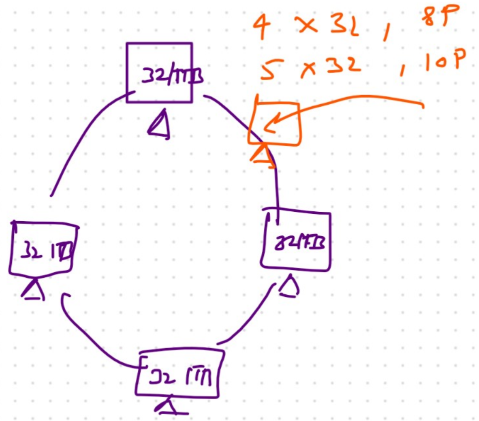


* performanc does not scale proportionally with resource upgrades

* Not scalable for big data problems

Distributed System

* multiple System



* resources are distributed across nodes

* easily scalable by adding more machines

* achieves true scaling

horizontal scaling

Scaling ∝ increase performance

Example Burners : 4 chefs 8 chefs
resource : data process

① Monolithic : A single kitchen with many chefs, adding more chefs eventually reduce efficiency due to space & workflow constraints

② Distributed System : Multiple kitchens, each chef where specialized in a cuisine. Adding more kitchens to increase efficiency & scalability

Key Takeaways

Monolithic System :

- limited scalability
- Not suitable for big data problems

Distributed Systems :

- Enabled scalability & efficient resource utilization
- foundation for all good big data systems

Big Data System Deployment : On-premise vs Cloud Solution

On-premise Infrastructure :

Characteristics

- Similar to buying an office / house
- high initial capital expendentiture (CapEx)
- Organizations must procure all hardware
  (eg 20-node cluster ⟶ purchasing 20 machines
- Organizations is responsible for all the setup & maintenance

Key Factors

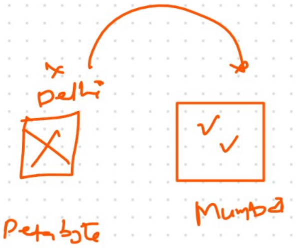
Deployement : Hardware & s/w hosted with in org facilities

Cost : high upfront cost for
- hardware
- maintenance
- IT staff

Scalability: Limitations by org hardware capacity

Maintenance: Org is responsible for managing

- hardware (failure)
- Software updates



93·1. **flexibility:** due to hardware resources

Security: Data remains within org : greater control

Disaster Recovery: requires internal backup & disaster recovery systems

Cloud Solutions characteristics:

- similar to renting a house/ co-working space
- pay-as-you-go model
- No ownership of hardware
- Access to machines via internet

Major Cloud Provider

- AWS
- Azure
- GCP
- IBM Cloud

Deployement: resources are hosted on provider's servers, we just need to access via internet

Cost: pay-use model ⟶ Operational expenses

Scalability: highly scalable, add resources on demand instantly

Maintenance: managed by cloud providers

flexibility: high flexible allows resources to scale up & down as needed

Security: managed by provider, concerns on sensitive data

Disaster Recovery: builtin-recovery & redundancy across multiple locations

Types of Cloud:

200 M → 400 M

↓

(1) Public cloud :

(1) AWS, Azure, GCP, IBM

(2) ideal for startups due to no initial infrastructure cost

(2) Private Cloud : VMware, OpenStack

ex: banks → ensure security & regulatory compliance

* different from on-premise by company cloudlike UI & management

(3) Hybrid Cloud :

→ combines public & private cloud

→ private cloud for certain usage / sensitive data

→ public cloud, for computations

→ Balances security & scalability need

(4) Community Cloud:

* Infrastructure shared by multiple organization with common concern

* Universities & hospitals share data & resources

* limited to specific community member