

# Introduction to Cloud Computing

## Summary

- Need for Cloud computing
- Cloud computing Architecture
- Cloud Services
- Possible challenges related to parallel processing
- *Wolfson* et al optimal data replication strategy
- Types of data replication
- Current trends in Cloud Computing

# Need for a new technology!

- Super processing power
- High Scalability with minimal additional cost
- No maintenance
- Fault-tolerance
- Pay for what I only consume
- Increase my sales and not invest on resources, thereby reducing the risk for a particular business idea
- No need to configure, install, upgrade, run a complex stack of softwares to support the idea
- Need for your idea to be up and running in few days

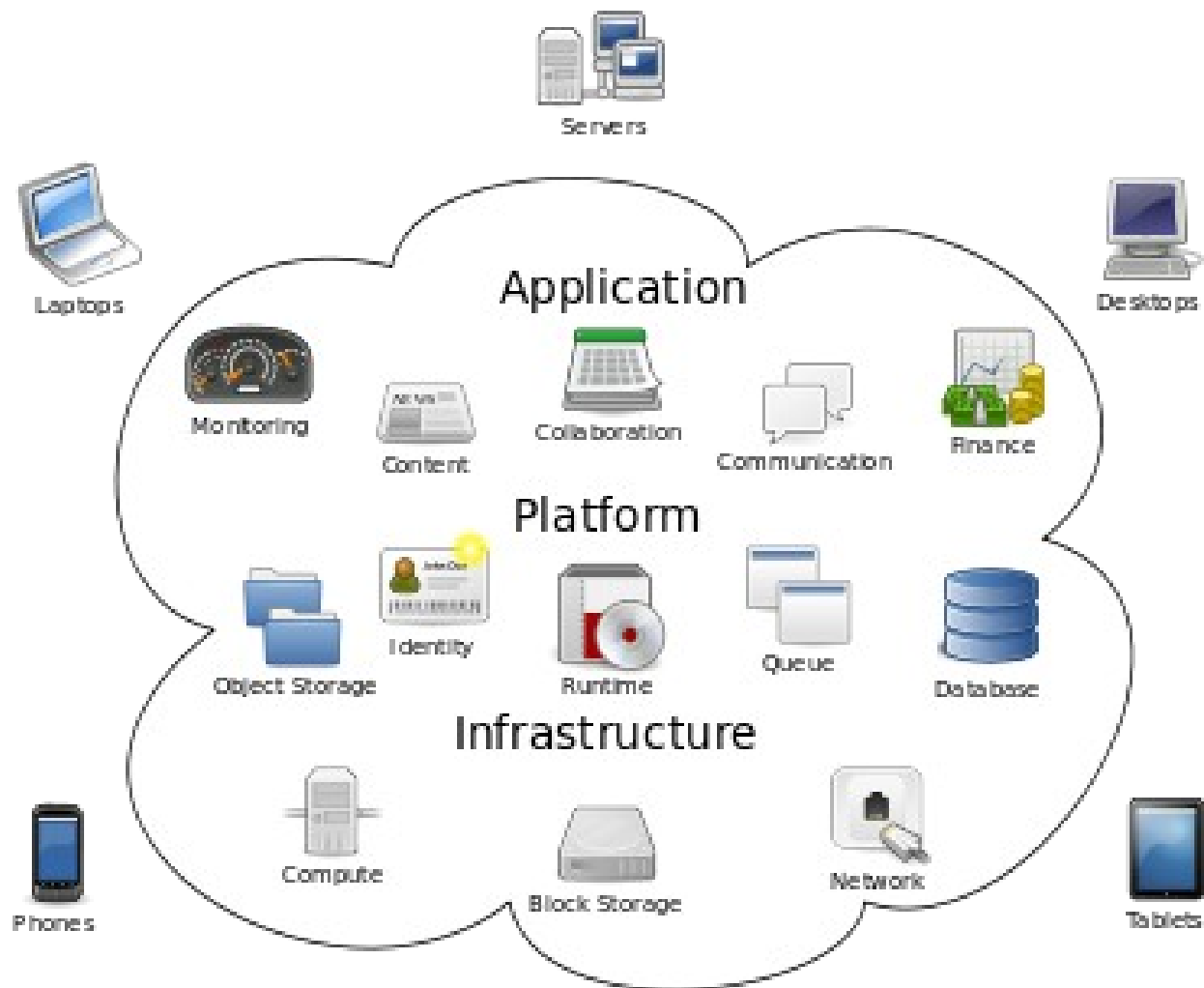


Power of internet !

# Cloud Computing

What is cloud computing ?

Cloud computing is the use of computing resources(hardware and software) that are delivered as a service over a network(typically the Internet)



# Cloud Computing

# Cloud Services

- Infrastructure as a Service
- Platform as a Service
- Software as a Service
- Storage as a Service
- Security as a Service
- Data as a Service
- Database as a Service
- Analytics as a Service

# Disadvantages despite several advantages

- Data insecure in public cloud
- No stringent standards yet

# Need to know terms

- Public cloud
- Private cloud



# Possible challenges related to Parallel Processing in Cloud Computing

- Resource optimization
- Managing data centers, moving rarely accessed data to highly compressed and low cost devices.
- Data consistency
- Data currency

*Wolfson* et al optimal  
data replication  
strategy

# Problems with Data Replication

Consider a weighted graph  $(N,L)$ , wherein  $k$  users are situated at some  $N_k \subseteq N$  nodes, and  $r$  replicas of a data item can be placed at some  $N_r \subseteq N$

- The Questions which arises is:

What is the optimal placement of the replicas if

- $k > r$  and
- the users access the data item in read-only mode?

# Problems with Data Replication

A Possible solution :

- Evaluating all placements of  $N_r$  among the nodes in  $N$  to identify

$$\min(\sum_{i \in N_k, r_i \in N_r} dist_{i,r_i})$$

Where  $dist_{i,r_i}$  is the cost from node  $i$  to  $r_i$ , the replica nearest to  $i$

# Problems with Data Replication

- If we assume that the read accesses from each of the users in  $N_k$  have a certain frequency (or weight), the minimization function would change.
- Bandwidth/capacity associated with each edge would also be an important factor in identifying a feasible solution
- Assume that a user access to the shared data is a
  - Read operation with probability  $x$ , and
  - An update operation with probability  $1-x$ .

An update operation also requires all replicas to be updated.

What is the optimal placement of the replicas if  $k > r$ ?

# Problem Definition

- Define a replication scheme as a subset  $R$  of  $V$  such that each node in  $R$  has a replica of the object.
- Let  $r_i$  denote the rates of reads issued by node  $i$  and  $w_i$  denotes the rates of writes issued by node  $i$
- Let  $cr(i)$  and  $cw(i)$  denote the cost of a read and write issued by node  $i$
- Let  $\mathcal{R}$  denote the set of all possible replication schemes

# Problem Definition

The goal is to minimize the cost of the replication scheme:

$$\min_{R \in \mathcal{R}} \left[ \sum_{i \in V} r_i \cdot c_r(i) + \sum_{i \in V} w_i \cdot c_w(i) \right].$$

# Some Basic Definitions

- Read Operation
  - A read operation is performed from the closest replica on the tree  $T$
  - If the node issuing the read query or receiving a forwarded read query is not in  $R$ , it forwards the query towards the nodes in  $R$  along the tree edges

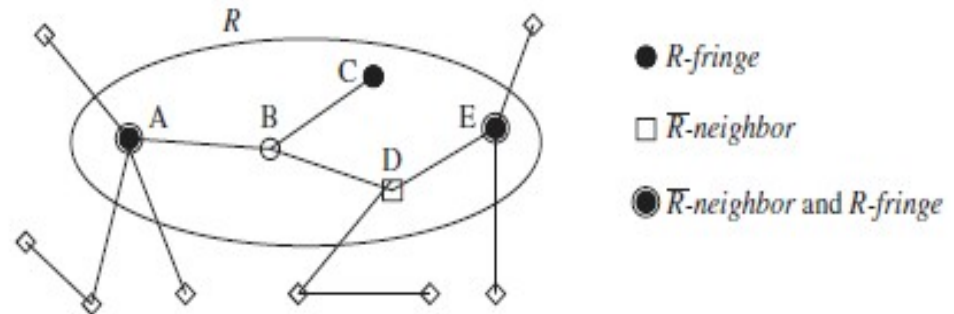


# Some Basic Definitions (contd)

- Write Operation
  - A write is performed to every replica in the current replication scheme R
  - If a write operation is issued by a node not in R, the operation request is propagated to the closest node in R
  - Once a write operation reaches a node  $i$  in R, the local replica is updated, and the operation is propagated to all neighbors of  $i$  that belong to R

# Basic Terminologies

- R-neighbor : Such a node  $i$  belongs to  $R$  but has at least one neighbor  $j$  that does not belong to  $R$ .
- R-fringe : Such a node  $i$  belongs to  $R$  and has only one neighbor  $j$  that belongs to  $R$ . Thus,  $i$  is a leaf node in the subgraph of  $T$  induced by  $R$  and  $j$  is the parent of  $i$ .



- *singleton*:  $|R| = 1$  and  $i \in R$

# Initial Information Required

Prior to actual implementation of algorithm, following information is required

- To determine whether a node is in R
- To find all neighbors of R
- For each neighbor, to determine whether it is in R

# Adjustment to Replication scheme using Three Tests

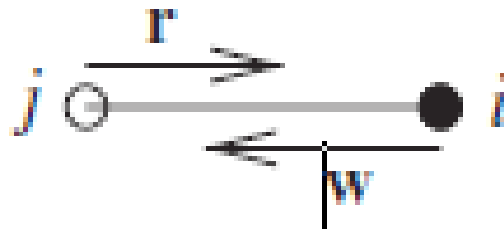
- Expansion test
- Contraction test
- Switch test

# Expansion Test

An *R-neighbor* node  $i$  examines each such neighbor  $j$  to determine whether  $j$  can be included in the replication scheme.

The test succeeds if

- Volume of reads coming from and via  $j$  is more than the volume of writes that would have to be propagated to  $j$  if  $j$  is included in the replication scheme.

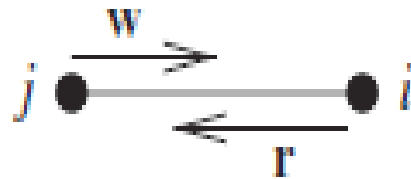


# Contraction Test

An *R-fringe* node  $i$  examines whether it can exclude itself from the replication scheme

The test succeeds if

- If the volume of writes being propagated to it from  $j$  is more than the volume of reads that  $i$  would have to forward to  $j$  if  $i$  were to exit the replication set

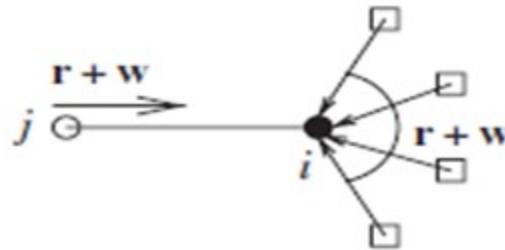


# Switch Test

A *singleton* node  $i$  executes the *switch test* to determine if it can transfer its replica to some neighbor to optimize the objective function.

The test succeeds if

- Volume of requests being forwarded by that neighbor is greater than the volume of requests the node would have to forward to that neighbor if the replica were shared with that neighbor



# Types of Data Replication

- Active replication :

Request processed by all replicas. It requires a atomic broadcast to all the replicas. This makes this method costly as it requires heavy communication for achieving a distributed consensus so that the same order is seen by all replicas.



# Types of Data Replication

- Primary back-up replication :

Updates are made on a single replica and are propagated to remaining replicas, while read operations can be directed to any node.

## Propagation :

- Eager : Master replies only after update has been propagated to all replicas. Strong consistency.
- Lazy : Master replies after update has been made locally.  
other unupdated replicas are stale.

Disadvantage?

# Types of Data Replication

- Multi-Master replication :  
Write and read operations can be made at any replica.  
Propagation :
  - Eager : All replicas must synchronize in order to decide on a single ordering. Hence, heavy communication.
  - Lazy : Possibility of conflicts.
- Advantage : Allows a better distribution of write operations on replicas.
- Disadvantage : Imposes a higher level of communication that leads to large number of messages.

# IaaS

- Stands for Infrastructure-as-a-Service.
- Cloud service where the user is provided computers and resources for storage and network.
- Other resources include block or file based storage, VLANs, load balancers, IP Addresses.
- User has to take care of patches and maintains OS.
- Storage-as-a-service is a part of IaaS managing the storage services.
- Examples include Google App Engine, Amazon Cloudformation, Rackspace Cloud.

# Motivations for IaaS

- Cost savings on hardware/infrastructure
- Capacity management
- Cost savings on IT staffing/administration
- Risk of hardware failure

# Service Provisioning

- Requests come in the form of different types of operating systems, storage sizes, network bandwidth, requests at different times.

Scenarios for providing services.

- Allocate a new physical machine for each user.
- Prepare a pool of pre-installed machines for different requests.
- Both lead to trouble.
- Best way is Virtualization

# Pros and Cons

- Advantages
  - Avoid capital expenditure on hardware and human resources.
  - Reduced ROI risk.
  - Low barriers to entry.
  - Streamlined and automated scaling.
- Disadvantages
  - Business efficiency and productivity largely depends on the vendor's capabilities.
  - Potentially greater long-term cost.
  - Centralization requires new/different security measures.

# Example

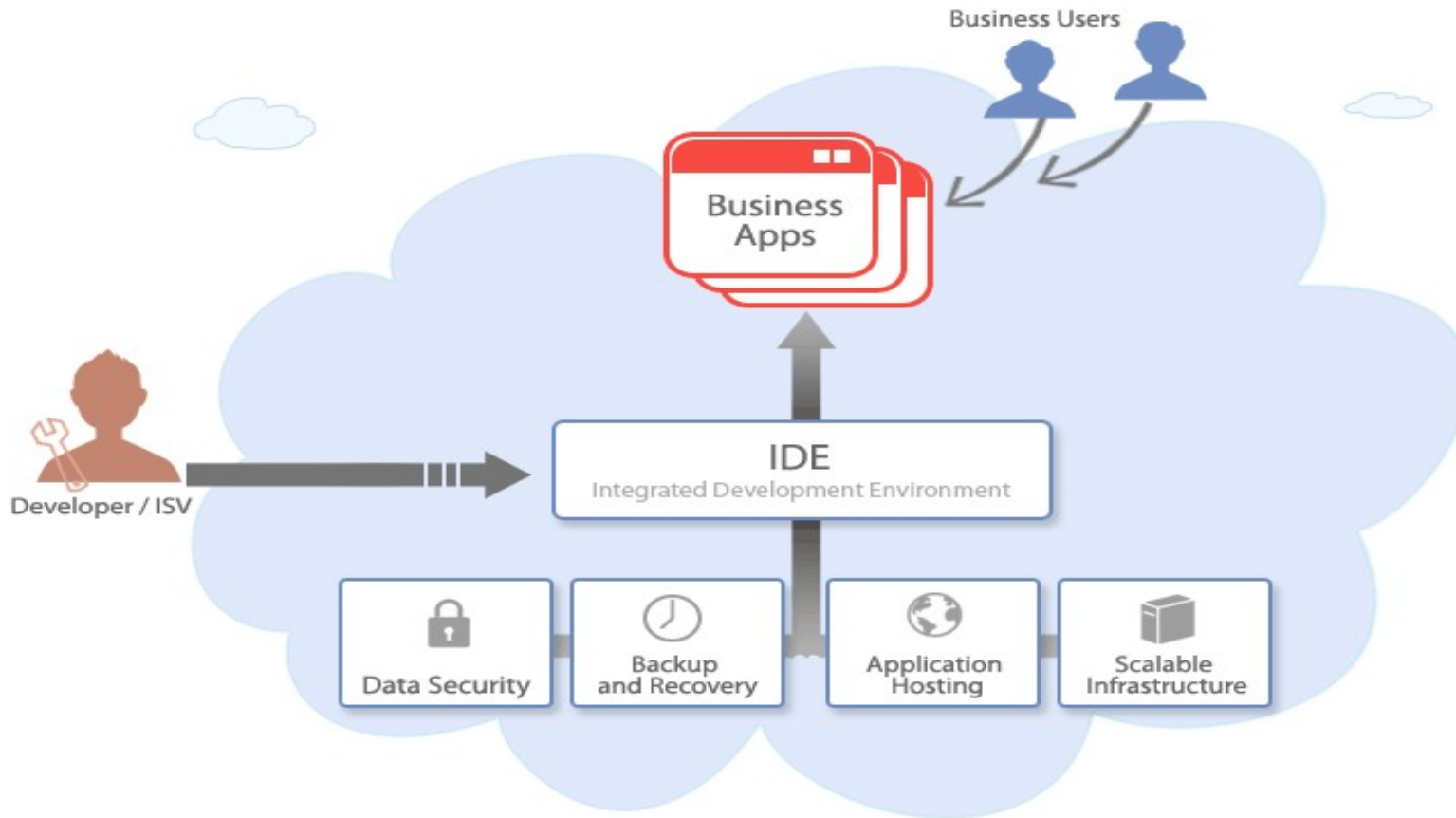
- Microsoft Azure
  - Create a VM
  - Create a virtual network.
  - Associate address space to it.
  - Create subnets
  - Setup DNS servers
  - Setup connectivity with on-premise network.

# PaaS

- Acronym for Platform-as-a-Service.
- Cloud service where the user can run existing applications and develop and test new ones.
- Clients are provided with servers, storage, networks and other services.
- Includes facilities for application design and development, testing and deployment and many more.
- Facilities are provided over the web.



# PaaS Characteristics



# PaaS Characteristics contd...

- Multi Tenant Architecture
- Customizable UI
- Unlimited Database Customizations
- Robust workflow engine
- Granular control over security
- Flexible Integration model

# Pros and Cons

- Advantages

- Geographically distributed teams can collaborate with much ease.
- OS features can be changed and upgraded frequently.
- Initial and ongoing costs are reduced by using services from single vendor than having multiple hardware facilities
- Cost related to electricity to power the data centers and to keep them cool are also reduced.
- Businesses need to worry about application development and completely do away with network and hardware management.

- Disadvantages

- Users gets locked in there is a requirement of some proprietary service interfaces.
- Flexibility in services may not meet some of the needs of users whose requirements rapidly evolve.

# Example

- Google App Engine
  - Run web applications on Google's infrastructure and easy to scale data and traffic
  - Sandbox – Isolates the application in its own secure, reliable environment which is independent of hardware, operating system and physical location of web server.
  - Application environment – Java and Python
  - Data Store – a powerful data store, can scale as required, has a query engine and transactions, uses optimistic concurrency control

# SaaS

- Abbreviation of Software-as-a-Service.
- Also known as on-demand service in which the software and associated data are centrally hosted on the cloud.
- Common delivery model for many business applications including accounting, HRM, CRM, MIS, ERP.
- Stats
  - Gartner group estimate SaaS sales in 2010 reached \$10 billion
  - Revenue will be more than double its 2010 numbers by 2015 and reach a projected \$21.3 billion.

# Characteristics

- Configuration and Customization
- Accelerated feature delivery
- Open integration protocols
- Collaborative and social functionality

# Pros and Cons

- Advantages
  - Less initial investment and less risk.
  - Immediate updates and new features.
  - Cost reduction, in addition to paying for only what you need.
- Disadvantages
  - Low confidence in data security.
  - Latency is introduced as applications hosted in the cloud are far away from users.
  - Integration with the rest of the systems applications.
  - Data transfer happens at internet speeds.

# Recent Trends in Cloud Computing

- Data Centers are expensive and known to be power consuming
- As cloud is becoming ubiquitous there is a need for internetworking in cloud.
  - Includes support for market oriented resource management
  - Improve mechanisms for allocation of VM resources
  - Interaction protocols should be extended to support interoperability between different cloud environments



# Parallel Data Pipeline

- Parallel operation cannot be broken down into a single Map-Shuffle-Reduce operations.
- Need for many such operations is needed resulting in a pipeline of MapReduce operations.
- Such pipelines require additional co-ordination in chaining these MapReduce operations, create and delete intermediate results

# Google Flume Java

- `PCollection<T>` class
  - has the immutable bag of elements.
  - Can be ordered or unordered
- `Ptable<K,V>` class
  - Subclass of `Pcollection`
  - Unordered bag of pairs
- `Paralleldo()` is called to invoke data-parallel operations.  
Maps the `PCollection<T>` to `PCollection<S>` using `Ptable<K,V>`

# Deferred Evaluation

- Calling a parallel operation does not actually execute the operation.
- Plan for the entire computation is generated
- Plan is optimized and then executed.
- While executing different strategies are considered (local sequential loop or remote parallel MapReduce)

# Optimizer Phase

- Fusion
  - If result of `parallelDo()` function is consumed by another `parallelDo()` then they can be fused together.
- MapShuffleCombineReduce
  - Combinations of `parallelDo()`, `GroupByKey()`, `CombineValues()` and `Flatten()` are mapped into one MapReduce

# Amazon RedShift

- Datawarehouse as a service
- Avoids setting up, running and scaling a datawarehouse
- Getting fast performance requires mastery over the indexing mechanisms, complicated query plans, access methods
- Costs involved is of the order of \$1000 per terabyte

# Architecture

- Can be single node or multi node cluster
- If multi node cluster then there should be a leader node. Other nodes are called compute nodes
- Leader node is connected via JDBC or ODBC endpoints
- Retains data integrity during node or disk failure.
  - Maintains 2 copies of data.
  - Monitors health of drives and moves if there is a problem.

# References

- Cloud computing and emerging IT platforms: Vision, hype, and reality for delivering computing as the 5th utility Buyya et al.
- <http://www.zoho.com/creator/paas.html>
- [http://en.wikipedia.org/wiki/Platform\\_as\\_a\\_service](http://en.wikipedia.org/wiki/Platform_as_a_service)
- <http://www.cloudbook.net/resources/stories/key-characteristics-of-paas>
- <http://searchcloudcomputing.techtarget.com/definition/Platform-as-a-Service-PaaS>
- [http://en.wikipedia.org/wiki/Software\\_as\\_a\\_service](http://en.wikipedia.org/wiki/Software_as_a_service)
- <http://www.ibm.com/developerworks/cloud/library/cl-cloudservices1iaas/>
- Distributed Computing by Ajay D.Kshemkalyani and Mukesh Singhal
- Wikipedia - [http://en.wikipedia.org/wiki/Cloud\\_computing](http://en.wikipedia.org/wiki/Cloud_computing)
- Geo Replication in Large Scale Cloud Computing Applications – Sergio Garrau Almeida
- Cloud Analytics – P.Radha Krishna and Kishore Indukuri Varma
- Cloud computing introduction [https://www.youtube.com/watch?feature=player\\_embedded&v=ae\\_DKNwK\\_ms](https://www.youtube.com/watch?feature=player_embedded&v=ae_DKNwK_ms)