CLOUD COMPUTING

Cloud Computing - Overview

PROF.SOUMYA K.GHOSH

SILICON INSTITUTE OF TECHNOLOGY
SAMBALPUR

May 7, 2023

Introduction

The ACM Computing Curricula 2005 defined "computing" as
"In a general way, we can define computing to mean any goal-oriented activity requiring, benefiting from, or creating computers. Thus, computing includes designing and building hardware and software systems for a wide range of purposes; processing, structuring, and managing various kinds of information; doing scientific studies using computers; making computer systems behave intelligently; creating and using communications and entertainment media; finding and gathering information relevant to any particular purpose, and so on. The list is virtually endless, and the possibilities are vast."

Cloud Computing Course - Overview

- Introduction to Cloud Computing
  - Overview of Computing
  - Cloud Computing (NIST Model)
  - Properties, Characteristics Disadvantages
  - Role of Open Standards
- Cloud Computing Architecture
  - Cloud computing stack
  - Service Models (XaaS)
    - Infrastructure as a Service(IaaS)
    - Platform as a Service(PaaS)
    - Software as a Service(SaaS)
  - Deployment Models
- Service Management in Cloud Computing
  - Service Level Agreements(SLAs)
  - Cloud Economics
- Resource Management in Cloud Computing

Cloud Computing Course (contd.)

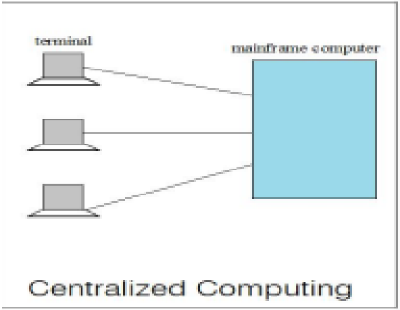
- Data Management in Cloud Computing
  - Looking at Data, Scalability Cloud Services
  - Database Data Stores in Cloud
  - Large Scale Data Processing
- Cloud Security
  - Infrastructure Security
  - Data security and Storage
  - Identity and Access Management
  - Access Control, Trust, Reputation, Risk
- Case Study on Open Source and Commercial Clouds, Cloud Simulator
- Research trend in Cloud Computing, Fog Computing

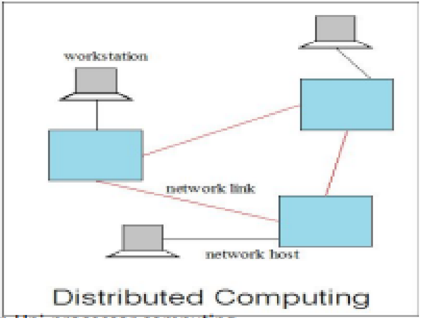
Trends in Computing

- Distributed Computing
- Grid Computing
- Cluster Computing
- Utility Computing
- **Cloud Computing**

Distributed Computing

Centralized vs. Distributed Computing

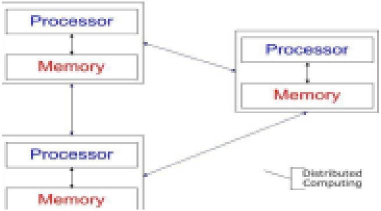




Early computing was performed on a single processor. Uni-processor computing can be called *centralized computing*.

Distributed Computing/System?

- Distributed Computing
  - Field of computing science that studies distributed system
  - Use of distributed systems to solve computational problems.
- Distributed Computing
  - Wikipedia
    - There are several autonomous computational entities, each of which has its own local memory.
    - The entities communicate with each other by message passing.
  - Operating System Concept
    - The processors communicate with one another through various communication lines, such as high-speed buses or telephone lines.
    - Each processor has its own local memory.



Example Distributed Systems

- Internet
- ATM(bank) machines
- Intranets/Workgroups
- Computing landscape will soon consist of ubiquitous network-connected devices

Computers in a Distributed System

- Workstations : Computers used by end-users to perform computing
- Server Systems: Computers which provide resources and services
- Intranets/Workgroups
- Personal Assistance Devices: Handheld computers connected to the system via a wireless communication link.

Common properties of Distributed Computing

- Fault tolerance
  - When one or some nodes fails, the whole system can still work fine except performance.
  - Need to check the status of each node
- Each node play partial role
  - Each computer has only a limited, incomplete view of the system
  - Each computer may know only one part of the input.
- Resource sharing
  - Each user can share the computing power and storage resource in the system with other users
- Load Sharing
  - Dispatching several tasks to each nodes can help share loading to the whole system.
- Easy to expand
  - We expect to use few time when adding nodes. Hope to spend no time if possible.
- Performance
  - Parallel computing can be considered a subset of distributed computing.

Why Distributed Computing?

- Nature of application
- Performance
  - – Computing Intensive
    - The task could consume a lot of time on computing. For example, Computation of Pi value using Monte Carlo simulation
  - – Data Intensive
    - The task that deals with a large amount or large size of files. For example, Facebook, LHC(Large Hadron Collider) experimental data processing.
- Robustness
  - – No SPOF (Single Point Of Failure)
  - – Other nodes can execute the same task executed on failed node.

THANK YOU!!

CLOUD COMPUTING

Cloud Computing - Overview

PROF.SOUMYA K.GHOSH

SILICON INSTITUTE OF TECHNOLOGY
SAMBALPUR

May 7, 2023

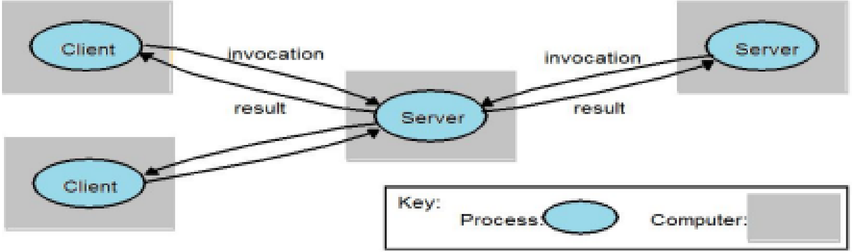
Why Distributed Computing?

- Nature of application
- Performance
  – Computing Intensive
    - The task could consume a lot of time on computing. For example, Computation of Pi value using Monte Carlo simulation
  – Data Intensive
    - The task that deals with a large amount or large size of files. For example, Facebook, LHC(Large Hadron Collider) experimental data processing.
- Robustness
  – No SPOF (Single Point Of Failure)
  – Other nodes can execute the same task executed on failed node.

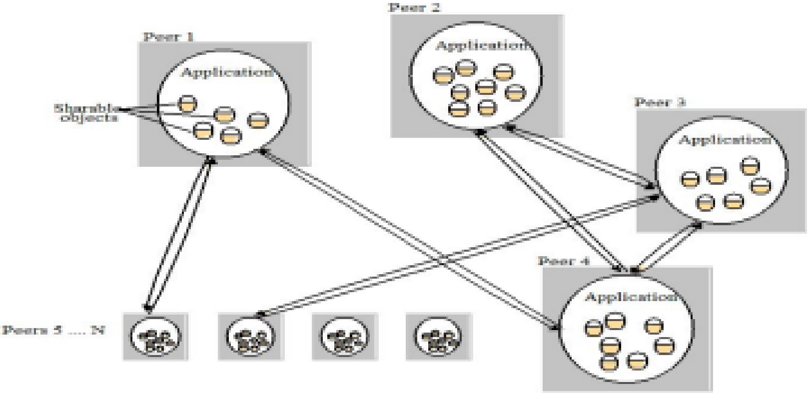
Distributed applications

- Applications that consist of a set of processes that are distributed across a network of machines and work together as an ensemble to solve a common problem
- In the past, mostly “client-server”
  - Resource management centralized at the server
- “Peer to Peer” computing represents a movement towards more “truly” distributed applications

Clients invoke individual servers



A typical distributed application based on peer processes



Grid Computing

Grid Computing?

- Pcwebopedia.com
  – A form of networking. unlike conventional networks that focus on communication among devices, grid computing harnesses unused processing cycles of all computers in a network for solving problems too intensive for any stand-alone machine.
- IBM
  – Grid computing enables the virtualization of distributed computing and data resources such as processing, network bandwidth and storage capacity to create a single system image, granting users and applications seamless access to vast IT capabilities. Just as an Internet user views a unified instance of content via the Web, a grid user essentially sees a single, large virtual computer.
- Sun Microsystems
  – Grid Computing is a computing infrastructure that provides dependable, consistent, pervasive and inexpensive access to computational capabilities

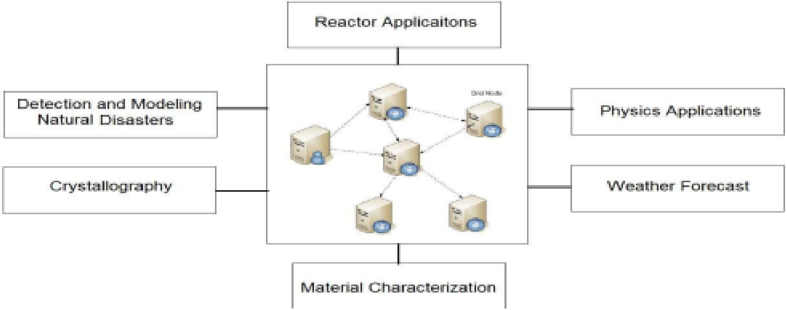
Grid Computing

1. Share more than information: Data, computing power, applications in dynamic environment, multi-institutional, virtual organizations
2. Efficient use of resources at many institutes. People from many institutions working to solve a common problem (virtual organisation).
3. Join local communities.
4. Interactions with the underneath layers must be transparent and seamless to the user.

Need Of Grid Computing?

- Today's Science/Research is based on computations, data analysis, data visualization collaborations
- Computer Simulations Modelling are more cost effectivethan experimental methods
- JScientific and Engineering problems are becoming more complex users need more accurate, precise solutions to their problems in shortest possible time
- Data Visualization is becoming very important
- Exploiting under utilized resources

Who uses Grid Computing ?

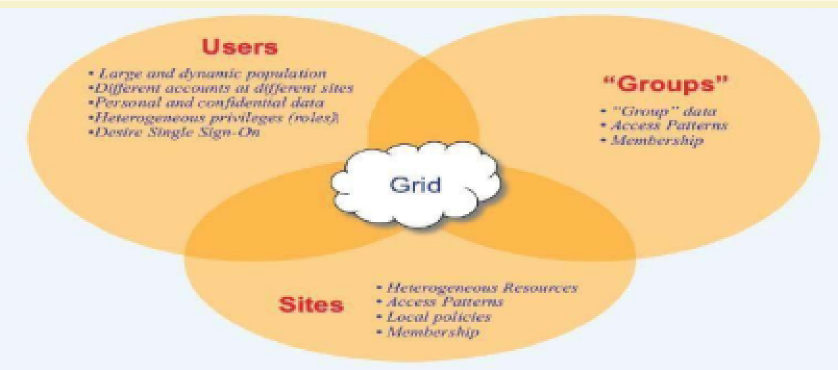


Types Of Grids

- **Computational Grid:** These grids provide secure access to huge pool of shared processing power suitable for high throughput applications and computation intensive computing.
- **Data Grid:** Data grids provide an infrastructure to support data storage, data discovery, data handling, data publication, and data manipulation of large volumes of data actually stored in various heterogeneous databases and file systems.
- **Collaboration Grid:** With the advent of Internet, there has been an increased demand for better collaboration. Such advanced collaboration is possible using the grid. For instance, persons from different companies in a virtual enterprise can work on different components of a CAD project without even disclosing their proprietary technologies

- **Network Grid:** A Network Grid provides fault-tolerant and high-performance communication services. Each grid node works as a data router between two communication points, providing data-caching and other facilities to speed up the communications between such points.
- **Utility Grid:** This is the ultimate form of the Grid, in which not only data and computation cycles are shared but software or just about any resource is shared. The main services provided through utility grids are software and special equipment. For instance, the applications can be run on one machine and all the users can send their data to be processed to that machine and receive the result back

Grid Components



Cluster Computing

What is Cluster Computing?

- A cluster is a type of parallelor distributed computer system, which consists of a collection of inter-connected stand-alone computers working together as a single integrated computing resource .
- Key components of a cluster include multiple standalone computers (PCs, Workstations, or SMPs), operating systems, high-performance interconnects, middleware, parallel programming environments, and applications.

Cluster Computing

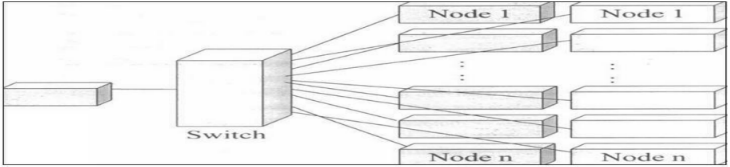
- Clusters are usually deployed to improve speed and/or reliability over that provided by a single computer, while typically being much more cost effective than single computer the of comparable speed or reliability
- In a typical cluster
  – Network: Faster, closer connection than a typical network (LAN)
  – Low latency communication protocols
  – Loosely coupled than SMP

Types of Cluster

- High Availability or Failover Clusters
- Load Balancing Cluster
- Parallel/Distributed Processing Clusters

Cluster Components

- Basic building blocks of clusters are broken down into multiple categories :
- **Cluster Nodes**
- **Cluster Network**
- **Network Characterization**



Key Operational Benefits of Clustering

- System availability: offer inherent high system availability due to the redundancy of hardware, operating systems, and applications.
- Hardware fault tolerance: redundancy for most system components (eg. disk-RAID), including both hardware and software.
- OS and application reliability: run multiple copies of the OS and applications, and through this redundancy
- Scalability. adding servers to the cluster or by adding more clusters to the network as the need arises or CPU to SMP.

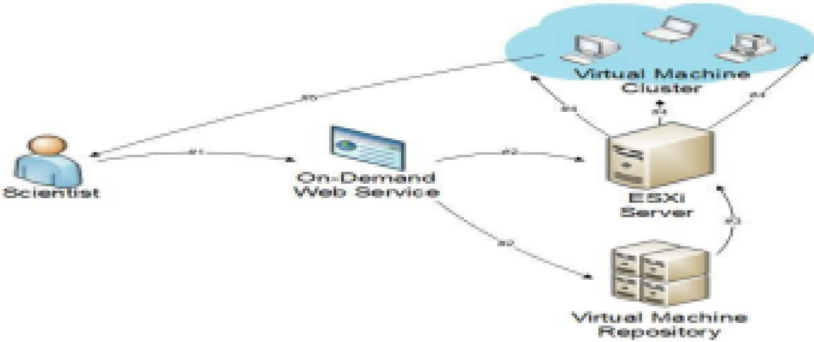
Utility Computing

“Utility” Computing?

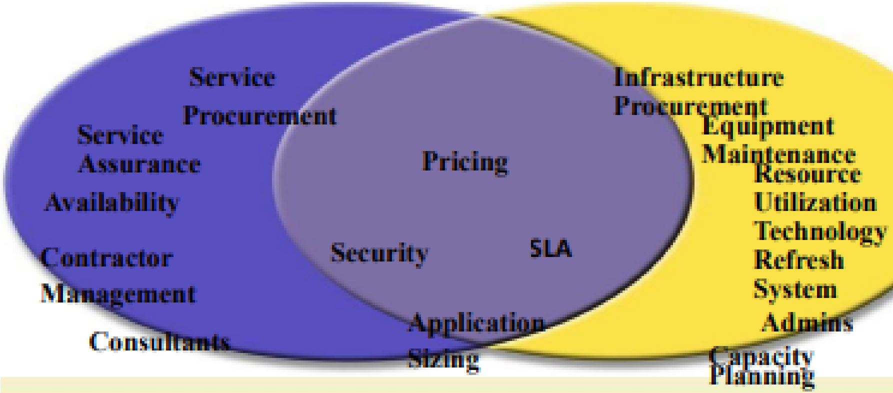
- Utility Computing is purely a concept which cloud computing practically implements.
- Utility computing is a service provisioning model in which a service provider makes computing resources and infrastructure management available to the customer as needed, and charges them for specific usage rather than a flat rate.
- This model has the advantage of a low or no initial cost to acquire computer resources; instead, computational resources are essentially rented.
- The word utility is used to make an analogy to other services, such as electrical power, that seek to meet fluctuating customer needs, and charge for the resources based on usage rather than on a flat-rate basis. This approach, sometimes known as pay-per-use

Utility Computing Example

- On-Demand Cyber
- Infrastructure



Utility Solution – Your Perspective Consumer Provider



Utility Computing Payment Models

- Same range of charging models as other utility providers: gas, electricity, telecommunications, water, television broadcasting
  - Flat rate
  - Tiered
  - Subscription
  - Metered
  - Pay as you go
  - Standing charges
- Different pricing models for different customers based on factors such as scale, commitment and payment frequency
- But the principle of utility computing remains
- The pricing model is simply an expression by the provider of the costs of provision of the resources and a profit margin

Overview

Normal text Alert Text Example Text **Emphasis Text**

Simple block

- ...

Example block

- ...

Alert block

- ...

A purple box

An orange box

A gray box

My price table

Color	Price 1	Price 2	Price 3
Red	10.00	20.00	30.00
Green	20.00	30.00	40.00
Blue	30.00	40.00	50.00
Orange	60.00	90.00	120.00

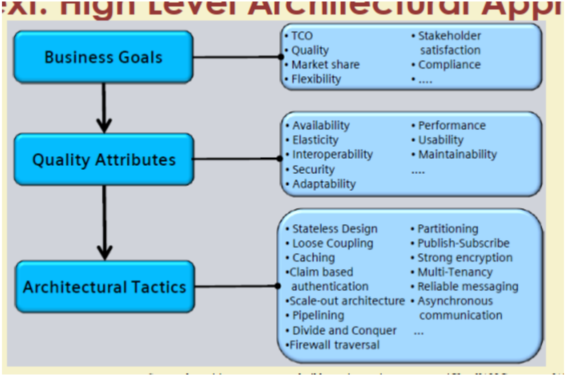
Cloud Computing

Cloud computing

this is tutorial

dataset	FABEF	FABEFOP	HAREF	HAREFOP	RR
200 × 20	381508.67	401105.67	166626.67	221278	902974.67
400 × 40	447298	471459.67	198433.67	269236	1052880.3
600 × 60	424855.33	462263.67	196371.67	248159.33	1130866
800 × 80	510789.33	5374060.7	243025	305499.67	1290577
1000 × 100	509360	550060	210131.3	314732	1411628

images


Business Goals
Quality Attributes
Architectural Tactics
• TCO
• Quality
• Market share
• Flexibility
• Stakeholder satisfaction
• Compliance
• ....
• Availability
• Elasticity
• Interoperability
• Security
• Adaptability
• Performance
• Usability
• Maintainability
....
• Stateless Design
• Loose Coupling
• Caching
•Claim based authentication
•Scale-out architecture
• Pipelining
• Divide and Conquer
•Firewall traversal
• Partitioning
• Publish-Subscribe
• Strong encryption
• Multi-Tenancy
• Reliable messaging
• Asynchronous communication
...

Blocks

Blocks types

Simple block

- First point
- Second point
- Third point

Examples block

- First point
- Second point
- Third point

Alert block

- First point
- Second point
- Third point

Boxes

Boxes

...

...

...

...

...

...

...

...

Lists

Items

- ...
- ...
- ...

Numbered

1. ...
2. ...
3. ...

Descriptive

Theme 1: ...
Theme 2: ...
Theme 3: ...

Tables

Tables 1

My price table		
Couleur	Prix 1	Prix 2
Rouge	10.00	20.00
Vert	20.00	30.00
Bleu	30.00	40.00
Orange	60.00	90.00

My price table		
Couleur	Prix 1	Prix 2
Rouge	10.00	20.00
Vert	20.00	30.00
Bleu	30.00	40.00
Orange	60.00	90.00

My price table		
Couleur	Prix 1	Prix 2
Rouge	10.00	20.00
Vert	20.00	30.00
Bleu	30.00	40.00
Orange	60.00	90.00

My price table		
Couleur	Prix 1	Prix 2
Rouge	10.00	20.00
Vert	20.00	30.00
Bleu	30.00	40.00
Orange	60.00	90.00

Tables 2

My price table		
Couleur	Prix 1	Prix 2
Rouge	10.00	20.00
Vert	20.00	30.00
Bleu	30.00	40.00
Orange	60.00	90.00

My price table		
Couleur	Prix 1	Prix 2
Rouge	10.00	20.00
Vert	20.00	30.00
Bleu	30.00	40.00
Orange	60.00	90.00

My price table		
Couleur	Prix 1	Prix 2
Rouge	10.00	20.00
Vert	20.00	30.00
Bleu	30.00	40.00
Orange	60.00	90.00

My price table		
Couleur	Prix 1	Prix 2
Rouge	10.00	20.00
Vert	20.00	30.00
Bleu	30.00	40.00
Orange	60.00	90.00

Figures

Figure Example

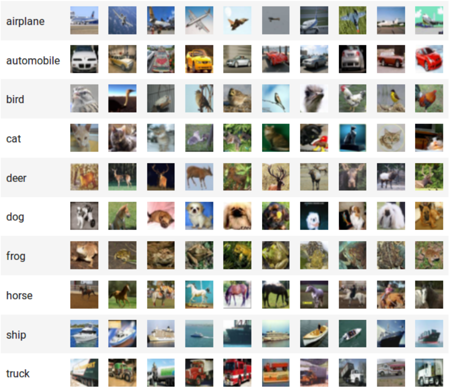


Figure: Example images from the CIFAR-10 dataset.

Equations and Codes

Equation Example

Some random equation:

$$\frac{\partial}{\partial \theta_k} J(\theta) = \frac{\partial}{\partial \theta_k} \left[\frac{1}{m} \sum_{k=1}^{m} log(1 + e^{-y^{(i)} \theta^T x^{(i)}}) \right]$$
$$= \frac{1}{m} \sum_{k=1}^{m} \frac{1}{1 + e^{-y^{(i)} \theta^T x^{(i)}}} y^{(i)} x_k^{(i)}$$
$$= -\frac{1}{m} \sum_{k=1}^{m} h_\theta(-y^{(i)} x^{(i)}) y^{(i)} x_k^{(i)}$$

Code Example

```python
def softmax_loss_naive(W, X, y, reg):
  """
  Softmax loss function, naive implementation (with loops)

  Inputs have dimension D, there are C classes, and we operate on minibatches
  of N examples.

  Inputs:
  - W: A numpy array of shape (D, C) containing weights.
  - X: A numpy array of shape (N, D) containing a minibatch of data.
  - y: A numpy array of shape (N,) containing training labels; y[i] = c means
    that X[i] has label c, where 0 <= c < C.
  - reg: (float) regularization strength

  Returns a tuple of:
  - loss as single float
  - gradient with respect to weights W; an array of same shape as W
  """
```

carbon
carbon.now.sh

1

[1] Carbon

Thank You!