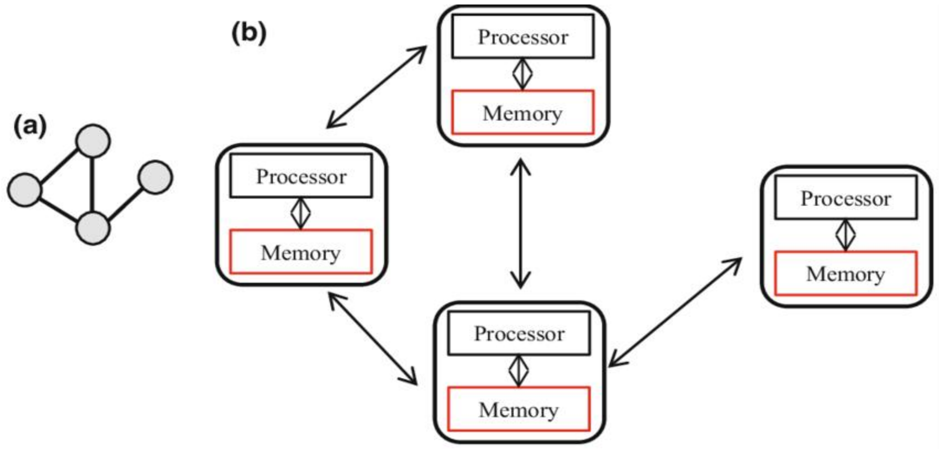
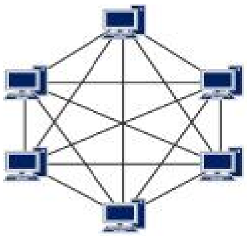
Chapter 1
INTRODUCTION

1. Introduction to Distributed Systems

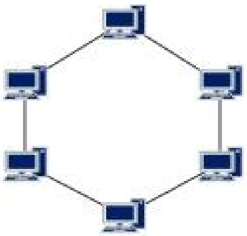
- A collection of autonomous independent computing elements that appear to its users as a single coherent system is known as distributed systems.
- All the components of the distributed systems interact to obtain a common goal.
- They could range from high performance mainframe computers to small devices in sensor networks.
- Examples: Internet, WWW, Intranet, Aircraft control systems, etc.



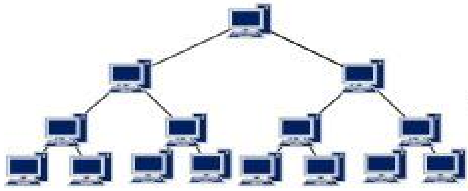
- DS differs from parallel system as each computer in a distributed system has its own memory i.e. distributed memory; but in parallel system, all processors have access to shared memory for information exchange.
- Generally, all the components of the distributed system interact so as to obtain a common goal. A computer program running in distributed system is called distributed program. Distributed programming is used to write distributed programs.
- Benefits of Distributed System over centralized system:
  - Scalability: Centralized systems can't scale up after a certain limit
  - No graceful degradation of system
  - Less possibility of data backup
  - Difficult server maintenance
  - Highly dependent on network connectivity
  - Bottlenecks can appear when traffic spikes e.g. DDoS attacks



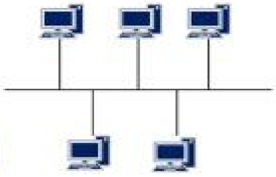
Fully Connected Distributed Systems



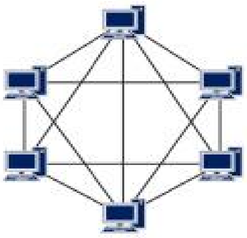
Ring Connected Distributed Systems



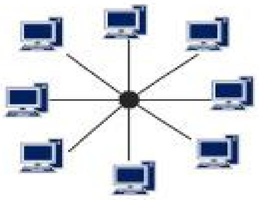
Hierarchically Connected Distributed Systems



Bus Connected Distributed Systems



Partially Connected Distributed Systems



Star Connected Distributed Systems

- A distributed system is a collection of independent computers at networked locations such that they communicate and interact only through ***message passing*** that is viewed as a single coherent system by its users.

- A distributed system should appear as a coherent system, meaning that an end user should not even notice that process, data and control are dispersed across a computer network. DS is coherent if it behaves according to the expectations of its users. In a single coherent system, the collection of nodes as a whole operates the same, no matter where, when and how interaction between a user and the system takes place.
- Offering a single coherent view is often challenging enough. Ex: it requires that an end user would not be able to tell exactly on which computer a process is currently executing, or even perhaps that part of the task has been spawned off to another process executing somewhere else. Likewise, where data is stored should be of no concern, and neither should it matter that the system may be replicating data to enhance performance. This so-called distribution transparency is an important design goal of distributed systems.

2. Examples of Distributed Systems

- Intranets, Internet, WWW, email
- Telecommunication networks: Telephone networks and Cellular networks
- The network of branch office computers -Information system to handle automatic processing of orders
- Real-time process control: Aircraft control systems
- Electronic banking
- Airline reservation systems
- Sensor networks
- Mobile and Pervasive Computing systems, etc.

Assignment: Internet, WWW, Client-Server Architecture, URL and URI

3. Main Characteristics (*NIC HFS*)

- **No Global Clock**
    - In order to coordinate to the remote networks it is necessary to synchronize clock in primitive technologies. But as there is no single correct global timing system, it is very difficult to synchronize global clock across all the networks of the system located at different locations.

    - The main characteristics of the distributed system is that it does not need any global clock system to coordinate with the network at different locations because the only way of communication is message passing through the network.

- **Independent Failure**
    - The computer system can fail or crash at any time. In primitive centralized system, if the server fails then the whole system fails to operate leaving the negative consequences to the end users.

    - With distributed system, even one network fails, it does not hamper the overall system. The workload of the failed network can be overcome by the other networks within the system situated at different locations.This provides reliable system to the end users.
    - “Graceful declination in system’s performance with failing network”

- **Concurrency**
    - Distributed system allows the programs that share resources to execute concurrently(i.e. At the same time). It is a property that enables processes running in different machines to form a common system that is capable of executing codes on multiple machines at the same time.
    - Concurrency helps to reduce latency and increase throughput as a unit of work can be concurrently done by sub-division.

- **Heterogeneity**
  - The internet enables users to access services and run applications over a heterogeneous collection of computers and networks. Heterogeneity applies to hardware devices, operating systems, network elements, programming language, etc. DS must ensure that program written in several languages must be able to communicate with each other.

- **Fault Tolerance**
  - Failures are of two types: Hardware and software failures. Any DS will carefully consider failure and its diagnostic scenarios. When designing the DS, the following assumptions should be considered false:
    - The network is reliable
    - Latency is zero
    - Bandwidth is infinite
    - Topology doesn't change
    - Transport cost is zero
    - The network is homogeneous

- **Security**
  - Security during storage as well as during transmission over a network is crucial in DS. The fundamental approaches used to ensure security are:
    - Symmetric Encryption
    - Asymmetric Encryption

4. Advantages and Disadvantages of Distributed System

- **Advantages:**
  - All the components work independently dividing the main task thus it increases performance of the system and reduces latency time.
  - Even if one component of system fails, the system as a whole does not crash. This improves reliability and availability of the system.
  - As load is distributed on the components of the system, it improves the total computational power of the system.
  - It allows user to share data and resources.
  - Local database administrator has different degree of local autonomy i.e. retains the control of system to some extent.
  - A collection of microprocessors offer a better-price/ performance than mainframes thus becomes economical option.

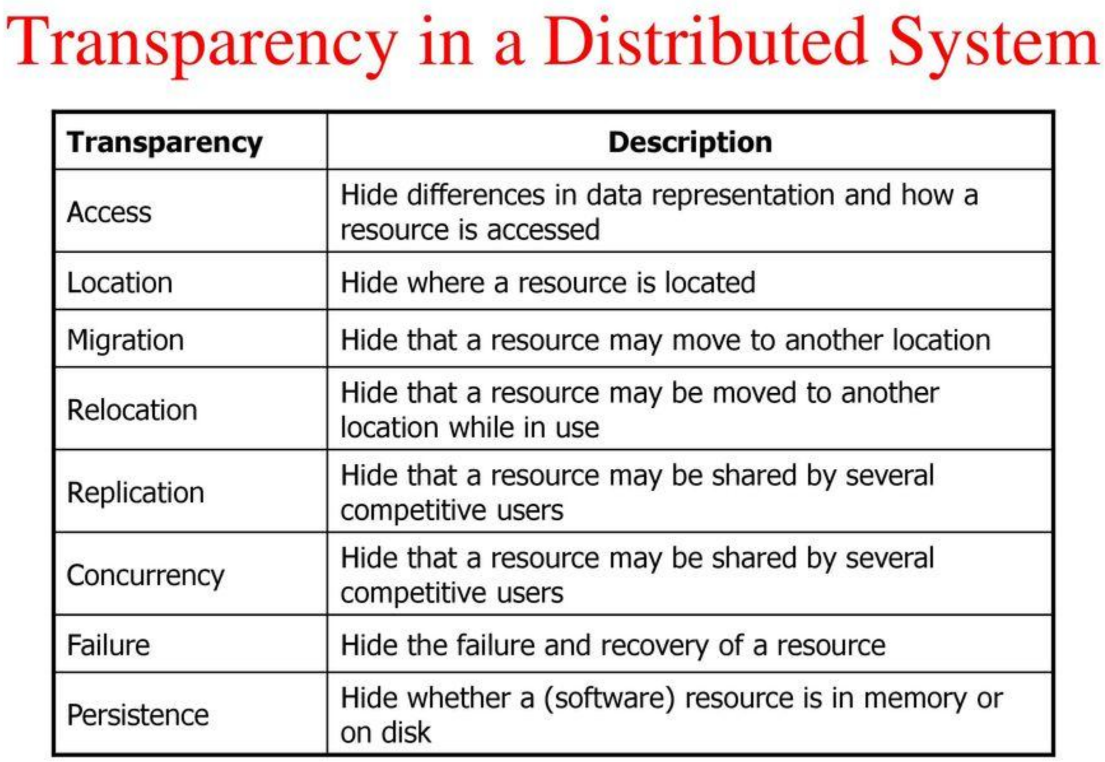
- **Disadvantages:**
  - High Cost
  - Difficult to implement and maintain DS
  - Exchange of information among the components requires coordination which creates processing overheads.
  - Difficult to ensure security.
  - Difficult to ensure the correctness of algorithm when some parts of the system is down and currently being recovered.

5. Design Goals

- **Resource Sharing**
  - The main goal of distributed system is to allow users to access remote resources and to share them in controlled and efficient manner. The resources may be printer, database, data, files and so on.
  - By sharing resources among the users, we can minimize the implementation cost of the system. It is also necessary for easy collaboration and information exchange among the users of the system.
  - Making resource sharing and information exchange easier result in increasing security related problems in the system which must be handled properly and efficiently to protect data from being compromised.

- **Openness**
  - A distributed system should be open. It must provides services with the standard rule following some protocols defining syntax and semantic of that service.
  - Services are specified through an interface defined in the Interface Definition Language (IDL). IDL must specify complete description of the service including function name, parameters, return values, possible exceptions and what the service is used for. The interface specification should be complete (All the necessary things required to implement the interface should be specified) and neutral (The structure of the implementation should not be specified).
  - Benefits:
    - Interoperability- components easily work together
    - Portability- applications easily transferred between DS
    - Extensibility- new services easily addition

- **Transparency**
  - Transparency is the ability to hide the fact that the processes and the resources of the distributed system are physically distributed across multiple computers or machines and makes users to realize that it is a single coherent system.
  - Transparency is defined as the hiding of the separation of components in a distributed systems from the user and the application programmer. With transparency the system is perceived as a whole rather than a collection of independent components.
  - Example: Let a person wants to print a file from the mobile device. Then it would be great if he/she could print it from a nearby printer than from printer located far from his current position. It is against the location transparency but provides reliable outcome to the system.
  - Different forms of transparencies are:



Transparency in a Distributed System

Transparency	Description
Access	Hide differences in data representation and how a resource is accessed
Location	Hide where a resource is located
Migration	Hide that a resource may move to another location
Relocation	Hide that a resource may be moved to another location while in use
Replication	Hide that a resource may be shared by several competitive users
Concurrency	Hide that a resource may be shared by several competitive users
Failure	Hide the failure and recovery of a resource
Persistence	Hide whether a (software) resource is in memory or on disk

- **Scalability**
  - A distributed system should be scalable with respect to size (Able to add any numbers of users and resources), geography (Users and resources may be in any locations) and administration (Easy management even if there are many independent administrations).

6. Main Problems

- **Transparency**
- **Scalability**
- **Dependability**
- **Performance**
- **Flexibility**

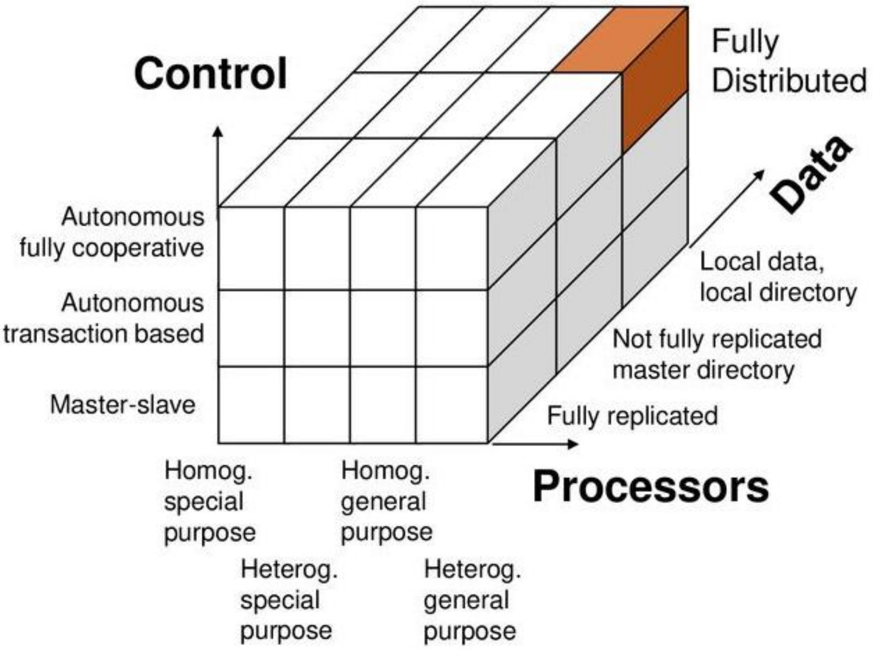
Assignment: Elaborate the problems

7. Models of Distributed System

- Hardware, Data and Controls are distributed in the distributed system environment.

- Enslow's Definition:

Distributed Systems = Distributed Hardware + Distributed Control + Distributed Data



- In a distributed system, hardware resources like processing power, storage, and network bandwidth are spread across multiple interconnected nodes, allowing for data and control to be distributed as well. This design enables parallel processing, enhances resilience, and facilitates resource sharing across different locations and devices.

- Hardware Distribution:
  - **Distributed Processing:**
    - ✓ Workloads are split across multiple nodes, allowing for parallel execution and improved performance.
  - **Resource Sharing:**
    - ✓ Resources like storage and computing power can be shared among nodes, increasing efficiency and reducing costs.
  - **Scalability:**
    - ✓ Additional nodes can be added to the system to accommodate growing demands without disrupting existing operations.

- Data Distribution:
  - **Data Replication:**
    - ✓ Data can be replicated across multiple nodes to ensure data availability and resilience.
  - **Data Partitioning:**
    - ✓ Large datasets can be divided and stored on different nodes based on specific criteria, improving query performance.
  - **Data Locality:**
    - ✓ Data can be strategically placed to minimize network traffic and improve performance for specific applications.

- Control Distribution:
  - **Decentralized Control:**
    - ✓ Control functions can be distributed among multiple nodes, enabling greater flexibility and autonomy.
  - **Autonomous Agents:**
    - ✓ Nodes can be equipped with autonomous control logic, allowing them to make decisions and respond to changes without central coordination.
  - **Middleware:**
    - ✓ Middleware services provide a common interface for nodes to communicate and interact, enabling interoperability and simplifying the development of distributed applications.

8. Resource Sharing and the Web Challenges

- Resource sharing means that the existing resources in a distributed system can be accessed or remotely accessed across multiple computers in the system.
- Computers in distributed systems shares resources like hardware (disks and printers), software (files, windows, and data objects), and data.
- Hardware resources are shared for reductions in cost and convenience. Data is shared for consistency and exchange of information.
- Resources are managed by a software module known as a resource manager. Every resource has its own management policies and methods.

- Users want to share data rather than the disks containing the data.
- Service manages a collection of related resources and provides the functionality to the users. Example: A file sharing is initiated by a file service providing read, write and delete operations or functions on the files.
- Services can be invoked from other computers only by communication (message passing in case of distributed system). Requests are sent from clients to a server via a message and the server reply to the clients via a message.
- When clients send a request for an operation to a server, it means client invokes an operation upon the server. Remote invocation means the complete interaction between a client and a server from request by a client to the response given by the server.

Client and Server

Client and server are the process rather than the computer. The process which requests an operation is called client and the process that provides the requested service is called server.

Client is active while server is passive. Server runs continuously while clients run until the applications lasts. An executing web browser is a client while the web server is a server.

WWW (World Wide Web)

- It is a system to publish and access services and resources throughout the world with the help of Internet. Web browsers should be used to access the world wide web. It presents data in the form of web pages. Each document or web page contains hyperlinks (references to other documents and resources).
- The web is based on following standard components:
  - HTML : It is a language to specify the contents of the web pages.
  - URL : It identifies the documents and other resources stored on the web.
  - Client Server Architecture: It provides standard rules by which browsers can access to the resources from the web servers.

Web Challenges

- If the resources are deleted or moved, the links to that resource still remains. Such links are called dangling links.
- The response of the web server to the users may be slow.
- The web page is not satisfactory user interface.

Semantic Web

- The web of linked metadata resources is called semantic web.
- Proper metadata helps the search engines to search the contents in effective and efficient way.

True Distributed System(TDS)

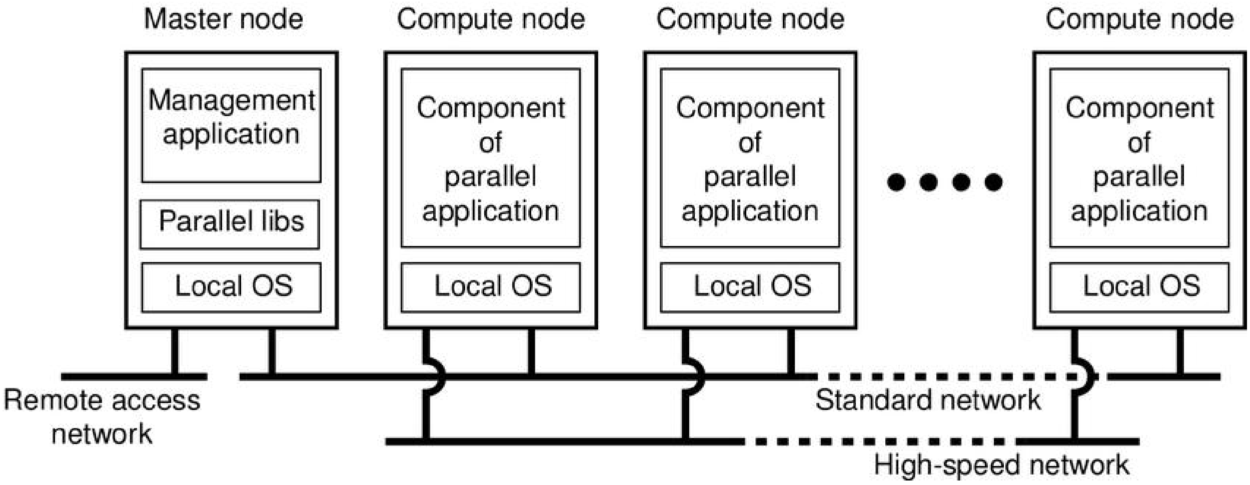
- A distributed system with all the challenges like transparency, scalability, Dependability, Performance, and Flexibility, being solved to its full extent, is called a True Distributed System (TDS).
- However, because of the different problems (like a need of new component (network), security and software complexity faced during the development of a distributed system, a TDS is virtually possible.

9. Types of Distributed System: Grid, Cluster, Cloud

- Distributed computing is a multifaceted field with infrastructures that can vary widely. It is thus nearly impossible to define all types of distributed computing. However, this field of computer science is commonly divided into three sub fields:

a) Cluster Computing System

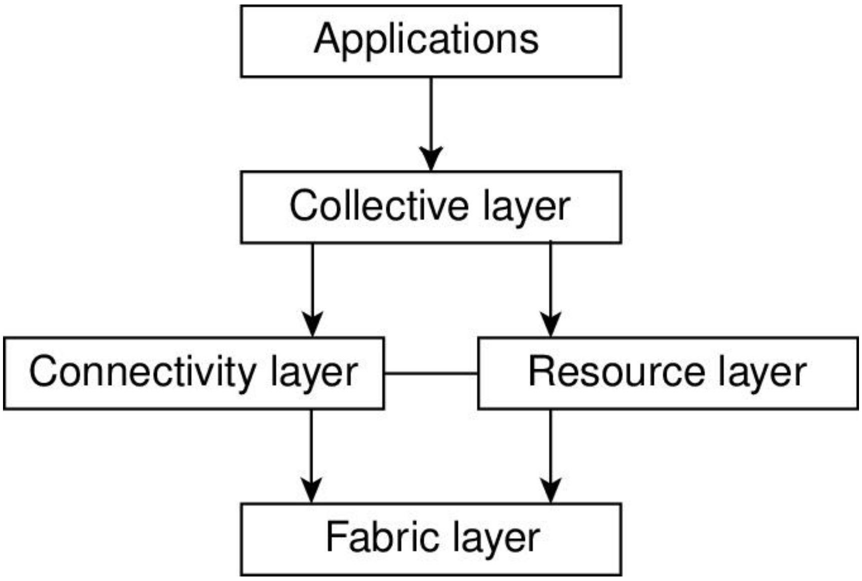
- In cluster computing, the underlying hardware consists of a collection of similar workstations or PCs closely connected by means of high speed LAN.
- The feature of cluster computing is homogeneity. It means all the computers within the cluster system must have same operating system and must be operated within the same network.
- At a certain point, it becomes financially and technically attractive to build a super computer using off the shell technology by simply hooking up a connection of relatively simple computers in a high speed network.
- Cluster system is generally used for parallel programming in which a single program is run in parallel on multiple computers within the system.



Each cluster consists of a collection of compute nodes that are controlled and accessed by means of a single master node. The master handles the allocation of nodes to a particular parallel program, maintains a batch queue of submitted jobs and provides an interface for the users of system as such the master actually runs the middleware needed for the exhibition of program and management of the cluster.

b) <u>Grid Computing System</u>

- Grid systems have high degree of heterogeneity. It means that the computers in the grid system must not be the same with respect to hardware, operating systems, network, security policy, administrative domain and so on.
- The resources from different organization can collaborate via a virtual organization with access rights with all the members of this virtual organization.
- The architecture of grid computing system is as:



- The fabric layer provides interface to local resources at a specific site which can be shared within a virtual organization. It provides functions to query state and capability of resources, and actual resource management.
- The connectivity layer communication protocols to support grid transactions. It supports delegation of rights from authenticated users to programs running on behalf of those users.
- The resource layer manages a single resource with the help of functions provided by connectivity layer and interfaces of fabric layer. It is responsible for access control.
- The collective layer handles access to multiple resources via resource discovery, allocation and scheduling of multiple resources along with data replication.
- The application layer consists of applications that uses the services provided by the grid computing system. Applications operate within the virtual organizations.

Since grid computing can create a virtual supercomputer from a cluster of loosely interconnected computers, it is specialized in solving problems that are particularly computationally intensive. This method is often used for ambitious scientific projects and decrypting cryptographic codes.

c) **Cloud Computing System**

- Cloud computing system is a type of Internet-based computing that provides shared computer processing resources and data to computers and other devices on demand. It enables on demand access to a shared pool of configurable computing resources.
- Cloud Computing uses distributed computing to provide customers with highly scalable cost-effective infrastructures and platforms. Cloud providers usually offer their resources through hosted services that can be used over the internet.

- The cloud models can be classified as: Service Models and Deployment model.

- The service models include:
  - ✧ Software as a Service (SaaS)- provides software application on-demand based on subscription
  - ✧ Platform as a Service (PaaS)- offer runtime environment for applications
  - ✧ Infrastructure as a Service (IaaS)- provides access to fundamental resources like virtual machines, storage, etc.

- The deployment models include: **(Assignment)**
  - Public Cloud
  - Private Cloud
  - Community Cloud
  - Hybrid Cloud