# CLOUD COMPUTING FUNDAMENTALS

MODULE 1

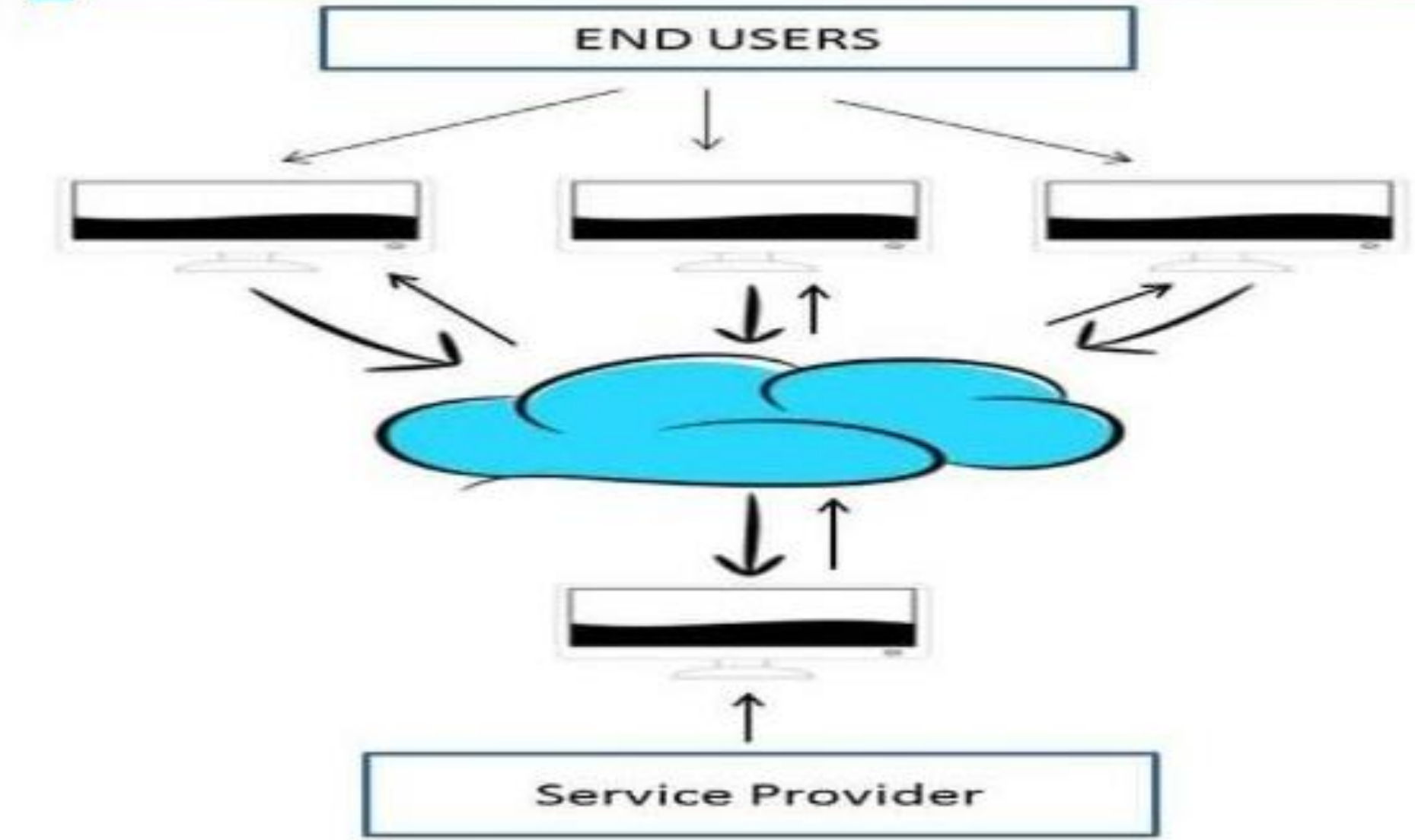
# What is Cloud ?

- The term Cloud refers to a Network or Internet. In other words, we can say that Cloud is something, which is present at remote location. Cloud can provide services over network, i.e., on public networks or on private networks, i.e., WAN, LAN or VPN.

- The term "Cloud" came from a network design that was used by network engineers to represent the location of various network devices and there inter-connection.

- "The cloud" refers to servers that are accessed over the Internet, and the software and databases that run on those servers. Cloud servers are located in data centres all over the world.

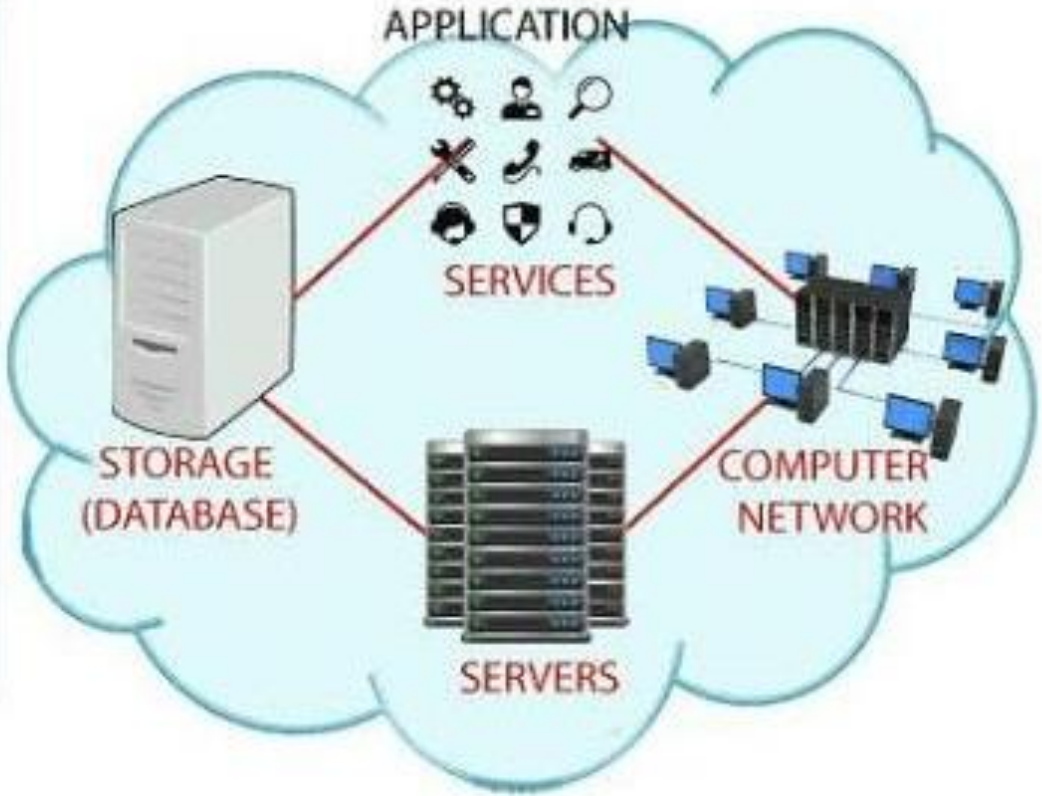
# What is Cloud ?



# What is Cloud Computing ?

Before Cloud Computing

After Cloud Computing



# What is Cloud ?

- The cloud enables users to access the same files and applications from almost any device, because the computing and storage takes place on servers in a data centre, instead of locally on the user device.

- This is why a user can log in to their Instagram account on a new phone after their old phone breaks and still find their old account in place, with all their photos, videos, and conversation history.

- In Simplest terms, cloud computing means storing and accessing the data and programs on remote servers that are hosted on the internet instead of the computer's hard drive or local server.

- You can access cloud-based applications and services from anywhere - all you need is a device with an internet connection.

# What is Cloud Computing ?

- Cloud computing is the delivery of different services through the Internet. These resources include tools and applications like data storage, servers, databases, networking, and software.

- Cloud computing is the delivery of computing services including servers, storage, databases, networking, software, analytics, and intelligence over the Internet ("the cloud") to offer faster innovation, flexible resources, and economies of scale.

- You typically pay only for cloud services you use, helping lower your operating costs, run your infrastructure more efficiently and scale as your business needs change.

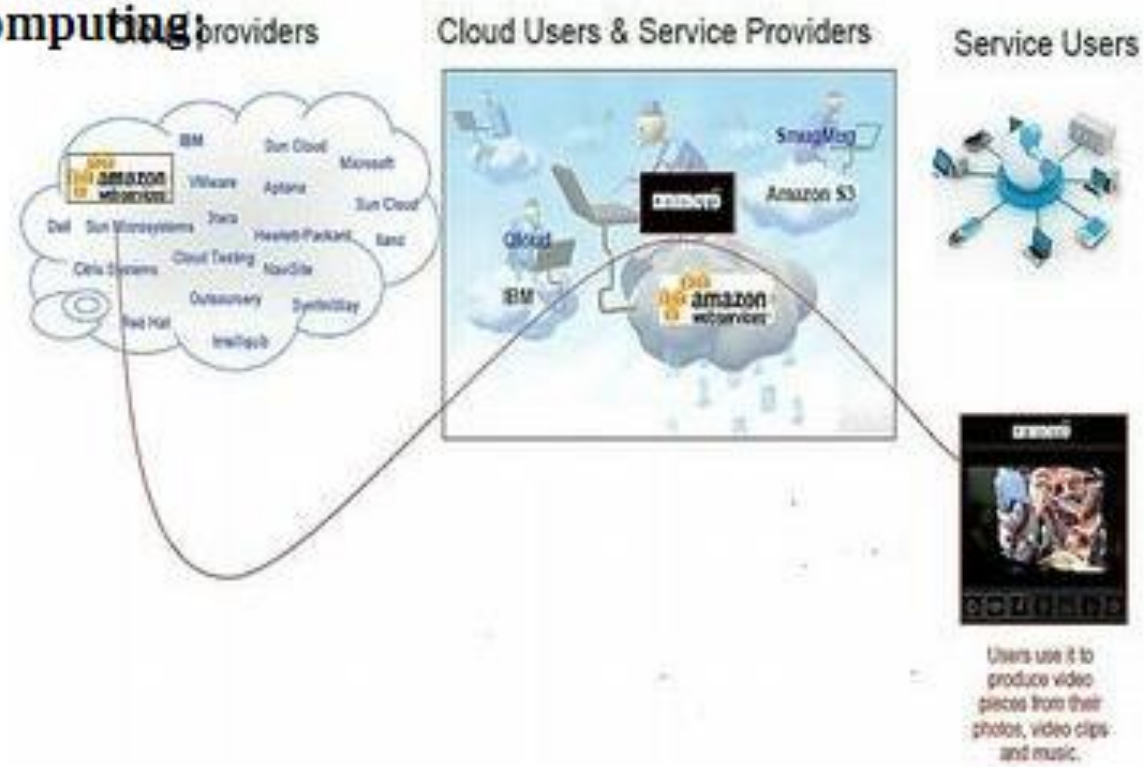
- Cloud computing is named as such because the information being accessed is found remotely in the cloud or a virtual space. Companies that provide cloud services enable users to store files and applications on remote servers and then access all the data via the Internet.

# What is Cloud Computing ?

- This means the user is not required to be in a specific place to gain access to it, allowing the user to work remotely.

- There are certain services and models working behind the scene making the cloud computing feasible and accessible to end users. Following are the working models for cloud computing:

  - ✓ Deployment Models
  - ✓ Service Models



# Definition of Cloud

- Cloud computing is a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g. networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction."

- "A Cloud is a type of parallel and distributed system consisting of a collection of interconnected and virtualized computers that are dynamically provisioned and presented as one or more unified computing resources based on service-level agreements established through negotiation between the service provider and consumers."

- Cloud computing refers to both the applications delivered as services over the Internet, and the hardware and system software in the datacentres that provide those services.

# Properties of cloud computing

- **User Centric :-** Cloud Computing Is user centric once as a user are connected to the cloud, whatever is stored there documents, messages, images, applications, whatever becomes authorized to the user access them. In addition, not only is the data, but one can also share it with others.

- **Task Centric :-** Cloud computing is task centric because the usage model is based entirely around what users want to achieve, rather than any particular software, hardware or network infrastructure. Users do not have to purchase or install anything before using a cloud computing resource.

- **Self-Healing :-** Cloud computing is Self-diagnosis and self-healing is one of the important ability with endowing clouds to reduce the complexity and increase the sustainability to the failures.

- **Flexible :-** Cloud computing allows your employees to be more flexible - both in and out of the workplace. Employees can access files using web-enabled devices such as smartphones, laptops and notebooks.

# Properties of cloud computing

- Programmable :- Many of the tasks necessary with cloud computing must be automated. For example, to protect the integrity of the data, information stored on a single computer in the cloud must be replicated on other computers in the cloud.

- Multi-tenancy :- In cloud computing, multi-tenancy can also refer to shared hosting, in which server resources are divided among different customers. Multi-tenancy is a software architecture where a single software instance can serve multiple, distinct user groups. Software-as-a-service (SaaS) offerings are an example of multitenant architecture.

# History of Cloud Computing

- **We are quick walkthrough of cloud computing history and evolution all these years.**

- **1960's** - One of the renowned names in Computer Science, John McCarthy, enabled enterprises to use expensive mainframe and introduced the whole concept of time-sharing. This turned out to be a huge contribution to the pioneering of Cloud computing concept and establishment of Internet.

- **1969** - With the vision to interconnect the global space, J.C.R. Licklider introduced the concepts of "Galactic Network" and "Intergalactic Computer Network" an also developed Advanced Research Projects Agency Network- ARPANET.

- **1970** - By this era, it was possible to run multiple Operating Systems in isolated environment.

- **1997**- Prof. Ramnath Chellappa introduced the concept of "Cloud Computing" in Dallas.

# History of Cloud Computing

- **1999** - Salesforce.com started the whole concept of enterprise applications through the medium of simple websites. Along with that, the services firm also covered the way to help experts deliver applications via the Internet.

- **2003** - The Virtual Machine Monitor (VMM), that allows running of multiple virtual guest operating systems on single device, paved way ahead for other huge inventions.

- **2006** - Amazon also started expanding in cloud services. From EC2 to Simple Storage Service S3, they introduced pay-as-you-go model, which has become a standard practice even today.

- **2013** - With IaaS, (Infrastructure-as-a-Service), the Worldwide Public Cloud Services Market was totalled at £78bn, which turned out to be the fastest growing market services of that year.

# The Era of Microservices and Containers: 2010s

- 2010: Microsoft launches Microsoft Azure.
- 2011: Microsoft launches Office 365 (SaaS).
- 2013: Release of Docker container technology.
- 2020 and beyond: Serverless and edge computing are expected to grow with advancements in AI and ML.

# Next-Generation Cloud: 2023 And Beyond

| | 2021 | 2022 | 2023 |
|---|---|---|---|
| Cloud Business Process Services (BPaaS) | 54,952 | 60,127 | 65,145 |
| Cloud Application Infrastructure Services (PaaS) | 89,910 | 110,677 | 136,408 |
| Cloud Application Services (SaaS) | 146,326 | 167,107 | 195,208 |
| Cloud Management and Security Services | 28,489 | 34,143 | 41,675 |
| Cloud System Infrastructure Services (IaaS) | 90,894 | 115,740 | 150,254 |
| Desktop-as-a-Service (DaaS) | 2,059 | 2,539 | 3,104 |
| **Total Market** | **412,632** | **490,333** | **591,794** |

# Advantages of Cloud Computing

- 1. Cost Optimization
- 2. Competitive Advantage
- 3. Rapid Deployment
- 4. Data backup and Restoration
- 5. Automatic Integration
- 6. Reliable Support
- 7. Easier Accessibility and Collaboration
- 8. Infinite Storage Capacity

# Disadvantages of Cloud Computing

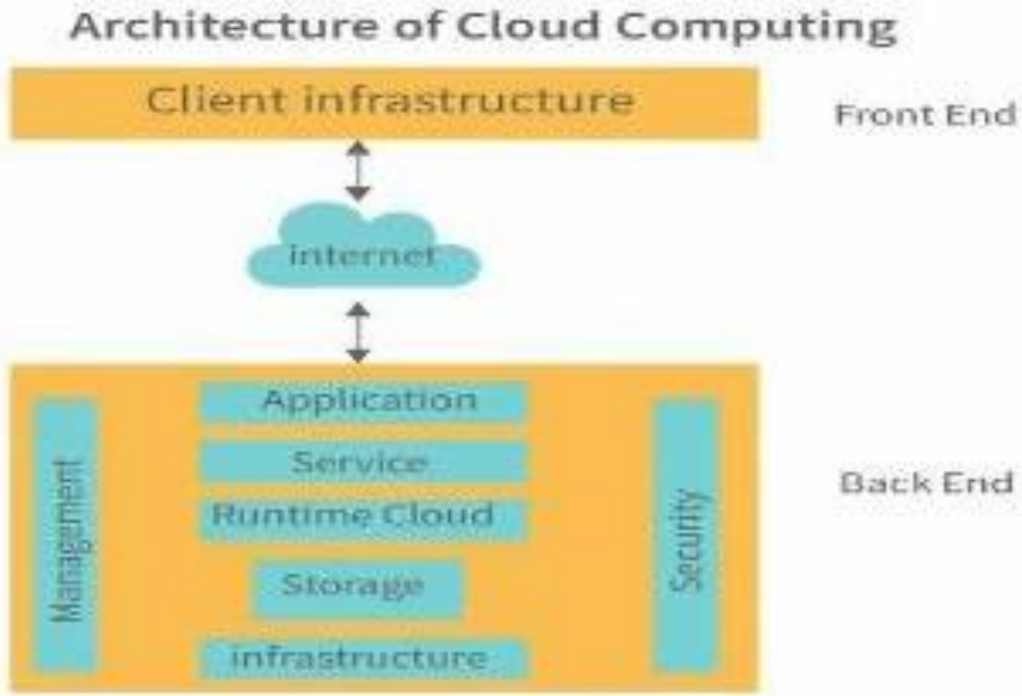
- 1. Downtime
- 2. Depends on internet connection
- 3. Security Concerns
- 4. Good Internet Connection
- 5. Reduced Bandwidth
- 6. Lack of Customer Assistance
- 7. Limited Control

# Cloud Ecosystem

- A cloud ecosystem is a dynamic system of interdependent elements, all of which work together to make cloud services possible. In nature, an ecosystem consists of objects that are linked and work together that are living and non-living. The ecosystem consists of hardware and software in cloud computing, as well as cloud clients, cloud developers, consultants, integrators, and collaborators.

# Cloud Computing Architecture

- Cloud computing architecture is a combination of service-oriented architecture and event-driven architecture. Cloud computing architecture is divided into the following two parts .




- Front End
- Back End

- 1. Frontend :- Frontend of the cloud architecture refers to the client side of cloud computing system. Means it contains all the user interfaces and applications which are used by the client to access the cloud computing services/resources. For example, use of a web browser to access the cloud platform.

# Cloud Computing Architecture

- Client Infrastructure

- 2. Backend :
  Backend refers to the cloud itself which is used by the service provider. It contains the resources as well as manages the resources and provides security mechanisms. Along with this, it includes huge storage, virtual applications, virtual machines, traffic control mechanisms, deployment models, etc.
- Application
- Service
- Runtime Cloud
- Storage
- Infrastructure
- Management
- Security
- Internet

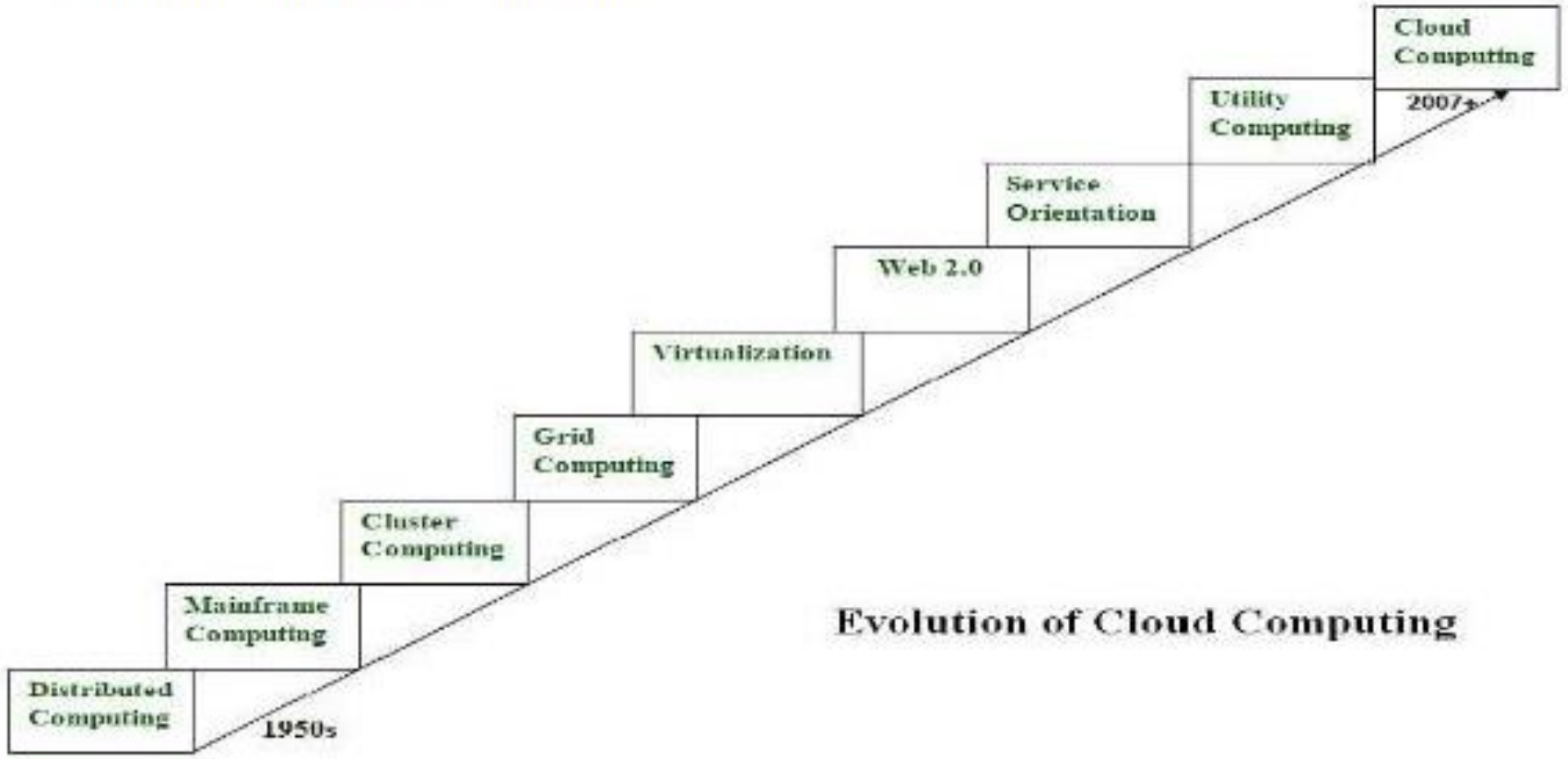
# Cloud Computing Architecture

- Benefits of Cloud Computing Architecture :-

- Makes overall cloud computing system simpler.
- Improves data processing requirements.
- Helps in providing high security.
- Makes it more modularized.
- Results in better disaster recovery.
- Gives good user accessibility.
- Reduces IT operating costs.

# Evolution of Cloud Computing

- Cloud computing evolved primarily from various computing technologies such as distributed systems and peripherals, virtualization, web 2.0, service orientation, and utility computing.




Evolution of Cloud Computing

# Evolution of Cloud Computing

- Cloud computing evolved primarily from various computing technologies such as distributed systems and peripherals, virtualization, web 2.0, service orientation, and utility computing.

- Distributed Systems :- A distributed system is a computing environment. Different components are distributed across multiple computers connected by a network. These devices divided the work, collaborating their efforts to accomplish the assignment more efficiently than when a single device was in charge.

- But the main problem with this system was that all the systems were required to be present at the same geographical location.

- Thus to solve this problem, distributed computing led to three more types of computing and they were Mainframe Computing, Cluster Computing, and Grid Computing.

# Evolution of Cloud Computing

- Mainframes Computing :- Mainframes were powerful, highly reliable computers specialized for large data movement and massive IO operations. They were mostly used by large organizations for bulk data processing such as online transactions, enterprise resource planning, and other operations involving the processing of significant amount of data.

- After distributed computing, these increased the processing capabilities of the system. But these were very expensive. To reduce this cost, cluster computing came as an alternative to mainframe technology.

# Evolution of Cloud Computing

- Cluster Computing :- Cluster computing started as a low-cost alternative to the use of mainframes and supercomputers.

- Each machine in the cluster was connected to each other by a network with high bandwidth. Also, new nodes could easily be added to the cluster if it was required.

- Thus, the problem of the cost was solved to some extent but the problem related to geographical restrictions still pertained. To solve this, the concept of grid computing was introduced.

# Evolution of Cloud Computing

- **Grid Computing :-** In 1990s, the concept of grid computing was introduced. It means that different systems were placed at entirely different geographical locations and these all were connected via the internet.

- These systems belonged to different organizations and thus the grid consisted of heterogeneous nodes. Although it solved some problems but new problems emerged as the distance between the nodes increased.

- The main problem which was encountered was the low availability of high bandwidth connectivity and with it other network associated issues.

- Cloud computing is often considered as the successor of Grid computing. In reality, it embodies aspects of all of these three major technologies.

# Evolution of Cloud Computing

- Virtualization :- Virtualization is another core technology for Cloud computing. Virtualization is the "creation of a virtual (rather than actual) version of something, such as a server, a desktop, a storage device, an operating system or network resources".

- In other words, Virtualization is a technique, which allows to share a single physical instance of a resource or an application among multiple customers and organizations.

- Virtualization is essentially a technology that allows creation of different computing environments. These environments are named as virtual, because they simulate the interface that is expected by a guest.

# Evolution of Cloud Computing

- Web 2.0 :- The Web is the primary interface through which Cloud computing deliver its services. At present time, it encompasses a set of technologies and services that facilitate interactive information sharing, collaboration, user-centered design, and application composition.

- This has transformed the Web into a rich platform for application development. Web 2.0 brings interactivity and flexibility into Web pages, which provide enhanced user experience by gaining Web-based access to all the functions that are normally found in desktop applications.

# Evolution of Cloud Computing

- **Service-Oriented :-** Service orientation is the core reference model for Cloud computing systems. This approach adopts the concept of services as main building blocks of application and system development.

- Service-Oriented supports the development of rapid, low-cost, flexible, interoperable, and evolvable applications and systems. Service-Oriented Computing introduces the two important concepts, which are also fundamental for Cloud computing, Quality of Service (QoS) and Software as a Service (SaaS).

- Quality of Service identifies a set of functional and non-functional attributes that can be used to evaluate the behaviour of a service from different perspectives.

- The concept of Software as a Service introduces a new delivery model for applications. It has been inherited from the world of Application Service Providers (ASPs).

# Evolution of Cloud Computing

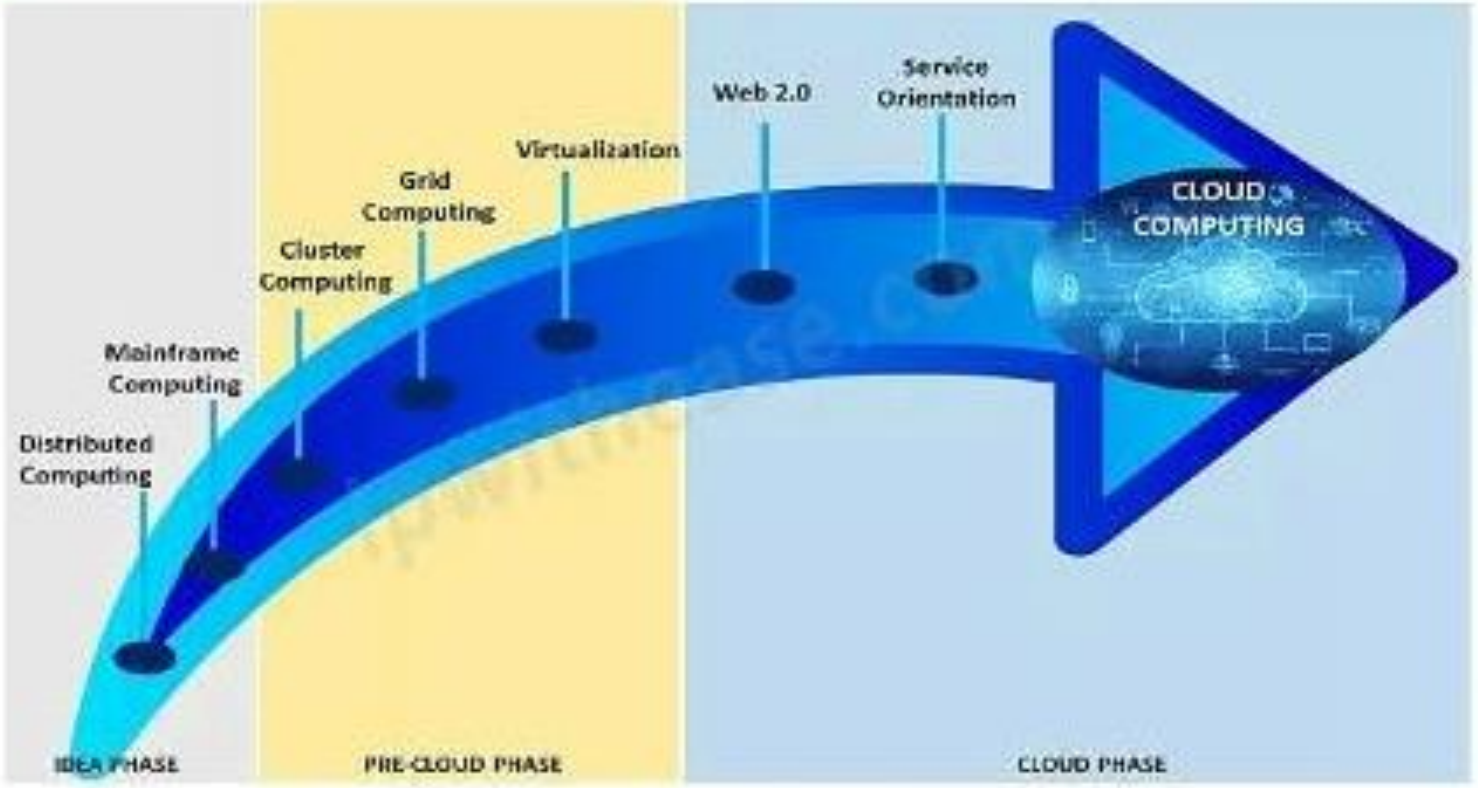
- Utility computing :- It is a computing model that defines service provisioning techniques for services such as compute services along with other major services such as storage, infrastructure, etc which are provisioned on a pay-per-use basis.

- Utility computing is a service provisioning model where a provider makes computing resources, infrastructure management and technical services available to customers as they need them. The provider then charges the customer for the amount of services they use rather than a flat-rate fee.

# Phases of the Evolution of Cloud Computing

- Cloud computing is playing an important role in everyone's life. We cannot imagine a world without Cloud computing techniques.

- Phase 1: Idea Phase
- Phase 2: Pre-Cloud Phase
- Phase 3: Cloud Phase



# Eras of Computing

- The two fundamental and dominant models of computing are: sequential and parallel. The sequential computing era began in 1940s; parallel (and distributed) computing era followed it within a decade.

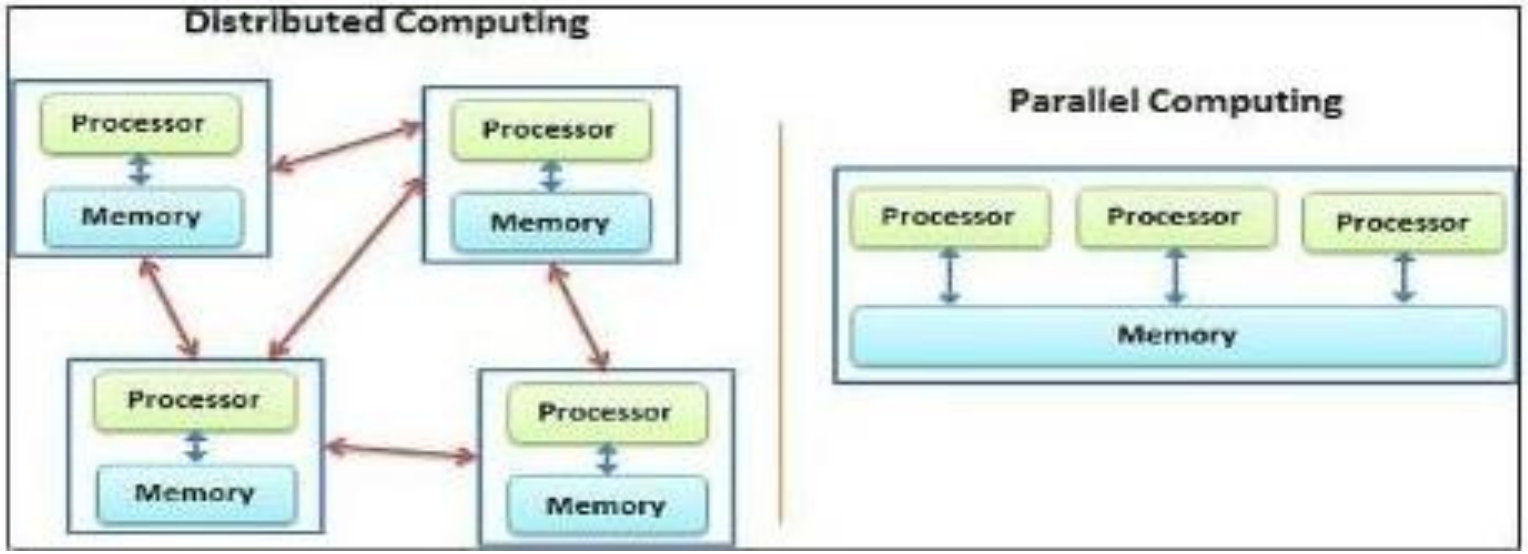
- The computing era started with a development in hardware architectures, which actually enabled the creation of system software particularly in the area of compilers and operating systems which supported the management of such systems and the development of applications.

- The development of applications and systems are the major element of interest and it comes to consolidation when problem-solving environments are designed and introduced to facilitate and empower engineers.

# Parallel & Distributed Computing

- Parallel Computing :- Parallel computing is the process of performing computational tasks across multiple processors at once to improve computing speed and efficiency. It divides tasks into sub-tasks and executes them simultaneously through different processors.

- Distributed Computing :- In distributed computing we have multiple autonomous computers which seems to the user as single system. In distributed systems there is no shared memory and computers communicate with each other through message passing. In distributed computing a single task is divided among different computers.



# Difference between Parallel Computing and Distributed Computing

| S.NO | Parallel Computing | Distributed Computing |
|---|---|---|
| 1. | Many operations are performed simultaneously | System components are located at different locations |
| 2. | Single computer is required | Uses multiple computers |
| 3. | Multiple processors perform multiple operations | Multiple computers perform multiple operations |
| 4. | It may have shared or distributed memory | It have only distributed memory |
| 5. | Processors communicate with each other through bus | Computer communicate with each other through message passing. |
| 6. | Improves the system performance | Improves system scalability, fault tolerance and resource sharing capabilities |

# Advantages and Disadvantages of Parallel Computing

- Advantages

1.It saves time and money because many resources working together cut down on time and costs.

2.It may be difficult to resolve larger problems on Serial Computing.

3.You can do many things at once using many computing resources.

4.Parallel computing is much better than serial computing for modeling, simulating, and comprehending complicated real-world events.

- Disadvantages

1.The multi-core architectures consume a lot of power.

2.Parallel solutions are more difficult to implement, debug, and prove right due to the complexity of communication and coordination, and they frequently perform worse than their serial equivalents.

# Advantages and Disadvantages of Distributed Computing

## Advantages

1.It is flexible, making it simple to install, use, and debug new services.

2.In distributed computing, you may add multiple machines as required.

3.If the system crashes on one server, that doesn't affect other servers.

4.A distributed computer system may combine the computational capacity of several computers, making it faster than traditional systems.

## Disadvantages

1.Data security and sharing are the main issues in distributed systems due to the features of open systems

2.Because of the distribution across multiple servers, troubleshooting and diagnostics are more challenging.

3.The main disadvantage of distributed computer systems is the lack of software support.

# Advantages and Disadvantages of Distributed Computing

## Advantages

1.It is flexible, making it simple to install, use, and debug new services.

2.In distributed computing, you may add multiple machines as required.

3.If the system crashes on one server, that doesn't affect other servers.

4.A distributed computer system may combine the computational capacity of several computers, making it faster than traditional systems.

## Disadvantages

1.Data security and sharing are the main issues in distributed systems due to the features of open systems

2.Because of the distribution across multiple servers, troubleshooting and diagnostics are more challenging.

3.The main disadvantage of distributed computer systems is the lack of software support.

# Types of Parallel Computing

1. Bit-level parallelism - It is the form of parallel computing which is based on the increasing processor's size. It reduces the number of instructions that the system must execute in order to perform a task on large-sized data.

2. Instruction-level parallelism - A processor can only address less than one instruction for each clock cycle phase. These instructions can be re-ordered and grouped which are later on executed concurrently without affecting the result of the program. This is called instruction-level parallelism.

3. Task Parallelism - Task parallelism employs the decomposition of a task into subtasks and then allocating each of the subtasks for execution. The processors perform the execution of sub-tasks concurrently.

# Parallel Processing

Processing of multiple tasks simultaneously on multiple processors is called parallel processing. The parallel program consists of multiple active processes (tasks) simultaneously solving a given problem.

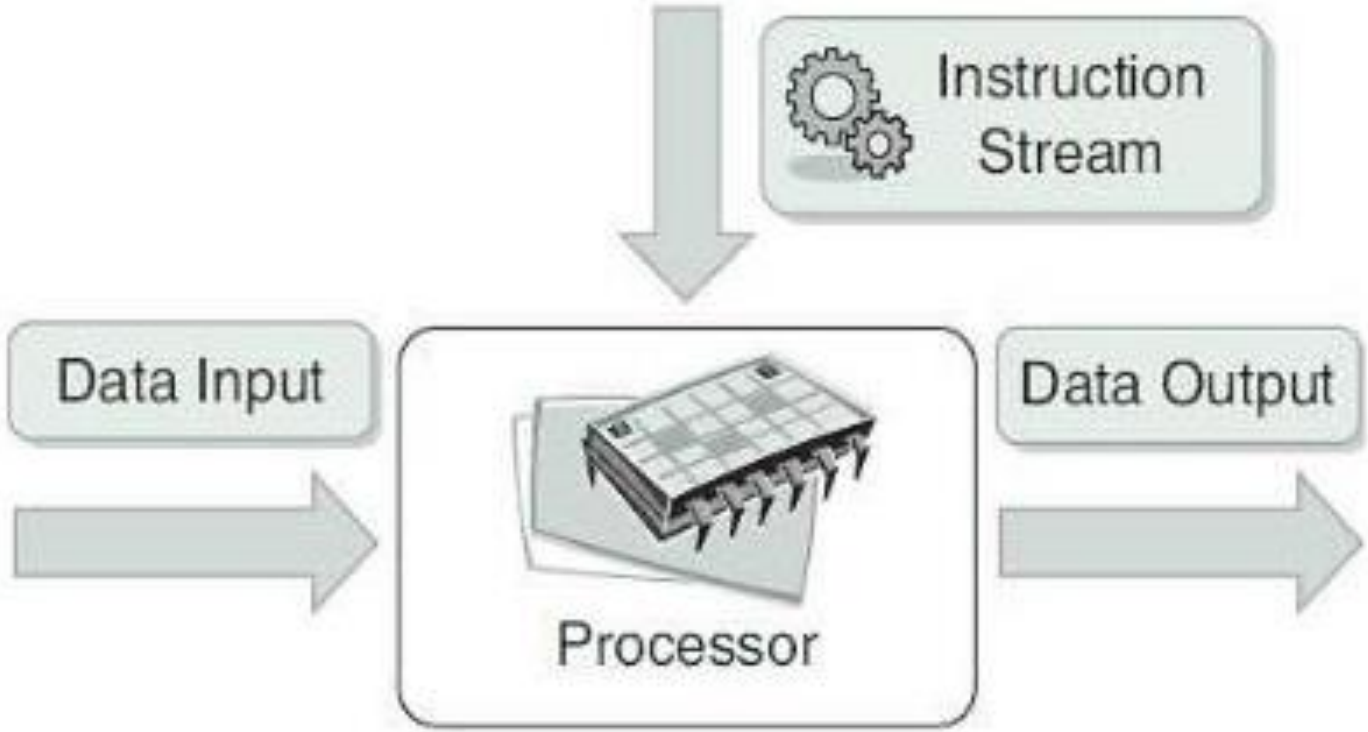
A given task is divided into multiple subtasks using divide-and-conquer technique, and each one of them is processed on different CPUs. Programming on multi-processor system using divide-and-conquer technique is called parallel programming.

The core elements of parallel processing are CPUs. Based on a number of instruction and data streams that can be processed simultaneously, computing systems are classified into the following four categories:

- Single Instruction Single Data (SISD)
- Single Instruction Multiple Data (SIMD)
- Multiple Instruction Single Data (MISD)
- Multiple Instruction Multiple Data (MIMD)

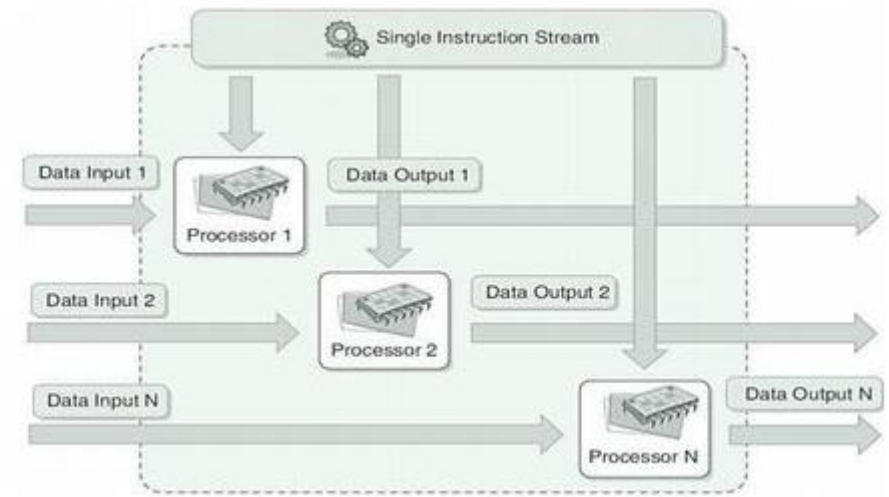
# Single Instruction Single Data (SISD)

- A SISD computing system is a uniprocessor machine capable of executing a single instruction, which operates on a single data stream. In SISD, machine instructions are processed sequentially, and hence computers adopting this model are popularly called sequential computers.



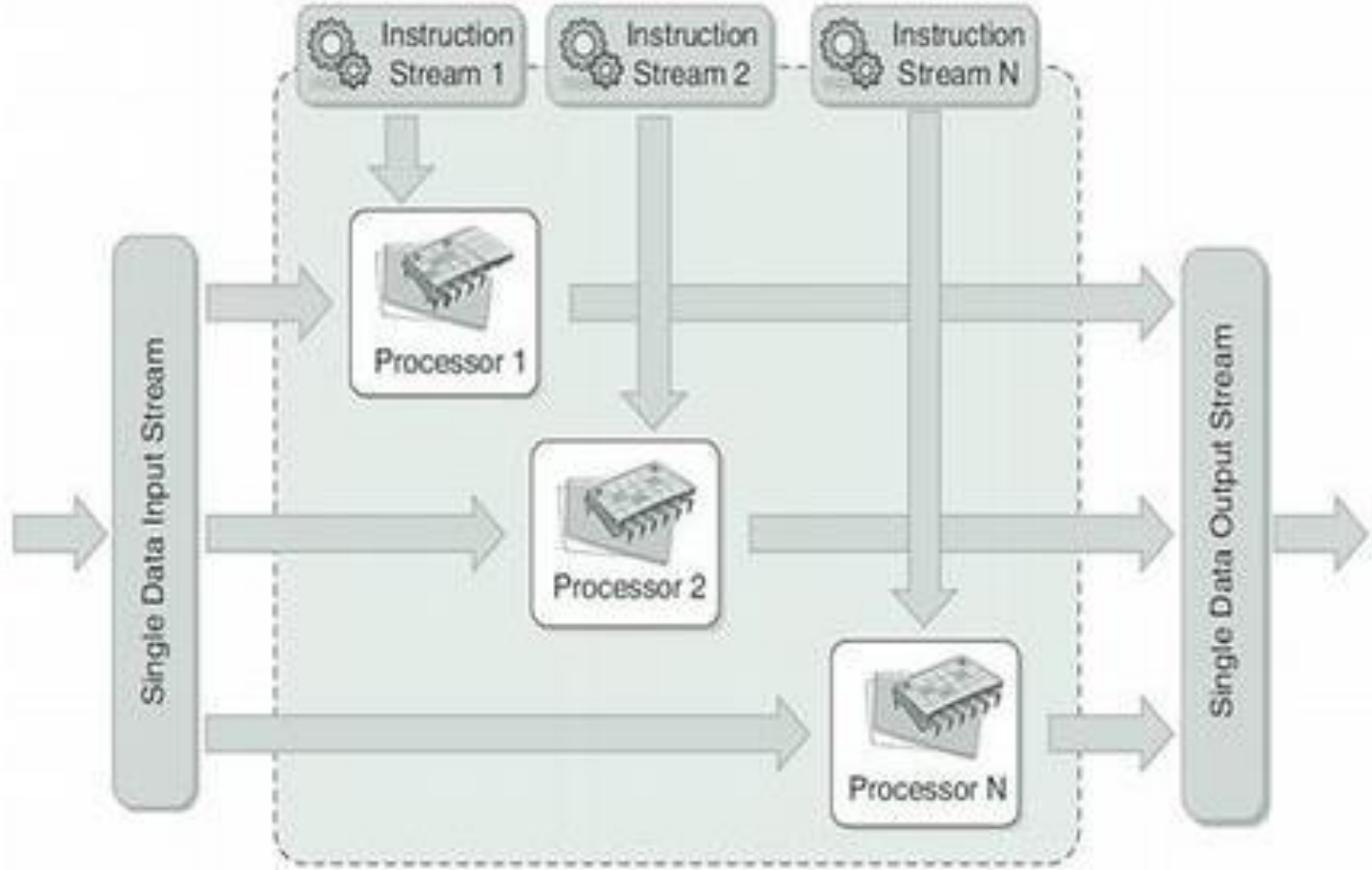
# Single Instruction Multiple Data (SIMD)

- A SIMD computing system is a multiprocessor machine capable of executing the same instruction on all the CPUs, but operating on different data streams. Machines based on SIMD model are well suited for scientific computing since they involve lots of vector and matrix operations.



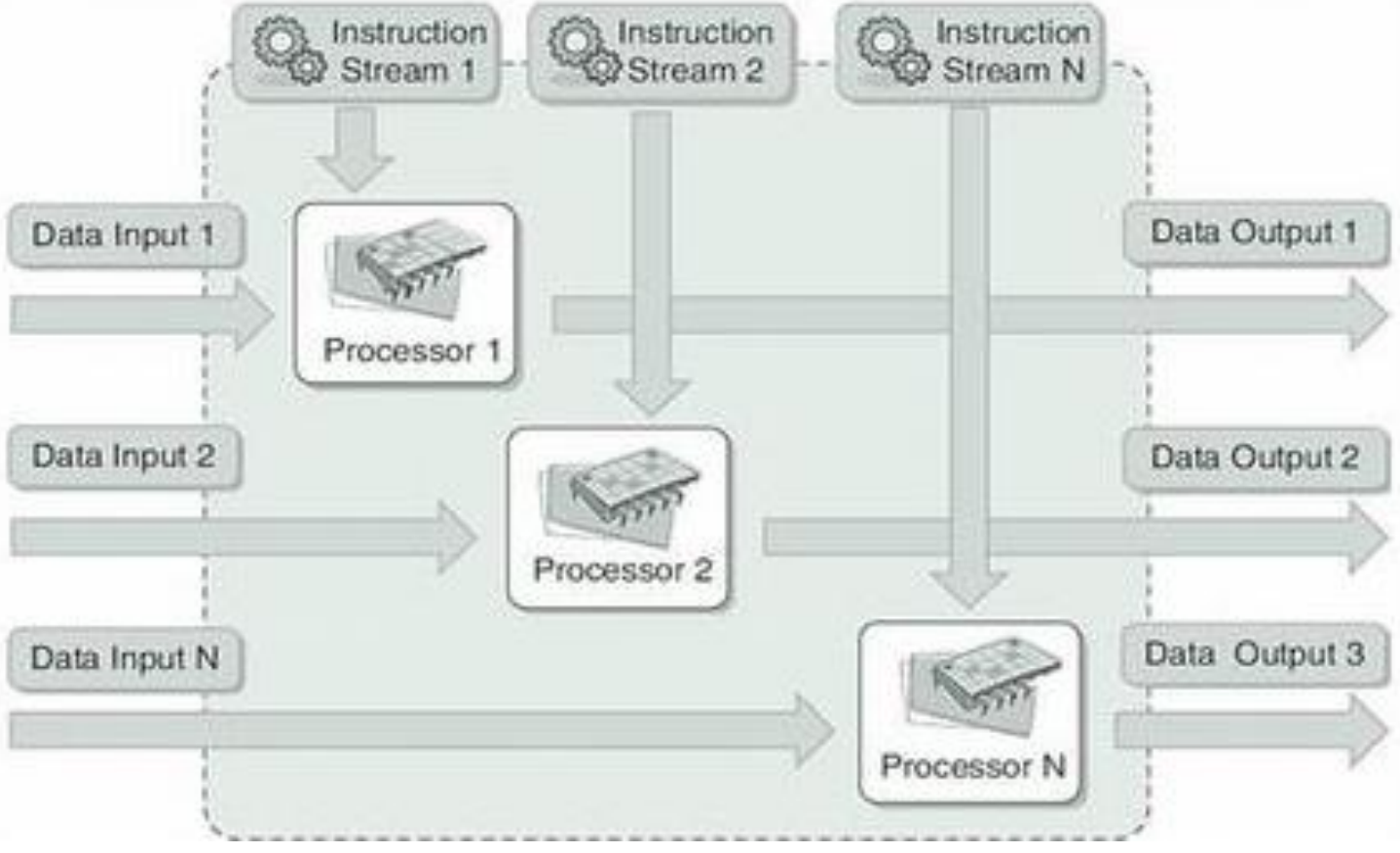
# Multiple Instruction Single Data (MISD)

- A MISD computing system is a multiprocessor machine capable of executing different instructions on different PEs (processing elements), but all of them operating on the same data-set.



# Multiple Instruction Multiple Data (MIMD)

- A MIMD computing system is a multiprocessor machine capable of executing multiple instructions on multiple data sets. Each PE (processing element) in the MIMD model has separate instructions and data streams, and hence machines built using this model are well suited for any kind of application.



# Parallel Programming Approaches

- To make many processors collectively work on a single program, the program must be divided into smaller independent chunks so that each processor can work on separate chunks of the problem.

- The program decomposed in this way is a parallel program. A wide variety of parallel programming approaches are available. The most prominent among them are the following:

- Data Parallelism
- Process Parallelism
- Farmer and Worker Model

# Parallel Programming Approaches

- Data Parallelism :- In case of data parallelism, divide-and-conquer technique is used to split data into multiple sets, and each data set is processed on different PEs by using the same instruction. This approach is highly suitable for processing on machines based on the SIMD model.

- Process Parallelism :- In the case of process parallelism, a given operation has multiple (but distinct) activities, which can be processed on multiple processors.

- Farmer and Worker Model :- In case of farmer and worker model, a job distribution approach is used, one processor is configured as master and all other remaining PEs are designated as slaves; the master assigns a job to slave PEs, and they on completion inform the master which in turn collects the results.

# Serial Computing

Serial Computing is the type of computing where one instruction is given at a particular time and the next instruction has to wait for the first instruction to execute. It is also known as a traditional computing method because all the instructions are executed in a sequence.

It is having a single processor with low performance and high work-load of the processor. The main disadvantage of using this computing is that it takes more time as a single instruction is getting executed at a given point of time.

# Difference between parallel and serial Computing

| Serial computing | Parallel computing |
| --- | --- |
| A single processor is used | Multiple processors are used with shared |
| A problem is divided into a series of instructions | A problem is divided into smaller ones that can be solved simultaneously |
| Instructions executed sequentially | Instructions executed simultaneously |
| One instruction is executed on a single processor at any moment | More than one instruction is executed on multiple processors at any moment of time. |

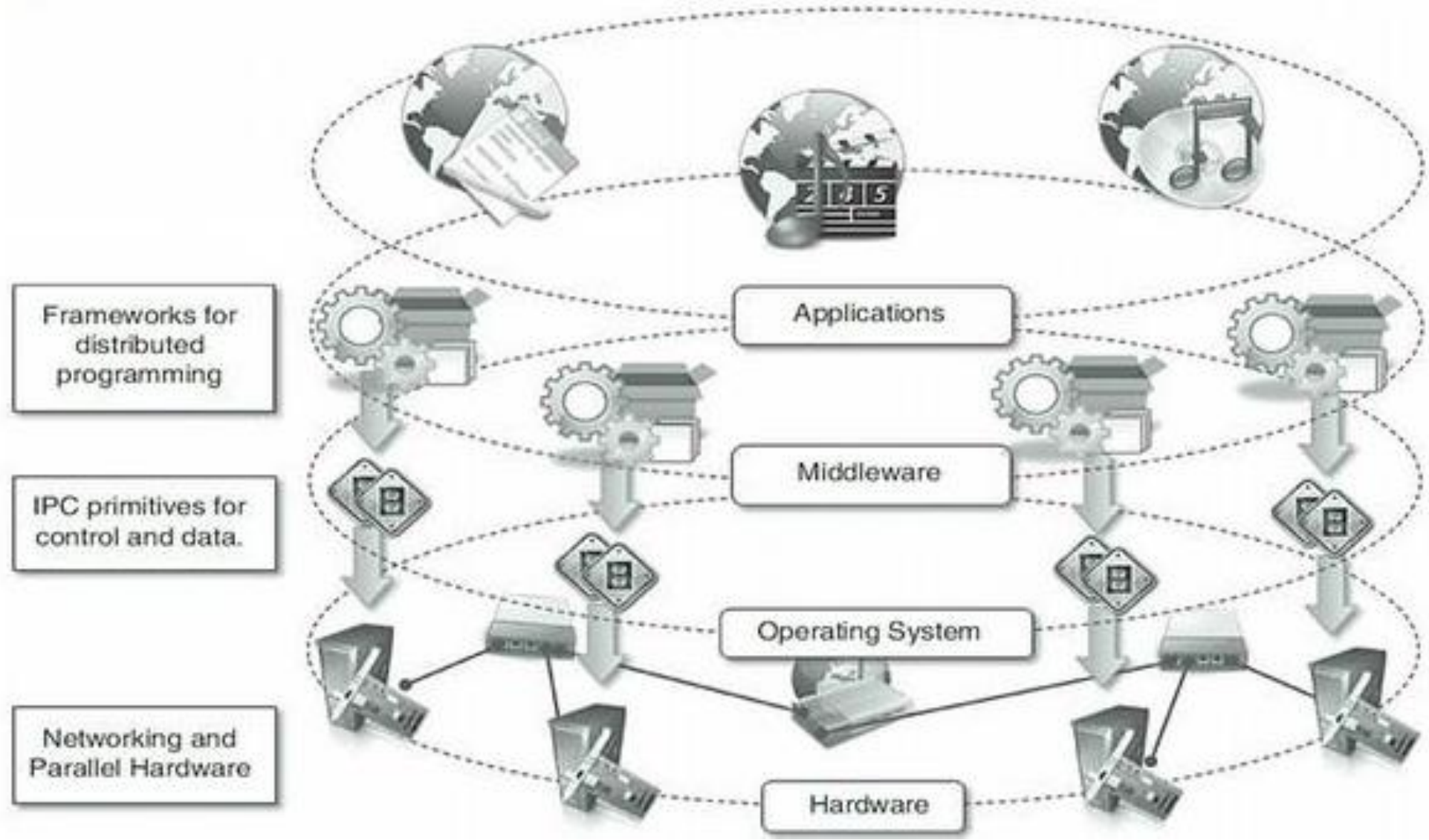
# Components of a Distributed System

- A distributed system is the result of the interaction of several components that traverse the entire computing stack from hardware to software. It emerges from the collaboration of several elements that working together.

- At the very bottom layer, computer and network hardware constitute the physical infrastructure, these components are directly managed by the operating system that provides the basic services for, inter-process communication, process scheduling and management, and resource management in terms of file system and local devices.

- Taken together, these two layers become the platform on top of which specialized software is deployed to turn a set of networked computers into a distributed system.

# Components of a Distributed System



# Components of a Distributed System

- Architectural styles for distributed systems are helpful in understanding the different roles of components in the system and how they are distributed across multiple machines. We organize the architectural styles into two major classes.

1. Software Architectural Styles
2. System Architectural Styles

- Software architectural styles :- Software architectural styles are based on the logical arrangement of software components. They are helpful because they provide an intuitive view of the whole system, despite its physical deployment.

- Data-Centered Architectures :- These architectures identify the data as the fundamental element of the software system and access to shared data is the core characteristic of the data centered architectures. Therefore, especially within the context of distributed and parallel computing systems, integrity of data is the overall goal for such systems.

# Components of a Distributed System

- Data-Flow Architectures :- In the case of data-flow architectures, it is the availability of data that controls the computation. With respect to the data-centered styles, where the access to data is the core feature, data-flow styles explicitly incorporate the pattern of data flow, since their design is determined by an orderly motion of data from component to component, which is the form of communication between them.

- Virtual Machine Architectures :- This class of architectural styles is characterized by the presence of an abstract execution environment (generally referred as virtual machine) that simulates features that are not available in the hardware or software. Applications and systems are implemented on top of this layer and become portable over different hardware and software environment as long as there is an implementation of the virtual machine they interface with.

# Components of a Distributed System

- Call and Return Architectures :- This category identifies all those systems that are composed by components mostly connected together by method calls. The activity of systems modelled in this way is characterized by a chain of method calls whose overall execution and composition identify the execution of one or more operations.

- System Architectural Styles :- System architectural styles cover the physical organization of components and processes over a distributed infrastructure.

- Client-Server :- This architecture is very popular in distributed computing and it is suitable for a wide variety of applications. the client-server model features two major components: a server and a client. These two components interact with each other through a network connection by using a given protocol.

- Peer-to-Peer :- Each peer acts as a server when it processes requests from other peers and as a client when it issues requests to other peers. the client-server model which partitions the responsibilities of the inter-process communication between server and clients, the peer-to-peer model attributes the same responsibilities to each component

# Models for Inter-Process Communication

- Distributed systems are composed of a collection of concurrent processes interacting with each other by means of a network connection.

- Therefore, inter-process communication (IPC) is a fundamental aspect of distributed systems design and implementation. IPC is used to either exchange data and information or to coordinate the activity of processes.

- It is what ties together the different components of a distributed system, thus making them acting as a single system.

- Inter-Process Communication:- Inter-process communication is the mechanism provided by the operating system that allows processes to communicate with each other. This communication could involve a process letting another process know that some event has occurred or transferring of data from one process to another.

# Models for Inter-Process Communication

- Message-based Communication :- The term "message", in this case, identifies any discrete amount of information that is passed from one entity to another. It encompasses any form of data representation that is limited in size and time, whereas this is an invocation to a remote procedure or a serialized object instance or a generic message.

- Message Passing
- Remote Procedure Call (RPC)
- Distributed Objects
- Distributed Agents and Active Objects
- Web Services

# Models for Message-based Communication

- Point-to-Point Message Model :- This model organizes the communication among single components. Each message is sent from one component to another, and there is a direct addressing to identify the message receiver. In a point-to-point communication model, it is necessary to know the location or how to address another component in the system.

- Publish-Subscribe Message Model :- This model introduces a different strategy which is based on notification among components. There are two major roles, the publisher and the subscriber.

- Push Strategy :- In this case, it is the responsibility of the publisher to notify all the subscribers, for example, with a method invocation.

- Pull Strategy :- In this case, the publisher simply makes available the message for a specific event, and it is the responsibility of the subscribers to check whether there are messages on the events that are registered.

# Models for Message-based Communication

- Request-Reply Message Model :- The request-reply message model identifies all those communication models, where for each of the message sent by a process, there is a reply. This model is quite popular, and provides a different classification that does not focus on the number of the components involved in the communication, but on how the dynamic of the interaction evolves.

# Service-Oriented Computing

- Service-oriented computing organizes distributed systems in terms of services, which represent the major abstraction for building systems. Service orientation expresses applications and software systems as aggregation of services that are coordinated within a Service Oriented Architecture (SOA).

- Web services, the fundamental component enabling Cloud computing systems, leverage the Internet as the main interaction channel between users and the system.

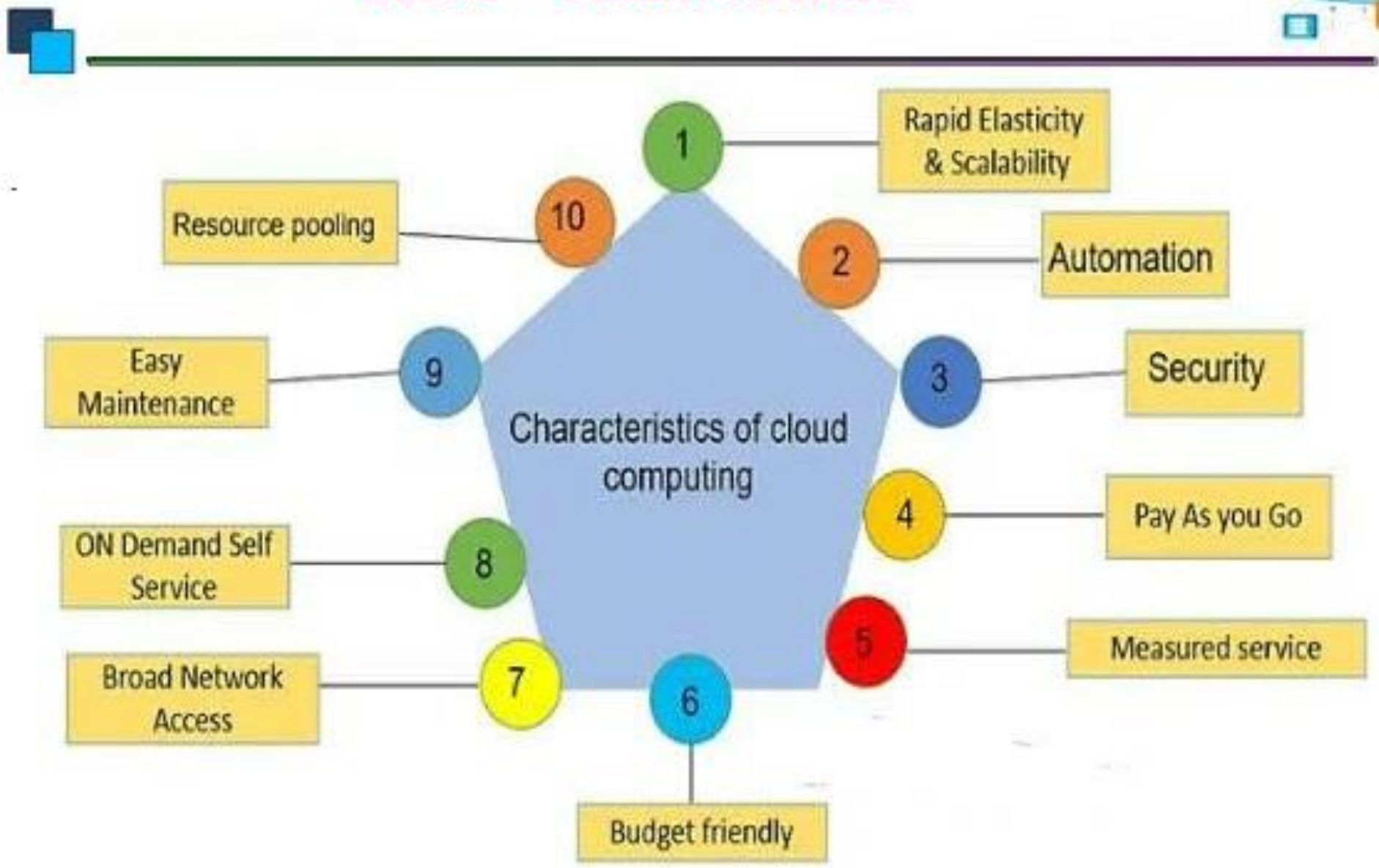
- What is Service :- A service encapsulates a software component providing a set of coherent and related functionalities that can be reused and integrated into bigger and more complex applications.

# Service-Oriented Computing

- Service Oriented Architecture (SOA) :- A Service-Oriented Architecture or SOA is a design pattern which is designed to build distributed systems that deliver services to other applications through the protocol. It is only a concept and not limited to any programming language or platform.

- Web Services :- Web Services are the prominent technology for implementing SOA systems and applications. They leverage Internet technologies and standards for building distributed systems. Several aspects make Web Services the technology of choice for SOA. First of all, they allow for interoperability across different platforms and programming languages. Secondly, they are based on well-known and vendor-independent standards such as HTTP, SOAP, XML, and WSDL. Thirdly, they provide an intuitive and simple way to connect heterogeneous software systems enabling the quick composition of services in a distributed environment.

# Cloud Characteristics



# Issues in Cloud Computing

1. **Privacy**
2. **Compliance**
3. **Security**
4. **Lack of resources/skilled expertise**
5. **Transparency of service provider**
6. **Computing Performance**

# Cloud Security Challenges

1. Data Loss
2. User Account Hijacking
3. Denial of Service (DoS) attack
4. Changing Service Provider
5. Insecure access control points
6. Lack of Skill
7. Avoiding Misconfigured Cloud Infrastructure
8. Lack of Cloud Security Strategy
9. External Sharing of Data
10. Malicious Insiders

# How is security provided to data at various stages in context of cloud

1. Prevent Insecure Interfaces and APIs
2. Prevent Data Privacy Issues
3. Data classification
4. Prevent Cloud Service Hijacking Issues
5. Malicious behaviour identification
6. Collaboration controls
7. Use Encryption
8. Apply Reliable Passwords
9. Test Your Security
10. Risk Assessment
11. Malware Prevention
12. Privileged Access

# Components of Cloud Computing

- Clients
- Data Centre
- Distributed Servers
- Cloud Services
- Cloud Applications
- Cloud Platform
- Internet

# Elasticity in Cloud

- Cloud Elasticity :- The Elasticity refers to the ability of a cloud to automatically expand or compress the infrastructural resources on a sudden-up and down in the requirement so that the workload can be managed efficiently. This elasticity helps to minimize infrastructural cost.

- Cloud Elasticity is the property of a cloud to grow or shrink capacity for CPU, memory, and storage resources to adapt to the changing demands of an organization.

- Cloud Elasticity can be automatic, without need to perform capacity planning in advance of the occasion, or it can be a manual process where the organization is notified they are running low on resources and can then decide to add or reduce capacity when needed.

- Monitoring tools offered by the cloud provider dynamically adjust the resources allocated to an organization without impacting existing cloud-based operations.

# Cloud Scalability

- Cloud Scalability : - Cloud scalability is used to handle the growing workload where good performance is also needed to work efficiently with software or applications. Scalability is commonly used where the persistent deployment of resources is required to handle the workload statically.

# Difference Between Cloud Elasticity and Scalability

| | Cloud Elasticity | Cloud Scalability |
|---|---|---|
| 1 | Elasticity is used just to meet the sudden up and down in the workload for a small period of time. | Scalability is used to meet the static increase in the workload. |
| 2 | Elasticity is used to meet dynamic changes, where the resources need can increase or decrease. | Scalability is always used to address the increase in workload in an organization. |
| 3 | Elasticity is commonly used by small companies whose workload and demand increases only for a specific period of time. | Scalability is used by giant companies whose customer circle persistently grows in order to do the operations efficiently. |
| 4 | It is a short term planning and adopted just to deal with an unexpected increase in demand or seasonal demands. | Scalability is a long term planning and adopted just to deal with an expected increase in demand. |