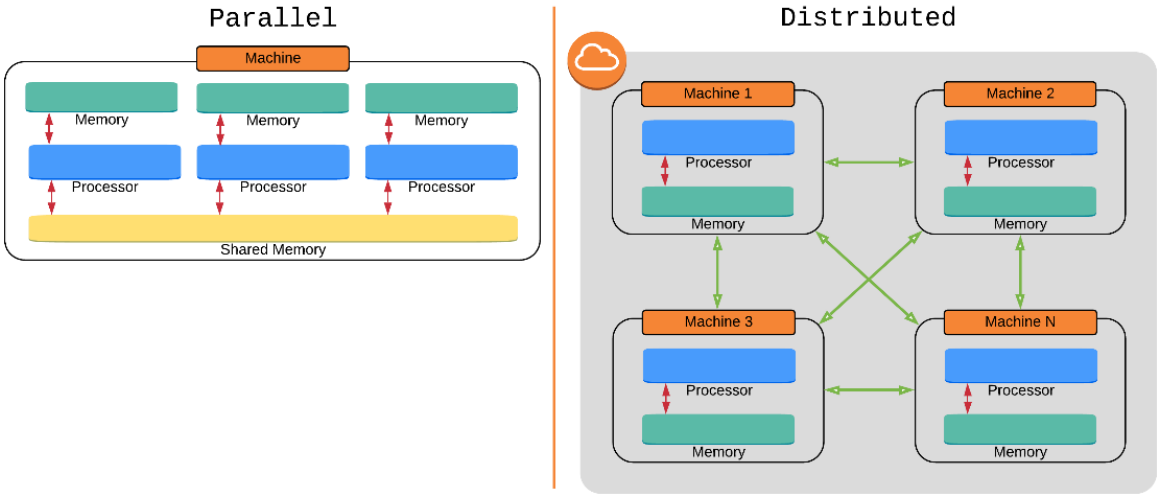
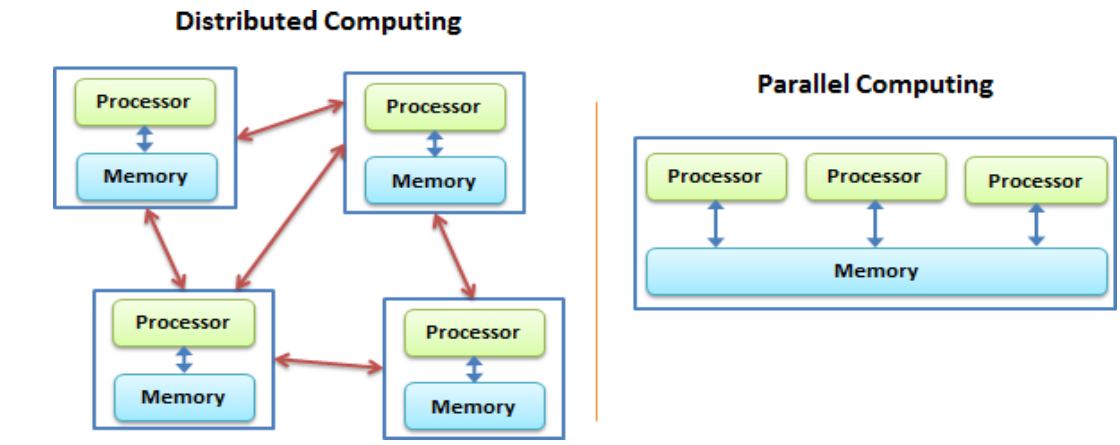
SEC-302-CLOUD COMPUTING UNIT-2

- **Principles of Parallel and Distributed Computing**

- Cloud computing is a new technological trend that supports better utilization of IT infrastructures, services, and applications.
- It adopts a service delivery model based on a pay-per-use approach, in which users do not own infrastructure, platform, or applications but use them for the time they need them.
- These IT assets are owned and maintained by service providers who make them accessible through the Internet.
- **Parallel Computing:**
  In parallel computing multiple processors performs multiple tasks assigned to them simultaneously. Memory in parallel systems can either be shared or distributed. Parallel computing provides concurrency and saves time and money.
- **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.





Prepared By:- Asst. Prof. Nirav Shukla(CPICA)

SEC-302-CLOUD COMPUTING UNIT-2

❖ **Parallel VS. Distributed Computing:**

	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 communicates with each other through message passing.
6.	Improves the system performance	Improves system scalability, fault tolerance and resource sharing capabilities

- **Eras of Computing :**

- Cloud computing an emerging technology provides various services to the users like infrastructure, hardware, software, storage etc.
- For working cloud in data security, it is necessary that cloud computing network should always be free from outside attack/ threats.
- The service of cloud like IaaS (Infrastructure as a Service) allows an internet trade a way to build up and produce on demand.
- PaaS (Platform as a Service): It is the client who controls the applications that run in the environment, but does not manage the operating system, hardware on which they are running.

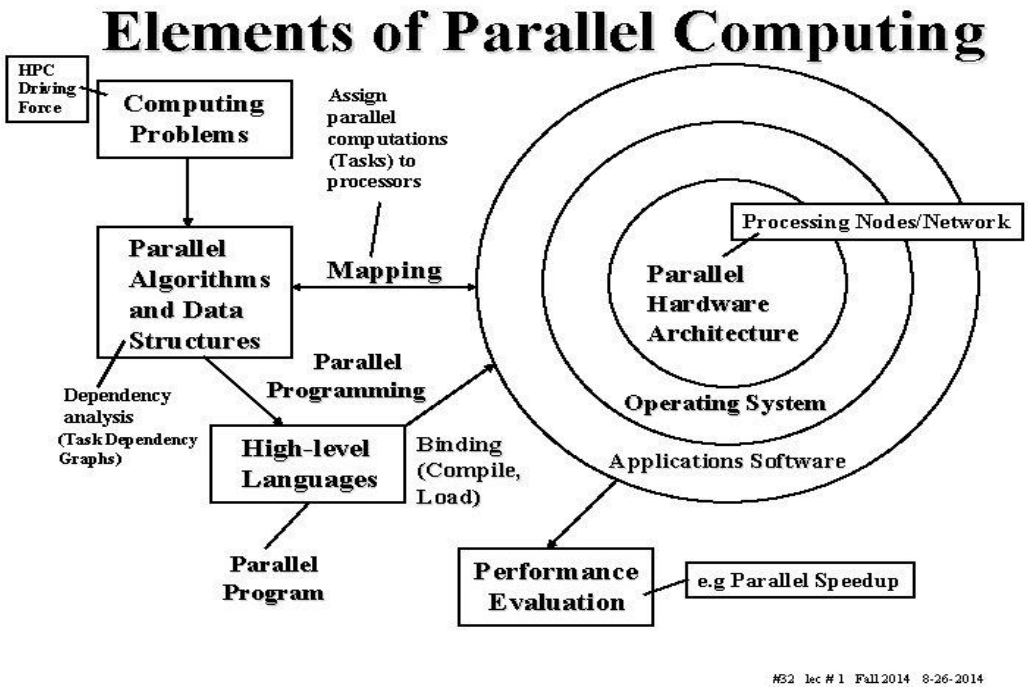
- Need of cloud computing is an expertise uses the internet and central remote servers to maintain data and applications. Some reasons are Speedy Elasticity, Measured Service, on-Demand Self-Service etc. Cloud computing is also known as fifth generation of computing after supercomputer, Personal Computer, Client-Server Computing, and the Web.
- One of the issues of cloud computing is data security in which data can be loss or hacked by the attacker, possible solution of this problem by applying encryption techniques on the data.
- Some challenges are also present in cloud computing like lack of resources in which staff needed new skills or updated knowledge related technology, possible solution is to recruit new staff or give training to the existing ones.
- This survey of cloud computing concluded that Cloud computing is increasing part of IT and many gigantic organizations are going to implement cloud computing.
- In the near future work on data science, artificial intelligence and machine learning service inside cloud provider to protect the customer sensitive data from the intruders attack.

❖ Elements of parallel computing:-

- 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.
- As we learn what is parallel computing and there type now we are going more deeply on the topic of the parallel computing and understand the concept of the hardware architecture of parallel computing.
- The whole real world runs in dynamic nature i.e. many things happen at a certain time but at different places concurrently. This data is extensively huge to manage.
- Real world data needs more dynamic simulation and modeling, and for achieving the same, parallel computing is the key.
- Parallel computing provides concurrency and saves time and money.
- Complex, large datasets, and their management can be organized only and only using parallel computing's approach.
- Ensures the effective utilization of the resources. The hardware is guaranteed to be used effectively whereas in serial computation only some part of hardware was used and the rest rendered idle.
- Also, it is impractical to implement real-time systems using serial computing.

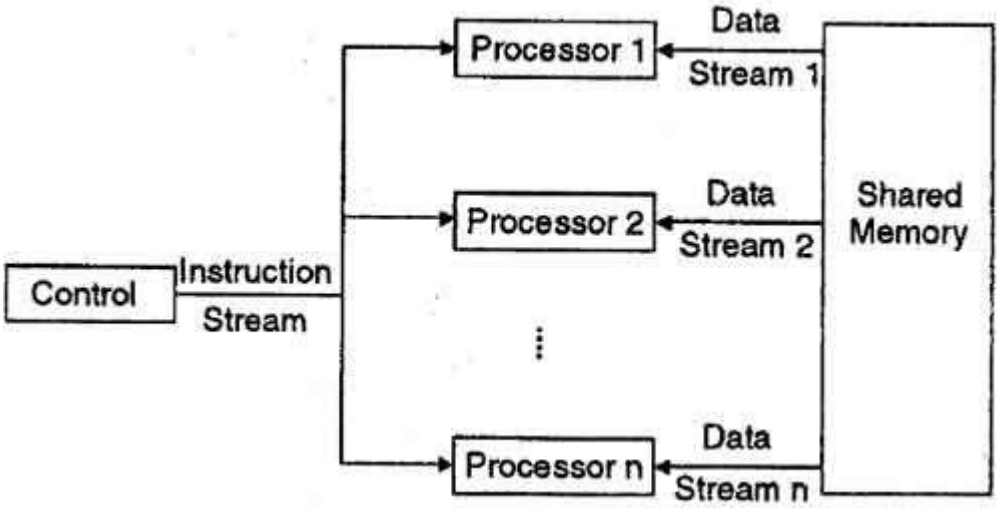
Applications of Parallel Computing:

- Data bases and Data mining.
- Real time simulation of systems.
- Science and Engineering.
- Advanced graphics, augmented reality and virtual reality.



❖ **What is parallel processing?**

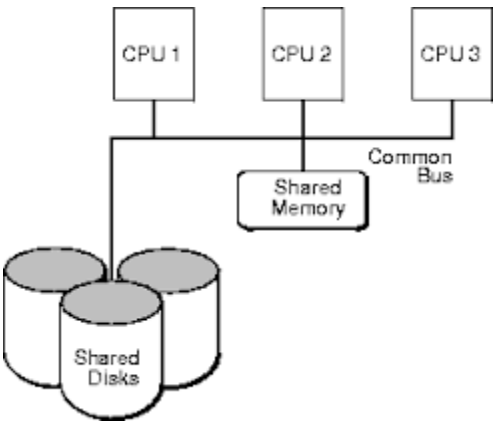
- Parallel processing can be described as a class of techniques which enables the system to achieve simultaneous data-processing tasks to increase the computational speed of a computer system.
- A parallel processing system can carry out simultaneous data-processing to achieve faster execution time. For instance, while an instruction is being processed in the ALU component of the CPU, the next instruction can be read from memory.
- The primary purpose of parallel processing is to enhance the computer processing capability and increase its throughput, i.e. the amount of processing that can be accomplished during a given interval of time.
- A parallel processing system can be achieved by having a multiplicity of functional units that perform identical or different operations simultaneously. The data can be distributed among various multiple functional units.
- The following diagram shows one possible way of separating the execution unit into eight functional units operating in parallel.



❖ Hardware architectures for parallel processing

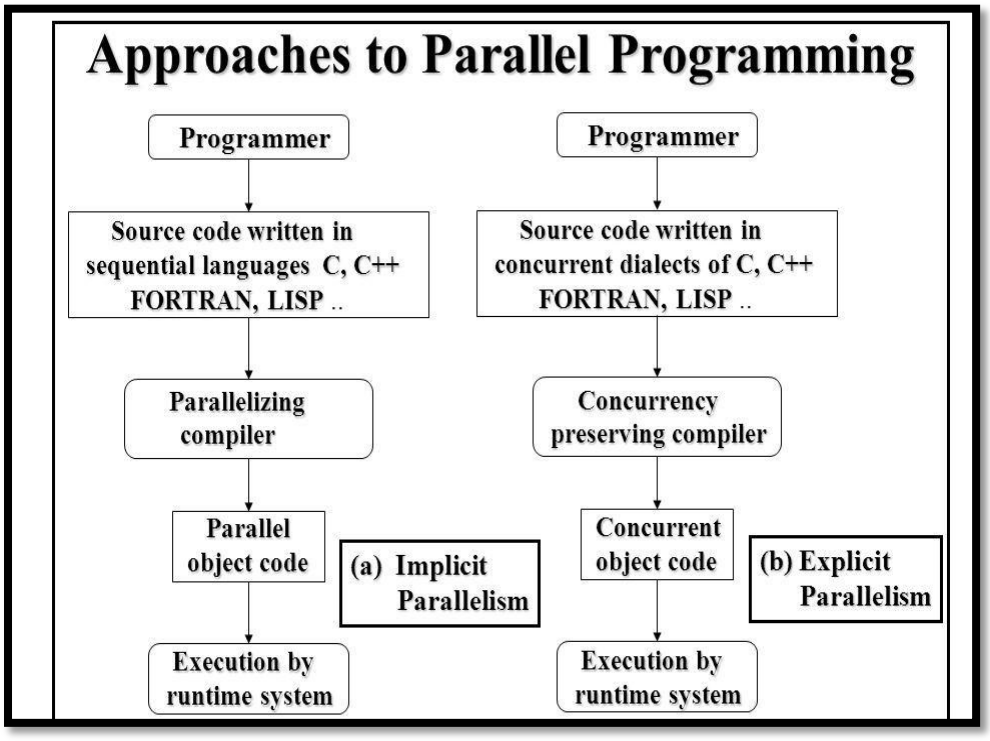
- Parallel processing refers to the use of multiple processors to reduce the time needed to complete a given task.
- Instead of one processor's executing an entire task, several processors each work on a separate piece of the task.
- Obviously, parallel processing requires computer hardware that can support more than a single processor.
- Several architectural approaches to multiple processor systems have been developed over the years.
- The hardware architecture of parallel computing is disturbed along the following categories as given below :

1. Single-instruction, single-data (SISD) systems
2. Single-instruction, multiple-data (SIMD) systems
3. Multiple-instruction, single-data (MISD) systems
4. Multiple-instruction, multiple-data (MIMD) systems



❖ Approaches to parallel programming

- Scientific and engineering computations focus on theories, methods and applications in such areas as large-scale simulations, time-critical computing, computer-aided design and engineering, computer-aided manufacturing, visualization of scientific data and human-machine interface technology.
- Parallelism offers one way to solve these computational problems quickly by creating and coordinating multiple execution processes.
- However, until recently parallelism has been extremely difficult to use because of the lack of suitable parallel programming approaches.
- In general, the acceptance of parallel computation has been facilitated by two major developments: massively parallel processors and the widespread use of distributed computing.
- A good parallel programming environment must fulfill a number of objectives:
- 1. It must augment the sequential programming language that is most appropriate for whatever the computational problem is.
- 2. It must support both process creation and interprocess communication as extensions of the high-level base programming language.
- 3. It must be able to be run on any parallel architecture or on any collection of networked computers.
- 4. It must be easy to use with regard to parallelism implementation, meaning that it must offer simple operations to create and coordinate parallel processes. Common parallel programming paradigms can be divided into two categories:
    - The master-slave model, in which a separate control process termed the master is responsible for process spawning, initialization, collection and display of results, and perhaps timing of functions.
    - The slave processes perform the actual computation involved. Either they are allocated their workloads by the master (statically or dynamically) or perform the allocations themselves.



SEC-302-CLOUD COMPUTING UNIT-2

❖ Elements of distributed computing : General concepts and definitions

- A distributed system is a system whose components are located on different networked computers, which communicate and coordinate their actions by passing messages to one another from any system.
- The components interact with one another in order to achieve a common goal.
- Three significant characteristics of distributed systems are concurrency of components, lack of a global clock, and independent failure of components.
- Examples of distributed systems vary from SOA-based systems to massively multiplayer online games to peer-to-peer applications.
- A computer program that runs within a distributed system is called a distributed program (and distributed programming is the process of writing such programs).
- There are many different types of implementations for the message passing mechanism, including pure HTTP, RPC-like connectors and message queues.
- Distributed computing also refers to the use of distributed systems to solve computational problems.
- In distributed computing, a problem is divided into many tasks, each of which is solved by one or more computers, which communicate with each other via message passing.
- A distributed system may have a common goal, such as solving a large computational problem; the user then perceives the collection of autonomous processors as a unit.
- Alternatively, each computer may have its own user with individual needs, and the purpose of the distributed system is to coordinate the use of shared resources or provide communication services to the users.
- Other typical properties of distributed systems include the following:
    - The system has to tolerate failures in individual computers.
    - The structure of the system (network topology, network latency, number of computers) is not known in advance, the system may consist of different kinds of computers and network links, and the system may change during the execution of a distributed program.

❖ Components of a distributed system :

- There are three basic components of a distributed system.
- These are concurrency of components, lack of a global clock, and independent failure of components.
- Distributed system is a group of networked computers, pursuing the same goal.
- Any distributed system can be used as parallel or distributed as the case may be, where the processors in each case of a typically distributed system run concurrently in parallel.
- In distributed computing system, each processor has its own private memory and information is exchanged by passing messages between the processor.

SEC-302-CLOUD COMPUTING UNIT-2

❖ System architectural styles :

- Distributed systems must have a network that connects all components (machines, hardware, or software) together so they can transfer messages to communicate with each other.
- That network could be connected with an IP address or use cables or even on a circuit board.
- The messages passed between machines contain forms of data that the systems want to share like databases, objects, and files.
- The way the messages are communicated reliably whether it's sent, received, acknowledged or how a node retries on failure is an important feature of a distributed system.
- Distributed systems were created out of necessity as services and applications needed to scale and new machines needed to be added and managed. In the design of distributed systems, the major trade-off to consider is complexity vs performance.

- Distributed applications and processes typically use one of four architecture types below:

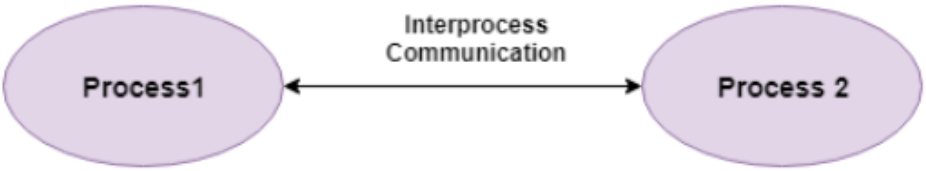
    - ### Client-server:
        - In the early days, distributed systems architecture consisted of a server as a shared resource like a printer, database, or a web server.
        - It had multiple clients (for example, users behind computers) that decide when to use the shared resource, how to use and display it, change data, and send it back to the server.

    - ### Today, distributed systems architecture has evolved with web applications into:

        - **Three-tier:** In this architecture, the clients no longer need to be intelligent and can rely on a middle tier to do the processing and decision making. Most of the first web applications fall under this category. The middle tier could be called an agent that receives requests from clients that could be stateless, processes the data and then forwards it on to the servers.
        - **Multi-tier:** Enterprise web services first created n-tier or multi-tier systems architectures. This popularized the application servers that contain the business logic and interacts both with the data tiers and presentation tiers.
        - **Peer-to-peer:** There are no centralized or special machine that does the heavy lifting and intelligent work in this architecture. All the decision making and responsibilities are split up amongst the machines involved and each could take on client or server roles. Blockchain is a good example of this.

❖ Models for interprocess communication :

- Interprocess 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.
- A diagram that illustrates interprocess communication is as follows :



- The models of interprocess communication are as follow :

Shared Memory Model

- Shared memory is the memory that can be simultaneously accessed by multiple processes. This is done so that the processes can communicate with each other. All POSIX systems, as well as Windows operating systems use shared memory.

Advantage of Shared Memory Model

- Memory communication is faster on the shared memory model as compared to the message passing model on the same machine.

Disadvantages of Shared Memory Model

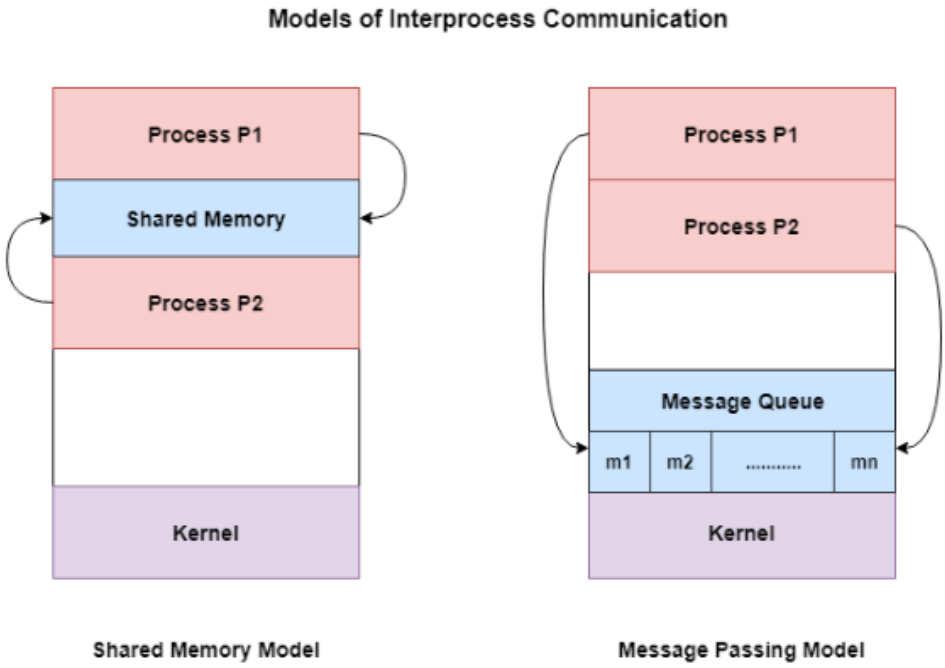
Some of the disadvantages of shared memory model are as follows −

- All the processes that use the shared memory model need to make sure that they are not writing to the same memory location.
- Shared memory model may create problems such as synchronization and memory protection that need to be addressed.

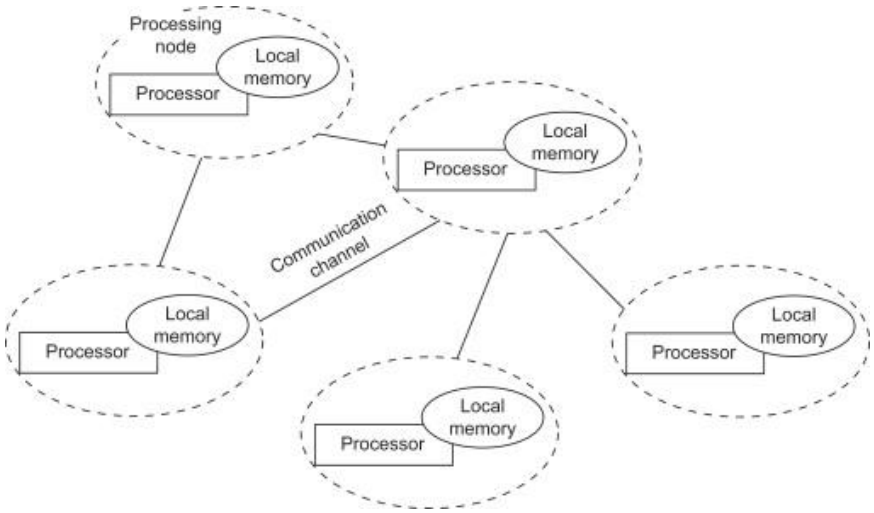
Message Passing Model

- Multiple processes can read and write data to the message queue without being connected to each other. Messages are stored on the queue until their recipient retrieves them. Message queues are quite useful for interprocess communication and are used by most operating systems.

SEC-302-CLOUD COMPUTING UNIT-2



❖ **Technologies for distributed computing :**



- Distributed computing is a much broader technology that has been around for more than three decades now.
- Simply stated, distributed computing is computing over distributed autonomous computers that communicate only over a network.
- Distributed computing systems are usually treated differently from parallel computing systems or shared-memory systems, where multiple computers share a common memory pool that is used for communication between the processors.

Prepared By:- Asst. Prof. Nirav Shukla(CPICA)

- Distributed memory systems use multiple computers to solve a common problem, with computation distributed among the connected computers (nodes) and using message-passing to communicate between the nodes.
- For example, grid computing, studied in the previous section, is a form of distributed computing where the nodes may belong to different administrative domains.
- Another example is the network-based storage virtualization solution, which used distributed computing between data and metadata servers. Developing applications for distributed memory machines is much more involved than traditional sequential machines.
- Sometimes new algorithms need to be developed to solve even a well-known problem (sorting huge sequences of numbers). In order to ease the burden on programmers, parallelizing compilers that convert sequential programs written for traditional computers to distributed message programs exist, particularly for distributed SMP (symmetric multiprocessor) clusters.
- Distributed computing, however, can include heterogeneous computations where some nodes may perform a lot more computation, some perform very little computation and a few others may perform specialized functionality (like processing visual graphics).
- One of the main advantages of using distributed computing is that efficient scalable programs can be designed so that independent processes are scheduled on different nodes and they communicate only occasionally to exchange results – as opposed to working out of a shared memory with multiple simultaneous accesses to a common memory.

❖ Service-oriented computing :

- A service-oriented architecture (SOA) focuses on services and is geared towards addressing the individual needs and processes of company.
- This allows individual services to be combined into a bespoke business process.
- For example, an SOA can cover the entire process of “ordering online” which involves the following services: “taking the order”, “credit checks” and “sending the invoice”.
- Technical components (e.g. servers, databases, etc.) are used as tools but are not the main focus here.
- In this type of distributed computing, priority is given to ensuring that services are effectively combined, work together well, and are smartly organized with the aim of making business processes as efficient and smooth as possible.
- In a service-oriented architecture, extra emphasis is placed on well-defined interfaces that functionally connect the components and increase efficiency.
- These can also benefit from the system’s flexibility since services can be used in a number of ways in different contexts and reused in business processes.
- Service-oriented architectures using distributed computing are often based on web services.
- They are implemented on distributed platforms, such as CORBA, MQSeries, and J2EE.