

## Part A - Comparing features of Operating system Realtime Operating system, Distributed operating system

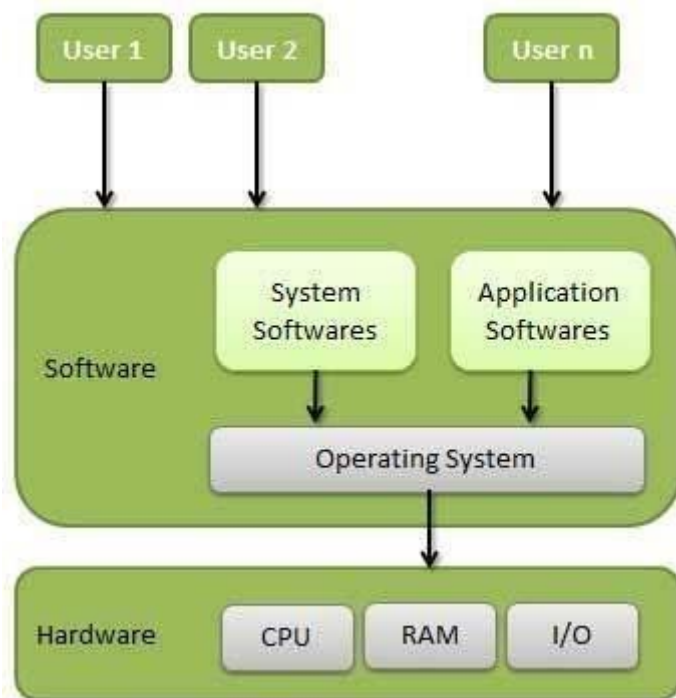
### 1. Brief Introduction:

An Operating System (OS) is an interface between a computer user and computer hardware. An operating system is a software which performs all the basic tasks like file management, memory management, process management, handling input and output, and controlling peripheral devices such as disk drives and printers.

Some popular Operating Systems include Linux Operating System, Windows Operating System, VMS, OS/400, AIX, z/OS, etc.

#### Definition

An operating system is a program that acts as an interface between the user and the computer hardware and controls the execution of all kinds of programs.



### Real-time Operating System

A real-time operating system is an operating system that guarantees to process events or data by a specific moment in time. A real-time operating system may be single- or multi-tasking, but when

multitasking, it uses specialized scheduling algorithms so that a deterministic nature of behavior is achieved. Such an event-driven system switches between tasks based on their priorities or external events, whereas time-sharing operating systems switch tasks based on clock interrupts

### **Distributed Operating System**

A distributed operating system manages a group of distinct, networked computers and makes them appear to be a single computer, as all computations are distributed (divided amongst the constituent computers).

## **1.0 Aim of the Micro-Project**

This Micro-Project aims at

1. manage the computer's resources, such as the central processing unit, memory, disk drives,
2. and printers.
3. establish a user interface.
4. execute and provide services for applications software.

## **2.0 Resources Requires :**

<b>Sr. No.</b>	<b>Name of resource / material</b>	<b>Specification</b>	<b>Quantity</b>	<b>Remarks</b>
1	Internet	To collect information from various website	1	Which ever is available
2	Desktop	Pentium iv or above with keyboard,mouse	1	Which ever is available

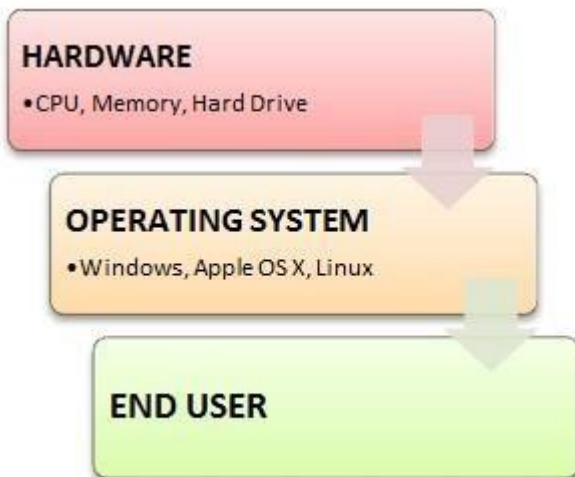
\*\*\*\*\*

## **Part B - Outcome after Execution Format for Micro-Project Report Comparing features of operating system Realtime Operating system, Distributed operating system**

### **1.0 Brief Description:**

#### **What is an Operating System?**

An **Operating System (OS)** is a software that acts as an interface between computer hardware components and the user. Every computer system must have at least one operating system to run other programs. Applications like Browsers, MS Office, Notepad Games, etc., need some environment to run and perform its tasks



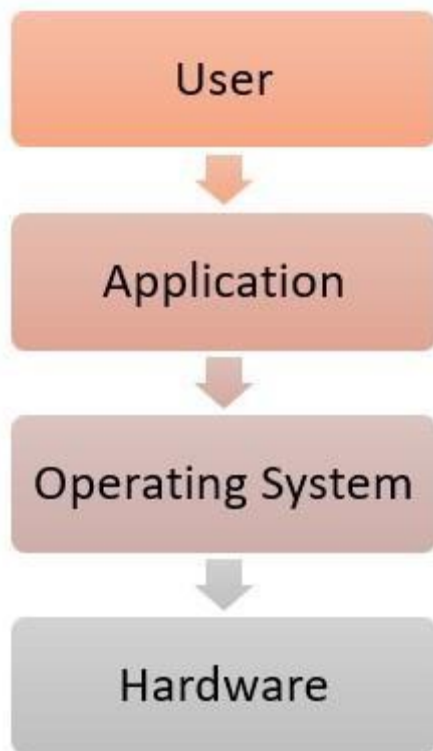
#### **1.1 Following are the popular types of Operating System:**

- Batch Operating System
- Multitasking/Time Sharing OS
- Multiprocessing OS
- Real Time OS
- Distributed OS
- Network OS
- Mobile OS

## 1.2 Features of Operating System

Here is a list important features of OS:

- Protected and supervisor mode
- Allows disk access and file systems Device drivers Networking Security
- Program Execution
- Memory management Virtual Memory Multitasking
- Handling I/O operations
- Manipulation of the file system
- Error Detection and handling
- Resource allocation
- Information and Resource Protection



## 1.3 Functions of Operating system

Below are the main functions of Operating System:



In an operating system software performs each of the function:

1. **Process management:-** Process management helps OS to create and delete processes. It also provides mechanisms for synchronization and communication among processes.
2. **Memory management:-** Memory management module performs the task of allocation and de-allocation of memory space to programs in need of this resources.
3. **File management:-** It manages all the file-related activities such as organization storage, retrieval, naming, sharing, and protection of files.
4. **Device Management:** Device management keeps tracks of all devices. This module also responsible for this task is known as the I/O controller. It also performs the task of allocation and de-allocation of the devices.
5. **I/O System Management:** One of the main objects of any OS is to hide the peculiarities of that hardware devices from the user.
6. **Secondary-Storage Management:** Systems have several levels of storage which includes primary storage, secondary storage, and cache storage. Instructions and data must be stored in primary storage or cache so that a running program can reference it.
7. **Security:-** Security module protects the data and information of a computer system against malware threat and authorized access.
8. **Command interpretation:** This module is interpreting commands given by the and acting system resources to process that commands.
9. **Networking:** A distributed system is a group of processors which do not share memory, hardware devices, or a clock. The processors communicate with one another through the network.

10. **Job accounting:** Keeping track of time & resource used by various job and users.

11. **Communication management:** Coordination and assignment of compilers, interpreters, and another software resource of the various users of the computer systems.

### **1.4 Advantages of Operating System:**

- Allows you to hide details of hardware by creating an abstraction
- Easy to use with a GUI
- Offers an environment in which a user may execute programs/applications
- The operating system must make sure that the computer system convenient to use
- Operating System acts as an intermediary among applications and the hardware components
- It provides the computer system resources with easy to use format
- Acts as an intermediary between all hardware's and software's of the system

### **1.5 Disadvantages of Operating System:**

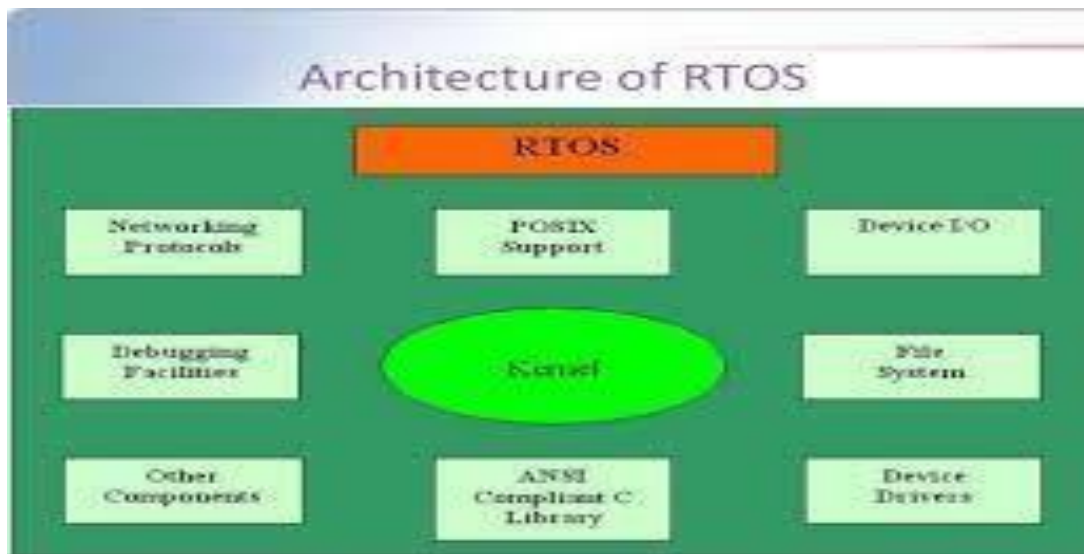
- If any issue occurs in OS, you may lose all the contents which have been stored in your system
- Operating system's software is quite expensive for small size organization which adds burden on them. Example Windows
- It is never entirely secure as a threat can occur at any time

\*\*\*\*\*

## **2.0 Real Time Operating System**

**Real-time operating system (RTOS)** is an operating system intended to serve real time application that process data as it comes in, mostly without buffer delay. The full form of RTOS is Real time operating system.

In a RTOS, Processing time requirement are calculated in tenths of seconds increments of time. It is time-bound system that can be defined as fixed time constraints. In this type of system, processing must be done inside the specified constraints. Otherwise, the system will fail.



## 2.1 Types of Real time operating system

Three types of Real time operating systems are:

### 1.Hard Real Time :

In Hard RTOS, the deadline is handled very strictly which means that given task must start executing on specified scheduled time, and must be completed within the assigned time duration.

Example: Medical critical care system, Aircraft systems, etc.

### 2.Firm Real time:

These type of RTOS also need to follow the deadlines. However, missing a deadline may not have big impact but could cause undesired affects, like a huge reduction in quality of a product.

Example: Various types of Multimedia applications.

### 3.Soft Real Time:

Soft Real time RTOS, accepts some delays by the Operating system. In this type of RTOS, there is a deadline assigned for a specific job, but a delay for a small amount of time is acceptable. So, deadlines are handled softly by this type of RTOS.

Example: Online Transaction system and Livestock price quotation System.

## 2.2 Terms used in Real time operating system

Here, are essential terms used in Real time operating system:

- **Task** – A set of related tasks that are jointly able to provide some system functionality.
- **Job** – A job is a small piece of work that can be assigned to a processor, and that may or may not require resources.
- **Release time of a job** – It's a time of a job at which job becomes ready for execution.
- **Execution time of a job:** It is time taken by job to finish its execution.
- **Deadline of a job:** It's time by which a job should finish its execution.
- **Processors:** They are also known as active resources. They are important for the execution of a job.
- **Maximum It is the** allowable response time of a job is called its relative deadline.
- **Response time of a job:** It is a length of time from the release time of a job when the instant finishes.
- **Absolute deadline:** This is the relative deadline, which also includes its release time. **2.3**

## Features of Real time operating system

Here are important features of RTOS:

- Occupy very less memory
- Consume fewer resources
- Response times are highly predictable
- Unpredictable environment
- The Kernel saves the state of the interrupted task and then determines which task it should run next.
- The Kernel restores the state of the task and passes control of the CPU for that task. **2.4**

## Applications of Real Time Operating System

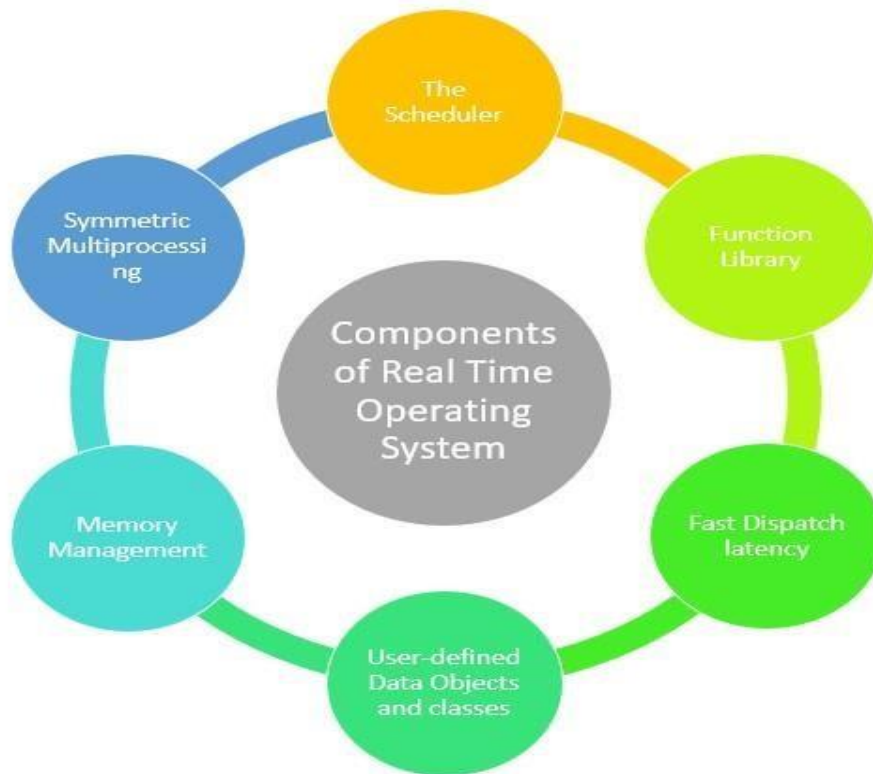
Real-time systems are used in:

- Airlines reservation system.
- Air traffic control system.
- Systems that provide immediate updating.
- Used in any system that provides up to date and minute information on stock prices.
- Defence application systems like RADAR.
- Networked Multimedia Systems
- Command Control Systems
- Internet Telephony
- Anti-lock Brake Systems



- Heart Pacemaker

## 2.5 Components of RTOS



Components of Real Time Operating System

Here, are important Component of Real Time Operating System

**1.The Scheduler:** This component of RTOS tells that in which order, the tasks can be executed which is generally based on the priority.

**2.Symmetric Multiprocessing (SMP):** It is a number of multiple different tasks that can be handled by the RTOS so that parallel processing can be done.

**3.Function Library:** It is an important element of RTOS that acts as an interface that helps you to connect kernel and application code. This application allows you to send the requests to the Kernel using a function library so that the application can give the desired results.

**4.Memory Management:** this element is needed in the system to allocate memory to every program, which is the most important element of the RTOS.

**5.Fast dispatch latency:** It is an interval between the termination of the task that can be identified by the OS and the actual time taken by the thread, which is in the ready queue, that has started processing.

**6.User-defined data objects and classes:** RTOS system makes use of programming languages like C or C++, which should be organized according to their operation.

## **2.6 Advantages of Real-Time Operating Systems**

- Priority Based Scheduling.
- Abstracting Timing Information.
- Maintainability/Extensibility.
- Modularity.
- Promotes Team Development.
- Easier Testing.
- Code Reuse.
- Improved Efficiency.

## **2.7 Disadvantages of Real time operating system**

Here, are drawbacks/cons of using RTOS system:

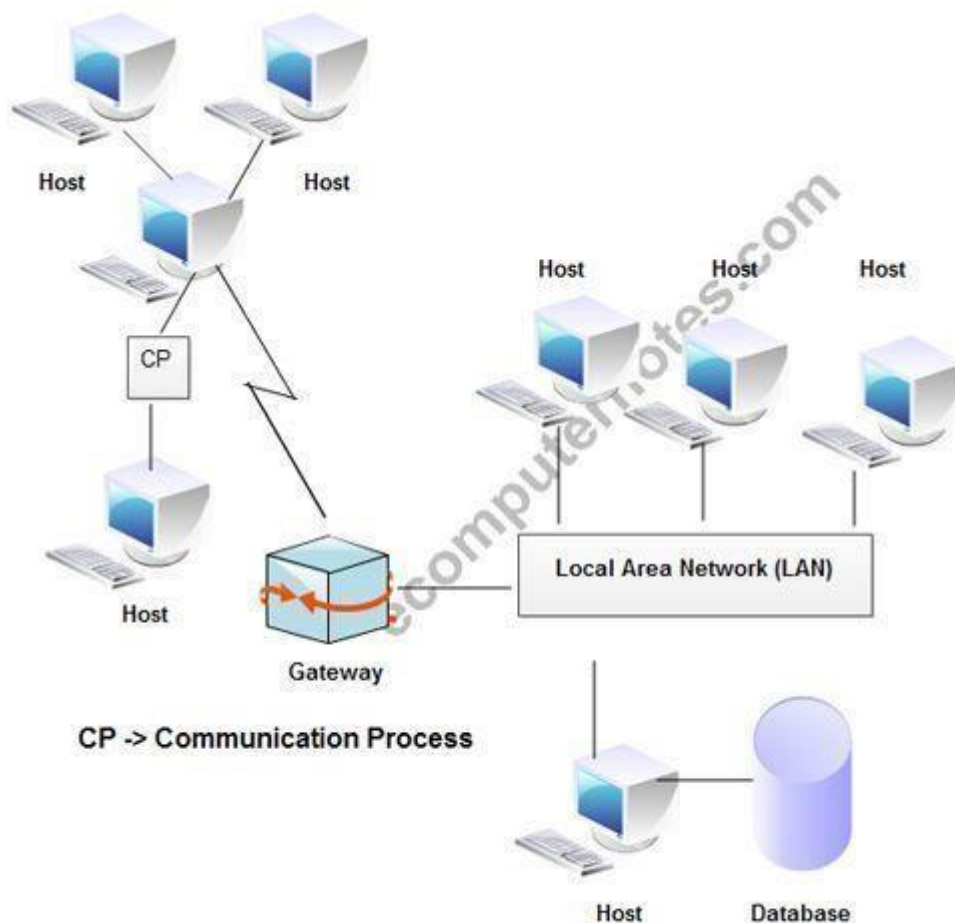
- RTOS system can run minimal tasks together, and it concentrates only on those applications which contain an error so that it can avoid them.
- RTOS is the system that concentrates on a few tasks. Therefore, it is really hard for these systems to do multi-tasking.
- Specific drivers are required for the RTOS so that it can offer fast response time to interrupt signals, which helps to maintain its speed.
- Plenty of resources are used by RTOS, which makes this system expensive.
- The tasks which have a low priority need to wait for a long time as the RTOS maintains the accuracy of the program, which are under execution.
- Minimum switching of tasks is done in Real time operating systems.
- It uses complex algorithms which is difficult to understand.
- RTOS uses lot of resources, which sometimes not suitable for the system.

### 3.0 Distributed Operating System

In this article, we will fully **explain distributed operating system**. Distributed operating system allows distributing of entire systems on the couples of center processors, and it serves on the multiple real time products as well as multiple users.

**Distributed Operating System** is a model where distributed applications are running on multiple computers linked by communications. A distributed operating system is an extension of the network operating system that supports higher levels of communication and integration of the machines on the network.

This system looks to its users like an ordinary centralized operating system but runs on multiple, independent central processing units (CPUs).



A Typical View of Distributed System

## **Types Of Distributed Operating System**

• **Client-Server Systems** • **Peer-to-Peer Systems** • **Middleware**

### **1.Client-Server Systems**

**Client-Server Systems** is known as “Tightly Coupled Operating System”. This system is designed mostly for multiprocessors and homogeneous multicomputer. Client-Server Systems works as a centralized server because it provides the approval to all requests, which are generated by client systems side.

Server systems can be divided into two segments: **i)**

#### **Computer Server System**

This system allows the interface, and then client sends own all requests for executing as action. Finally it sends to back response after executing action, and transfer result to client.

#### **ii) File Server System**

It allows the file system interface for clients because their clients can be performed various tasks such as creation, updating, deletion files, and more.

**Objective** – Hide and manage hardware resources.

### **2.Peer-to-Peer System**

**Peer-to-Peer System** is known as a “Loosely Couple System”. This concept is implemented in the computer network application because it contains the bunch of processors, and they are not shareable memories or clocks as well. Every processors consist own local memory, and these processors make communication with each other through various communication medium such as high speed buses or telephone lines.

**Objective** – It provides local services to remote clients.

### **3. Middleware**

**Middleware** allows the interoperability in the between of all applications, which are running on other operating systems. With using these services those applications are capable for transferring all data each other.

**Objective** – It allows the distribution transparency

## **Terms Used In Distributed Operating System**

### **1. Concurrency:-**

**Concurrency** is a big-picture word describing what happens when we break down activities into smaller tasks that run together

### **2.Tasks & Threads:-**

A **Task** is an abstraction for a small unit of work, when you break it down. Hammering a nail is a task. Watering plants and picking up take-out are also tasks. Ordering an airline ticket is a task, and so is an airline company updating its ticket prices. In computer terms a task is a **Thread**.

### **3.process:-**

A **Process** is at the operating-system level, independent and may include multiple threads running in the same memory space.

### **4.Synchronized:-**

Synchronized code can only be run / accessed by a single thread at a time, which means one program cannot access it while another is updating. Code synchronization is more like a queue, where threads wait their turn before going through the activity and each thread may execute the activity separately.

## **3.3 Features of distributed Operating System**

- Resource sharing.
- Openess.
- Concurrency.
- Scalability.
- Fault Tolerance.
- Transparency.

### 3.4 Application of distributed Operating System

- Air Traffic Control System
- Automated Banking System
- Multiplayer online gaming
- Airplane Control Towers System
- Scientific Computing System
- Cluster Computing
- Grid Computing
- Data rendering
- Wireless sensor N/W System
- Routing Algorithms

### 3.5 Advantages of distributed Operating System

- All the nodes in the **distributed system** are connected to each other. ...
- More nodes can easily be added to the **distributed system** i.e. it can be scaled as required.
- Failure of one node does not lead to the failure of the entire **distributed system**.

### 3.6 Disadvantages of Distributed Systems

- It is difficult to provide adequate security in **distributed systems** because the nodes as well as the connections need to be secured.
- Some messages and data can be lost in the network while moving from one node to another.

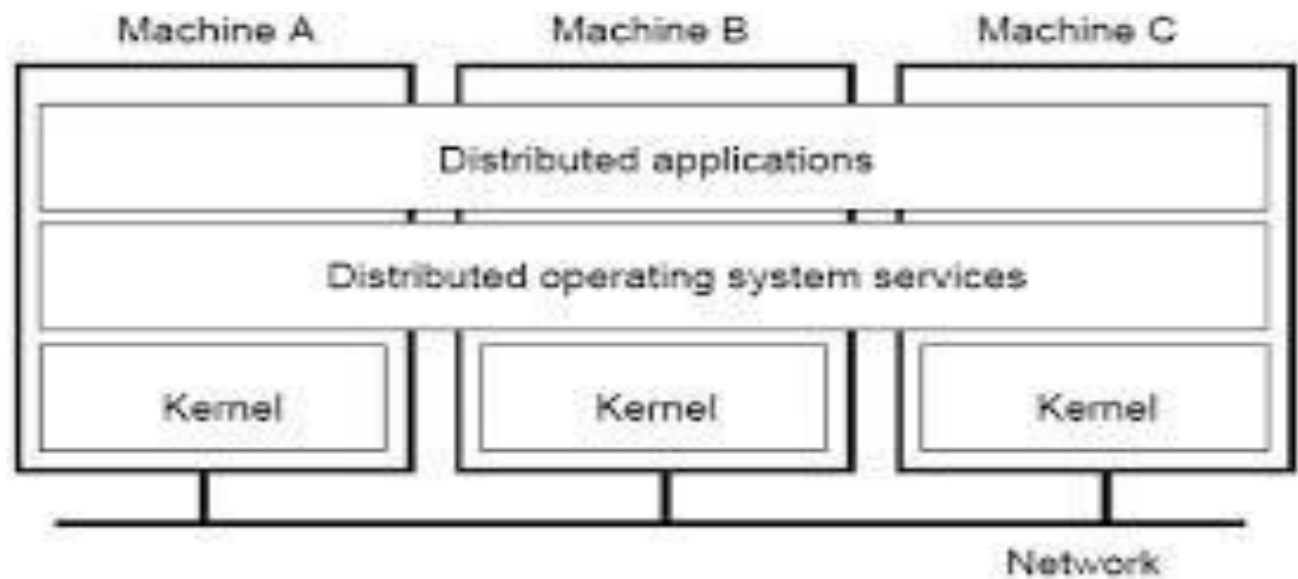


Fig. 1-14. General structure of a multicomputer operating system.



#### 4.0 Comparing features of Real time operating System and Distributed Operating System

Real time operating System	Distributed Operating System
1. <b>Distributed Operating System</b> is a model where distributed applications are running on multiple computers linked by communications.	1. The time interval required to process and respond to inputs is very small. This time interval is called <b>response time</b> .
2. Real time Operating System are used in large organizations.	2. Distributed operating System are most common in middle and large organization.
3. Distributed system a system in which components are distributed across multiple locations and computernetwork.	3. A <b>real-time system</b> is any information processing system which has to respond to externally generated input stimuli within a finite and specified period
4. These are referred to as <b>loosely coupled systems</b> or distributed systems.	4. <b>Real-time systems</b> are used when there are time requirements that are very strict like missile systems, air traffic control systems, robots, etc.
5. Types of Real time operating system:-hard real time, firm real time and soft real time	5. Types of Real time operating system:-Client sever system, peer to peer and middleware



**5.0 Conclusion:-** • Without operating system the computer cannot run the application and we cannot do work in the computer, therefore operating system is very important in computer.

- As operating system controls and coordinates the use of the hardware among the various uses, we can say that it plays a very important role in the computer system.
- An operating system is needed in order to use various application on your computer and wireless devices