UNIT-3

1) Middleware

Middleware is a more effective program that acts as bridge in between various Applications and other databases otherwise tools.

*Middleware Allows making better communication application services, Messaging, Authentication, API Management to exchange data

*Middleware is also known as plumbing because its both sides are connected with different Applications, and it helps to transfer data in both sides.

*Middleware helps to provide better environment for developer to design several applications with more efficiently.

Middleware Architecture working

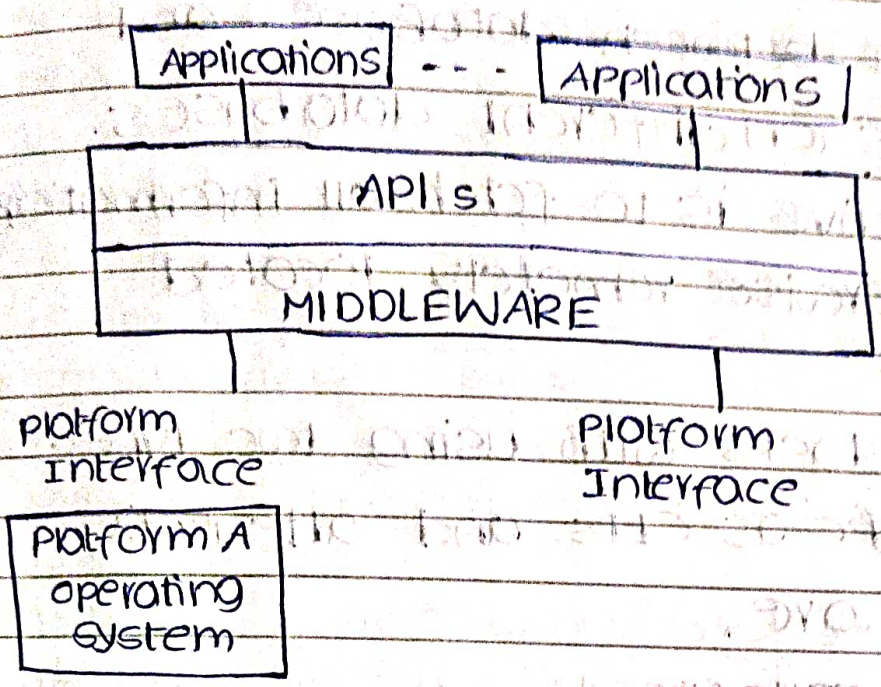
In this Architecture to make interaction with backend data from several network based requests, and this data may be into various form like as simple picture for showing, streaming video for playing motion, and it might be complex data such as million transactions of banking sensors

Expt. No. _____

Expt. Name _____ Page No. _____

Date : _____

Middleware Architecture.



This data that is requested can take on several format and it might be hold in different variant of Methods, like as file server ~~path~~ push data and received from message queue, otherwise, stored into database

* Middleware programs help to deliver Messaging service for different applications to send all types of data, like as simple object access protocol (SOAP).
* Main objective of Middleware's architecture allows accessing all back end resources with easier way.

Different types of Middlewares are

Database Middleware :-

* The database Middleware helps to make communication in between database and other Applications (or) different databases.
* Its main objective is to fetch all information from local otherwise remotely located databases.
* It performs all tasks with using two types of databases like as CLIs and alternative database Middleware.

Remote procedure calls (RPC)

* RPC is traditional Middleware that was introduced in 1970's because it works as client/server principle.
* RPC Middleware works as point to point communication so it is not scalable and it consumes huge amount of resources while its processing
* It contains more issues such as degrade performance and high complexity.

Expt. No. _____ Page No. _____

Expt. Name _____ Date : _____

Object Middleware :

* It is also known as "Object Request Broker(ORB)" and its main goal is to handle all communication in between all objects in distributed computing system.
* It helps to make program from one computer to another computer system via computer network.

Message oriented Middleware:

* This Middleware helps to send and receive all messages over different types of distributed applications.
* It allows less complication for using of all applications move on several types of platforms, and it is also more comfortable to work across all types of operating system.

Applications of Middleware Are:

1. Enterprise Middleware
2. platform Middleware

Real-time Distributed system:-

A real time system is any information processing system which has to respond to externally generated input stimuli within a finite and specified period.

* The correctness depends not only on the logical result but also the time it was delivered.

Distributed system

Distributed system is a system in which components are distributed across multiple locations and computer network.

* Today's Information systems are no longer computer-based system. Instead, they are built on some combination of networks to form Distributed systems.
* A Distributed system is one in which the components of an Information system are distributed to multiple locations in a computer Network.
* The opposite of Distributed systems are centralized systems. In centralized systems, a central Multi-user computer hosts all components of an information system.

Expt. No. _____ Page No. _____

Expt. Name _____ Date : _____

* Distributed systems are inherently more complicated and more difficult to implement than centralized solutions.
* Modern businesses are already distributed and thus they need distributed system solutions
* Distributed computing moves information and services closer to the customers that need them.
* In general, distributed system solutions are more user friendly because they use the PC as the user interface processor.
* personal computers and network servers are much less expensive than main-frames, the total cost of ownership is atleast as expensive
* There is a price to be paid for distributed systems. Network data traffic can cause congestion that actually slows performance. Data security and integrity can also be more easily compromised in a distributed solution.

Signal processing :-

* signal processing is a field of Engineering, Mathematics and computer science that deals with processing, analyzing and manipulating Analog and digital signals.
* signals can be Audio, video, sensor data, Images and many other types of data.
* signal processing techniques are used in wide range of applications, including telecommunications, audio and video processing, Image processing, speech recognition and control systems.
* some common signal processing tasks include filtering, noise reduction, compression and feature extraction.

Applications

1. Telecommunications :-

signal processing techniques are used in telecommunications to transmit, receive and process signals over communication channels this includes tasks such as Modulation, demodulation, error correction and signal Amplification.

Expt. No. _____ Page No. _____

Expt. Name _____ Date : _____

2. Audio and video processing:
They are used to enhance the quality and clarity of Audio and video signals as well as to extract features such as speech, Music and Moving objects.

3. Image processing:-
These are used to improve the quality and resolution of Images as well as to extract features such as edges, shapes and textures.

4. Speech Recognition:-
These are used to analyze and Interpret speech signals enabling the development of systems that can transcribe speech or recognize spoken commands.

5. Control Systems:-
These are used in control systems to stabilize and optimize the performance of systems by processing feedback signals from sensors and Actuators.

6. Biomedical Engineering
These are used in biomedical Engineering to Analyze and interpret signals from medical devices, such as electrocardiograms (ECG) and Magnetic resonance imaging (MRI) scanners.

7. Financial Engineering:
These are used in financial Engineering to Analyze and Interpret financial data to develop predictive models for financial markets.

The process of signal processing:

1. Signal acquisition :-

The first step in signal processing is to acquire the input signal. This may involve using sensors to measure physical quantities such as temperature, pressure, or acceleration or it may involve capturing Audio, video or other types of data.

2. Signal conversion :-

If the Input signal is in Analog format, it may need to converted into digital format using an Analog-to-digital converter (ADC).

3. Signal Representation :-

The Input signal is then represented in a suitable form for processing. This may involve representing the signal as a direct sequence

4. Signal Manipulation :-

The input signal is then manipulated using various techniques such as filtering, noise reduction, compression and feature extraction.

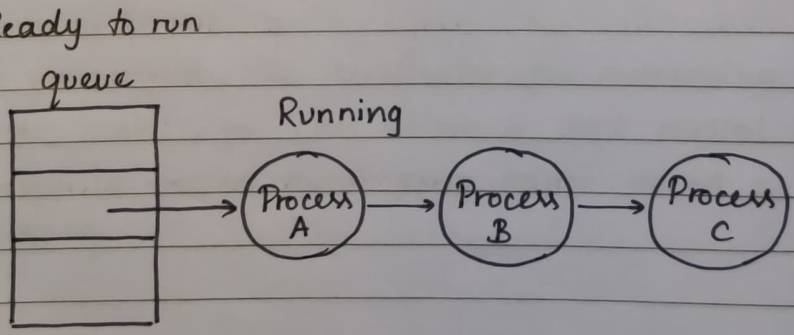
Unit - 3

Q1. Task scheduling algorithms →

- The process of deciding which task will utilize CPU time is called task scheduling.
- The scheduling of tasks may be on the basis of their priorities.
- The priority assigning mechanism for the tasks may be static or dynamic.
- In case of static priority assignment, the priority of the task is assigned as soon as the task is created, and cannot be changed thereafter.
- In dynamic assigning, priorities of the tasks can be changed during runtime.

1. First in first out (FIFO) →

- The task that came in first in the ready to run state will go out first into the running state.

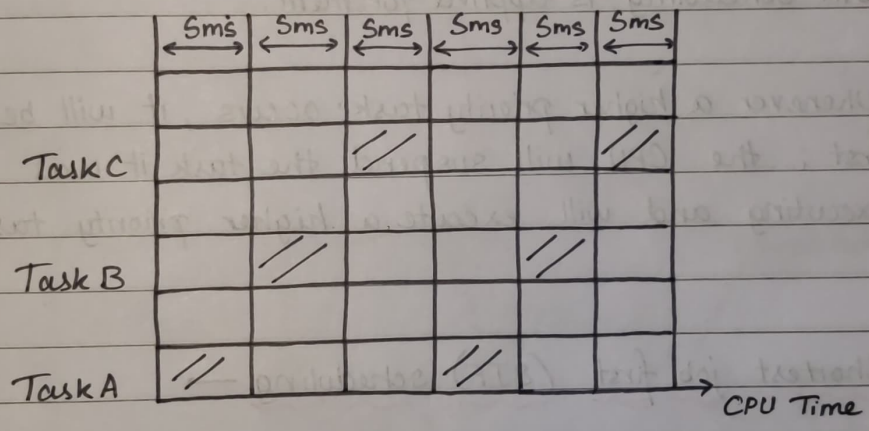


Advantage - It is very easy to implement

Disadvantage - We cannot have any priority mechanisms. In real time examples, each task has a different priority and it is to be implemented

2. Round Robin Scheduling →

- In this case, each task gets their turn after all the tasks have been allotted their time slots.
- The time for each time slot is the same (time slicing) and is given to each task one by one.
- Round Robin scheduling for 3 tasks with time slots of 5 ms is shown below



- The switching of tasks occurs in the following case →
  → current task has completed it's work before the completion of it's time slot.
  → current task has no work to be done
  → current task has completed the time ~~slot~~ slice allocated to it.

Advantage - very easy to implement.

Disadvantage - all the tasks are considered at the same level.

3. Round Robin scheduling with priority →

- It is the modified version of round robin scheduling.
- In this case, the tasks are given priorities based on their significance and deadlines.
  Thus tasks with higher priority can interrupt the processor and utilize CPU time.
- If multiple tasks have the same priority, round robin scheduling is applied for them.
- Whenever a higher priority task occurs, it will be executed first; the CPU will suspend the task it was executing and will execute a higher priority task.

4. Shortest job first (SJF) scheduling →

- The task with the shortest execution time is executed first. The priority is higher for a task with lesser execution time.
- This ensures lesser number of tasks in ready state.

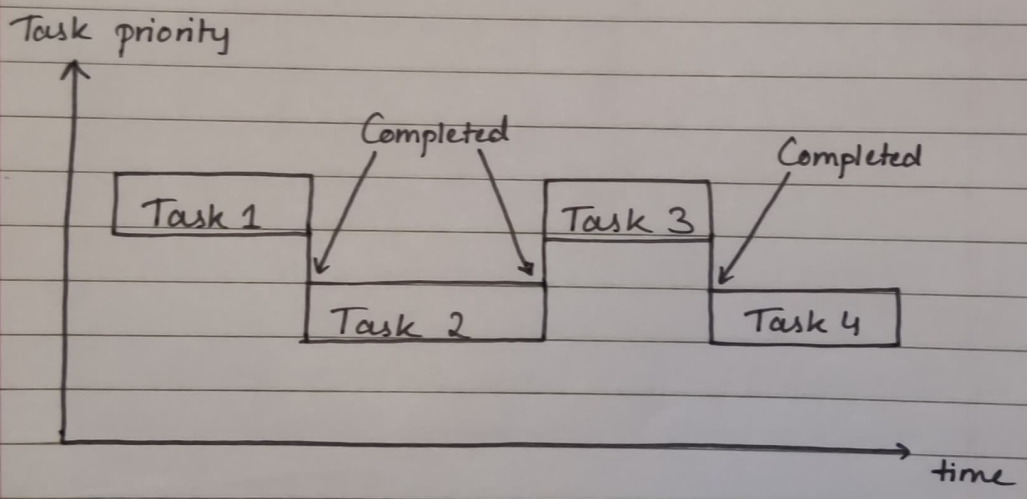
Advantage- implementation is simpler as only the execution time needs to be compared.

Disadvantage- If there are too many short execution time tasks, the task with more execution time may never be executed.

5. Non - preemptive scheduling →

- This scheduling mechanism can be implemented in any of the previously mentioned mechanisms involving the concept of priority.

- As the name says, if a task (task 1) is in running state and another task (task 2) with higher priority enters into the ready to run state, the earlier task ie. task 1 continuous it's execution until it's time slice and the higher priority task, ie. task 2 has to wait for its turn.



6. Preemptive scheduling →

- This ~~test~~ scheduling can be implemented on any of the mechanisms having the concept of priority.

- As the name says, if task 1 is in running state and another task (task 2) with higher priority enters the ready to run state, then task 1 has to release the CPU and task 2 will get executed.

- Thus the higher priority task will get the CPU as soon as it enters ready to run state, and it will now enter running state