International Journal of Scientific Research and Engineering Development-– Volume 8 Issue 5, Sep-Oct 2025
Available at www.ijsred.com

RESEARCH ARTICLE OPEN ACCESS

Review on Role of Multithreading and Thread in Operating System

Prof.Nisha Pawar*, Syed Zain Ali**, Devarsh Murkute***, Shaikh Basit****
*(Lecturer at P.E.S Polytechnic Aurangabad, Maharashtra, India
Email: pawarnisha1797@gmail.com)
**(Computer Engineering, P.E.S Polytechnic Aurangabad
Email: syedzain221ali@gmail.com)
***(Computer Engineering, P.E.S Polytechnic Aurangabad
Email: devarshmurkute82@gmail.com)
****(Computer Engineering, P.E.S Polytechnic Aurangabad
Email: shaikhbasit958@gmail.com)

----------**********----------

Abstract:

Threads are an important component of operating systems that help in smooth operation, resource sharing, and concurrent task handling. Unlike processes with separate memory spaces, threads allow efficient execution within the same program. This review paper explains thread types, implementations, benefits, and challenges.

Keywords — Central Processing Unit (CPU), Protocol Data Unit (PDU), Light Weight Process (LWP).

----------**********----------

I. INTRODUCTION

Thread is a sequential flow of task within a process. Thread are also known as the smaller part of the process. There is a possibility that a process is allowed to have more than one thread. In the sub concept of operating system which is thread it has two types first one single thread and multi thread.

In the single level thread only one register and stack will be given.

In multi thread register and stack are separated for each other. Multi thread are useful in multiprocesser system where threads runs parallel to each other.

Every thread has its own register and stack. A thread sometime can known as light weight process (LWP). A word processer may have thread for showing graphics, another thread for reading information from the user and the third thread used for checking spelling and grammer in the back of software.

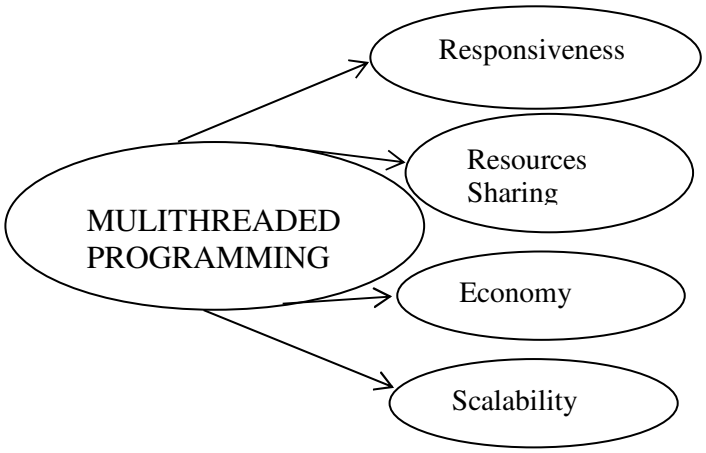
II. BACKGROUND OF STUDY

In an environment where both clients and servers communicate, clients send their requests to the server. The server stores these requests in a queue so that they can be processed step by step. This entire process is managed by separate processes within the server's operating system.

In the early stages of system development, servers operated in cycles or phases. The setup at that time was such that the computer would wait until a request arrived. As soon as a request arrived, the server would begin work. It would first complete all the tasks of the request, and only then handle the next request.

When the server was running, it would first fetch data from the hard drive, then process that data, create a response in PDU format, and finally send it to the user. This entire process would then be repeated repeatedly for the next incoming request.

 ©IJSRED: All Rights are Reserved



III. MULTITHREADING CONCEPT

Multithreading allows us to run programs without needing separate copies in memory. To save memory, instead of writing codes again and again and data, we can have just one copy of the code, one copy of the data, and one copy of the address space. Each product within an operation may access these shared resources. If something can't be shared, then it will be kept from others.

Threads are called lightweight because they don't have their own separate code or data; rather, they depend on the main process for every product. The main benefits of multithreading are: Programs are able to react quick to users. Resource swapping - Threads use the same memory and resources, prevent making duplicates.

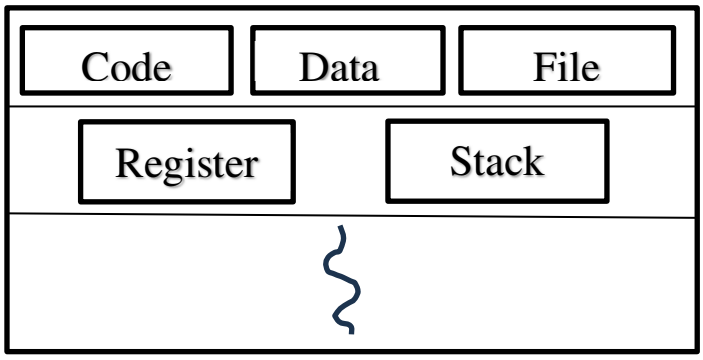
Creating a thread is cheaper as compared to others (only few materials are in need) compared to start making a new process. Flexible programs can use multi-core CPUs by running multiple threads simultaneously. Thread control block is smaller as compared to process control block, so there's less load when switching between multiple threads compared to the processes.

A. Lifecycle of thread

A process moves through the states of creating the to the termination of the process. These states are nothing but New, Ready, Running, Blocked or Waiting and the last state of lifecycle is Termination. In the first state which is New state is when the process is being created and its not yet started. And in the second state the process Ready and waits in the ready queue to get started execution. When the CPU start executing the processes it is now in the Running state.The process waits for some events like input or output of the data then it goes in the Waiting or Blocked state. After execution of all tasks the process finally goes in the termination state once it enters this state it cannot be run again for execution

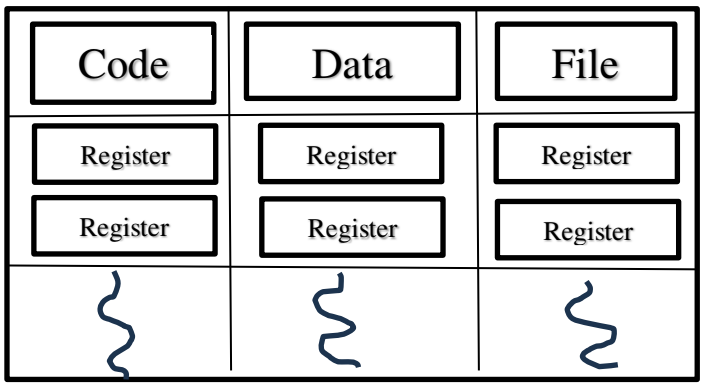
B. Single thread

In the concept of single thread in operating system there will be only one register and stack will be given for the process. This can perform only one task at a time. Process executes one by one.



C. Multi thread

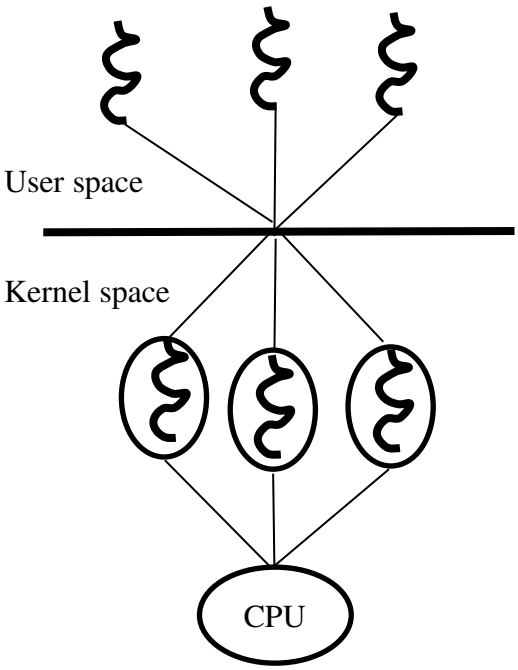
In Multi threading each process has their own register and stack. This can perform multiple tasks at the same time parallel to each other. Multi threading is more complex as compared to single thread.



 ©IJSRED: All Rights are Reserved

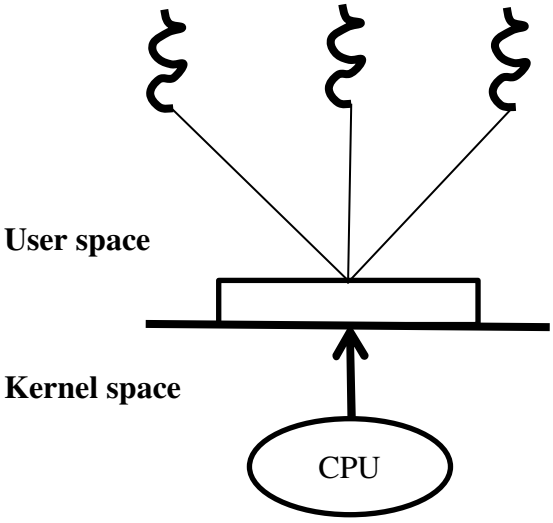
D. Kernal level thread

This level thread are implemented at kernel level. These threads are straight supported by the operating system.



E. User level thread

This thread is implemented at user level known as user level thread. In this thread management is done by the application. The kernel is not aware of existence of thread.



CONCLUSIONS

Threads and multi-threading are crucial parts of operating system which is used to increased performance and efficiency of the system. Threads are lightweight and small process which can be perform one or more task simultaneously. Due to which efficient use of resources, its scalibility and ability to give response increases. The system of single thread are simple but their performance is slow and limited and multithread system are difficult in structure but they are fast and efficient in execution time.

In this review paper we get to know the concept of multi-threading which is very useful in computing environment which helps in increased computation level which needs multitasking which will helps in future for advance architecture and security system.

ACKNOWLEDGMENT

We are very thankful to the Department of Computer Engineering P.E.S College of Polytechnic Aurangabad, Maharashtra. They supported us and gave us guidance regarding the resources that helped us to succesfully complete our review paper. Under the guidance of Prof. Nisha Pawar.

REFERENCES

[1] Krishna Kavi The University of Texas at Arlington 1998. https://www.researchgate.net/publication/2734966_Multithreading_Implementations

[2] Role of Threads in Operating System Saurabh Sharma*, Sidharth Sharma, Lotus Student Department of School of Computer Science and Engineering 2024. https://www.researchtrend.net/ijeece/pdf/Role-of-Threads-in-Operating-System-Saurabh-Sharma-20.pdf

[3] Silberschatz, A., Galvin, P. B., & Gagne, G. (2018). Operating System Concepts (10th ed.). Wiley. https://archive.org/details/operatingsystemconcepts10th/page/n265/mode/2up

 ©IJSRED: All Rights are Reserved

[4] Tanenbaum, A. S., & Bos, H. (2014). Modern Operating Systems (4th ed.). Pearson. https://archive.org/details/modernoperatings0000tane

[5] Role-of-Threads-in-Operating-System-Saurabh-Sharma https://www.researchtrend.net/ijeece/pdf/

[6] Thread in Operating System 2025 https://www.geeksforgeeks.org/operating-systems/thread-in-operating-system/

 ©IJSRED: All Rights are Reserved