Process vs Thread: What’s the difference?

By Lawrence Williams Updated March 5, 2022

What is a Process?

A process is the execution of a program that allows you to perform the appropriate actions specified in a program. It can be defined as an execution unit where a program runs. The OS helps you to create, schedule, and terminates the processes which is used by CPU. The other processes created by the main process are called child process.

A process operations can be easily controlled with the help of PCB(Process Control Block). You can consider it as the brain of the process, which contains all the crucial information related to processing like process id, priority, state, and contents CPU register, etc.

In this Process vs Thread tutorial, you will learn:

What is Thread?

Thread is an execution unit that is part of a process. A process can have multiple threads, all executing at the same time. It is a unit of execution in concurrent programming. A thread is lightweight and can be managed independently by a scheduler. It helps you to improve the application performance using parallelism.

Multiple threads share information like data, code, files, etc. We can implement threads in three different ways:

1. Kernel-level threads
2. User-level threads
3. Hybrid threads

KEY DIFFERENCE

- Process means a program is in execution, whereas thread means a segment of a process.
- A Process is not Lightweight, whereas Threads are Lightweight.
- A Process takes more time to terminate, and the thread takes less time to terminate.
- Process takes more time for creation, whereas Thread takes less time for creation.
- Process likely takes more time for context switching whereas as Threads takes less time for context switching.
- A Process is mostly isolated, whereas Threads share memory.
- Process does not share data, and Threads share data with each other.

Properties of Process

Here are the important properties of the process:

- Creation of each process requires separate system calls for each process.
- It is an isolated execution entity and does not share data and information.
- Processes use the IPC(Inter-Process Communication) mechanism for communication that significantly increases the number of system calls.
- Process management takes more system calls.
- A process has its stack, heap memory with memory, and data map.

Properties of Thread

Here are important properties of Thread:

- Single system call can create more than one thread
- Threads share data and information.
- Threads shares instruction, global, and heap regions. However, it has its register and stack.
- Thread management consumes very few, or no system calls because of communication between threads that can be achieved using shared memory.

Difference between Process and Thread

Here, are the important differences between Process and Thread

Parameter	Process	Thread
Definition	Process means a program is in execution.	Thread means a segment of a process.
Lightweight	The process is not Lightweight.	Threads are Lightweight.
Termination time	The process takes more time to terminate.	The thread takes less time to terminate.
Creation time	It takes more time for creation.	It takes less time for creation.
Communication	Communication between processes needs more time compared to thread.	Communication between threads requires less time compared to processes.
Context switching time	It takes more time for context switching.	It takes less time for context switching.
Resource	Process consume more resources.	Thread consume fewer resources.
Treatment by OS	Different process are tread separately by OS.	All the level peer threads are treated as a single task by OS.

Parameter	Process	Thread
OS	separately by OS.	treated as a single task by OS.
Memory	The process is mostly isolated.	Threads share memory.
Sharing	It does not share data	Threads share data with each other.

What is Multithreading?

Multithreading refers to multiple threads of execution within an operating system. In simple terms, two or more threads of a same process are executing simultaneously.

You Might Like:

- Mutex vs Semaphore: What's the Difference?
- Real-time operating system (RTOS): Components, Types, Examples
- Round Robin Scheduling Algorithm with Example
- Process Synchronization: Critical Section Problem in OS
- 20 Best Zip File Software | Unzip Program | File Compression

← Prev Report a Bug Next →

About

About Us
Advertise with Us
Write For Us
Contact Us

Career Suggestion

SAP Career Suggestion Tool
Software Testing as a Career

Interesting

eBook

Blog

Quiz

SAP eBook

Execute online

Execute Java Online

Execute Javascript

Execute HTML

Execute Python