A thread is a single sequential flow of control within a program.

A thread is the smallest unit of processing that can be performed in an OS.

A thread exists within a process. A single process may contain multiple threads.

Types of threads

1. User threads
2. Kernel threads

Working of threads

Each thread in the process shares the same memory and resources. Because threads share the same address space as the process and other threads within the process, the operational cost of communication between the threads is low, which is an advantage.

Threads are not independent of one another like processes are, and as a result threads share with other threads their code section, data section, and OS resources (like open files and signals).

But, like process, a thread has its own program counter (PC), register set, and stack space.

Benefits of Threads in OS

1. Threads minimize the context switching time.
2. Use of threads provides concurrency within a process.
3. Efficient communication.

4. It is more economical to create and context switch threads.
5. Threads allow utilization of multiprocessor architectures to a greater scale and efficiency.