



Maria Eugenia Herrera Sanchez

Montserrat Olan López

Ingeniería en Sistemas Computacionales y Diseño de Software, Instituto Universitario de Yucatán

(46220517343724M): Sistemas operativos

Perla Alejandra Landero Heredia

(Fecha con: 20 de julio 2025)

1. ¿Qué es un proceso?

Un **proceso** es una instancia en ejecución de un programa. Cuando ejecutas una aplicación (por ejemplo, un navegador o un procesador de texto), el sistema operativo crea un proceso para esa aplicación.

Características de un proceso:

- **Aislamiento:** Cada proceso tiene su propio espacio de direcciones en memoria. Esto significa que un proceso no puede acceder directamente a la memoria de otro proceso.
- **Recursos asociados:** Cada proceso tiene recursos asignados por el sistema operativo, como memoria, archivos abiertos, identificador de proceso (PID), estado de ejecución, etc.
- **Pesado:** Cambiar de un proceso a otro (conmutación de contexto) puede ser costoso en términos de rendimiento porque implica guardar y restaurar el contexto completo del CPU.

Componentes de un proceso:

- Código del programa
 - Contador de programa (PC)
 - Pila (Stack)
 - Datos del programa (segmento de datos)
 - Sección de heap (memoria dinámica)
 - Registros de CPU
 - Información del sistema operativo (estado, prioridad, recursos, etc.)
-

2. ¿Qué es un hilo? (Thread)

Un **hilo** es la unidad básica de ejecución dentro de un proceso. Un proceso puede tener uno o varios hilos, que comparten el mismo espacio de direcciones y recursos del proceso que los contiene.

Características de los hilos:

- **Compartición de memoria:** Los hilos de un mismo proceso comparten memoria, archivos abiertos y otros recursos.
- **Livianos:** Son más livianos que los procesos. Cambiar de un hilo a otro dentro del mismo proceso es más rápido que cambiar entre procesos.
- **Concurrentes:** Los hilos pueden ejecutarse en paralelo si el sistema tiene múltiples núcleos de CPU.

Tipos de hilos:

- **Hilos del usuario:** Gestionados por una biblioteca a nivel de usuario, sin intervención del núcleo del sistema operativo.
- **Hilos del kernel:** Gestionados directamente por el núcleo del sistema operativo.

3. Diferencias clave entre procesos e hilos:

Aspecto	Proceso	Hilo
Espacio memoria	de Separado para cada proceso	Compartido dentro del proceso
Costo de creación	Alto	Bajo
Costo de conmutación	de Alto	Bajo
Comunicación entre ellos	IPC (Inter-Process Communication)	Compartiendo memoria
Independencia	Totalmente independientes	Parcialmente (comparten recursos)
Fallo	Un proceso no afecta a otro	Un hilo puede afectar a los demás del mismo proceso

4. Ventajas y desventajas

Procesos

Ventajas:

- Aislamiento entre procesos proporciona seguridad.
- Si un proceso falla, no afecta a otros.

Desventajas:

- Comunicación entre procesos más complicada.
- Más uso de recursos.

Hilos

Ventajas:

- Más eficientes en uso de CPU y memoria.
- Compartir recursos facilita la comunicación entre tareas.

Desventajas:

- Errores en un hilo pueden afectar a todo el proceso.
 - Requiere sincronización cuidadosa (por ejemplo, uso de **mutex**, **semaphores**, etc.).
-

5. Uso de procesos e hilos en la práctica

- **Procesos:** Se usan cuando se requiere fuerte aislamiento, por ejemplo, navegadores que crean un proceso por pestaña para evitar que un error en una página afecte a todas.
 - **Hilos:** Se usan para tareas paralelas que deben compartir datos, como un servidor web que atiende múltiples solicitudes simultáneamente.
-

6. Ejemplo ilustrativo (en pseudocódigo)

Un proceso con múltiples hilos:

Proceso: ServidorWeb

Hilo 1: Aceptar nuevas conexiones

Hilo 2: Procesar petición del cliente

Hilo 3: Enviar respuesta al cliente

Todos los hilos comparten la base de datos de usuarios, archivos de registro, configuración, etc.

7. Sincronización de hilos

Como los hilos comparten memoria, se necesita coordinación para evitar condiciones de carrera (**race conditions**), errores de concurrencia y bloqueos mutuos (**deadlocks**).

Mecanismos comunes:

- **Mutex (Mutual Exclusion):** Solo un hilo puede acceder a un recurso a la vez.
 - **Semáforos:** Contadores usados para gestionar acceso a recursos múltiples.
 - **Monitores:** Estructuras que encapsulan datos con mecanismos de sincronización.
-

8. Multithreading y Multiprocessing

- **Multithreading:** Varios hilos dentro de un mismo proceso ejecutándose de forma concurrente. Útil para dividir tareas en subprocesos (hilos) que pueden compartir datos fácilmente.
 - **Multiprocessing:** Varios procesos ejecutándose en paralelo, cada uno con su propio espacio de memoria. Más seguro pero más pesado.
-

9. Modelos de hilo

- **Modelo uno a uno (1:1):** Cada hilo de usuario se asigna a un hilo del kernel.
- **Modelo muchos a uno (N:1):** Todos los hilos de usuario se ejecutan en un solo hilo del kernel.
- **Modelo muchos a muchos (N:M):** Los hilos de usuario se asignan dinámicamente a hilos del kern

Conclusión

Los procesos y los hilos son componentes esenciales en el funcionamiento de los sistemas operativos y el diseño de software moderno. Los **procesos** permiten ejecutar programas de forma aislada, con su propia memoria y recursos, lo que garantiza seguridad y estabilidad. Son ideales cuando se necesita separar tareas críticas, aunque su creación y gestión consume más recursos.

Por otro lado, los **hilos** ofrecen una forma más ligera y eficiente de ejecutar múltiples tareas dentro de un mismo proceso, compartiendo memoria y otros recursos. Esto permite una mayor velocidad, mejor uso del CPU y la posibilidad de realizar múltiples tareas en paralelo, aunque requiere un manejo cuidadoso de la sincronización para evitar errores.

En conjunto, ambos conceptos permiten construir aplicaciones **más rápidas, escalables y confiables**, siempre que se utilicen correctamente según las necesidades del sistema. Comprender cómo y cuándo usar procesos o hilos es clave para desarrollar software moderno, eficiente y seguro.