

Thread

Thread

- Proses dengan thread tunggal – proses menjalankan satu tugas pada satu waktu
- Sistem operasi modern – multithread – proses menjalankan banyak tugas/thread pada satu waktu
- Contoh : secara bersamaan mengetik dan menjalankan pemeriksaan ejaan di dalam proses yang sama

- Multithread pada satu prosesor – dijalankan bergantian dengan waktu yang sangat cepat sehingga tampak bersamaan
- Multithread pada multi prosesor – benar-benar dijalankan secara bersamaan
- Hyper threading (HT) – teknologi simultan multithreading oleh Intel
 - Menjalankan banyak thread secara bersama-sama
 - Pada tiap inti prosesor, sistem operasi mengamati 2 virtual prosesor
 - OS harus mendukung pemrosesan multiprosesor

Thread

- Suatu *thread* (atau *lightweight process* disingkat LWP) berisi :
 - Thread ID
 - Program counter
 - Register set
 - Stack space
- Suatu *thread* bersama thread-thread lainnya bisa berbagi :
 - Code section
 - Data section
 - Operating system resources

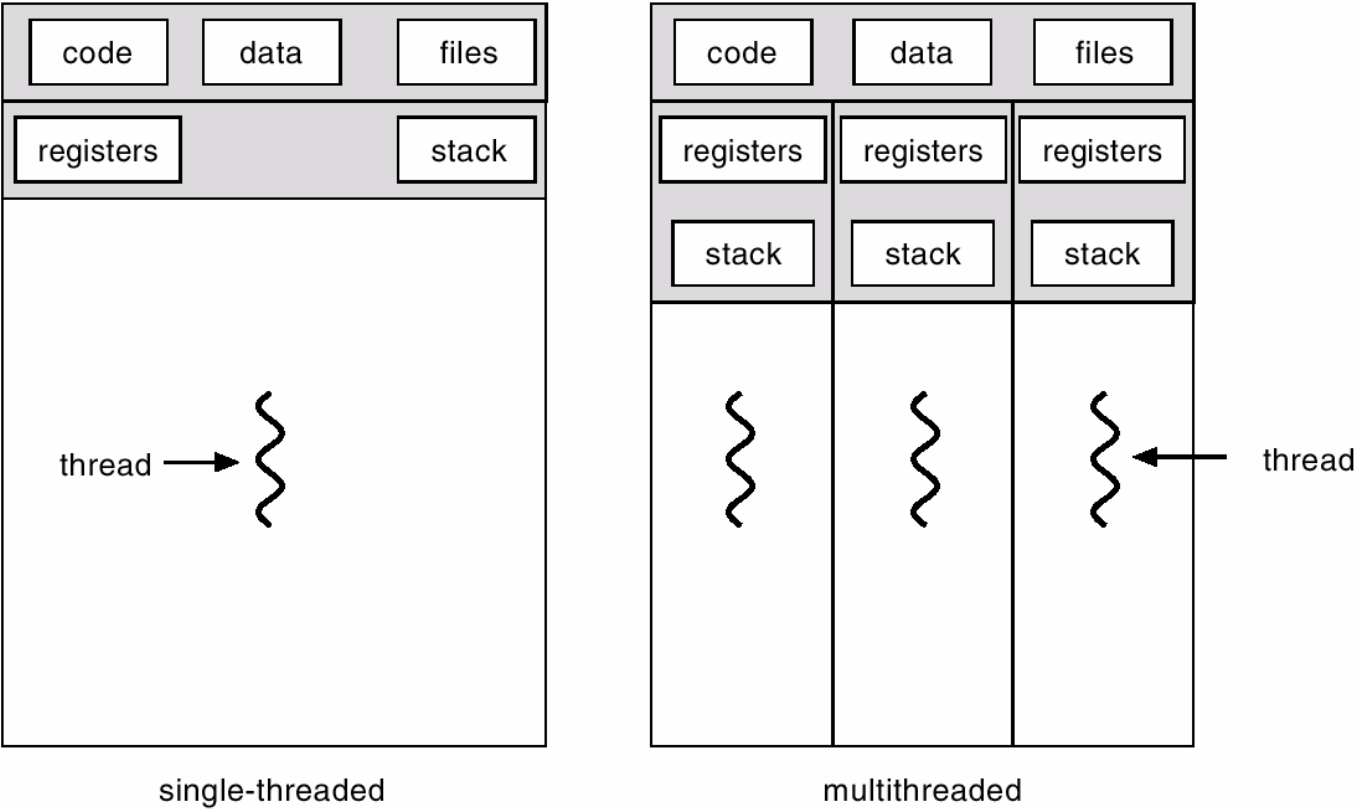
Perbedaan Thread Dengan Proses

- Thread adalah bagian dari suatu proses
- Tiap proses mempunyai informasi status dan sumber daya sendiri, thread berbagi informasi status dan sumber daya dengan thread yang lain dalam satu proses
- Tiap proses mempunyai alamat yang berbeda, sedangkan thread berbagi alamat yang sama
- Pergantian antar threads lebih cepat daripada antar proses

Contoh Multithreading

- Web browser : memiliki satu thread untuk display image, dan thread yang lain untuk mengambil data dari network.
- Web server : menerima request dari client untuk halaman web, image, sound, dll. Web server melayani client secara bersamaan (*concurrent*).

Single & Multithreaded Processes



Keuntungan Multithreading

- Responsif
 - Multithreading memungkinkan suatu program running walaupun sebagian program di-block atau menjalankan operasi yang lain.
- Resource Sharing
 - Thread dalam suatu proses yang sama dapat men-share memori dan resource dengan thread yang lain. Mengijinkan sebuah aplikasi untuk memiliki beberapa thread yang berbeda dalam lokasi memori yang sama
- Ekonomis
 - Alokasi memori dan resource untuk pembuatan proses memiliki *cost* yang mahal. Dalam hal context-switch, lebih ekonomis bila dilakukan dengan thread context-switch.
- Utilisasi CPU
 - Meningkatkan kinerja pada arsitektur multiprocessor, dimana thread running secara paralel pada prosesor yang berbeda.

User Threads

- Implementasi thread di level pengguna (dengan menggunakan pustaka/library thread).
- Pustaka menyediakan fasilitas untuk pembuatan thread, penjadwalan thread, dan manajemen thread tanpa dukungan dari kernel.
- Keuntungan : Dapat lebih cepat dibuat dan dikendalikan karena tanpa campur tangan kernel
- Kelemahan : salah 1 thread pengguna menjalankan blocking system call maka mengakibatkan seluruh proses diblok

Kernel Threads

- Didukung langsung oleh sistem operasi
- Pembuatan, penjadwalan, dan manajemen thread dilakukan oleh kernel pada kernel space
- Keuntungan :
 - jika thread menjalankan blocking system call kernel dapat menjalankan thread lain di aplikasi untuk melakukan eksekusi.
 - Pada lingkungan multiprosesor, kernel dapat menjadwalkan thread-thread pada prosesor yang berbeda
- Kelemahan : lebih lambat daripada user kernel

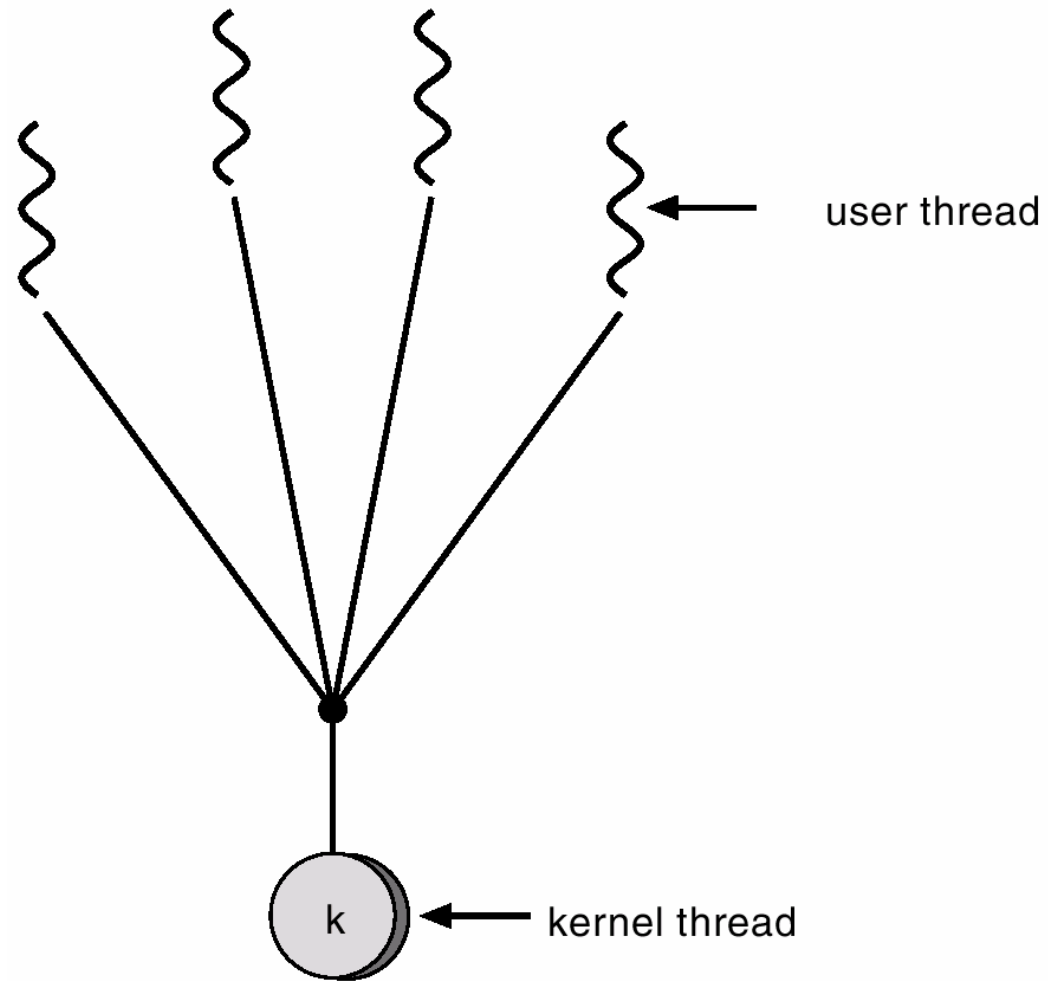
Model-model Multithreading

- Many-to-One
- One-to-One
- Many-to-Many

Many-to-One (User level Threading)

- Memetakan beberapa user thread ke satu kernel thread
- Hanya 1 thread pengguna yang dapat mengakses thread kernel pada satu saat \Rightarrow Multi thread tidak dapat berjalan secara paralel pada multiprosesor

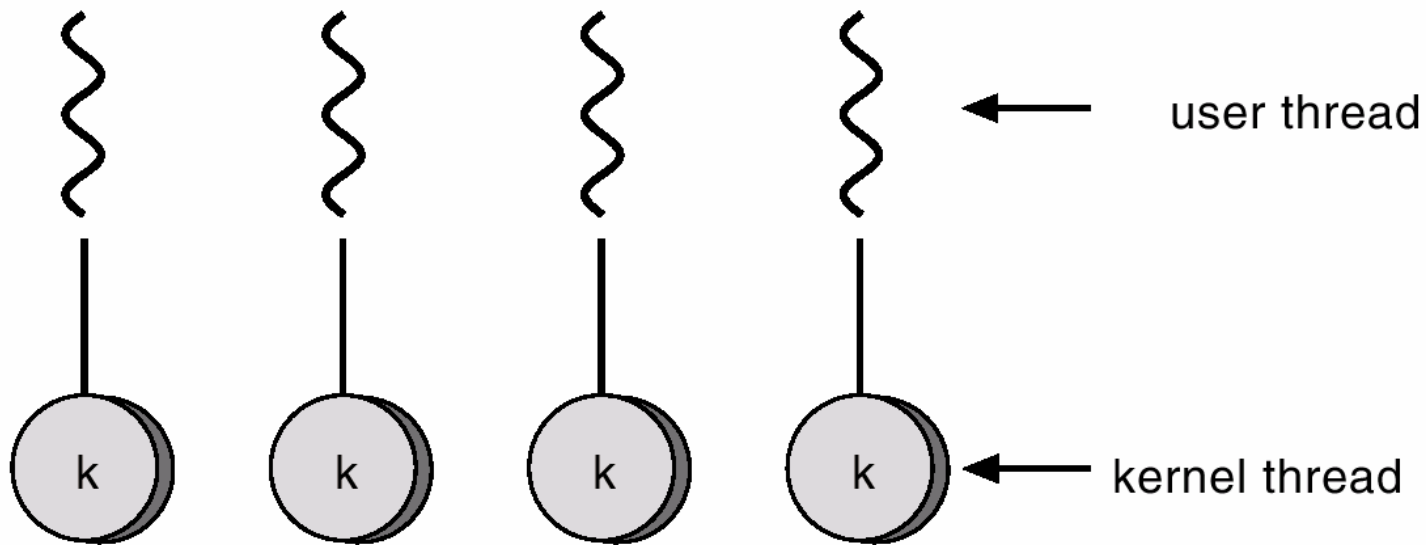
Many-to-One Model



One-to-One

- Masing-masing user-level thread dipetakan ke kernel thread
- Keuntungan = keuntungan thread kernel
- Kelemahan = setiap pembuatan user thread memerlukan jg pembuatan kernel thread sehingga dapat menurunkan kinerja aplikasi \Rightarrow jumlah thread dibatasi oleh sistem

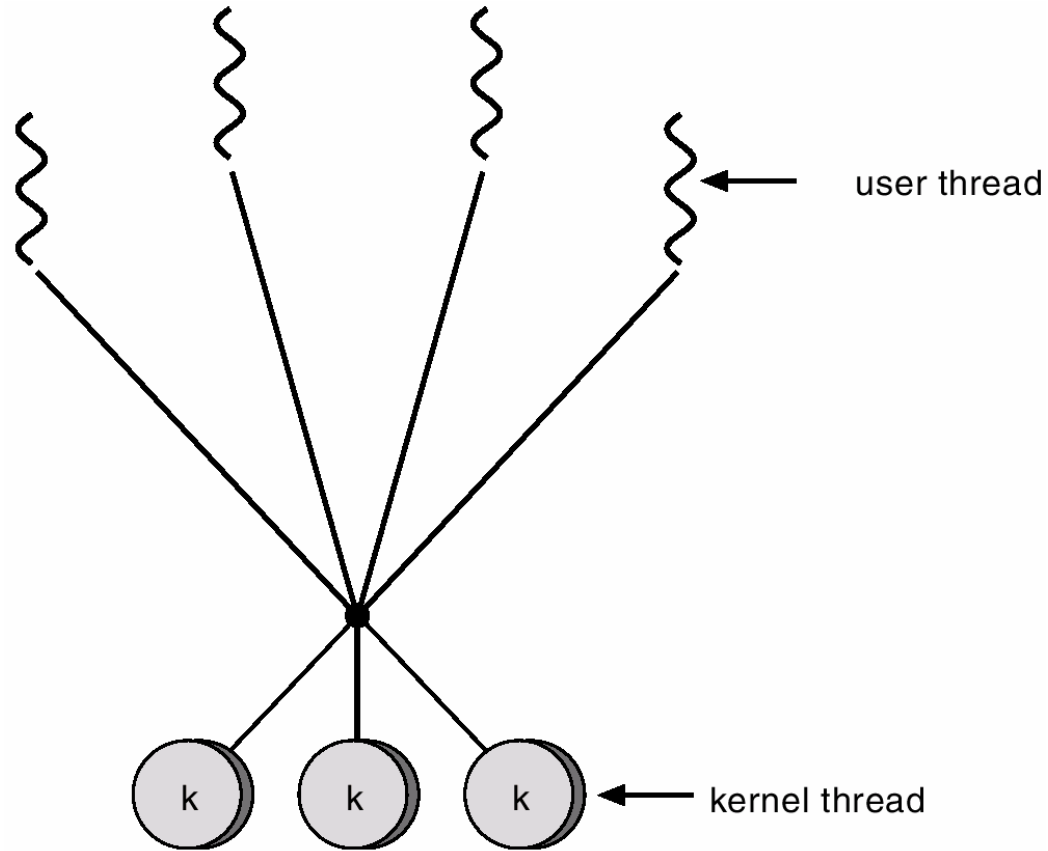
One-to-one Model



Many-to-Many

- Memungkinkan banyak user-level thread dipetakan ke banyak kernel thread
- Memungkinkan sistem operasi membuat jumlah kernel thread yang mencukupi
- Keuntungan : dapat berjalan pada multiprosesor

Many-to-Many Model



Cancellation

- Pemberhentian thread sebelum tugasnya selesai
- Target thread : thread yang akan diberhentikan
- Dapat terjadi melalui dua cara :
 - Asynchronous cancellation : suatu thread seketika itu juga memberhentikan target thread
 - Deferred cancellation : target thread memeriksa apakah dia harus berhenti, cara ini memperbolehkan target thread untuk memberhentikan dirinya sendiri secara terurut

Penanganan Sinyal

- Sinyal dimunculkan oleh suatu kejadian, sinyal tersebut dikirim ke proses, kemudian ditangani.
- Sinyal diterima secara synchronous / asynchronous tergantung dari sumber dan alasan kenapa peristiwa itu memberi sinyal
- Contoh synchronous : pembagian dengan nol, sinyal dikirim ke proses yg melakukan operasi tsb
- Contoh asynchronous : sinyal untuk mematikan proses dengan keyboard (Alt+F4)
 - Dimunculkan oleh peristiwa di luar proses yang sedang berjalan