

# SISTEM OPERASI

Pertemuan 4 – Thread, Symmetric Multiprocessing, dan Microkernel

# Menu

- Definisi Thread
- Hubungan Proses dan Thread
- Multithreading
- Definisi Symmetric Multiprocessing
- Organisasi SMP
- Microkernel

# Definisi Thread

- Sebuah thread adalah bagian dari proses, sehingga jika ada proses yang berjalan setidaknya ada satu thread di dalamnya.
- Thread dapat mengakses langsung data segmen dari proses utama nya.
- Thread dapat berkomunikasi dengan thread lain tanpa harus menggunakan Inter-process communication

# Dalam Thread

- Status eksekusi (running, ready, dll.)
- Konteks thread tersimpan saat tidak berjalan (not running)
- Stack eksekusi
- Beberapa storage statis per-thread bagi variabel lokal Akses terhadap memory & sumber daya dari prosesnya (semua thread dari proses saling berbagi pakai)

# Jenis Thread

- **User Level Thread (ULT)**
- **Kernel level Thread (KLT) juga disebut:**
  - kernel-supported threads
  - lightweight processes.

# Keuntungan Thread

- Pembuatan thread baru lebih cepat dari pada proses baru
- Penghentian thread lebih singkat daripada penghentian proses
- Switching antara dua thread lebih singkat daripada switching antar proses
- Thread dapat berkomunikasi satu sama lain
- Tanpa memanggil kernel

# Kernel-Level Thread

- **Kernel memelihara informasi konteks bagi proses & thread**
  - Manajemen thread dilakukan oleh kernel, bukan aplikasi
- **Scheduling dilakukan pada basis thread**
- **Windows menggunakan pendekatan ini**
-

# Keuntungan KLT

- Kernel dapat secara simultan menjadwal banyak thread dari proses yang sama pada banyak processor.
- Jika satu thread dalam suatu proses diblock, kernel dapat menjadwal thread yang lain dari proses yang sama.
- Rutin kernel sendiri dapat dibuat multithread.

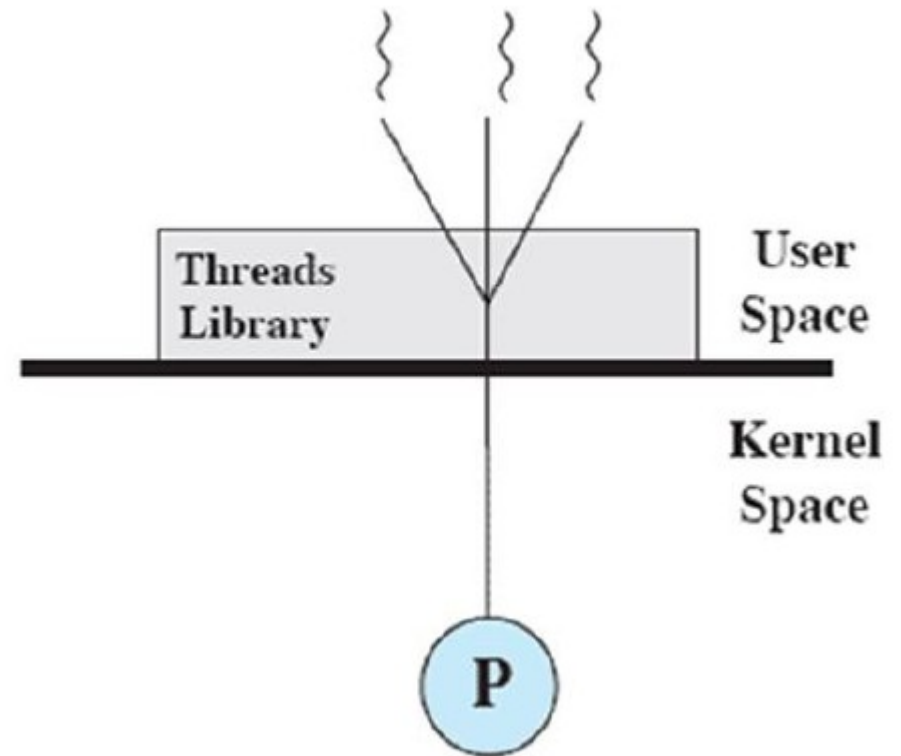


# Kerugian KLT

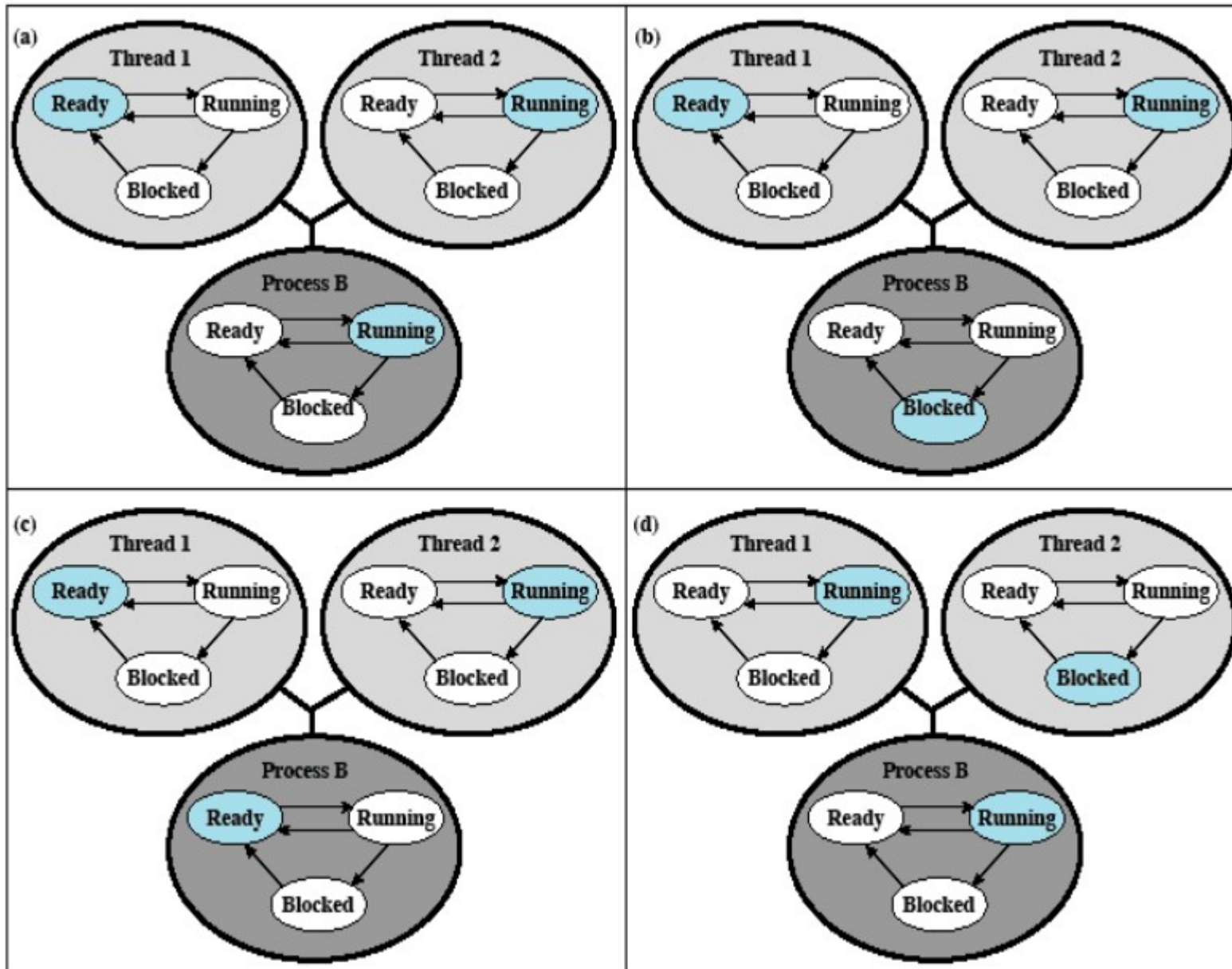
- Transfer kontrol dari satu thread ke lainnya di dalam proses yang sama mengharuskan switch modus ke kernel

# User Level Thread

Semua manajemen thread dikerjakan oleh aplikasi  
Kernel is not aware of keberadaan thread.



# Hubungan ULT dan Proses



# Process & Thread

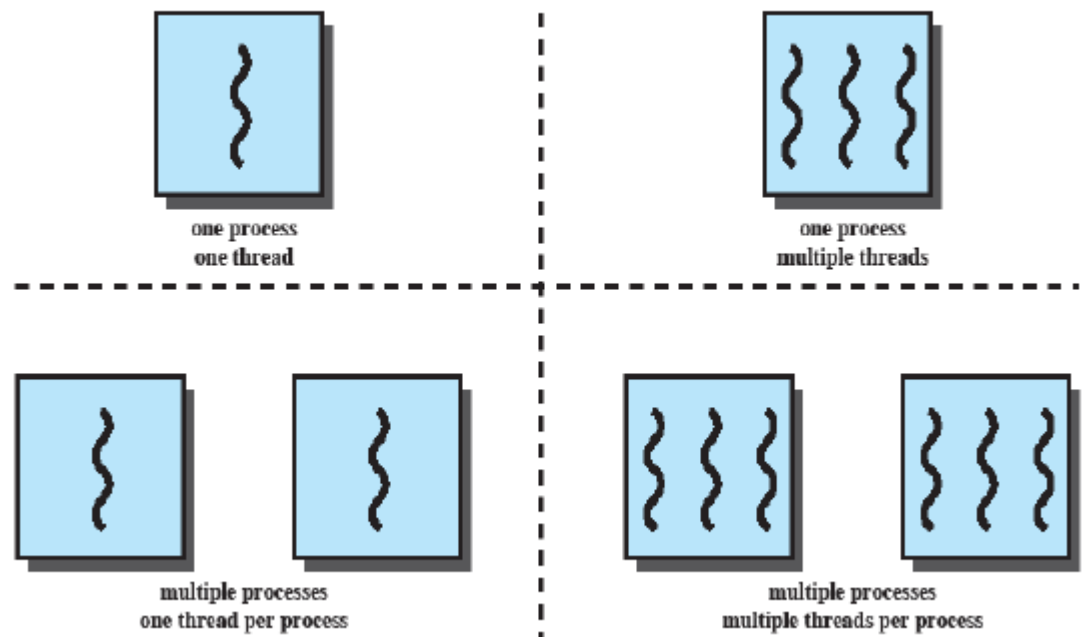
- Thread merupakan bagian dari Proses, bisa disimpulkan tidak ada Proses yang berjalan berarti tidak ada Thread
- Proses dapat memunculkan banyak thread jika diperlukan, selayaknya proses induk membuat proses anak.
- Jika proses dikendalikan oleh sistem operasi, thread dikendalikan oleh programmer.
- Thread dapat digunakan untuk aplikasi untuk memroses tugas-tugas tertentu.

# Thread dalam Sistem Single User

- Kerja foreground & background
- Pemrosesan asynchronous
- Kecepatan eksekusi
- Struktur program modular

# Multi Threading dalam Sistem Operasi

- Pada dasarnya Sistem Operasi mendukung banyak konkurensi dalam proses tunggal.



# Pendekatan Thread Tunggal

- Dalam sistem operasi Ms. DOS, satu proses hanya dibatasi satu thread saja. Di karenakan oleh kemampuan saat itu yang belum optimal seperti sekarang
- Sedangkan sistem operasi UNIX, UNIX-like mampu mendukung banyak proses dari pengguna. Namun hanya terdapat satu thread per proses

# Pendekatan Multithreading

- Program yang dibuat dengan menggunakan bahasa Java memiliki kelebihan dengan multithreading. Di karenakan program/aplikasi dengan bahasa Java tidak dijalankan langsung ke dalam sistem. Melainkan masuk ke dalam ruang virtual dari Java itu sendiri
- Sehingga Java memungkinkan sebuah proses untuk memiliki banyak thread. Yang kini disebut dengan Multithreading.



# Lanjutan

- Dan itulah kemudian kemampuan multithreading kembali dikembangkan oleh sistem-sistem operasi modern untuk melakukan multithreading dalam satu proses

# Symmetric Multi Processing

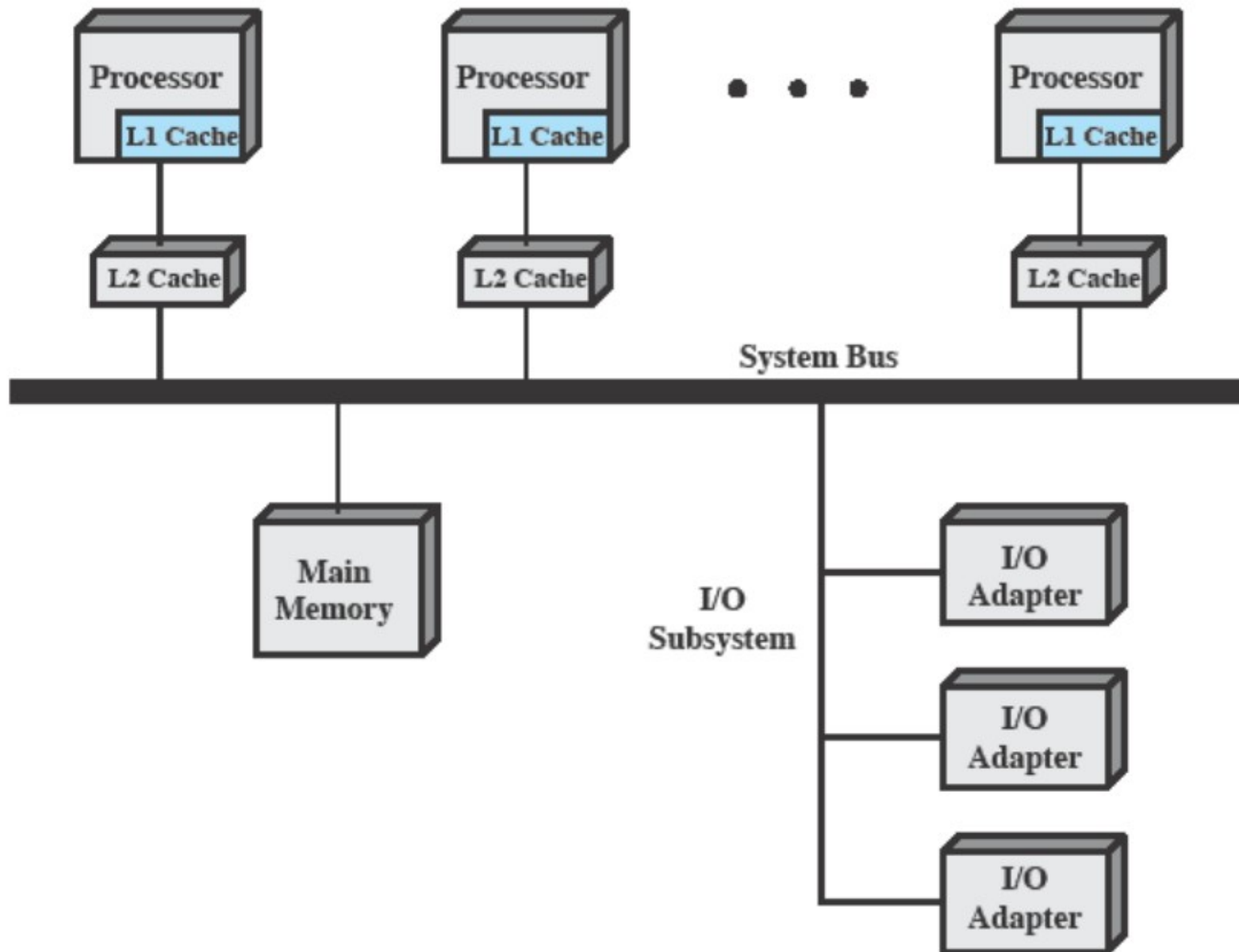
Secara tradisional, komputer ditampilkan sebagai mesin sequential.

- Processor mengeksekusi satu demi satu instruksi (satu pada satu waktu) secara urut (sequence)
- Setiap instruksi adalah suatu urutan operasi Dua pendekatan populer untuk penyediaan parallelism
- Symmetric MultiProcessors (SMP)

# Lanjutan

- **Kernel dapat berjalan pada processor manapun**
  - Memungkinkan bagian-bagian kernel berjalan secara paralel
- **Biasanya setiap processor melakukan self-scheduling dari pool proses atau thread yang tersedia**

# Arsitektur SMP



# Kategori Sistem Komputer

## Single Instruction Single Data (SISD) stream

- Processor tunggal mengeksekusi stream instruksi tunggal untuk beroperasi pada data yang disimpan dalam memory tunggal

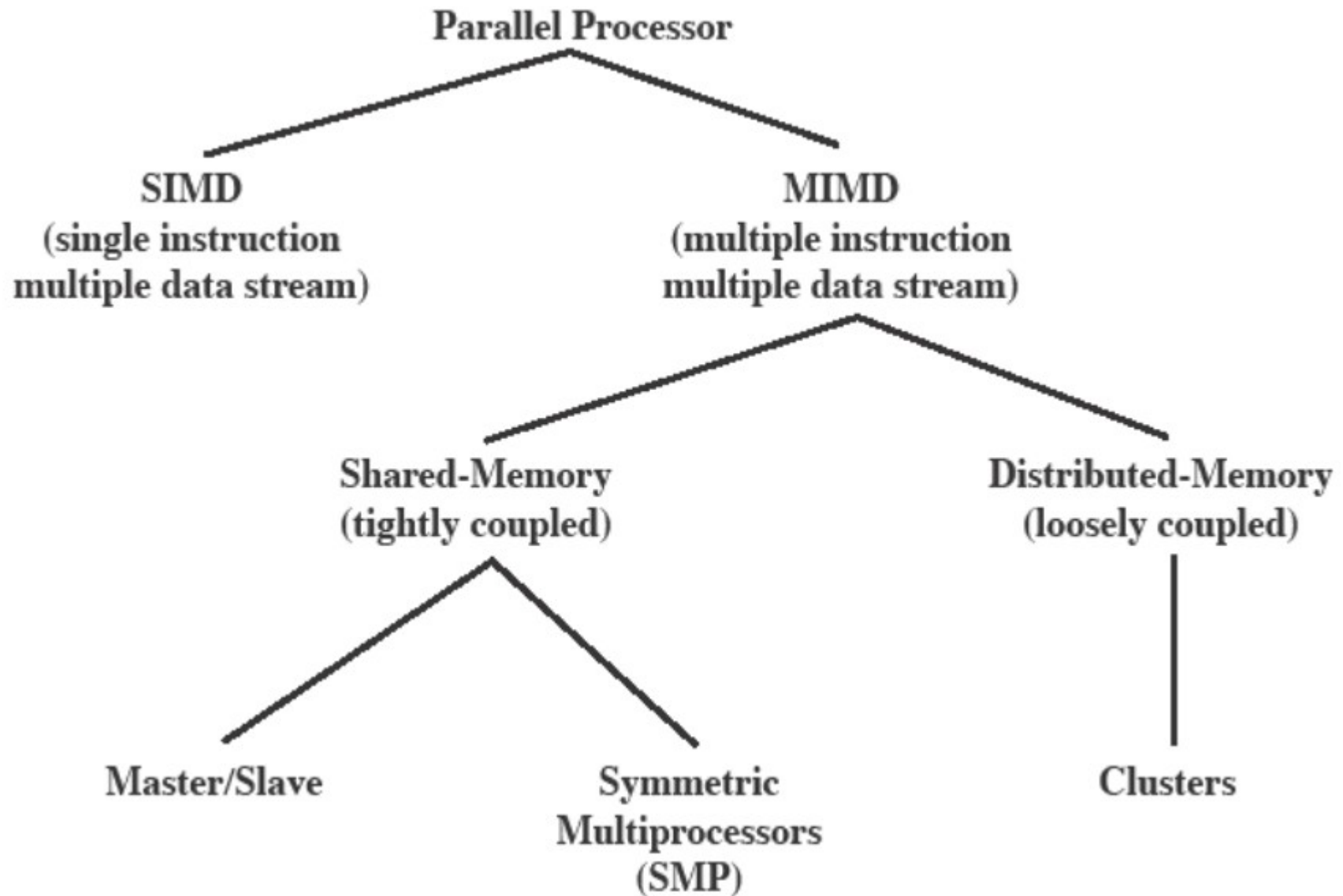
## Single Instruction Multiple Data (SIMD) stream

- Setiap instruksi dieksekusi pada himpunan data berbeda oleh processor berbeda

# Lanjutan

- Multiple Instruction Single Data (MISD) stream (Tidak pernah diimplementasikan)
- Serangkaian data ditransmisikan ke suatu himpunan processor, masing-masing mengeksekusi rangkaian instruksi berbeda
- Multiple Instruction Multiple Data (MIMD)
- Sekumpulan processor secara simultan mengeksekusi rangkaian instruksi berbeda pada kumpulan data berbeda

# Arsitektur



# Rancangan SO Multi Processor

Isu rancangan utama mencakup

- Proses atau thread concurrent simultan
- Scheduling (penjadwalan)
- Synchronization (sinkronisasi)
- Manajemen Memory
- Reliability dan Fault Tolerance



# Dukungan SMP Windows

- **Thread dapat berjalan pada processor apapun**
  - Tetapi aplikasi dapat membatasi persamaan
- **Pertalian soft**
  - Dispatcher mencoba untuk melewatkan suatu thread ready ke processor yang sama saat berjalan terakhir.
  - Ini membantu reuse data masih dalam yang cache memory processor dari eksekusi thread sebelumnya.
- **Pertalian hard**
  - Suatu aplikasi membatasi thread untuk processor tertentu

# Microkernel

- Microkernel merupakan suatu inti SO kecil yang menyediakan fondasi bagi ekstensi-ekstensi modular.
- Pertanyaan besar: seberapa kecil harusnya suatu kernel agar dapat disebut sebagai microkernel
  - Haruskah driver ada dalam user space?
- Dalam teori, pendekatan ini memberikan fleksibilitas & modularitas derajat tinggi.

# Keuntungan Microkernel

- Interface seragam pada request dibuat oleh suatu proses.
- Extensibility
- Flexibility
- Portability
- Reliability
- Dukungan sistem terdistribusi
- Sistem operasi berorientasi obyek

# Kuis Pengganti Kelas

1. Apa itu Sistem Operasi?
2. Jelaskan Fungsi dari Sistem Operasi!
3. Jelaskan Apa Itu Proses?
4. Jelaskan Peran Process Control Block!
5. Buatlah Ilustrasi dari
  - Model Proses Dua Status
  - Model Proses Lima Status