

MANAJEMEN MEMORI

Memori dibedakan menjadi dua yaitu

Primary Memory



- Karakteristik
- Ø Volatil
 - Ø Kecepatan tinggi
 - Ø Akses random

Secondary Memory



- Karakteristik
- Ø non volatil
 - Ø kecepatan relatif rendah
 - Ø sekuensial

PENDAHULUAN

- Ø Organisasi dan manajemen memori sangat mempengaruhi kinerja komputer. Manajemen memori melakukan tugas penting dan kompleks berkaitan dengan:
 - q Sumberdaya yang harus dipakai bersama di antara sejumlah proses yang aktif.
 - q Upaya agar pemrogram dan proses tidak dibatasi kapasitas memori fisik di sistem komputer.

1. MANAJEMEN MEMORI

Ø Manajemen memori mempunyai fungsi sebagai berikut:

- Mengelola informasi memori yang dipakai dan tidak dipakai
- Mengalokasikan memori ke proses yang memerlukan
- Menddealokasikan memori dari proses telah selesai
- Mngelola swapping antar memori utama dan disk

Ø Swapping adalah pemindahan proses dari memori utama ke disk dan sebaliknya

1.1 Manajemen Memori pada Sistem Multiprogramming

Q Dengan banyaknya alamat milik proses-proses di memori utama, maka sistem operasi harus mendukung 2 kebutuhan yg saling bertentangan, yaitu:

1. Pemisahan ruang-ruang alamat
2. Pemakaian bersama memori

Q Manajer memori harus memaksakan isolasi ruang-ruang alamat tiap proses agar mencegah proses yang ingin berlaku jahat mengakses dan merusak ruang alamat proses lain.

Q Manajer memori di lingkungan multiprogramming, sekaligus melakukan dua hal, yaitu:

1. Proteksi memori dengan isolasi ruang-ruang alamat secara disjoint
2. Pemakaian bersama memori

FUNGSI DARI MANAJEMEN MEMORI

- q Mengelola informasi mengenai memori yang dipakai dan tidak dipakai sistem
- q Mengalokasikan memori ke proses yang memerlukan
- q Mengalokasikan memori dari proses telah selesai digunakan
- q Mengalokasikan swapping antara memori utama dan harddisk

Manajemen Memori Pada Sistem Multiprogramming

1.2 Klasifikasi Manajemen Memori

Klasifikasi manajemen memori dapat digambarkan dalam skema berikut ini a proses tidak saling mengganggu

- o pemakaian memori secara bersama sehingga memungkinkan proses bekerja sama mengakses daerah memori bersama

Nyata	Nyata		Maya		
(1) Sistem khusus untuk pemakai tunggal	Sistem Multiprogramming dengan memori nyata		Sistem Multiprogramming dengan memori maya		
	Multiprogramming dengan pemartisian tetap	(4) Multiprogramming dengan pemartisian tetap	(5) Sistem paging murni	(6) Sistem segmentasi murni	(7) Kombinasi peging dan segmentasi
	(2) Ditempatkan secara absolut	(2) Dapat direlokasi			

Manajemen Memori Berdasarkan Keberadaan Swapping

q Berdasarkan keberadaan swapping terbagi menjadi 2:

1. Manajemen dengan swapping, yaitu manajemen tanpa pemindahan citra/gambaran proses antara memori utama dan disk selama eksekusi.
2. Manajemen tanpa swapping, yaitu manajemen memori dengan pemindahan citra/gambaran proses antara memori utama dan disk selama eksekusi.

Manajemen Memori Berdasarkan Alokasi Memori

1. Alokasi memori berurutan (kontinyu)

Tiap proses menempati satu blok tunggal lokasi memori yang berurutan.

∅ Keunggulan:

- Sederhana
- tidak akan terbentuk lubang-lubang (rongga) memori bersebaran.
- Karena berurutan, proses dapat dieksekusi dengan cepat.

∅ Kelemahannya:

- dapat memboroskan Memori
- tidak dapat memuatkan proses bila tidak ada satu blok memori yang mencukupi.

2. Alokasi memori tak berurutan (non kontinyu)

Program dibagi menjadi beberapa blok atau segmen. Blok- blok program ditempatkan di memori dalam potongan-potongan tanpa perlu saling berdekatan. Teknik ini biasa digunakan pada sistem memori mayasebagai alokasi page-page dilakukan secara global.

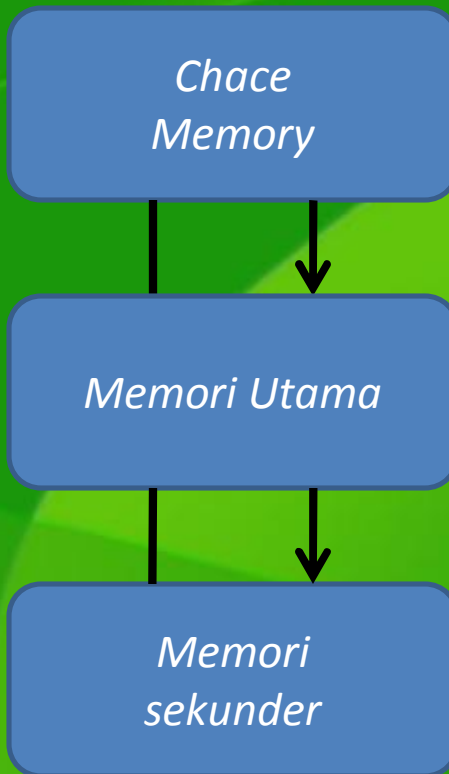
Ø Keuntungan:

- Sistem dapat memanfaatkan memori utama secaralebih efisien
- Sistem operasi masih mampu memuatkan proses bilajumlah total lubang-lubang memori cukup untukmemuat proses yang akan dieksekusi

Ø Kelemahan:

- Memerlukan pengendalian yang lebih rumit dan sulit
- Memori dapat menjadi banyak lubang tersebar (memori tak terpakai bertebaran)

HIRARKI MEMORI



3. MANAJEMEN MEMORI TANPA SWAPPING

Manajemen memori tanpa swapping, terdiri dari:

1. Monoprogramming, sistem komputer hanya mengizinkan satu program/pemakai berjalan pada satu waktu.
2. Multiprogramming dengan pemartisian statis, membagi memori menjadi sejumlah partisi tetap, pada partisi-partisi tersebut proses-proses ditempatkan.

Manajemen Memori Tanpa *Swapping* Terdiri dari

Q Monoprogramming

Ciri–Ciri Monoprogramming

- Hanya satu Proses pada satu saat
- Hanya satu proses menggunakan semua memori
- Pemakai memuatkan program ke seluruh memori dari *disk* atau *tipe*
- Program mengambil kendali seluruh mesin.

Ø *Embedded Systems*

Teknik *monoprogramming* masih dapat dipakai untuk sistem kecil yaitu *sistem tempelan (embedded-system)* yang menempel atau terdapat di sistem lain. sistem tempelan biasanya mengendalikan satu alat

Contoh sistem tempelan yang ada di sebuah mobil anantara lain:

- § Pengendalian Pengapian
- § Pengendalian Kemudi
- § Pengendalian Pengereman
- § Pengendalian Bahan Bakar
- § dan sebagainya

Ø Proteksi pada *monoprogramming* sederhana

Pada *monoprogramming*, pemakai mempunyai kendali penuh terhadap seluruh memori utama

Memori terbagi menjadi tiga bagian, yaitu:

1. Bagian yang berisi rutin-rutin sistem operasi
2. Bagian yang berisi program pemakai
3. Bagian yang tidak digunakan

q *Multiprogramming* Dengan Pemartisian Statis

- Mempermudah pemrogram
- Agar dapat memberi layanan interaktif ke beberapa orang secara simultan
- Efisiensi penggunaan sumber daya
- Eksekusi lebih murah jika proses besar dipecah menjadi beberapa proses kecil
- Dapat mengerjakan sejumlah proses secara simultan

3.1 Monoprogramming

q Ciri-cirinya:

Ø Satu proses pada satu alat

Ø Hanya satu proses menggunakan semua memori

Ø Pemakai memuatkan program ke seluruh memori dari disk atau tape
Program mengambil kendali seluruh sistem

q Karena hanya terdapat satu proses dan menguasai seluruh sistem, maka alokasi memori dilakukan secara berurutan.

q Teknik monoprogramming masih dipakai untuk sistem kecil atau terdapat di sistem lain.

q Sistem yaitu sistem tempelan (embedded-system) yang menempel operasi harus diproteksi dari modifikasi program pemakai dengan cara memproteksi rutin sistem operasi. Proteksi ini diimplementasikan menggunakan satu register batas (boundary register) di pemroses.

3.2 Multiprogramming dengan Pemartisian Statis

q Beberapa alasan mengapa multiprogramming digunakan:

- Ø Mempermudah pemrogram
- Ø Agar dapat memberikan layanan interaktif ke beberapa orang secara simultan
- Ø Efisiensi penggunaan sumber daya
- Ø Eksekusi lebih murah jika proses besar dipecah menjadi beberapa proses kecil
- Ø Dapat mengerjakan sejumlah job secara simultan

q Kelemahannya:

- Ø Bila program ukuran lebih besar dibanding partisi yang tersedia maka tidak dapat dimuatkan, tidak dapat dijalankan.
- Ø Untuk program yang sangat kecil dibanding ukuran partisi yang ditetapkan, maka banyak ruang yang tidak dipakai yang diborosan, disebut fragmentasi internal.

Pemberian Alamat

Q Sebelum masuk ke memori, suatu proses harus menunggu. Hal ini disebut Input Queue

Pemberian Alamat (2)

Penjilidan alamat dapat terjadi pada 3 saat, yaitu :

Compile Time : pada saat proses di-compile,
menggunakan kode absolut.

Load : pada saat proses dipanggil,
menggunakan kode yang direlokasi.

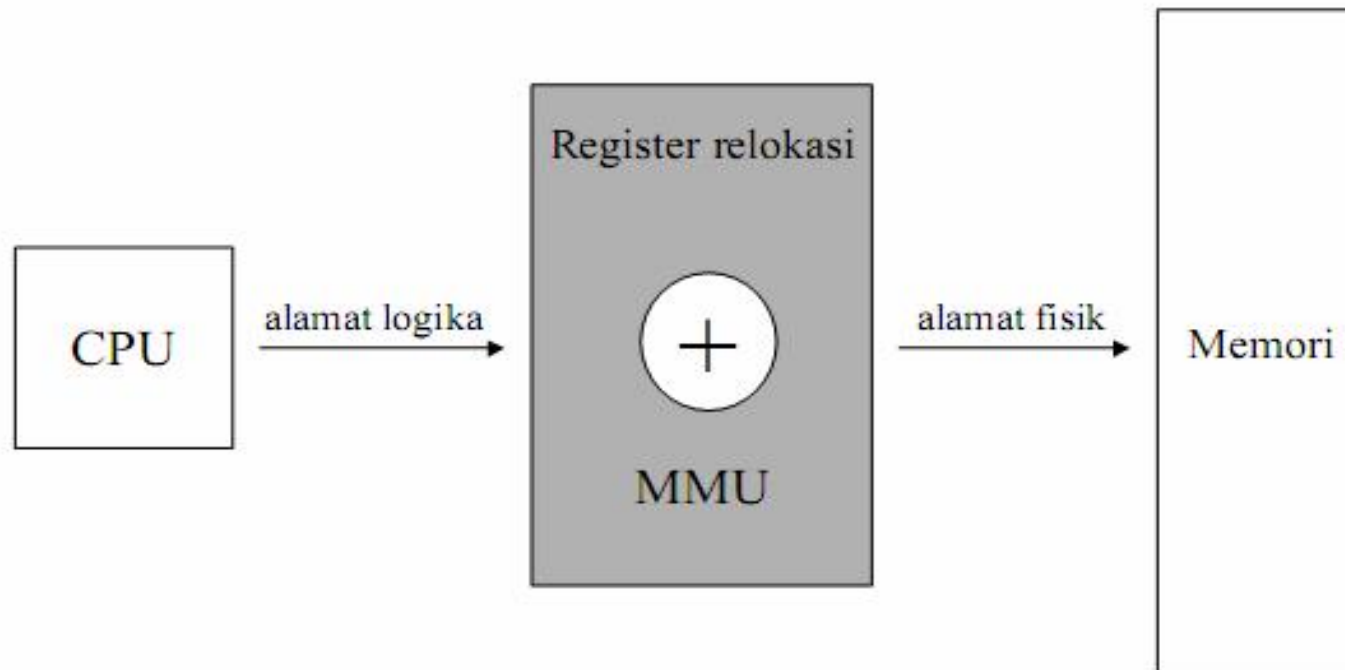
Execution Time : pada saat proses dijalankan,
memerlukan perangkat
kerastersendiri.

Ruang Alamat Logika & Fisik

Alamat Logika adalah alamat yg dibentuk di CPU, disebut juga alamat virtual.

- ✓ Alamat fisik adalah alamat yang terlihat oleh memori.
- ✓ Untuk mengubah dari alamat logika ke alamat fisik diperlukan suatu perangkat keras yang bernama MMU (Memory Management Unit).
- ✓ Pengubahan dari alamat logika ke alamat fisik adalah pusat dari manajemen memori.

MMU (*Memory Management Unit*)



PEMANGGILAN DINAMIS

- q Memanggil routine yang diperlukan untuk menjalankan suatu proses.
- q Routine yang tidak diperlukan, tidak akan dipanggil.
- q Tidak memerlukan bantuan sistem operasi.

RUANG ALAMAT LOGIKA & FISIK

Penghubungan Dinamis dan Kumpulan Data Bersama

- q Menghubungkan semua routine yang ada di kumpulan data.
- q Tidak membuang-buang tempat di disk dan memori.
- q Kumpulan data yang ada dapat digunakan bersama-sama.
- q Membutuhkan bantuan sistem operasi.

PENGHUBUNGAN STATIS

- ❑ Menghubungkan seluruh routine yang ada ke dalam suatu ruang alamat.
- ❑ Setiap program memiliki salinan dari seluruh kumpulandata.

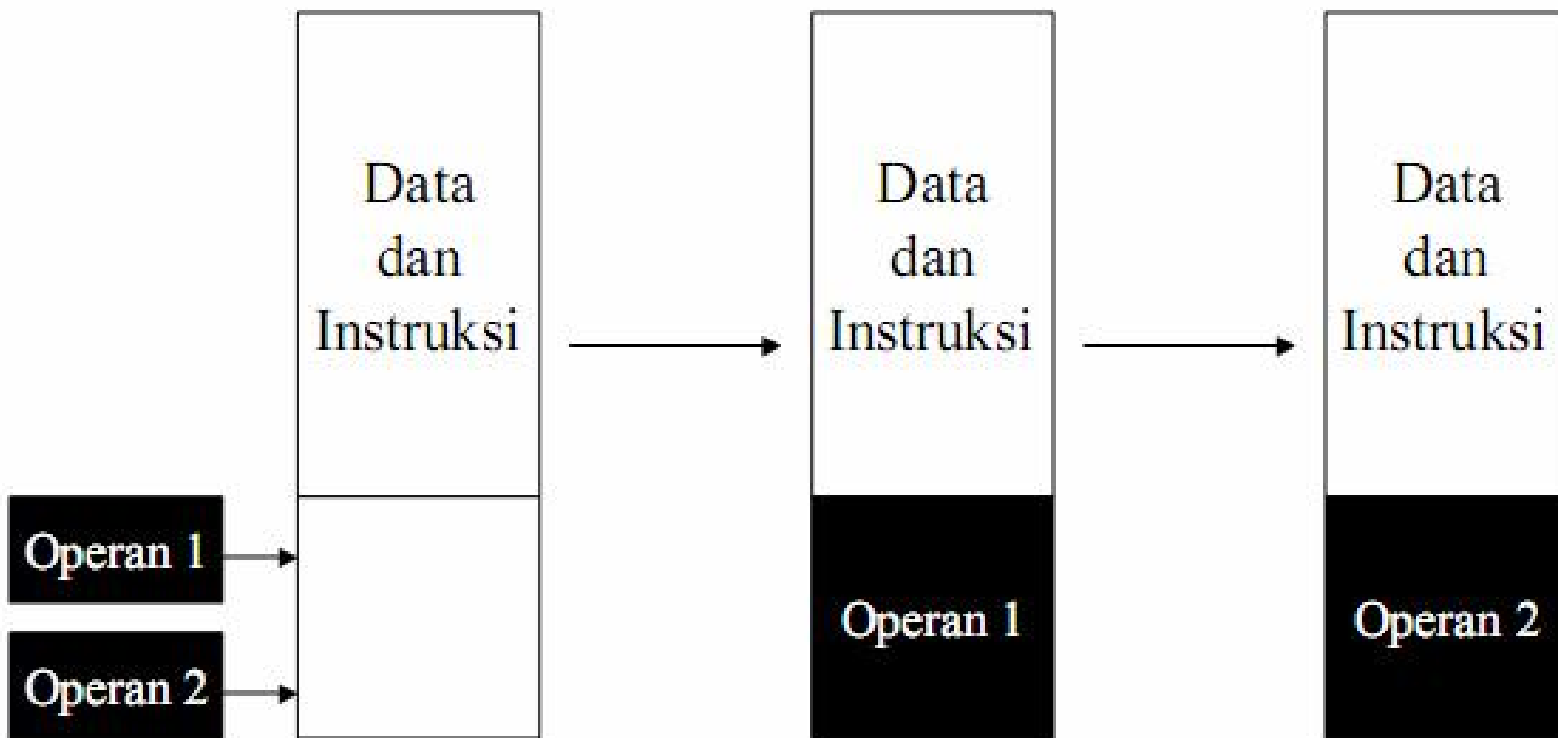
OVERLAYS (1)

- q Untuk memasukkan suatu proses yang membutuhkan memori lebih besar dari yang tersedia.
- q Data dan instruksi yang diperlukan dimasukkan langsung ke memori.
- q Routine-nya dimasukkan ke memori secara bergantian.
- q Memerlukan algoritma tambahan untuk melakukan overlays.

OVERLAYS (2)

- q Tidak memerlukan bantuan dari sistem operasi.
- q Sangat sulit untuk dilakukan.
- q Dapat dilakukan di komputer mikro.

Perakit dengan 2 operan



SWAPPING

- q Proses harus berada di dalam memori untuk dapat dijalankan.
- q Sebuah proses dapat di-swap sementara keluar memori ke sebuah penyimpanan cadangan untuk kemudian dikembalikan lagi ke memori.
- q Roll out, roll in adalah penjadualan swapping berbasis pada prioritas (proses berprioritas rendah di-swap keluar memori agar proses berprioritas tinggi dapat masuk dan dijalankan di memori).

PENGALOKASIAN MEMORI (1)

Q Salah satu tanggung jawab dari Sistem Operasi adalah mengontrol akses ke sumberdaya sistem. Salah satunya adalah memori.

Q Contiguous Memory Allocation: alamat memori diberikan kepada proses secara berurutan dari kecil kebesar.

Q Keuntungan Contiguous daripada Non-contiguous: sederhana, cepat, mendukung proteksi memori.

Q Kerugian Contiguous daripada Non-contiguous: jika tidak semua proses dialokasikan di waktu yang sama, akan menjadi sangat tidak efektif dan mempercepat habisnya memori.

ADA 2 TIPE CONTIGUOUS MEMORY ALLOCATION:

Partisi tunggal dan partisi banyak.

∅ Partisi tunggal adalah alamat pertama yang dialokasikan untuk proses adalah yang berikutnya da alamat yang dialokasikan untuk proses sebelumnya.

∅ Partisi banyak adalah dimana Sistem Operasi menyimpan informasi tentang semua bagian memori yang tersedia untuk digunakan (disebut hole).

∅ Proses yang akan dialokasikan dimasukkan ke dalam antrian dan algoritma penjadualan digunakan untuk menentukan proses mana yang akan dialokasikan berikutnya.

ADA 2 CARA PENGATURAN PARTISI PADA SISTEM PARTISI BANYAK:

Partisi tetap, dan partisi dinamis.

∅ Partisi tetap adalah apabila memori dipartisi menjadi blok-blok yang ukurannya ditentukan dari awal.

Terbagi lagi atas partisi tetap berukuran sama, dan partisi tetap berukuran berbeda.

∅ Partisi dinamis adalah memori dipartisi menjadi bagian-bagian dengan jumlah dan besar yang tidak tentu.

Algoritma Pengalokasian Memori dengan Partisi Dinamis

- Ø First fit : Mengalokasikan hole pertama yang besarnya mencukupi. Pencarian dimulai dari awal.
- Ø Best fit : Mengalokasikan hole terkecil yang besarnya mencukupi.
- Ø Next fit : Mengalokasikan hole pertama yang besarnya mencukupi. Pencarian dimulai dari akhir pencarian sebelumnya.
- Ø Worst fit : Mengalokasikan hole terbesar yang tersedia.

Ruang Alamat Logika & Fisik



Blok terakhir yang dialokasikan

