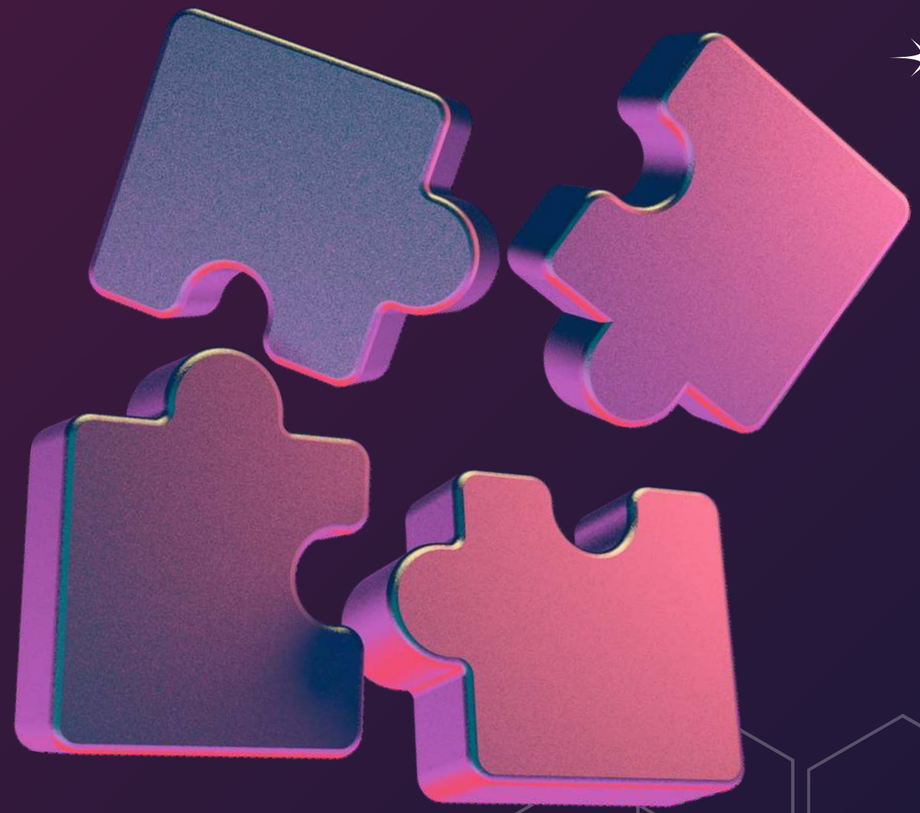
MEMORY
MANAGEMENT

SISTEM OPERASI

MEMORY MANAGEMENT (MANAJEMEN MEMORI)



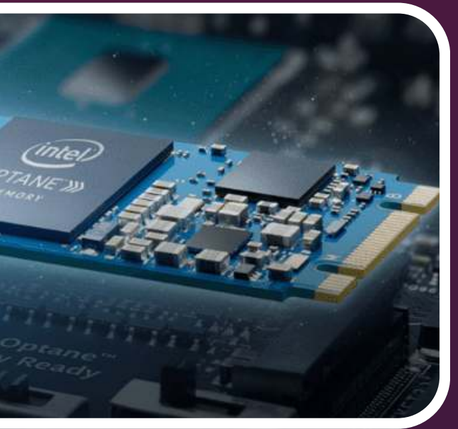
KELOMPOK 2

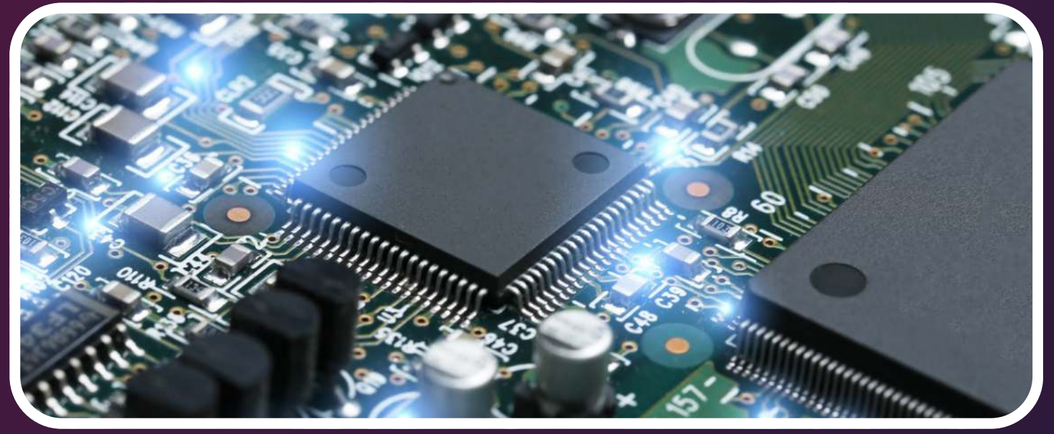
MEMORY MANAGEMENT

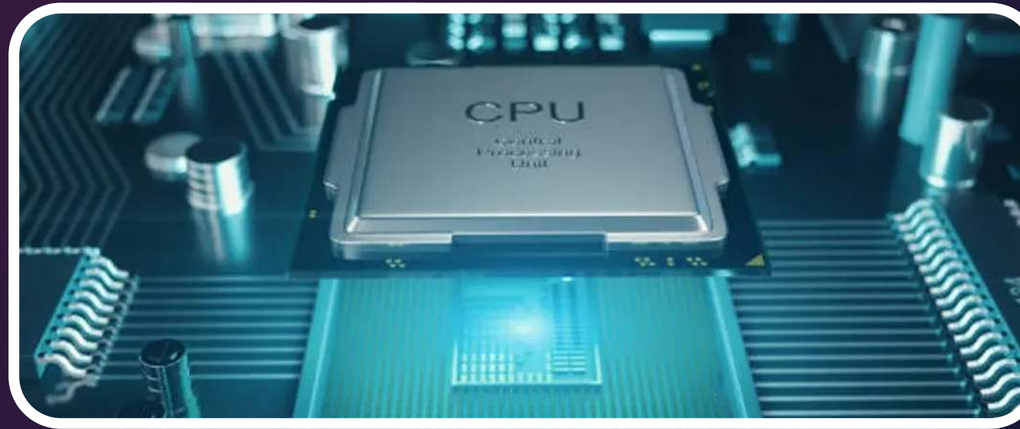
SISTEM OPERASI

ANGGOTA KELOMPOK 2

1. ATHAYA RAIHAN ANNAFI (H1D023001)
2. IRFAN ROMADHON WIDODO (H1D023023)
3. PRASETYO ANGGA PERMANA (H1D023028)
4. AUDI MAKRUFIANTO AFETAMA (H1D023037)
5. AYU FITRIANINGSIH (H1D023038)
6. AISYAH SYIFA KARIMA (H1D023043)
7. DHIMAS WILDAN NUR ZAKARIYA (H1D023050)







MEMORY MANAGEMENT

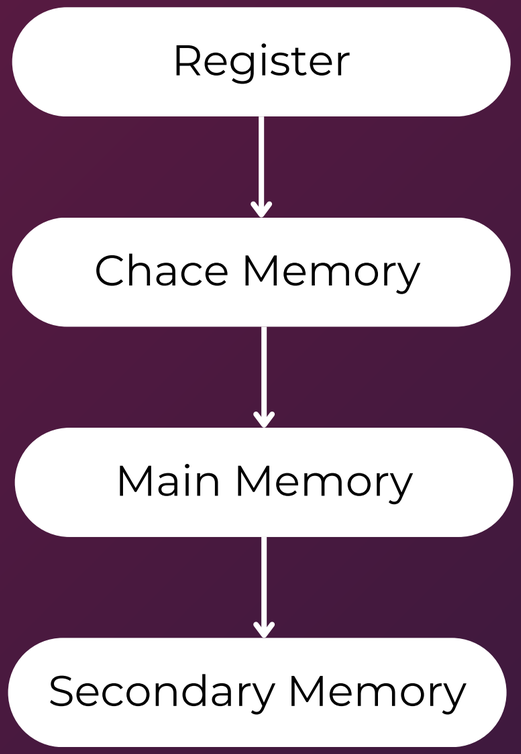
SISTEM OPERASI

MANAJEMEN MEMORI

- Memori adalah pusat dari operasi pada sistem komputer modern yang berfungsi sebagai tempat penyimpanan informasi yang harus diatur dan dijaga sebaik-baiknya, memori adalah array besar dari word atau byte, yang disebut alamat.
- Sedangkan manajemen memori adalah suatu kegiatan untuk mengelola memori komputer, proses ini menyediakan cara untuk mengalokasikan memori.

MEMORY MANAGEMENT

KONSEP DASAR



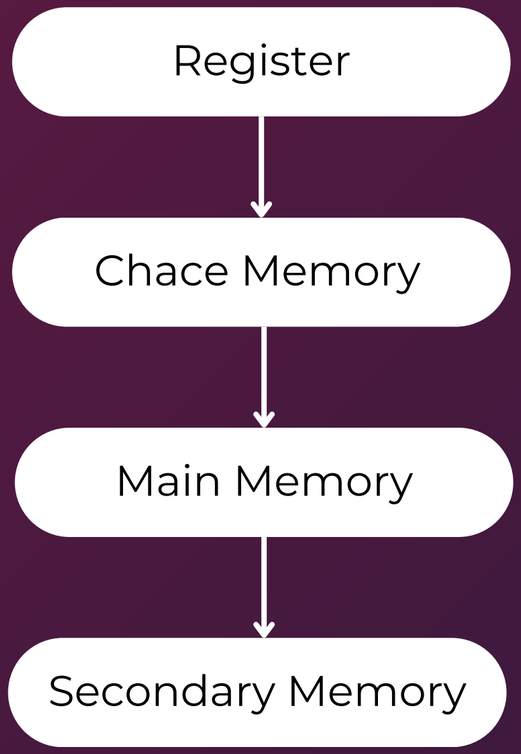
Hierarki organisasi memori pada sistem komputer

- Register : Contoh memori register adalah IR (Instruction Register) untuk menampung kode instruksi yang akan dieksekusi serta AX, BX, CX, DX untuk menampung data dan informasi. Kapasitas memori register sangat terbatas sehingga memerlukan memori utama.
- Cache Memory : Untuk mengatasi perbedaan kecepatan, digunakan teknik caching untuk memori utama dengan menggunakan memori cache. Umumnya berada dalam prosesor. Kapasitas jauh lebih kecil dari memori utamadan Kecepatan transfer mengikuti clock processor. Prinsip kerjanya sebagai salinan bayangan dari data dan kode instruksi di memori utama.

MEMORY MANAGEMENT

SISTEM OPERASI

KONSEP DASAR



Hierarki organisasi memori pada sistem komputer

- Main Memory : Pada umumnya dapat diakses secara random (RAM) dan volatile. Namun sayangnya kecepatan transfer data dari memori utama ke prosesor sangat lambat jika dibandingkan dengan eksekusi prosesor.
- Secondary Memory : Pada umumnya berupa disk dan bersifat non-volatile. Kecepatan transfer jauh lebih lambat dari memori utama. Untuk mengatasi kekurangan tempat pada ruang memori utama bisa menggunakan virtual memory.

MEMORY MANAGEMENT

SISTEM OPERASI

TUJUAN PENGORGANISASIAN MEMORI KOMPUTER

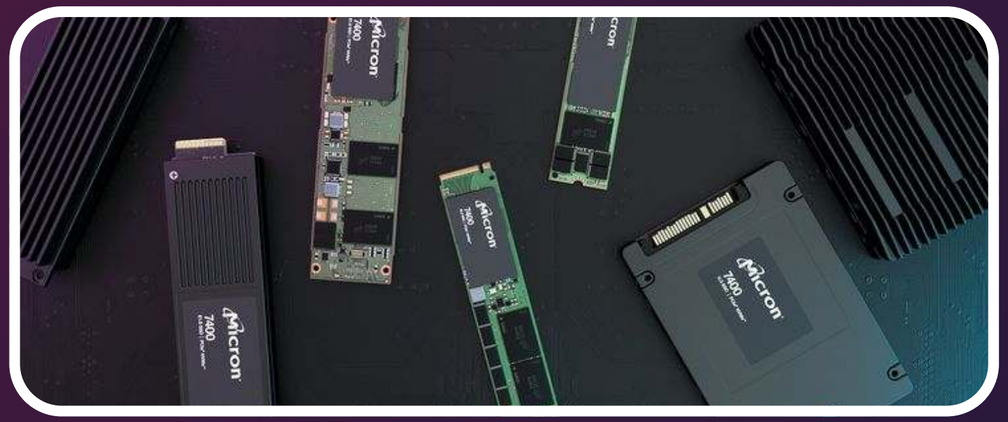
Meningkatkan kecepatan akses kode instruksi dan data oleh prosesor

Pengorganisasian memori yang tepat memungkinkan prosesor untuk mengakses instruksi dan data dengan lebih cepat, sehingga meningkatkan kinerja keseluruhan sistem komputer. Ini dicapai dengan menyimpan instruksi dan data yang sering diakses di jenis memori yang lebih cepat, seperti cache atau memori utama.

Mengurangi waktu menganggur (idle) prosesor

Waktu menganggur prosesor terjadi ketika prosesor harus menunggu untuk mengakses instruksi atau data dari memori. Dengan pengorganisasian memori yang baik, waktu menganggur prosesor dapat diminimalkan dengan menyediakan akses yang cepat ke instruksi dan data yang dibutuhkan.

MEMORY MANAGEMENT



Memperbesar kapasitas penyimpanan sistem memori komputer

Pengorganisasian memori yang efisien memungkinkan sistem komputer untuk menggunakan berbagai jenis memori dengan karakteristik yang berbeda, seperti memori utama (RAM) dan memori sekunder (hard disk, SSD, dll.). Dengan demikian, kapasitas penyimpanan total sistem komputer dapat diperbesar dengan memadukan berbagai jenis memori sesuai dengan kebutuhan.

Secara umum, semakin bawah tingkatan pada hirarki organisasi komputer, maka

- Harga per satuan byte semakin rendah
- Kapasitas penyimpanan semakin besar
- Frekuensi pengaksesan semakin kurang
- Kecepatan akses semakin lambat

MEMORY MANAGEMENT

SISTEM OPERASI

KELEBIHAN

KEKURANGAN

KELEBIHAN MANAJEMEN MEMORI

Manajemen memori memiliki kelebihan yang membuat penting bagi sistem komputer, meliputi:

- Efisiensi Pengguna Memori
- Mencegah Kelebihan Beban pada Memori
- Peningkatan Kinerja Sistem
- Mencegah Kerusakan Data
- Meningkatkan Keamanan Sistem
- Meningkatkan Stabilitas Sistem

MEMORY MANAGEMENT

KELEBIHAN

KEKURANGAN

KEKURANGAN MANAJEMEN MEMORI

Manajemen memori memiliki kelemahan yang harus dipertimbangkan, meliputi:

- Sistem Overhead
- Kemungkinan Kegagalan
- Kompleksitas
- Fragmentasi Memori
- Keterlambatan Proses
- Ketergantungan pada Sistem Operasi

MEMORY MANAGEMENT

SISTEM OPERASI

PENGALAMATAN MEMORI

Pengalamatan memori bertugas untuk mereferensi kode instruksi atau data di memori utama secara tepat merupakan tanggung jawab dari compiler

- Compiler berfungsi mengubah source code yang ditulis programmer menjadi file yang berisi kode instruksi program yang dapat dijalankan prosessor

- Dalam menentukan alamat instruksi atau data, compiler mengacu pada metode pengalamatan memori yang dipakai sistem komputer

MEMORY MANAGEMENT

SISTEM OPERASI

PENGALAMATAN MEMORI

FISIK

Pengalamatan ini yaitu dimana alamat yang ditulis pada program hasil kompilasi merupakan alamat fisik memori utama yang sesungguhnya. Namun, konsekuensinya pada saat penyalinan image proses ke memori Utama maka kode dan intruksi data program harus disalin pada posisi yang sesuai dengan referensi tersebut. Jadi nanti pada saat eksekusi prosesor akan memproses alamat pada kode intruksi program secara langsung tanpa membutuhkan translasi alamat memori.

LOGIKA

Pengalamatan ini digunakan pada system yang menggunakan alokasi memori system berurut dimana keseluruhan image proses harus terletak pada satu area memori yang utuh. Alamat pada kode intruksi program merupakan alamat relative atau offside terhadap posisi awal program. Jadi pada saat image proses dari program tersebut disalin atau dialokasikan ke memori utama alamat awal memorinya dicatat pada suatu register alokasi. Pada saat eksekusi pengaksesan alamat akan ditranslasi dengan menjumlahkan alamat referensi awal pada intruksi dengan isi register alokasi untuk mendapatkan alamat fisik memori yang akan benar-benar diakses. Proses translasi biasanya menggunakan pearangkat keras khusus yaitu memory Management Unit (MMU)

MEMORY MANAGEMENT

ADDRESS BINDING

- Aktivitas translasi alamat ini disebut dengan address binding
- Alamat yang terdapat dalam kode instruksi tidak selamanya berupa alamat fisik, tapi dapat berupa alamat logika yang perlu ditranslasi lebih dahulu

```
planations see
/dev.mysql.com/doc/mysql/en/server-system-variables.html

        = /var/run/mysqld/mysqld.pid
        = /var/run/mysqld/mysqld.sock
        = /var/lib/mysql
r       = /var/log/mysql/error.log
ault we only accept connections from localhost
ress    = 0.0.0.0
ing symbolic-links is recommended to prevent assorted security ri
-links=0
```

MEMORY MANAGEMENT

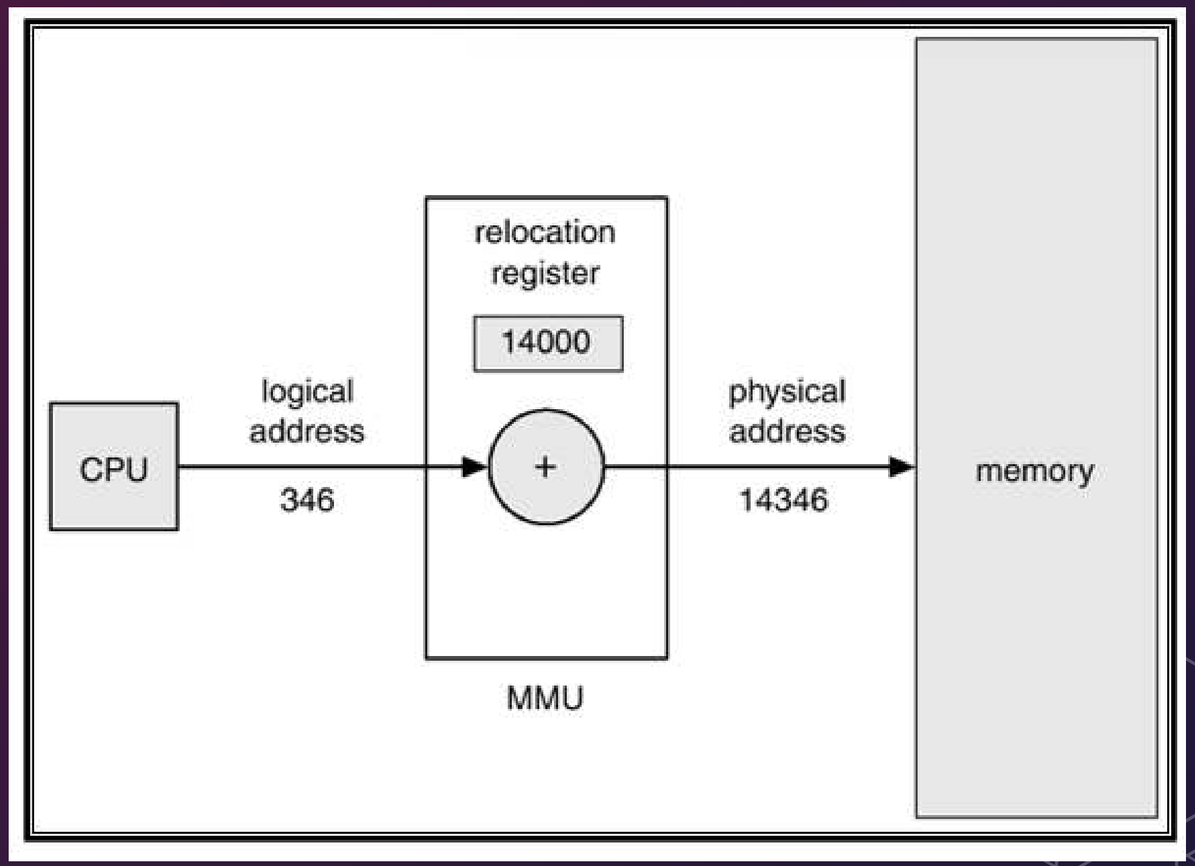
SISTEM OPERASI

ADDRESS BINDING

Address binding dapat terjadi pada saat :

- Compile Time
  - Apabila dimungkinkan letak / alamat fisik memori diketahui sebelum diekseskusi agar langsung dapat ditulis pada source code.
  - Kelemahan: program tidak dapat direlokasi selama eksekusi.
- Loading Time
  - Dilakukan pada saat loading program ke memori utama
  - Hasil kompilasi disimpan dalam file yang berisi alamat fisik.
  - Jika terjadi perubahan relokasi maka code di-load ulang
- Execution Time
  - Membutuhkan perangkat keras seperti MMU (Memory Management Unit)
  - MMU bertanggung jawab membantu proses perhitungan transasi alamat logika ke alamat fisik pada saat eksekusi.
  - Dimungkinkan suatu proses berpindah alamat sewaktu dieksekusi.

RELOKASI DINAMIK MENGGUNAKAN REGISTER RELOKASI



MEMORY MANAGEMENT

SISTEM OPERASI

DYNAMIC LOADING

Apa itu Dynamic Loading ?

Dynamic loading adalah teknik di mana suatu program atau modul dimuat ke dalam memori hanya saat diperlukan selama runtime, bukan saat aplikasi dimulai. Ini membantu mengoptimalkan penggunaan memori dan meningkatkan efisiensi aplikasi.

Mekanisme dasar :

- Program utama di-load dan dieksekusi.
- Pada saat suatu routine butuh memanggil routine yang lain, maka pertama routine pemanggil mengecek apakah rotine yang dibutuhkan sudah pernah diambil. Jika belum, maka routine yang dipanggil tersebut akan diambil dan dialokasikan di memori utama

MEMORY MANAGEMENT

DYNAMIC LOADING

Contoh Penggunaan :

Banyak aplikasi modern, terutama yang kompleks seperti perangkat lunak grafis atau aplikasi web, menggunakan dynamic loading untuk mengelola sumber daya dan meningkatkan kinerja.

Keuntungan Dynamic Loading :

- Rutin yang tidak digunakan tak akan pernah di-load ke memori.
- Untuk menghindari pemakaian rutin yang salah dalam program dengan jumlah kode yang besar.
- Tidak memerlukan bantuan sistem operasi. Metode ini menjadi tanggung jawab user/programmer. SO hanya menyediakan routine library

MEMORY MANAGEMENT

SISTEM OPERASI

DYNAMIC LINKING

Definisi Dynamic Linking

Sama seperti dynamic loading, dynamic linking adalah proses yang digunakan dalam sistem operasi untuk menghubungkan program ke library pada saat runtime, bukan kompilasi, melainkan referesi ke pustaka eksternal yang akan dihubungkan saat program dijalankan.

1. **Keuntungan Dynamic Linking**
- memungkinkan berbagai program untuk menggunakan library yang sama di memori
- dapat memperbarui library yang berubah tanpa perlu mengompile ulang semua program
- size program dapat menjadi lebih kecil karena library hanya dimuat saat program dijalankan dan tidak perlu menyertakan semua kode library dalam setiap program

1. **Kelemahan Dynamic Linking**
- memiliki ketergantungan pada library eksternal
- ketergantungan pada library eksternal meningkatkan risiko keamanan karena library tersebut dapat disalahgunakan atau dimanipulasi

MEMORY MANAGEMENT

SISTEM OPERASI

DYNAMIC LINKING

Shared Library

SHARED LIBRARY: pustaka yang dimuat ke memori sekali dan diAgunakan oleh beberapa program sekaligus

Dynamical Library

DYNAMICALLY LIBRARY: library yang dimuat ke memori hanya ketika diperlukan oleh program dan dapat dimuat dan dibebaskan secara dinamis selama runtime

MEMORY MANAGEMENT

SISTEM OPERASI

OVERLAY

Apa itu Overlay ?

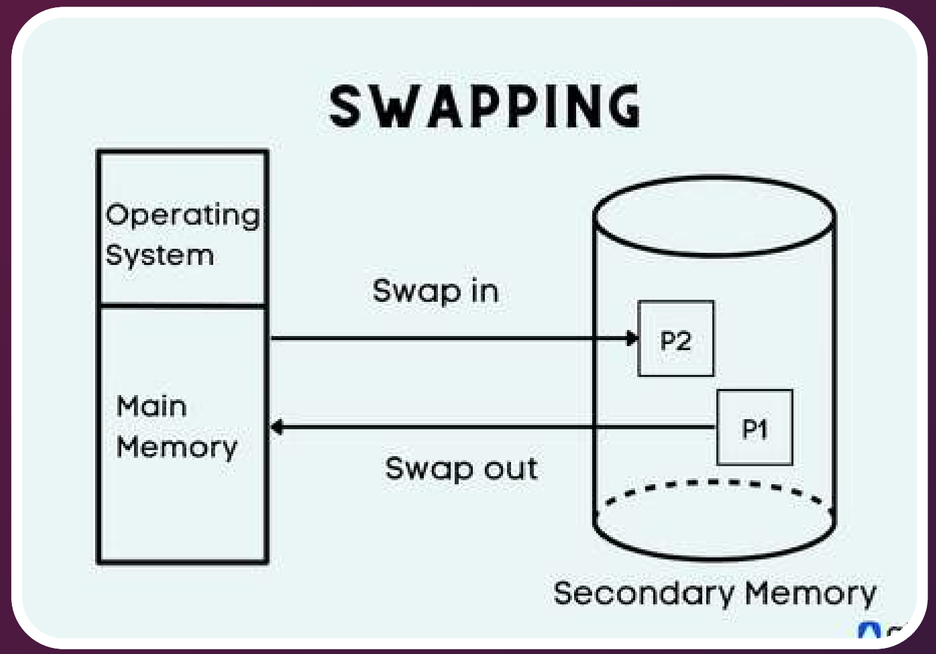
Overlay adalah proses membagi program yang besar menjadi bagian-bagian yang lebih kecil dan dapat dimuat dalam memori utama.

Konsep utama :

- Ide dari overlay yang disimpan di memori adalah hanya instruksi dan data yang diperlukan pada waktu tertentu. Bila instruksi lain yang diperlukan, maka akan diletakkan di tempat instruksi lama yang tidak diperlukan lagi.

MEMORY MANAGEMENT

SISTEM OPERASI



SKEMA SWAPPING

Apa Itu Konsep Swapping?

Teknik manajemen memori yang digunakan oleh sistem operasi untuk mengoptimalkan penggunaan memori fisik (RAM) dengan memindahkan data atau program yang tidak aktif dari memori utama ke area penyimpanan sementara, biasanya pada disk keras, yang dikenal sebagai memori swap atau swap space

Proses Swapping ini bekerja dengan memindahkan proses yang bersifat sementara dari memori utama ke disk, kemudian memanggilnya kembali ke memori utama ketika akan dieksekusi. Tujuannya untuk meningkatkan kinerja sistem, terutama saat multiprogramming dalam sistem time sharing. Metode Swapping ini juga berfungsi sebagai memori cadangan atau virtual, mendukung sistem untuk melakukan multitasking, dan membantu komputer menyimpan data atau proses yang berjalan sebelum masuk dalam mode hibernasi.

MEMORY MANAGEMENT

SISTEM OPERASI

SKEMA SWAPPING

Variasi Pada Swapping

1. Roll Out: Proses dengan prioritas lebih rendah dipindahkan dari RAM ke disk untuk memberi ruang bagi proses dengan prioritas lebih tinggi.

2. Roll In: Proses yang sebelumnya dipindahkan ke disk diambil kembali ke RAM ketika sudah ada cukup ruang

Dalam men-Swapping, diperlukan Backing store, ialah disk cepat yang cukup besar untuk mengakomodasi copy memori image pada semua user; menyediakan akses langsung ke memori image. Selain itu, Konteks waktu pergantian pada sistem swapping, lumayan tinggi. Total transfer time atau waktu pergantian dapat dihitung secara proporsional dari jumlah memori yang di swap.

Pertimbangan Kinerja:

1. Waktu transfer data antara memori dan disk sangat memengaruhi efisiensi swapping.
2. Latensi disk juga menjadi faktor penting dalam menentukan waktu yang dibutuhkan untuk proses swapping.
3. Terlalu sering swapping dapat memperlambat sistem karena overhead yang terkait dengan akses disk yang lebih lambat dibandingkan dengan akses RAM.

MEMORY MANAGEMENT

SISTEM OPERASI

TERIMA KASIH ATAS PERHATIANNYA

Sampai Jumpa

.....