



MODUL III CCS 210 SISTEM OPERASI

Judul	STRUKTUR SISTEM OPERASI	
Penyusun	Distribusi	Perkuliahan
Nixon Erzed	FASILKOM UNIVERSITAS ESA UNGGUL	Pertemuan – III ON LINE

Tujuan :

Mahasiswa mengenal dan memahami struktur sistem operasi

Materi:

- 1) Komponen Sistem Operasi
- 2) Pelayanan Sistem Operasi
- 3) Sistem Call,
- 4) Sistem Program
- 5) Struktur Sistem Operasi
 - DOS dan Unix
 - Sistem Monolitik
 - Multi layer model
 - Struktur THE
 - Model Client Server

Referensi :

1. Modern Operating System 3th Edition Andrew S Tanembaun 2009
2. Operating System, Internals and design Principles, William Stallings 7th Ed. 2012
3. Operating System Concepts, Abraham Silberschatz, 9th Ed, 2012
4. Sistem Operasi, Bambang Haryanto, Rev.5 2012
5. Arsitektur dan Organisasi Komputer, William Stalling, Prehalindo

KOMPONEN SISTEM OPERASI

Terdapat 5 komponen pokok sistem operasi, yaitu :

- Manajemen proses (inti)
- Manajemen memory utama
- Manajemen memory sekunder
- Manajemen I/O
- Manajemen File

Modul manajemen proses	Modul manajemen memory	Modul Manajemen Secondary Storage
	Modul manajemen File	Modul Manajemen I/O

1. *Manajemen Proses*

Sistem operasi memberikan tanggapan aktivitas-aktivitas yang berkaitan dengan manajemen/pengelolaan proses :

- a. Pembuatan/penciptaan atau penghapusan/pemusnahan proses yang dibuat oleh user atau sistem
- b. Pengelolaan proses-proses aktif (penjadwalan, blocked, timeout, dispatch)
- c. Suspensi (tunda) dan resume (pulihkan) proses (swap out, swap in)
- d. Kelengkapan mekanisme untuk sinkronisasi proses-proses
- e. Kelengkapan mekanisme untuk komunikasi antar proses
- f. Kelengkapan mekanisme untuk pengendalian deadlock.

2. *Manajemen Memori Utama*

Sistem operasi memberikan tanggapan aktivitas-aktivitas yang berkaitan dengan fungsi-fungsi pengelolaan memory :

- a. Mengelola informasi memori terpakai dan memory belum terpakai
- b. Menjaga dan memelihara bagian-bagian memori (mekanisme proteksi hw)
- c. Memutuskan proses-proses mana saja yang harus dipanggil ke memori jika masih ada ruang di memori.
- d. Mengalokasikan dan menddealokasikan ruang memori jika diperlukan.

3. Manajemen Memori Sekunder

Sistem operasi memberikan tanggapan aktivitas-aktivitas yang berkaitan dengan fungsi-fungsi pengelolaan memory sekunder:

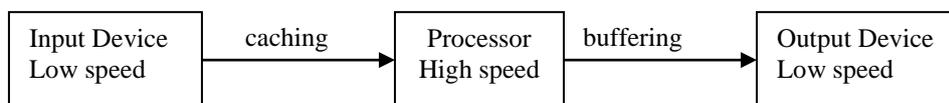
- a. Pengaturan ruang kosong dan primitive-primitif pengelolaan (format, partisi defragmentasi dll)
- b. Alokasi penyimpanan, dan pemetaan pemakaian ruang (FAT, NTFS dll), dealokasi ruang
- c. Penjadwalan disk (proses baca tulis)

4. Manajemen I/O

Sistem operasi memberikan tanggapan aktivitas-aktivitas yang berkaitan dengan fungsi-fungsi pengelolaan I/O :

- a. Sistem buffering-caching

Menyeimbangkan perbedaan speed Prosesor dan I/O device



- b. Antarmuka device-driver secara umum (driver docking/interface)
- c. Driver untuk [device/hardware-hardware tertentu](#).

5. Manajemen File (organisasi logika data)

Sistem operasi memberikan tanggapan aktivitas-aktivitas yang berkaitan dengan fungsi-fungsi pengelolaan I/O :

- a. Pembuatan dan penghapusan file
- b. Pembuatan dan penghapusan direktori
- c. Primitif-primitif (rutin dasar) yang mendukung untuk manipulasi file dan direktori (copy, rename, move, setting atribut dlsb)
- d. Pemetaan file ke memori sekunder
- e. Backup file ke media penyimpanan yang stabil (nonvolatil).

PELAYANAN SISTEM OPERASI

Sistem operasi harus dapat melayani programmer sehingga dapat melakukan pemrograman dengan mudah. Layanan pokok yang mesti disediakan sistem operasi meliputi :

- a. Eksekusi Program.
Sistem harus dapat memanggil program ke memori dan menjalankannya. Program tersebut harus dapat mengakhiri eksekusinya dalam bentuk normal atau abnormal (indikasi error).
- b. Operasi-operasi I/O.
Pada saat running program kemungkinan dibutuhkan I/O, mungkin berupa file atau peralatan I/O. Agar efisien dan aman, maka user tidak boleh mengontrol I/O secara langsung, dengan mendefinisikan instruksi-instruksi I/O pada mode monitor sehingga pengontrolan hanya dapat dilakukan oleh SO.
- c. Manipulasi sistem file.
Meliputi pembuatan, penghapusan, read dan write.
- d. Komunikasi.
Komunikasi dibutuhkan jika beberapa proses saling tukar-menukar informasi. Ada dua cara yang dapat dilakukan, yaitu :
 1. Tukar-menukar data oleh beberapa proses dalam satu komputer,
 2. Tukar-menukar data oleh beberapa proses dalam komputer yang berbeda melalui sistem jaringan.Komunikasi dilakukan dengan cara berbagi memori atau dengan cara pengiriman pesan.
- e. Mendeteksi kesalahan.
Untuk masing-masing kesalahan, sistem operasi harus memberikan aksi yang cocok agar komputasinya menjadi konsisten.

SYSTEM CALL

System call adalah layanan sistem operasi bagi pemrogram, yang memberikan tata cara pemanggilan layanan sistem operasi oleh sistem aplikasi. System call berupa rutin sistem operasi untuk keperluan yang spesifik. Pada dasarnya System call dapat dikelompokkan dalam 5 kategori sebagai berikut :

1. Kontrol Proses

Hal-hal yang dilakukan :

- Mengakhiri (end) dan membatalkan (abort);
- Mengambil (load) dan eksekusi (execute);
- Membuat dan mengakhiri proses;
- Menentukan dan mengeset atribut proses;
- Wait for time;
- Wait event, signal event;
- Mengalokasikan dan membebaskan memori.

Contoh: Sistem operasi pada MS-DOS menggunakan sistem single-tasking yang memiliki command interpreter yang akan bekerja pada saat start.

Karena singletasking, maka akan menggunakan metode yang sederhana untuk menjalankan program dan tidak akan membuat proses baru.

2. Manipulasi File

Hal-hal yang dilakukan:

- Membuat dan menghapus file;
- Membuka dan menutup file;
- Membaca, menulis, dan mereposisi file;
- Menentukan dan mengeset atribut file;

3. Manipulasi Device

Hal-hal yang dilakukan:

- Meminta dan membebaskan device;
- Membaca, menulis dan mereposisi file;
- Menentukan dan mengeset atribut device;

4. Informasi Lingkungan

Hal-hal yang dilakukan:

- Mengambil atau mengeset waktu atau tanggal;
- Mengambil atau mengeset sistem data;
- Mengambil atau mengeset proses, file atau atribut-atribut device;

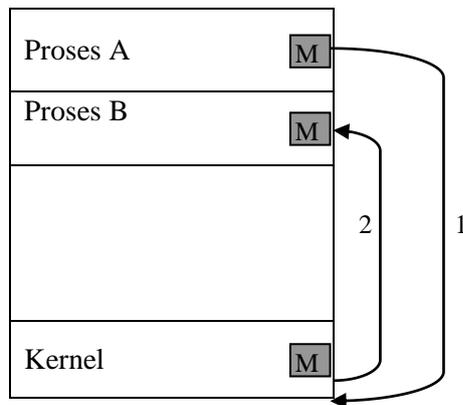
5. Komunikasi

Hal-hal yang dilakukan:

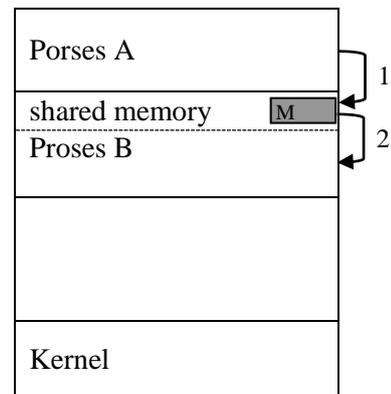
- Membuat dan menghapus sambungan komunikasi;
- Mengirim dan menerima pesan;
- Mentransfer satu informasi;

Ada 2 model komunikasi :

- a. Message-passing model. Informasi saling ditukarkan melalui fasilitas yang telah ditentukan oleh sistem operasi.



Message Passing



Shared Memory

- b. Shared-memory Model. Proses-proses menggunakan map memory untuk mengakses daerah-daerah di memori dengan proses-proses yang lain.

SISTEM PROGRAM

Sistem program adalah masalah yang relatif kompleks, namun dapat dibagi menjadi beberapa kategori, antara lain:

a. Manipulasi File.

Meliputi: membuat, menghapus, mengcopy, rename, print, dump, list pada file dan direktori.

b. Status Informasi.

Meliputi: tanggal, waktu (jam, menit, detik), penggunaan memori atau disk space, banyaknya user.

c. Modifikasi File.

Ada beberapa editor yang sanggup digunakan sebagai sarana untuk menulis atau memodifikasi file yang tersimpan dalam disk atau tape.

d. Bahasa Pemrograman yang mendukung.

Meliputi: compiler, assembler, dan interpreter untuk beberapa bahasa pemrograman (seperti : Fortran, Cobol, Pascal, Basic, C, dan LISP).

e. Pemanggilan dan Eksekusi Program.

Pada saat program dicompile, maka harus dipanggil ke memori untuk di eksekusi. Suatu sistem biasanya memiliki absolute loader, melokasikan loader, linkage editor, dan overlay loader. Juga dibutuhkan debugging sistem untuk bahasa tingkat tinggi.

f. Komunikasi.

Sebagai mekanisme untuk membuat hubungan virtual antar proses, user, dan sistem komputer yang berbeda.

g. Program-program aplikasi.

Sistem operasi harus menyokong program-program yang berguna untuk menyelesaikan permasalahan secara umum, atau membentuk operasi-operasi secara umum, seperti kompiler, pemformat teks, paket plot, sistem basis data, spreadsheet, paket analisis statistik, dan games.

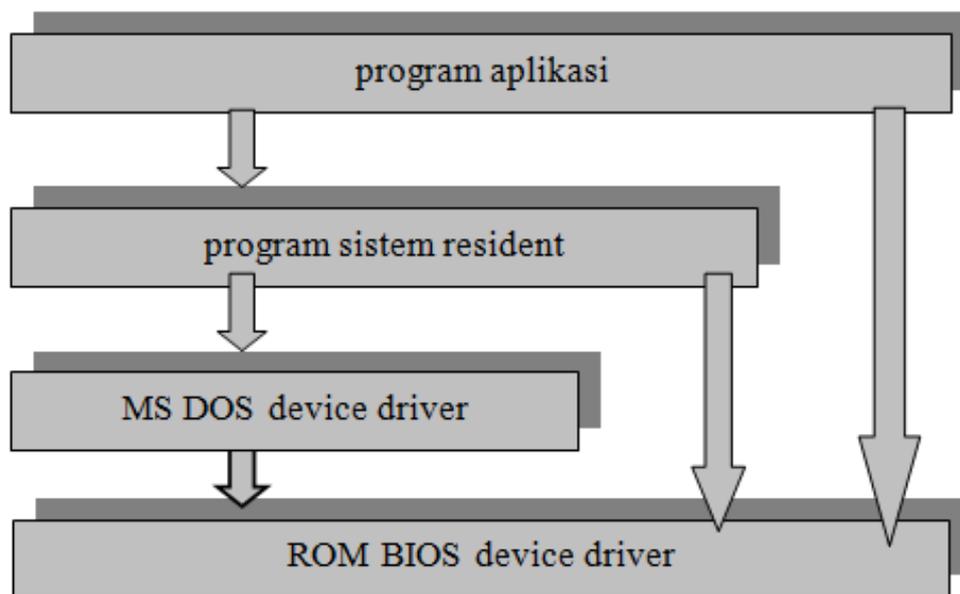
STRUKTUR SISTEM OPERASI

Sistem komputer modern yang semakin kompleks dan rumit memerlukan sistem operasi yang dirancang dengan sangat hati-hati agar dapat berfungsi secara optimum dan mudah untuk dimodifikasi.

1. Struktur Sistem Operasi Sederhana

Ada sejumlah sistem komersial yang tidak memiliki struktur yang cukup baik. Sistem operasi tersebut sangat kecil, sederhana dan memiliki banyak keterbatasan. Salah satu contoh sistem tersebut adalah MS-DOS. MS-DOS dirancang oleh orang-orang yang tidak memikirkan kepopuleran software tersebut. Sistem operasi tersebut terbatas pada hardware sehingga tidak terbagi menjadi modul-modul. Karena Intel 8088 tidak menggunakan dual-mode sehingga tidak ada proteksi hardware.

Sistem MS DOS



Contoh lainnya adalah UNIX. Sistem operasi UNIX juga terbatas pada hardware. UNIX hanya terdiri atas 2 bagian, yaitu Kernel dan program sistem. Kernel terbagi menjadi beberapa antarmuka dan device driver. Kernel ini berisi sistem file, penjadwalan CPU, manajemen memori, dan fungsi sistem operasi lainnya yang ada pada sistem call. Program sistem meminta bantuan kernel untuk memanggil fungsi-fungsi dalam kompilasi dan manipulasi file.

Struktur sistem UNIX.

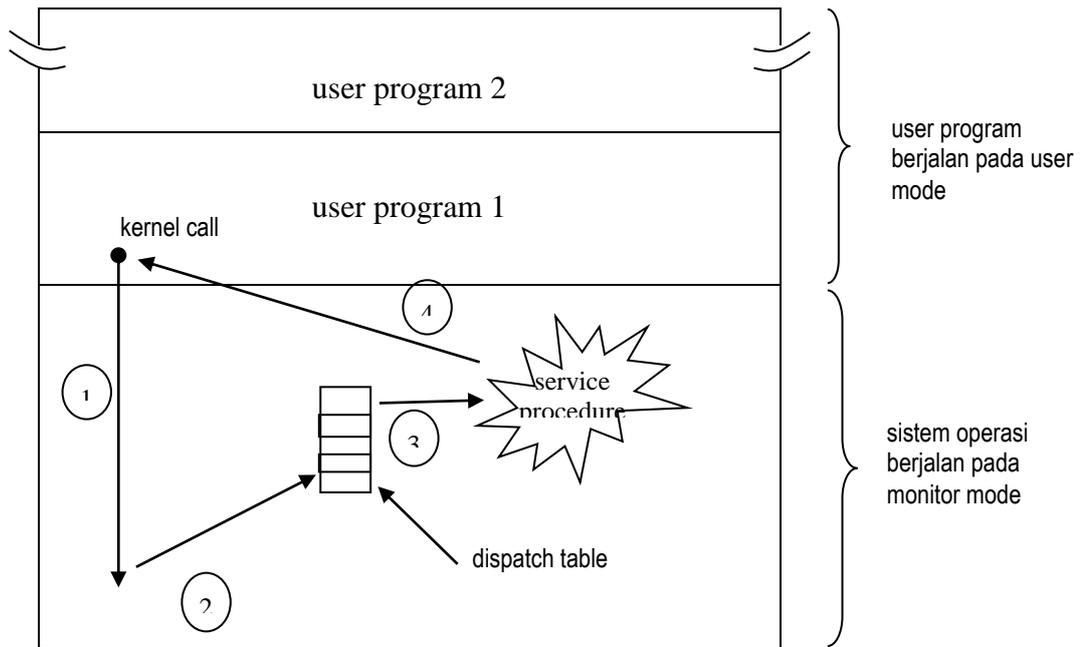
users		
shell dan perintah-perintah; compiler dan interpreter system libraries		
antar muka system call ke kernel		
sinyal; pengendali terminal; sistem I/O; terminal drivers	sistem file ; swapping; sistem blok I/O disk & tape drive	penjadwalan CPU; page replacement; demand paging; vittual memory;
terminal controlers; terminal	device controlers; disk & tapes	memory controlers memory fisik

2. MONOLITHIC SYSTEM

Pada dasarnya, sistem monolithic merupakan struktur sederhana yang dilengkapi dengan operasi dual-mode. Pelayanan (system call) yang diberikan oleh sistem operasi model ini dilakukan dengan cara mengambil sejumlah paramater pada tempat yang telah ditentukan sebelumnya, seperti register atau stack, dan kemudian mengeksekusi suatu instruksi trap tertentu pada monitor mode.

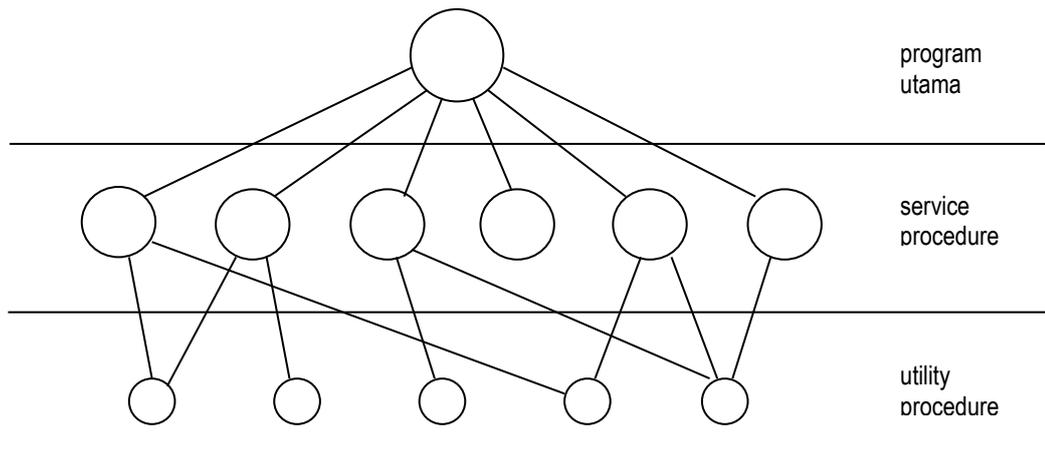
Perhatikan gambar dihalaman berikut :

1. User program melakukan "trap" pada kernel. Instruksi berpindah dari user-mode ke monitor mode dan mentransfer kontrol ke sistem operasi;
2. Sistem operasi mengecek parameter-parameter dari pemanggilan tersebut untuk menentukan system call mana yang memanggil;
3. Sistem operasi menunjuk ke suatu tabel yang berisi slot ke-k, yang menunjukkan system call k.
4. Setelah system call selesai mengerjakan tugasnya, kontrol akan dikembalikan pada user program.



Tatanan ini memberikan suatu struktur dasar dari sistem operasi sebagai berikut:

- a. Program utama yang meminta service procedure;
- b. Kumpulan service procedure yang dibawa oleh system call;
- c. Kumpulan utility procedure yang membantu service procedure.



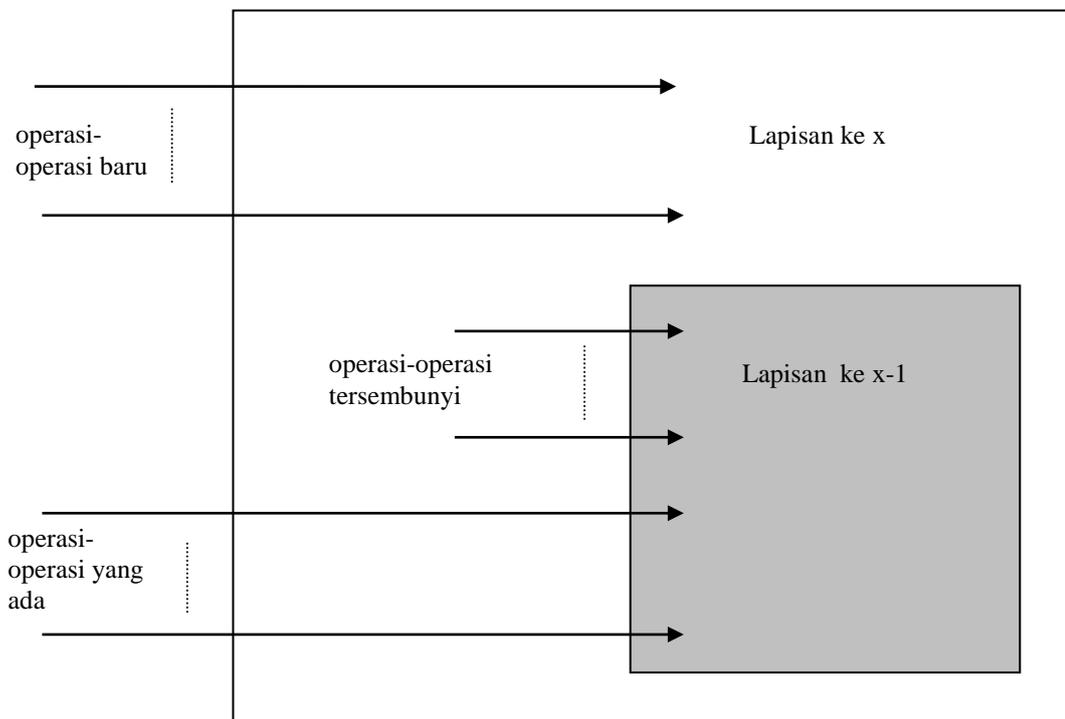
Pada model ini, tiap-tiap system call memiliki satu service procedure. Utility procedure mengerjakan segala sesuatu yang dibutuhkan oleh beberapa service procedure, seperti mengambil data dari user program.

3. PENDEKATAN TERLAPIS

Teknik pendekatan terlapis pada dasarnya dibuat dengan cara membentuk sistem operasi menjadi bentuk modular. Dengan menggunakan pendekatan top-down, semua fungsi ditentukan dan dibagi menjadi komponen-komponen. Modularisasi sistem dilakukan dengan cara memecah sistem operasi menjadi beberapa lapis (tingkat). Lapisan terendah (lapis-0) adalah hardware dan lapisan teratas (lapisan N) adalah user interface. Tiap-tiap lapisan diimplementasikan hanya dengan menggunakan operasi-operasi yang disediakan oleh lapisan yang lebih rendah (lapisan bawah akan memberikan layanan pada lapisan lebih atas).

Struktur ini dimaksudkan untuk mengurangi kompleksitas rancangan dan implementasi sistem operasi. Sistem operasi yang menggunakan sistem ini adalah: UNIX termodifikasi, THE, Venus dan OS2.

Lapisan-lapisan Sistem Operasi



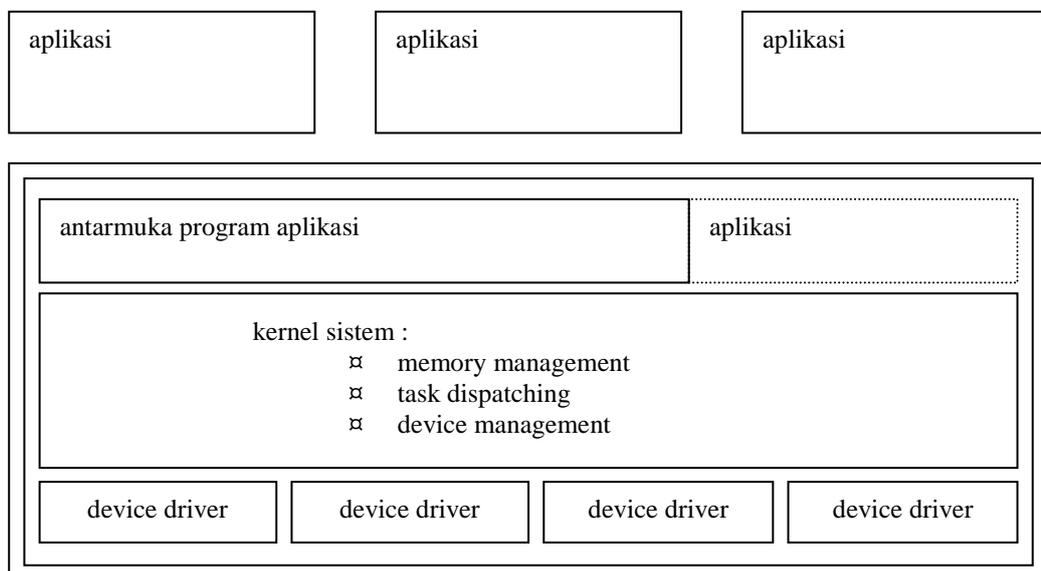
Struktur The

<i>Lapisan</i>	<i>Nama</i>
<i>Lapis ke 5</i>	<i>user program</i>
<i>Lapis ke 4</i>	<i>buffering untuk I/O device</i>
<i>Lapis ke 3</i>	<i>operator – console device driver</i>
<i>Lapis ke 2</i>	<i>menejemen memori</i>
<i>Lapis ke 1</i>	<i>Penjadwalan CPU</i>
<i>Lapis ke 0</i>	<i>hardware</i>

Struktur Venus

<i>Lapisan</i>	<i>Nama</i>
<i>Lapis ke 6</i>	<i>user program</i>
<i>Lapis ke 5</i>	<i>device driver dan scheduler</i>
<i>Lapis ke 4</i>	<i>virtual memory</i>
<i>Lapis ke 3</i>	<i>I/O channel</i>
<i>Lapis ke 2</i>	<i>Penjadwalan CPU</i>
<i>Lapis ke 1</i>	<i>instruksi interpreter</i>
<i>Lapis ke 0</i>	<i>hardware</i>

Struktur OS 2



4. MESIN VIRTUAL

Konsep dasar dari mesin virtual ini tidak jauh berbeda dengan pendekatan terlapis, hanya saja konsep ini memberikan sedikit tambahan berupa antarmuka yang menghubungkan hardware dengan kernel untuk tiap-tiap proses, gambar menunjukkan konsep tersebut.

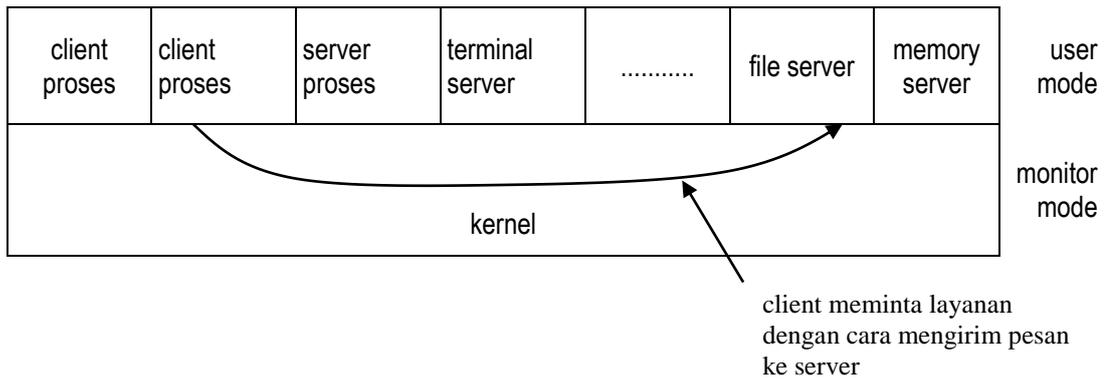
Meskipun konsep ini cukup baik, namun sulit untuk diimplementasikan, ingat bahwa sistem menggunakan metode dual-mode. Mesin virtual hanya dapat berjalan pada monitor-mode jika berupa sistem operasi, sedangkan mesin virtual itu sendiri berjalan dalam bentuk user-mode. Konsekuensinya, baik virtual monitor-mode maupun virtual user-mode harus dijalankan melalui physical user mode. Hal ini menyebabkan adanya transfer dari user-mode ke monitor-mode pada mesin nyata, yang juga akan menyebabkan adanya transfer dari virtual user-mode ke virtual monitor-mode pada mesin virtual.

Keuntungannya adalah bahwa konsep tersebut sepenuhnya melakukan proteksi, sehingga keamanan resource, terutama untuk resource-resource yang digunakan secara bersama-sama, akan terjamin. Contoh sistem operasi yang memakai mesin virtual adalah IBM VM system.

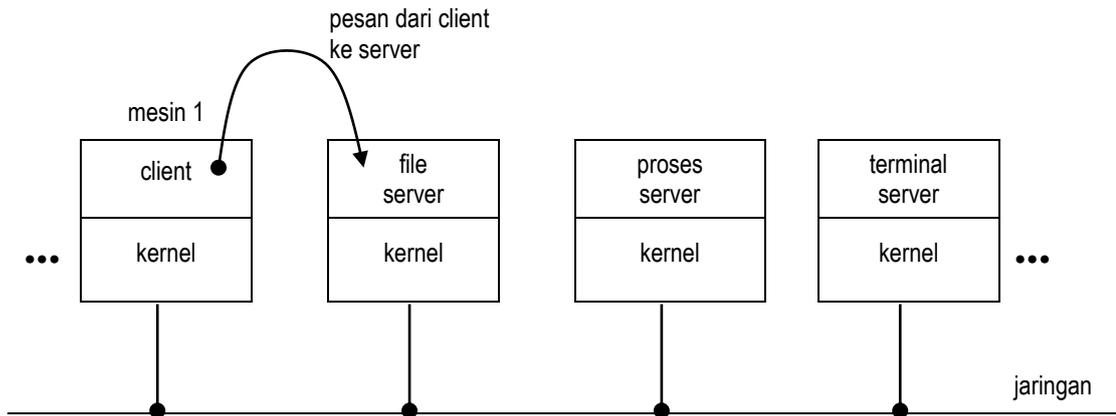
5. CLIENT SERVER MODEL

Trend dari sistem operasi modern adalah memindahkan kode ke lapisan yang lebih tinggi dan menghapusnya sebanyak mungkin dari sistem operasi sehingga akan meninggalkan kernel yang minimal. Konsep ini biasanya diimplementasikan dengan cara menjadikan fungsi-fungsi yang ada pada sistem operasi menjadi user proses. Jika suatu proses minta untuk dilayani, misalkan saja satu blok file, maka user proses (disini dinamakan: client proses) mengirim permintaan tersebut ke server proses. Server proses akan melayani permintaan tersebut kemudian mengirimkan jawabannya kembali. Pada model ini, seperti terlihat pada gambar, semua pekerjaan kernel ditekankan pada pengendalian komunikasi antara client dan server. Dengan membagi sistem operasi menjadi beberapa bagian, dimana tiap-tiap bagian mengendalikan satu segi sistem, seperti pelayanan file, pelayanan proses, pelayanan terminal, atau pelayanan memori, maka tiap-tiap bagian menjadi lebih sederhana dan dapat diatur. Selain itu, oleh karena semua server berjalan pada user-mode proses, dan bukan merupakan monitor mode, maka server tidak dapat mengakses hardware secara langsung. Akibatnya, jika terjadi kerusakan pada file server, maka pelayanan file akan terganggu. Namun hal ini tidak akan sampai mengganggu sistem lainnya.

Model Client Server



Model Client Server pada Sistem Terdistribusi



Keuntungan lain dari model client-server ini adalah dapat diadaptasikan pada sistem terdistribusi. Jika suatu client berkomunikasi dengan server dengan cara mengirimkan pesan, maka server tidak perlu tahu apakah pesan tersebut dikirim dari mesin itu sendiri (lokal) atau dikirim oleh mesin yang lain melalui jaringan.