



SISTEM OPERASI

Semester 4 Sistem Informasi

MODUL 3 STRUKTUR SISTEM OPERASI

STRUKTUR SO



1. Struktur Sederhana
2. Struktur Berlapis (*Layered*)
3. Microkernel

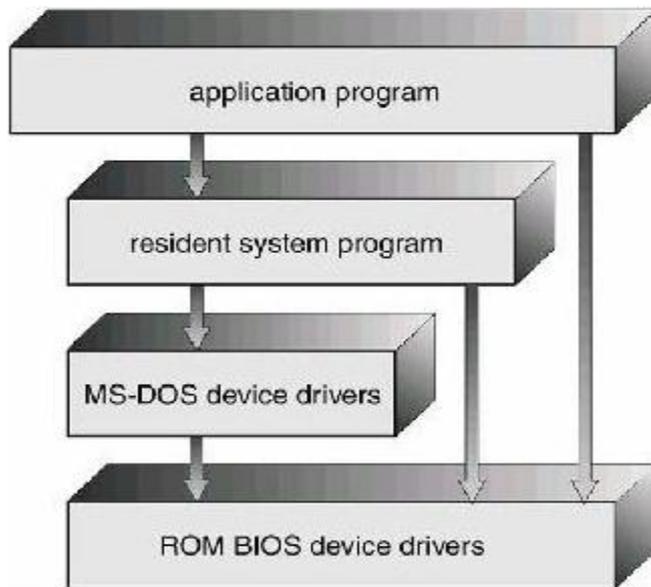
STRUKTUR SEDERHANA



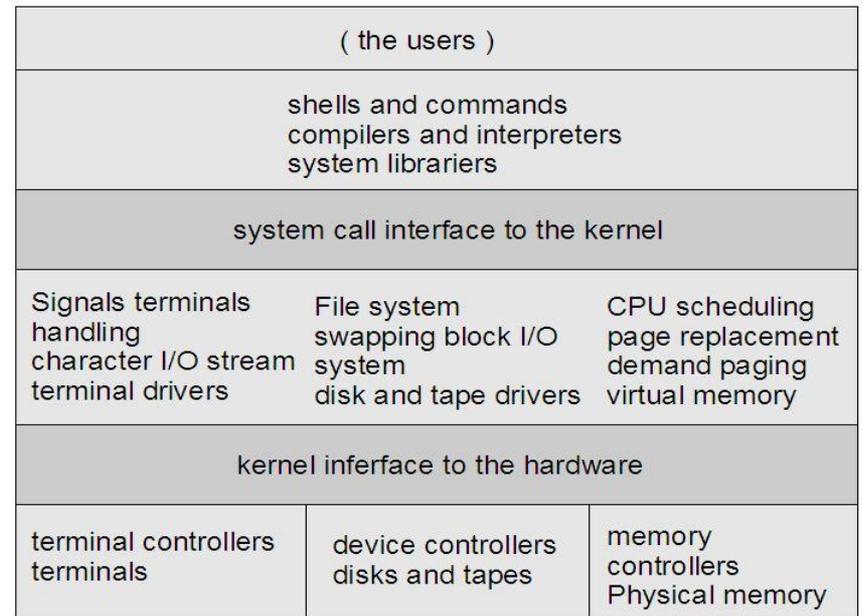
- Semua komponen sistem operasi bercampur
- Semua rutin sistem operasi dapat mengakses rutin yang lainnya
- Tidak ada pemisahan yang jelas antara aplikasi dan sistem operasi
 - ✓ Akibatnya program-program malware mudah memodifikasi dan merusak sistem operasi
 - ✓ Program aplikasi memiliki akses untuk memodifikasi bagian sistem operasi

Contoh : Ms-Dos, UNIX

STRUKTUR SEDERHANA



Struktur MS-DOS



Struktur UNIX

STRUKTUR SEDERHANA



Keunggulan

- Layanan dapat dilakukan sangat cepat karena terdapat di satu ruang alamat.

Kelemahan

- Pengujian dan penghilangan kesalahan sulit karena tidak dapat dipisahkan dan dilokalisasi.
- Sulit dalam menyediakan fasilitas pengamanan.
- Merupakan pemborosan bila setiap komputer harus menjalankan kernel monolitik sangat besar sementara sebenarnya tidak memerlukan seluruh layanan yang disediakan kernel.
- Kesalahan pemrograman satu bagian dari kernel menyebabkan matinya seluruh sistem.

SISTEM BERLAPIS (LAYERED)



- Rutin-rutin sistem operasi dikelompokkan dalam lapisan-lapisan (layered)
- Lapisan yang lebih bawah bertugas menangani detail operasi perangkat keras
- Lapisan atas fokus menangani interface user atau program aplikasi
- Rutin-rutin pada suatu lapisan hanya boleh menggunakan rutin-rutin lapisan di bawahnya

Contoh : THE System

SISTEM BERLAPIS (LAYERED)



Sistem Berlapis Tanenbaum

LAYER	FUNCTION
5	The Operator
4	User Programs
3	Input / Output Management
2	Operator-Process Communication
1	Memory and Drum Management
0	Processor Allocation and Multiprogramming

SISTEM BERLAPIS (LAYERED)



Menurut Tanenbaum dan Woodhull, sistem berlapis terdiri dari enam lapisan, yaitu:

- **Lapisan 0.** Mengatur alokasi prosesor, pertukaran antar proses ketika interupsi terjadi atau waktu habis. Lapisan ini mendukung dasar multi-programming pada CPU.
- **Lapisan 1.** Mengalokasikan ruang untuk proses di memori utama dan pada 512 kilo word drum yang digunakan untuk menahan bagian proses ketika tidak ada ruang di memori utama.
- **Lapisan 2.** Menangani komunikasi antara masing-masing proses dan operator console. Pada lapis ini masing-masing proses secara efektif memiliki operator console sendiri.
- **Lapisan 3.** Mengatur peranti M/K dan menampung informasi yang mengalir dari dan ke proses tersebut.
- **Lapisan 4.** Tempat program pengguna. Pengguna tidak perlu memikirkan tentang proses, memori, console, atau manajemen M/K.
- **Lapisan 5.** Merupakan operator sistem.

SISTEM BERLAPIS (LAYERED)



Stallings memberi model yang lebih detail, sebagai berikut:

- **Lapisan 1.** Berisi berbagai sirkuit elektronik, misal register, memory cells, dan logic gate.
- **Lapisan 2.** Berisi instruksi prosesor, misal instruksi aritmatika, instruksi transfer data, dsb.
- **Lapisan 3.** Penambahan konsep seperti prosedur/subrutin, maupun fungsi yang me-return nilai tertentu.
- **Lapisan 4.** Penambahan interrupt.
- **Lapisan 5.** Program sebagai sekumpulan instruksi yang dijalankan oleh prosesor.
- **Lapisan 6.** Berhubungan dengan secondary storage device, yaitu membaca/menulis head, track, dan sektor.
- **Lapisan 7.** Menciptakan alamat logika untuk proses. Mengatur hubungan antara main memory, virtual memory, dan secondary memory.
- **Lapisan 8.** Program sebagai sekumpulan instruksi yang dijalankan oleh prosesor.
- **Lapisan 9.** Berhubungan dengan secondary storage device, yaitu membaca/menulis head, track, dan sektor.
- **Lapisan 10.** Menciptakan alamat logika untuk proses. Mengatur hubungan antara main memory, virtual memory, dan secondary memory.
- **Lapisan 11.** Program sebagai sekumpulan instruksi yang dijalankan oleh prosesor.
- **Lapisan 12.** File adalah objek yang memiliki nama dan ukuran. Abstraksi dari lapisan 9.
- **Lapisan 13.** Menyediakan interface agar bisa berinteraksi dengan pengguna.

SISTEM BERLAPIS (LAYERED)



Level	nama	objek
13	shell	user programming environment
12	proses pengguna	proses pengguna
11	direktori	direktori
10	peranti	peranti eksternal
9	sistem berkas	berkas
8	komunikasi	pipa
7	memori virtual	segmen, halaman
6	penyimpanan sekunder lokal	blok data, saluran peranti
5	proses primitif	proses primitif, semafon, ready list
4	interupsi	program penanganan interupsi
3	prosedur	prosedur, call-stack, tampilan
2	set instruksi	stack, microprogram interpreter, scalar, dan array data
1	sirkuit elektronik	register, gerbang, bus, dll

Sistem Berlapis
Stallings

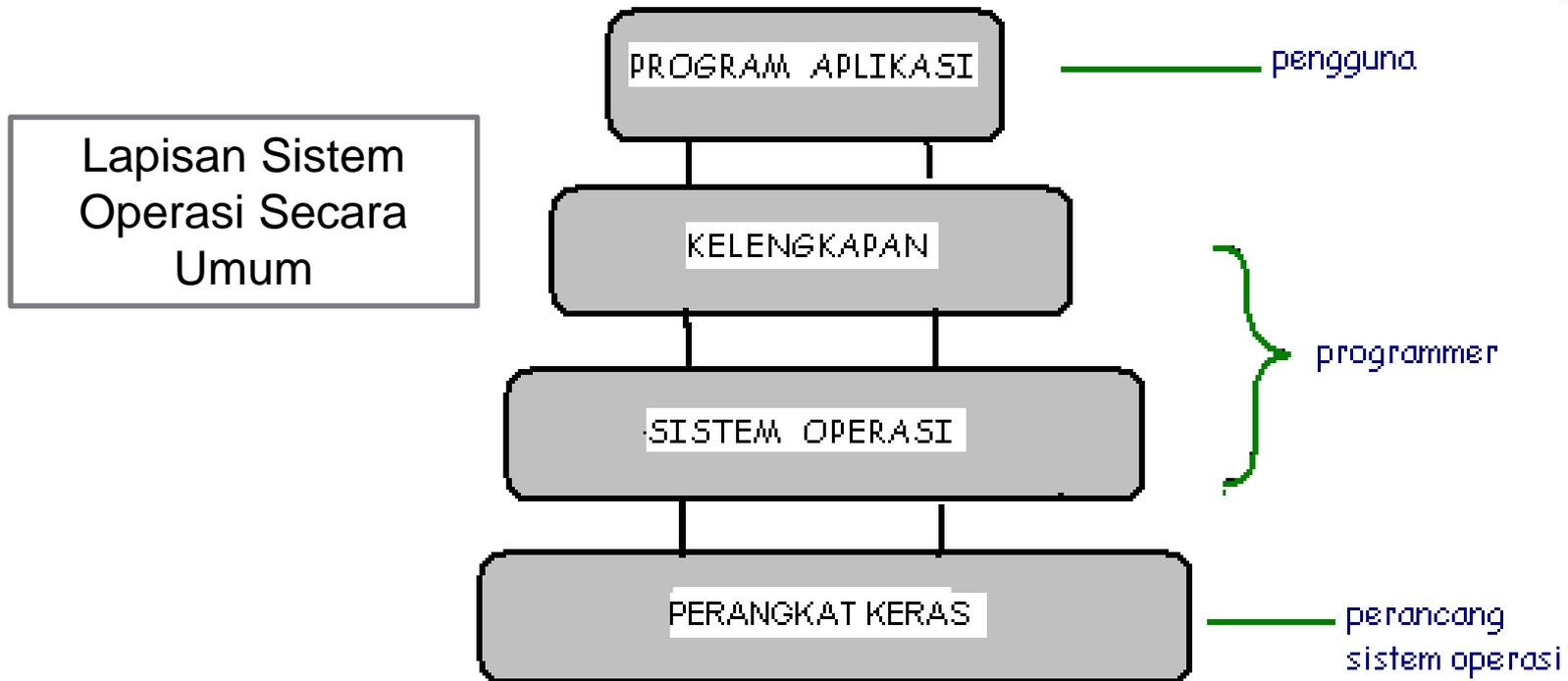
SISTEM BERLAPIS (LAYERED)



Dapat disimpulkan bahwa lapisan sistem operasi secara umum terdiri atas empat bagian, yaitu:

1. **Perangkat keras.** Lebih berhubungan kepada perancang sistem. Lapisan ini mencakup lapisan 0 dan 1 menurut Tanenbaum, dan level 1 sampai dengan level 4 menurut Stallings.
2. **Sistem operasi.** Lebih berhubungan kepada programmer. Lapisan ini mencakup lapisan 2 menurut Tanenbaum, dan level 5 sampai dengan level 7 menurut Stallings.
3. **Kelengkapan.** Lebih berhubungan kepada programmer. Lapisan ini mencakup lapisan 3 menurut Tanenbaum, dan level 8 sampai dengan level 11 menurut Stallings.
4. **Program aplikasi.** Lebih berhubungan kepada pengguna aplikasi komputer. Lapisan ini mencakup lapisan 4 dan lapisan 5 menurut Tanebaum, dan level 12 dan level 13 menurut Stallings.

SISTEM BERLAPIS (LAYERED)



SISTEM BERLAPIS (LAYERED)



Keuntungan: modularitas

- Mempermudah *debug* dan verifikasi sistem
- Lapisan pertama bisa *debug* tanpa mengganggu sistem yang lain

Kesulitan:

- Hanya bisa menggunakan lapisan dibawahnya
- Tidak efisien dibandingkan tipe yang lain

MIKROKERNEL



- Menyusun sistem operasi dengan menghapus semua komponen yang tidak esensial dari kernel, dan mengimplementasikannya sebagai sistem program dan level pengguna
- Fungsi utama: mendukung fasilitas komunikasi antara program klien dan bermacam-macam layanan yang juga berjalan di user-space

MIKROKERNEL



➤ Keuntungan:

- ketika layanan baru akan ditambahkan ke *user-space*, kernel tidak perlu dimodif
- OS lebih mudah ditempatkan pada suatu desain perangkat keras ke desain lainnya
- mendukung keamanan reliabilitas lebih

➤ Contoh sistem operasi:

- Tru64 UNIX, MacOSX, QNX

VIRTUAL MACHINE



- Mesin virtual mengambil pendekatan terlapis sebagai kesimpulan logis. Mesin virtual memperlakukan hardware dan sistem operasi seolah-olah berada pada level yang sama sebagai perangkat keras.
- Pendekatan Mesin virtual menyediakan sebuah antarmuka yang identik dengan underlying bare hardware.
- Sistem Operasi membuat ilusi dari banyak proses, masing-masing dieksekusi pada prosesoranya sendiri dengan virtual memorinya sendiri.
- VM dibuat dengan pembagian sumber daya oleh komputer fisik

VIRTUAL MACHINE



Keuntungan Penggunaan Virtual Machine

- Keamanan bukanlah masalah
 - VM mempunyai perlindungan lengkap pada berbagai sistem sumber daya
 - Tidak ada pembagian sumber daya secara langsung. Pembagian disk mini dan jaringan diimplementasikan pada perangkat lunak
- VM sistem adalah kendaraan yang “sempurna” untuk penelitian dan pengembangan sistem operasi
 - Dengan VM perubahan suatu bagian tidak mempengaruhi komponen yang lain

VIRTUAL MACHINE



Kerugian Penggunaan Virtual Machine

- VM sulit di implementasikan karena banyak syarat yang dibutuhkan untuk menyediakan duplikat yang tepat dari *underlying machine*
 - Harus punya *virtual-user mode* dan *virtual-monitor mode* yang keduanya berjalan di *physical mode*. Akibatnya, saat instruksi yang hanya membutuhkan *virtual monitor mode* dijalankan, register berubah dan bisa berefek pada *virtual user mode*, bahkan bisa me-restart VM
- Waktu yang dibutuhkan I/O bisa lebih cepat(karena ada *spooling*), tapi bisa lebih lambat(karena *diinterpreted*)



TERIMA KASIH