

BAB 4. Sistem Operasi Terdistribusi

4.1. Sistem Operasi

Pengertian Sistem Operasi

Sistem operasi (*Operating System* atau *OS*) adalah perangkat lunak sistem yang bertugas untuk melakukan kontrol dan manajemen perangkat keras serta operasi-operasi dasar sistem, termasuk menjalankan software aplikasi seperti program-program pengolah kata dan browser web.

Secara umum, Sistem Operasi adalah software pada lapisan pertama yang ditaruh pada memori komputer pada saat komputer dinyalakan. Sedangkan software-software lainnya dijalankan setelah Sistem Operasi berjalan, dan Sistem Operasi akan melakukan layanan inti umum untuk software-software itu. Layanan inti umum tersebut seperti akses ke disk, manajemen memori, skeduling task, dan antar-muka user. Sehingga masing-masing software tidak perlu lagi melakukan tugas-tugas inti umum tersebut, karena dapat dilayani dan dilakukan oleh Sistem Operasi. Bagian kode yang melakukan tugas-tugas inti dan umum tersebut dinamakan dengan "kernel" suatu Sistem Operasi.

Sistem Operasi secara umum terdiri dari beberapa bagian:

1. Mekanisme [Boot](#), yaitu meletakkan kernel ke dalam memory
2. [Kernel](#), yaitu inti dari sebuah Sistem Operasi
3. *Command Interpreter* atau *shell*, yang bertugas membaca input dari pengguna
4. Pustaka-pustaka, yaitu yang menyediakan kumpulan fungsi dasar dan standar yang dapat dipanggil oleh aplikasi lain
5. Driver untuk berinteraksi dengan hardware eksternal, sekaligus untuk mengontrol mereka.

Komponen Sistem Operasi

Komponen sistem operasi terdiri dari:

A. Manajemen Proses

Proses adalah keadaan ketika sebuah program sedang di eksekusi. Sebuah proses membutuhkan beberapa sumber daya untuk menyelesaikan tugasnya. sumber daya tersebut dapat berupa *CPU time*, memori, berkas-berkas, dan perangkat-perangkat I/O. Sistem operasi bertanggung jawab atas aktivitas-aktivitas yang berkaitan dengan manajemen proses seperti:

- Pembuatan dan penghapusan proses pengguna dan sistem proses.
- Menunda atau melanjutkan proses.
- Menyediakan mekanisme untuk proses sinkronisasi.
- Menyediakan mekanisme untuk proses komunikasi.
- Menyediakan mekanisme untuk penanganan *deadlock*.

B. Manajemen Memori Utama

Memori utama atau lebih dikenal sebagai memori adalah sebuah *array* yang besar dari *word* atau *byte*, yang ukurannya mencapai ratusan, ribuan, atau bahkan jutaan. Setiap *word* atau *byte* mempunyai alamat tersendiri. Memori Utama berfungsi sebagai tempat penyimpanan yang akses datanya digunakan oleh CPU

atau perangkat I/O. Memori utama termasuk tempat penyimpanan data yang sementara (*volatile*), artinya data dapat hilang begitu sistem dimatikan.

Sistem operasi bertanggung jawab atas aktivitas-aktivitas yang berkaitan dengan manajemen memori seperti:

- Menjaga *track* dari memori yang sedang digunakan dan siapa yang menggunakannya.
- Memilih program yang akan di-*load* ke memori.
- Mengalokasikan dan meng-*dealokasikan* ruang memori sesuai kebutuhan.

C. Manajemen Berkas

Berkas adalah kumpulan informasi yang berhubungan sesuai dengan tujuan pembuat berkas tersebut. Berkas dapat mempunyai struktur yang bersifat hirarkis (direktori, volume, dll.). Sistem operasi bertanggung-jawab:

- Pembuatan dan penghapusan berkas.
- Pembuatan dan penghapusan direktori.
- Mendukung manipulasi berkas dan direktori.
- Memetakan berkas ke *secondary storage*.
- Mem-backup berkas ke media penyimpanan yang permanen (*non-volatile*)

D. Manajemen Sistem I/O

Sering disebut *device manager*. Menyediakan "*device driver*" yang umum sehingga operasi I/O dapat seragam (membuka, membaca, menulis, menutup). Contoh: pengguna menggunakan operasi yang sama untuk membaca berkas pada *hard-disk*, CD-ROM dan *floppy disk*.

Komponen Sistem Operasi untuk sistem I/O:

- *Buffer*: menampung sementara data dari/ ke perangkat I/O.
- *Spooling*: melakukan penjadualan pemakaian I/O sistem supaya lebih efisien (antrian dsb.).
- Menyediakan *driver* untuk dapat melakukan operasi "rinci" untuk perangkat keras I/O tertentu.

E. Manajemen Penyimpanan Sekunder

Data yang disimpan dalam memori utama bersifat sementara dan jumlahnya sangat kecil. Oleh karena itu, untuk menyimpan keseluruhan data dan program komputer dibutuhkan *secondary-storage* yang bersifat permanen dan mampu menampung banyak data. Contoh dari *secondary-storage* adalah *harddisk*, disket, dll. Sistem operasi bertanggung-jawab atas aktivitas-aktivitas yang berkaitan dengan *disk-management* seperti: *free-space management*, alokasi penyimpanan, penjadualan disk.

F. Sistem Proteksi

Proteksi mengacu pada mekanisme untuk mengontrol akses yang dilakukan oleh program, prosesor, atau pengguna ke sistem sumber daya.

Mekanisme proteksi harus:

- membedakan antara penggunaan yang sudah diberi izin dan yang belum.
- *specify the controls to be imposed*.
- *provide a means of enforcement*.

G. Command-Interpreter System

Sistem Operasi menunggu instruksi dari pengguna (*command driven*). Program yang membaca instruksi dan mengartikan *control statements* umumnya disebut:

control-card interpreter, command-line interpreter, dan UNIX shell. Command-Interpreter System sangat bervariasi dari satu sistem operasi ke sistem operasi yang lain dan disesuaikan dengan tujuan dan teknologi *I/O devices* yang ada. Contohnya: *CLI, Windows, Pen-based (touch)*, dan lain-lain.

H. Jaringan

Sistem terdistribusi adalah sekumpulan prosesor yang tidak berbagi memori atau *clock*. Tiap prosesor mempunyai memori sendiri. Prosesor-prosesor tersebut terhubung melalui jaringan komunikasi. Sistem terdistribusi menyediakan akses pengguna ke bermacam sumber-daya sistem.

- *Increased data availability.*
- Enhanced reliability.
- *Computation speed-up.*
- *Increased data availability.*
- Enhanced reliability.

4.2. Sistem Operasi Terdistribusi

Pengertian Sistem Operasi Terdistribusi

Sistem operasi terdistribusi adalah salah satu implementasi dari sistem terdistribusi, di mana sekumpulan komputer dan prosesor yang heterogen terhubung dalam satu jaringan. Koleksi-koleksi dari objek-objek ini secara tertutup bekerja secara bersama-sama untuk melakukan suatu tugas atau pekerjaan tertentu. Tujuan utamanya adalah untuk memberikan hasil secara lebih, terutama dalam:

- *file system*
- *name space*
- Waktu pengolahan
- Keamanan
- Akses ke seluruh *resources*, seperti prosesor, memori, penyimpanan sekunder, dan perangkat keras.

Sistem operasi terdistribusi bertindak sebagai sebuah infrastruktur/rangka dasar untuk *network-transparent resource management*. Infrastruktur mengatur *low-level resources* (seperti Processor, memory, *network interface* dan *peripheral device* yang lain) untuk menyediakan sebuah *platform* untuk pembentukan/penyusunan *higher-level resources* (seperti *Spreadsheet, electronic mail messages, windows*).

Sistem Operasi Jaringan Versus Sistem Operasi Terdistribusi

Suatu sistem operasi terdistribusi yang sejati adalah yang berjalan pada beberapa buah mesin, yang tidak melakukan sharing memori, tetapi terlihat bagi user sebagai satu buah komputer single. Contoh dari sistem seperti ini adalah Amoeba.

Sistem operasi terdistribusi berbeda dengan sistem operasi jaringan. Untuk dapat membedakannya, sistem operasi jaringan memiliki ciri-ciri sebagai berikut:

- a. Tiap komputer memiliki sistem operasi sendiri
- b. Tiap personal komputer memiliki sistem file sendiri, di mana data-data disimpan
- c. Sistem operasi tiap komputer dapat berbeda-beda atau heterogen
- d. Pengguna harus memikirkan keberadaan komputer lain yang terhubung, dan harus mengakses, biasanya menggunakan remote login (telnet)
- e. File system dapat digunakan dengan dukungan NFS

Manfaat Sistem Operasi Terdistribusi

Sistem operasi terdistribusi memiliki manfaat dalam banyak sistem dan dunia komputasi yang luas. Manfaat-manfaat ini termasuk dalam *sharing resource*, waktu komputasi dan komunikasi.

1. Shared Resource

Walaupun perangkat sekarang sudah memiliki kemampuan yang cepat dalam proses-proses komputasi, atau misal dalam mengakses data, tetapi pengguna masih saja menginginkan sistem berjalan dengan lebih cepat. Apabila *hardware* terbatas, kecepatan yang diinginkan user dapat di atasi dengan menggabung perangkat yang ada dengan sistem DOS.

2. Manfaat Komputasi

Salah satu keunggulan sistem operasi terdistribusi ini adalah bahwa komputasi berjalan dalam keadaan paralel. Proses komputasi ini dipecah dalam banyak titik, yang mungkin berupa komputer pribadi, prosesor tersendiri, dan kemungkinan perangkat prosesor-prosesor yang lain. Sistem operasi terdistribusi ini bekerja baik dalam memecah komputasi ini dan baik pula dalam mengambil kembali hasil komputasi dari titik-titik *cluster* untuk ditampilkan hasilnya.

3. Reliabilitas

Fitur unik yang dimiliki oleh DOS ini adalah reliabilitas. Berdasarkan *design* dan implementasi dari *design* sistem ini, maka hilangnya satu node tidak akan berdampak terhadap integritas sistem. Hal ini berbeda dengan PC, apabila ada salah satu *hardware* yang mengalami kerusakan, maka sistem akan berjalan tidak seimbang, bahkan sistem bisa tidak dapat berjalan atau mati.

4. Komunikasi

Sistem operasi terdistribusi biasanya berjalan dalam jaringan dan biasanya melayani koneksi jaringan. Sistem ini biasanya digunakan user untuk proses *networking*. Users dapat saling bertukar data, atau saling berkomunikasi antara titik baik secara LAN maupun WAN.

Hardware Sistem Operasi Terdistribusi

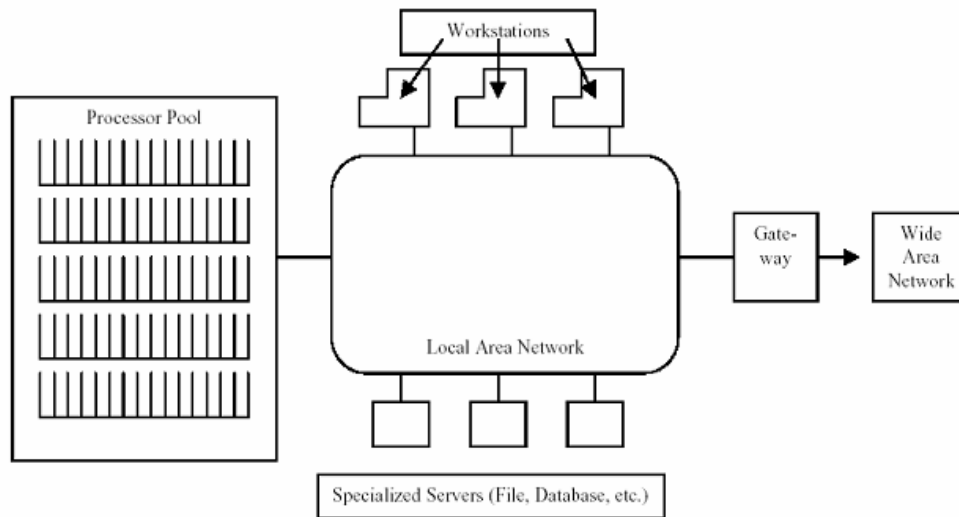
Sistem operasi terdistribusi, yang saat ini akan dibahas sebagai titik tolak adalah Amoeba, yang saat ini banyak digunakan sebagai salah satu implementasi dari sistem operasi terdistribusi itu sendiri. Sistem Amoeba ini tumbuh dari bawah hingga akhirnya tumbuh menjadi sistem operasi terdistribusi.

Sistem operasi terdistribusi pada umumnya memerlukan hardware secara spesifik. Komponen utama dalam sistem ini adalah : workstation, LAN, gateway, dan processor pool, seperti yang diilustrasikan pada gambar di atas. Workstation atau komputer personal mengeksekusi proses yang memerlukan interaksi dari user seperti text editor atau manager berbasis window. Server khusus memiliki fungsi untuk melakukan tugas yang spesifik. Server ini mengambil alih proses yang memerlukan I/O yang khusus dari lirikan disk. Gateway berfungsi untuk mengambil alih tugas untuk terhubung ke jaringan WAN.

Prosesor pool mengambil alih semua proses yang lain. Tiap unit ini biasanya terdiri dari prosesor, memori lokal, dan koneksi jaringan. Tiap prosesor mengerjakan satu buah proses sampai prosesor yang tidak digunakan habis. Untuk selanjutnya proses yang lain berada dalam antrian menunggu proses yang lain selesai.

Inilah keunggulan sistem operasi terdistribusi dalam hal reliabilitas. Apabila ada satu unit pemroses yang mati, maka proses yang dialokasikan harus di restart, tetapi integritas sistem tidak akan terganggu, apabila proses deteksi berjalan dengan baik. Desain sistem ini memungkinkan untuk 10 sampai 100 prosesor. Spesifikasi perangkat keras yang harus disediakan pada tiap cluster minimalnya adalah :

- File server: 16 MB RAM, 300MB HD, Ethernet card.
- Workstation: 8 MB RAM, monitor, keyboard, mouse
- Pool processor: 4 MB RAM, 3.5" floppy drive



Gambar Design Sistem Operasi Amoeba

Arsitektur Software

Sistem operasi terdistribusi sejati memiliki arsitektur software yang unik. Arsitektur software ini dikarakterkan dalam objek di dalam hubungan antara klien dan server. Proses-proses yang terjadi di klien menggunakan remote procedure yang memanggil dan mengirimkan request ke server untuk memproses data atau objek yang dibawa. Tiap objek yang dibawa memiliki karakteristik yang disebut sebagai kapabilitas. Kapabilitas ini besarnya adalah 128 bits. 48 bits pertama menunjukkan servis mana yang memiliki objek tersebut. 24 bits berikutnya adalah nomor dari objek. 8 bits berikutnya menampilkan operasi yang diijinkan terhadap objek yang bersangkutan. Dan 48 bits terakhir merupakan "check field" yang merupakan field yang telah terenkripsi agar tidak dapat dimodifikasi oleh proses yang lain.

Operasi diselesaikan oleh RPC (remote procedure calls) yang dibuat oleh klien di dalam proses yang kecil dan ringan. Proses dengan tipe seperti ini memiliki bidang alamat sendiri, dan bisa saja memiliki satu atau lebih hubungan. Hubungan ini ketika berjalan memiliki program counter dan stack sendiri, tetapi dapat saling berbagi kode dan data antara hubungan lain di dalam proses. Ada 3 macam basis panggilan sistem yang dapat digunakan dalam proses yang dimiliki user, yaitu `do_operation`, `get_request`, dan `send_reply`. Bagian yang pertama mengirimkan pesan ke server, setelah proses memblok sampai server mengirimkan balasan.

Server menggunakan panggilan sistem ke dua untuk mengindikasikan bahwa server akan menerima pesan pada port tertentu. Server juga menggunakan panggilan sistem ke tiga untuk mengirimkan kembali informasi ke proses yang dipanggil.

Dengan dibangun dari perintah sistem yang primitif, maka sistem ini menjadi antarmuka untuk program aplikasi. Hal ini diselesaikan oleh tingkat dari pengarahan yang mengizinkan pengguna untuk berfikir terhadap struktur ini sebagai objek dan operasi-operasi terhadap objek ini. Berhubungan dengan objek-objek adalah class. Kelas dapat berisi kelas yang lain dan juga hierarki secara alami. Pewarisan membuat antarmuka objek untuk implementasi manipulasi objek seperti menghapus, membaca, menulis, dan sebagainya.

Jenis Sistem Operasi Terdistribusi

Ada berbagai macam sistem operasi terdistribusi yang saat ini beredar dan banyak digunakan. Keanekaragaman sistem ini dikarenakan semakin banyaknya sistem yang bersifat opensource sehingga banyak yang membangun OS sendiri sesuai dengan kebutuhan masing-masing, yang merupakan pengembangan dari OS opensource yang sudah ada. Beberapa contoh dari sistem operasi terdistribusi ini diantaranya :

- **Amoeba (Vrije Universiteit).** Amoeba adalah sistem berbasis mikro-kernel yang tangguh yang menjadikan banyak workstation personal menjadi satu sistem terdistribusi secara transparan. Sistem ini sudah banyak digunakan di kalangan akademik, industri, dan pemerintah selama sekitar 5 tahun.
- **Angel (City University of London).** Angel didesain sebagai sistem operasi terdistribusi yang paralel, walaupun sekarang ditargetkan untuk PC dengan jaringan berkecepatan tinggi. Model komputasi ini memiliki manfaat ganda, yaitu memiliki biaya awal yang cukup murah dan juga biaya incremental yang rendah. Dengan memproses titik-titik di jaringan sebagai mesin single yang bersifat shared memory, menggunakan teknik distributed virtual shared memory (DVSM), sistem ini ditujukan baik bagi yang ingin meningkatkan performa dan menyediakan sistem yang portabel dan memiliki kegunaan yang tinggi pada setiap platform aplikasi.
- **Chorus (Sun Microsystems).** CHORUS merupakan keluarga dari sistem operasi berbasis mikro-kernel untuk mengatasi kebutuhan komputasi terdistribusi tingkat tinggi di dalam bidang telekomunikasi, internetworking, sistem tambahan, realtime, sistem UNIX, supercomputing, dan kegunaan yang tinggi. Multiserver CHORUS/MiX merupakan implementasi dari UNIX yang memberi kebebasan untuk secara dinamis mengintegrasikan bagian-bagian dari fungsi standar di UNIX dan juga service dan aplikasi-aplikasi di dalamnya.
- **GLUnix (University of California, Berkeley).** Sampai saat ini, workstation dengan modem tidak memberikan hasil yang baik untuk membuat eksekusi suatu sistem operasi terdistribusi dalam lingkungan yang shared dengan aplikasi yang berurutan. Hasil dari penelitian ini adalah untuk menempatkan resource untuk performa yang lebih baik baik untuk aplikasi paralel maupun yang seri/berurutan. Untuk merealisasikan hal ini, maka sistem operasi harus menjadwalkan pencabangan dari program paralel, mengidentifikasi idle resource di jaringan, mengizinkan migrasi proses untuk mendukung keseimbangan loading, dan menghasilkan tumpuan untuk antar proses komunikasi

Referensi:

1. http://id.wikipedia.org/wiki/Sistem_operasi
2. vlsn.org, Komponen Sistem Operasi,
<http://bebas.vlsn.org/v06/Kuliah/SistemOperasi/BUKU/SistemOperasi-4.X-1/ch05.html#c20501>
3. Wahyu Wijanarko, Sistem Operasi Terdistribusi,
<http://ilmukomputer.com/2006/08/20/sistem-operasi-terdistribusi/>