

SISTEM BERKAS

KONSEP DASAR SISTEM BERKAS

Sistem berkas merupakan mekanisme penyimpanan *on-line* serta untuk akses, baik data maupun program yang berada dalam sistem operasi.

Terdapat dua bagian penting dalam sistem berkas, yaitu:

- Kumpulan berkas, sebagai tempat penyimpanan data
- Struktur direktori, yang mengatur dan menyediakan informasi mengenai seluruh berkas dalam sistem.

Sistem Berkas

Sebuah sistem berkas sangat membantu para *programmer* untuk memungkinkan mereka mengakses berkas, tanpa memperhatikan *detail* dari karakteristik dan waktu penyimpanan. Sistem berkas ini juga yang mengatur direktori, *device access* dan *buffer*.

Tugas dari Sistem Berkas

- Memelihara direktori dari identifikasi berkas dan lokasi informasi.
- Menentukan jalan (*pathway*) bagi aliran data antara *main memory* dan alat penyimpanan sekunder.
- Mengkoordinasi komunikasi antara CPU dan alat penyimpanan sekunder, dan sebaliknya.
- Menyiapkan berkas penggunaan input atau output.
- Mengatur berkas bila penggunaan input atau output telah selesai.

BERKAS DAN AKSES

Sistem penyimpanan, pengelolaan dan penyimpanan data pada alat penyimpanan eksternal. Pada berkas dan akses penyimpanan data dilakukan secara fisik.

Berkas

- Kumpulan dari record-record yang saling berhubungan.
- Kumpulan dari statement-statement yang saling berhubungan
- Kumpulan informasi terkait yang diberi nama dan direkam pada penyimpanan sekunder.

KLASIFIKASI DATA

- **Kelompok Data Tetap**

Kelompok data yang tidak mengalami perubahan, paling tidak dalam kurun waktu yang lama.

Contoh: Data Pribadi Mahasiswa, Data Mata Kuliah.

- **Kelompok Data Tidak Tetap**

Kelompok data yang secara rutin mengalami perubahan.

Contoh: Data Rencana Studi Mahasiswa.

- **Kelompok Data yang bertambah menurut kurun waktu**

Kelompok data ini biasanya merupakan data akumulasi dari kelompok data tetap dan data tidak tetap.

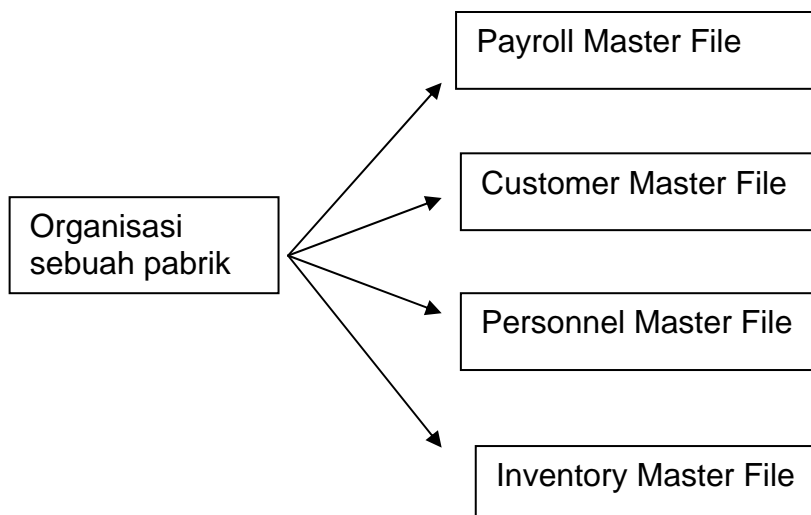
Contoh: Data Transkrip, Master Nilai.

KLASIFIKASI BERKAS

- **Master File (Berkas Induk)**

File yang berisi data yang relatif tetap

Contoh:



Gambar 1. Contoh Master File

Ada 2 jenis Master File

1. Reference Master File

File yang berisi record yang tidak berubah / jarang berubah

Contoh:

File pelanggan yang berisi field:
nomor rekening, nama dan alamat

2. Dynamic Master File

File yang berisi record yang terus menerus berubah dalam kurun waktu tertentu atau berdasarkan suatu peristiwa transaksi.

Contoh:

- File stock barang
- File pemesanan tempat duduk

- **Transaction File (Berkas Transaksi)**

File yang berisi record-record yang akan memperbaharui atau meng-*update* record-record yang ada pada master file.

Meng-*update* dapat berupa: penambahan record, penghapusan dan perbaikan record.

- **Report File (Berkas Laporan)**

File yang berisi data yang dibuat untuk laporan atau keperluan *user*.

File tersebut dapat dicetak pada kertas printer atau hanya ditampilkan di layar.

- **Work File (Berkas Kerja)**

Merupakan file sementara dalam sistem. Suatu *work file* merupakan alat untuk melewati data yang dibuat oleh sebuah program ke program lain. Biasanya file ini dibuat pada waktu proses sortir.

- **Program File (Berkas Program)**

File yang berisi instruksi-instruksi untuk memproses data yang akan disimpan pada file lain atau pada memori utama.

Instruksi-instruksi tersebut dapat ditulis dalam bahasa tingkat tinggi (COBOL, FORTRAN, BASIC, dll), bahasa assembler dan bahasa mesin.

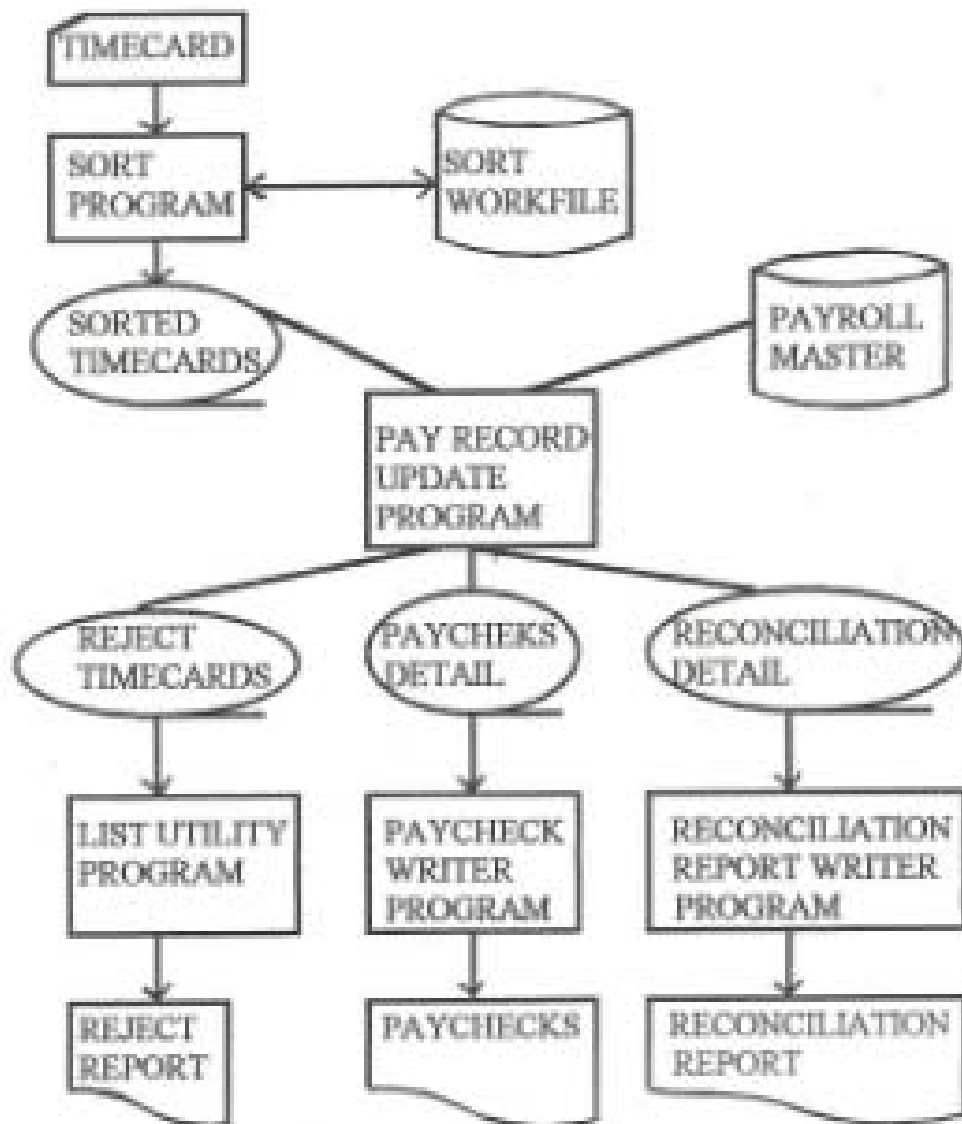
- **Text File (Berkas Teks)**

File yang berisi input data alfanumerik dan grafik yang digunakan oleh sebuah *text editor* program. *Text file* hanya dapat diproses dengan *text editor*.

- **Dump File (Berkas Tampung)**
File yang digunakan untuk tujuan pengamanan (*security*), mencatat tentang kegiatan peng-*update*-an, sekumpulan transaksi yang telah diproses atau sebuah program yang mengalami kekeliruan.
- **Library File (Berkas Pustaka)**
File yang digunakan untuk penyimpanan program aplikasi, program utilitas atau program lainnya.
- **History File (Berkas Sejarah)**
File ini merupakan tempat akumulasi dari hasil pemrosesan *master file* dan *transaction file*. File ini berisikan data yang selalu bertambah, sehingga file ini terus berkembang, sesuai dengan kegiatan yang terjadi.

Contoh:

Pada gambar 2 menunjukkan *system flow diagram* dari sistem penggajian sementara untuk menghasilkan *paycheck* berdasarkan *timecards* dan *payroll information*.



Gambar 2. System Flow Diagram

Tabel 1 menunjukkan klasifikasi file dari *system flow diagram*

Tabel 1. Klasifikasi File dan Fungsi

File	Fungsi
Timecards	Transaction
Sort Program	Program
Sort Workfile	Work
Sorted-Timecards	Transaction
Payroll Master	Master
Pay Record Update Program	Program
Reject Timecards	Work
Paycheck Detail	Work
Reconciliation Detail	Work
List Utility Program	Program
Reject Report	Report
Paycheck Writer Program	Program
Paychecks	Report
Reconciliation Report Writer Program	Program
Reconciliation Report	Report

MODEL AKSES BERKAS

Ada 3 model akses yang mungkin oleh sebuah program terhadap file, yaitu:

1. Input File

File yang hanya dapat dibaca dengan program.

Contoh:

- *Transaction file* merupakan input file untuk meng-*update* program.
- *Program file* dari *source code* merupakan input file untuk program *compiler*.

2. Output File

File yang hanya dapat ditulis oleh sebuah program / file yang dibuat dengan program.

Contoh:

- *Report file* merupakan output dari program yang meng-*update master file*.
- *Program file* yang berupa *object code* merupakan *output file* dari *program compiler*.

3. Input/Output File

File yang dapat dibaca dari dan ditulis ke selama eksekusi program.

Contoh:

- *Master File*
- *Work File* dengan *sort program*

Tabel 2 menunjukkan model akses file dari sistem *flow diagram*

Tabel 2. Model Akses File

Program File	Input File	Output File	Input/Output File
Sort	Timecards	SortedTimecards	Sort workfile
Pay Record Update	Sorted Timecards	Reject Timecards Paycheck Detail Reconciliation Detail	Payroll Master
Paycheck Writer	Paycheck Detail	Paychecks	
Reconciliation Report Writer	Reconciliation Detail	Reconciliation Report	
List Utility	Reject Timecards	Reject Report	

Note: Sebuah file mempunyai lebih dari satu fungsi, jika digunakan oleh lebih dari satu program.

ORGANISASI BERKAS

Suatu teknik/cara yang digunakan untuk menyatakan/menggambarkan dan menyimpan record-record dalam sebuah berkas.

Ada 4 teknik dasar organisasi berkas, yaitu:

1. Sequential File
2. Relative File
3. Index Sequential File
4. Multi-Key File

Secara umum keempat teknik dasar tersebut berbeda dalam cara pengaksesannya, yaitu:

- **Direct Access**

Suatu cara pengaksesan record yang langsung tanpa mengakses seluruh record yang ada.

Contoh: Magnetic Disk, CD

- **Sequential Access**

Suatu cara pengaksesan record, yang didahului pengaksesan record-record didepannya.

Contoh: Magnetic Tape, Punch Card

Faktor-faktor yang mempengaruhi dalam proses pemilihan organisasi berkas, yaitu:

- Karakteristik dari media penyimpanan yang digunakan.
- Volume dan frekuensi dari transaksi yang diproses
- Respon time yang diperlukan

OPERASI BERKAS

Cara memilih organisasi berkas tidak terlepas dari 2 aspek utama, yaitu:

1. Model penggunaannya ada 2 cara, yaitu:

- **Batch Processing**

Suatu proses yang dilakukan secara group atau kelompok.

- **Interactive Processing**

Suatu proses yang dilakukan secara satu persatu, yaitu record demi record.

2. Menurut model operasi file ada 4 cara, yaitu:

- **Creation**

Membuat struktur file lebih terdahulu, lalu menentukan banyak record baru kemudian record-record tersebut dimuat kedalam file tersebut atau membuat file dengan cara merekam record demi record.

- **Update**

Untuk menjaga agar file tetap *up to date*

- Insert / Add
- Modification
- Deletion

- **Retrieval**

Pengaksesan sebuah file dengan tujuan untuk mendapatkan informasi.

- **Inquiry**
Volume data rendah, model proses *interactive*
- **Report Generation**
Volume data tinggi, model proses *batch*

File retrieval dibagi menjadi 2, yaitu:

1. **Comprehensive Retrieval**

Mendapatkan informasi dari semua record dalam sebuah file.

Contoh:

Display all
List nama, alamat

2. **Selective Retrieval**

Mendapatkan informasi dari record-record tertentu berdasarkan persyaratan tertentu

Contoh:

List for Gaji = 1000000
List NPM, Nama, Alamat for angkatan = 99

- **Maintenance**

Perubahan yang dibuat terhadap file dengan tujuan memperbaiki penampilan program dalam mengakses file tersebut.

- **Restructuring:** perubahan struktur file
Contoh:
Panjang field diubah
Penambahan field baru
Panjang record diubah

- **Reorganization:** perubahan organisasi file dari organisasi yang satu, menjadi organisasi file yang lain.

Contoh:

organisasi *sequential file* menjadi *indexed sequential*
organisasi *direct file* menjadi *sequential file*

Secara umum dapat disimpulkan, bahwa:

- *Master file* dan *program file*, kita dapat melakukan *created*, *updated*, *retrieved from* dan *maintained*.
- *Workfile*, kita dapat melakukan *created*, *updated* dan *retrieved from*, tetapi tidak dapat kita *maintained*.
- *Report file* umumnya tidak di *updated*, *retrieved from* atau *maintained*.
- *Transaction file*, umumnya hanya dapat di *create* dan digunakan untuk sekali proses.