**DATA FLOW DIAGRAM**

Data Flow Diagram (DFD) adalah alat yang digunakan dalam rekayasa perangkat lunak untuk menggambarkan aliran data dalam sebuah sistem. DFD membantu untuk memahami bagaimana data mengalir melalui sistem dan bagaimana berbagai komponen berinteraksi

DFD atau *data* *flow* *diagram* adalah representasi grafik yang menggambarkan aliran dan transformasi informasi yang diaplikasikan sebagai data yang mengalir dari *input* maupun *output*. Sederhananya, diagram ini menggambarkan apa yang terjadi dalam sebuah sistem. DFD disajikan dalam bentuk gambar yang berisi simbol atau notasi yang digunakan untuk memahami mekanisme aliran data dalam suatu sistem.

**Fungsi DFD**

DFD merupakan alat perancangan sistem yang berorientasi pada alur data dengan konsep dekomposisi. DFD dapat pula digunakan untuk menggambarkan suatu analisa maupun rancangan sistem yg mudah dikomunikasikan oleh profesional sistem (system analyst) kepada pemakai (user) maupun pembuat program (programmer). Fungsi dan manfaat DFD antara lain adalah sebagai berikut.

1. DFD membantu para analis sistem meringkas informasi tentang sistem, mengetahui hubungan antar sub-sub sistem, membantu perkembangan aplikasi secara efektif.
2. DFD berfungsi sebagai alat komunikasi yang baik antara pemakai dan analis sistem.
3. DFD dapat menggambarkan sejumlah batasan otomasi untuk pengembangan alternatif sistem

**Komponen DFD**

1. Terminator/External Entity

Terminator dalam Diagram Alur Data (DFD) dapat kita gambarkan sebagai sebuah kotak persegi panjang yang merepresentasikan sesuatu di luar sistem yang sedang kita modelkan, seperti orang, kelompok orang, atau organisasi

2. Proses

Komponen proses dalam DFD dapat dijelaskan sebagai suatu langkah di mana *input* diubah menjadi *output*(Hasanah & Untari, 2020, hlm. 51). Nama proses ini harus mencerminkan kegiatan atau tugas yang sedang dilakukan, dan penamaannya sebaiknya menggunakan kata kerja transistif, yaitu kata kerja yang membutuhkan objek.

terdapat beberapa hal yang harus diperhatikan tentang komponen proses dalam diagram alur data antara lain adalah sebagai berikut.

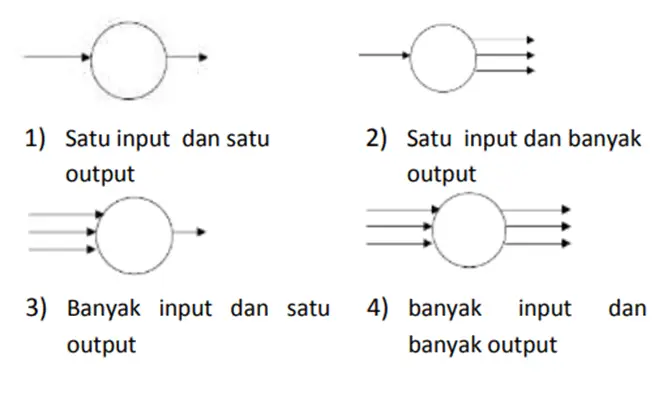
* Setiap komponen proses harus memiliki *input* dan *output*.
* proses dapat dihubungkan dengan komponen *terminator (external entity)*, *data* *store* atau alur data.

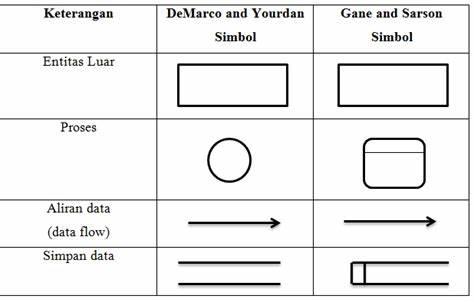
Sistem, bagian, divisi atau departemen yang sedang dianalisis oleh profesional sistem dapat digambarkan dengan komponen proses.

Terdapat 4 kemungkinan yang dapat terjadi dalam suatu proses sehubungan dengan pengolahan komponen input menjadi *output*, yakni sebagai berikut.

* Satu input dan satu output.
* Satu input dan banyak output.
* Banyak input dan satu output.
* Banyak input dan banyak output (Hasanah & Untari, 2020, hlm. 51).

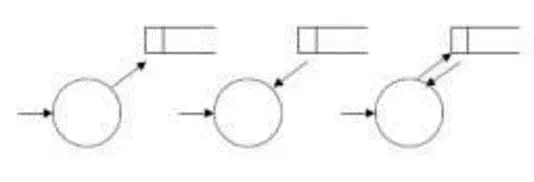
Dalam DFD, simbol untuk komponen proses adalah lingkaran dengan nama proses ditulis di dalamnya seperti pada contoh gambar di bawah ini.





3. Penyimpanan Data (Data Store)

*Data* *store* dalam DFD (*diagram* *alur* *data*) digunakan untuk merepresentasikan kumpulan data atau paket data. Ini berfungsi sebagai tempat penyimpanan untuk informasi yang dapat diakses oleh proses atau entitas tertentu. *Data* *store* dapat mencakup *file* atau basis data yang disimpan di media penyimpanan seperti disket atau *harddisk*. Fungsinya sering kali diatur oleh batasan waktu tertentu.

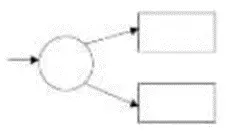


4. Aliran data (*flow*)

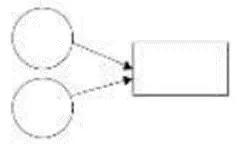
Alur data dalam Diagram Alur Data (DFD) adalah elemen yang menjelaskan perpindahan data atau paket data dari satu bagian sistem ke bagian lainnya. Alur data ini mencerminkan aliran informasi yang bisa berupa masukan untuk sistem atau hasil dari suatu proses sistem.

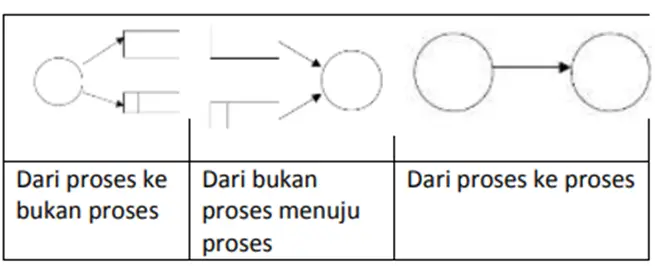
Terdapat beberapa konsep tentang alur data yang akan dipaparkan sebagai berikut.

1. ***Packets of data.***  
   Apabila ada 2 data atau lebih yg mengalir dari 1 sumber yang sama menuju pada tujuan yang sama pula dan mempunyai hubungan maka digambarkan dengan 1 alur data.  
   
2. ***Diverging data flow.***  
   Apabila ada sejumlah paket data yg berasal dari sumber yg sama menuju pada tujuan yang berbeda atau paket data yg kompleks maka diagram dibagi menjadi beberapa elemen data yg dikirim ke tujuan yg berbeda.



1. ***Converging data flow.***  
   Alur data yang mengumpul menunjukkan beberapa alur data yang berbeda dari sumber data yang berbeda bergabung bersama-sama menuju tujuan yang sama.



1. **Sumber dan Tujuan.**  
   Arus data harus dihubungkan pada proses, baik dari maupun menuju proses.  
   

**Tingkatan/Level DFD**

DFD dapat dibagi menjadi beberapa level yang lebih detail untuk merepresentasikan aliran informasi atau fungsi yang lebih detail. Dalam penerapannya tidak ada aturan yang baku tentang tingkatan atau level DFD. Secara umum terdapat beberapa tingkatan yang sering digunakan antara lain adalah sebagai berikut.

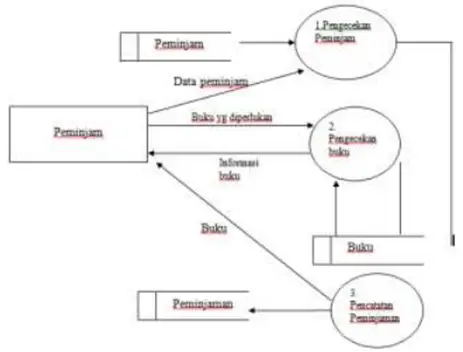
1. DFD level 0 atau biasa disebut diagram konteks. Diagram konteks menggambarkan secara global aliran informasi dan data yang akan dilakukan oleh sistem. Diagram konteks ini merupakan level tertinggi (top level) yang menggambarkan hubungan antar sistem dengan entitas di luar sistem dan merupakan gambaran sistem secara keseluruhan. Komponen yang ada dalam diagram ini biasanya komponen proses, *external* *entity* dan alur data.
2. DFD level 1, diagram ini menjelaskan lebih detail dari diagram konteks. Diagram ini merupakan dekomposisi diagram konteks. Beberapa proses dalam diagram konteks dapat dijelaskan lebih rinci.
3. DFD level 2, diagram ini merupakan dekomposisi dari diagram level 1.
4. Diagram level 3 dan seterusnya, diagram ini merupakan dekomposisi dari diagram level 2

**Langkah menyusun DFD**

Menurut Hasanah & Untari (2020, hlm. 55) langkah-langkah yang dapat dilakukan untuk membuat diagram alur data atau DFD ini di antaranya adalah sebagai berikut.

1. **Membuat diagram konteks**  
   a) Tentukan nama sistemnya; b) Tentukan batasan sistemnya; c) Tentukan terminator apa saja yg ada dalam sistem; d) Tentukan apa yg diterima dan diberikan terminator dari/pada sistem; e) Gambarkan diagram context.



1. **Membuat diagram level**  
   a) Tentukan proses utama yg ada pada sistem; b) Tentukan apa yg diberikan/diterima masing-masing proses pada/dari sistem sambil memperhatikan konsep keseimbangan (alur data yg keluar/masuk dari suatu level harus sama dgn alur data yg masuk/keluar pada level berikutnya); c) Apabila diperlukan, munculkan data store (master) sebagai sumber maupun tujuan alur data; d) Gambarkan diagram level zero; e) Hindari perpotongan arus data; f) Beri nomor pada proses utama (nomor tidak menunjukkan urutan proses).  
   
2. **Membuat diagram level 2**a) Tentukan proses yg lebih kecil (sub-proses) dari proses utama yg ada di level zero; b) Tentukan apa yg diberikan/diterima masing-masing subproses pada/dari sistem dan perhatikan konsep keseimbangan; c) Apabila diperlukan, munculkan data store (transaksi) sbg sumber maupun tujuan alur data; d) Gambarkan DFD level Satu; e) Hindari perpotongan arus data; f) Beri nomor pada masing-masing sub-proses yg menunjukkan dekomposisi dari proses sebelumnya. Misalnya 1.1, 1.2, 2.1.

