

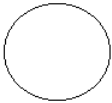







DATA FLOW DIAGRAM

Sumber : Metodologi Pengembangan Sistem Informasi, 1998. Leman, PT Elex Media Komputindo, Jakarta
Pengantar Sistem Informasi Bisnis, 1995. Lani Sidarta, PT Elex Media Komputindo, Jakarta

Data Flow Diagram (DFD) adalah representasi grafik dari sebuah sistem. DFD menggambarkan komponen-komponen sebuah sistem, aliran-aliran data di mana komponen-komponen tersebut, dan asal, tujuan, dan penyimpanan dari data tersebut.

Kita dapat menggunakan DFD untuk dua hal utama, yaitu untuk membuat

Notasi Yourdon / DeMarco	Notasi Gane & Sarson	
		Simbol Entitas eksternal/Terminator menggambarkan asal atau tujuan data di luar sistem
		Simbol lingkaran menggambarkan entitas atau proses dimana aliran data masuk ditransformasikan ke aliran data keluar.
		Simbol aliran data menggambarkan aliran data
		Simbol file menggambarkan tempat data di simpan

dokumentasi dari sistem informasi yang ada, atau untuk menyusun dokumentasi untuk sistem informasi yang baru.

Empat simbol yang digunakan :

Ada 3 (tiga) jenis DFD, yaitu ;

- Context Diagram (CD)
- DFD Fisik
- DFD Logis

DFD Level

DFD dapat digambarkan dalam Diagram Context dan Level n. Huruf n dapat menggambarkan level dan proses di setiap lingkaran.

- Diagram Context
- Diagram Level n
 - DFD Logis
 - DFD Fisik

Context Diagram (CD)

Jenis pertama Context Diagram, adalah data flow diagram tingkat atas (DFD Top Level), yaitu diagram yang paling tidak detail, dari sebuah sistem informasi yang menggambarkan aliran-aliran data ke dalam dan ke luar sistem dan ke dalam dan ke luar entitas-entitas eksternal. (CD menggambarkan sistem dalam satu lingkaran dan hubungan dengan entitas luar. Lingkaran tersebut menggambarkan keseluruhan proses dalam sistem).

Beberapa hal yang harus diperhatikan dalam menggambar CD;

- Terminologi sistem :
 - *Batas Sistem* adalah batas antara “daerah kepentingan sistem”.
 - *Lingkungan Sistem* adalah segala sesuatu yang berhubungan atau mempengaruhi sistem tersebut.
 - *Interface* adalah aliran yang menghubungkan sebuah sistem dengan lingkungan sistem tersebut.

Sebagai contoh, dalam gambar 1.

- Menggunakan satu simbol proses,
Catatan:

Yang masuk didalam lingkaran konteks (simbol proses) adalah kegiatan pemrosesan informasi (Batas Sistem). Kegiatan informasi adalah mengambil data dari file, mentransformasikan data, atau melakukan filing data, misalnya mempersiapkan dokumen, memasukkan, memeriksa, mengklasifikasi, mengatur, menyortir, menghitung, meringkas data, dan melakukan filing data (baik yang melakukan secara manual maupun yang dilakukan secara terotomasi).

- Nama/keterangan di simbol proses tersebut sesuai dengan fungsi sistem tersebut,
- Antara Entitas Eksternal/Terminator tidak diperbolehkan komunikasi langsung
- Jika terdapat termintor yang mempunyai banyak masukan dan keluaran, diperbolehkan untuk digambarkan lebih dari satu sehingga mencegah penggambaran yang terlalu rumit, dengan memberikan tanda asterik (*) atau garis silang (#).
- Jika Terminator mewakili individu (personil) sebaiknya diwakili oleh peran yang dipermainkan personil tersebut.
- Aliran data ke proses dan keluar sebagai output keterangan aliran data berbeda.

Diagram Level n / Data Flow Diagram Levelled

Dalam diagram n DFD dapat digunakan untuk menggambarkan diagram fisik maupun diagram diagram logis. Dimana Diagram Level n merupakan hasil pengembangan dari *Context Diagram* ke dalam komponen yang lebih detail tersebut disebut dengan top-down partitioning. Jika kita melakukan pengembangan dengan benar, kita akan mendapatkan DFD-DFD yang seimbang. Sebagai contoh, gambar 1.1, gambar 1.2, gambar 1.3, gambar 1.4 dan gambar 1.5.

Beberapa hal yang harus diperhatikan dalam membuat DFD ialah:

- Pemberian Nomor pada diagram level n dengan ketentuan sebagai berikut:
 - Setiap penurunan ke level yang lebih rendah harus mampu merepresentasikan proses tersebut dalam spesifikasi proses yang jelas. Sehingga seandainya belum cukup jelas maka seharusnya diturunkan ke level yang lebih rendah.
 - Setiap penurunan harus dilakukan hanya jika perlu.
 - Tidak semua bagian dari sistem harus diturunkan dengan jumlah level yang sama karena yang kompleks bisa saja diturunkan, dan yang sederhana mungkin tidak perlu diturunkan. Selain itu, karena tidak semua proses dalam level yang sama punya derajat kompleksitas yang sama juga.
 - Konfirmasikan DFD yang telah dibuat pada pemakai dengan cara top-down.
 - Aliran data yang masuk dan keluar pada suatu proses di level n harus berhubungan dengan aliran data yang masuk dan keluar pada level $n+1$. Dimana level $n+1$ tersebut mendefinisikan sub-proses pada level n tersebut.
 - Penyimpanan yang muncul pada level n harus didefinisikan kembali pada level $n+1$, sedangkan penyimpanan yang muncul pada level n tidak harus muncul pada level $n-1$ karena penyimpanan tersebut bersifat lokal.
 - Ketika mulai menurunkan DFD dari level tertinggi, cobalah untuk mengidentifikasi external events dimana sistem harus memberikan respon. External events dalam hal ini berarti suatu kejadian yang berkaitan dengan pengolahan data di luar sistem, dan menyebabkan sistem kita memberikan respon.
- Jangan menghubungkan langsung antara satu penyimpanan dengan penyimpanan lainnya (harus melalui proses).
- Jangan menghubungkan langsung dengan tempat penyimpanan data dengan entitas eksternal / terminator (harus melalui proses), atau sebaliknya.
- Jangan membuat suatu proses menerima input tetapi tidak pernah mengeluarkan output yang disebut dengan istilah "black hole".
- Jangan membuat suatu tempat penyimpanan menerima input tetapi tidak pernah digunakan untuk proses.
- Jangan membuat suatu hasil proses yang lengkap dengan data yang terbatas yang disebut dengan istilah "magic process".
- Jika terdapat terminator yang mempunyai banyak masukan dan keluaran, diperbolehkan untuk digambarkan lebih dari satu sehingga mencegah penggambaran yang terlalu rumit, dengan memberikan tanda asterik (*) atau garis silang (#), begitu dengan bentuk penyimpanan.
- Aliran data ke proses dan keluar sebagai output keterangan aliran data berbeda.

DFD Fisik

Adalah representasi grafik dari sebuah sistem yang menunjukkan entitas-entitas internal dan eksternal dari sistem tersebut, dan aliran-aliran data ke dalam dan keluar dari entitas-entitas tersebut. Entitas-entitas internal adalah personel, tempat (sebuah bagian), atau mesin (misalnya, sebuah komputer) dalam sistem tersebut yang mentransformasikan data. Maka DFD fisik tidak menunjukkan apa yang dilakukan, tetapi menunjukkan dimana, bagaimana, dan oleh siapa proses-proses dalam sebuah sistem dilakukan. (Tidak Bahas).

Perlu diperhatikan didalam memberikan keterangan di lingkaran-lingkaran (simbol proses) dan aliran-aliran data (simbol aliran data) dalam DFD fisik menggunakan label/keterangan dari kata benda untuk menunjukkan bagaimana sistem mentransmisikan data antara lingkaran-lingkaran tersebut.

Misal :

Aliran Data : Kas, Formulir 66W, Slip Setoran

Proses : Cleck Penjualan, Kasir, Pembukuan, dll.

DFD Logis

Adalah representasi grafik dari sebuah sistem yang menunjukkan proses-proses dalam sistem tersebut dan aliran-aliran data ke dalam dan ke luar dari proses-proses tersebut. Kita menggunakan DFD logis untuk membuat dokumentasi sebuah sistem informasi karena DFD logis dapat mewakili logika tersebut, yaitu apa yang dilakukan oleh sistem tersebut, tanpa perlu menspesifikasi dimana, bagaimana, dan oleh siapa proses-proses dalam sistem tersebut dilakukan.

Keuntungan dari DFD logis dibandingkan dengan DFD fisik adalah dapat memusatkan perhatian pada fungsi-fungsi yang dilakukan sistem.

Perlu diperhatikan di dalam pemberian Keterangan/ Label;

- Lingkaran-lingkaran (simbol proses) menjelaskan apa yang dilakukan sistem
Misal : Menerima Pembayaran, Mencatat Penjualan, Membandingkan kas dan Daftar Penerimaan, Mempersiapkan Setoran, dll.
- Aliran-aliran data (simbol aliran data) menggambarkan sifat data.
Misal : Pembayaran (bukan “Cek”, “Kas”, “Kartu Kredit”
Jurnal Penjualan (bukan “Buku Penjualan”), dll

Usulan dari analisis (berupa DFD dalam bab 4), beberapa hal yang umum yang mendapat perhatian dalam mendesain baru tersebut ialah:

- Menggabungkan beberapa tugas menjadi Satu
- Master Detail Update
- Meminimalkan tugas-tugas yang tidak penting
- Menghilangkan tugas-tugas yang duplikat
- Menambahkan proses baru
- Meminimalkan proses input
- Menetapkan bagian mana yang harus dikerjakan komputer dan bagian mana yang harus dikerjakan manual

