

Analisis Terstruktur

Universitas Indraprasta PGRI



Sekilas sejarah



- ▶ Dipopularkan oleh DeMarco (1979)
 - ▶ Dikembangkan lebih lanjut oleh Page-Jones (1980), Gane dan Sarson (1982)
 - ▶ Dikembangkan untuk sistem waktu nyata (Real Time) oleh Ward dan Mellor (1985) kemudian Hatley dan Pirbhai (1987)
 - ▶ Merupakan teknik pemodelan *information flow* dan *information content*
- 

Diagram Aliran Data

- ▶ Teknik grafis yang menggambarkan aliran informasi dan perubahan yang dilakukan terhadap data dari masukan menjadi keluaran
 - ▶ Nama lain : data Flow Diagram, Data Flow Graph, Bubble Chart.
 - ▶ Banyak digunakan karena kesederhanaannya.
- 

▶ Diagram Aliran Data

- ▶ Terdiri dari : Proses, Entitas Eksternal (External Entity), penyimpanan data (Data Store), dan Aliran Data (Data Flow).
- ▶ Bukan merupakan flowchart dengan lingkaran-lingkaran. DAD menggambarkan aliran informasi tanpa representasi eksplisit dari logika proseduralnya.

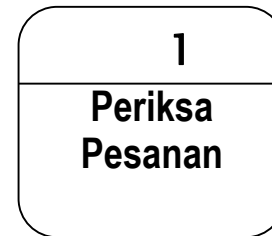
Proses

Merupakan bagian dari DAD yang mengubah (transform) satu atau lebih masukan (aliran data yang masuk) menjadi keluaran–keluaran (aliran data yang keluar).

- ▶ Nama lain : Bubble, fuction dan transform



Model DeMarco/Yourdon



Model Gane & Sarson

- ▶ Notasi :

Entitas Eksternal

- ▶ Merupakan seseorang/sekelompok orang atau suatu organisasi luar, atau suatu kelompok/departemen di dalam perusahaan/organisasi yang sama, tetapi di luar kendali dari sistem yang dibuat modelnya.
- ▶ Nama lain : *External Entity, Source and Destination, Terminator*
- ▶ Notasi :

Konsumen

Penyimpanan Data

- ▶ Merupakan bagian dari DAD yang digunakan untuk menunjukkan suatu kumpulan dari paket data yang diam.
- ▶ Nama lain : Data Store File
- ▶ Notasi :

Pesanan

□ Pesanan

Aliran Data

- ▶ Digunakan untuk menunjukkan pergerakan dari paket data atau informasi dari satu bagian sistem ke bagian yang lain
- ▶ Nama lain : Data Item, Data Flow
- ▶ Notasi :

pesanan barang
→

Diagram Aliran Data Bertingkat

▶ Dasar Pemikiran

◦ ROSS

- Pikiran manusia dapat menerima segala bentuk kerumitan, asalkan disajikan dalam susunan yang terdiri bagian-bagian kecil yang mudah dimengerti

◦ GEORGE MILLER

- Pikiran manusia paling banyak dapat mengerti sesuatu yang terbagi menjadi 7 ± 2 bagian dan tetap masih dapat mengerti konsep dari sesuatu tadi secara keseluruhan

Diagram Aliran Data Bertingkat

- ▶ Jenis DAD dalam DAD Bertingkat
 - Diagram konteks (*Context Diagram*)
 - Diagram paling atas, terdiri dari satu proses dan menggambarkan ruang lingkup sistem
 - Diagram Prinsip Fungsional (*Functional Primitive*)
 - Diagram–diagram paling bawah, yang tidak dapat dibagi lagi atau memiliki masukan tunggal dan keluaran tunggal atau telah sangat sederhana (narasi untuk deskripsi dapat dituliskan secara singkat)

Diagram Aliran Data Bertingkat

- ▶ Jenis DAD dalam DAD Bertingkat (lanjutan) :
 - Diagram Tengah
 - Diagram–diagram yang terletak antara Diagram Konteks dan Primitif Fungsional

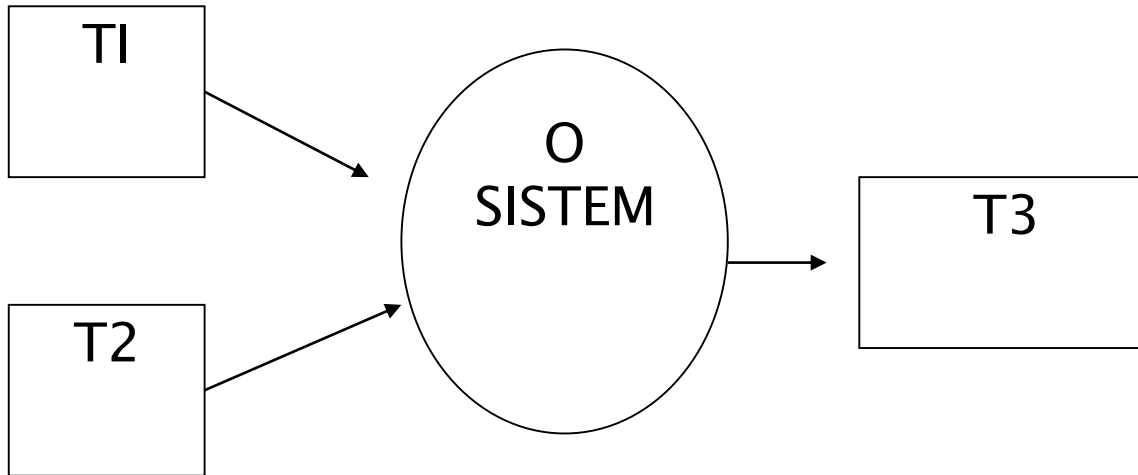
Penyusunan DAD

▶ Penomoran

- Diagram konteks biasanya diberi nomor 0
- Proses-proses pada DAD paling atas diberi nomor mulai dari 1 dan seterusnya sampai semua proses bernomor
- Pada saat setiap proses dipecah menjadi DAD dengan tingkat yang lebih rendah, maka proses pada DAD tersebut diberi nomor sesuai dengan nomor proses tadi
- Setiap proses diberi nomor yang merupakan kombinasi dari nomor diagram diikuti dengan nomor urut dalam tingkat yang bersangkutan.

Penyusunan DAD

▶ Contoh Diagram Konteks



Penyusunan DAD

- ▶ Nomor Diagram “ ANAK” harus diawali dengan nomor proses pada diagram ‘ ORANG TUA ‘ yang terkait,

Diagram 0

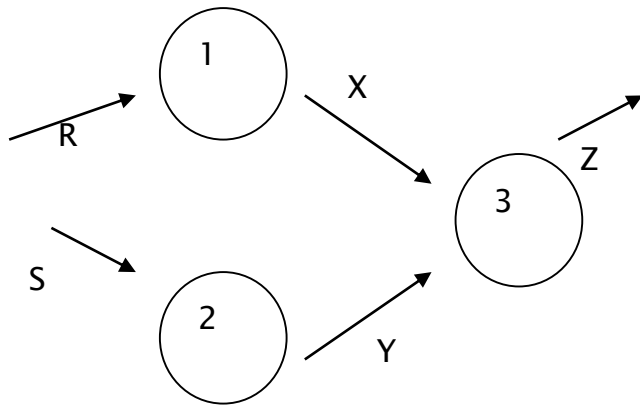


Diagram 3

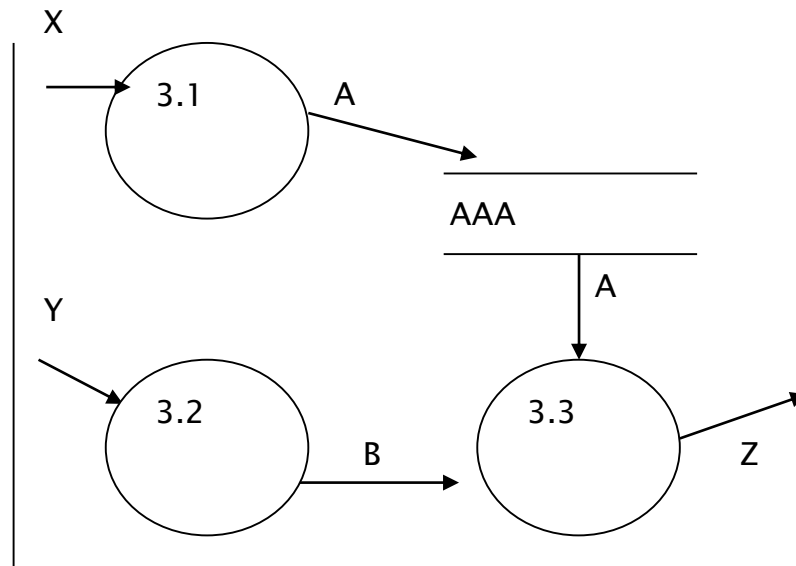


DIAGRAM KONTEKS

Diagram konteks adalah diagram yang terdiri dari suatu proses dan menggambarkan ruang lingkup suatu sistem. Diagram konteks merupakan level tertinggi dari DFD yang menggambarkan seluruh input ke sistem atau output dari sistem. Ia akan memberi gambaran tentang keseluruhan sistem. Dalam **diagram konteks hanya ada satu proses.**




DIAGRAM NOL

Tujuan dari diagram nol adalah untuk “memerinci” sebuah sistem menjadi “proses–proses” yang harus dilakukan ‘orang dalam.’ Atau jika dibuat dalam kalimat adalah : “Apa saja proses yang harus dilakukan agar mencapai sistem tersebut. Jadi, diagram ini adalah kelanjutan dari diagram konteks, yang “memperbanyak lingkaran (proses), sedangkan untuk (jumlah dan isi) *terminator (entitas)* serta (jumlah dan isi) *data flow* dari dan ke terminator tersebut harus tetap.

DIAGRAM NOL

Pada diagram ini pula mulai ditampilkan *data store* (penyimpanan data/ *file*) yang dibutuhkan.

DIAGRAM RINCI

Diagram Rinci adalah diagram yang memungkinkan proses yang ada di diagram nol lebih diperinci lagi.

PENOMORAN PROSES

Penomoran proses pada tiap tingkat (level) di DFD :

Level Name	Diagram Name	Number of Proses
0	Context	
1	Diagram 0	1.0, 2.0, 3.0, ...
2	Diagram 1.0	1.1, 1.2, 1.3, ...
3	Diagram 1.1	1.1.1, 1.1.2, ...

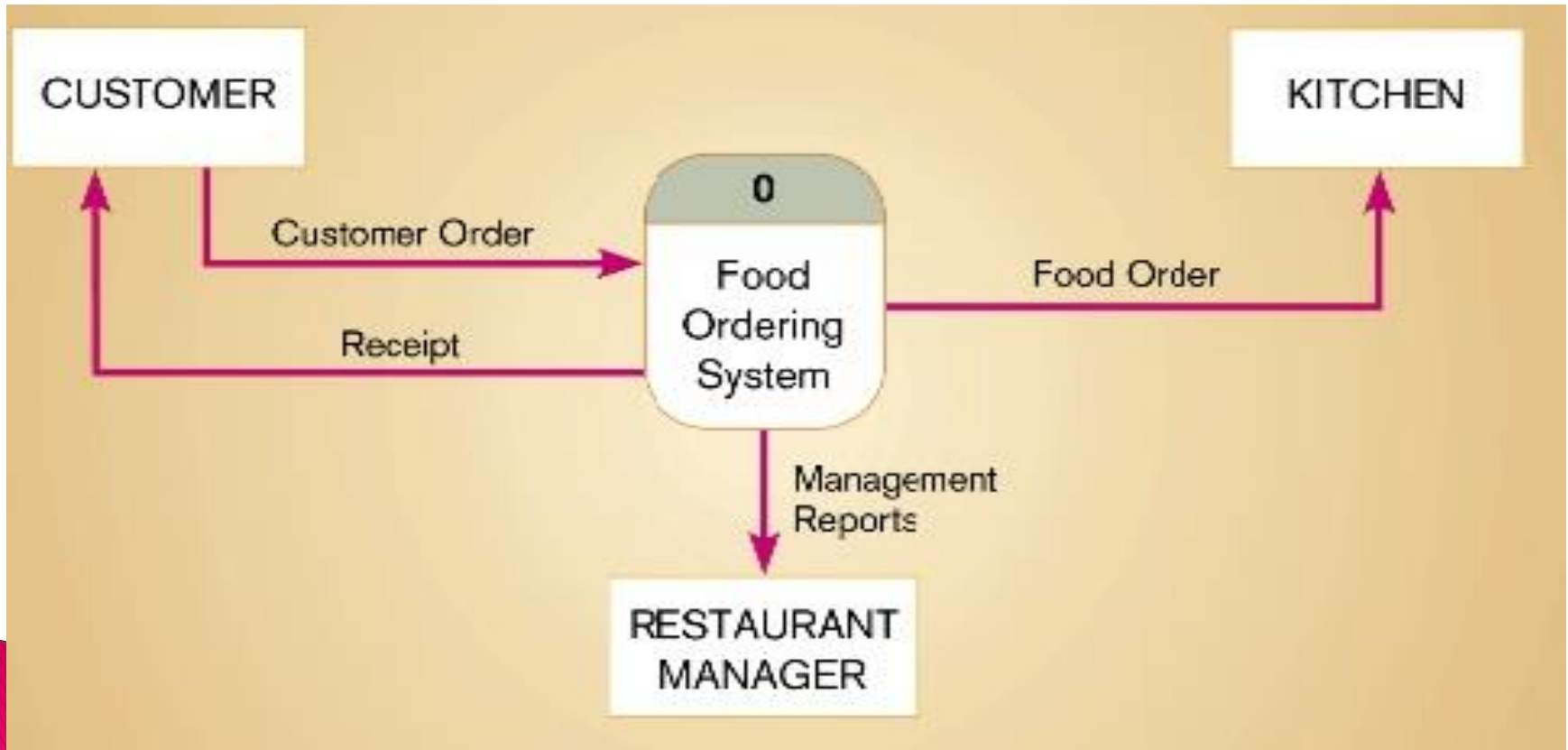
Dalam satu level tidak boleh ada lebih dari 7 unit dan maksimal 9, jika lebih harus dilakukan dekomposisi (pemecahan lebih terinci).

BALANCING

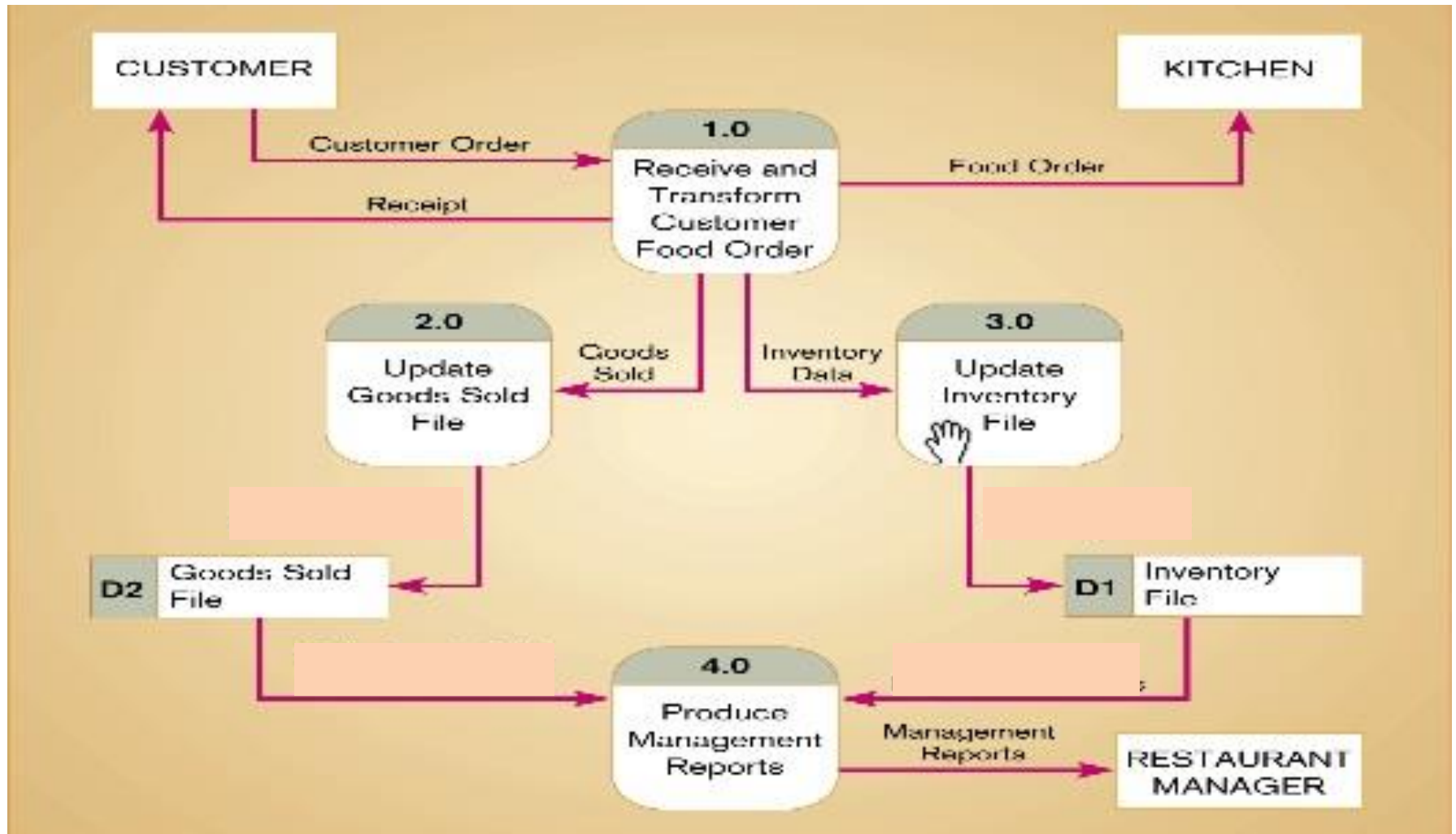
Balancing (keseimbangan) dalam penggambaran levelisasi DFD perlu diperhatikan. Balancing DFD ini maksudnya keseimbangan antara alur data yang masuk/keluar dari suatu level harus sama dengan alur data yang masuk/keluar pada level berikutnya.

BALANCING

Contoh DFD Balancing dapat dilihat pada gambar di bawah ini: Pada gambar dapat dilihat ada satu input ke dalam sistem yaitu Customer Order dan tiga output yang keluar dari sistem yaitu Receipt, Food Order, dan Management Reports.



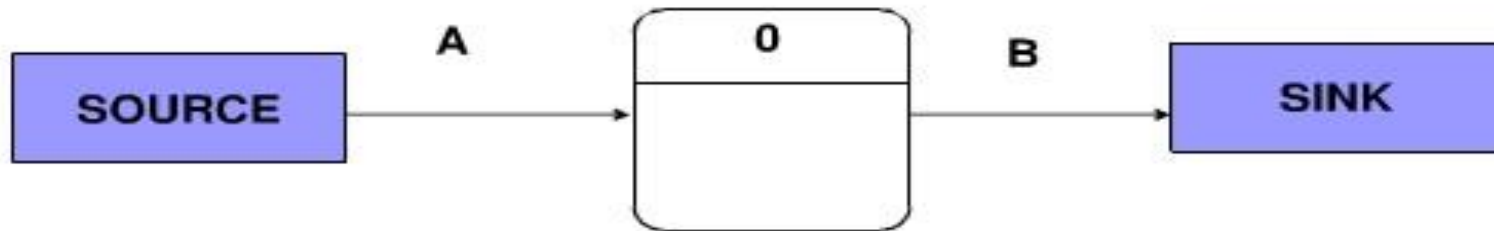
BALANCING



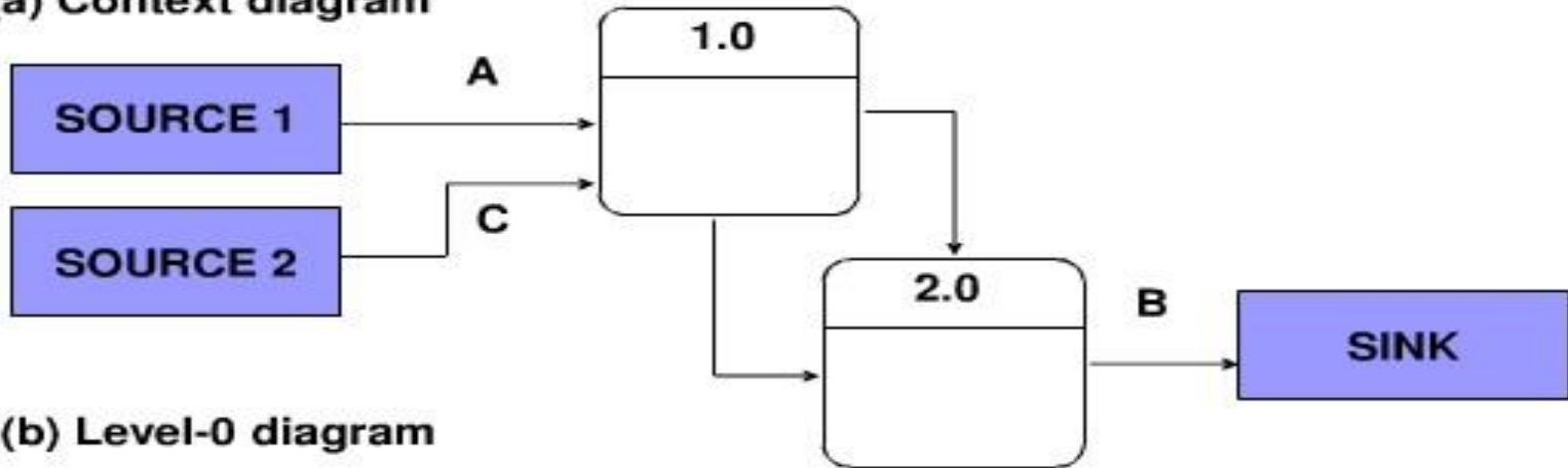
Gambar diatas sudah dikatakan **seimbang** karena mempunyai input dan output yang sama dan tidak memasukan input ataupun output baru.

BALANCING

Contoh DFD Unbalancing



(a) Context diagram



(b) Level-0 diagram

Dalam diagram konteks mempunyai satu inputan yaitu A dan satu keluaran yaitu B. Sedangkan pada diagram level 0 ada penambahan inputan baru yaitu C, berarti DFD tersebut dapat dikatakan **tidak seimbang**.

KAMUS DATA


Kamus Data adalah katalog fakta tentang data dan kebutuhan-kebutuhan informasi dari suatu sistem informasi (Jogiyanto, 1990). Kamus Data digunakan untuk :

- Merancang Input
- Merancang laporan- laporan dan database

Kamus data dibuat berdasarkan arus data yang terdapat pada data flow diagram. (Jogiyanto, 1990).

KAMUS DATA

Kamus data selain digunakan untuk dokumentasi dan mengurangi redundansi, juga dapat digunakan untuk:

- Memvalidasi diagram aliran data dalam hal kelengkapan dan keakuratan
 - Menyediakan suatu titik awal untuk mengembangkan layar dan laporan-laporan
 - Menentukan muatan data yang disimpan dalam file-file
 - Mengembangkan logika untuk proses-proses diagram aliran data
- 

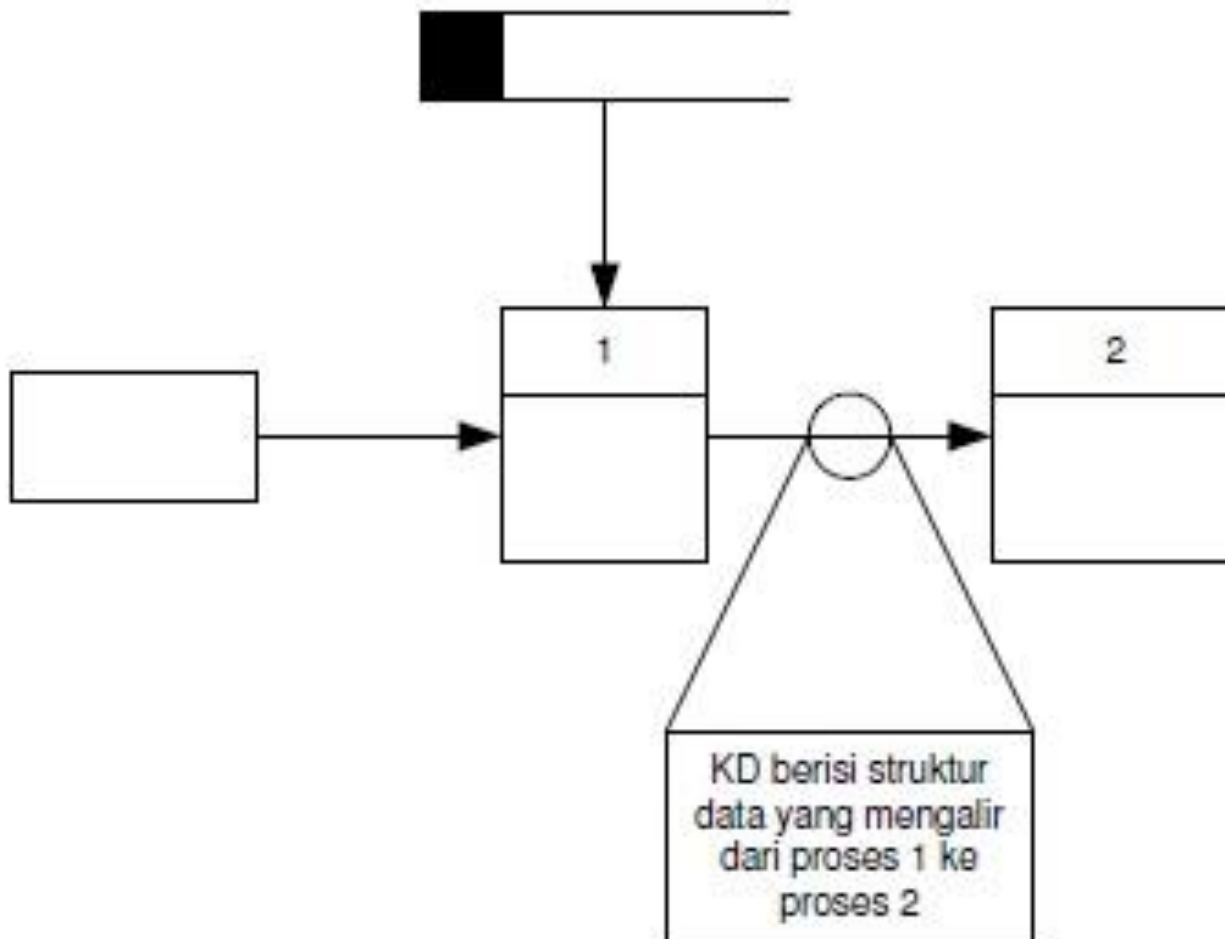
KAMUS DATA

KD mendefinisikan elemen data dengan fungsi sebagai berikut:

- Menjelaskan arti aliran data dan penyimpanan data dalam DFD
- Mendeskripsikan komposisi paket data yang bergerak melalui aliran (misalnya alamat diuraikan menjadi kota, negara dan kode pos)
- Mendeskripsikan komposisi penyimpanan data
- Menspesifikasikan nilai dan satuan yang relevan bagi penyimpanan dan aliran
- Mendeskripsikan hubungan detail antar penyimpanan (yang akan menjadi titik perhatian dalam *entity-relationship diagram*)

KAMUS DATA

Gambaran hubungan DAD dan KD



ELEMEN-ELEMEN DATA

Kamus data harus dapat mencerminkan keterangan yang jelas tentang data yang akan dicatat. Untuk maksud keperluan ini, maka kamus data harus memuat hal-hal berikut:

- **Nama Arus Data**, karena kamus data dibuat berdasarkan arus data yang mengalir di DAD, maka nama dari arus data juga harus dicatat di KD.
- **Alias**, alias atau nama lain dari data dapat dituliskan bila nama lain ini ada. Alias perlu ditulis karena data yang sama mempunyai nama yang berbeda untuk orang atau departemen satu dengan yang lainnya. Misalnya bagian pembuat faktur dan langganan menyebut bukti penjualan sebagai faktur, sedangkan bagian gudang menyebutnya sebagai tembusan permintaan persediaan. Baik faktur dan tembusan permintaan persediaan ini mempunyai struktur data yang sama, tetapi mempunyai struktur yang berbeda.

ELEMEN-ELEMEN DATA

- **Bentuk data**, arus data dapat mengalir:
 - Dari kesatuan luar ke suatu proses, data yang mengalir ini biasanya tercatat di suatu dokumen atau formulir.
 - Hasil dari suatu proses ke kesatuan luar, data yang mengalir ini biasanya terdapat di media laporan atau query tampilan layar atau dokumen hasil cetakan komputer;
 - Hasil suatu proses ke proses yang lain, data yang mengalir ini biasanya dalam bentuk variabel atau parameter yang dibutuhkan oleh proses penerimanya;
 - Hasil suatu proses yang direkamkan ke simpanan data, data yang mengalir ini biasanya berbentuk suatu variabel.
 - Dari simpanan data dibaca oleh suatu proses, data yang mengalir ini biasanya berupa suatu field (item data).

ELEMEN-ELEMEN DATA

- **Arus Data**, arus data menunjukkan dari mana data mengalir, dan kemana data akan menuju. Keterangan ini perlu dicatat di Kamus Data agar mudah mencari kamus data di DFD
- **Penjelasan**, Untuk lebih memperjelas lagi tentang makna dari arus data yang dicatat di KD, maka bagian penjelasan dapat diisi dengan keterangan-keterangan tentang arus data tersebut. Misalnya nama dari arus data adalah Tembusan Permintaan Persediaan, maka dapat lebih dijelaskan sebagai tembusan dari faktur penjualan untuk meminta barang dari gudang.

ELEMEN-ELEMEN DATA

- **Periode**, periode ini menunjukkan kapan terjadinya arus data ini. Periode perlu dicatat di KD karena dapat digunakan untuk mengidentifikasi kapan input data harus dimasukkan ke sistem, kapan proses dari program harus dilakukan dan kapan laporan-laporan harus dihasilkan.
- **Volume**, volume yang perlu dicatat di KD adalah tentang volume rata-rata dan volume puncak dari arus data. Volume rata-rata menunjukkan banyaknya rata-rata arus data yang mengalir dalam satu periode tertentu dan volume puncak menunjukkan volume yang terbanyak. Volume ini digunakan untuk mengidentifikasi besarnya simpanan luar yang akan digunakan, kapasitas dan jumlah dari alat input, alat pemroses dan alat output.
- **Struktur data**, struktur data menunjukkan arus data yang dicatat di KD terdiri dari item-item data apa saja.

MENGGAMBARKAN STRUKTUR DATA (KAMUS DATA KOMPOSIT)

Pada kebanyakan sistem dalam dunia nyata (dimana kita bekerja), kadang-kadang elemen data terlalu kompleks untuk didefinisikan. Kekompleksan tersebut seharusnya diuraikan melalui sejumlah elemen data yang lebih sederhana. Kemudian elemen data yang lebih sederhana tersebut didefinisikan kembali hingga nilai dan satuan relevan (yang sifatnya elementer).

MENGGAMBARKAN STRUKTUR DATA (KAMUS DATA KOMPOSIT)

Contoh dari pemakaian tersebut diatas:

Tembusan Permintaan Persediaan = Kode Langganan +
Nama Langganan +
Tanggal Penjualan +
Nomor Faktur +
1 { Informasi Barang } 5 +
Total Penjualan +
(Potongan Penjualan) +
Pajak Penjualan +
Total Dibayar +
Jenis Penjualan

MENGGAMBARKAN STRUKTUR DATA (KAMUS DATA KOMPOSIT)

Informasi Barang = Kode Barang +
Nama Barang +
Unit Jual +
Harga Satuan +
Total Harga

Jenis Penjualan = [Cash | Credit]

Notasi Kamus Data

SIMBOL

ARTI

=

Terdiri dari

+

Dan

}

Iterasi

[]

Pilihan salah Satu

()

Boleh ada, boleh tidak

* *

Komentar

Contoh Isi Kamus Data

2. Arus Data : *Surat Pengeluaran Barang*

Nama Arus : Sales Order

Alias : SO

Bentuk Data : Dokumen/Cetakan komputer

Arus Data : Pelanggan – Proses pemesanan barang
Proses pemesanan – Supllier

Penjelasan : Untuk mencatat pemesanan barang

Periode : Setiap ada pesanan

Volume : Rata – rata tiap hari adalah 35

Struktur Data <Surat Pesanan Barang> = HEADER + ISI
+ FOOTER

Contoh Isi Kamus Data

HEADER = No_SO + Tanggal + Tgl-Po + No PO Costumer +
Nama_Pelanggan + Alamat + Telp

- No_SO : * Terdiri dari lima belas digit *
- Tanggal : Tanggal + Bulan + Tahun
- Nama_Pelangan : (Titel) + Nama_depan + Nama_belakang
- Alamat : Jalan + Nomor + Kota
- Telepon : * Maksimal 14 digit *

ISI = 1 {KD_Item + Item + Nama_Barang + Satuan + Quantity +
Harga/Unit + Disc (%)
+ Jumlah} 20

- Nomor : * Nomor urut *
- Nama Barang : * Jenis barang yang dipesan *
- Unit : * Maximal tiga digit *
- Harga/unit : * Dalam Rupiah *
- Jumlah : * Dihitung dari unit dikali harga satuan dikurangi discount *

FOOTER = Total




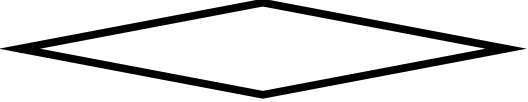


- Total : * Total semua penjualan *

BAGAN TERSTRUKTUR

- ▶ Bagan susunan merupakan susunan hirarki dari modul-modul
- ▶ Bagan susunan menunjukkan
 - pembagian sistem menjadi modul-modul
 - hirarki dan organisasi modul-modul
 - komunikasi antar modul (masukan dan keluaran)
 - nama modul, yang berarti juga fungsi modul.
- ▶ Bagan Susunan tidak menunjukkan :
 - Mekanik didalam modul (seperti ukuran pemanggilan modul lain, *loops* dsb
 - Data internal dari modul

BAGAN TERSTRUKTUR

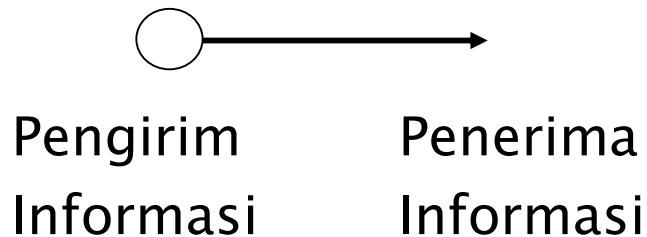
Simbol-simbol yang digunakan :

- 1  : Modul
- 2  : hubungan (Connection)
- 3  : pengulangan (*loop*)
- 4  : Seleksi / pemilihan
- 5  : Kopel Kontrol
- 6  : Kopel Data

BAGAN TERSTRUKTUR

KOPEL DATA (DATA COUPLE) adalah aliran data dari modul yang memanggil ke modul yang dipanggil.

KOPEL KONTROL (CONTROL COUPLE) adalah elemen yang dikirimkan oleh modul yang dipanggil kepada modul yang memanggil sebagai tanda bahwa proses pemanggilan selesai (biasanya untuk proses pengulangan)



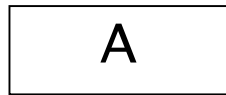
Perbedaan Kopel Data dan Kopel Kontrol :

- Data biasanya berhubungan dengan permasalahan (diproses)
- Kontrol adalah alat bantu untuk implementasi tidak diproses

Contoh Bagan Terstruktur



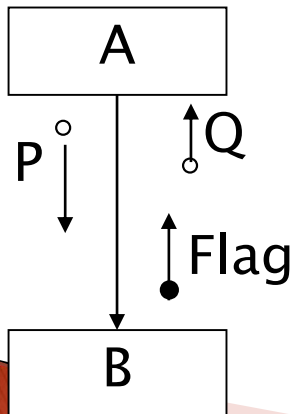
: Suatu modul dengan nama “Hitung Potongan”



A

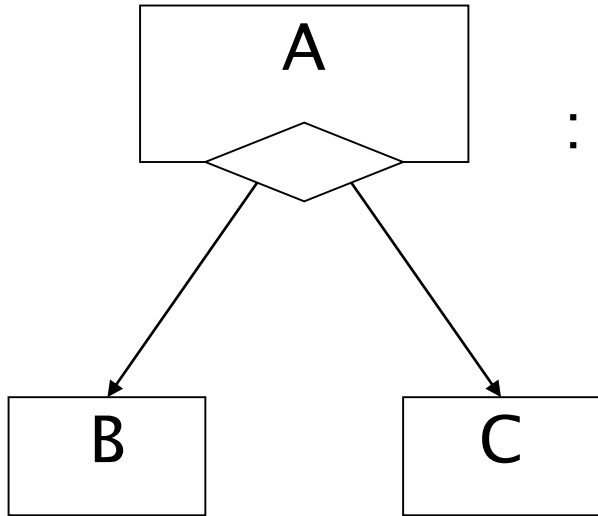
B

: Modul A memanggil modul B, setelah proses modul B selesai maka proses kembali ke modul A

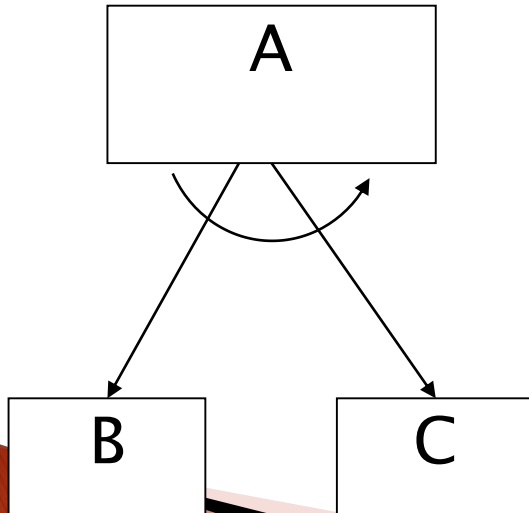


: Modul A memanggil modul B dan elemen data P dikirimkan dari modul A ke modul B. Kemudian modul B mengirimkan hasil proses (*elemen data Q dan elemen kontrol Flag*) ke modul A .

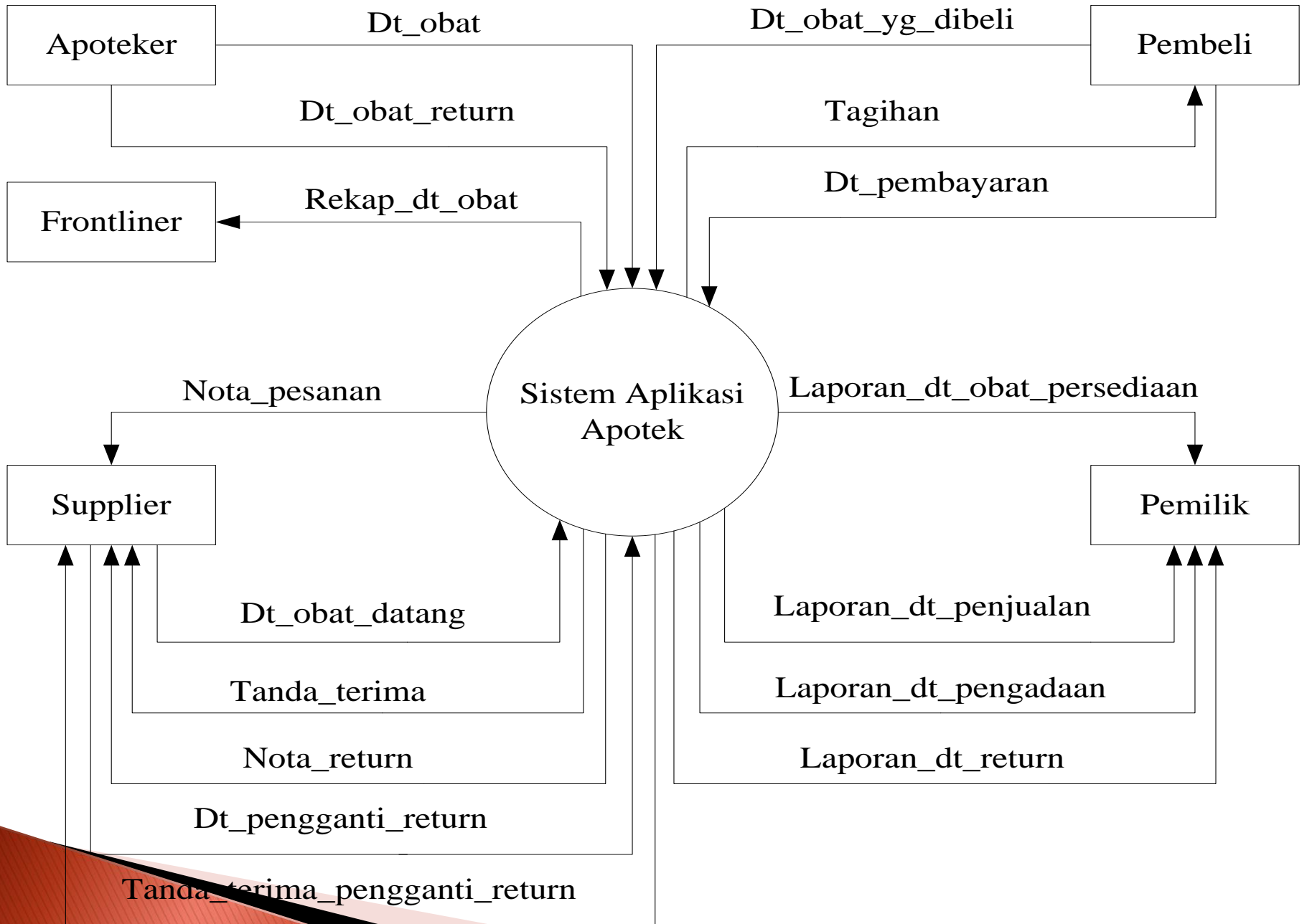
Contoh Bagan Terstruktur

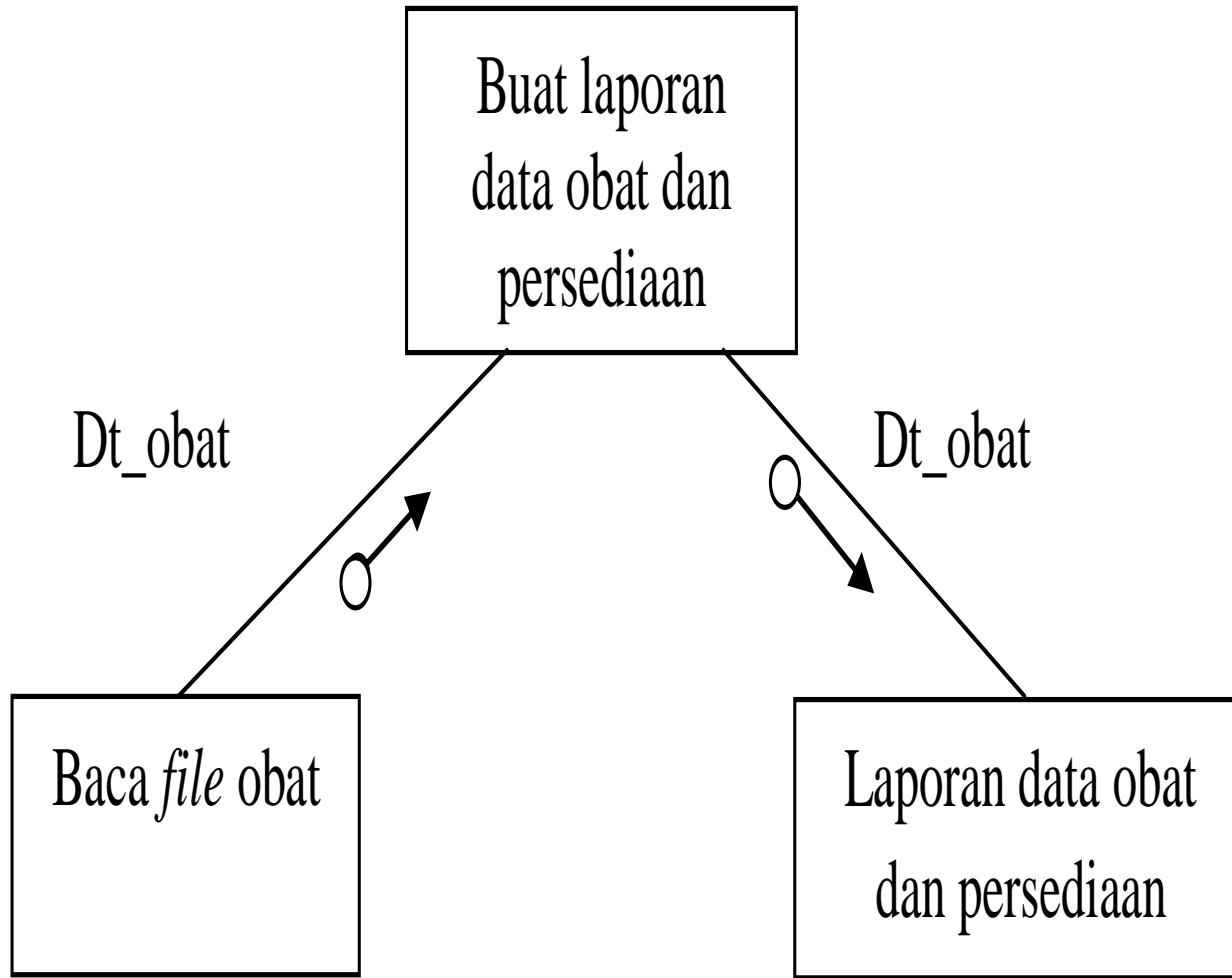


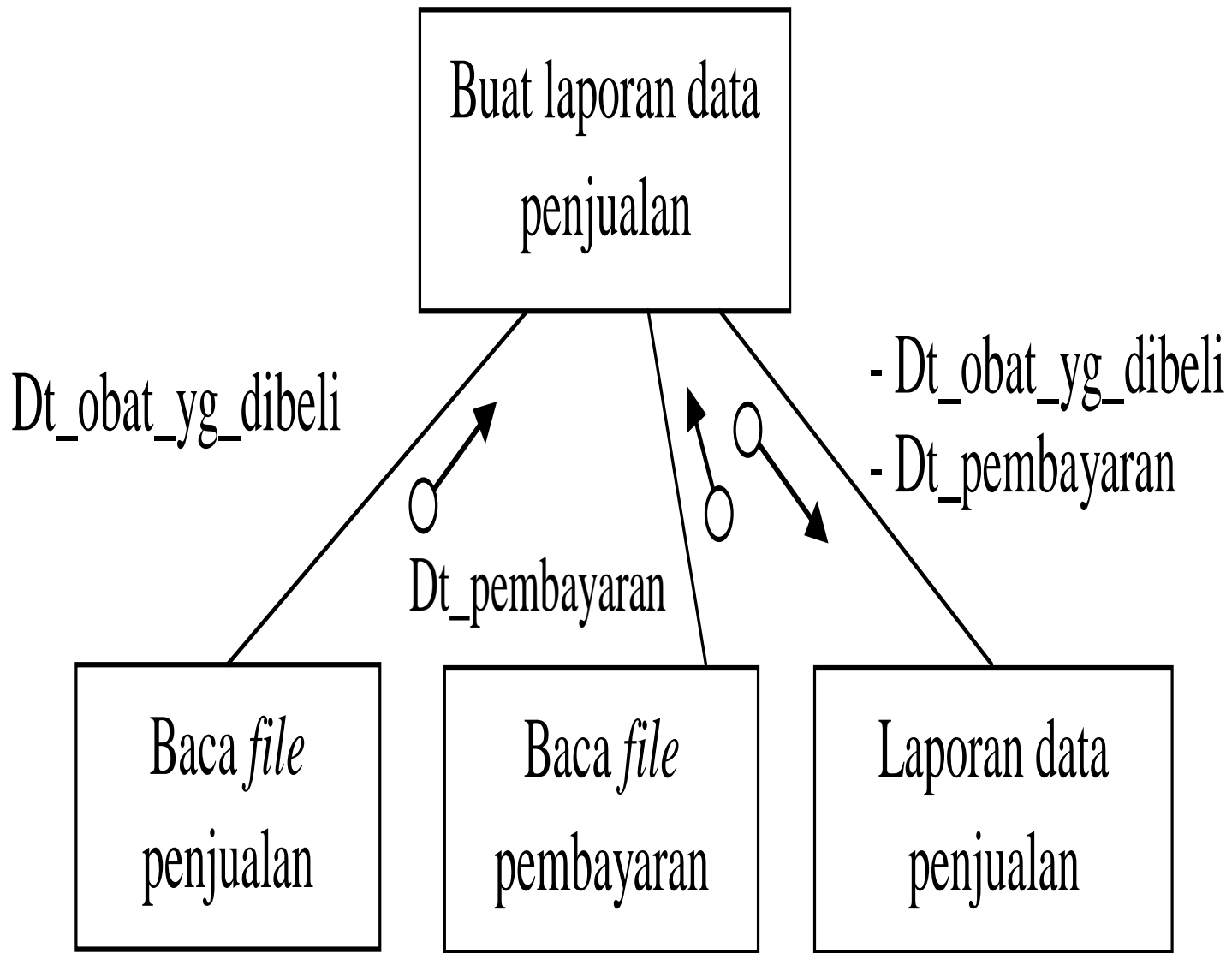
: Modul A memanggil modul B atau C sesuai dengan nilai kondisi yang diseleksi di modul terpenuhi

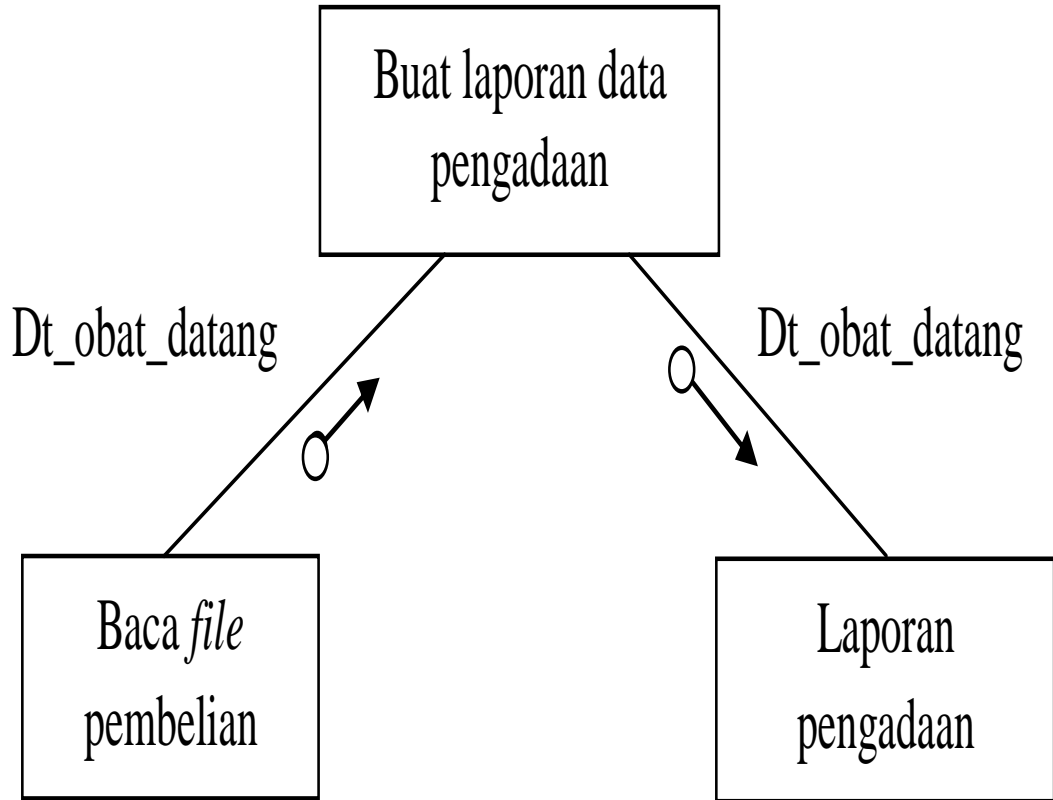


: Modul A memanggil modul B dan C secara berulang









Tugas 4

Deskripsi proses pada sistem pengelolaan rental mobil adalah sebagai berikut : Setiap pelanggan yang akan menyewa mobil mengisi/mengajukan formulir penyewaan kemudian oleh sistem diverifikasi mengenai ketersediaan mobilnya. Jika tercapai kesempatan (mobil tersedia) maka data penyewaan tersebut disimpan. Setelah itu dibuatlah tagihan sewa kepada pelanggan. Setiap terjadi pembayaran sewa dari pelanggan maka datanya disimpan. Setiap periode tertentu dilakukan pembayaran sewa sekaligus laporan kepada pemilik mobil (karena tidak semua mobil yang ada adalah milik pengelola), serta laporan seluruh kegiatan rental kepada manajemen rental.

Dari deskripsi di atas buatlah Diagram Konteks dan Diagram Nol.