

Bab 6: Process Modeling

Analisis dan Perancangan Sistem Informasi

Monica A. Kappiantari - 2009

Source: Whitten, J.L., L.D. Lonnie and K.C. Dittman, Systems Analysis and Design Methods, 6th ed., McGraw-Hill, Boston, 2004.

Sesi 6: Pemodelan Proses

Referensi:

- Whitten, p. 342-399
- www.ait.unl.edu/siau/mgmt454/chapter7-2.ppt
- www.cse.iitb.ac.in/~cs682/lecture/Process-Modeling-DFD-2.ppt
- www.cob.niu.edu/faculty/m10n1r1/omis651/dfd.ppt

Topik:

1. Perbedaan model sistem logik dan fisik
2. Data Flow Diagram
3. Diagram konteks

1. Perbedaan *Logical* & *Physical Models*

Model – gambaran dari realitas

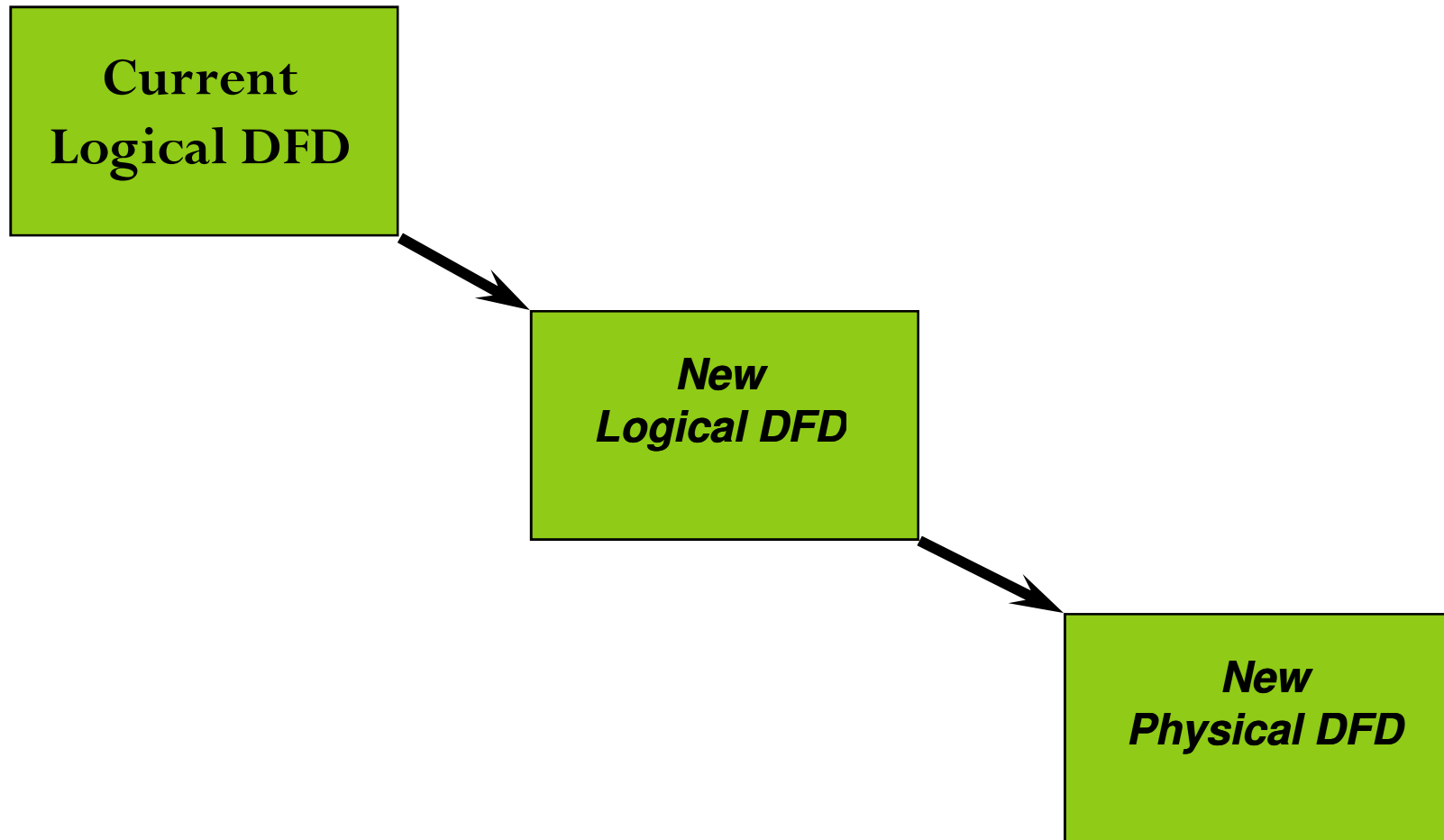
Logical model – sebuah gambaran non-teknis yang melukiskan sebuah sistem.

Physical model – sebuah gambaran teknis yang melukiskan sebuah sistem.

Mengapa menggunakan **sistem logik**?

- Meniadakan bias cara kerja sistem saat ini
- Mengurangi resiko hilangnya *business requirement* karena terlalu terpengaruh dengan hasil-hasil teknis
- Memudahkan komunikasi dengan *end-users* dalam bahasa non-teknis

Progression of Models



2. Data Flow Diagram

Process Modeling and DFDs

Process modeling – teknik yang digunakan untuk mengatur dan mendokumentasikan proses sebuah sistem

- Aliran data melalui proses
- Logik
- Kebijakan (*policies*)
- Prosedur

Data flow diagram (DFD) – proses yang digunakan untuk menggambarkan aliran data melalui sistem dan kerja yang dilakukan oleh sistem

Composite data flow – data flow yang terdiri dari beberapa data flow yang lain

Perbedaan DFD dan Flowchart

DFD

- Proses dapat dilakukan secara paralel (pada saat yang sama)
- memperlihatkan aliran data melalui sebuah sistem
- Proses dapat memiliki satuan waktu yang berbeda (*daily, weekly, on demand*)

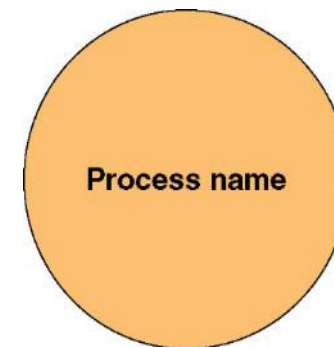
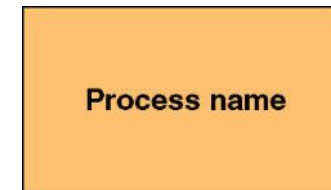
Flowchart

- dilakukan satu per satu (*one at a time*)
- memperlihatkan aliran kontrol (urutan dan pemindahan kontrol)
- Proses memiliki satuan waktu yang konsisten (*a single program with consistent timing*)

Konsep Proses

Process – kerja yang dilakukan oleh sebuah sistem menanggapi aliran data yang masuk atau kondisi-kondisi tertentu.

Sinonimnya adalah *transform*.



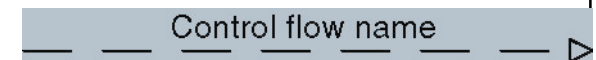
Data Flows & Control Flows

Data flow – data yang merupakan “input ke” atau “output dari” sebuah proses

- *data flow* adalah data yang bergerak
- *Data flow* bisa digunakan untuk menampilkan pembuatan, pembacaan, penghapusan, atau perbaikan (*creation, reading, deletion, or updating*) data dalam sebuah database.



Control flow – sebuah kondisi atau kejadian non-data (*nondata event*) yang memicu sebuah proses



External Agents

External agent – orang, unit organisasi, sistem atau organisasi dari luar yang berinteraksi dengan sistem. Disebut juga *external entity*.

Biasanya merupakan:

- *Office, department, division.*
- *An external organization or agency.*
- *Another business or another information system.*
- *One of your system's end-users or managers*



Gane and Sarson shape



DeMarco/Yourdon shape

Data Stores

Data store – data yang disimpan untuk digunakan kemudian. Sinonimnya adalah *file* dan *database*.

- data store adalah “*data at rest*” dibandingkan dengan data flow yang merupakan “*data in motion*.”
- Biasanya merupakan:
 - *Persons (or groups of persons)*
 - *Places*
 - *Objects*
 - *Events (about which data is captured)*
 - *Concepts (about which data is important)*



Gane and Sarson shape



DeMarco/Yourdon shape

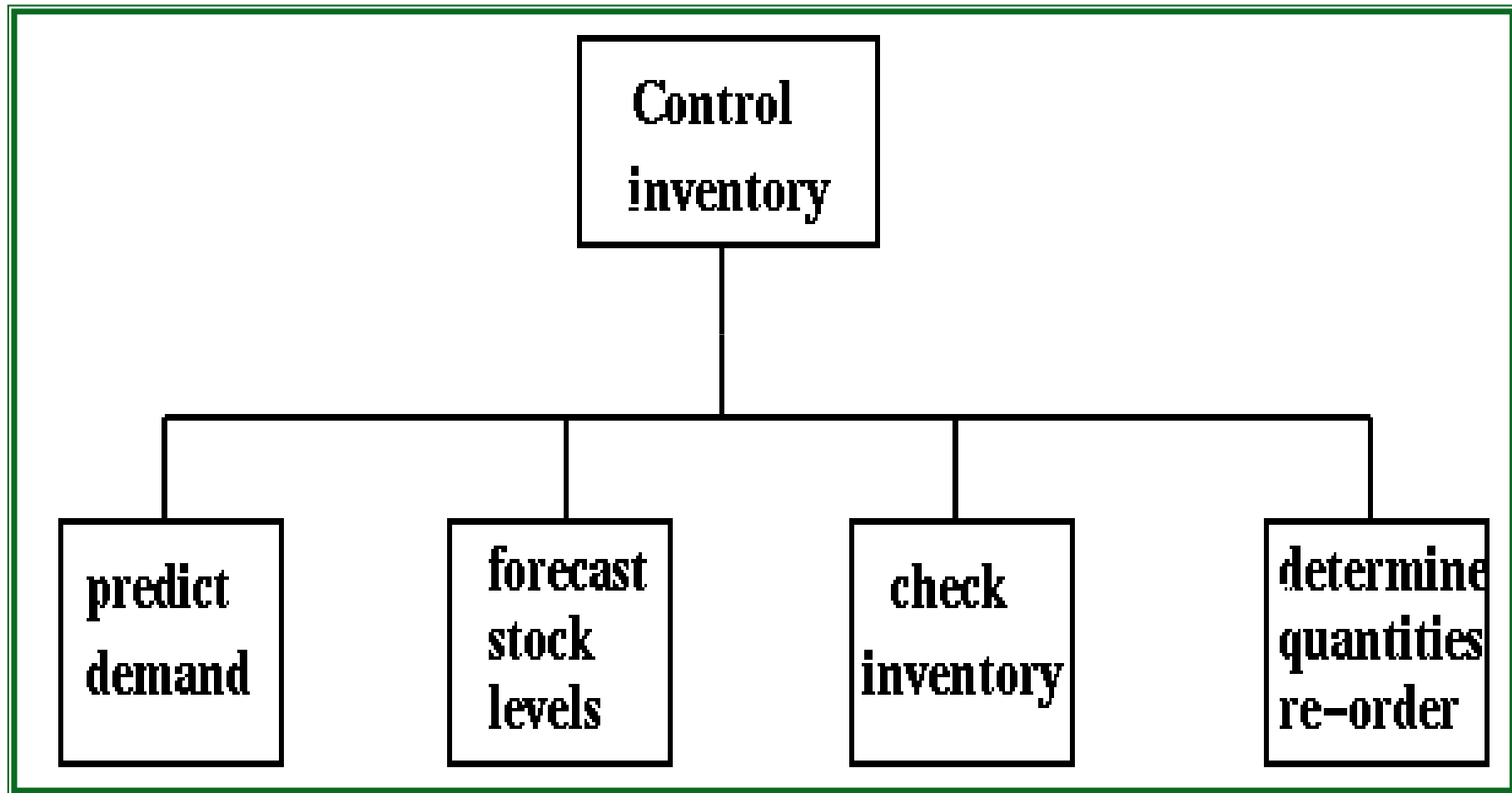
Dekomposisi Proses (*Process Decomposition*)

Decomposition – membagi sistem ke dalam beberapa sub-komponen. Masing-masing tingkat abstraksi menampilkan lebih banyak atau lebih sedikit rincian.

Decomposition diagram –

Alat yang digunakan untuk menggambarkan dekomposisi sebuah sistem. Disebut juga *hierarchy chart*.

Decomposition Diagrams: contoh



Data Types and Domains

Atribut data biasanya didefinisikan melalui tipe dan domain data.

Data type - kelas data yang disimpan dalam sebuah atribut.

- *Character, integers, real numbers, dates, pictures, etc.*

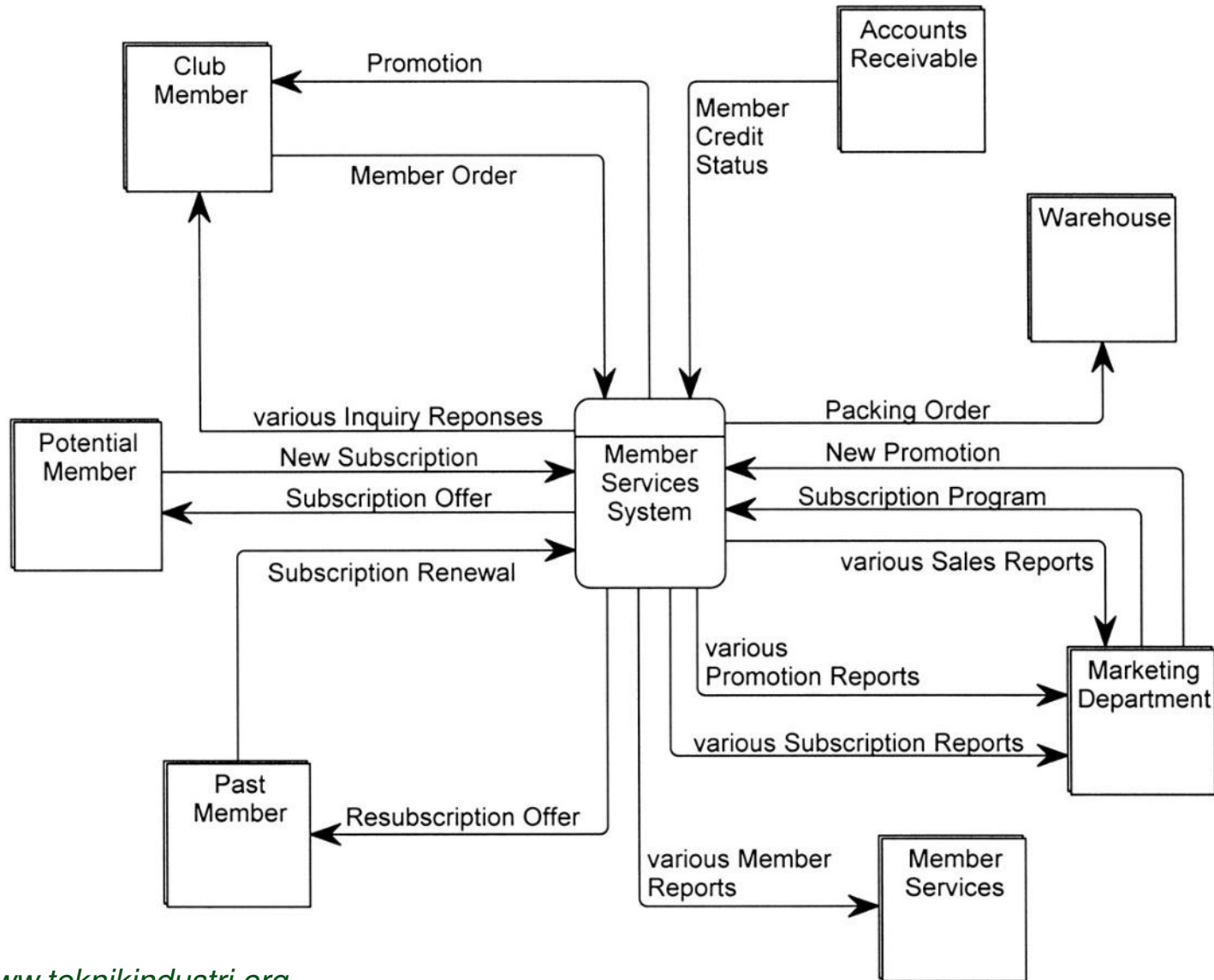
Domain – nilai yang mungkin dari sebuah atribut

3. Diagram Konteks

Diagram konteks

- *Upper-most level, most abstract view of system.*
- *The “outside” view of the system.*
- *Shows a single process, the net inputs and outputs of entire system, and the external agents with which they communicate.*
- *Purpose is to delineate the domain (scope) of the system.*

Contoh Diagram Konteks (Context Diagram)



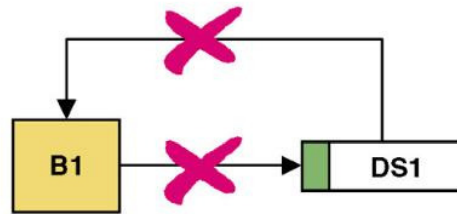
Langkah-langkah pembuatan DFD

1. Buat Diagram konteks (DFD level 0) sebagai gambaran secara umum kerja sistem, gunakan simbol-simbol DFD yang ada
2. Apabila sistem terlalu besar dan rumit, dekomposisikan ke dalam beberapa sub-komponen
3. Buat DFD pada masing-masing sub-komponen tersebut
4. Perhatikan “do and donts” dalam pembuatan DFD

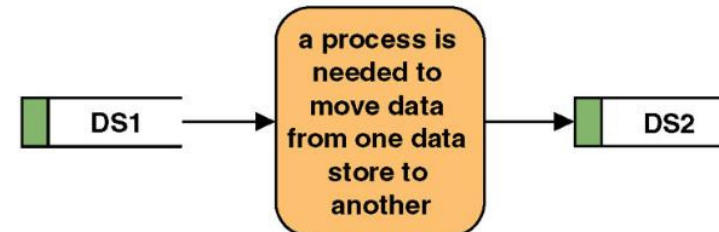
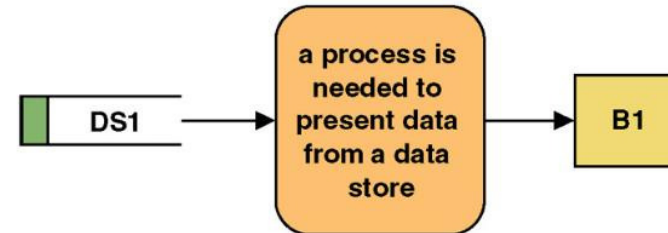
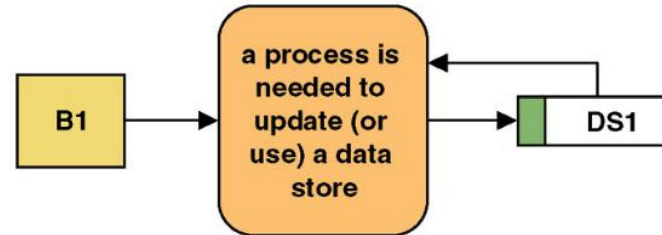
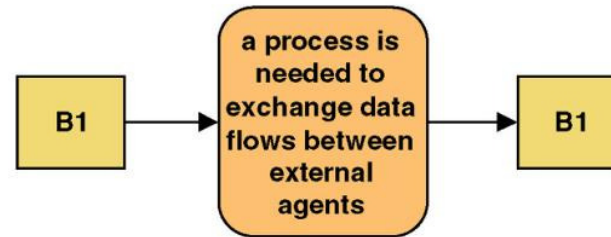
Illegal Data Flows



Illegal data flows



Corrected data flows



Simple Data Flow Diagram

