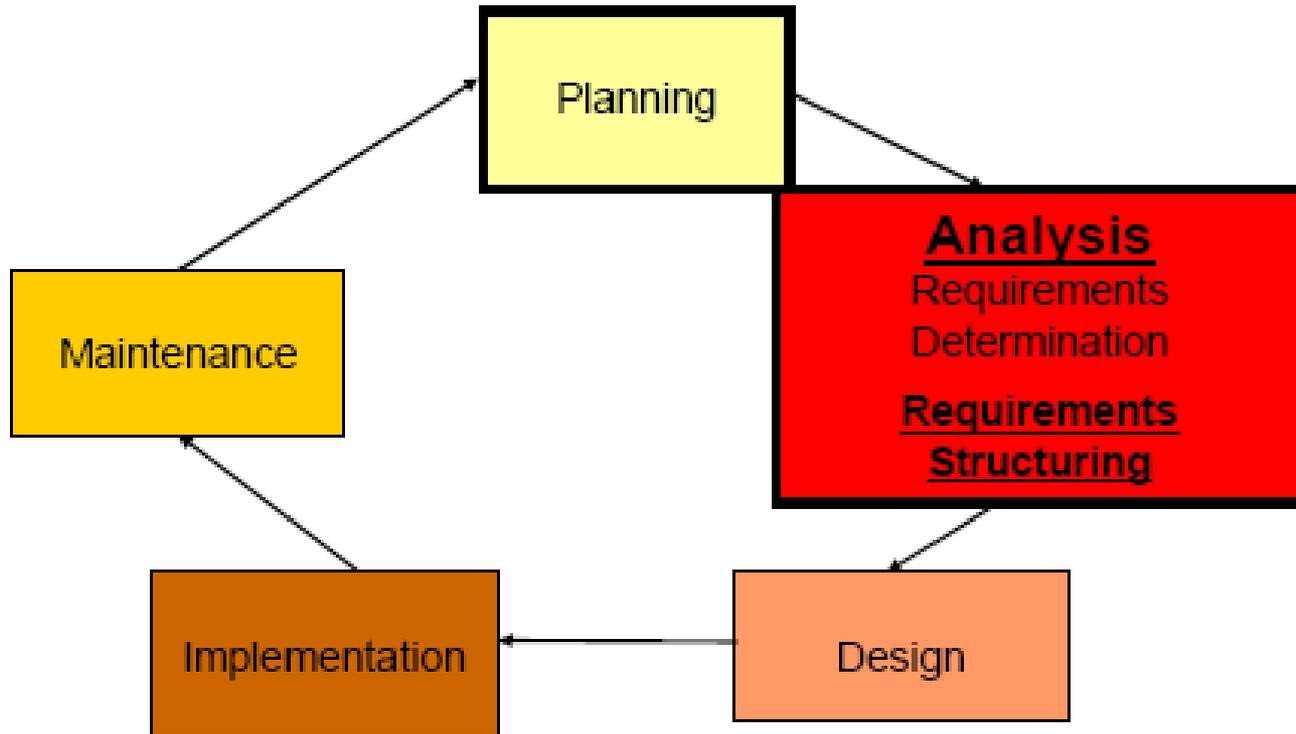


# System Analysis

Sistem dan Teknologi Informasi  
TIP – FTP – UB

# SDLC

(System Development Life Cycle)



# Fase Analisis

## Dua komponen utama dalam fase analisis:

- *Requirement Determination* (Identifikasi kebutuhan)  
Mengidentifikasi kebutuhan user atas sistem
- *Requirement Structuring* (Strukturisasi kebutuhan)  
Menyatakan kebutuhan dalam bentuk terstruktur

Indikator capaian dalam fase ini adalah tersampainya spesifikasi kebutuhan atas sistem dalam bentuk terstruktur

# Strukturisasi Kebutuhan

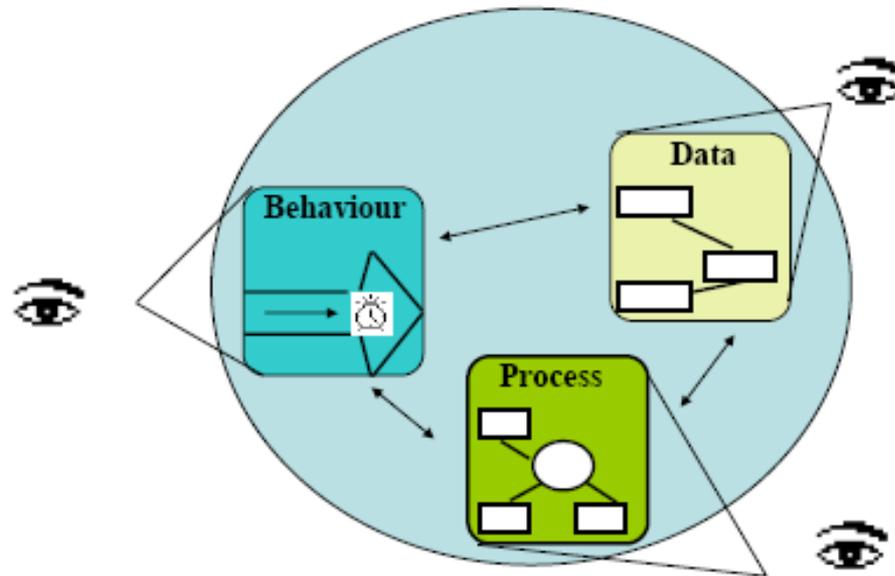
- *Logical system model*
  - Data, E-R (*Entity Relationship*) diagram
  - Proses, DFD (*Data Flow Diagram*)
  - *Interface*, hubungan antar elemen dalam sistem
- *Prototyping*
- *Tools (Diagram, CASE)*

# Model dalam analisis sistem

- Mengapa model ....?
  - untuk pemahaman dan komunikasi
- Simplifikasi
  - setiap model adalah penyederhanaan dunia nyata
  - menghilangkan sebagian dan menekankan di bagian tertentu
- Representasi
  - setiap model adalah sebuah simulasi yang mewakili kenyataan

# Pemodelan sistem informasi

Tiga sudut pandang dalam memodelkan sebuah sistem informasi



# Pemodelan proses

- Proses adalah bagian aksi sebenarnya dalam sebuah bisnis
- Dua tipe pemodelan proses:
  - Diagram dekomposisi fungsi
  - Diagram aliran data (DFD)
- Pemodelan proses dilakukan dengan menggunakan informasi yang diperoleh selama fase *requirement determination*

# Pemodelan proses dengan DFD

## DFD:

Pernyataaan grafis dari proses-proses yang .....

Menghimpun, memanipulasi, menyimpan dan mendistribusikan

**data**

antara sebuah sistem dan lingkungannya serta antar komponen dalam sebuah sistem

# DFD

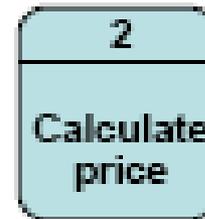
- Menggambarkan aliran data masuk, di dalam, dan keluar sebuah sistem informasi
- Menggambarkan sistem informasi sebagai sebuah jaringan proses yang saling berkomunikasi, antara lain:
  - proses yang mentransformasi data
  - pergerakan data antara proses-proses
- Keuntungan: sudah terkenal dan mudah dipahami

# Tipe-tipe DFD

- **Context DFD (diagram konteks)**
  - level tertinggi yang menunjukkan cakupan sistem
- **DFD sistem yang sedang berjalan**
  - Membantu analisis memahami sistem yang sedang berjalan
- **DFD sistem yang diusulkan**
  - menunjukkan aliran data, struktur, dan kebutuhan fungsional sistem baru yang diusulkan

# Komponenten DFD

Processes



Data flows



Data stores



Sources/ sinks  
(external agents)



# Process

- Proses
  - Menyatakan kerja atau aksi yang dilaksanakan pada data (didalam sistem)
  - Mengubah aliran data masuk menjadi aliran data keluar
  - Memiliki nomor dan nama yang *unique*



# Nama proses

- Setiap proses dinamai dengan menggunakan kata kerja (verb) atau frase kata benda (noun phrase), contoh
  - menghitung harga
  - validasi data konsumen
  - menerima pengiriman supplier
- Nama proses harus menggambarkan dengan jelas apa yang dilakukan proses tersebut.
  - hindari penggunaan nama yang ambigu, seperti “memproses data”
- Nomor proses adalah sebuah pengenal dan tidak menunjukkan urutan-urutan proses

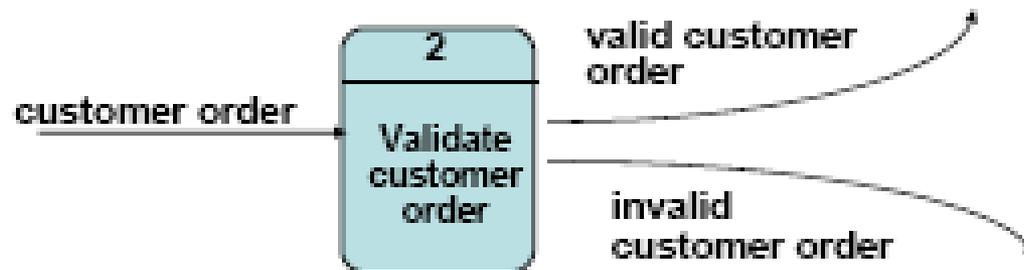
# Data flow (aliran data)

- Untuk menyatakan aliran data digunakan simbol panah
- Dilakukan dengan mendeskripsikan paket data (data yang berpindah/berubah bersamaan)
- Sebuah panah mungkin berisi beberapa data yang saling berhubungan dan bergerak bersama menuju tujuan yang sama



# Nama data flow

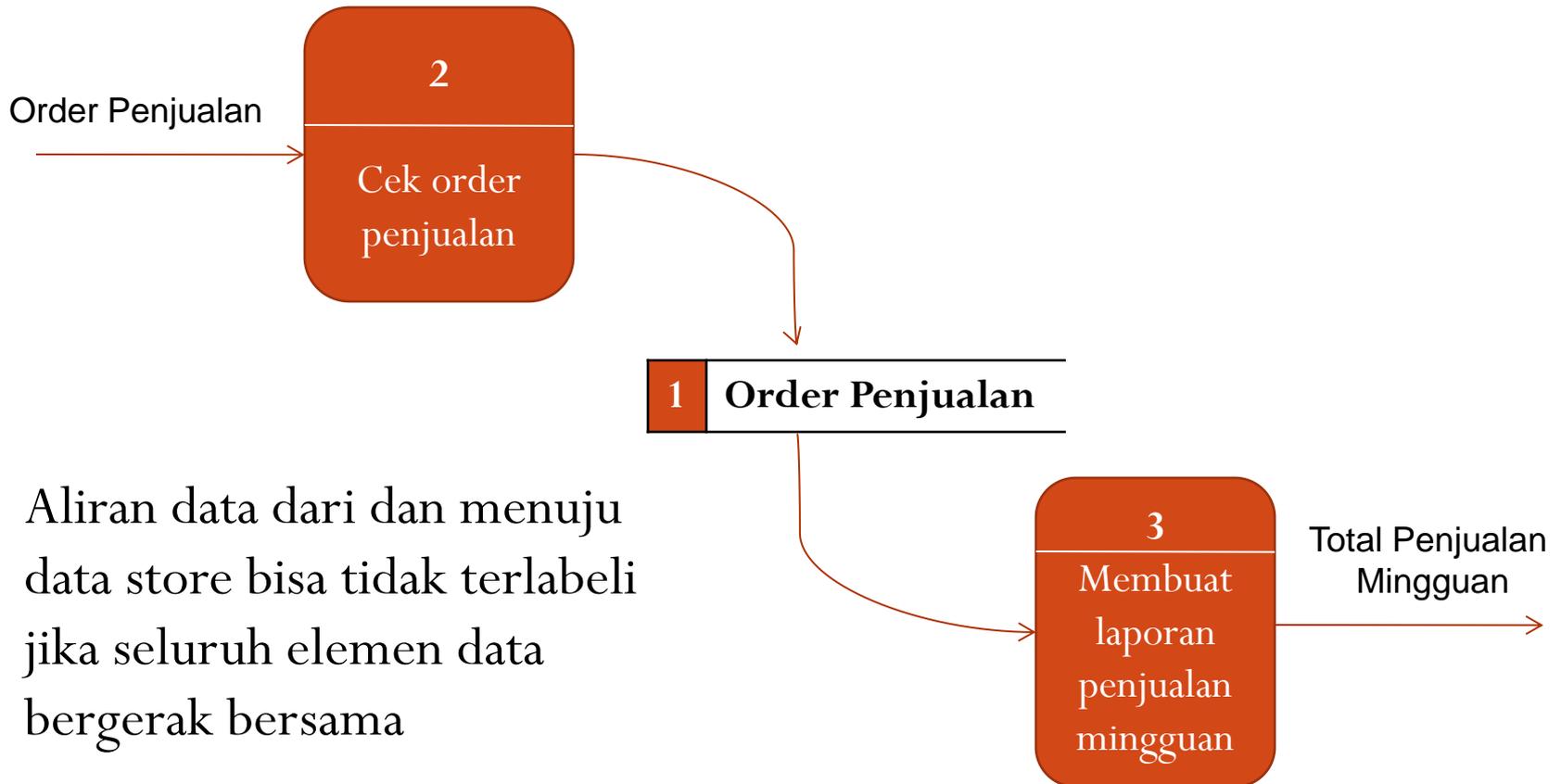
- Setiap aliran data diberi nama dengan kata benda (noun) atau frase kata benda (noun phrase)
- Nama harus menggambarkan isi dan menyertakan sebanyak mungkin informasi tentang aliran data



# Data store (Penyimpanan data)

- Data store:
  - Menyatakan kumpulan aliran data saat tidak berpindah
  - Memiliki nama unik yang harus menyatakan isi dari data store
  - Bisa mengandung beberapa data yang disimpan di beberapa lokasi penyimpanan (hard disk)
  - Dapat berupa penyimpanan temporal atau permanen

# Data store



Aliran data dari dan menuju data store bisa tidak terlabeli jika seluruh elemen data bergerak bersama

# Aliran data dari & ke data store

- Aliran data dari sebuah proses ke sebuah data store berarti *update* (perubahan), antara lain *insert* (masukkan), *delete* (hapus) atau *change* (ubah)
- Aliran data dari sebuah data store ke sebuah proses berarti penggunaan, yaitu *retrieve* (ambil) dan *use* (gunakan)
- Label aliran data seharusnya berupa frase kata benda (*noun phrase*)

# Source & Sink (External agent)

- Sebuah *source/sink*:
  - Entitas eksternal yang menjadi asal atau tujuan data saat meninggalkan sistem
  - Sebuah entitas di lingkungan di luar sistem yang berkomunikasi dengan sistem
- Sebuah *external agent* adalah *source* jika ia merupakan sumber data yang masuk ke dalam sistem
- Sebuah *external agent* adalah *sink* jika ia merupakan tujuan data yang keluar dari sistem

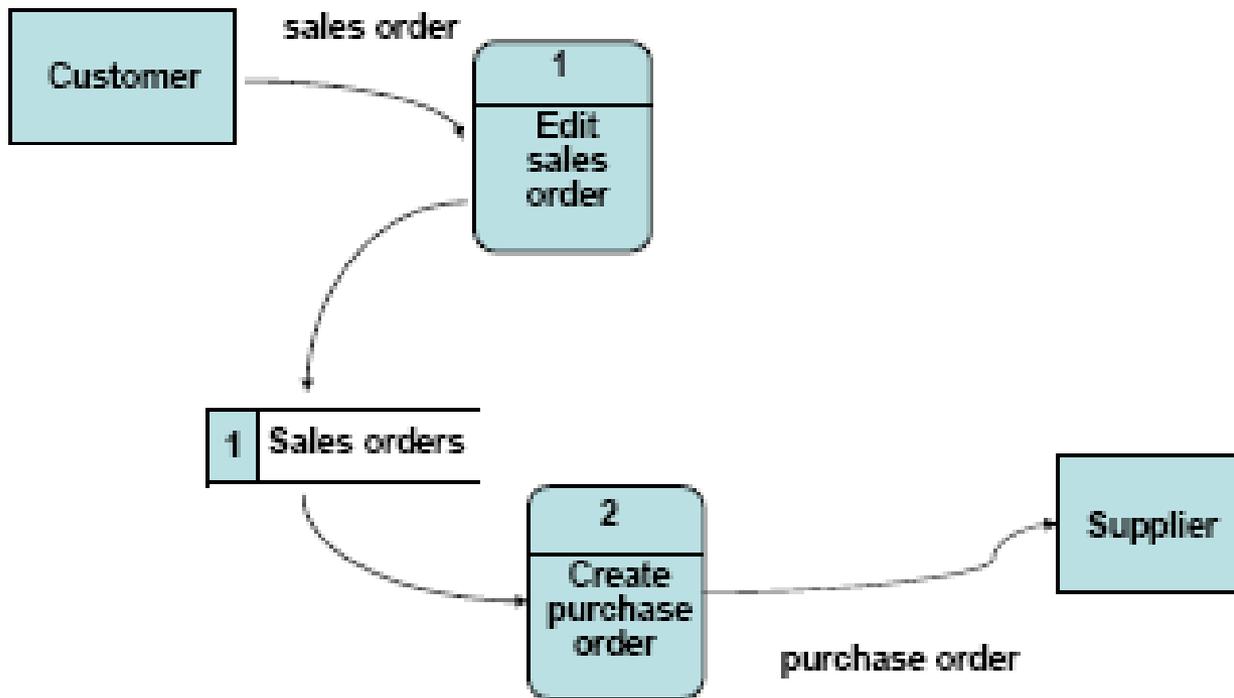


Supplier

# Source & Sink (External agent)

- Aliran data yang menghubungkan *source/sink* dengan proses di dalam sistem adalah *interface* antara sistem dan lingkungannya
- *Source* dan *sink* berada di luar sistem dan menjadi pembatas antara sistem dan lingkungannya
- Apa yang dilakukan sink terhadap data yang ia terima dari sistem dan bagaimana *source* mendapatkan data yang ia masukkan ke sistem adalah **di luar batas sistem**, sehingga seharusnya tidak ditampilkan dalam sebuah DFD

# DFD

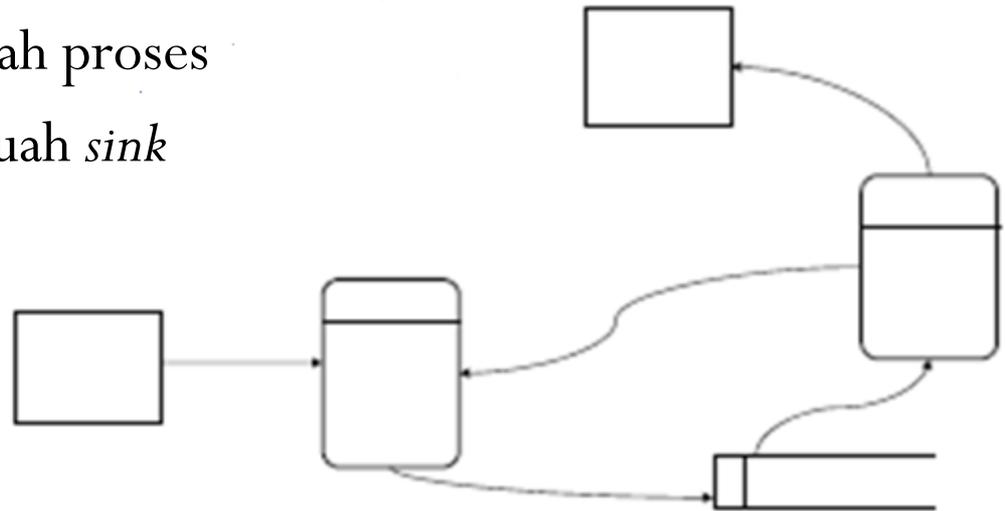


# Menggambar DFD

- Setiap *object* dalam DFD harus memiliki nama yang *unique*
- Setiap proses harus memiliki sedikitnya satu aliran data masuk (*input*) dan satu aliran data keluar (*output*)
- Input untuk sebuah proses berbeda dengan output dari proses tersebut
- Sebuah proses harus dapat memberikan output hanya berdasarkan informasi dari aliran data yang masuk kepadanya dan informasi konstan yang ada

# Menggambar DFD

- Aliran data diperbolehkan:
  - Antara dua proses
  - Dari sebuah *data store* ke sebuah proses
  - Dari sebuah proses ke sebuah *data store*
  - Dari sebuah *source* ke sebuah proses
  - Dari sebuah proses ke sebuah *sink*



# Menggambar DFD

- Aliran data **tidak** diperbolehkan:
  - Antara dua *external agent*
  - Antara dua *data store*
  - Dari sebuah *external agent* ke sebuah *data store*

