

Pertemuan 10

METODE DESAIN (1)

Pokok Bahasan dalam RPL :

- ❖ Desain Data
- ❖ Desain Arsitektur
- ❖ Proses Desain Arsitektur
- ❖ Pasca Pemrosesan Desain
- ❖ Optimasi Desain Arsitektur

Buku Referensi :

- Pressman, RS., 2008, Software Engineering: A Practitioner's Approach, New York: McGraw-Hill
- Sommerville, I, 2007, Software Engineering, Addison Wesley

Tujuan Metode Desain

- Menjelaskan maksud dari arsitektur PL dan kenapa sangat penting.
- Memahami model data, struktur data, database, data warehouse, desain data pada level komponen

DESAIN DATA

- Aktivitas pertama (dan beberapa sering mengatakan yang terpenting) dari 4 aktivitas desain yang dilakukan selama RPL.
- Prinsip-prinsip :
 1. Prinsip analisis sistematis yang diaplikasikan pada fungsi dan perilaku seharusnya diaplikasikan juga pada data.
 2. Semua struktur data dan operasi yang dilakukan pada masing-masing struktur data harus diidentifikasi.
 3. Kamus data harus dibangun dan digunakan untuk menentukan baik data maupun desain program.
 4. Keputusan desain data tingkat rendah harus ditunda sampai akhir proses desain.
 5. Representasi struktur data hanya boleh diketahui oleh modul-modul yang harus menggunakan secara langsung data yang diisikan didalam struktur tersebut.
 6. Pustaka struktur data dan operasi yang berguna yang dapat diaplikasikan pada struktur data tersebut harus dikembangkan.
 7. Desain PL dan bahasa pemrograman harus mendukung spesifikasi dan realisasi dari tipe-tipe data abstrak.

DESAIN ARSITEKTUR

- Untuk mengembangkan struktur program modular dan merepresentasikan hubungan kontrol antar modul.
- Membentuk struktur program dan struktur data dengan menentukan interface yang memungkinkan data mengalir melalui program.
- Kontributor
Perintis desain PL yang didasarkan pada aliran data melalui sebuah sistem.
- Area Aplikasi
luasnya aplikasi dimana aplikasi dapat diaplikasikan.

PROSES DESAIN ARSITEKTUR

Langkahnya :

1. Tipe aliran informasi dibangun
2. Batas aliran diindikasikan.
3. DFD dipetakan kedalam struktur program.
4. Hierarki kontrol ditentukan dengan pemfaktoran.
5. Struktur resultan disaring/diperhalus dengan menggunakan pengukuran desain dan heuristik.

PROSES DESAIN ARSITEKTUR (lanjutan)

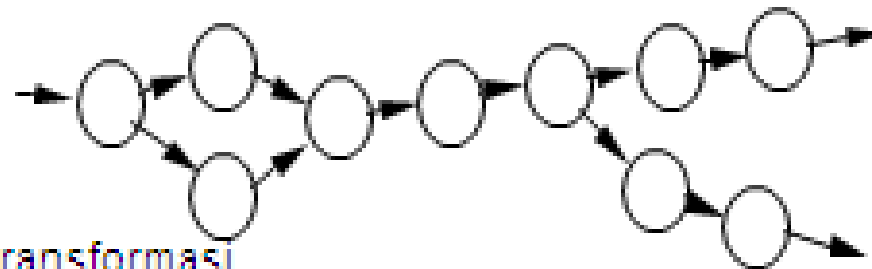
- Aliran Transformasi

Keseluruhan aliran data terjadi dalam cara yang berurutan dan mengikuti satu atau hanya beberapa jalur “garis lurus”, bila segmen dari aliran data menunjukkan karakteristik tersebut, maka disitu ada aliran transformasi.

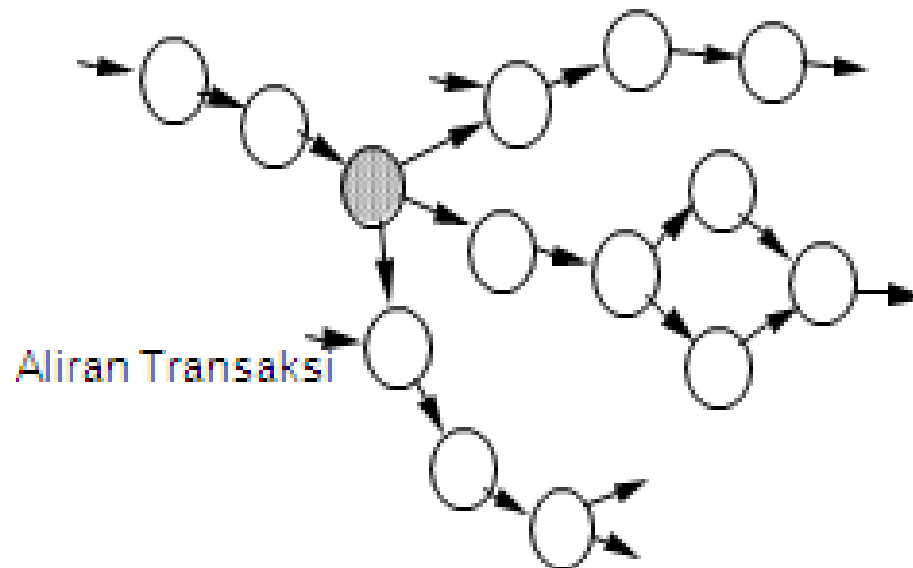
- Aliran transaksi

Ditandai dengan pergerakan data sepanjang jalur masuk yang mengkonversikan informasi dunia eksternal ke dalam suatu transaksi.

Karakteristik Aliran



Aliran Transformasi

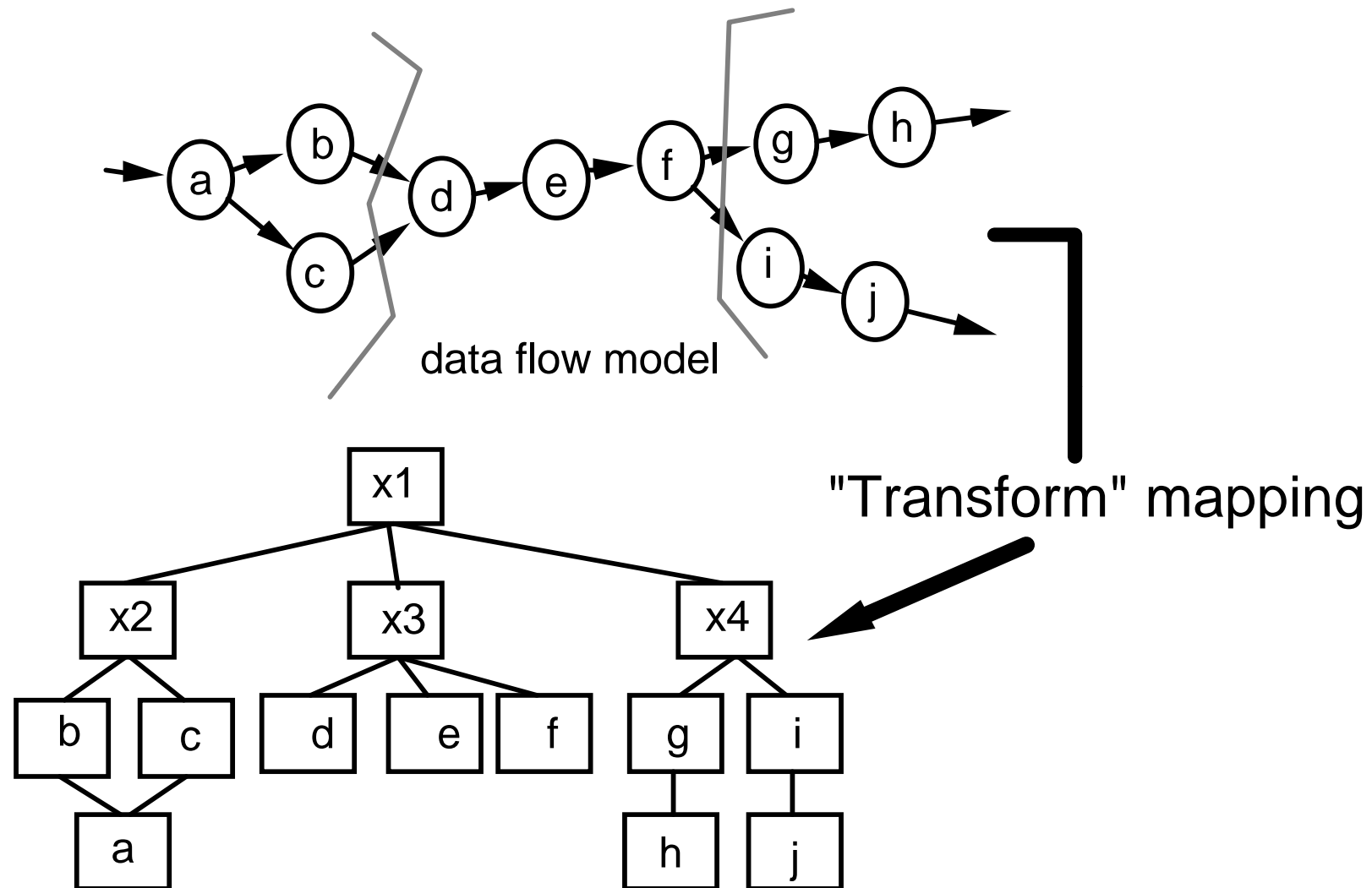


Aliran Transaksi

PEMETAAN TRANSFORMASI

- Serangkaian langkah desain yang memungkinkan sebuah DFD dengan karakteristik aliran transformasi untuk dipetakan ke dalam tempalte yang telah ditentukan untuk struktur program.
- Langkah-langkah :
 1. Kaji model sistem fundamental.
 2. Kaji dan saring diagram aliran data untuk PL.
 3. Tentukan apakah DFD memiliki karakteristik aliran transformasi atau transaksi.
 4. Isolasi pusat transformasi dengan mengkhususkan batas aliran masuk dan keluar.
 5. Lakukan “pemfaktoran tingkat pertama”.
 6. Lakukan “pemfaktoran tingkat kedua”.
 7. Saringlah struktur program iterasi pertama dengan menggunakan heuristik desain bagi kualitas perangkat lunak yang telah ditingkatkan.

PEMETAAN TRANSFORMASI (LANJUTAN)

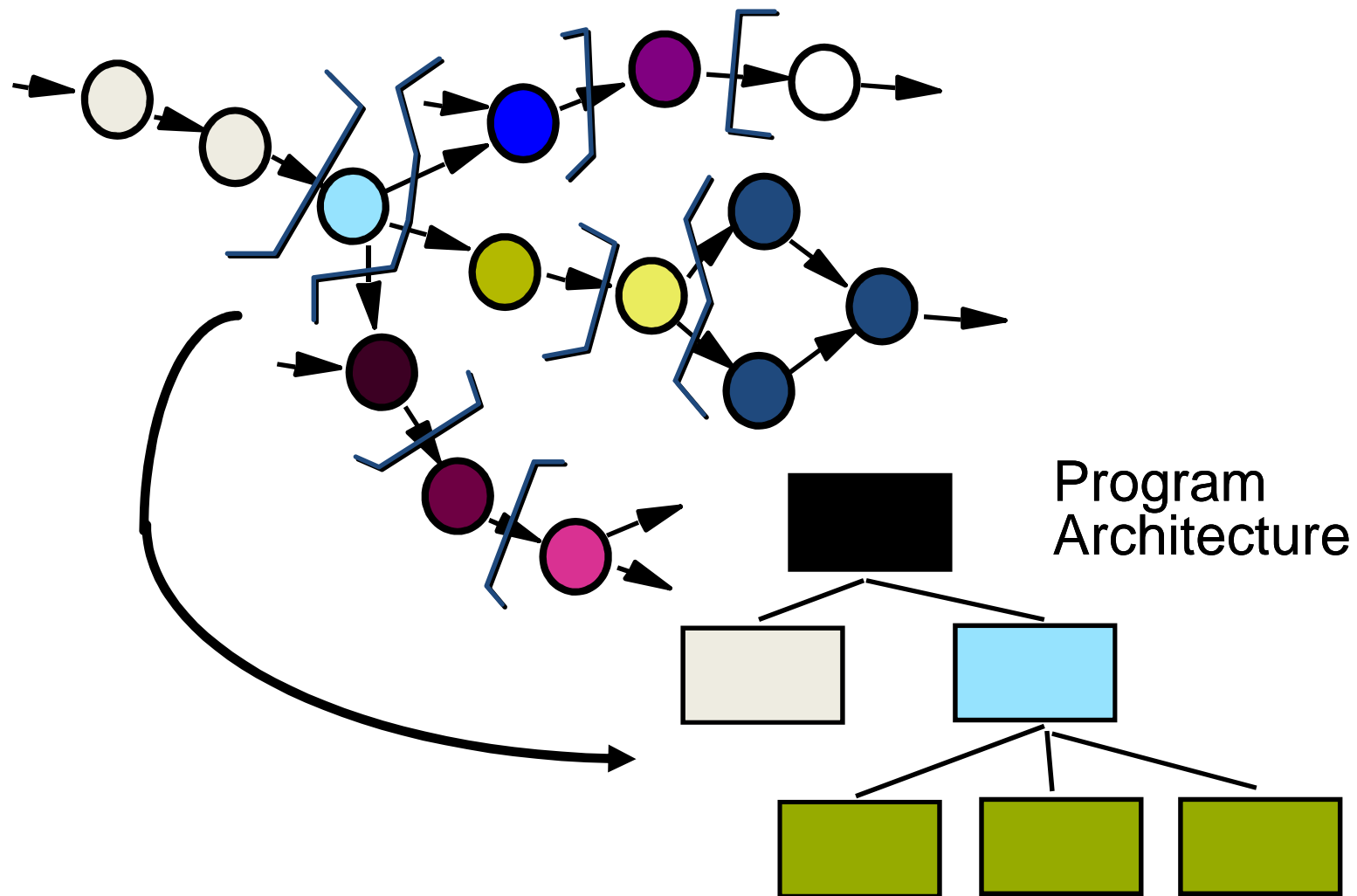


PEMETAAN TRANSAKSI

Langkah-langkah desain :

1. Kaji model sistem fundamental.
2. Kaji dan saring diagram aliran data untuk PL.
3. Tentukan apakah DFD memiliki karakteristik aliran transformasi atau transaksi.
4. Identifikasi pusat transaksi dan karakteristik aliran sepanjang masing-masing jalur aksi.
5. Petakan DFD pada sebuah struktur program yang sesuai dengan pemrosesan transaksi.
6. Faktorkan dan saringkan struktur transaksi dan struktur masing-masing jalur aksi.
7. Saring struktur program iterasi pertama dengan menggunakan heuristik desain untuk kualitas PL yang dikembangkan.

PEMETAAN TRANSAKSI (lanjutan)



PASCA PEMROSESAN DESAIN

Tugas yang harus dilakukan :

- Mengembangkan narasi pemrosesan untuk masing-masing modul.
- Menyediakan deskripsi interface untuk masing-masing modul.
- Menentukan struktur data lokal dan global.
- Mencatat semua batasan desain.
- Mengkaji desain.
- Mempertimbangkan “optimasi” (bila diperlukan dan dibenarkan).

OPTIMASI DESAIN ARSITEKTUR

Pendekatan :

1. Kembangkan dan saring struktur program tanpa memperatikan optimasi kinerja-kritis.
2. Gunakan Peranti CASE yang mensimulasi kinerja run-time untuk mengisolasi area inefisiensi.
3. Selama iterasi desain selanjutnya, pilihlah judul yang dicurigai “time-hot” dan dengan hati-hati kembangkanlah prosedur (algoritma) untuk efisiensi waktu.
4. Kodekan sebuah bahasa pemrograman yang sesuai.
5. Instrumentasikan PL untuk mengisolasi modul yang menjelaskan utilisasi proses yang berat.
6. Bila perlu, desain ulang atau kodekan kembali bahasa yang tergantung pada mesin untuk meningkatkan efisiensi.