INTERAKSI MANUSIA DAN MESIN

Ragam Dialog



Budhi Irawan, S.Si, M.T

RAGAM DIALOG

- <u>Dialog</u> dalam arti umum adalah *Percakapan* antara dua kelompok atau lebih
- <u>Dialog</u> dalam konteks perancangan user interface adalah *Struktur dari* percakapan antara user dan sistem komputer
- Ragam dialog (Dialoque Style) adalah "Cara yang digunakan untuk mengorganisasikan berbagai teknik dialog"

SIFAT RAGAM DIALOG

a) Inisiatif

Pengguna memberikan tanggapan atas permintaan yang diberikan oleh komputer untuk memasukkan perintah atau parameter perintah

b) Keluwesan

Pengguna mempunyai *kesempatan melakukan customizing* dan memperluas antarmuka dari sebuah sistem untuk memenuhi kebutuhan.

SIFAT RAGAM DIALOG

c) Kekuatan

Kekuatan didefinisikan sebagai *jumlah pekerjaan yang dapat dilakukan oleh* sistem untuk setiap perintah yang diberikan oleh pengguna.

d) Kompleksitas

Pengelompokan dalam menerapkan model yang diinginkan pengguna ke dalam sistem, dan hal ini dapat diperoleh dengan menggunakan hirarkhi atau ortogonalitas atau keduanya.

SIFAT RAGAM DIALOG

e) Beban Informasi

Dalam hal ini menitik beratkan pada *penyajian informasi yang dihasilkan komputer kepada pengguna*. Agar penyampaian informasi itu dapat berdaya guna dan beban informasi yang terkandung di dalam suatu ragam dialog seharusnya disesuaikan dengan kebutuhan pengguna.

KARAKTERITIK RAGAM DIALOG

1) Konsistensi

Sistem yang konsisten akan mendorong pengembangan mentalitas dengan memberikan petunjuk kepada pengguna untuk mengekstrapolasi pengetahuan yang ia miliki untuk memahami perintah yang baru lengkap dengan pilihan yang ada.

2) Umpan Balik

Pada ragam dialog jika pengguna melakukan kesalahan komputasi, maka program akan menampilkan suatu pesan kesalahan.

KARAKTERITIK RAGAM DIALOG

3) Observabilitas

Sistem dikatakan mempunyai sifat observabilitas apabila *sistem itu berfungsi* secara benar dan nampak sederhana bagi pengguna, meskipun sesungguhnya pengolahan secara internalnya sangat rumit.

4) Kontrolabilitas

Kontrolabilitas merupakan kebalikan dari observabilitas, dan hal ini berimplikasi bahwa *sistem selalu berada di bawah kontrol pengguna*. Agar hal ini tidak tercapai, antarmukanya harus mempunyai sarana yang memungkinkan pengguna untuk dapat melakukan kendali.

KARAKTERITIK RAGAM DIALOG

5) Efisiensi

Efisiensi melibatkan unjuk kerja manusia dan komputer secara bersama-sama adalah throughput yang diperoleh dari kerjasama tersebut.

6) Keseimbangan

Strategi yang diambil dalam perancangan sistem manusia-komputer haruslah dapat membagi-bagi pekerjaan antara manusia dan komputer seoptimal mungkin.

1. Dialog berbasis perintah tunggal (Command line dialogue).

"Perintah-perintah tunggal yang dioperasikan tergantung dengan sistem operasi komputer yang dipakai ".

contoh:

DOS (dir, delete, format, copy, dll)

Unix / Linux (ls, vi, who, passwd, dll)

- 2. Dialog berbasis bahasa pemrograman (*Programming language dialogue*).
 - "Dialog yang dikemas sejumlah perintah ke dalam suatu bentuk berkas (file) berupa batch file".
- 3. Antarmuka berbasis bahasa alami (*Natural language interface*).
 - "Dialog yang berisikan instruksi-instruksi dalam bahasa alami (manusia) yang diterjemahkan oleh sistem penterjemah".

4. Sistem Menu

"Dialog yang menampilkan daftar sejumlah pilihan dalam jumlah terbatas"

Sistem Menu terbagi menjadi 2 yaitu:

- a. Sistem Menu Datar
 - Selektor pilihan
 - Penggunaan Tanda terang (highlight marker)
- b. Sistem Menu Tarik (Pulldown)

5. Dialog berbasis pengisian formulir (Form filling dialogue).

"Dialog dimana pengguna (user) dihadapkan ke suatu bentuk formulir dilayar komputer yang berisi sejumlah pengisian data dan opsi (option) yang telah ditentukan".

6. Antarmuka berbasis ikon (icon)

"Dialog yang menggunakan simbol atau tanda untuk menunjukan suatu pilihan aktifitas tertentu".

7. Sistem Penjendelaan

"Sistem antarmuka yang memungkinkan pengguna untuk menampilkan berbagai informasi pada satu atau lebih jendela (windows) ".

Manipulasi Langsung.

"Penyajian langsung aktifitas kepada pengguna (user) sehingga aktifitas akan dikerjakan oleh komputer ketika pengguna memberikan instruksi langsung yang ada pada layar komputer".

9. Antarmuka Berbasis Interaksi Grafis.

"Dialog berbentuk pesan atau informasi pada suatu gambar atau *link* yang tampil ketika pengguna melakukan suatu aktifitas".

MANFAAT DIALOG BERBASIS JENDELA

Ada beberapa fungsi yang dapat dimanfaatkan oleh dengan adanya **Dialog berbasis Jendela** ini, yaitu :

- a. Lebih *banyak informasi* yang bisa ditampilkan.
- b. Kemungkinan memasukkan *lebih dari sebuah sumber informasi*
- c. mengkombinasikan *sejumlah sumber informasi yang berbeda*
- d. Pengendalian *secara bebas dari setiap program yang ada*
- e. Sebagai saran untuk mengingatkan, bisa digunakan sebagai sarana untuk penyajian berganda.

JENIS – JENIS JENDELA (WINDOWS)

- a) Jendela TTY (TeleTYpe atau TeleTYpewriter)
- b) Time-Multiplexed Windows
- c) Space Multiplexed Windows

a) JENDELA TTY

- Jendela TTY merupakan jenis jendela yang paling sederhana yang mana secara sekilas jenis jendela ini mirip dengan tampilan apa adanya karena jendela TTY hanya terdiri atas sebuah jendela yang mempunyai fasilitas pemindahan halaman (scrolling) secara otomatis pada satu arah.
- Contoh sederhana dari jendela TTY adalah jendela (tampilan) pada saat Anda berada pada dot prompt.

b) TIME MULTIPLEXED WINDOWS

- Pemikiran yang mendasari digunakannya istilah time-multiplexed windows adalah bahwa layar tampilan merupakan sumber daya yang bisa digunakan secara bergantian oleh sejumlah jendela pada waktu yang berlainan.
- Jenis jendela ini banyak diterapkan pada editor teks

c) SPACE MULTIPLEXED WINDOWS

- Dalam space-multiplexed windows, lebar layar dibagi bagi menjadi beberapa jendela dengan ukuran yang bervariasi, dan jenis jendelanya dapat ditentukan berdasarkan ketergantungan antara satu bisa diletakkan "diatas" jendela yang lain, dan apakah masing-masing jendela bisa diubah ukurannya.
- Contoh sistem jendela pada perangkat lunak Bravo, pada pengolah kata Word Perfect versi DOS misalnya Word Perfect versi 5.0 atau5.1, pada Lotus 123 versi DOS

LEVEL DIALOG

1) Leksikal

- Merupakan tingkat yang paling rendah
- Yaitu bentuk icon pada layar
- Pada bahasa manusia, ekuivalen dengan bunyi dan ejaan suatu kata

2) Sintaksis

- Yaitu urutan dan struktur dari input dan output
- Pada bahasa manusia, ekuivalen dengan grammar suatu kalimat

3) Semantik

- Yaitu arti dari percakapan yang berkaitan dengan pengaruhnya pada struktur data internal komputer dan/atau dunia eksternal.
- Pada bahasa manusia, ekuivalen dengan arti yang berasal dari para partisipan dalam percakapan

DIALOG MANUSIA-KOMPUTER

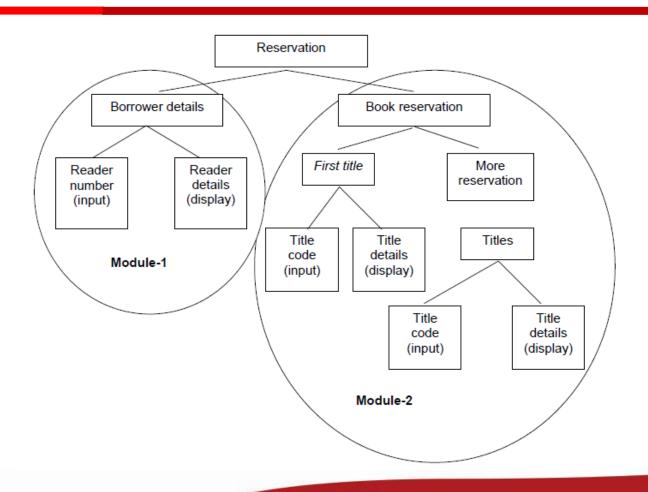
- Berbeda dengan dialog antar manusia pada umumnya, dialog dengan komputer biasanya terstruktur dan terbatas
- Beberapa ciri-ciri dari dialog terstruktur yang nantinya ditemukan dalam dialog komputer:
 - menyebutkan beberapa hal tertentu secara berurutan
 - beberapa bagian dari dialog dilakukan secara bersamaan (concurrently)
 - dialog berikutnya tergantung pada *respons dari partisipan*
 - dialog terstruktur biasanya tidak langsung menuju pada arti kata-katanya / semantik tapi pada level sintaksis

PROSES PERANCANGAN DIALOG

- a) Rangkaian Dialog menggambarkan struktur tugas, misalnya : 1 rangkaian dialog untuk mencapai 1 goal
- b) Beberapa rangkaian dialog tambahan digunakan untuk user support, misalnya help system, tutorial sub-sytem
- c) Rangkaian dialog diurutkan sesuai dengan *struktur tugas*

DESAIN DIALOG

- Prinsip yang digunakan dalam desain dialog adalah membagi sistem menjadi beberapa bagian yang disebut *module*
- Contoh : Sistem peminjaman buku di perpustakaan



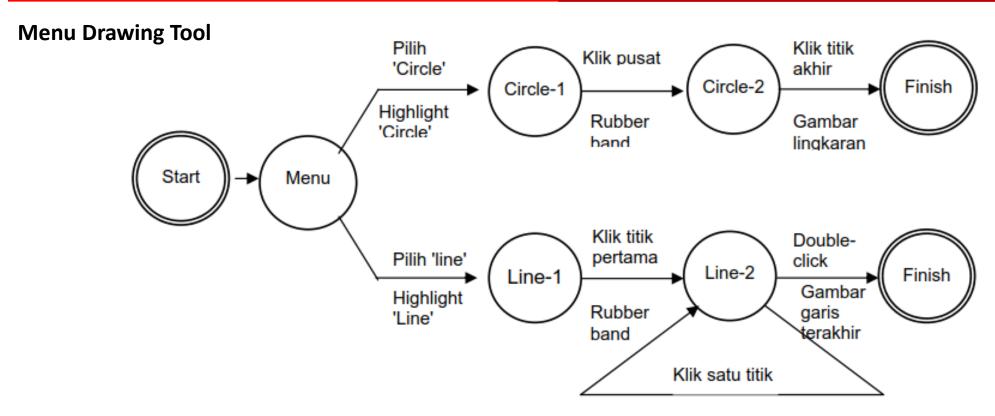
NOTASI DIAGRAMATIK

- Notasi diagramatik merupakan bentuk yang paling sering digunakan dalam desain dialog.
- Kelebihan dari bentuk ini adalah memungkinkan desainer untuk melihat secara sekilas struktur dialog namun kadangkala sulit untuk menjelaskan struktur dialog yang lebih luas dan kompleks.
- Ada beberapa bentuk <u>notasi diagramatik</u> yang dibahas pada bagian ini , yaitu :
 - a) State Transition Networks (STN)
 - b) Hierarchical State Transition Networks
 - c) Harel's State Charts
 - d) Flow Charts
 - e) JSD Diagrams

A) STATE TRANSITION NETWORKS (STN)

- STN telah lama digunakan untuk mendeskripsikan dialog dan berikut ini dicontohkan sebuah menu dari drawing tool sederhana berbasiskan mouse pada gambar dibawah.
- Menu tool ini terdiri dari dua pilihan yaitu 'Circle' dan 'Line', Jika menu 'Circle' yang dipilih maka user diperkenankan untuk memilih dua titik pada kertas gambar, yang pertama adalah pusat dari lingkaran dan yang lainnya adalah titik kedua sebagai penutup / jarak dari lingkaran.
- Jika menu 'Line' dipilih maka user akan menggambar polyline dan kemudian User dapat memilih beberapa titik pada bidang gambar.

STATE TRANSITION NETWORKS (STN)



Gambar garis dan rubber band dari titik yang baru

STATE TRANSITION NETWORKS (STN)

- Setiap lingkaran menandakan state / keadaan dari sistem, misalnya 'menu' adalah state sistem menunggu user untuk memilih 'Circle' atau 'Line' dan 'circle-2' adalah state setelah user memilih sebuah titik sebagai pusat lingkaran dan menunggu user untuk menentukan titik akhir lingkaran.
- Di antara state-state tersebut terdapat tanda panah yang disebut transisi dan tanda panah ini diberi label yang menjelaskan tentang tindakan user yang menyebabkan transisi perpindahan state dan respon dari sistem.
- Sebagai contoh, state 'circle-1' adalah state sistem menunggu user untuk memilih pusat lingkaran dan Jika user telah memilih / meng-klik pusat lingkaran maka state sistem akan bertransisi ke 'circle-2' dan direspon oleh sistem dengan menggambar rubber band.

STATE TRANSITION NETWORKS (STN)

Dari *gambar STN* diatas, kita dapat menyimpulkan bahwa STN dapat merepresentasikan beberapa hal yang terkait dengan dialog yaitu :

- Urutan (sequence) dari aksi yang dilakukan user dan respon yang diberikan oleh sistem.
- Pilihan bagi user (choice)

Contoh: dari state Menu, user dapat memilih "Circle" sehingga sistem berpindah ke state Circle-1 dan pilihan "Circle" pada menu di-highlight; alternatif lain, user dapat memilih "Line" sehingga sistem berpindah ke state Line-1.

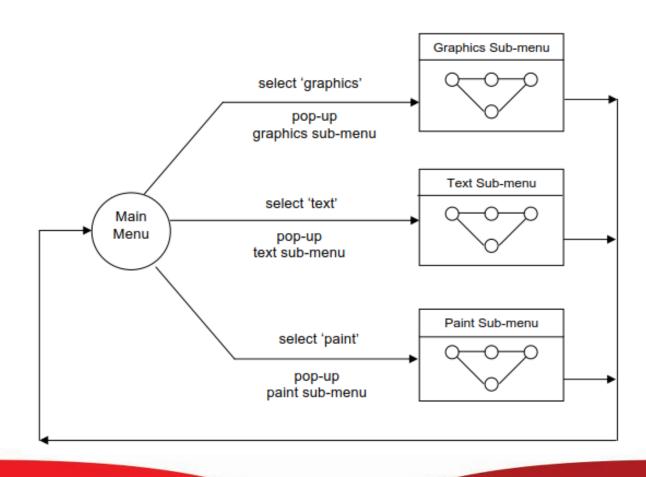
Iterasi (iteration)

Contoh: pada state Line-2, transisi dapat kembali ke state Line-2 jika user menambah titik baru pada polyline dan akan berpindah ke state Finish hingga user melakukan double-click.

HIERARCHICAL STATE TRANSITION NETWORK

- Misalkan drawing tool pada bahasan sebelumnya memiliki menu utama yang terdiri dari tiga submenu seperti submenu graphic, text dan paint.
- Untuk mendeskripsikan sistem yang lengkap dapat digunakan Hierarchical State
 Transition Networks seperti yang ditunjukkan pada gambar di bawah ini.

HIERARCHICAL STATE TRANSITION NETWORK



HIERARCHICAL STATE TRANSITION NETWORK

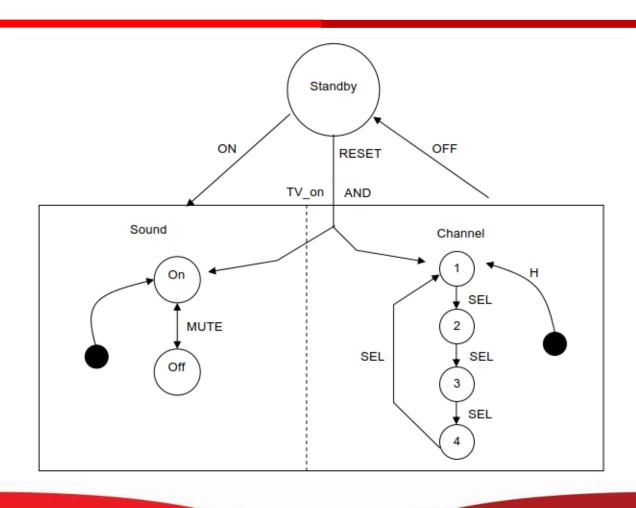
- Struktur Hierarchical STN mirip dengan STN namun memiliki tambahan berupa gabungan state (composite state) yang digambarkan dengan persegi panjang dengan gambar struktur STN berukuran kecil di dalamnya.
- Masing-masing persegi panjang ini menggambarkan submenu yang berkaitan dan submenu ini dapat dispesifikasikan dengan rinci pada STN tersendiri dengan menaruh label nama submenu yang bersangkutan pada simbol "start"-nya.
- Pengunaan hirarki ini tidak mengubah fungsi notasi dasar STN namun seperti menggabungkan beberapa STN ke dalam satu diagram besar sehingga model ini dapat digunakan untuk sistem-sistem yang besar.

- Harel's state chart dapat digolongkan sebagai kelompok STN dan jenis chart ini dibangun untuk menspesifikasikan secara visual sistem reaktif yang kompleks dan mampu mengakomodasi masalah seperti concurrency dan escape.
- Chart ini memiliki karakter struktur hirarki dalam satu diagram tunggal yang membagi elemen mana yang merepresentasikan state alternatif dan yang merepresentasikan aktifitas concurrent.

- Pada gambar dibawah ini dicontohkan state chart dari panel kendali televisi yang mana panel kendali ini memiliki lima tombol berlabel 'ON', 'OFF', 'MUTE', 'SEL', dan 'RESET'. Televisi yang dimaksud hanya dapat berada pada keadaan ON atau standby.
- Dimisalkan kita mulai pada keadaan standby kemudian jika Menekan tombol ON atau RESET akan menyebabkan televisi menyala (TV_on) dan pada saat TV menyala, tombol OFF akan menyebabkan TV kembali ke keadaan standby.
- Pada saat TV menyala, user dapat mengendalikan suara (sound) dengan tombol 'MUTE' yang mengatur suara menjadi ON atau OF; dan saluran TV (channel) dengan tombol 'SEL' yang dapat memilih diantara empat channel yang ada.

- Sub-dialog SOUND dan CHANNEL merupakan bagian dari state gabungan (composite state) TV_ON dan garis putus-putus diantara kedua sub-dialog dan keyword AND menyatakan bahwa kedua sub-dialog tersebut dapat dijalankan bersama-sama (concurrent) dan dalam urutan yang bebas.
- Sub-dialog SOUND memiliki lingkaran hitam kecil yang dihubungkan dengan garis lengkung ke state ON yang menunjukkan state awal dan nilai default dari state awal tersebut.
- Hal ini berarti bahwa setiap kali TV dinyalakan maka state SOUND pun akan ON dan sub-dialog CHANNEL sedikit berbeda, yaitu ada penandaan huruf 'H' kependekan dari 'History' pada state awal dan Ini mengindikasikan bahwa sub-dialog channel akan mengingat posisi channel TV terakhir yang diaktifkan user.

- Pada saat awal TV dihidupkan akan dimulai pada channel 1, namun setelah itu akan berada di channel terakhir yang diaktifkan user dan RESET akan menghapus default state awal dan Jika garis ON menuju hanya ke kotak TV_ON, RESET langsung menuju ke state tertentu pada dialog TV_ON.
- Hal ini mengakibatkan, setiap kali tombol RESET ditekan dari standby mode maka TV akan hidup dengan setting sound 'ON' dan channel '1' kemudian Tombol 'OFF' juga dapat bertindak sebagai escape pada dialog TV_ON ini.



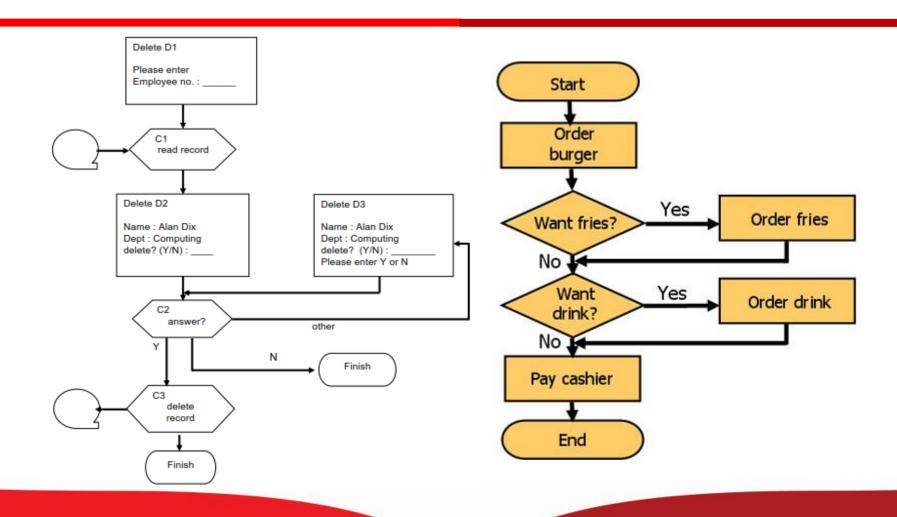
FLOW CHART

- Meskipun kelihatannya telah jauh tertinggal, Flow Chart dengan berbagai bentuknya mungkin merupakan notasi diagramatik yang paling banyak digunakan dan Flow Chart digunakan untuk mendeskripsikan dialog yang sederhana dan memiliki
- kelebihan dalam hal kesederhanaannya dan mudah dimengerti dan Kotak pada Flow Chart merepresentasikan proses atau keputusan sehingga tidak ekuivalen dengan state pada STN.
- Flow Chart menggunakan *berbagai jenis kotak untuk merepresentasikan berbagai jenis aktifitas yang berbeda*, namun lebih merefleksikan sudut pandang programer dibandingkan user.

FLOW CHART

- Pada gambar dibawah ini adalah flow chart subdialog delete dari sistem update database karyawan dan flow chart terdiri dari dua tipe kotak, yaitu persegi panjang adalah screen yang digunakan untuk berkomunikasi dengan user, dan segi enam adalah proses dan keputusan yang dibuat oleh sistem.
- Dengan tambahan elips "Finish" yang berarti kembali ke menu utama dan simbol tape yang berarti membaca atau mengubah database.
- Perbedaan utama antara menggunakan flow chart untuk perancangan dialog dengan pemrograman adalah masalah tingkat detail pada sisi program, misalkan, jika pembacaan record karyawan melibatkan pencarian secara sequential pada file untuk mencari record tertentu, maka flow chart program akan menyertakan loop pencarian tersebut, Sedangkan pada flow chart dialog, sama sekali tidak akan dicantumkan loop semacam itu.

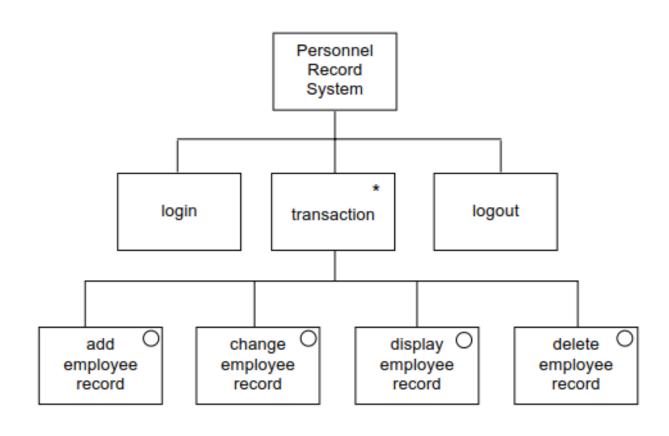
FLOW CHART



JSD DIAGRAM

- JSD (Jackson Structured Design) Diagram telah digunakan untuk berbagai aspek dari analisis tugas dan notasi dialog, seperti halnya flow chart, JSD memiliki kelebihan dalam hal model ini telah dikenal luas oleh para programer.
- Gambar di bawah ini dicontohkan sebuah JSD Diagram untuk struktur top level dari sistem kepegawaian dan sistem ini dipergunakan untuk meng-update record pegawai dalam berbagai cara, yaitu menambah baru, menampilkan, mengubah dan menghapus.

JSD DIAGRAM



JSD DIAGRAM

- Diagram ini mirip dengan diagram hirarki yang digunakan pada analisis tugas yang sudah kita bahas sebelumnya dan pada diagram JSD sistem kepegawaian ini dibagi menjadi tiga bagian yaitu login, transaction, dan logout.
- Urutan pengoperasiannya berjalan dari kiri ke kanan, yang berarti seorang user untuk dapat melakukan transaksi harus login terlebih dahulu dan terakhir logout.
- Tanda asterik (*) merepresentasikan iterasi atau pengulangan dan Tanda (o) merepresentasikan pilihan atau opsional dan dengan demikian, dialog ini berarti bahwa operasi dalam sistem kepegawaian terdiri dari sebuah login, diikuti sejumlah transaksi yang terdiri dari empat jenis pilihan yaitumenambah, mengubah, menampilkan atau menghapus record pegawai, dan akhirnya logout.

NOTASI DIALOG TEKSTUAL

- Tata Bahasa (Grammars)
- Production Rules
- CSP dan Event Algebras

TATA BAHASA (GRAMMARS)

- Karena diagram yang digunakan untuk menggambarkan dialog komputer juga memiliki sifat linguistik, maka tidak mengherankan jika formal grammar juga dipakai sebagai notasi dialog.
- Salah satu bentuk formal yang sering digunakan untuk notasi dialog tekstual adalah BNF (Backus-Naur Form) dan Regular Expression.
- BNF dan Regular Expression berfokus pada aksi yang dilakukan user bertolak belakang dengan STN yang berkonsentrasi pada state (kondisi).
- Regular Expression lebih banyak digunakan untuk mendeskripsikan kriteria pencarian tekstual yang kompleks dan analisis leksikal bahasa pemrograman.

TATA BAHASA (GRAMMARS)

- Regular expression serupa dengan BNF namun memiliki keterbatasan dan notasi yang digunakan pada regular expression berbeda dengan BNF, dan buruknya lagi terdapat beberapa jenis regular expression yang memiliki notasi yang berbeda.
- Sebagai contoh, proses menggambar garis seperti yang terdapat pada gambar STN dapat direpresentasikan dengan regular expression:

select-line click click* double-click

• ekspresi ini berarti bahwa user harus memilih *menu line, click posisi awal garis pada di manapun*, kemudian posisi garis berikut dapat terdiri dari beberapa titik dengan melakukan *click berkali-kali* dan *diakhiri dengan double-click*.

TATA BAHASA (GRAMMARS)

■ Tanda *asteriks (*)* pada <u>regular expression</u> berarti *iterasi atau pengulangan* dan pada <u>BNF</u> iterasi ini direpresentasikan dengan *sequencing (+)* menjadi bentuk :

```
some-thing ::= thing + some-thing
```

- Seperti halnya STN, BNF dan regular expression tidak dapat digunakan untuk merepresentasikan concurrency dialog (dialog yang dilakukan bersamaan).
- Salah satu kelebihan penggunaan BNF dan regular expression adalah kedua notasi mudah diimplementasikan karena telah tersedia tools yang memadai.

Bentuk umum dari production rule adalah :

If condition then action

Production rule ini juga memiliki beberapa bentuk representasi yang lain yaitu :

condition -> action

condition : action

event : condition -> action

- Semua rule umumnya bersifat aktif dan sistem secara konstan mencocokan isi dari bagian condition pada rule dengan event (kejadian) yang dimunculkan oleh user dan isi memori.
- Jika isi dari condition terpenuhi, maka bagian action dijalankan dan Isi dari action dapat berupa respon kepada user atau perubahan terhadap memori sistem dan urutan penulisan rule tidak begitu penting, rule manapun dapat diaktifkan pada saat kapanpun selama kondisi terpenuhi.

- Sistem production rule dapat berorientasi pada event, state, ataupun gabungan dari keduanya.
- Pertama akan dibahas production rule yang berorientasi pada event dan Sebagai contoh adalah aktifitas menggambar garis / polyline seperti pada gambar STN digambarkan sebagai production rule :

```
Sel-line -> start-line <highlight `line'>
C-point start-line -> rest-line <rubber band on>
C-point rest-line -> rest-line <draw line>
D-point rest-line -> <draw line> <rubber band off>
```

Pada contoh di atas, bagian *condition* dan *action* menggunakan bentuk sederhana dari nama event. Terdapat *tiga jenis event*, yaitu :

- User events, yang diawali dengan huruf kapital. Pada contoh diatas adalah Sel-line (user memilih 'line' pada menu), C-point dan D-point (merepresentasikan single click dan double click pada bidang gambar yang dilakukan user).
- Internal events, yang dimulai dengan huruf kecil. Bagian ini digunakan untuk mencatat sejarah state dialog, sebagai contoh rest-line adalah state setelah titik pertama garis dipilih oleh user.
- System response events, yang diapit oleh tanda kurung siku seperti <draw line> merupakan efek sistem yang dapat dilihat atau didengar.

- Production rule yang berorientasi state memiliki perilaku yang berbeda.
- Memori sistem terdiri dari sekumpulan variabel dengan nilai namun tidak dihapus pada saat rule dijalankan, sebaliknya mereka harus dihapus secara eksplisit oleh isi dari bagian action.
- Contoh dari state-oriented production rule yang lain adalah Olsen's propositional production system (PPS).
- Pada PPS, state dari sistem dibagi menjadi rangkaian hingga atribut, yang masing-masing memiliki nilai. Sebagian dari atribut merupakan hasil dari aksi yang dilakukan user, dan lainnya adalah efek pada display sistem.

Sebagai contoh, menu pada gambar STN dideskripsikan menjadi :

```
Mouse: {mouse-off, select-line, click-point, double-click}
Line-state: {menu, start-line, rest-line}
Rubber-band: { rubber-band-off, rubber-band-on}
Menu: {highlight-off, highlight-line, highlight-circle}
Draw: {draw-nothing, draw-line}
```

Atribut yang pertama, mouse diatur secara otomatis setiap kali user melakukan aksi yang relevan, atribut yang kedua digunakan untuk menyimpan sejarah state dan tiga atribut yang terakhir untuk mengendalikan respon sistem. Dengan menggunakan production rule PPS, maka contoh menggambar garis / polyline dideskripsikan menjadi:

```
select-line -> mouse-off start-line highlight-line
click-point start-line -> mouse-off rest-line rubber-band-on
click-point rest-line -> mouse-off draw-line
double-click rest-line -> mouse-off menu draw-line rubber-band-off
```

CSP DAN EVENT ALGEBRAS

- STN adalah contoh notasi dialog yang bagus digunakan untuk menunjukan urutan (sequence) namun tidak dapat merepresentasikan concurrency.
- Dan sebaliknya *production rule* dapat digunakan untuk *concurrency* namun tidak bagus untuk *sequence*.
- Masalah sequence dan concurrency adalah masalah yang umum dihadapi dalam area komputasi terutama dalam protokol komunikasi dan concurrent programming.
- Process algebras adalah sebuah kelas notasi formal yang dibangun untuk menangani situasi yang kompleks.
- Salah satu contohnya adalah CSP (Communicating Sequential Processes) yang dapat menspesifikasikan concurrency dan sequence sama baiknya dan mudah untuk dibaca

CSP DAN EVENT ALGEBRAS

Berikut ini adalah contoh notasi CSP untuk menu Draw pada gambar STN:

CSP DAN EVENT ALGEBRAS

Pada deskripsi di atas digunakan beberapa simbol operator, yaitu :

- Simbol (?) adalah event yang berupa aksi mouse yang dilakukan user. Event lain yang tidak diberi simbol merupakan event internal sistem.
- Simbol (=) digunakan untuk membangun deskripsi yang berarti "didefinisikan sebagai".
- Simbol (->) berarti urutan (sequence) event
- Simbol (;) menunjukkan urutan proses
- Simbol [] untuk menunjukkan pilihan (choice)
- Semua nama event pada dialog ditulis dalam huruf kecil, sedangkan nama proses diawali dengan huruf besar.