

# DIAGRAM STATE

Ama Fariza, S.Kom, M.Kom

The document name can go here Company Proprietary and Confidential

## MATERI

2

- Pendahuluan
- State
- Event
- Composite State
- Transisi

COMPANY

## PENDAHULUAN

3

- Diagram State atau diagram state chart menggambarkan siklus elemen yang terdapat dalam sistem.
- Pemodelan state adalah jenis khusus dari pemodelan perilaku yang berhubungan dengan pemodelan siklus dari elemen.

COMPANY

## STATE

4

- Elemen berkomunikasi dengan elemen lain dalam obyek, setiap elemen mempunyai siklus yaitu dibuat, mengetahui sesuatu, mengerjakan sesuatu, dapat berkomunikasi dengan elemen lain untuk meminta proses dari elemen lain, dapat memiliki elemen lain yang berkomunikasi dengannya untuk meminta prosesnya dan dihapus
- State adalah kondisi atau situasi khusus dari suatu elemen selama siklus hidup.
- State saat ini dari elemen disebut state aktif dan elemen dikatakan berada dalam state tsb.
- Terdapat beberapa jenis state: **simple, initial dan final**

COMPANY

## SIMPLE STATE

- SIMPLE STATE menyatakan kondisi atau situasi dari elemen
- Contoh: project management system mungkin berada pada satu simple state berikut:
  - **Inactive (tidak aktif)** : menyatakan project management system tidak tersedia untuk user nya, karena belum dimulai atau shutdown
  - **Active (aktif)** : menyatakan bahwa project management system sudah dimulai dan tersedia untuk user nya
  - **Suspended (dihentikan sementara)** : menyatakan project management sistem mengalami beberapa error, mungkin karena berjalan lambat pada penyimpan sekunder dan membutuhkan campur tangan user untuk aktif kembali

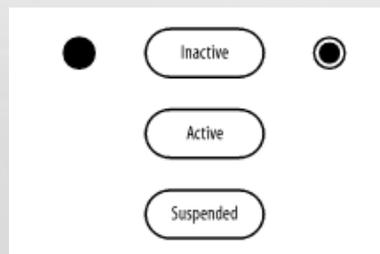


COMPANY

## INITIAL STATE DAN FINAL STATE

6

- Initial state menyatakan state dari elemen ketika dibuat.
- Final state menyatakan state dari elemen saat dihapus.
- Sebuah diagram state terdiri dari satu initial state dan beberapa final state



COMPANY

## EVENT # 1

7

- Adalah suatu kejadian/peristiwa, termasuk penerimaan dari permintaan
- Beberapa jenis kejadian:
  - CallEvent. Terkait dengan operasi dari sebuah kelas, kejadian disebabkan oleh panggilan untuk operasi yang diberikan. Efek yang diharapkan adalah bahwa langkah-langkah operasi akan dieksekusi.
  - SignalEvent. Berhubungan dengan sinyal, kejadian ini disebabkan oleh sinyal yang dibangkitkan.
  - TimeEvent. Suatu kejadian karena berakhirnya batas waktu waktu.
  - ChangeEvent. Sebuah kejadian yang disebabkan ekspresi tertentu (atribut dan asosiasi) menjadi benar.
- Event direpresentasikan dengan nama

COMPANY

## EVENT #2

8

- Contoh pada project management system merespon event berikut:
  - **Startup**: Menunjukkan bahwa project management system akan menjadi aktif jika tidak aktif.
  - **Shutdown**: Menunjukkan bahwa project management system akan menjadi tidak aktif jika aktif.
  - **Severe Error (Kesalahan berat)** : Menunjukkan bahwa project management system mengalami kesalahan yang parah, mungkin karena kehabisan penyimpanan sekunder, dan akan menjadi ditangguhkan jika aktif.
  - **Reset**: Menunjukkan bahwa project management system akan menjadi aktif jika ditangguhkan.

COMPANY

## EVENT #3

9

- Dalam UML, event dinyatakan menggunakan sintak:  
*event\_name (parameter\_list) (guard)*
- Dimana:
  - *event\_name*: adalah nama event. Event biasanya mempunyai nama yang sama dengan operasi dari elemen pada diagram state terkait; sehingga ketika elemen menerima event, operasi dipanggil
  - *parameter\_list*: sifatnya opsional, tanda koma memisahkan parameter yang melewati event. Setiap parameter kemungkinan adalah nilai eksplisit atau variabel. Parameter tidak digunakan jika event tidak membutuhkan parameter
  - *guard*: sifatnya opsional dan menyatakan kondisi yang harus dipenuhi untuk transisi atau kejadian.

COMPANY

## EVENT #4

10

- Selain UML, kita dapat menggunakan pseudocode atau bahasa pemrograman lain. Contoh, dapat menggunakan sintak Java, C++, C# atau bahasa pemrograman lain
- Contoh event yang menggunakan sintak lain:
  - **Startup** Jika event ini memerlukan identifikasi user, dapat menggunakan **Startup (UserID)** Jika project management system merespon jika dapat memulai, mungkin jika memori yang tersedia, Anda dapat memperbarui transisi sebagai berikut: **Startup (UserID) (memori yang cukup tersedia)**

COMPANY

## COMPOSITE STATE # 1

11

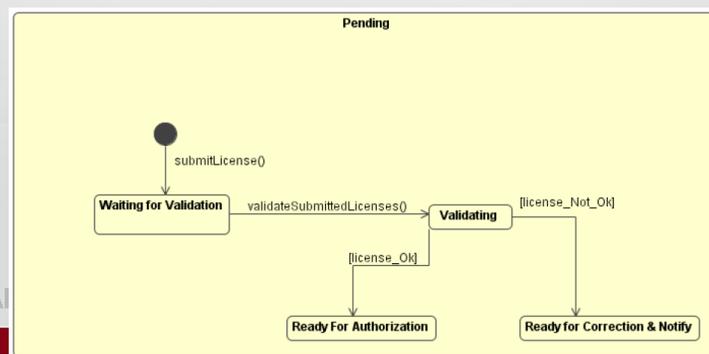
- Composite state adalah state sederhana yang terdiri dari satu atau lebih diagram statechart
- Composite state mungkin berisi:
  - 1) Sekumpulan state mutually exclusive; sebuah diagram statechart dalam sebuah state
  - 2) Sekumpulan state yang konkuren: state yang berbeda dibagi ke dalam region, aktif pada saat yang sama
- Sebuah composite state juga disebut super-state, state generalisasi yang berisi sekumpulan state khusus yang disebut substate

COMPANY

## COMPOSITE STATE #2

12

- Composite state (super-state) mungkin dikomposisi dalam dua atau lebih state level lebih rendah (sub-state)
- Semua aturan dan notasi adalah sama untuk substate dalam diagram statechart
- Dekomposisi dilakukan dalam beberapa level sesuai kebutuhan



COMPAN

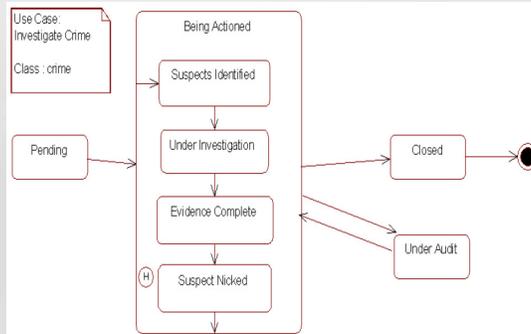
## COMPOSITE STATE #3

13

- **History State** : Aliran mungkin mengharuskan objek masuk ke trance, atau menunggu state, dan pada saat terjadipistiwa tertentu, kembali ke state saat berada di wait state (last active state)



Sebuah investigasi kriminalitas dimulai dari state "pending". Setelah beralih ke state "Being Actioned", dapat berada di sejumlah substates. Namun, pada interval waktu acak, kasus ini dapat diaudit. Selama audit, penyelidikan secara singkat ditangguhkan. Setelah audit selesai, penyelidikan harus melanjutkan di state dari mana itu sebelum audit. "notasi history" (lingkaran "H") yang digunakan :



## TRANSISI # 1

14

- Transisi merelasikan satu state dengan state lain
- Transisi antar state terjadi apabila
  - 1) Elemen adalah source state
  - 2) Terjadi event
  - 3) Aksi dilakukan
  - 4) Elemen memasuki target state
- Tanda panah menandakan obyek bertransisi dari satu state ke state lain
- Event yang menyebabkan transisi ditulis pada panah dan dipisahkan dengan tanda slash '/'
- Transisi yang terjadi karena state menyelesaikan aktifitas disebut transisi 'triggerless'
- Jika event terjadi setelah menyelesaikan beberapa event atau aksi, event atau aksi disebut kondisi 'guard'
- Kondisi 'guard' / event / action menggunakan kurung siku

COMPANY

Event / Action



## TRANSISI #2

15

- **Entry Actions**

- Lebih dari satu event dapat mentrigger sebuah transisi dari obyek ke state yang sama
- Jika aksi sama berlaku dalam sebuah state, aksi dapat ditulis sekali sebagai entry action
- Notasi
  - 1) Gunakan keyword entry diikuti tanda slash '/' dan aksi dibentuk setiap saat state masuk
  - 2) Entry action adalah bagian dari transisi internal

- **Exit Actions**

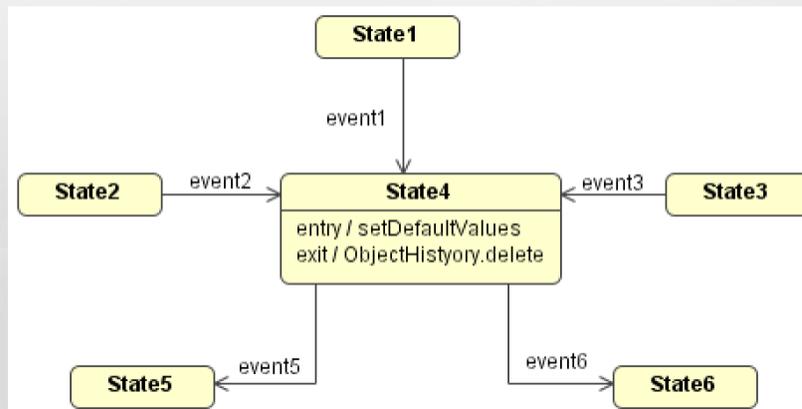
- Event dapat mentrigger transisi keluar dari state yang disebut exit action
- Notasi
  - 1) Gunakan keyword exit diikuti slash '/' dan aksi yang dibentuk setiap saat state keluar
  - 2) Exit action adalah bagian dari transisi internal hanya jika aksi berlaku saat state keluar

COMPANY

## TRANSISI #3

16

- Contoh:



COMPANY

## TRANSISI #4

17

- **Send Event**

- Terkadang obyek dimodelkan dengan statechart membutuhkan mengirim pesan ke obyek lainnya, dalam hal ini event yang keluar harus didefinisikan mana obyek yang menerima, dan ditangkap oleh obyek yang menerima
- Pesan dikirim ke obyek lain disebut send event
- Notasi: membutuhkan nama obyek sebelum ekspresi aksi

**Approved**

exit / Printer.sendPrinterOrder

COMPANY