

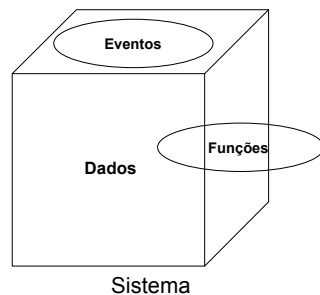
Modelagem Temporal com UML

Modelagem Temporal com UML

- Vários diagramas da UML podem ser usados para expressar os aspectos temporais
 - Diagrama de Atividades
 - Diagrama de Transição de Estados
 - Diagramas de Interação
 - Diagrama de Seqüência
 - Diagrama de Colaboração

Modelagem Temporal com UML

- Aspectos principais tratados pelos Diagramas: Funções e Eventos

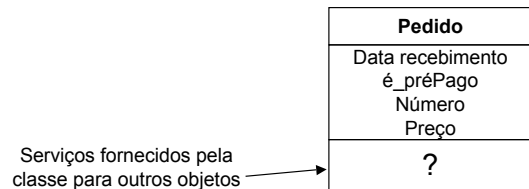


Modelagem Temporal com UML

- Objetivos da Modelagem Temporal
 - Descrever detalhadamente as funções a serem desempenhadas em um sistema OO
 - Fluxo de mensagens entre objetos
 - Métodos necessários para cada classe
 - Métodos de classe
 - Métodos de objetos
 - "O que o sistema deve realizar?"
 - "Quando cada função será realizada?"

Modelagem Temporal com UML

- Modelagem Funcional e Temporal
 - Definição de Métodos



Modelagem Temporal com UML

- Modelagem Funcional e Temporal
 - Tipos de métodos
 - Serviços algoritmicamente simples (90%)
 - Criar
 - Conectar
 - Acessar
 - Liberar
 - Serviços algoritmicamente complexos
 - Calcular
 - Calcula um resultado a partir dos valores de atributos do objeto
 - Monitorar
 - Monitora um sistema ou dispositivo externo
 - Lida com E/S externas ao sistema

Modelagem Temporal com UML

Diagrama de Transição de Estados

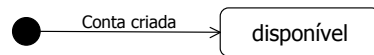
DTE

- Histórico
 - Notação proposta por David Harel-1987
 - Máquina com número finito de estados
 - A máquina pode receber eventos, e cada evento pode gerar uma transição de estado

DTE

Estado

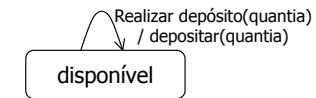
- Situação na vida de um objeto
- Estado é determinado pelo conjunto de valores armazenados em um objeto
- Estados Especiais
 - Inicial: somente um em um DTE
 - Final



DTE

Transições

- Um estado é associado a outro (ou a si mesmo) pelas transições
- Formato:
 - evento(lista-parâmetros) [guarda] / ação



DTE

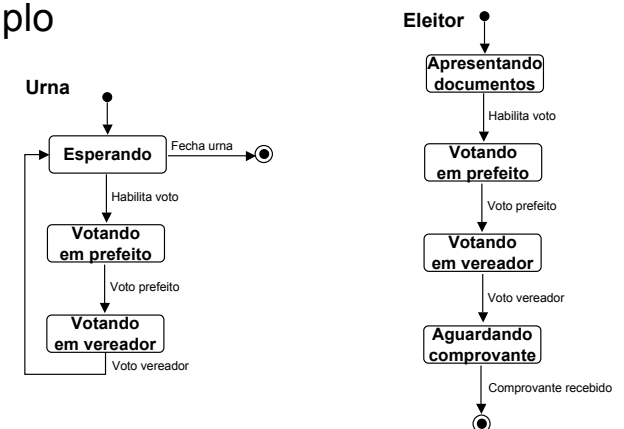
Eventos

- Uma transição é associada a um evento
- Evento: “algo que acontece em algum ponto no tempo e que pode modificar o estado de um objeto” [Bezerra, 2002]

Eventos em Software	Eventos do Domínio do Problema
Mouse pressionado	Pedido realizado
Impressora sem papel	Fatura paga
	Cheque devolvido

DTE

Exemplo



DTE

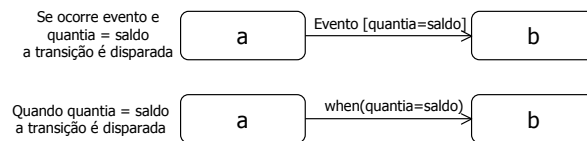
■ Eventos: tipos

- Evento de chamada: recebimento de uma mensagem de outro objeto (síncrono)
- Evento de sinal: recebimento de um sinal (assíncrono)
- Evento temporal: passagem de intervalo de tempo pré-definido (cláusula *after*)
 - Exemplo: `after(30 segundos)`
- Evento de alteração/tempo: condição que se torna verdadeira (cláusula *when*)
 - Exemplos:
 - `when(saldo > 0)`
 - `when(horário = 00:00 h)`

DTE

■ Guardas

- Condição lógica que precisa ser satisfeita para habilitar a transição
- Diferença entre guardas e evento de mudança



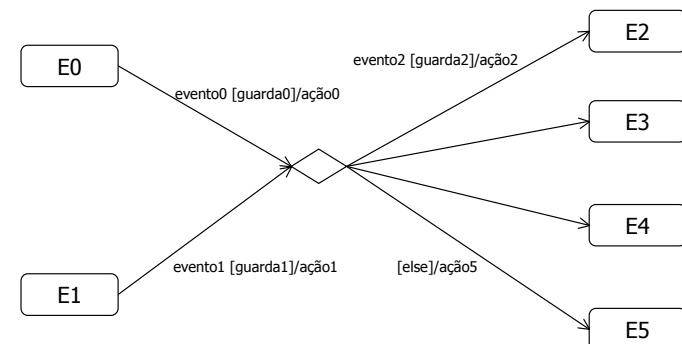
DTE

■ Ações

- "Ao transitar de um estado para outro, um objeto pode realizar uma ou mais ações"
- Representação:
 - `/ação`

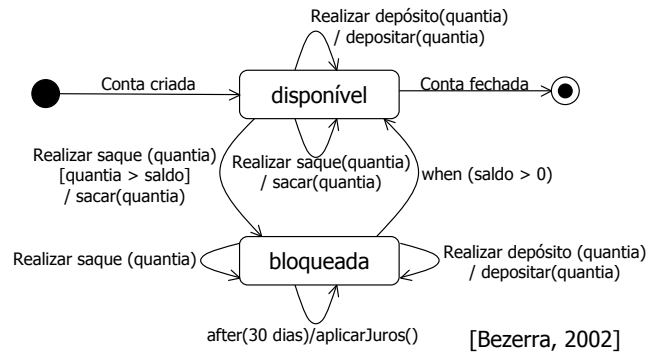
DTE

■ Exemplo: Ponto de Junção



DTE

Exemplo: Conta Bancária



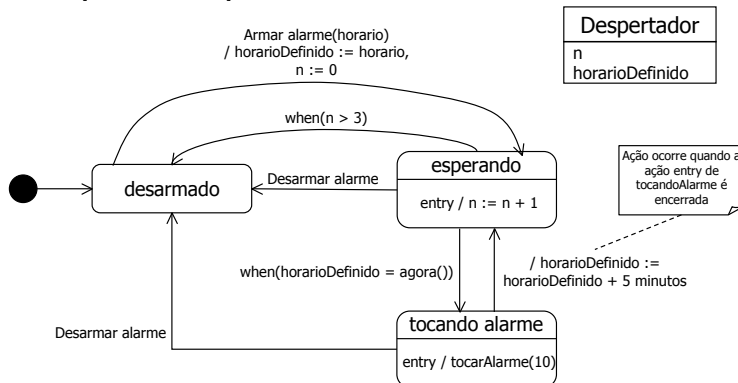
DTE

Exemplo: despertador



DTE

Exemplo: Despertador



DTE

- Entry:
 - Ação executada quando da entrada naquele estado;
- Exit:
 - Ação executada quando da saída daquele estado;
- Entry e Exit permitem a simplificação de DTEs, visto que **não** é necessário repetir a mesma ação para todas as transições de entrada ou saída de um estado;
- Quando Entry ou Exit **não** são indicados, a idéia é que a ação seja executada continuamente dentro do estado;

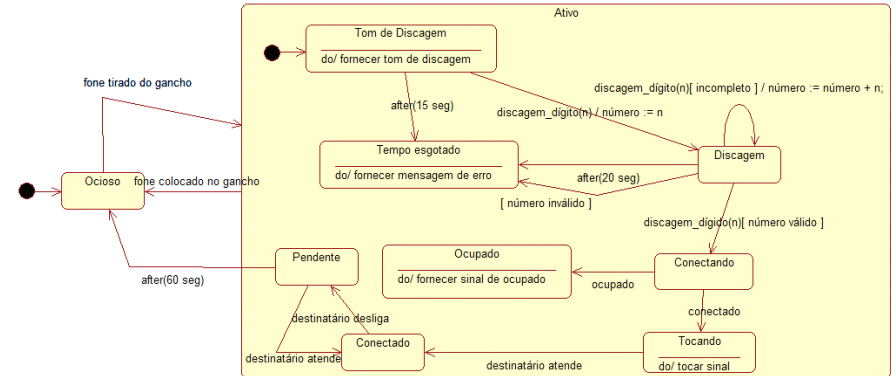
DTE

■ Estados e Sub-estados

- Um estado pode ser decomposto em sub-estados
- Cada estado composto deve possuir os estados inicial e final

DTE

■ Estados e Sub-estados

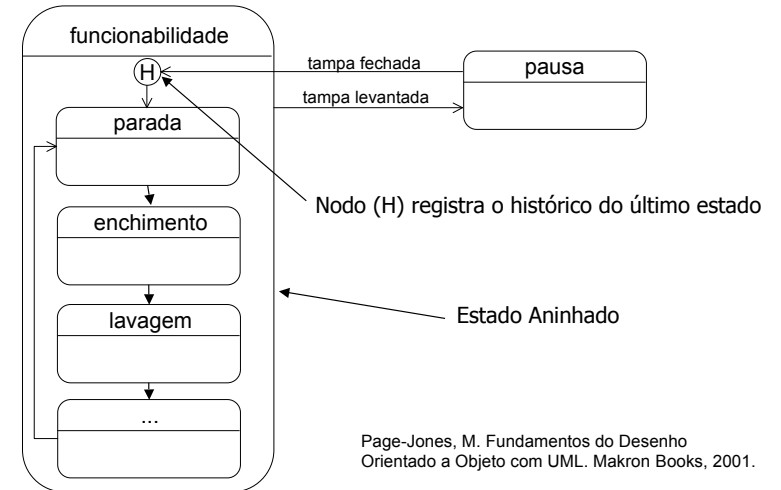


DTE

■ Histórico

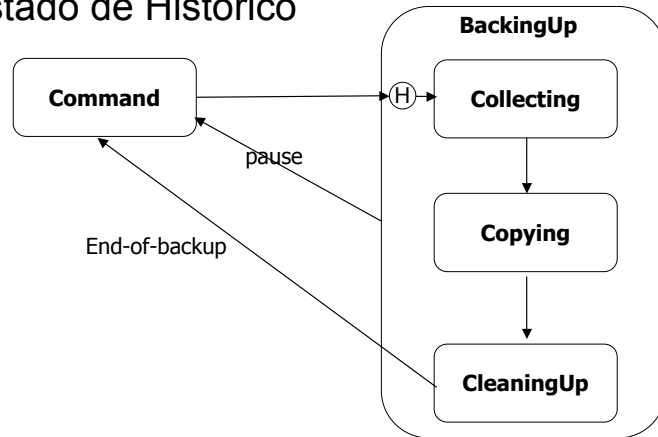
- Mecanismo de ativação de subestados
- Se a entrada em um estado for por história simples (H), o subestado mais recentemente visitado é escolhido, em detrimento do estado *default*.
- Se for história do tipo H* então considera-se como candidato a ser ativado o subestado mais recentemente visitado independente do nível hierárquico em que se encontra

DTE



DTE

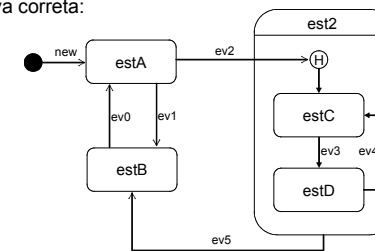
Estado de Histórico



DTE

Exercício

Considerando o diagrama de estados abaixo e a seqüência de eventos listada abaixo, marque a alternativa correta:



Eventos: new, ev1, ev0, ev2, ev3, ev5, ev0, ev2

- a) Após a ocorrência de eventos listada o objeto encontra-se no estado estA
- b) Após a ocorrência de eventos listada o objeto encontra-se no estado estC
- c) Após a ocorrência de eventos listada o objeto encontra-se no estado estB
- d) Após a ocorrência de eventos listada o objeto encontra-se no estado estD
- e) Após a ocorrência de eventos listada o objeto encontra-se no estado inicial

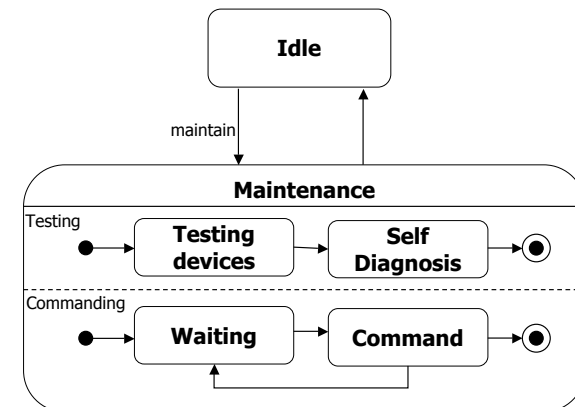
DTE

Exercício

Construa um DTE para a classe professor universitário. Considere a carreira de um professor universitário de uma instituição federal de ensino (como a UFPA). Existem quatro categorias possíveis: auxiliar, assistente, adjunto e titular, onde cada categoria corresponde a qualificação do professor. Auxiliares possuem graduação e/ou especialização, Assistentes possuem mestrado e Adjuntos possuem doutorado. Professores titulares possuem doutorado e devem trabalhar na mesma instituição a mais de 8 anos mas também tem de comprovar excelência acadêmica. Esta classificação é inicialmente válida durante o processo de contratação dos professores e /ou quando estes professores continuam se qualificando. Um professor também pode mudar sua classificação de acordo com o tempo de serviço na instituição: após 8 anos como professor auxiliar, ele se torna professor assistente e da mesma forma após 8 anos como professor assistente um professor se torna professor adjunto.

DTE

Estados concorrentes

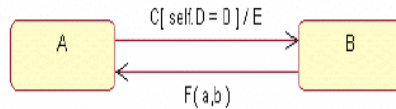


Modelagem Temporal com UML

Processo e Heurísticas para Construção de diagramas de transições de estados

```
class N
{
private
final static int state_A = 0; // constant state attribute
final static int state_B = 1; // constant state attribute
int state; // store the current state of an object
int D;
public
void C () // signal event
{
switch (state)
{
case state_A :
if (D == 0) { // check guard condition
state = state_B; // state transitions
E(); // perform action
Break; }
}
}

int F (int a, int b) // call event
{
switch (state)
{
case state_B :
state = state_A;
break;
}
}
}
```



DTE

■ Quantidade de Estados

- Em sistemas bastante simples, a definição dos estados de todos os objetos não é tão trabalhosa.
- No entanto, o crescimento dos estados de um sistema cresce exponencialmente com o número de objetos
- Torna impraticável a construção de um único diagrama para todo o sistema

DTE

■ Um DTE por classe

- Para resolver o problema da explosão exponencial de estados, os diagramas de estados são desenhados por classe.
 - Desvantagem: dificuldade na visualização do estado do sistema como um todo.
 - Essa desvantagem é parcialmente compensada pelos diagramas de interação.



DTE

■ Identificação de Transições

- Nem todas as classes de um sistema precisam de um DTE.
 - Somente classes que exibem um comportamento dinâmico relevante.
 - Objetos cujo histórico precisa ser rastreado pelo sistema são típicos para se construir um diagrama de estados.



DTE

■ Processo de construção de DTEs (1)

- Identifique os estados relevantes para a classe.
- Identifique os eventos relevantes. Para cada evento, identifique qual a transição que ele ocasiona.
- Para cada estado: identifique as transições possíveis quando um evento ocorre.
- Para cada estado, identifique os eventos internos e ações correspondentes.



DTE

■ Processo de construção de DTEs (2)

- Para cada transição, verifique se há fatores que influenciam no seu disparo. (definição de condições de guarda e ações).
- Para cada condição de guarda e para cada ação, identifique os atributos e ligações que estão envolvidos.
- Defina o estado inicial e os eventuais estados finais.
- Desenhe o DTE.



DTE

■ Informações para o modelo de classes

- A construção de um DTE freqüentemente leva à descoberta de novos atributos para uma classe
 - principalmente atributos para servirem de abstrações para estados.
- Além disso, este processo de construção permite identificar novas operações na classe
 - pois os objetos precisam reagir aos eventos que eles recebem.

DTE

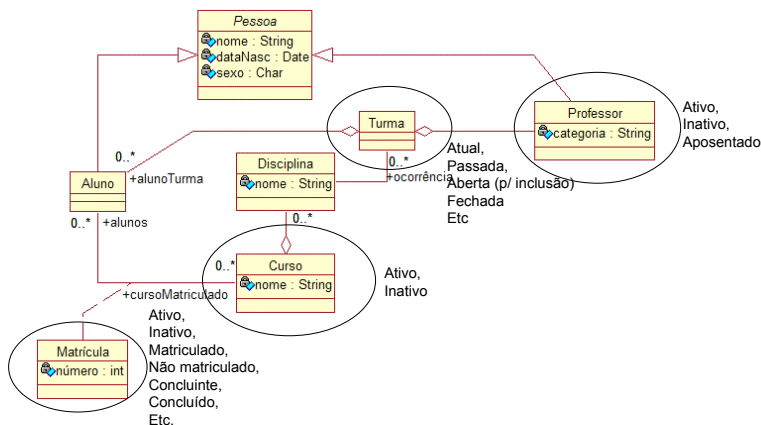
- Uso do DTE no processo de desenvolvimento
 - Os diagramas de estados podem ser construídos com base nos diagramas de interação e nos diagramas de classes.
 - Durante a construção do diagrama de estados para uma classe, novos atributos e novas operações podem surgir. Essas novas propriedades devem ser adicionadas ao modelo de classes.

DTE

- Uso do DTE no processo de desenvolvimento (continuação)
 - O comportamento de um objeto varia em função do estado no qual ele se encontra.
 - Pode ser necessária a atualização de uma ou mais operações de uma classe para refletir o comportamento do objetos em cada estado.
 - Por exemplo, o comportamento da operação **sacar()** da classe **ContaBancária** varia em função do estado no qual esta classe se encontra
 - saques não podem ser realizados em uma conta que esteja no estado **bloqueada**.

DTE

- Quais são as classes candidatas a possuírem DTE?



DTE

- Observação importante
 - Muitas vezes as regras do negócio precisam ser flexibilizadas no sistema implementado
 - Tipicamente necessário em decorrência de erros no preenchimento de dados ou ainda em função de decisão administrativa de alto nível
 - Prever as situações e criar transições com estereótipo `<<exception>>`

