

# Análise e Projeto de Sistemas

Universidade Federal do Ceará – UFC

Campus de Quixadá

Curso de Sistemas de Informação

Prof. Marcos Antonio de Oliveira (deoliveira.ma@gmail.com)

“Todos os adultos um dia foram crianças, mas poucos se lembram disso.”

(O Pequeno Príncipe, Antoine de Saint-Exupéry)

# **MODELAGEM DE ESTADOS**

Esses slides são uma adaptação das notas de aula do professor Eduardo Bezerra autor do livro Princípios de Análise e Projeto de Sistemas com UML

# Índice

- Introdução
- Diagramas de transição de estados
- Identificação dos elementos de um diagrama de estados
- Construção de diagramas de transição de estados
- Modelagem de estados no processo de desenvolvimento

# INTRODUÇÃO

# Introdução

- Objetos do mundo real se encontram em estados particulares a cada momento
  - Uma jarra está cheia de líquido
  - Uma pessoa está cansada
- Da mesma forma, cada objeto participante de um sistema de software orientado a objetos se encontra em um *estado* particular
- Um objeto muda de estado quando acontece algum *evento* interno ou externo ao sistema
- Durante a transição de um estado para outro, um objeto realiza determinadas *ações* dentro do sistema
- Quando um objeto transita de um estado para outro, significa que o sistema no qual ele está inserido também está mudando de estado

# **DIAGRAMA DE TRANSIÇÃO DE ESTADOS**

# Diagrama de Transição de Estado

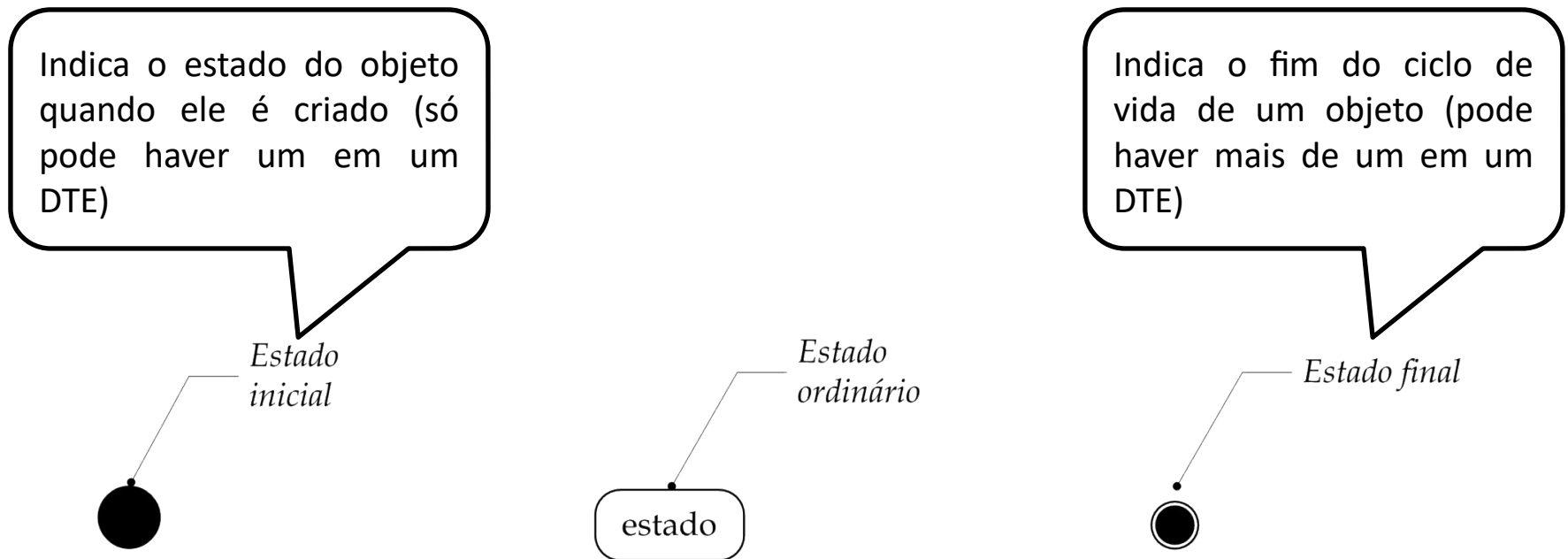
- Através da análise das **transições** entre estados dos objetos de um sistema de software, podem-se prever todas as possíveis **operações** realizadas, em função de **eventos** que possam ocorrer
- O diagrama da UML que é utilizado para realizar esta análise é o ***diagrama de transição de estado*** (DTE)
- A UML tem um conjunto rico de notações para desenhar um DTE
  - Estados, transições, evento, ação, atividade, transições internas, estados aninhados e estados concorrentes

# Estado

- Situação na vida de um objeto em que ele satisfaz a alguma condição ou realiza alguma atividade. É função dos **valores dos atributos** e (ou) das **ligações com outros objetos**
  - Uma conta bancária passa para o *vermelho* quando o seu saldo fica *negativo*
  - Um professor está *licenciado* quando não está ministrando curso algum durante o semestre
  - Um tanque está *na reserva* quando nível de óleo está abaixo de 20%



# Estados Inicial e Final



# Transições

- Os estados estão associados a outros pelas transições
- Quando uma transição entre estados ocorre, diz-se que a transição foi disparada

evento (lista-parâmetros) [guarda] / ação

# Eventos

- Uma transição possui um evento associado
- Um evento é algo que acontece em algum ponto no tempo e que pode modificar o estado de um objeto
  - Pedido realizado
  - Fatura paga
  - Cheque devolvido

# Eventos

- Os eventos relevantes a um sistema de software podem ser classificados nos seguintes tipos
  - **Evento de chamada:** recebimento de uma mensagem de outro objeto
  - **Evento de sinal:** recebimento de um sinal
  - **Evento temporal:** passagem de um intervalo de tempo predefinido
  - **Evento de mudança:** uma condição que se torna verdadeira

# Tipos de Eventos

- Evento de Chamada
  - Corresponde ao recebimento de uma mensagem de outro objeto
  - Pode-se pensar neste tipo de evento como uma solicitação de serviço de um objeto a outro
- Evento de Sinal
  - Neste evento o objeto recebe um sinal de outro objeto que pode fazê-lo mudar de estado
  - A diferença básica entre o evento de sinal e o evento de chamada é que neste último o objeto que envia a mensagem fica esperando a execução da mesma

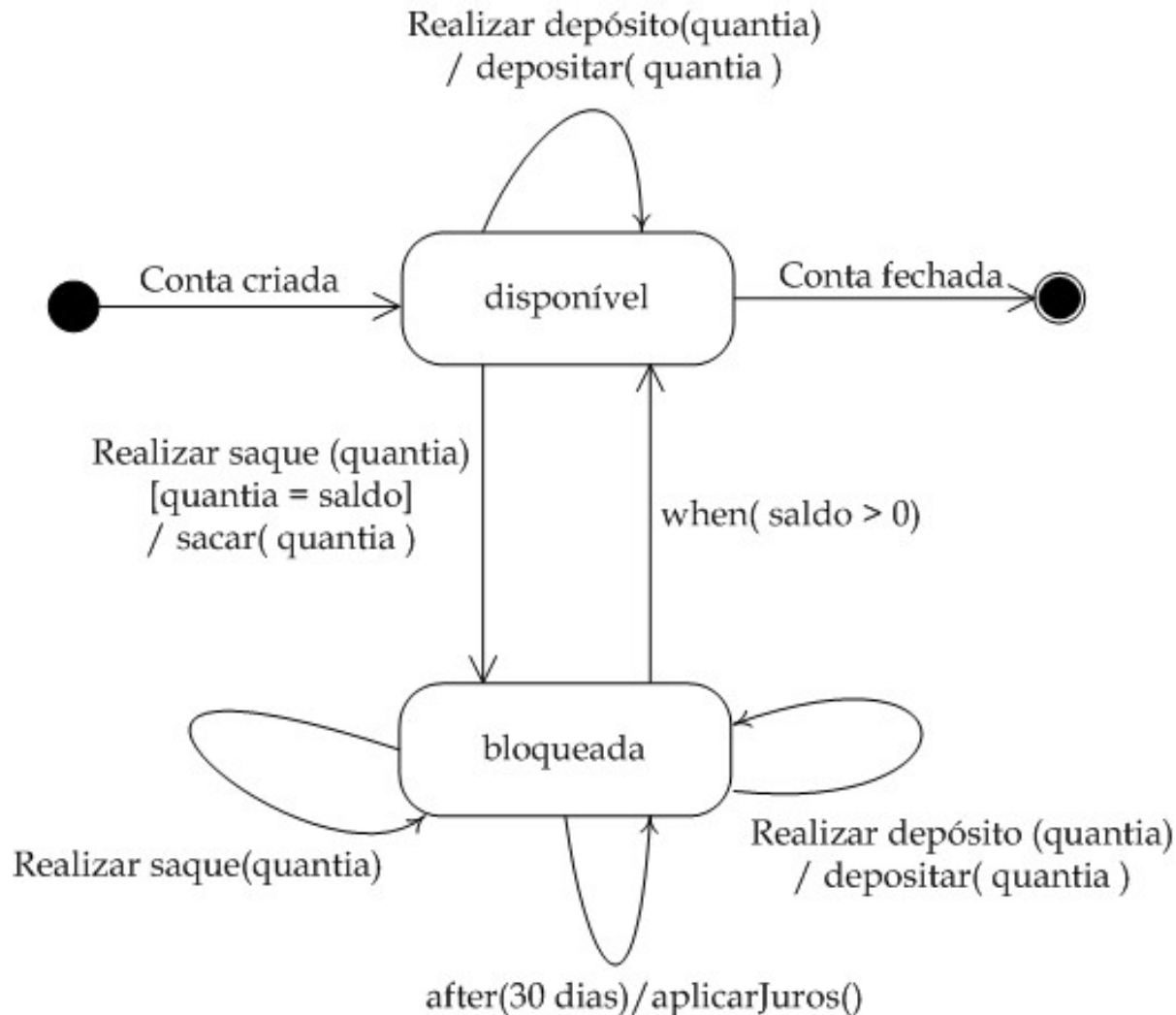
# Tipos de Eventos

- Evento de Temporal
  - Corresponde à passagem de um intervalo de tempo predefinido
  - O objeto pode interpretar a passagem de um certo intervalo de tempo como sendo um evento
  - É especificado com a cláusula **after** seguida de um parâmetro que especifica um intervalo de tempo
    - **after**(30 segundos)

# Tipos de Eventos

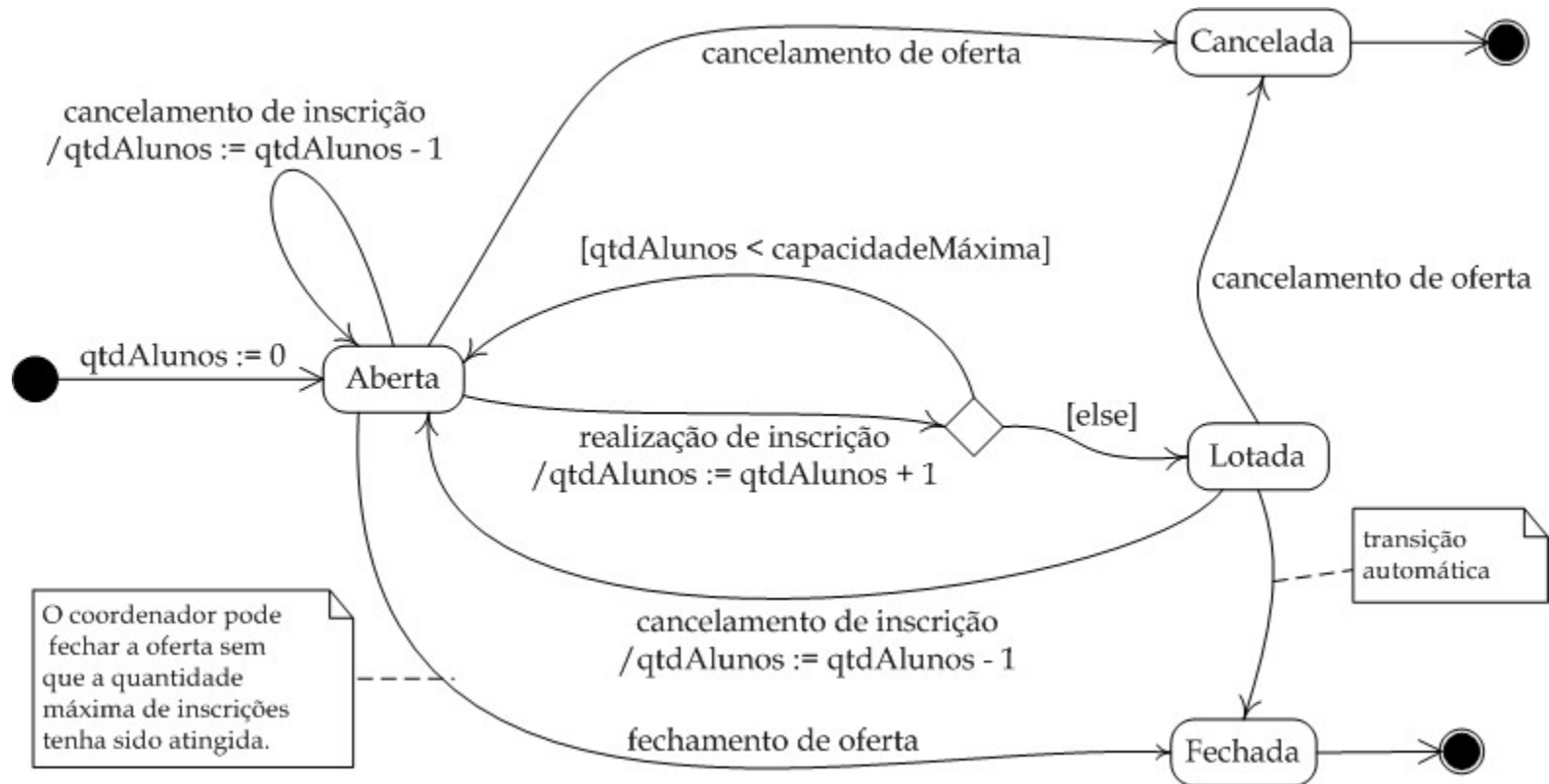
- Evento de Mudança
  - Corresponde a uma condição que se torna verdadeira
  - É representado por uma expressão de valor lógico (verdadeiro ou falso) e é especificado utilizando-se a cláusula **when**
    - **when**(saldo > 0): significa que a transição é disparada quando o valor do atributo saldo for positivo
  - Eventos temporais também podem ser definidos utilizando-se a cláusula **when**
    - **when**(data = 13/07/2002) e **when**(horário = 00:00h)

# Exemplo (ContaBancária)





# Exemplo (OfertaDisciplina)



# Condição de Guarda

- É uma expressão de valor lógico que condiciona o disparo de uma transição
  - A transição correspondente é disparada se e somente se o evento associado ocorre *e* a condição de guarda é verdadeira
- A condição de guarda pode ser definida utilizando-se parâmetros passados no evento e também atributos e referências a ligações da classe em questão

# Ações

- Ao transitar de um estado para outro, um objeto pode realizar uma ou mais **ações**
- Uma ação é uma expressão definida em termo dos atributos, operações, associações da classe ou dos parâmetros do evento também podem ser utilizados
- A ação associada a uma transição é executada se e somente se a transição for disparada

# Atividades

- Semelhantes a ações, atividades são algo que deve ser executado
- No entanto, uma atividade pode ser *interrompida*
  - Por exemplo, enquanto a atividade estiver em execução, pode acontecer um evento que a interrompa
- Outra diferença: uma atividade sempre está associada a um estado (ao contrário, uma ação está associada a uma transição)

# Cláusulas

- No compartimento adicional de um retângulo de estado podem-se especificar ações ou atividades a serem executadas
  - Sintaxe geral: **evento / [ação | atividade]**
  - Há três cláusulas predefinidas
    - Entry
    - Exit
    - Do

# Cláusulas

- Cláusula Entry
  - É usada para especificar uma ação a ser realizada no momento em que o objeto entra em um estado
  - A ação desta cláusula é sempre executada, independentemente do estado do qual o objeto veio
    - É como se a ação especificada estivesse associada a todas as transições de entrada no estado.

# Cláusulas

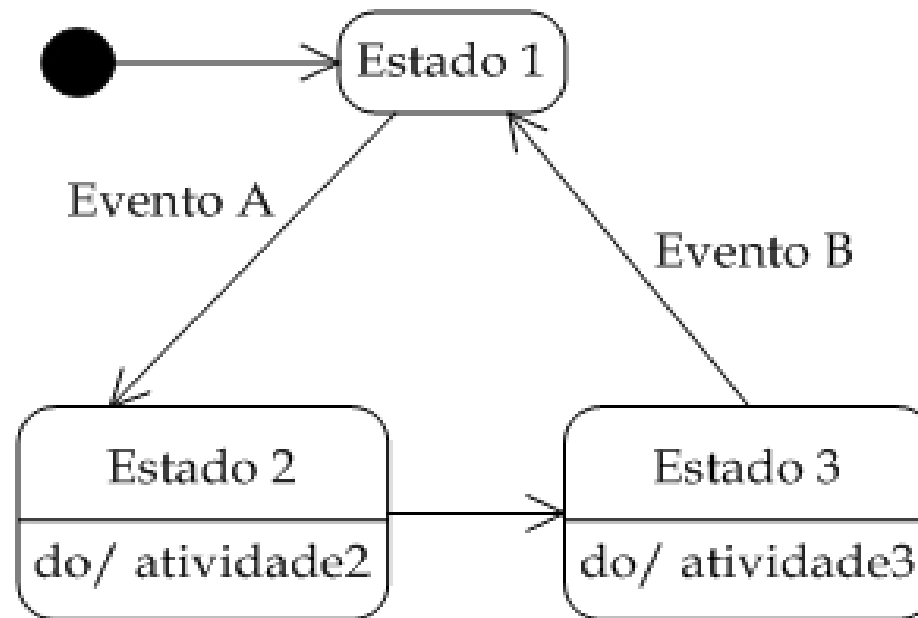
- Cláusula Exit
  - Serve para declarar ações que são executadas sempre que o objeto sai de um estado
  - É sempre executada, independentemente do estado para o qual o objeto vai
    - É como se a ação especificada estivesse associada a todas as transições de saída do estado

# Cláusulas

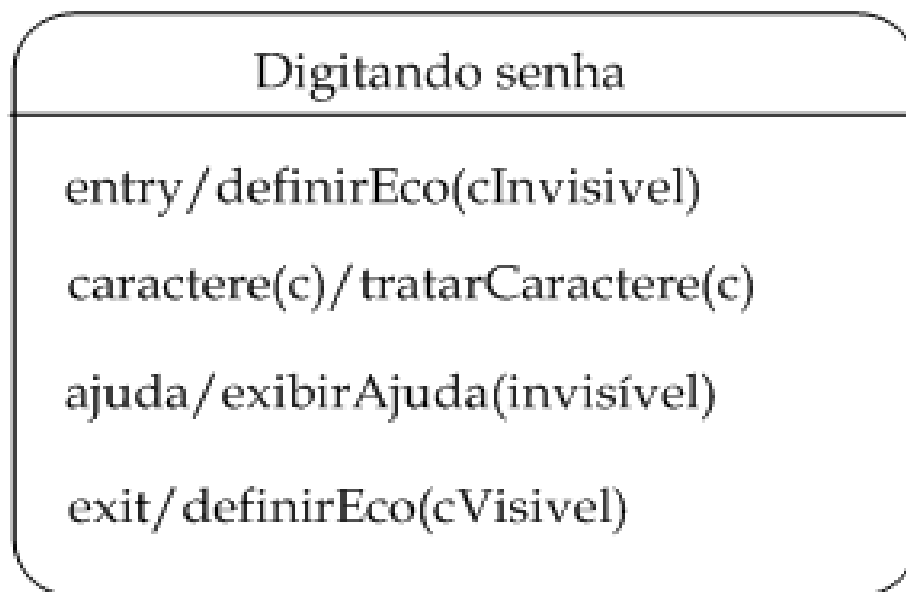
- Cláusula Do
  - Usada para definir alguma atividade a ser executada quando o objeto passa para um determinado estado
  - Ao contrário da cláusula entry, serve para especificar uma atividade, em vez de uma ação



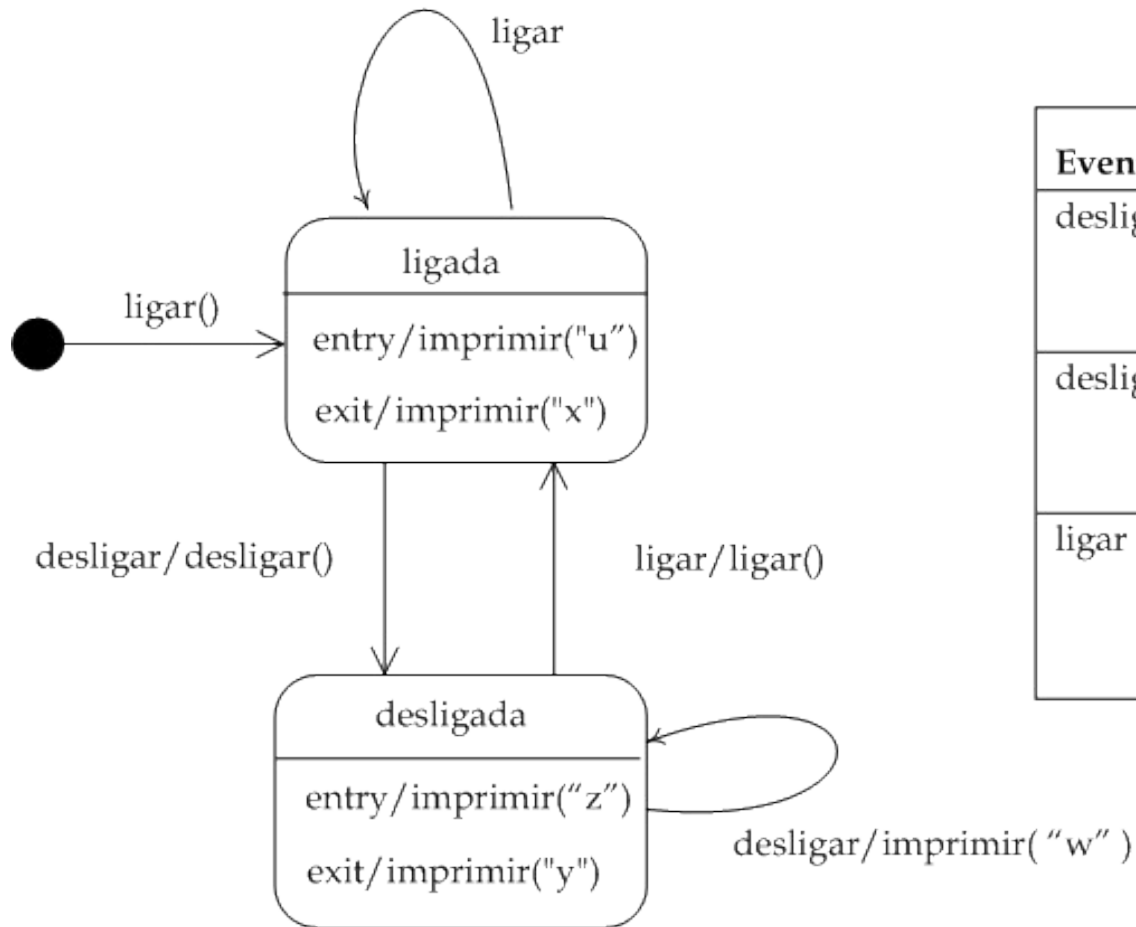
# Exemplo – Cláusula Do



# Exemplo – Cláusula Entry e Exit

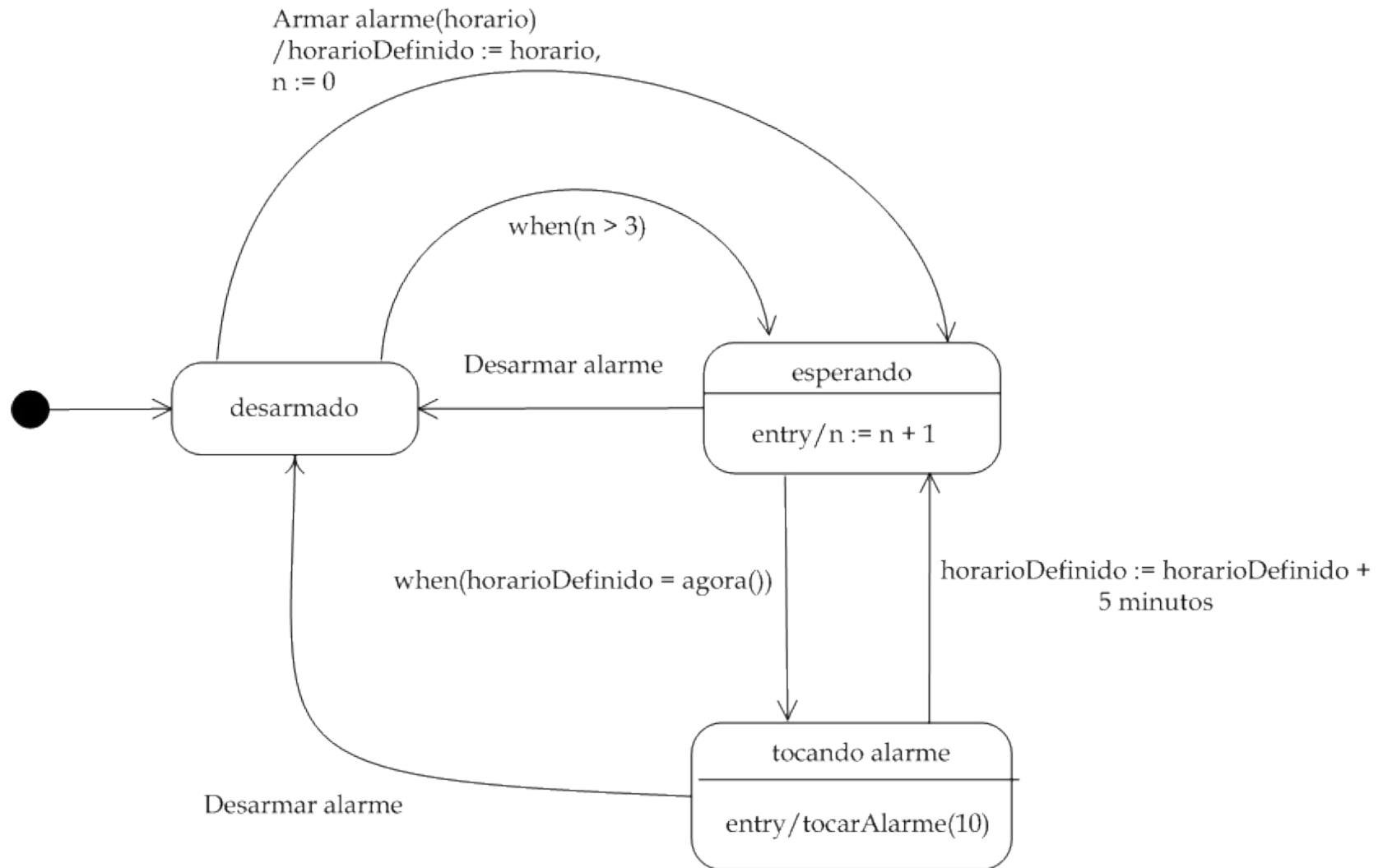


# Exemplo



Evento	Ações executadas
desligar	imprimir("x") desligar() imprimir("z")
desligar	imprimir("y") imprimir("w") imprimir("z")
ligar	imprimir("y") ligar() imprimir("u")

# Exemplo - Despertador



# Referências

- BEZERRA, E. Princípios de Análise e Projeto de Sistemas com UML. 2ª ed. Rio de Janeiro: Elsevier, 2007.
- FOWLER, M. 3. UML Essencial. 3. ed. Porto Alegre: Bookman, 2007.