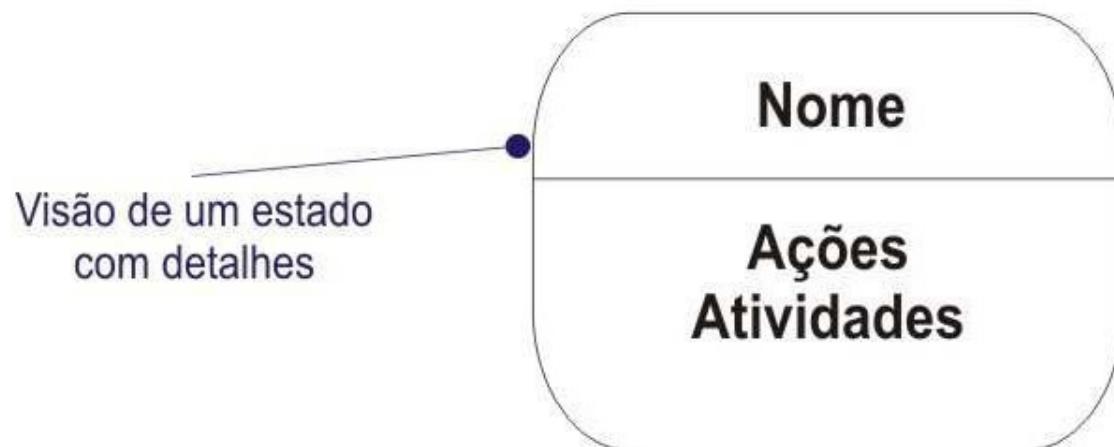
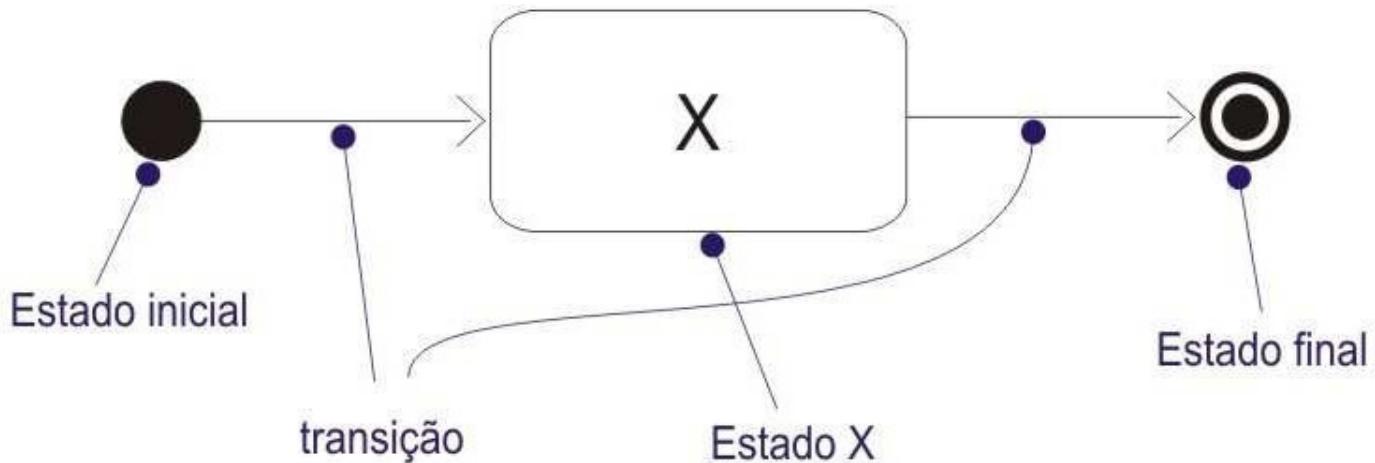


# Diagramas de Estado, Atividades, Componentes e Instalação

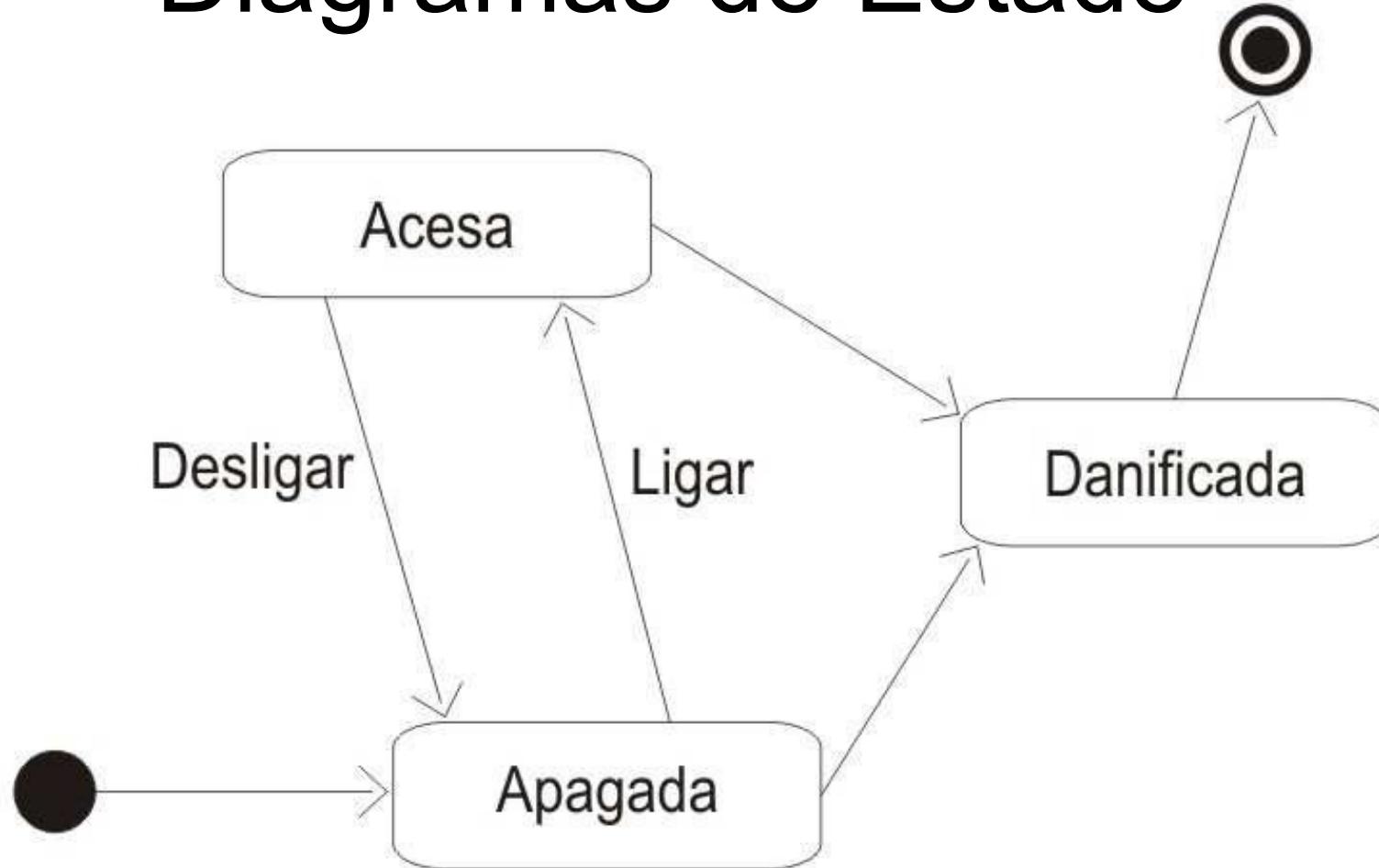
# Diagrama de Estado

- Um diagrama de estados (statechart), também conhecido por diagrama de transição de estado ou por máquina de estados, permite modelar o comportamento interno de um determinado objeto, subsistema ou sistema global.
- Estes diagramas representam os possíveis estados de um objeto, as correspondentes transições entre estados, os eventos que fazem desencadear as transições, e as operações (ações e atividades) que são executadas dentro de um estado ou durante uma transição. Os objetos evoluem ao longo do tempo através de um conjunto de estados como resposta a eventos e à passagem de tempo.

# Diagramas de Estado

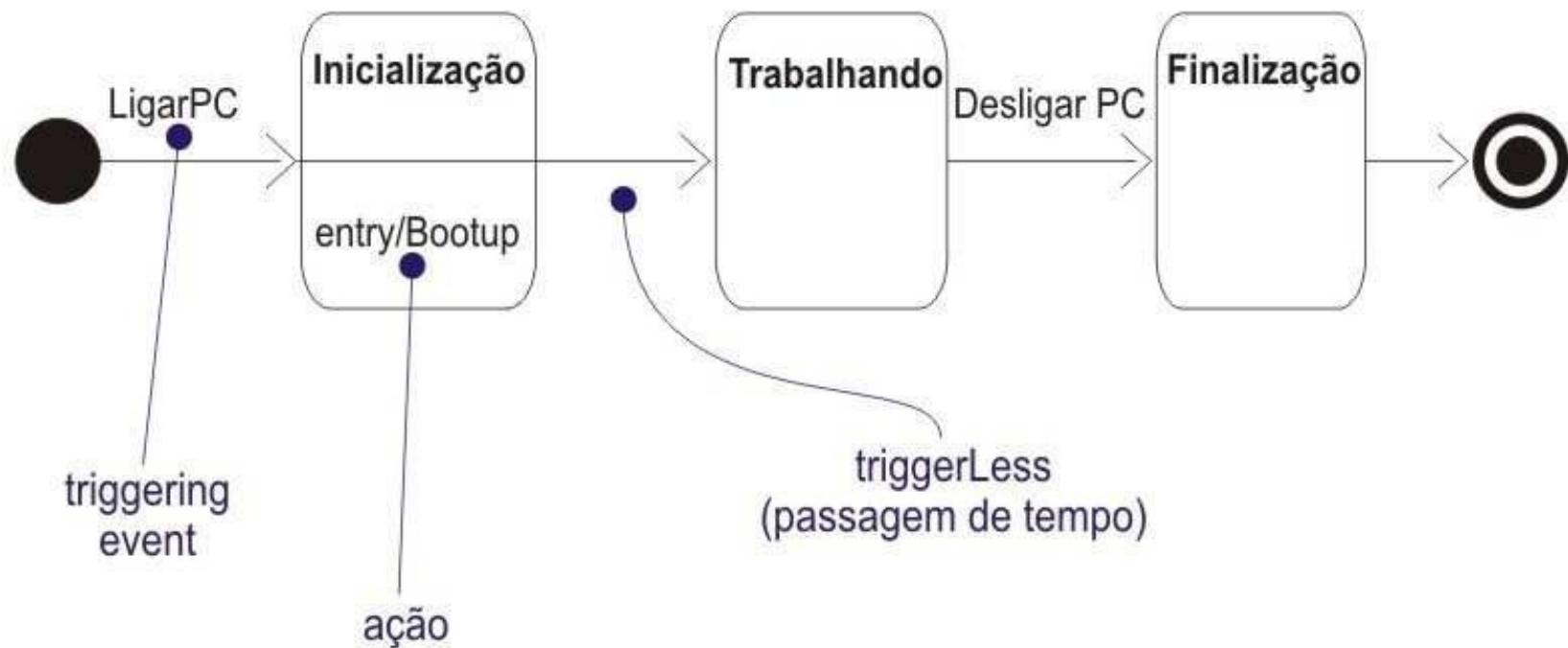


# Diagramas de Estado

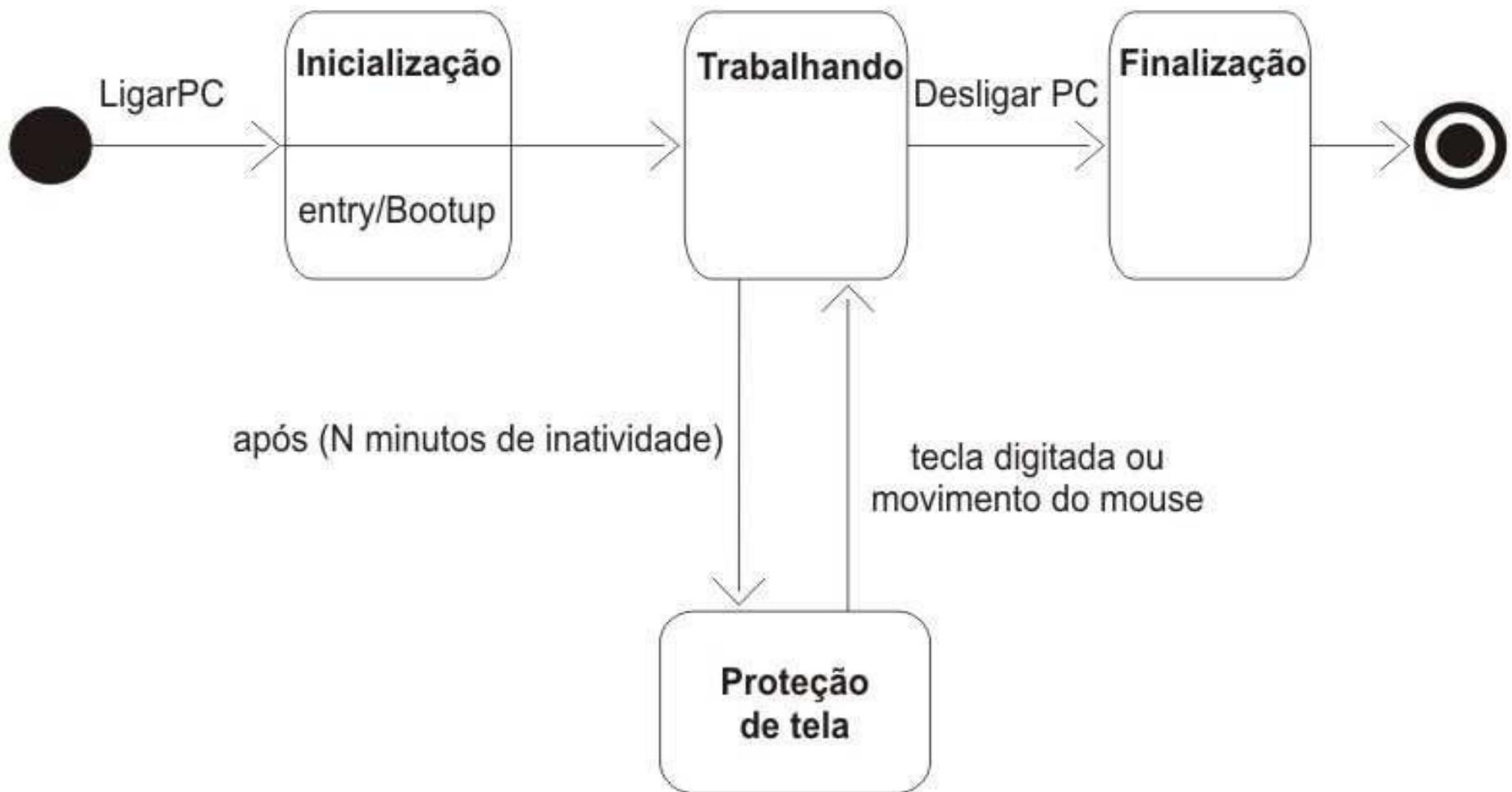


Uma lâmpada: que evolui entre os estados “acesa” e “apagada”, conforme se liga e desliga um interruptor

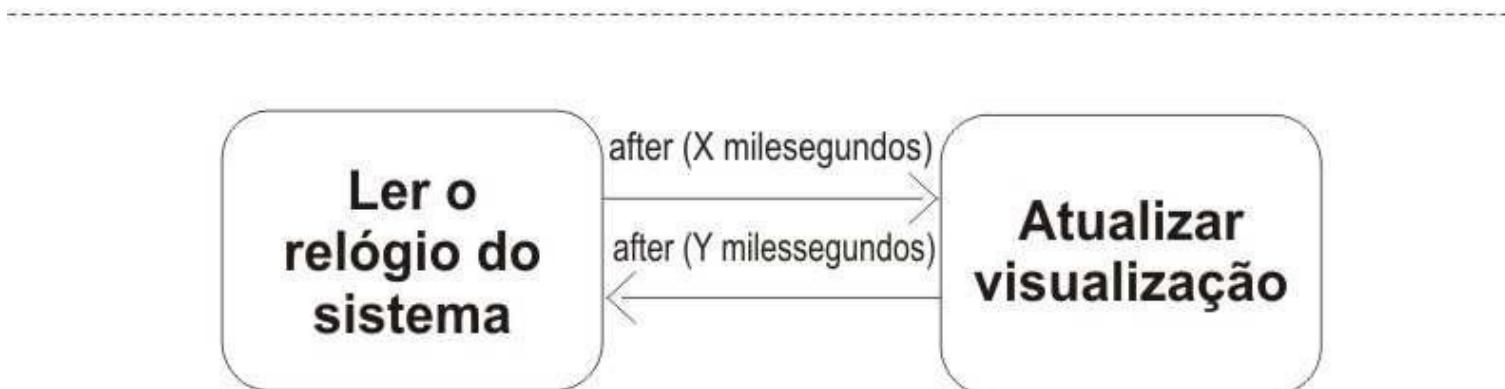
# Diagrama de Estados de um PC



# Diagrama de Estados de um PC (variante do anterior)



# Diagrama de Estados de um PC (foco no estado “Trabalhando”)



# Exercício – Faça um diagrama de Estados para a situação:

1. Uma máquina de lavar roupa: depois da passagem de um determinado período de tempo, a máquina de lavar termina o seu programa de lavagem, e inicia o de secagem.

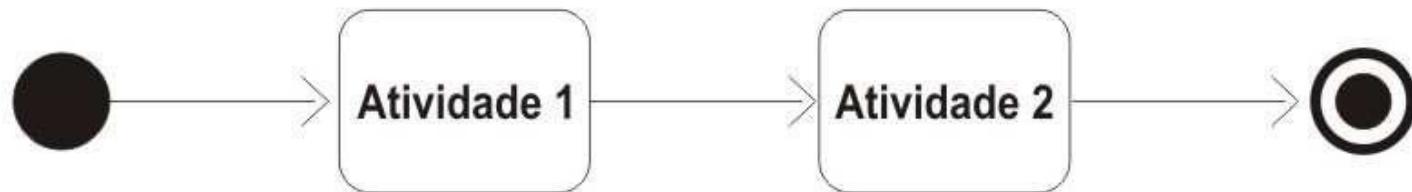
# Diagrama de Atividades

- Um diagrama de atividades é um caso particular de um diagrama de estados, no qual todos ou a maioria dos estados são “estados de atividades” e todas ou a maioria das transições são desencadeadas pela conclusão das atividades dos estados anteriores;
- Uma atividade corresponde a uma execução não atômica dentro de uma máquina de estados, ou de outra forma, corresponde à execução de um conjunto de ações.

# Diagrama de Atividades x Estados

- Ambos os tipos de diagramas são utilizados para modelar o tempo de vida de um objeto ou sistema.
- Contudo, um diagrama de atividades ilustra o **fluxo de controle entre atividades**, enquanto que um diagrama de estados **ilustra o fluxo de controle entre estados**.

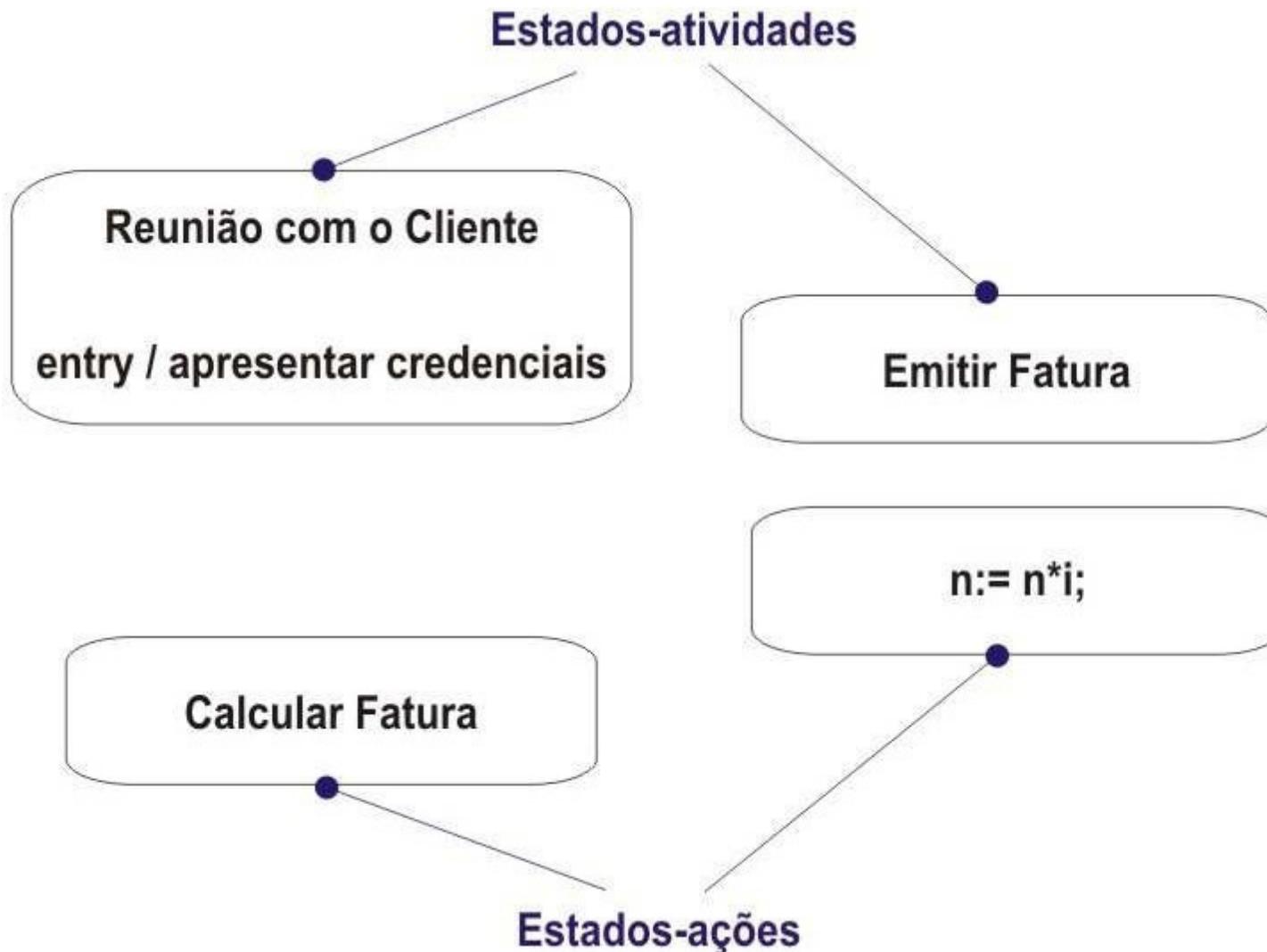
# Exemplo Genérico de Diagrama de Atividades



# Diagramas de Atividades

- Estes diagramas contêm genericamente:
  - **Estados-ação**: execuções atômicas, não interrompíveis, com tempo de execução irrelevante.
  - **Estados-atividade**: execuções não atômicas (decompostas), interrompíveis, em que o tempo de execução é normalmente relevante.
  - **Transições**,
  - **Objetos**.

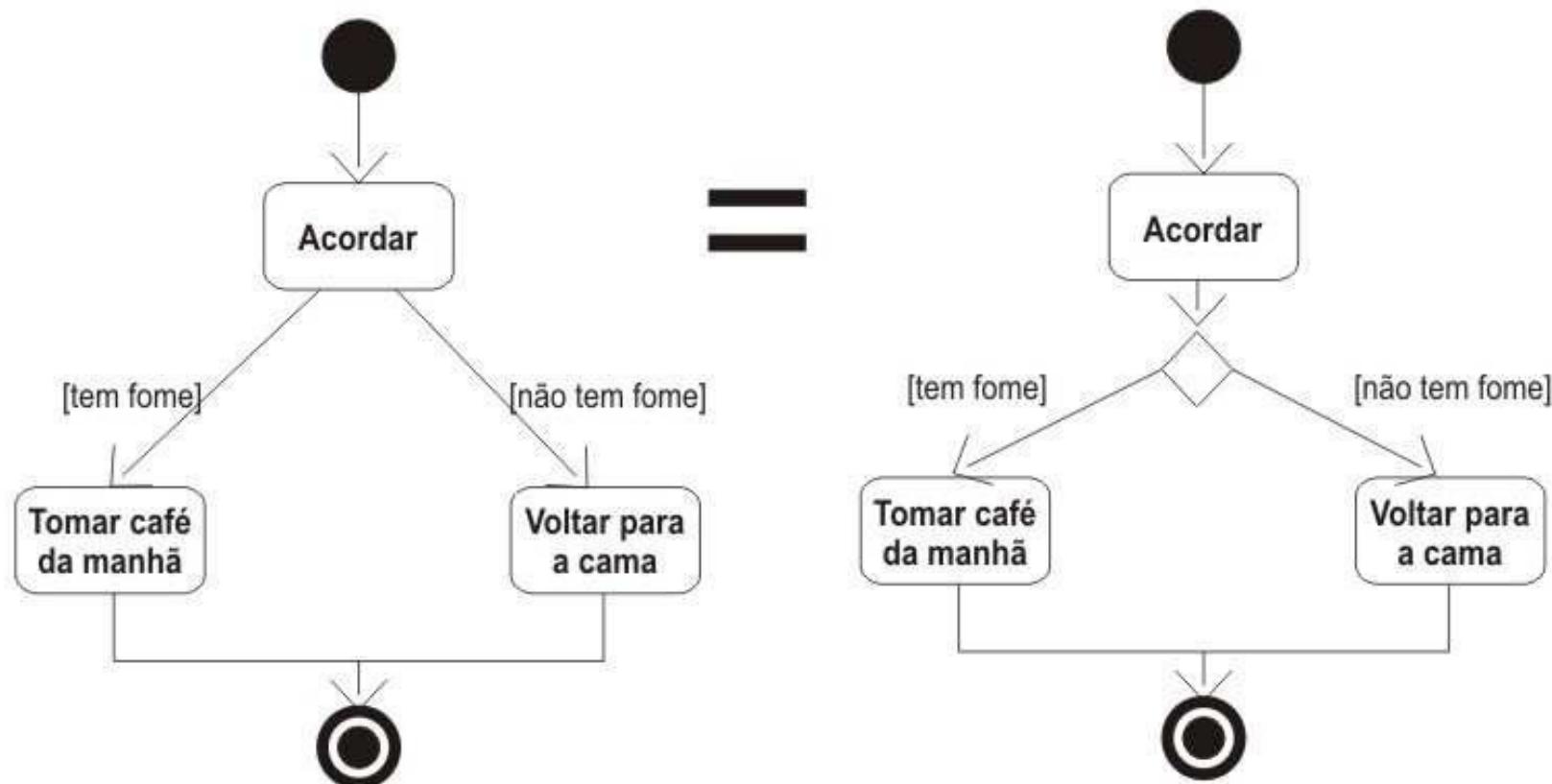
# Estados-atividades X estados-ações



# Diagramas de Atividades (Decisões)

- A tomada de decisão é um mecanismo comum no projeto de diagramas de atividades (e de estado), que consiste em especificar que atividade deve ser realizada após a execução da atividade corrente.
- Tal especificação é suportada por uma condição com guarda (ex.: expressão lógica) que é colocada de forma adjacente à transição de estado correspondente.

# Diagramas de Atividades (Decisões) - Exemplo

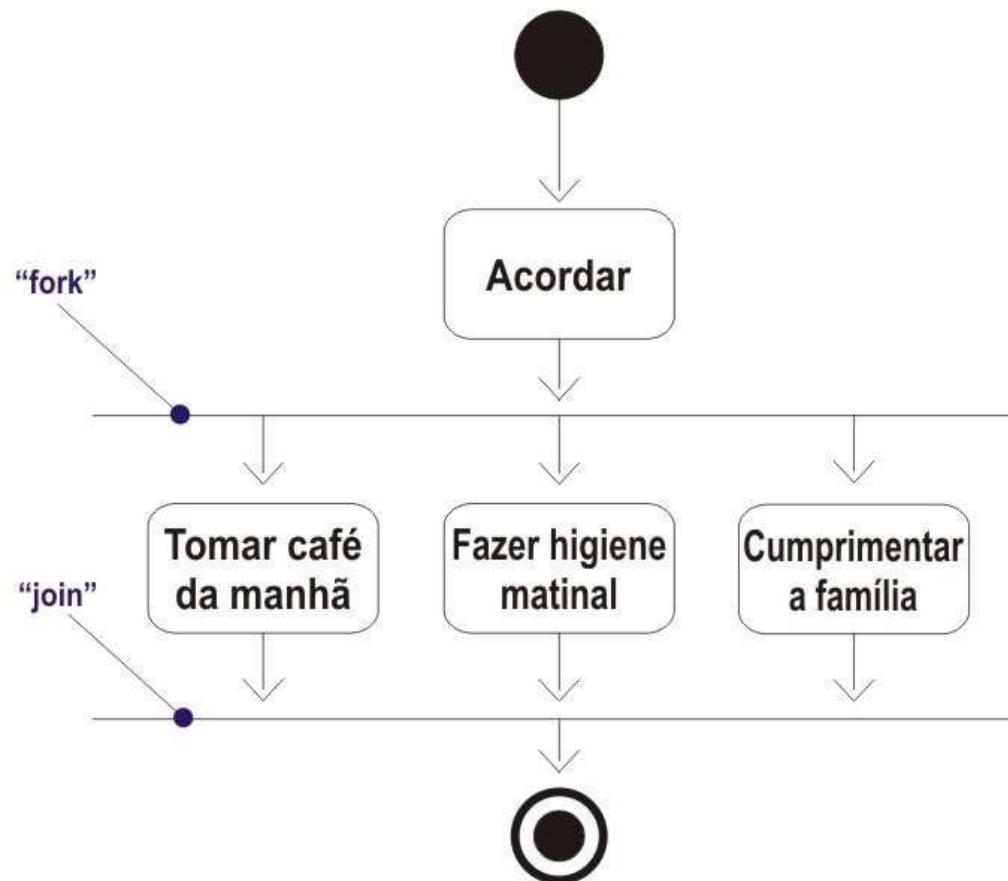


# Diagramas de Atividades (Caminhos Concorrentes )

- Considere que o processo de “levantar da cama” implica a execução das seguintes atividades “**tomar café da manhã**”, “**fazer a higiene matinal**” e “**cumprimentar a família**”.
- Considere que essas atividades têm de se realizar obrigatoriamente, embora não seja relevante a sua ordem de execução.
- O problema colocado representa uma situação na modelagem de workflows: representar a execução independente e concorrente de um conjunto de atividades.

# Diagramas de Atividades (Caminhos Concorrentes) – cont.

A UML providencia a solução a esta questão através dos conceitos de difusão (fork) e de junção (join) de atividades, representados graficamente por linhas



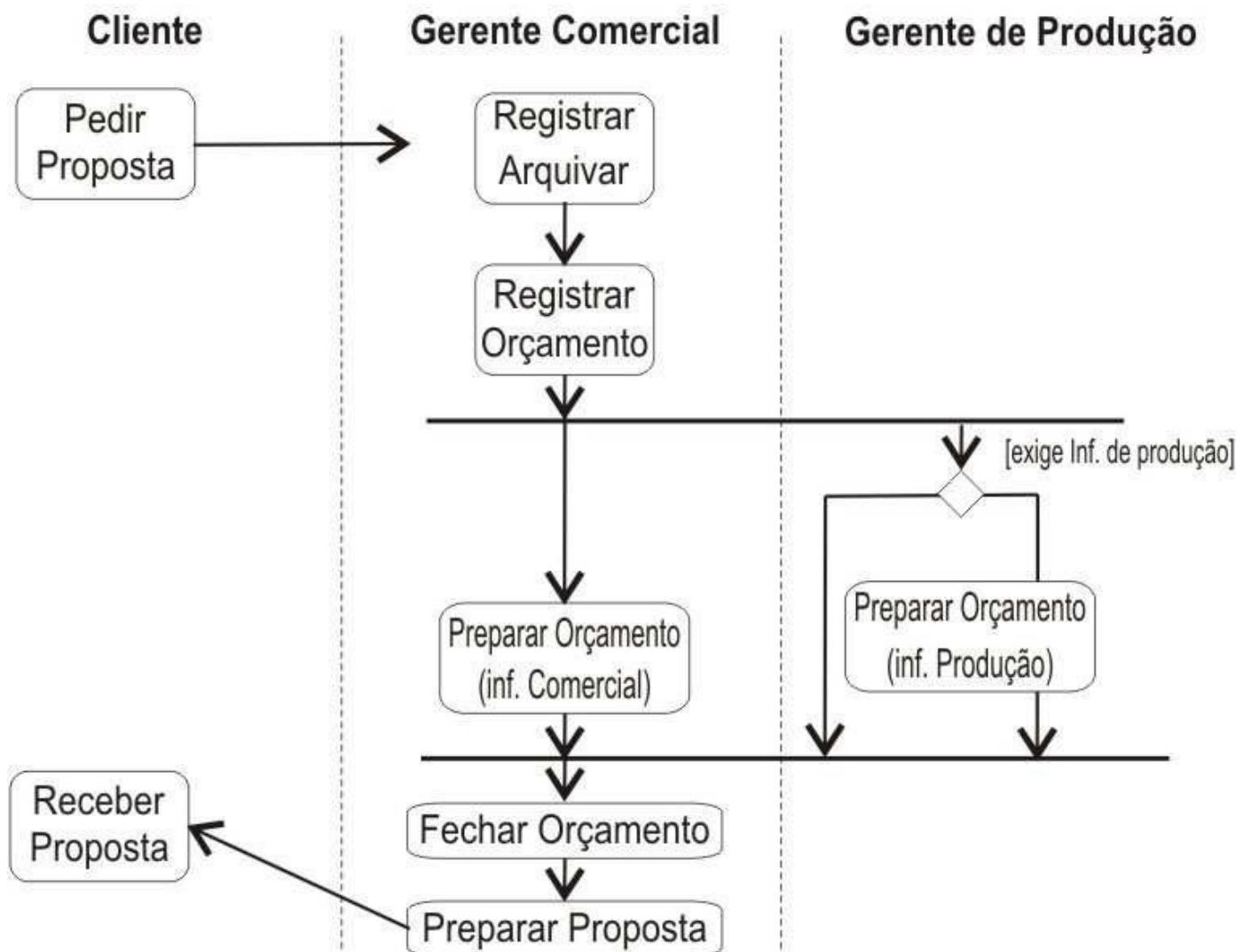
# Diagramas de Atividades (Pistas)

- Na modelagem de processos de negócio é comum a realização de atividades por várias entidades, participantes no dito processo.
- A UML propõe o conceito de pistas (swimlanes) como elemento que permite agrupar as várias atividades da responsabilidade de cada entidade participante. Cada grupo é separado por uma linha vertical.

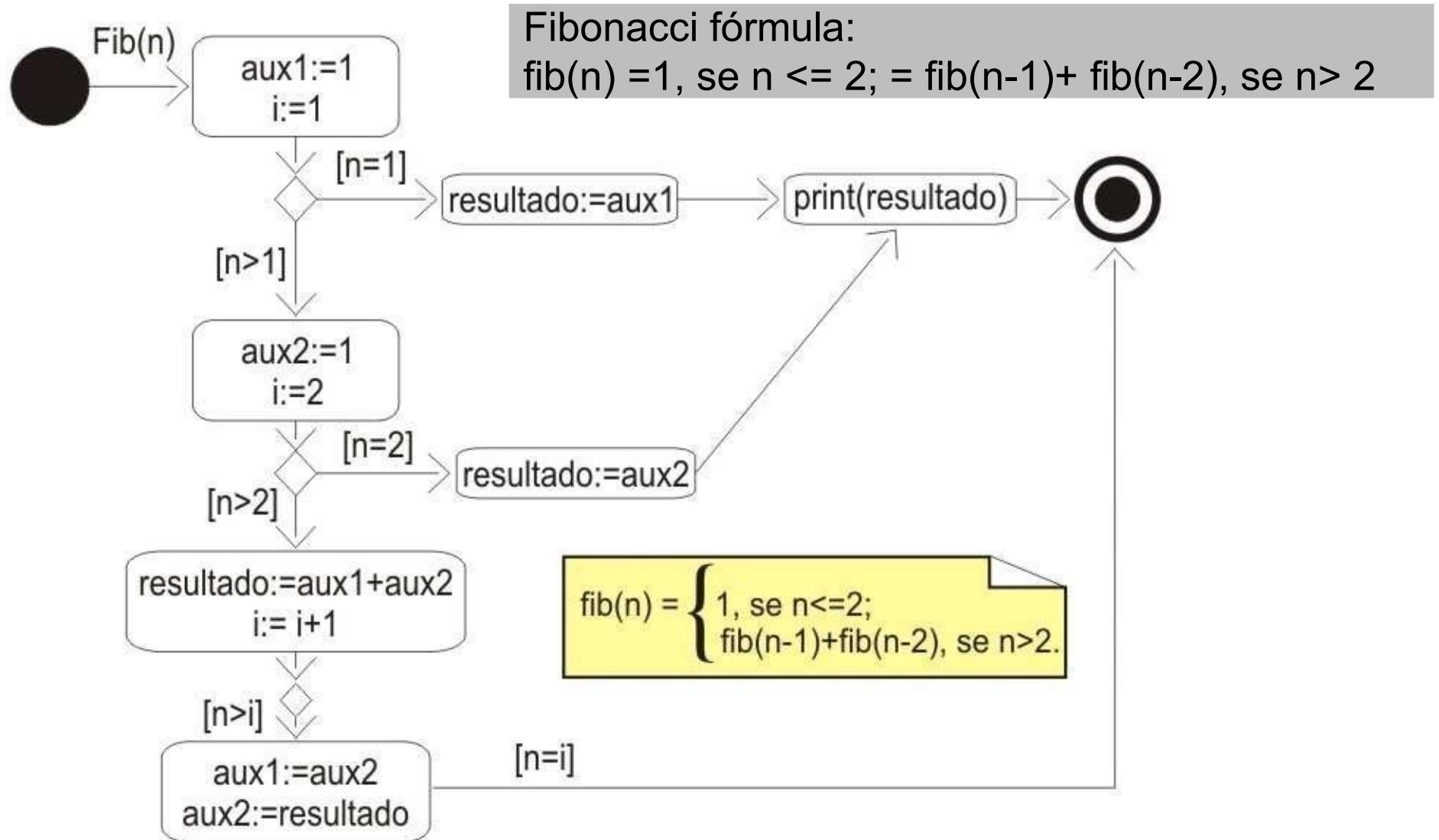
# Diagrams de Atividades (Pistas) – Cont.

- Cada pista tem um nome único dentro do seu diagrama, que deve corresponder ao nome da entidade participante, a qual deve ser uma entidade do mundo real.
- Por exemplo, o nome de um perfil de usuário, o nome de uma organização, ou o nome de um sistema de informação.

# Diagramas de Atividades (Pistas) – Exemplo

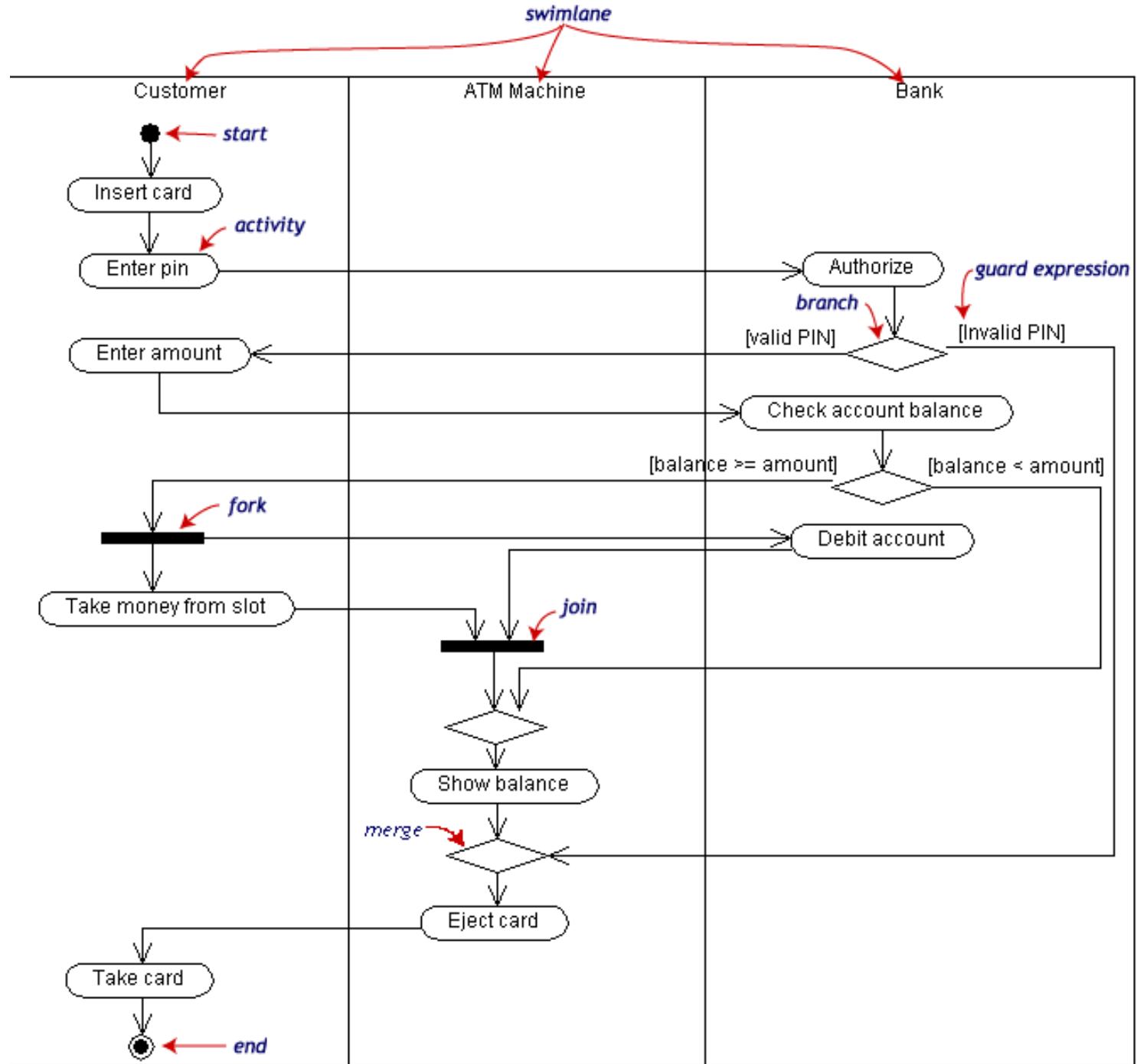


# Exemplo de Diagrama de Atividades da Operação de Fibonacci



# Exercício

- Faça um diagrama de Atividades para a situação:
  - Retirando dinheiro de um caixa eletrônico  
(para cartões de crédito)



# Diagramas de Componentes

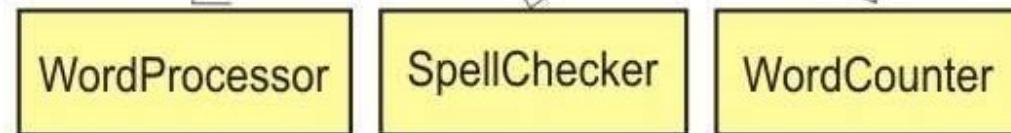
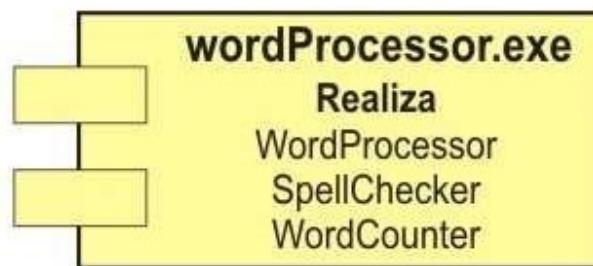
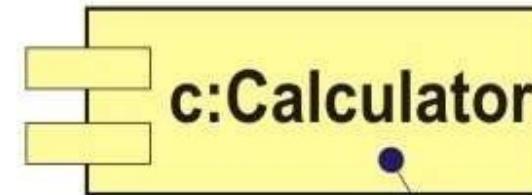
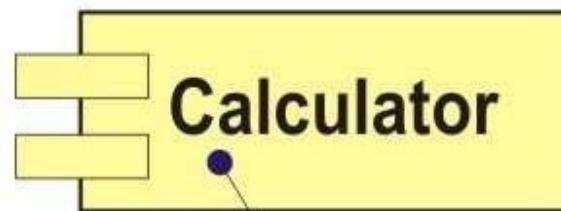
- Um componente é uma peça básica na implementação de um sistema; consiste, na prática, num conjunto de artefatos físicos em formato digital, por exemplo, arquivos de código (fonte, binário ou executáveis) ou arquivos de documentos relativos ao negócio.

# Diagramas de Componentes - Cont.

Definem-se pelo menos três tipos distintos de componentes:

- Componentes de instalação: constituem a base dos sistemas executáveis (e.g., DLL, executáveis, controles Active-X, classes Java).
- Componentes de trabalho: a partir dos quais são criados os componentes de instalação (e.g., arquivos com código fonte, arquivos de dados, documentos).
- Componentes de execução: criados como resultado da execução de um sistema (e.g., processos, threads, agentes de software).

# Representação Gráfica de Componentes



# Componente de Software

- Um componente de software é uma parte física de um sistema: existe de fato num determinado computador e não apenas na mente do analista, como acontece com o conceito de classe.
- Adicionalmente, um componente implementa uma ou mais classes, as quais são representadas dentro do ícone de componente ou com relações explícitas de dependência de implementação.

# Componente de Software – Cont.

A UML identifica os seguintes estereótipos para componentes:

- «document»: denota um documento.
- «executable»: denota um programa que possa ser executado num nó.
- «file»: denota um documento contendo código fonte ou dados.
- «library»: denota uma biblioteca dinâmica ou estática.
- «table»: denota uma tabela de uma base de dados.

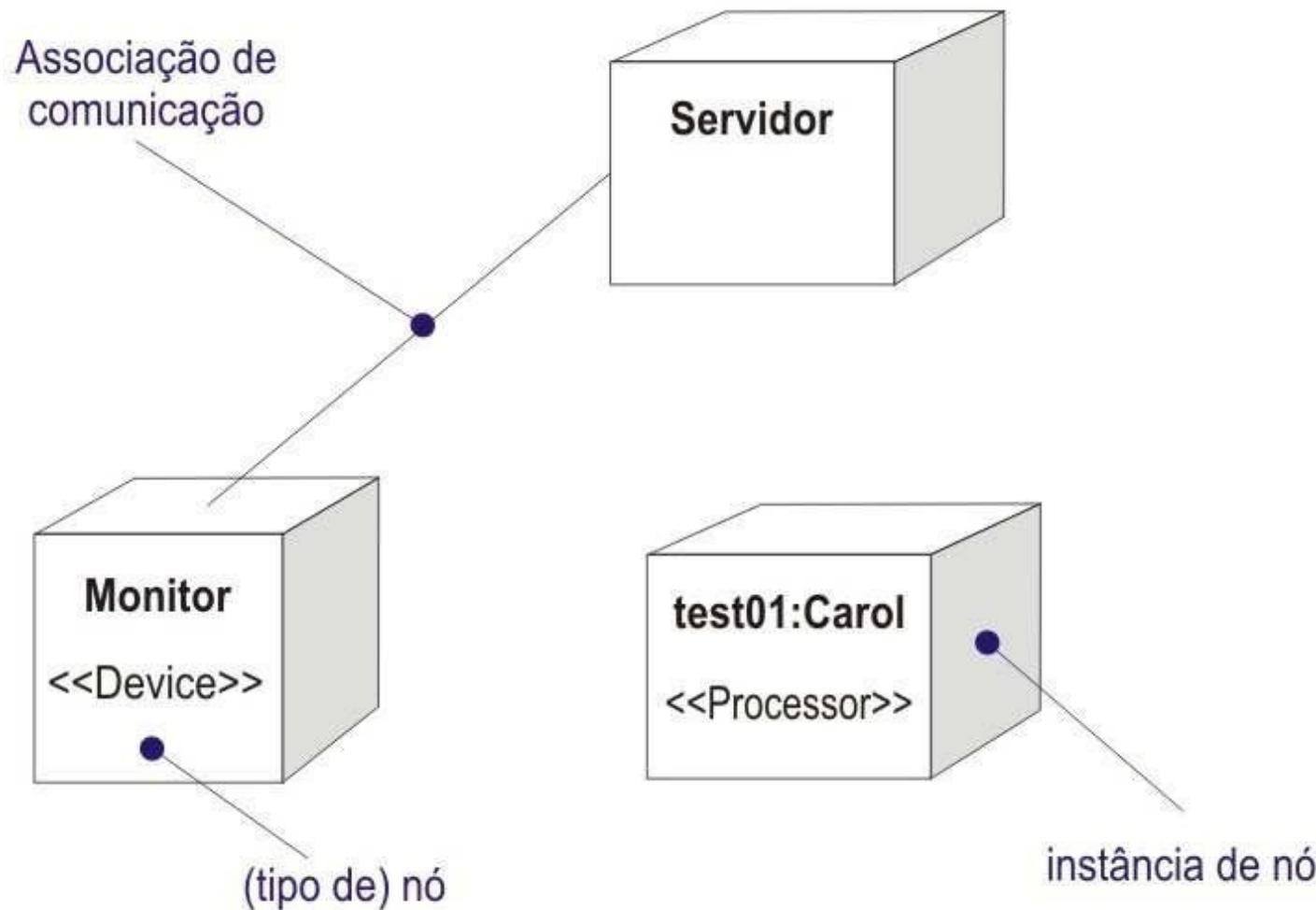
# Diagramas de Componentes (Nós)

- Um nó é um objeto físico que representa um recurso de processamento, geralmente tendo capacidades de memória e de processamento.
- Os nós podem consistir em recursos computacionais (hardware), mas também em recursos humanos ou recursos de processamento mecânico.
- Os nós podem ser representados como tipos e como instâncias. Instâncias de nós podem conter instâncias de objetos e de componentes.

# Diagramas de Componentes (Nós) – Cont.

- Um nó é representado como um cubo tridimensional.
- Dois nós podem-se encontrar ligados através de relações de associação. Estas especificam a existência de caminhos de comunicação entre os correspondentes nós e podem ser caracterizadas por um estereótipo, de modo a explicitar o tipo de comunicação envolvido (ex.: o tipo de canal ou o tipo de rede).

# Diagramas de Componentes (Nós)



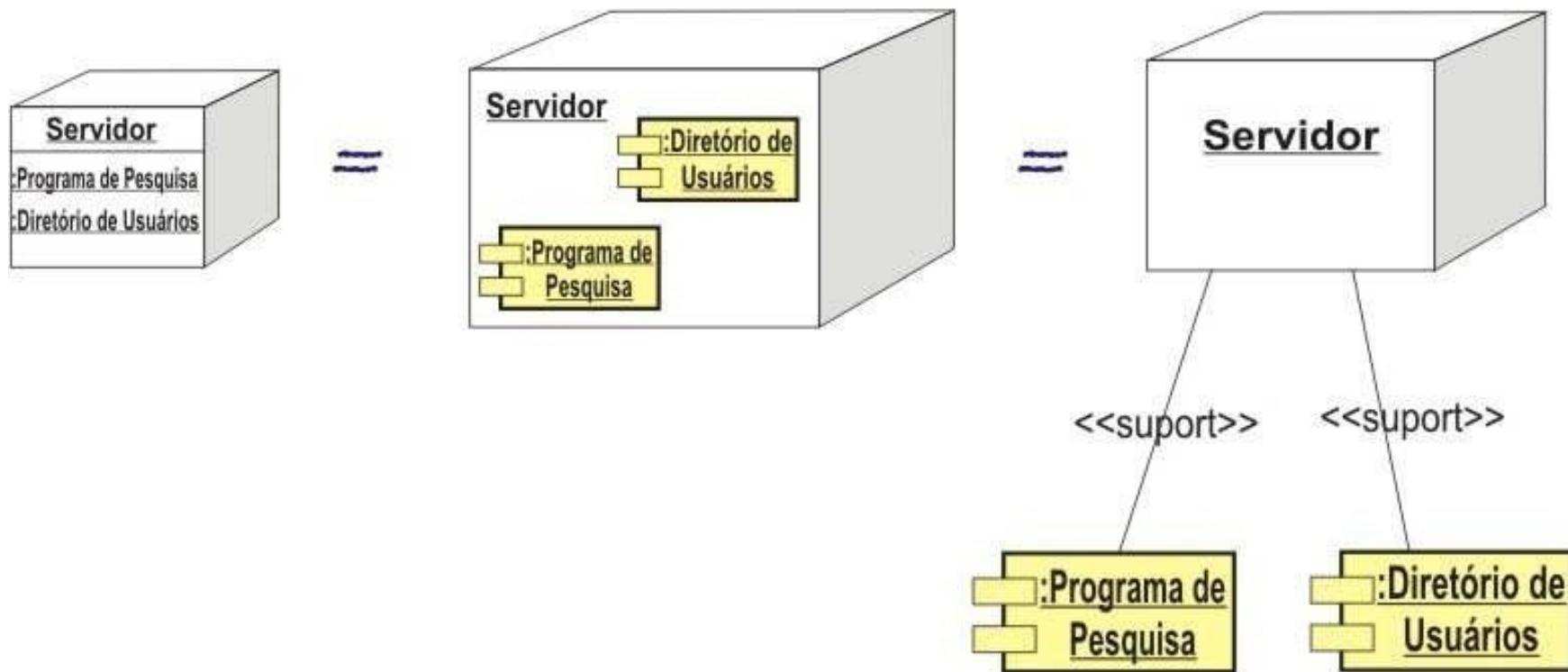
# Diagramas de Componentes (Nós) – Cont.

- As propriedades dos nós (ex.: capacidade de memória principal, número de processadores, data de aquisição, entre outros) são representadas por marcas com valores. Por outro lado, podem-se definir estereótipos, com ícones correspondentes, para modelar diferentes tipos de recursos de processamento.

# Relações entre Nós e Componentes

- Um nó pode conter componentes. Tal fato pode ser traduzido pela inclusão dos componentes no símbolo do nó, ou pelo estabelecimento de uma relação de dependência, de estereótipo «support» entre o nó e os componentes suportados.

# Relações entre Nós e Componentes



# Nós e Componentes

## Semelhanças e Diferenças

- As semelhanças são que ambos podem:
  - (1) participar em relações de generalização, dependência e associação;
  - (2) ser aninhados;
  - (3) ter instâncias; e
  - (4) participar em interações.
- As diferenças são que os:
  - (1) componentes são elementos que participam na execução de um sistema; nós são elementos que suportam e executam componentes; e
  - (2) componentes representam agrupamento físico de elementos lógicos; nós representam a instalação física de componentes.

# Diagramas de Componentes (Exemplo)

- Diagrama de Componentes relativo a uma Página HTML.
- Considere a página Web Example1.html com uma referência a um applet Java:

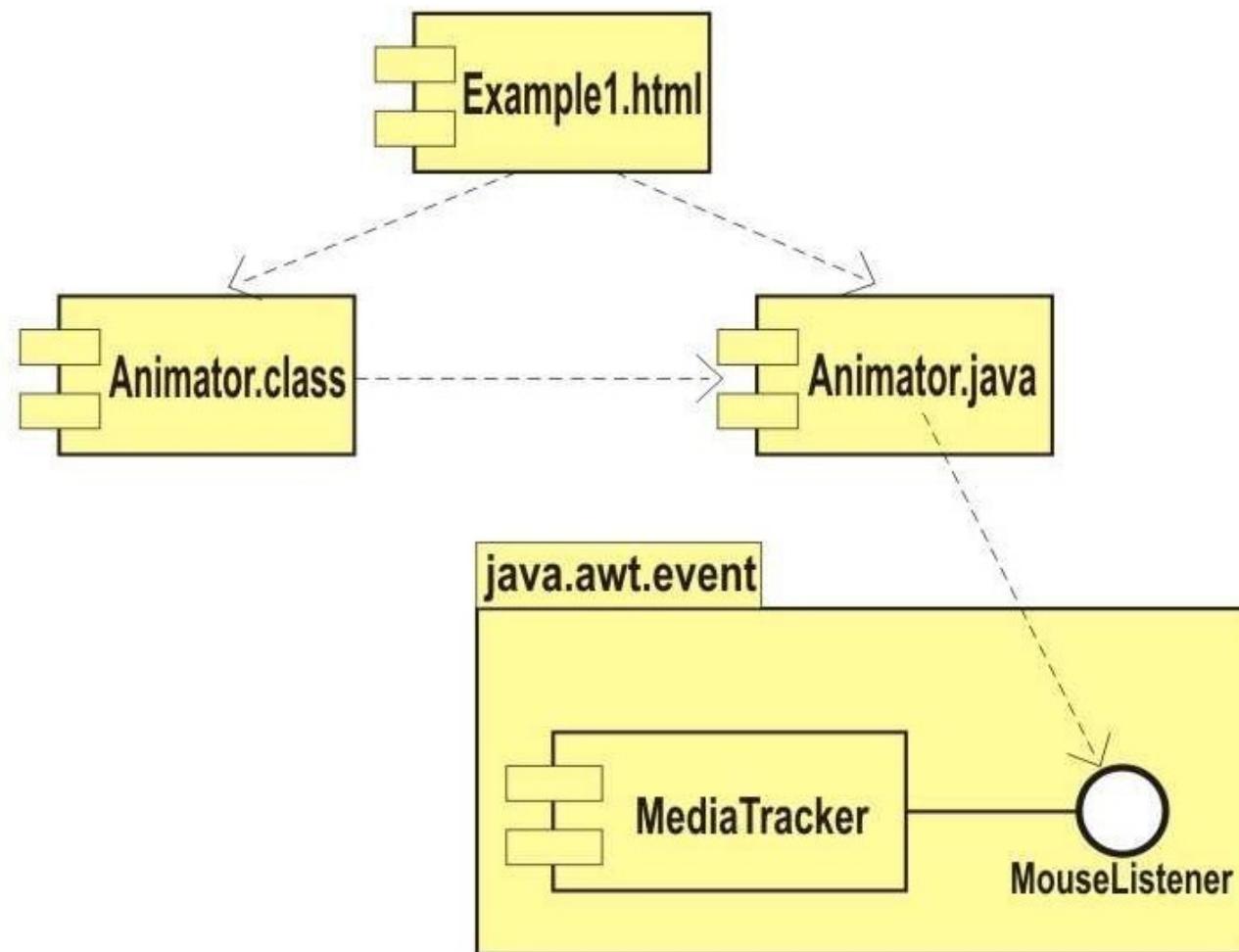
# Página Web Example1.html

```
<html>
  <head>
    <title>The Animator Applet (1.1) – example 1</title>
  </head>
  <body>
    <h1>The Animator Applet (1.1) - example 1</h1>
    <applet codebase=". " code=Animator.class width=460 height=160>
    </applet>
    <a href="Animator.java">The source.</a>
    <hr>
  </body>
</html>
```

# Diagramas de Componentes (Exemplo) Página Web

- O diagrama de componentes correspondente a este “mini-sistema” consiste nos seguintes arquivos: example1.html, Animator.class, e Animator.java.

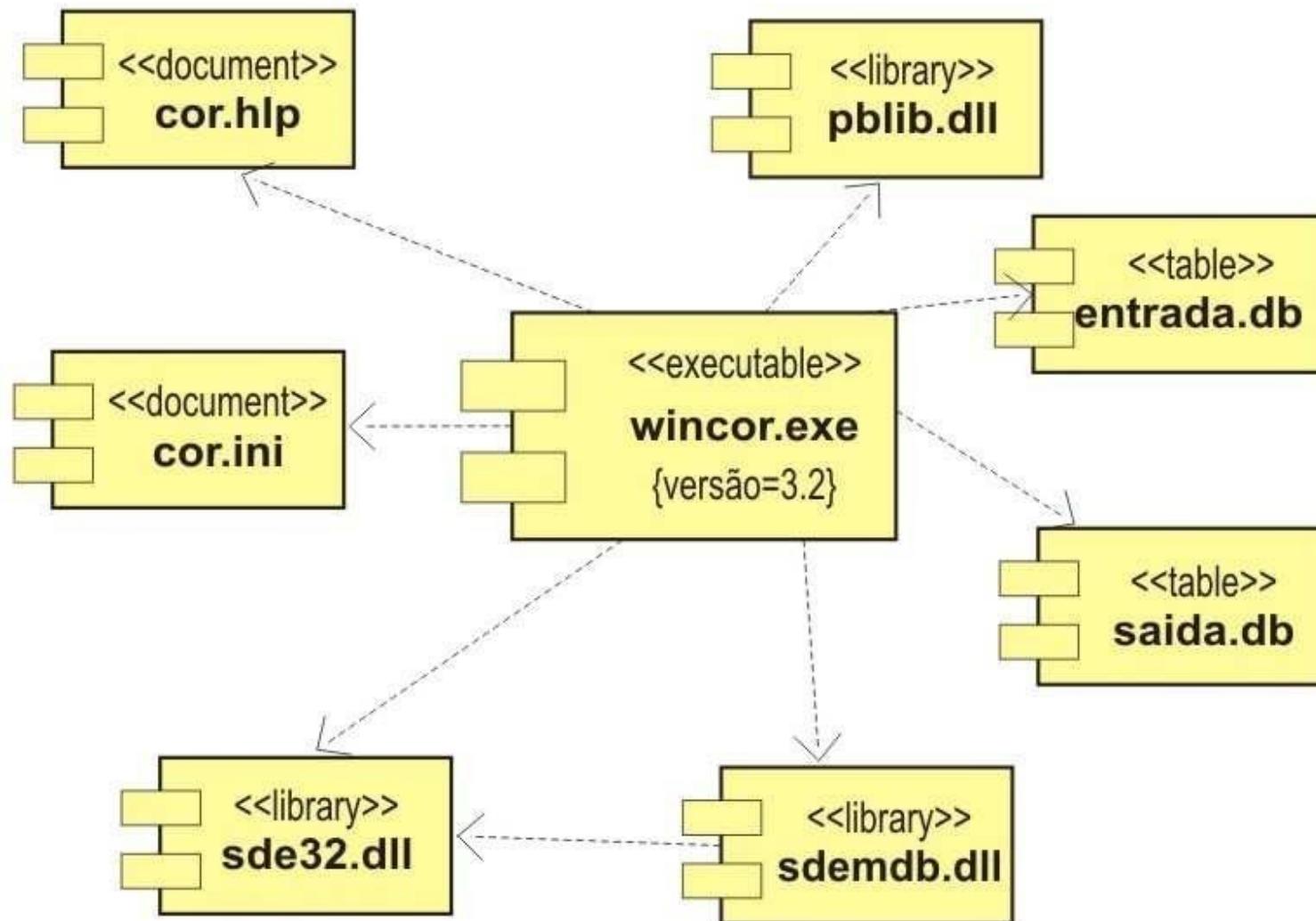
# Diagramas de Componentes (Exemplo) Página Web



# Exercício: Faça um Diagrama de Componentes relativo à instalação de uma aplicação.

- Considere a aplicação WinCOR desenvolvida sobre ambiente MSWindows e responsável pelo gerenciamento de (entrada e saída de) correspondência de uma organização. A aplicação consiste num conjunto variado de componentes de instalação, chamados:
  - wincor.exe: arquivo que contêm o executável da aplicação;
  - pplib.dll, sde32.dll, sdemdb.dll: bibliotecas com código binário que providenciam funcionalidades adicionais;
  - wincor.hlp: arquivo de ajuda sobre a aplicação;
  - wincor.ini: arquivo de configuração da aplicação;
  - entrada.db, saida.db: arquivos/tabelas da base de dados de suporte

# Resposta



# Diagrams de Instalação

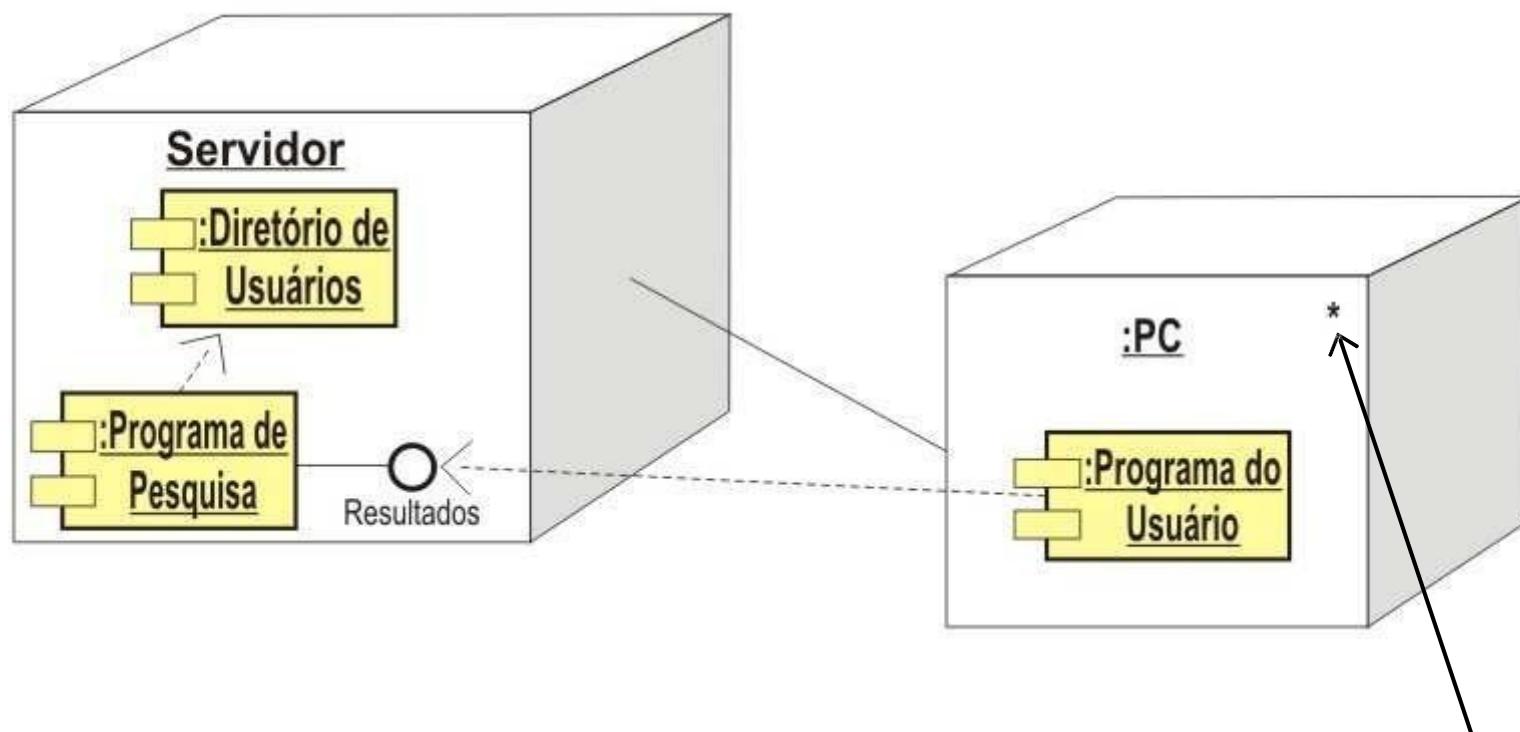
- Um diagrama de instalação ilustra a configuração dos elementos de processamento e dos componentes de software, processos e objetos neles suportados. Instâncias de componentes de software representam manifestações de execução das unidades de código.

# Diagrams de Instalação

- Um diagrama de instalação consiste num conjunto de nós ligados por associações de comunicação.
- Os nós podem conter instâncias de componentes (de execução), o que significa que um componente é instalado e executado num nó.

# Diagramas de Instalação (Exemplo 1)

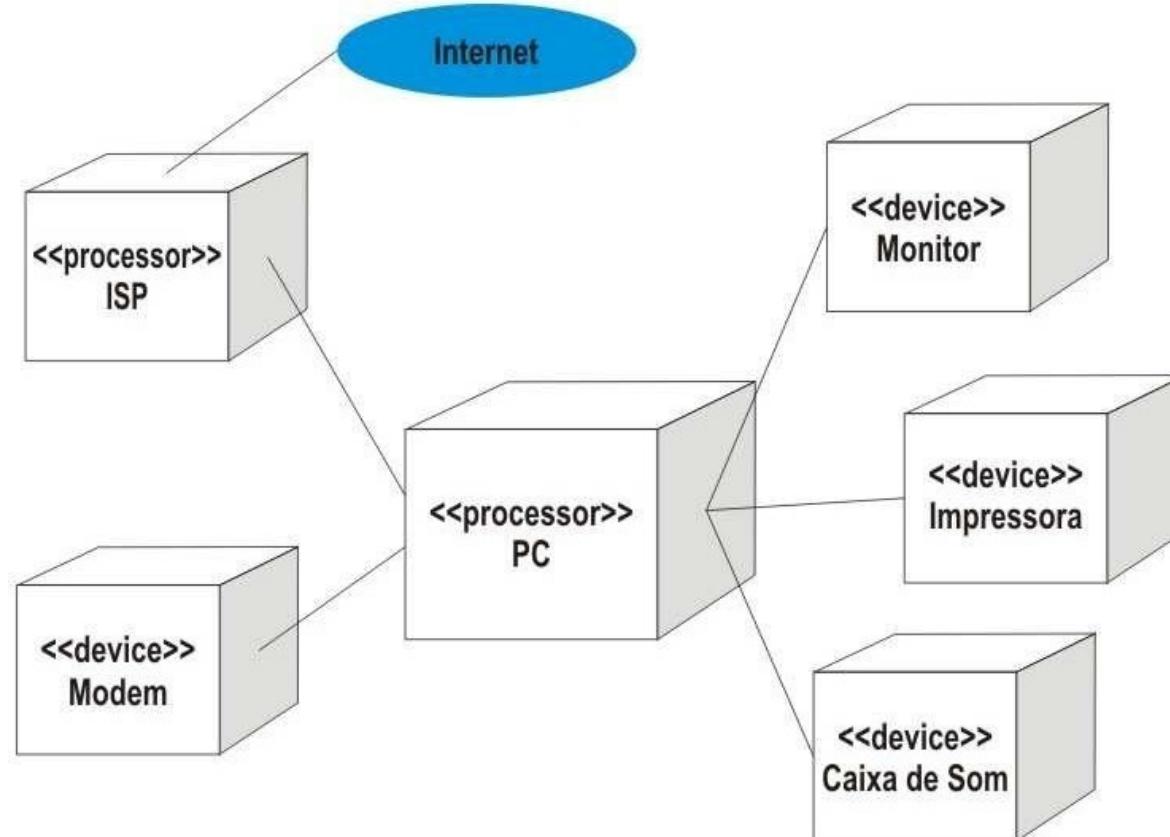
**serviço de conexão a Internet por linha telefônica - cliente/servidor**



**instalação, da existência de vários PC**

# Diagramas de Instalação (Exemplo 2)

Equipamento de hardware tipicamente existente numa configuração doméstica



# Diagramas de Instalação (Exemplo 3)

