



Diagrama de Estado

Universidade Federal do Maranhão - UFMA

Departamento de Informática

Processo de Desenvolvimento de Software

Prof^a.MSc Simara Rocha

simararocha@gmail.com/simara@deinf.ufma.br

<http://www.deinf.ufma.br/~simara>

Referências: Booch, G. et al. The Unified Modeling Language User Guide
Medeiros, E. Desenvolvendo Software com UML 2.0: Definitivo, Makron Books, 2006.



Definições

- Trata-se de um complemento para a descrição das classes, documentando os **estados possíveis que objetos de uma certa classe podem assumir**, além de mostrar ainda os eventos do sistema que geram tais mudanças.
- **Os diagramas de estado não são escritos para todas as classes de um sistema**, mas apenas para aquelas que possuem um número definido de estados conhecidos e onde o comportamento das classes é afetado e modificado pelos diferentes estados.



Definições

- Os diagramas de estado permitem especificar a dinâmica de um sistema
 - devem reunir o comportamento completo de uma classe em todos os casos de uso, ou seja, deve-se fazer um levantamento de todos os casos de uso onde esta classe participa.
- Dessa forma, o diagrama de estados é uma descrição global do comportamento dos objetos desta classe em todo o sistema.



Comportamento

- Através da análise da mudança de estados dos tipos de objetos de um sistema, podemos prever todos os possíveis comportamentos de um objeto de acordo com os eventos que o mesmo possa sofrer.
- Diagramas de estado capturam o ciclo de vida dos objetos, subsistemas e sistemas.



Visualização

- Eles mostram os estados que um objeto pode possuir e como os eventos (mensagens recebidas, timer, erros, e condições sendo satisfeitas) afetam estes estados ao passar do tempo.
- Todos os objetos possuem um estado que significa o resultado de atividades executadas pelo objeto, e é normalmente determinada pelos valores de seus atributos e ligações com outros objetos.



Quando usar

- Se o relacionamento de classes não está claro o suficiente em função do estado dos objetos, isso será uma pista de que deve usar este diagrama. Essa percepção é pessoal
- Diagrama de Estados enfatizam os estados dos objetos e as transições entre estes estados enquanto o Diagrama de Atividades enfatiza o fluxo de controle de uma atividade para outra.



Evento

- Um objeto muda de estado quando acontece algo, o fato de acontecer alguma coisa com o objeto é chamado de evento.
- Os objetos de uma classe habitualmente possuem um ciclo de vida: são gerados, assumem posições durante a sua vida, dão origem a outros objetos em classes relacionadas e deixam de existir no momento de sua destruição.



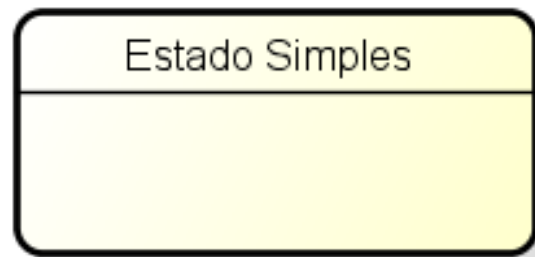
Notações

- Os estados são representados graficamente por retângulos com bordas arredondadas e as transições são representadas por setas.
- Um estado pode ser compreendido como um momento na vida de um objeto, onde este se encontra em uma determinada situação.
- Já a transição de estados é o avanço que um objeto faz ao passar de um estado para outro.



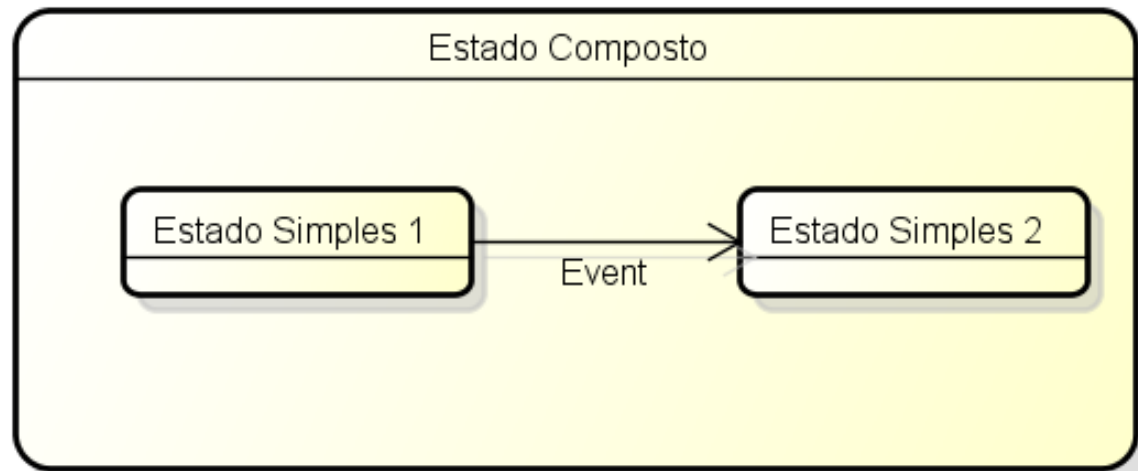
O Estado

- Os nomes dos estados geralmente são identificados como verbos no gerúndio (ex.: “calculando valor”) ou no particípio (ex.: “dados recebidos”), sendo que aqueles estados com nome no particípio geralmente podem ser eliminados, pois são opcionais.
- O estado tem 2 compartimentos, um com seu nome e o outro com suas ações internas.



Estado Composto

- Um retângulo maior designa um estado composto ou máquina de estado, significando que nele existirão outros estados.





Transições

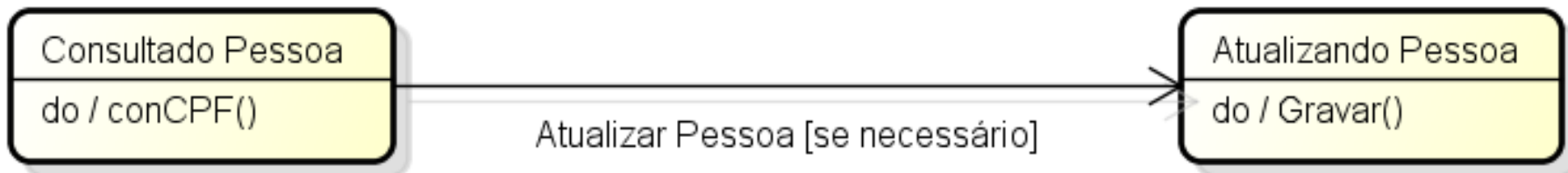
- Uma transição representa um evento que causa a mudança de estado de um objeto.
 - Evento de Ativação





Evento de Transição

- A transição de estados é criada a partir de um evento
 - Este pode ou não conter uma descrição
 - Pode conter uma ordem para realizar uma tarefa
 - Pode também conter condições de guarda

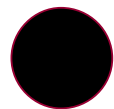


powered by astah* 



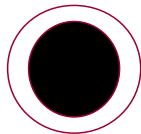
Estados de Início/Fim

- Um círculo preenchido designa um estado inicial chamado de pseudoestado inicial.
- O estado final é representado por um círculo vazado e, dentro deste, um círculo cheio. O último estado de um objeto será representado por este estado final.
- Diagramas de Estado possuem um único Estado Inicial e um ou mais Estados Finais



Início

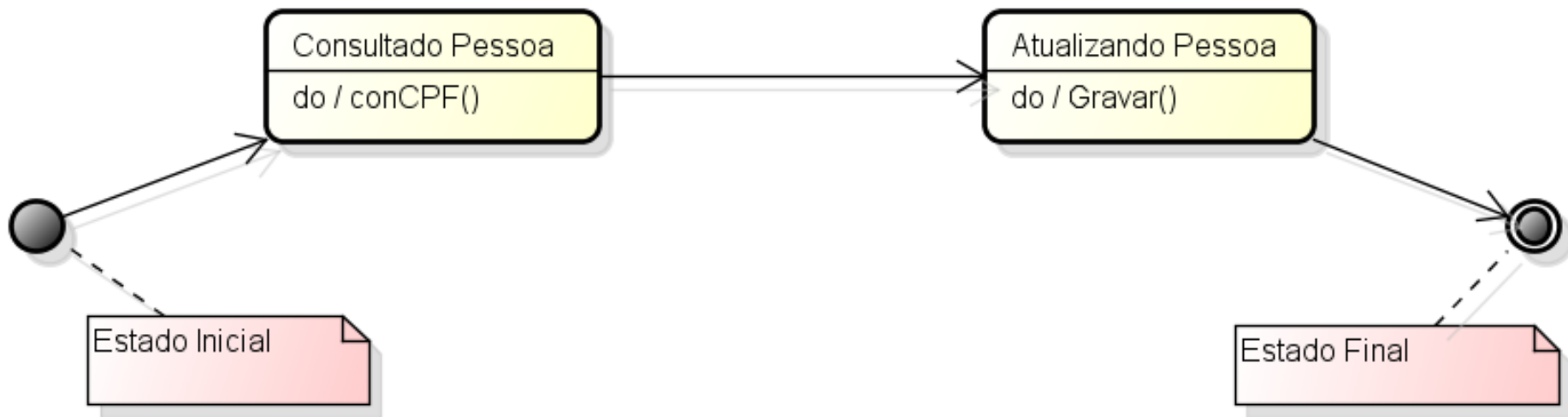
→ Estado Inicial



Fim

→ Estado Final

Estados de Início/Fim



powered by astah*



Notações

- Um Estado tem várias partes:
 - Nome.
 - Ações de entrada/saída: ações executadas na entrada ou saída do estado, respectivamente.
 - Transições internas: transições que não causam a mudança do estado.



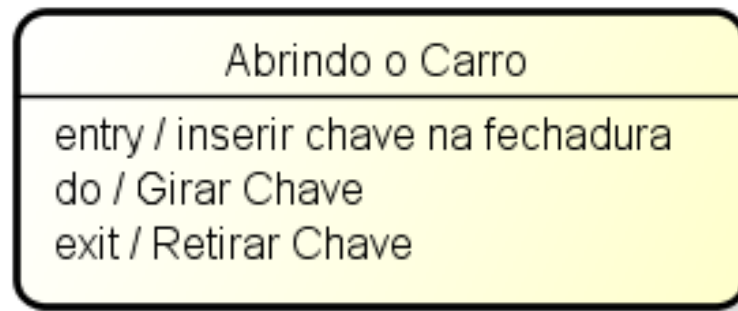
Notações

- Um Estado tem várias partes:
 - Sub-estados: a estrutura aninhada de um estado, envolvendo sub-estados disjuntos (seqüencialmente ativos) ou concorrentes (concorrentemente ativos).
 - Eventos adiados: uma lista de eventos que não são manipulados neste estado, mas, em vez disso, são adiados e colocados em fila para serem manipulados pelo objeto em outro estado.



Ações Internas

- Entry: ações realizadas no momento em que o objeto assume o estado em questão
- Exit: ações realizadas antes do objeto mudar de estado
- Do: atividades executadas enquanto o objeto se encontra em determinado estado



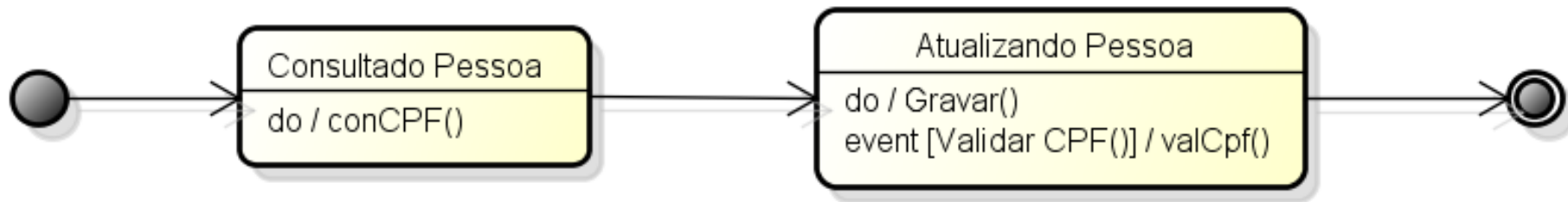


Dicas

- Ações são associadas com transição e são considerados como processos de curta duração, não podendo ser interrompidos
- Atividades são associados a estados e podem levar mais tempo. Uma atividade pode ser interrompida por algum evento.

Transições Internas

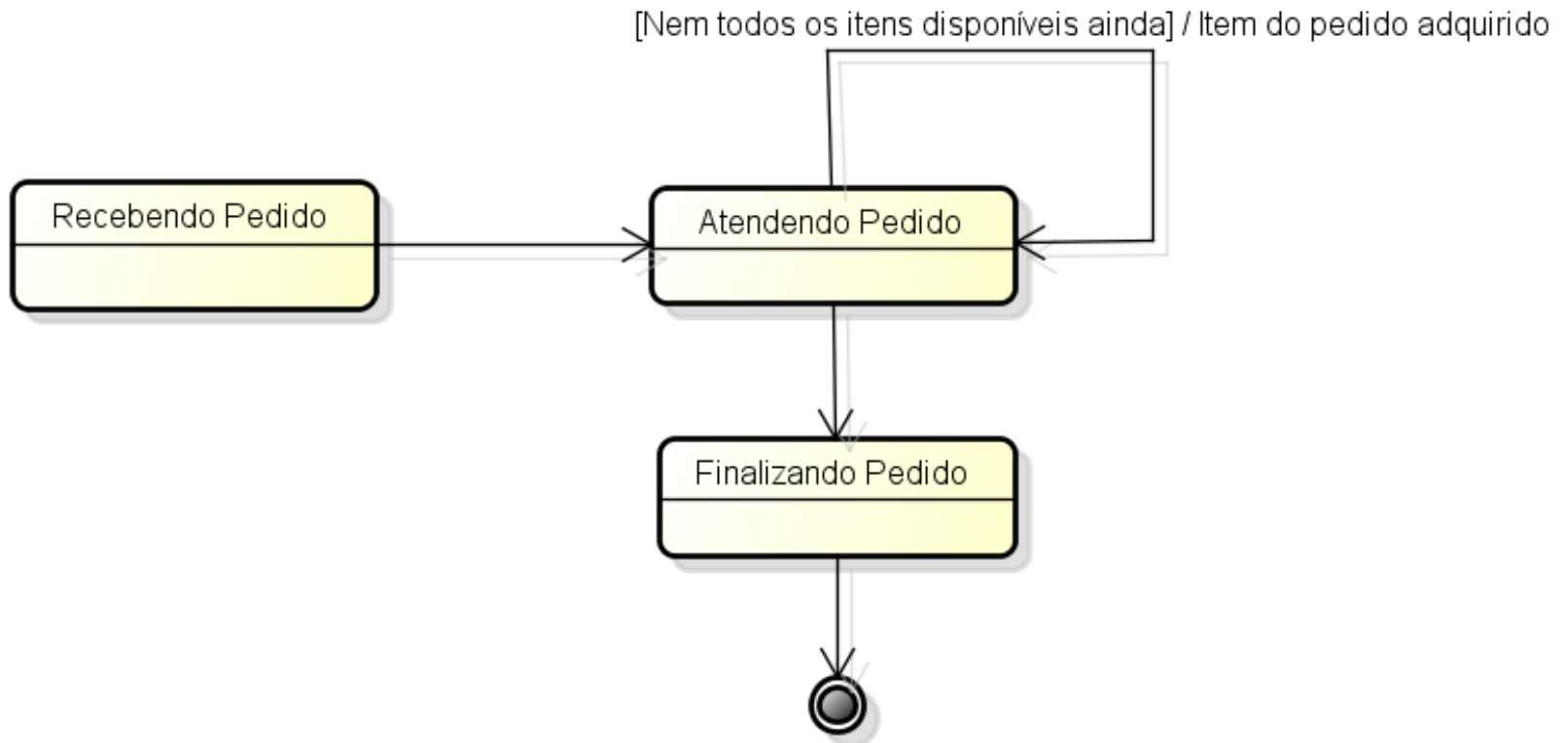
- Transições Internas não produzem modificação do estado do objeto



powered by astah*

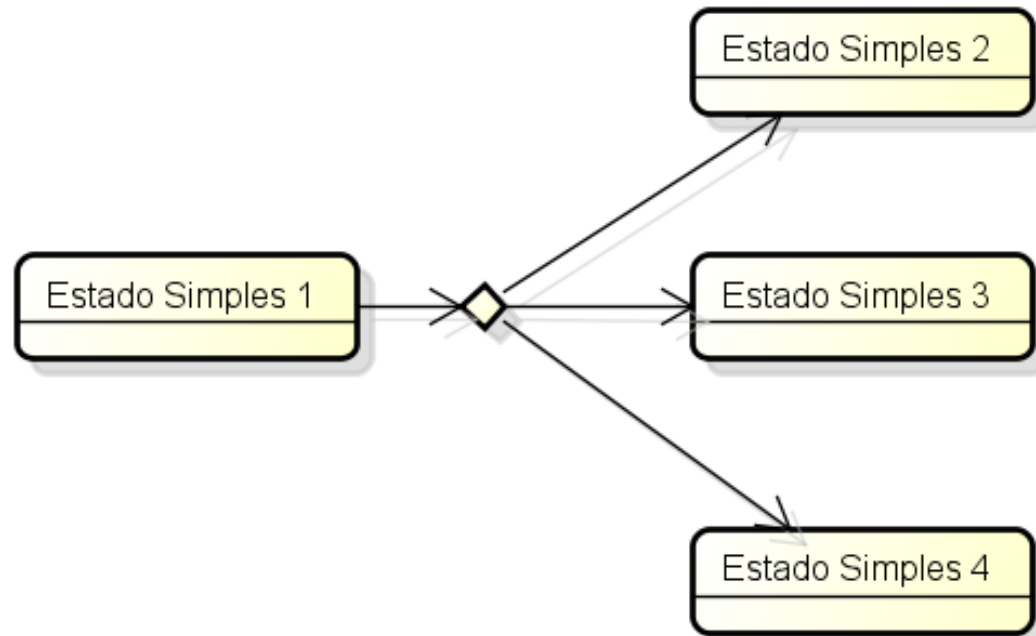
Auto-Transições

- Saem do estado atual do objeto (podendo executar alguma ação na saída) e voltam para o mesmo estado



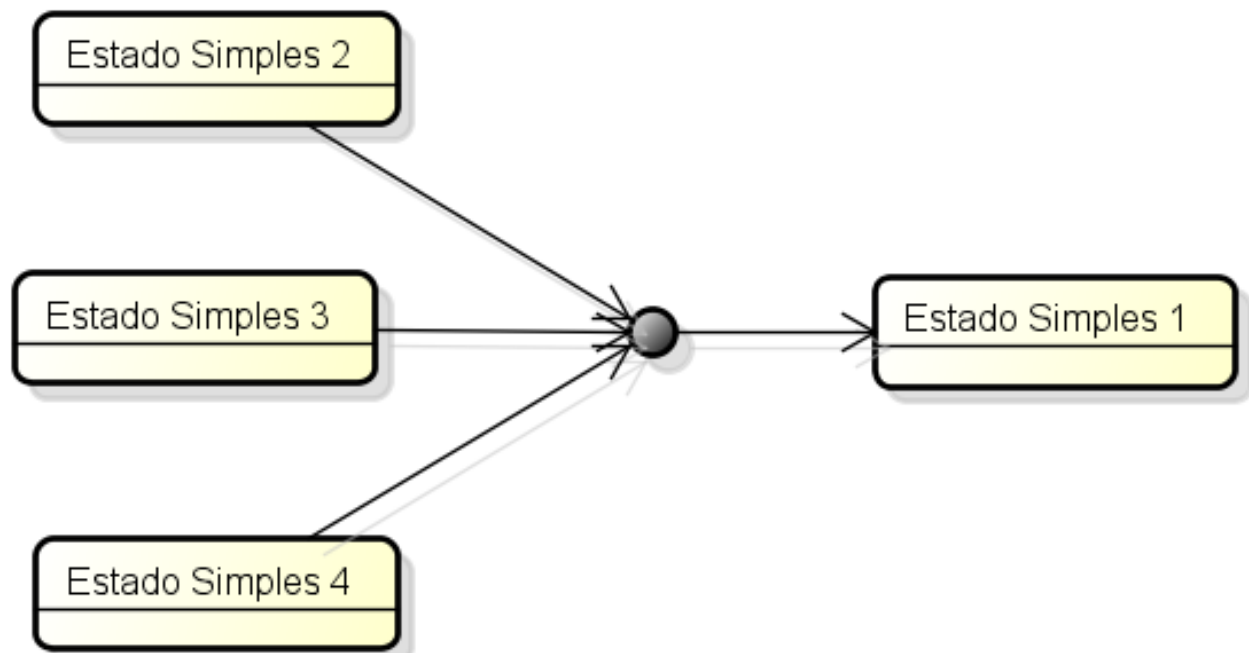
Escolha

- Um círculo simples é um estado de escolha. Ao se chegar a um estado de escolha, o software deve tomar uma decisão e ir para qualquer estado possível, dos que saem do estado de escolha.



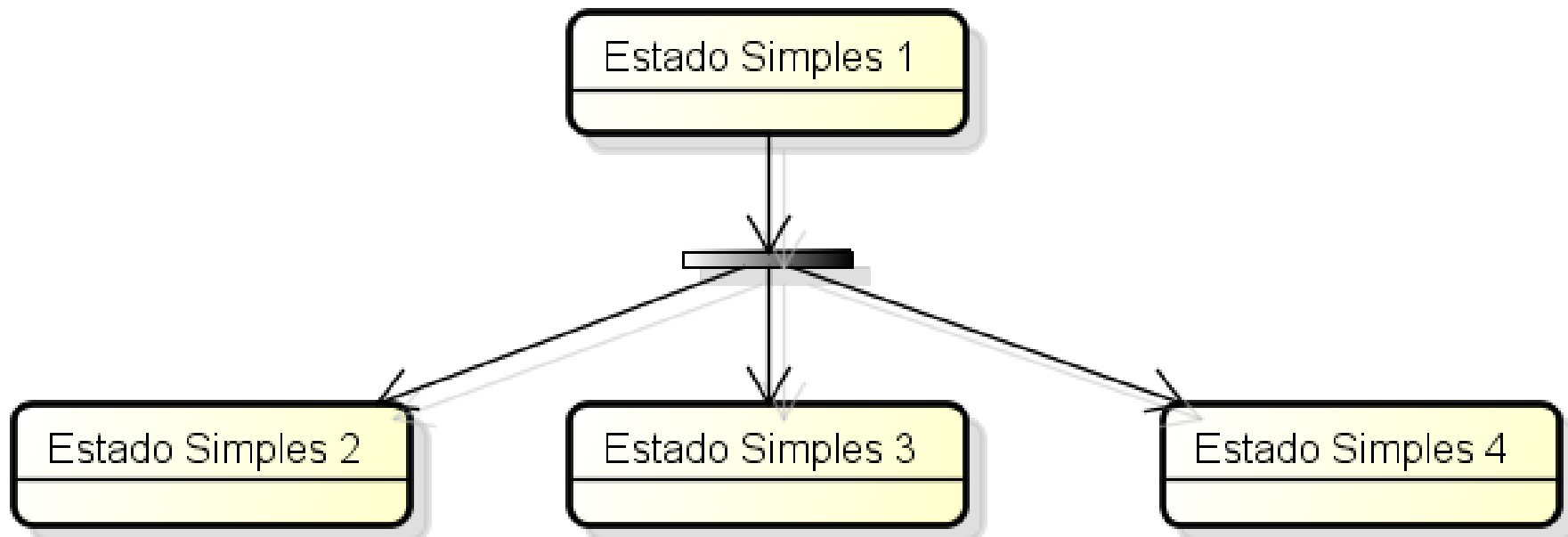
Pontos de Junção

- Podemos ter a necessidade de representar que vários estados podem chegar em um único ponto. Este ponto é conhecido como junção: um círculo preenchido.

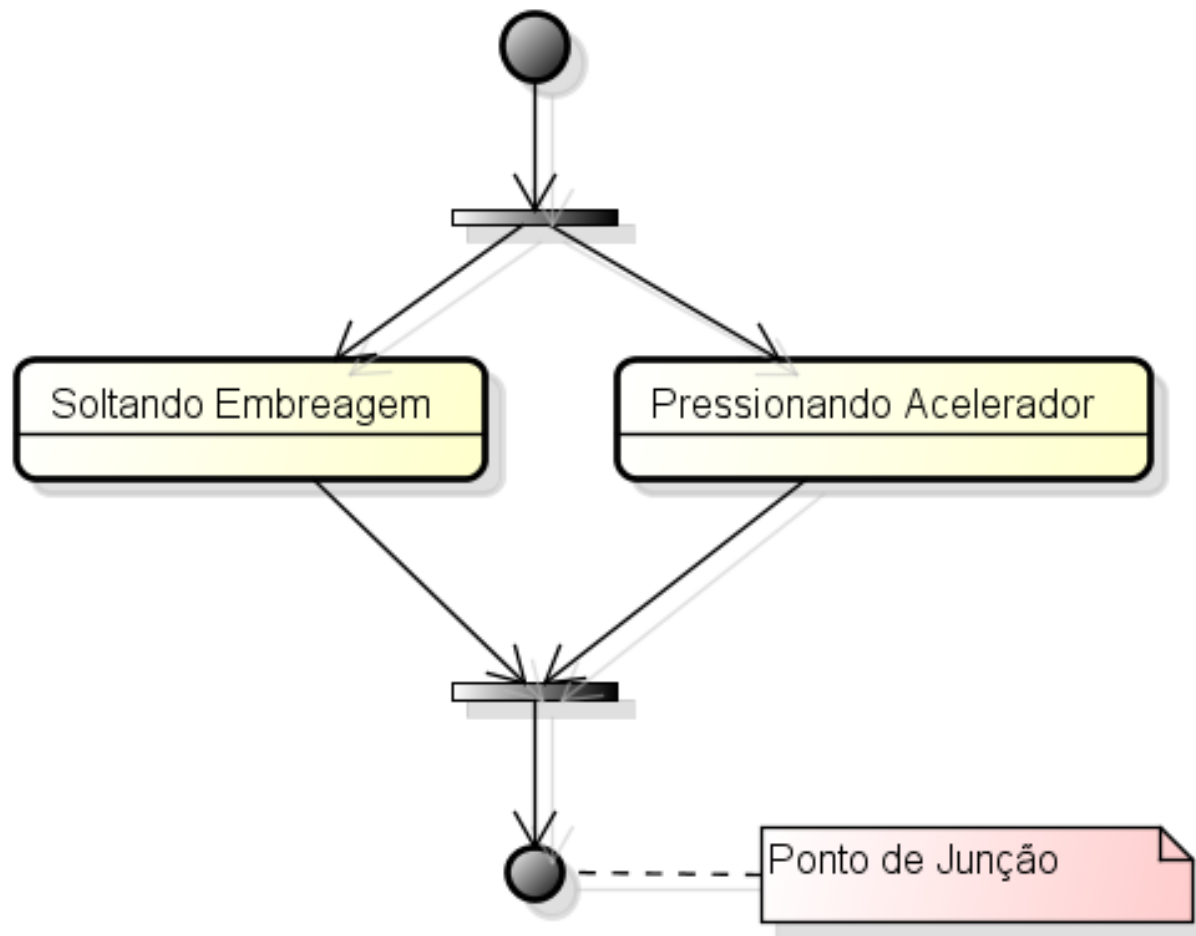


Barra de Sincronização

- A chegada de um estado a um *fork* (separa uma transição em várias transições executadas ao mesmo tempo) pode desencadear um ou vários estados concorrentes.



Barra de Sincronização

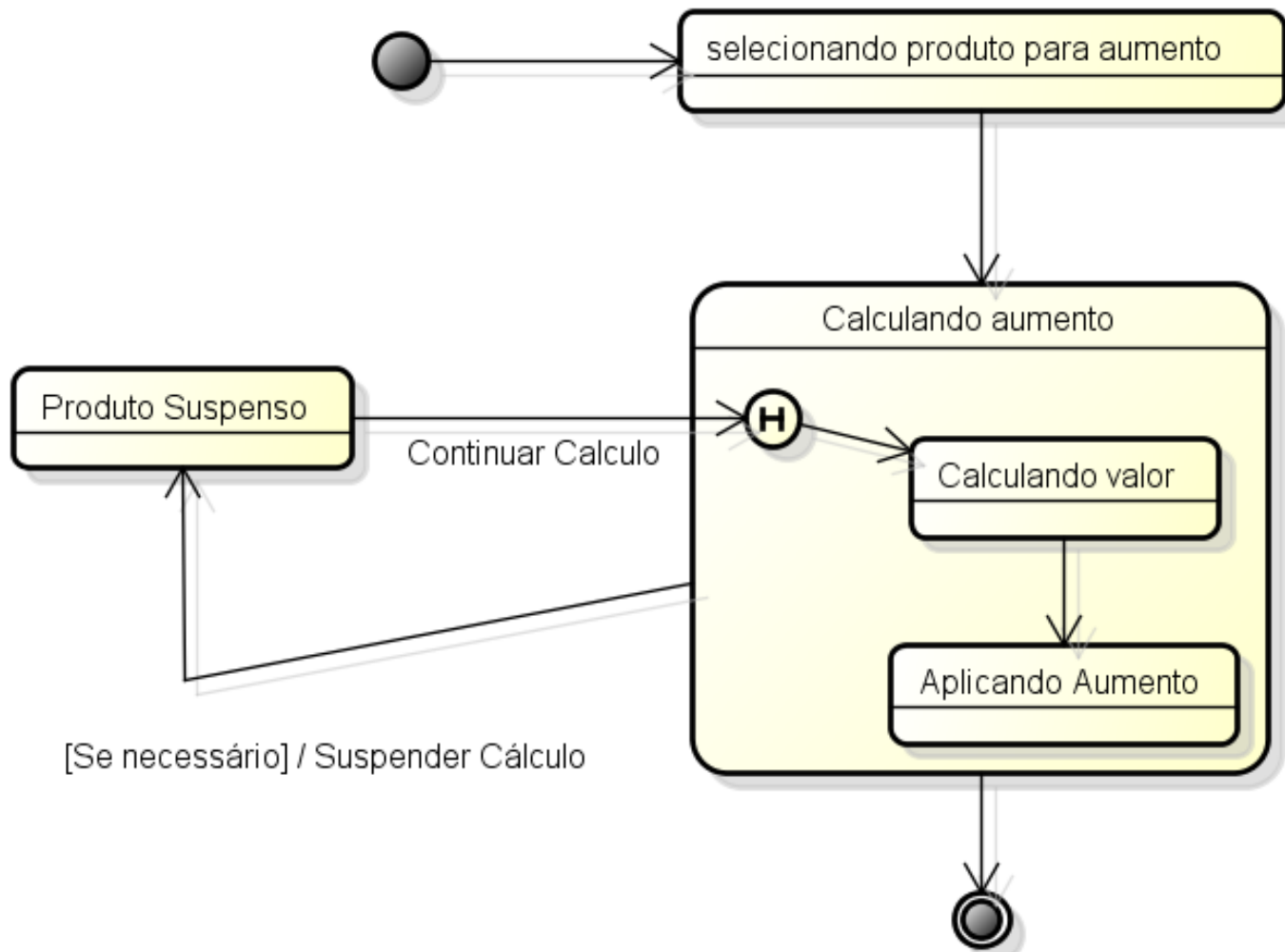




Estado de História

- Representa o registro do último subestado que um objeto se encontrava, quando por algum motivo o processo foi interrompido
- Útil quando se deseja suspender uma execução, mas sem perder o processamento prévio

Estado de História



Exemplo

- Diagrama de Estado do Objeto Venda

