

# Outros Diagramas

Prof. Marcelo Werneck  
PUC Minas Praça da Liberdade  
Engenharia de Software

# Descrição de Casos de Uso

- Casos de uso
  - Definem a interação dos atores com o sistema
    - Diagrama básico
  - Descrevem os cenários de uso:
    - Principal, Alternativos
  - Podem ser agrupados
    - Pacotes lógicos
  - Podem ser fatorados e classificados
    - Inclusão, extensão, generalização
  - Contêm relações com informações extras
    - Regras de negócio
    - Interface
    - Requisitos não-funcionais
    - ...

⇒ É o suficiente?

# Descrição de Casos de Uso

- Não é fácil obter entendimento completo através da leitura individual dos casos de uso
  - Casos de uso apresentam poucas pistas sobre sua ordem de execução
    - Inclusão, extensão
    - Pré-condições
- A ordem da especificação de requisitos não é necessariamente a melhor ordem para a leitura
  - Em sistema complexo, a quantidade de casos de uso é grande
  - Organização por pacotes pode não representar diretamente o fluxo de operação
- Solução: utilizar outro diagrama da UML para indicar a ordem relativa
  - Diagrama de atividade

# Diagrama de Atividade

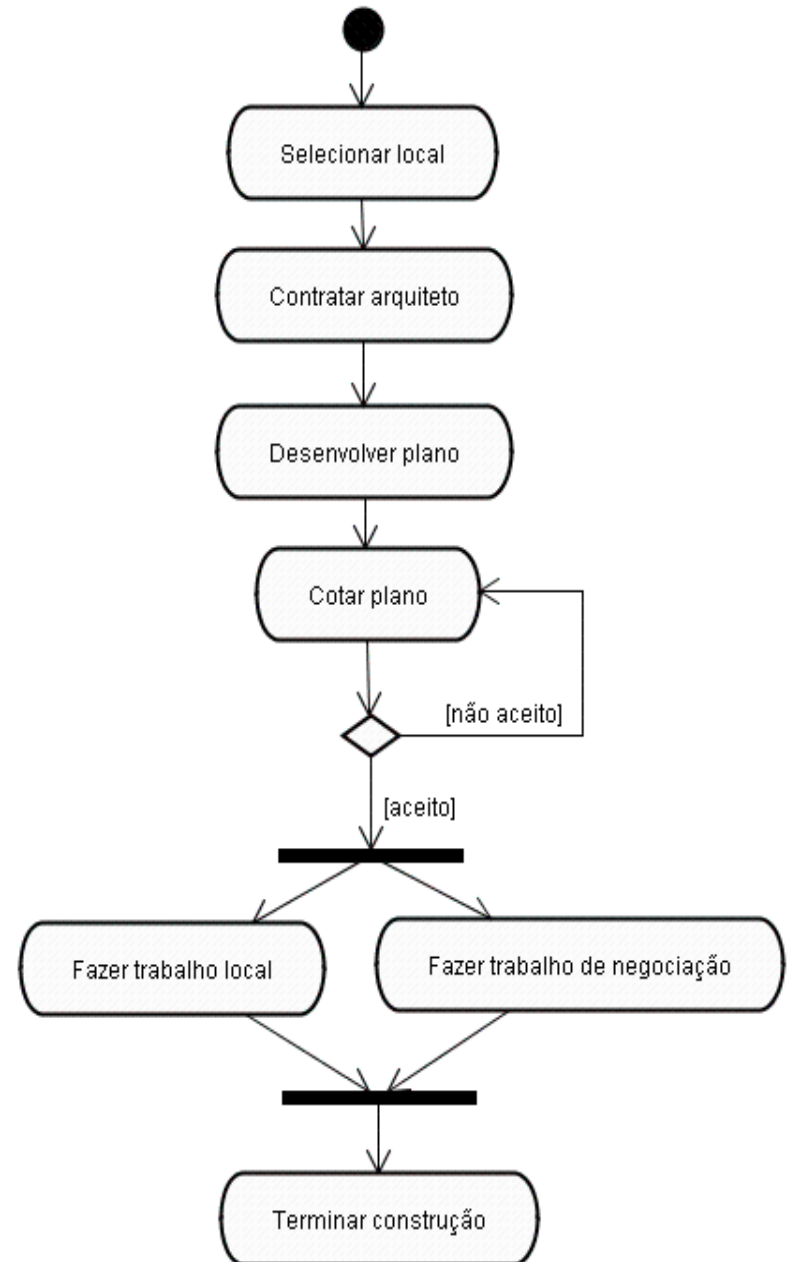
- Consiste em representação de um fluxo de controle de um processo
- Geralmente envolve representação da modelagem seqüencial ou concorrente do sistema
  - Representa atividades, decisões e branches com o intuito de descrever um processo

# Diagrama de Atividade

- Diagramas de atividade permitem modelar
  - O fluxo de trabalho de um processo
  - Informação correspondente à implementação de um método de uma classe
  - Controle entre objetos
  - A seqüência de interações de um caso de uso (fluxo de eventos do caso de uso)
  - Algoritmos de qualquer natureza

# Exemplo

- Construção de uma casa



# Diagrama de Atividade

- Pode parecer muito similar a um fluxograma
  - mostra o fluxo de atividade para atividade
- Componentes
  - Ações
  - Objetos
  - Nodos
  - Fluxo
  - Grupos de atividades
- Outros Elementos

# Ações

- Ações são primitivas de comportamento:
  - Recebem conjunto de valores
  - Alteram (opcionalmente o comportamento do sistema)
  - Retornam valores solicitados
- Ações são atômicas
  - Isto é, indivisíveis
- Exemplos:
  - Funções matemáticas
  - Processamento de dados
  - ...
- Representação básica



Calcular frete



# Nodos de controle

- Nodos de coordenam o fluxo entre outros nodos de atividade:
  - Inicial
  - Final de Atividade
  - Final de Fluxo
  - Decisão
  - Intercalação (merge)
  - Disparo simultâneo (fork)
  - Sincronização (join)

# Nodos de controle

- Inicial:
  - Ponto onde o fluxo se inicia quando a atividade é invocada
- Final de Atividade:
  - Ponto onde um ou mais fluxos de uma dada atividade se encerram.
  - Significa que todos os fluxos da atividade são encerrados
- Final de fluxo
  - Ponto onde um único fluxo é encerrado



Nodo inicial



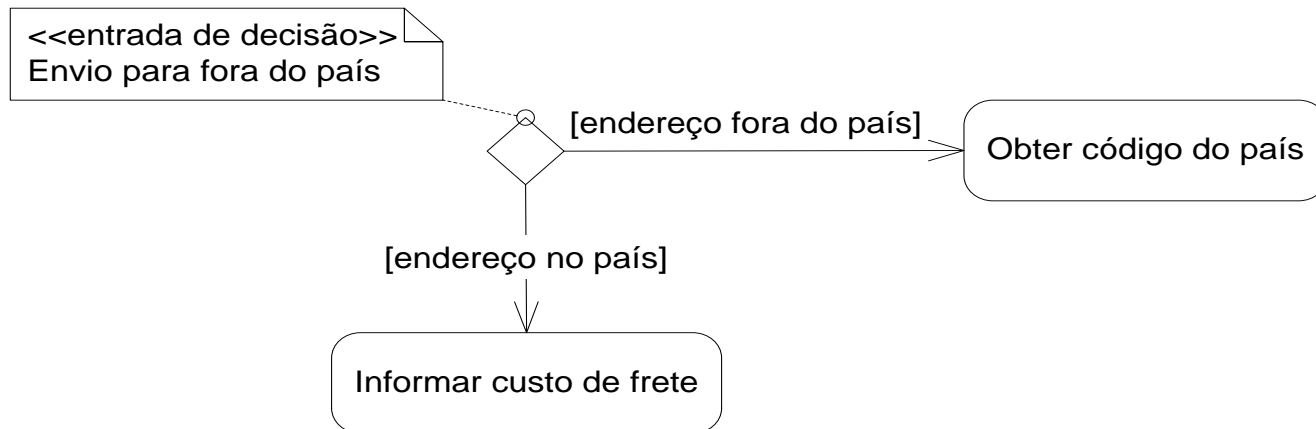
Final de atividade



Final de fluxo

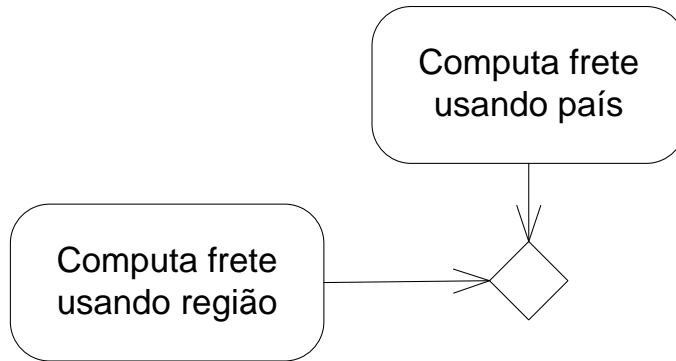
# Nodos de controle

- Decisão:
  - Oferece a escolha entre duas ou mais opções
  - Cada uma possui uma informação de guarda associada
    - Expressão que deve ser verdadeira para que a opção seja tomada
  - Estereótipo opcional



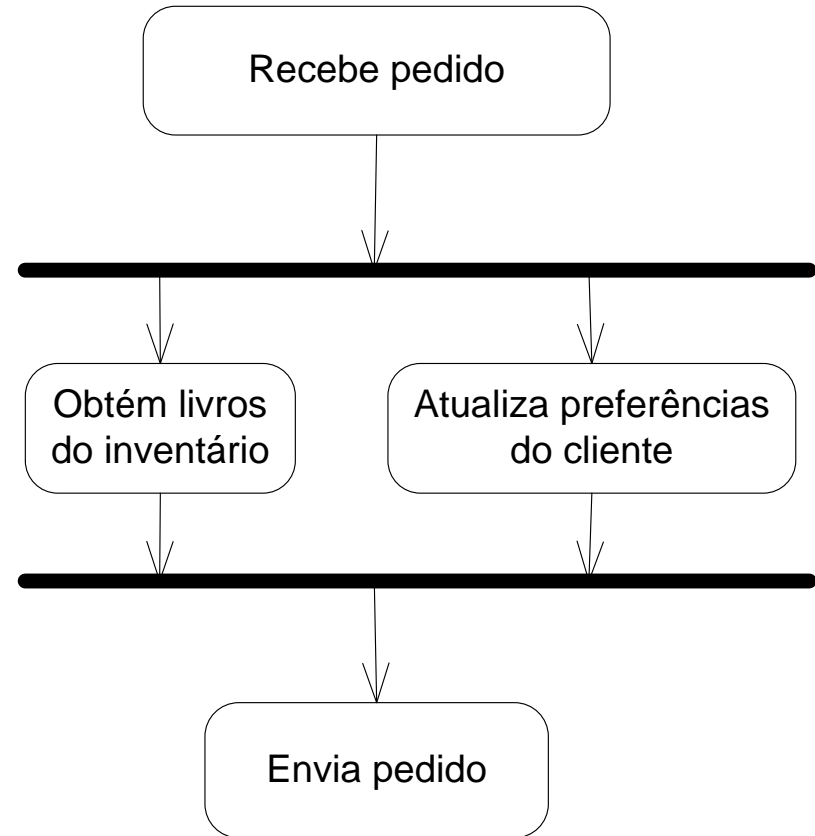
# Nodos de controle

- Intercalação (merge):
  - Reúne múltiplos fluxos de controle



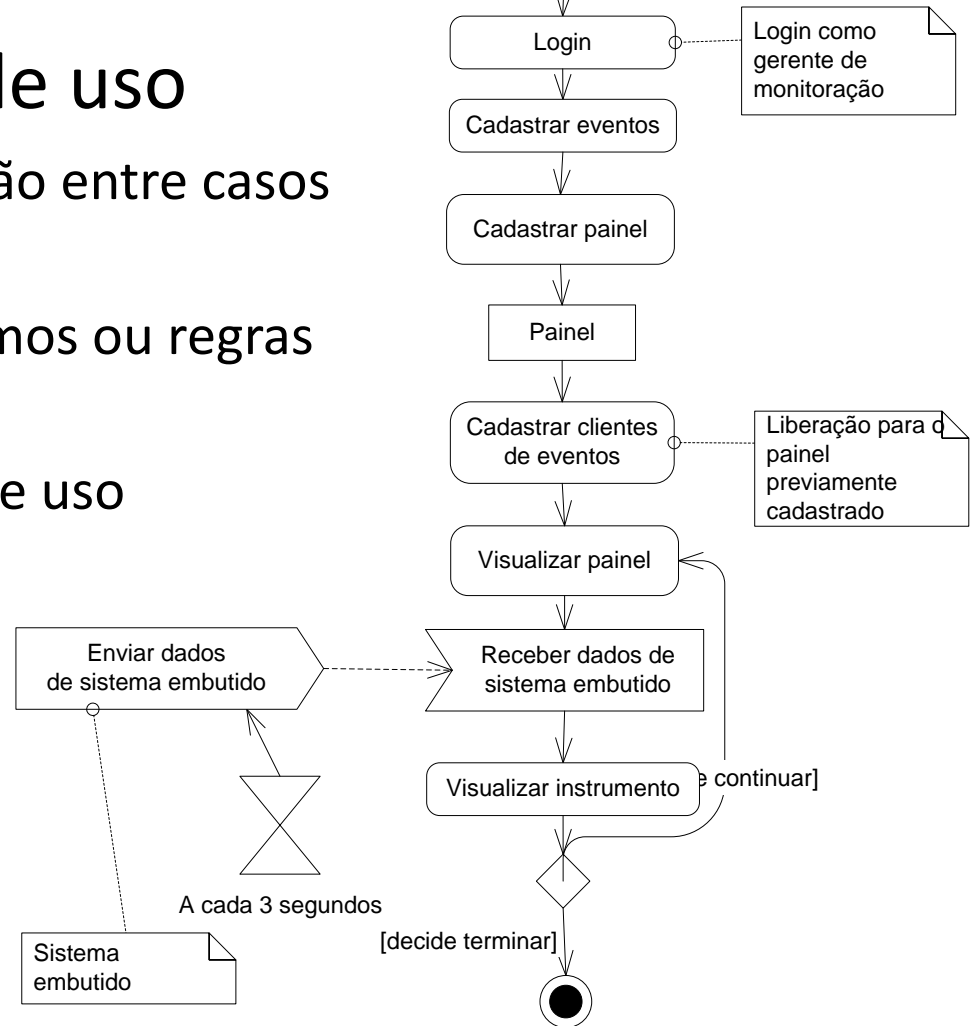
# Nodos de controle

- Disparo simultâneo (fork):
  - Divide o fluxo em múltiplos fluxos concorrentes
- Sincronização (join)
  - Sincroniza múltiplos fluxos simultâneos
  - Mesma que o disparo simultâneo



# Diagrama de atividades●

- Aplicação em casos de uso
  - Detalhamento de interação entre casos de uso
  - Detalhamento de algoritmos ou regras de negócio complexas
  - Detalhamento de casos de uso complexos



# Diagramas de Estados

- 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.

# Introdução

- 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.
- 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.



# Introdução

- Pode-se visualizar uma máquina de estado de duas formas
  - Enfatizando o fluxo de controle de atividade para atividade (diagramas de atividade)
  - Enfatizando os estados potenciais dos objetos e as transições entre estes estados (diagramas de estados)

# Definição

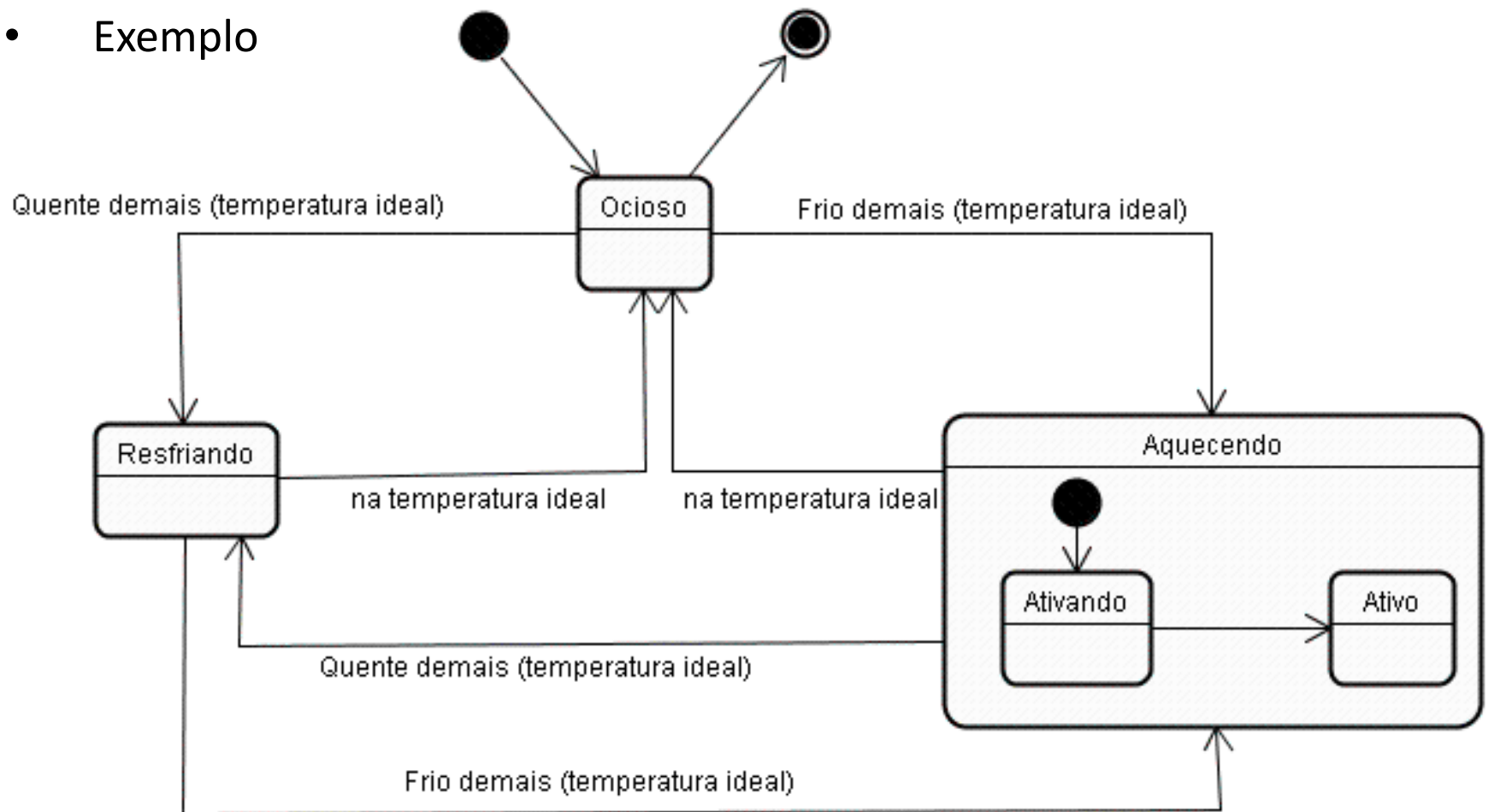
- Usa-se uma interação para modelar o comportamento de um conjunto de objetos
- Usa-se uma máquina de estados para modelar o comportamento de um objeto individual, seja ele uma instância de uma classe, um caso de uso, ou um sistema inteiro
  - Um objeto está exposto a vários eventos: invocação de operação, criação ou destruição, passar do tempo, mudança em alguma condição
  - Uma máquina de estado especifica as seqüências de estados de um objeto em resposta a eventos, junto com suas respostas a estes eventos

# Definição (cont.)

- Em resposta aos eventos, um objeto reage com alguma ação, que representa uma computação que resulta na mudança de estado do objeto ou no retorno de algum valor
- Comportamento do objeto é afetado por seu passado (histórico de estados)

# Diagramas de Estado

- Exemplo



# Estado

- Componentes
  - Estados
  - Transições

# Estado

- Situação na vida de um objeto em que ele satisfaz a alguma condição ou realiza alguma atividade.
  - 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%.
    - Pedido está *atendido* quando todos os itens estão atendidos
- Pode ser visto como abstração dos atributos e associações de um objeto.

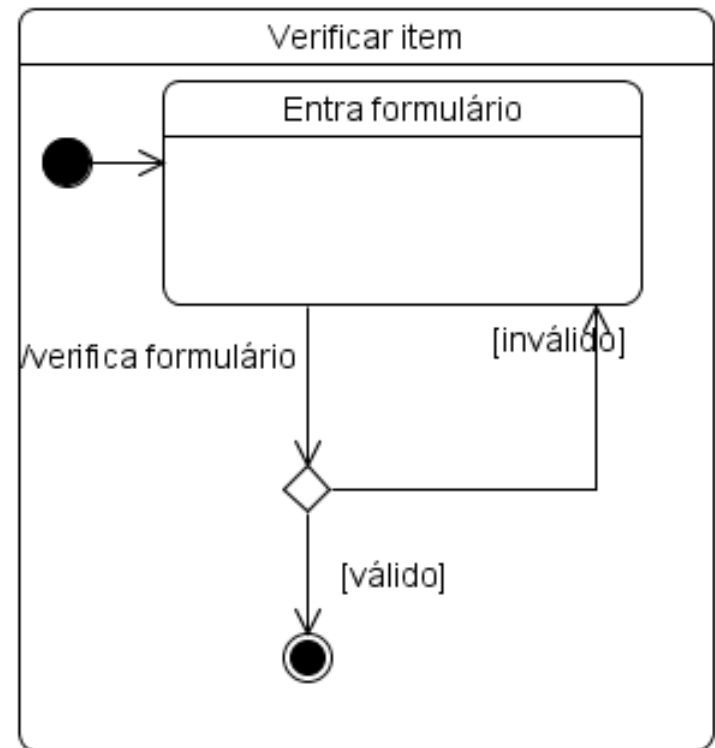
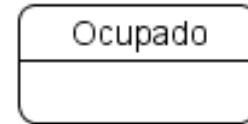
# Estado

- Estado inicial
  - Indica o estado de um objeto quando ele é criado.
  - Só pode haver um estado inicial em um contexto ou região associada
- Estado final
  - Indica o fim do ciclo de vida de um objeto.
  - Opcional
  - Pode haver mais de um estado final em um contexto ou região associada.



# Estado

- Estado simples
  - Estado de uso geral
  - Não associado a outras regiões ou outros estados
- Estado composto
  - Apresenta estado com submáquina visível
  - Obs
    - Estados inicial e final são opcionais
    - Conexões externas podem ser feitas diretamente





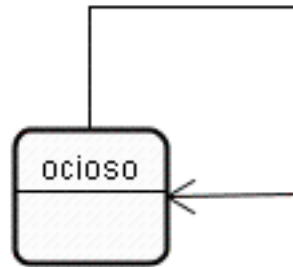
# Transições

- Os estados estão associados a outros pelas transições.
- Uma transição é mostrada como uma linha conectando estados, com uma seta apontando para um dos estados.
- Quando uma transição entre estados ocorre, diz-se que a transição foi disparada.
- Uma transição pode ter os mesmos estados de origem e destino

# Transições

- Uma transição pode ser rotulada com uma expressão da seguinte forma:

após (1 hora) [existe papel na bandeja] / imprimir auto teste



evento (lista-parâmetros) [guarda] / atividade