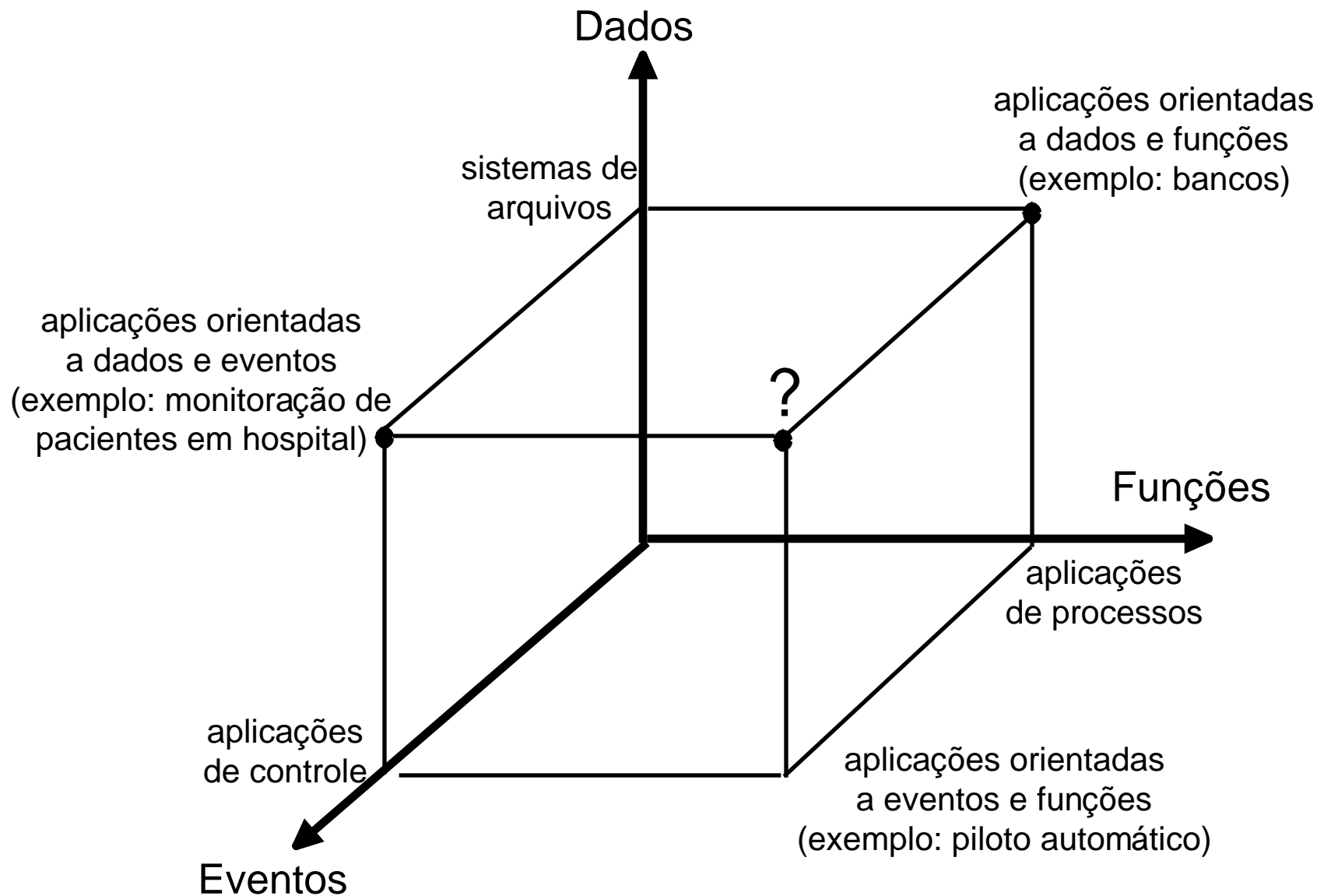


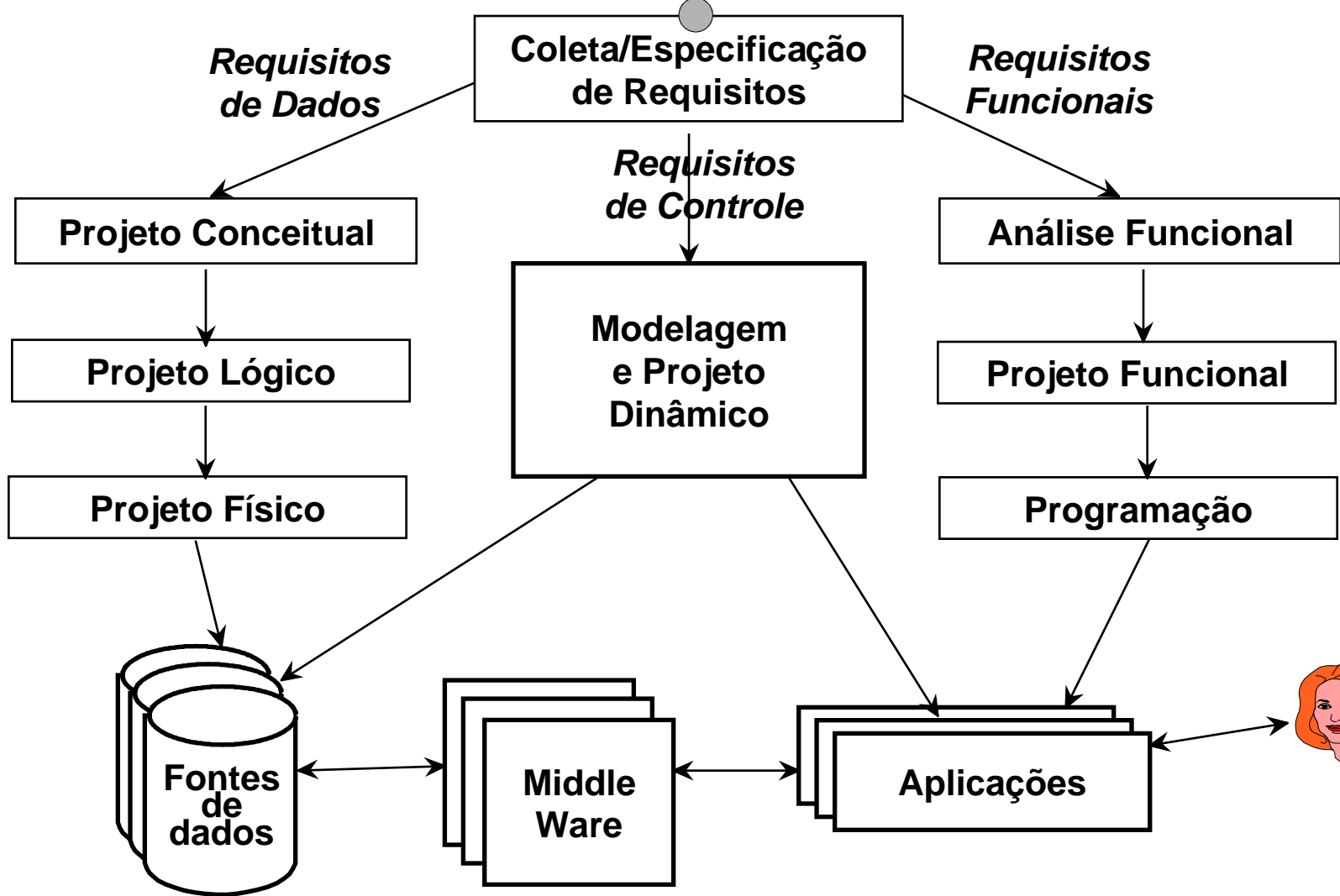
# Modelagem de Sistemas

- Modelagem Funcional
  - DFDs, Use Cases
- Modelagem de Dados
  - Modelo Entidades-Relacionamentos; Modelo de Objetos
- Modelagem Dinâmica
  - Diagramas de Estado

# Visão Tridimensional dos Sistemas de Informação



Mundo Real



# Modelagem Dinâmica

- Modelo Estático
  - Objetos (entidades) e seus relacionamentos num dado momento.
- Modelo Dinâmico
  - Mudanças dos objetos com o tempo.
  - Controle: descrição da seqüência de operações (ações) que ocorrem, sob certas condições, em resposta a estímulos externos (eventos), mudando os estados de objetos.

# Eventos

- Um evento é um fato que ocorre num ponto do tempo, e não tem duração. Pode ocorrer em consequência de:
  - uma ação externa (ex: Usuário aperta o botão direito);
  - uma ação interna (ex: O motor parou)
  - tempo (ex: Dia 31/05/2001 às 23:00 horas)
- Um evento pode ser representado pela ação que o causa ou pelo resultado da execução da ação (isto é, o evento ocorrido)
- Eventos podem ter relacionamentos causais. Quando não são relacionados, são chamados concorrentes, isto é, podem ocorrer sem qualquer influência uns aos outros.
- Eventos podem ser simplesmente sinais de que um fato ocorreu, ou podem carregar atributos.
  - Ex: O avião decolou (VARIG, voo 123, Rio de Janeiro)

# Estados

- Um estado é uma abstração dos valores de atributos ou ligações de um objeto. Conjuntos de valores são agrupados em um estado de acordo com as propriedades que afetam o comportamento do objeto.
  - Ex: o estado de um banco pode ser solvente ou insolvente, dependendo do valor do seu patrimônio ser maior do que suas dívidas
- Um estado especifica a resposta do objeto a um evento de entrada. Corresponde a um intervalo entre dois eventos recebidos por um objeto. Portanto, estados têm duração.
- Estados podem ser
  - associados a condições (ex: o estado “líquido” para a água está associado à condição “temperatura maior do que 0 e menor do que 100 °C”);
  - valores enumerados (ex: uma transmissão de automóvel pode estar nos estados “Ré”, “Neutro”, “Primeira”, “Segunda”, “Terceira”, “Quarta”).

# Eventos e Estados

- Eventos e estados são duais um do outro; um evento separa dois estados, e um estado separa dois eventos.
- Tanto eventos quanto estados dependem do nível de abstração usado na modelagem.
  - Por exemplo, um agente de viagem pode tratar cada segmento de um roteiro como um evento; um painel de vôos de um aeroporto distinguiria entre chegadas e partidas; um sistema de controle de tráfico aéreo desdobraria cada vôo em muitos trechos geográficos.
- Na definição de estados, são ignorados os atributos que não afetam o comportamento do objeto, e são agrupados num único estado todas as combinações de valores de atributos e ligações que tenham a mesma resposta a eventos.

# Diferentes caracterizações de um Estado

**Estado:** *Alarme tocando*

**Descrição:** O alarme do relógio está tocando para indicar um tempo alvo pré-estabelecido.

**Seqüência de eventos que produz o estado:**

*estabelecer alarme* (tempo alvo)

qualquer seqüência não incluindo *apaga alarme*

tempo atual = tempo alvo

**Condição que caracteriza o estado**

alarme = ligado **AND** tempo alvo    tempo atual    tempo alvo + 20 segundos

**AND** nenhum botão foi pressionado desde o tempo alvo

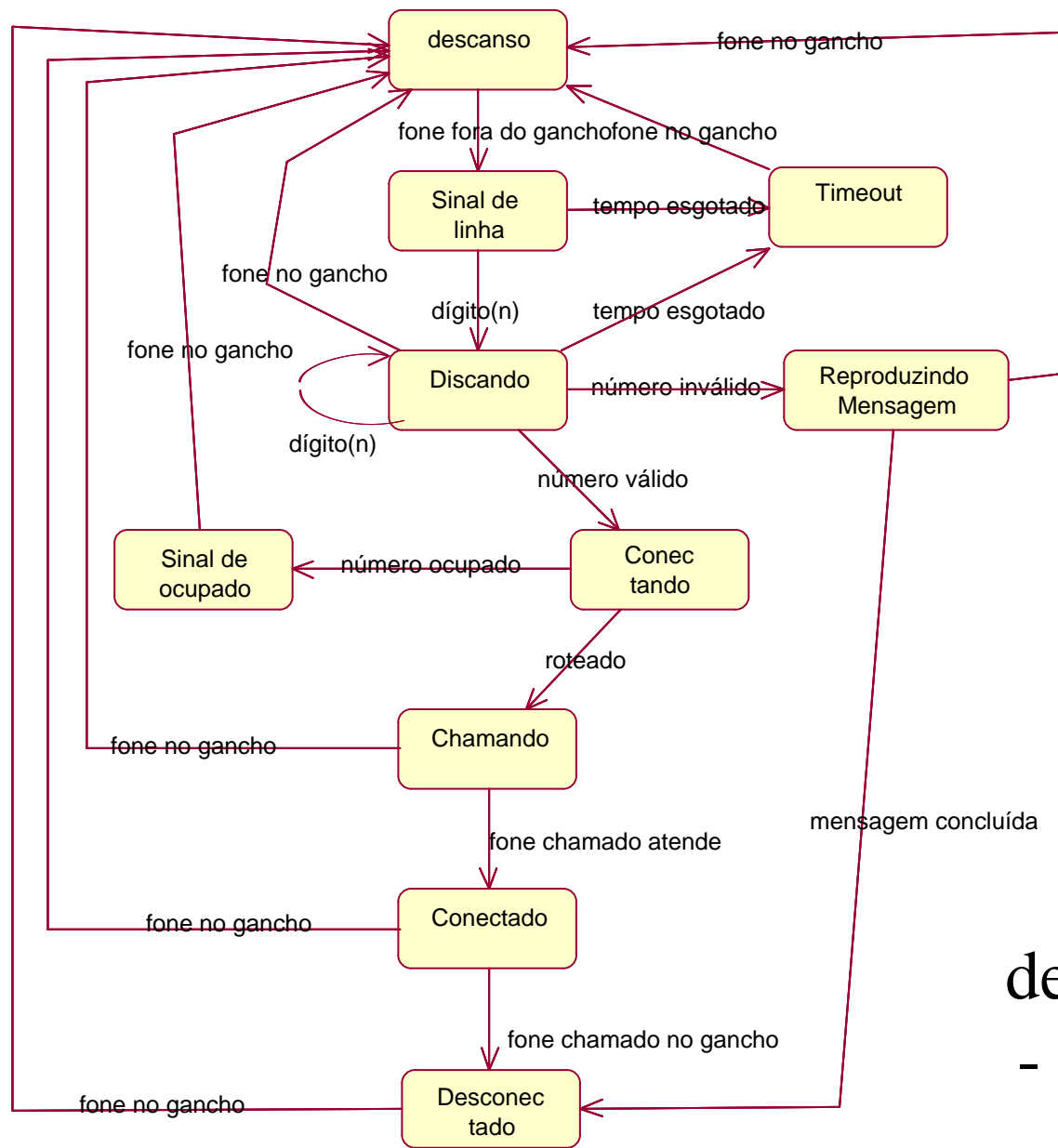
**Eventos aceitos no estado**

<b>evento</b>	<b>ação</b>	<b>próximo estado</b>
tempo atual = tempo alvo + 20''	desligar alarme	normal
botão pressionado (qualquer botão)	desligar alarme	normal



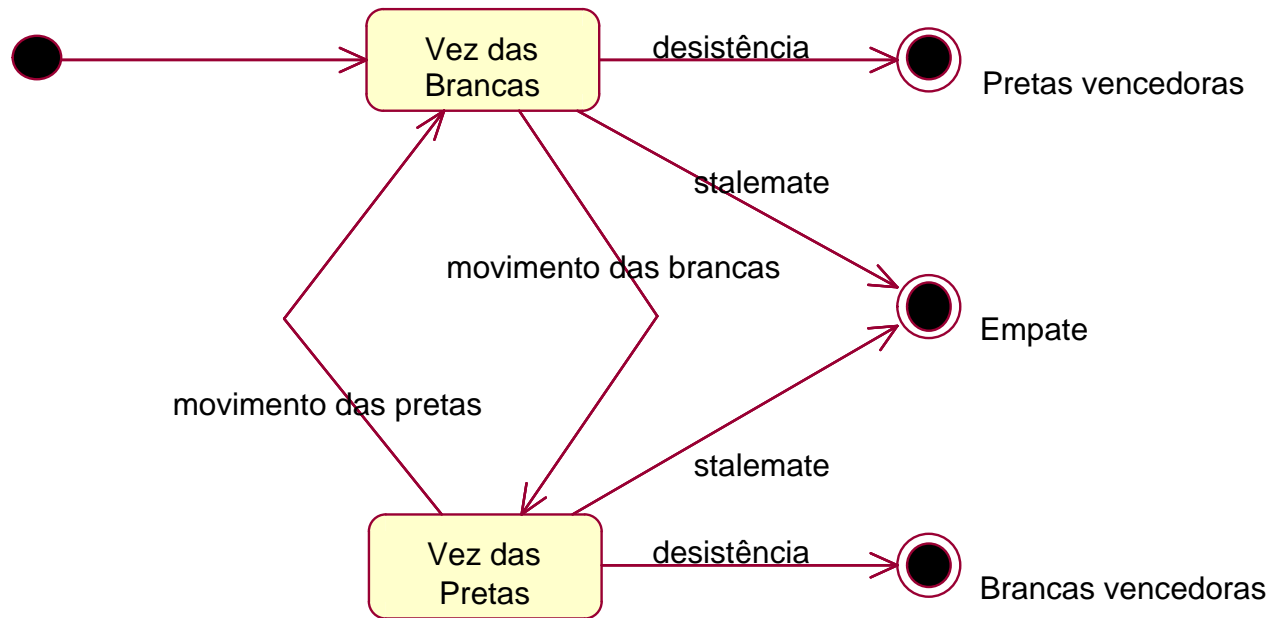
# Diagramas de Estado

- Um diagrama de estado descreve o comportamento de uma classe de objetos (conjunto de entidades), associando eventos e estados. Quando um evento ocorre, o próximo estado depende do estado atual bem como do evento ocorrido; a mudança de estado causada por um evento é chamada de *transição*.
- Um diagrama de estados é um grafo cujos nós são estados e cujos arcos orientados são transições rotuladas por nomes de eventos. Uma seqüência de eventos é um *caminho* dentro do grafo.
- Se mais de uma transição saem de um mesmo estado, então o primeiro evento que ocorrer causará a correspondente transição. Quando ocorre um evento que não tem transição saindo do estado atual, então o evento é ignorado.
- Um modelo dinâmico é uma coleção de diagramas de estado que interagem uns com outros via eventos compartilhados.



# Diagrama de Estados de “loop” contínuo - Linha Telefônica

# Diagrama de Estados “one-shot” (objetos com vidas finitas) - Jogo de Xadrez



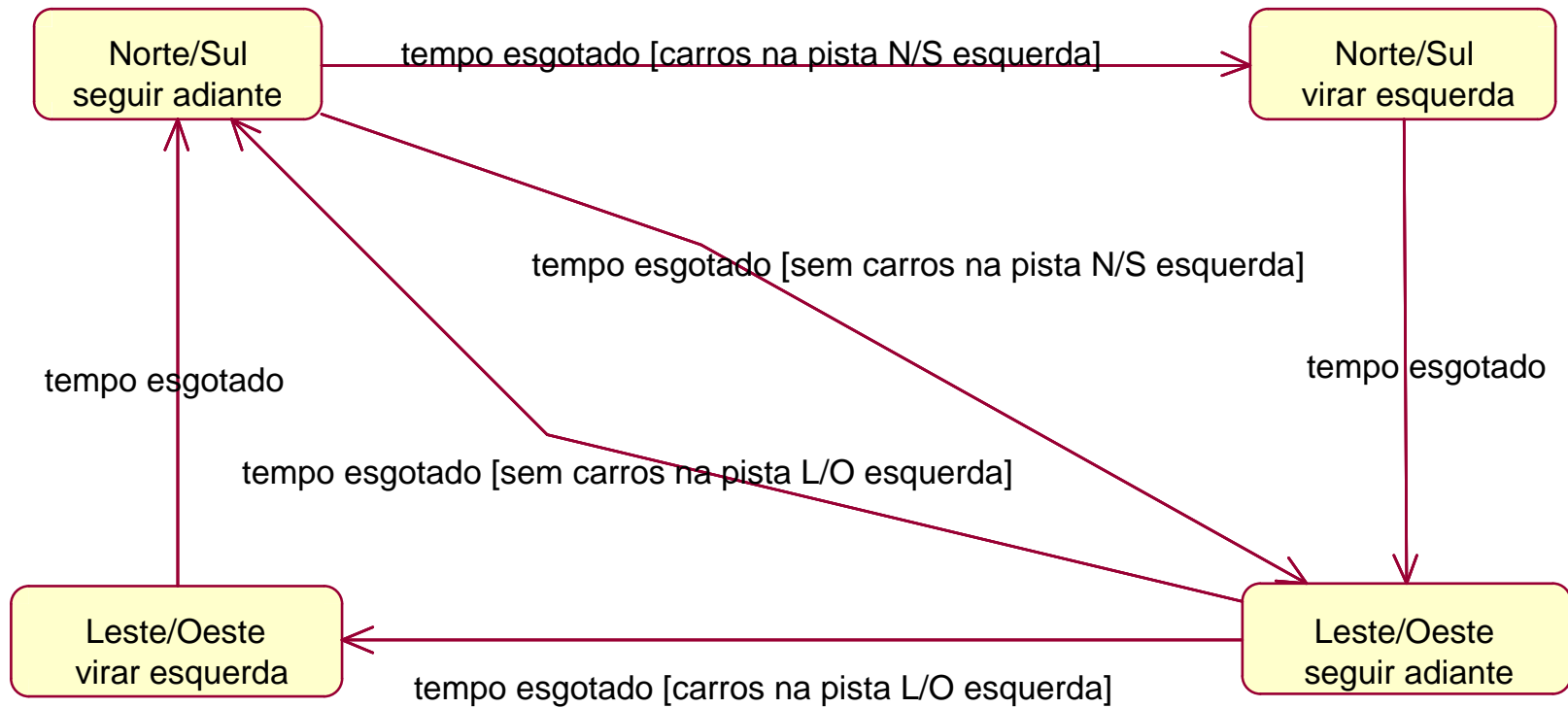
● Estado inicial

⦿ Estado final

# Condições

- Uma condição é uma função booleana de valores de objetos (ex: “temperatura abaixo de zero”). É válida dentro de um intervalo de tempo (ex: “a temperatura estava abaixo de zero entre 2:00 e 5:00 horas”).
- É importante distinguir condições de eventos; estes não têm duração. Um estado pode ser definido em termos de uma condição; inversamente, estar num estado é uma condição.
- Condições podem ser usadas como “guardas” em transições. Uma transição guardada acontece quando seu evento ocorre, mas somente quando a condição de guarda é verdadeira. Uma condição de guarda é especificada como uma expressão booleana entre colchetes seguindo o nome do evento na transição.

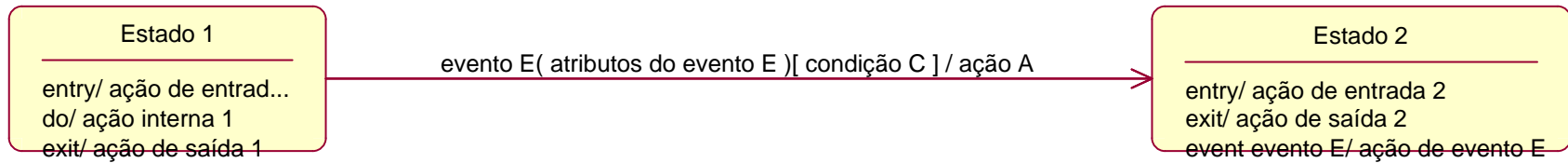
# Diagrama de Estados com transições “guardadas” por condições - Semáforo com entrada à esquerda



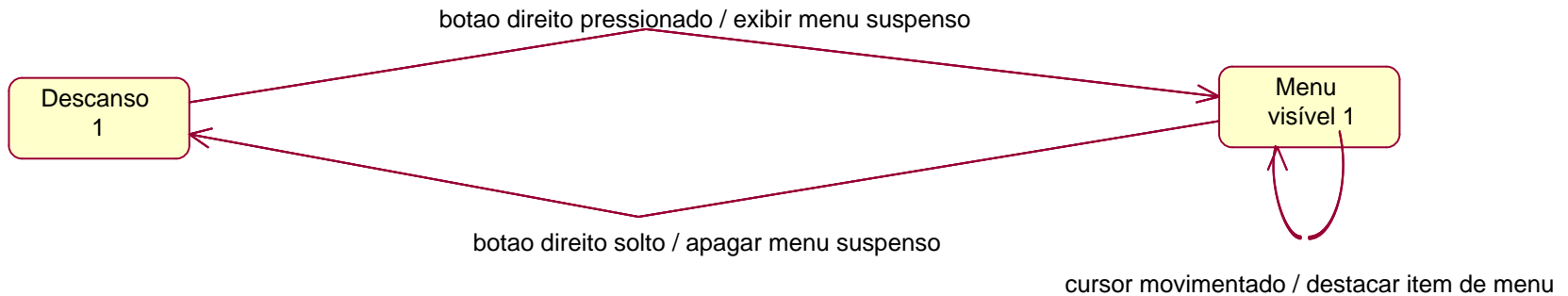
# Ações (do sistema)

- Diagramas de estado seriam pouco úteis se apenas descrevessem padrões de eventos. Uma descrição comportamental de um objeto deve especificar O QUE o objeto faz em resposta a eventos. Ações anexadas a estados ou transições são executadas em resposta aos correspondentes estados ou eventos.
- Ações em estados podem ocorrer em 4 situações:
  - “on entry” : a operação executada quando o objeto entra no estado;
  - “on exit”: operação executada quando o objeto sai do estado;
  - “do”: operação executada enquanto o objeto está no estado que continua até que o objeto saia do estado;
  - “on event”: operação disparada quando um evento específico ocorre.
- Uma ação em uma transição é especificada junto com o evento, após uma possível condição, separada por “/”.
  - evento (argumentos) [condição] / ação

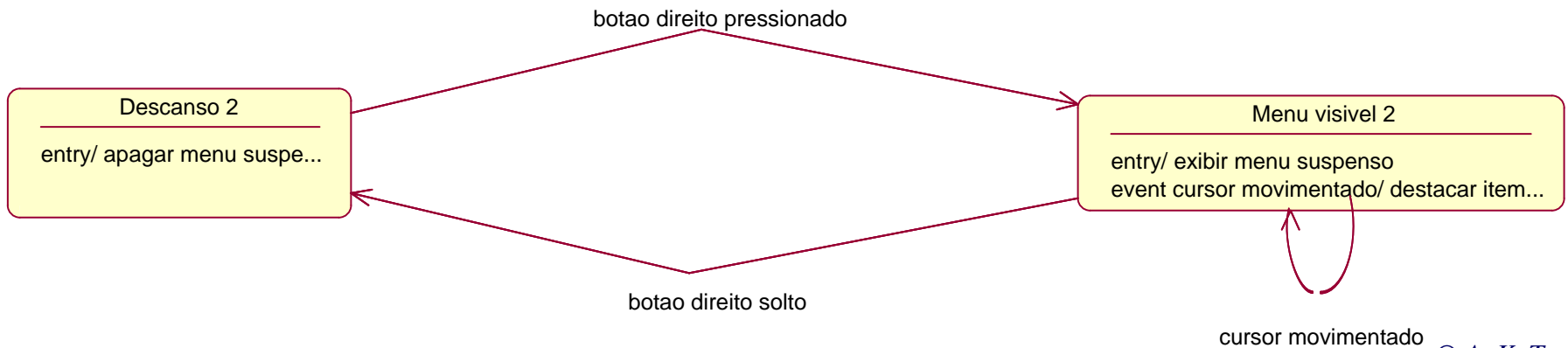
# Ações em Estados e Transições



## Sumário da notação de diagramas de estado



## Duas formas equivalentes de modelar ações em um menu suspenso (em transições ou em estados)

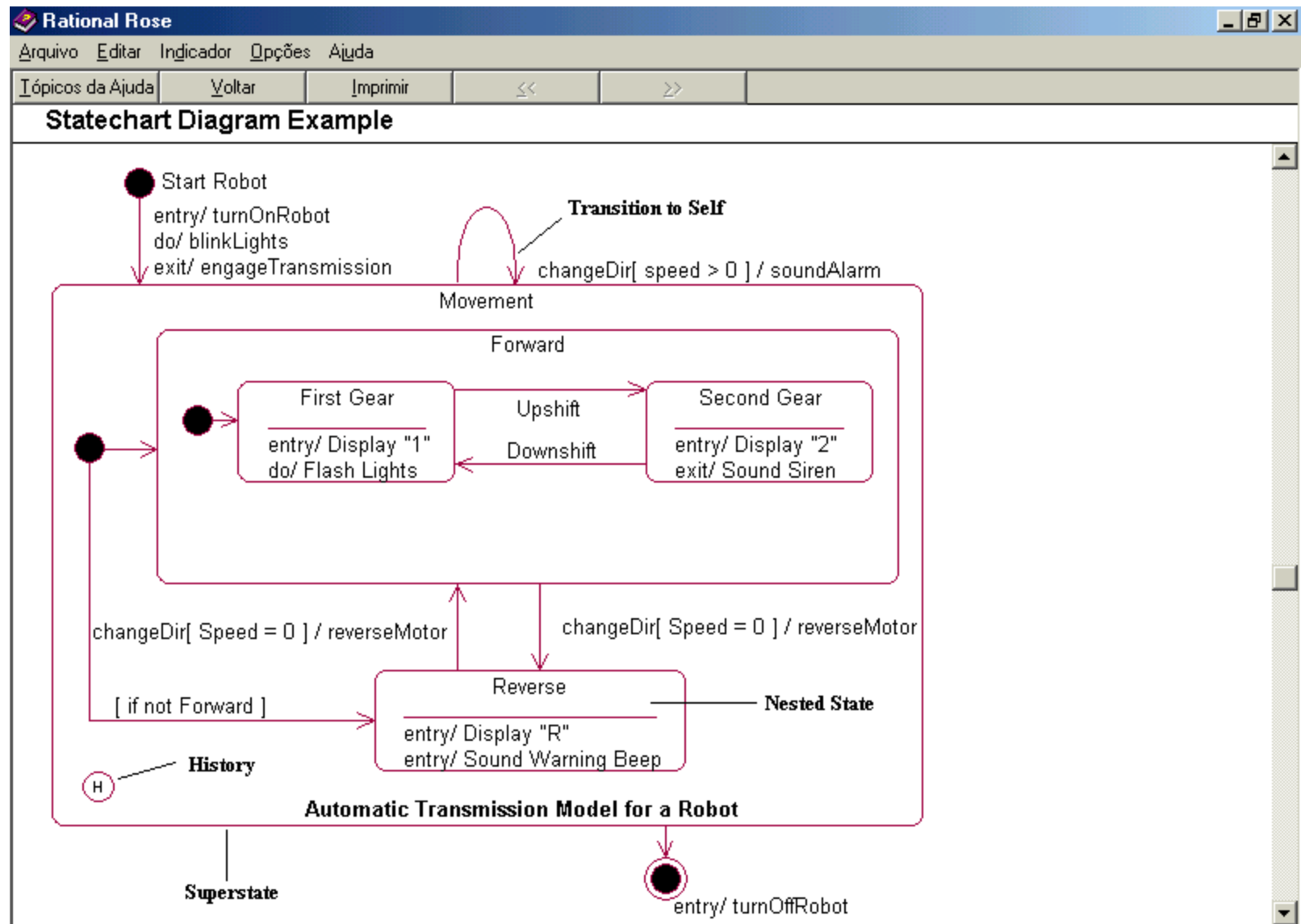


# Hierarquia em Diagramas de Estado

- Diagramas de estado podem ser estruturados hierarquicamente para permitir descrições concisas de sistemas complexos, de forma semelhante aos objetos no modelo estático.
- A generalização é equivalente ao aninhamento de estados. O estado mais genérico é chamado super-estado e os estados aninhados são sub-estados, isto é, refinamentos do super-estado.
- Sub-estados herdam as ações e transições de seus super-estados, da mesma forma que sub-classes herdam atributos, operações e relacionamentos de suas super-classes.
- A representação de super-estados e sub-estados aninhados em contornos (sub-estados dentro de super-estados), é chamada de “statecharts”(cartas de estado).



# Modelo Dinâmico em UML - Diagramas de Statecharts



# Dicas práticas para Modelagem Dinâmica

- Construir diagramas de estado somente para classes de objeto com comportamento dinâmico significativo. Nem todas as classes de objetos requerem um modelo dinâmico.
- Verificar os vários diagramas de estado quanto à consistência dos eventos compartilhados de modo que o modelo dinâmico completo seja correto.
- Usar cenários (situações ou casos de uso) para ajudar no processo de construir diagramas de estado.
- Considerar apenas os atributos relevantes do objeto quando definir um estado. Nem todos os atributos representados num modelo de objetos necessitam ser usados em diagramas de estado.
- Considerar as necessidades da aplicação quando decidir sobre a granularidade de eventos e estados.
- Quando um estado tem múltiplas transições de entrada e todas as transições causam a execução da mesma ação, ponha as ações dentro da caixa do estado sob a forma de ação “on entry”. O mesmo se aplica a transições de saída de um estado.
- Use estados aninhados quando a mesma transição se aplica a muitos estados.
- Procure fazer diagramas de estado para subclasses independentes de diagramas de estado para suas superclasses. Os diagramas de estado de subclasses deveriam se concentrar em atributos específicos das subclasses.
- Cuidado com condições indesejáveis de disputa em diagramas de estado. Condições de disputa podem ocorrer quando um estado pode aceitar eventos de mais de um objeto.

# Exercício: Portão automático de garagem

- O motorista gera eventos “botão pressionado”, com o controle remoto, para abrir e para fechar o portão vertical. Cada evento inverte o sentido do portão mas, por segurança, o portão deve abrir completamente antes que possa ser fechado. O controle gera ações “acionar motor acima” e “acionar motor abaixo” para o motor. Os eventos “portão aberto” e “portão fechado” são gerados quando os movimentos são completados.

# Exercício

## Relógio Digital de 2 botões

- Um relógio digital tem um display e dois botões, A e B. O relógio tem dois modos de operação: mostrar tempo e acertar tempo. No modo mostrar, as horas e minutos são mostradas separados por um sinal de dois pontos piscando. O modo acertar tempo tem dois submodos, acertar horas e acertar minutos. O botão A é usado para selecionar os modos. Cada vez que é pressionado, o modo avança na seqüência mostrar, acertar horas, acertar minutos, mostrar, etc. Dentro dos submodos, o botão B é usado para avançar as horas ou minutos um de cada vez que é pressionado. Os botões devem ser liberados antes que possam gerar outro evento.