

Metodologia de modelagem

Etapa 4

Prof. Murillo G. Carneiro
FACOM/UFU

Objetivo

- Apresentar a quarta etapa da metodologia de modelagem
 - Etapa 4 – Modelagem de estados associada a classe

Diagrama UML usado

- Diagrama de máquina de estados

Etapa 4 – Modelagem de estados associada a classe

- Descrever a existência de um objeto desde a criação até a destruição
 - Todas as situações em que um objeto pode se encontrar ao longo de sua existência
- Modelagem dinâmica focada em uma única classe

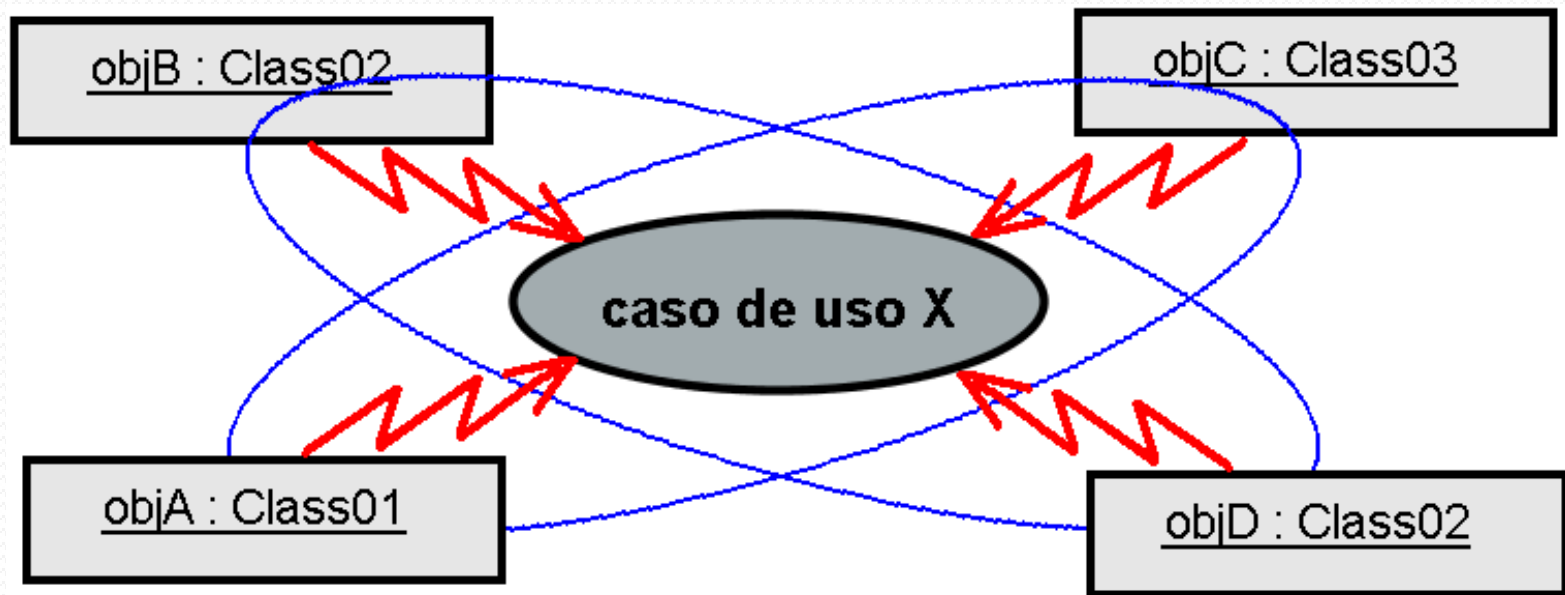
Etapa 4 – Modelagem de estados associada a classe

- Finalidade 1 → A avaliação dos atributos até então identificados e identificação de novos
 - Secundariamente, identificação de métodos
- Finalidade 2 → Boa estruturação de algoritmos complexos

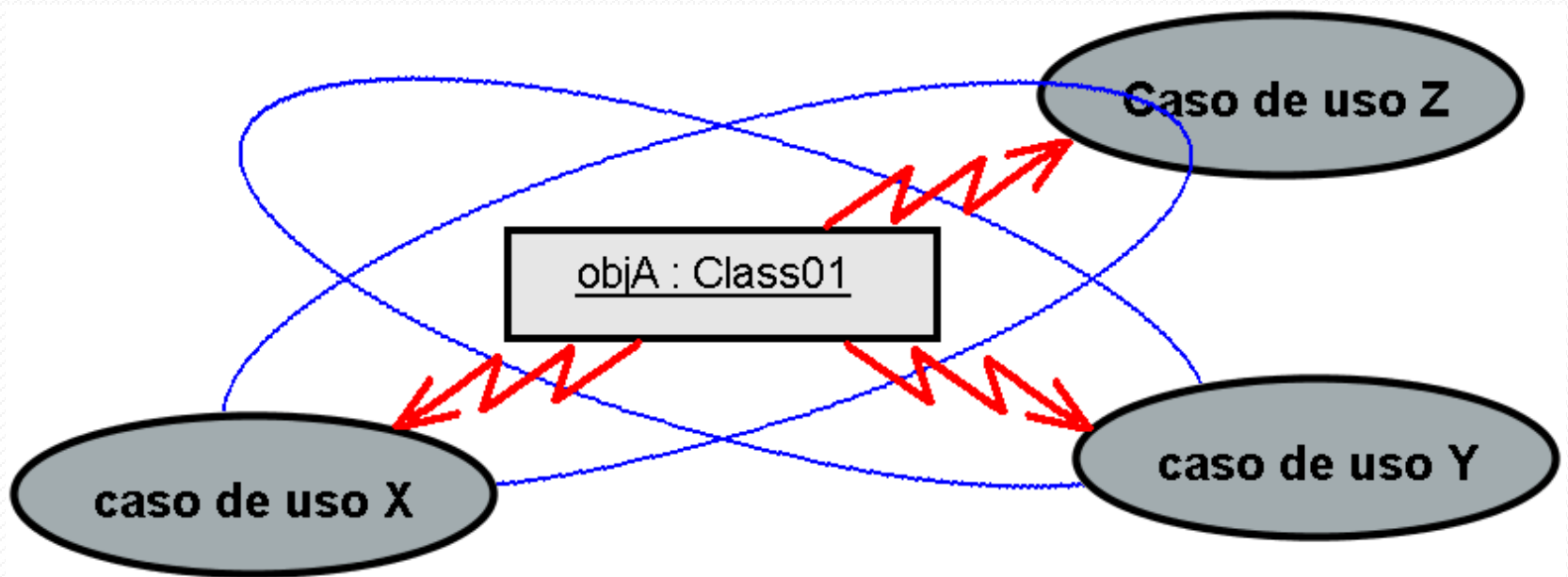
Um ponto de vista diferente, em relação às etapas anteriores

- Refinamento de caso de uso
 - Foco no caso de uso
 - Participação de vários objetos
- Modelagem de estados de classe
 - Foco na existência de um objeto
 - Objeto participa de vários casos de uso

Foco no caso de uso



Foco na existência de um objeto



Passos do esforço de modelagem de estados (finalidade 1)

- Avaliação da adequação da modelagem de estados para cada classe
- Definição da ênfase adequada para a modelagem de estados
- Identificação de estados
- Identificação de possíveis percursos de transição de estados
- Avaliação dos atributos e respectivos valores associados a cada estado

Avaliação da adequação da modelagem de estados para cada classe

- Apenas para classes cujas instâncias passam por um conjunto significativo de estados
 - Julgamento subjetivo
- Um critério → apenas para classes cuja existência de uma instância mereça uma *“historinha”*

Avaliação da adequação da modelagem de estados – exemplo

*Um **carro** pode chegar à oficina com problemas mecânicos, de lataria ou ambos; pode precisar de orçamento referente a problemas mecânicos, de lataria ou ambos; com orçamento produzido, precisa ter esse orçamento aprovado; com orçamento aprovado pode ser submetido à manutenção mecânica, de lataria ou ambas; uma vez consertado, aguarda a quitação das despesas e, após isso, é liberado para o proprietário, quando sai da oficina*

- *E ‘proprietário’? Mereceria uma “historinha”?*

Definição da ênfase adequada na modelagem de estados

- Elementos de domínios de problema admitem modelagens de estados bastante distintas
- É necessário definir a ênfase de modelagem mais adequada

Definição da ênfase adequada na modelagem de estados – exemplo

- Uma outra “historinha” pode ser contada a respeito do elemento ‘carro’ do exemplo anterior

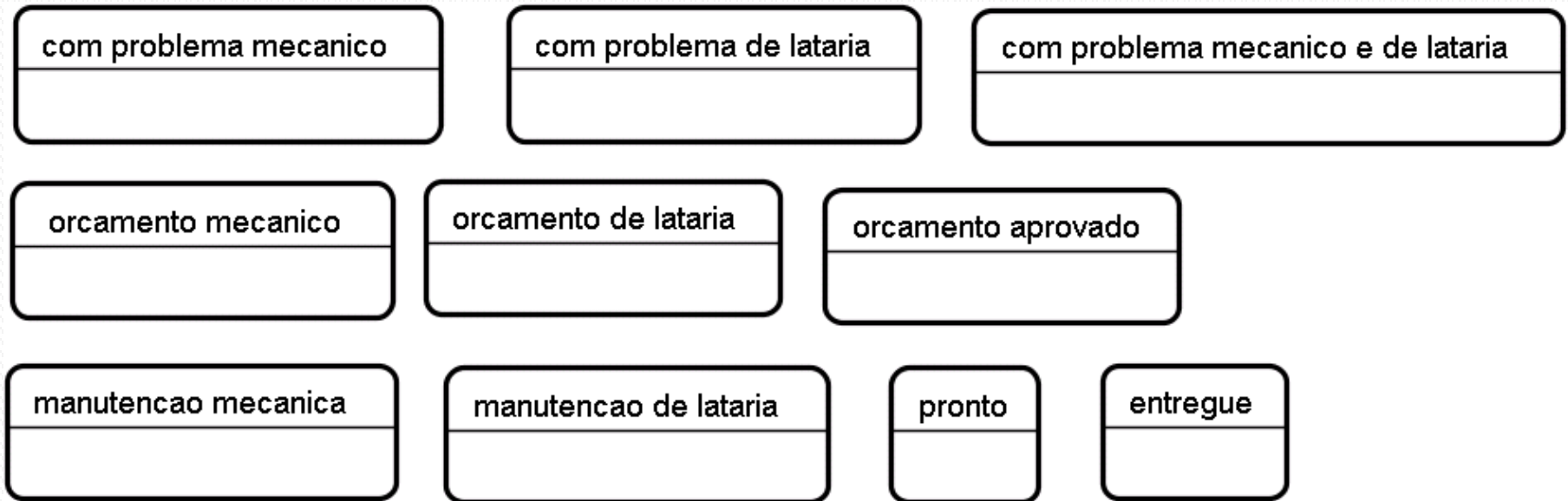
Um carro está inicialmente parado e com o motor desligado. O motor pode estar ligado e o carro ainda parado. Com o motor ligado, se o carro for acelerado ele passará a estar em movimento. O carro em movimento pode estar com velocidade constante, crescente ou decrescente

- Qual a ênfase mais adequada?
 - Depende do que se deseja tratar: movimento ou gerência de oficina?

Identificação de estados

- Definida a ênfase, quais as situações (estados) por que uma instância pode passar?
- Exemplo → 'carro' (no contexto da gerência de oficina)
 - *Com problema mecânico*
 - *Com problema de lataria etc.*

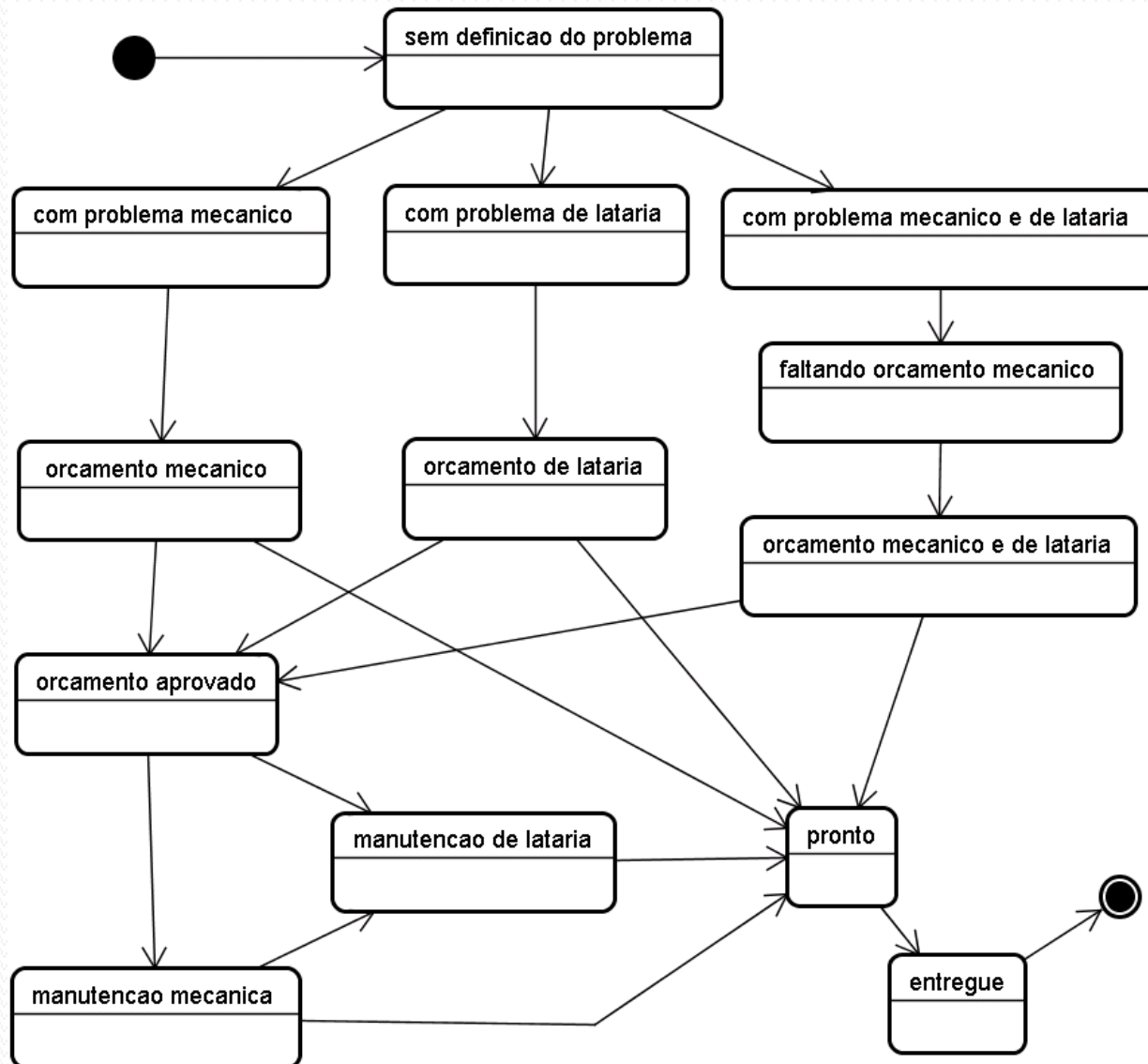
Identificação de estados – exemplo



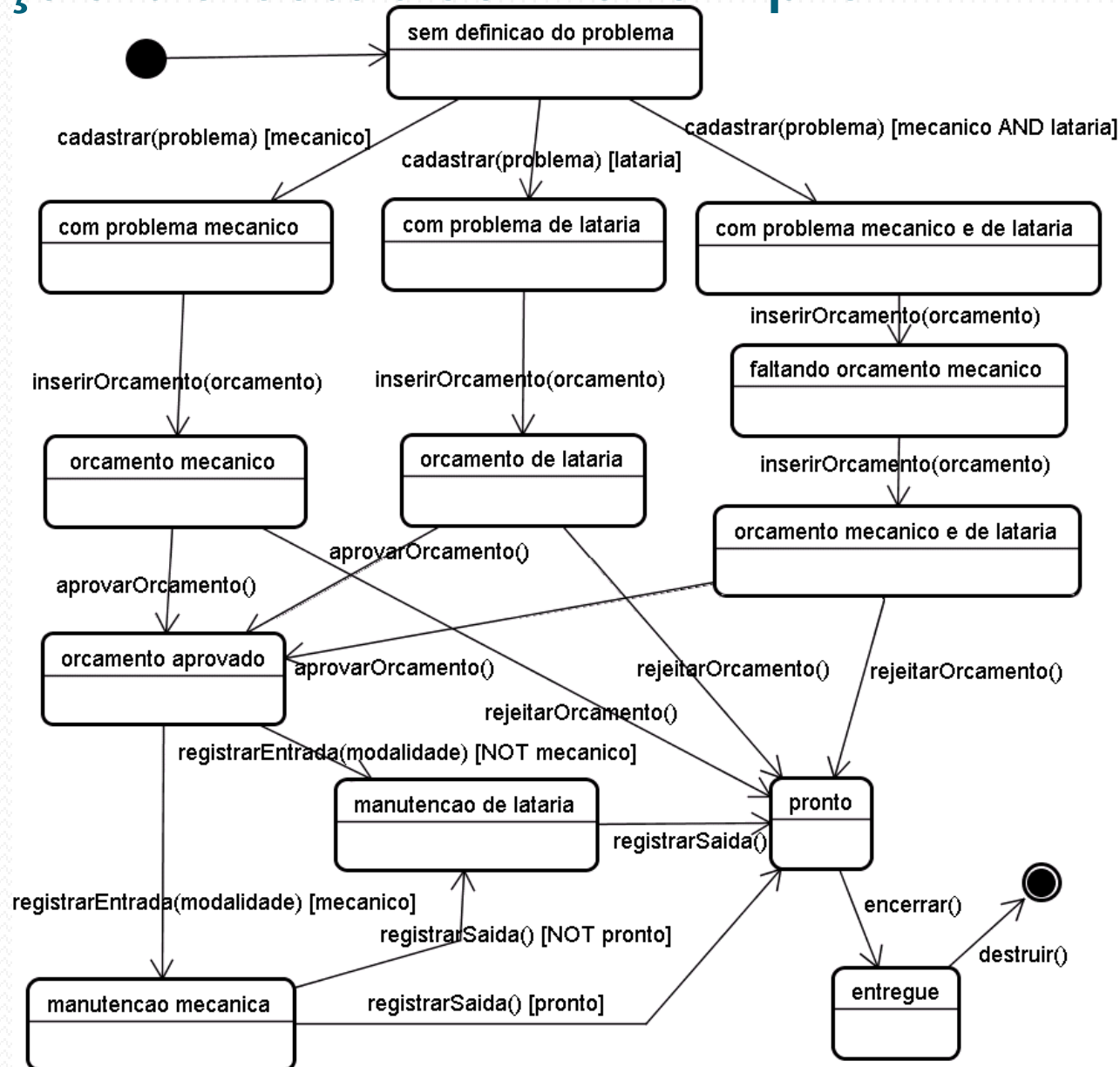
Identificação de possíveis percursos de transição de estados

- Identificar percursos
 - Inserir transições
- Identificar método que provoca cada transição
 - Rotular transições
 - Possibilidade de identificação de novos métodos
- Avaliar a necessidade de guarda

Identificação de possíveis percursos de transição de estados – exemplo



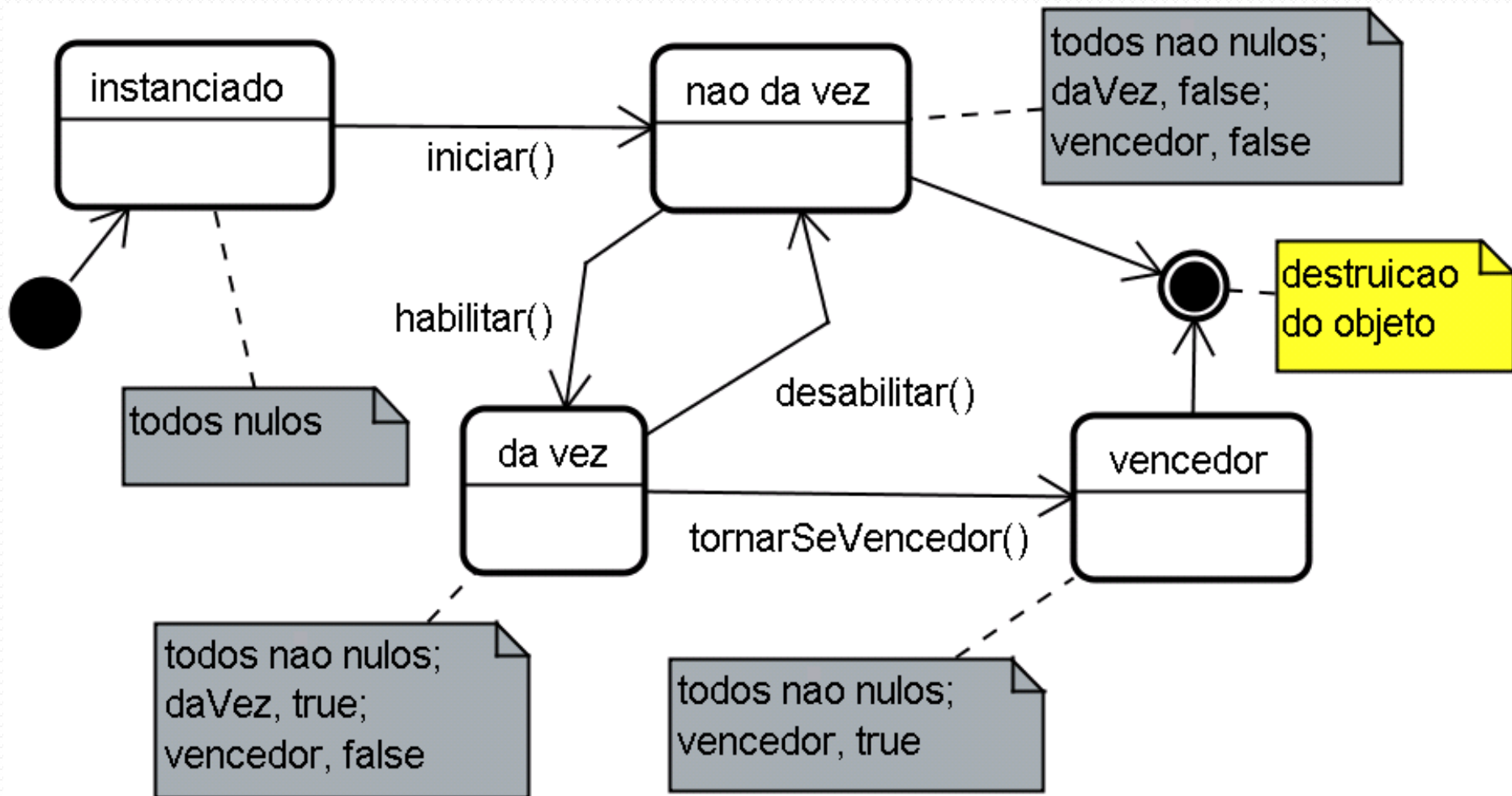
Identificação de possíveis percursos de transição de estados – exemplo



Avaliação dos atributos e respectivos valores associados a cada estado

- Avaliar que valores de atributo caracterizam cada estado
- Caso não seja possível
 - Atributos atuais inadequados
 - Os atributos ou seus tipos → possibilidade de alterá-los
 - Atributos atuais insuficientes
 - Possibilidade de identificar novos

Exemplo → valores de atributos para JogadorHumano (Jogo-da-velha)



Exemplo → valores de atributos para JogadorHumano (Jogo-da-velha)

- Para este exemplo
 - Não houve necessidade de alterar atributos
 - Não houve necessidade de criar novos atributos
 - Modelagem de estados mostrou que atributos previamente identificados são suficientes para descrever todos os estados
 - Cada estado, um conjunto de valores único

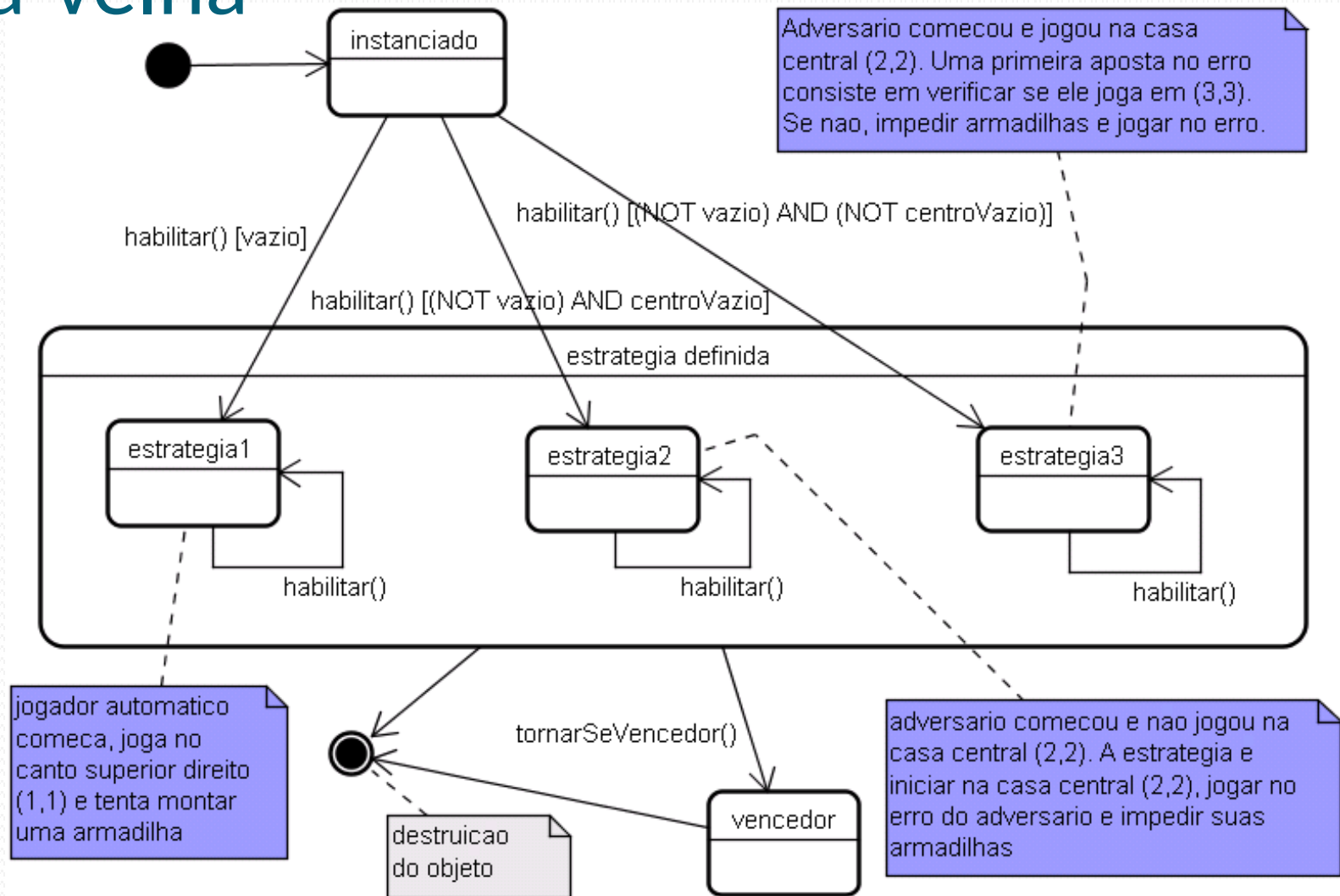
Boa estruturação de algoritmos complexos (finalidade 2)

- Caso de uso → conjunto de *procedimentos*
- Refinamento de procedimento com diagrama de sequência
 - Quando composto por interações entre diferentes objetos
- Refinamento de procedimento com diagrama de máquina de estados
 - Quando composto por responsabilidades alocadas a um único objeto

Exemplo – JogadorAutomático do Jogo-da-velha

- Quando executa o método *habilitar()* deve retornar seu lance
- Algoritmo complexo
 - Responsabilidade de um objeto

Exemplo - JogadorAutomático do Jogo-da-velha



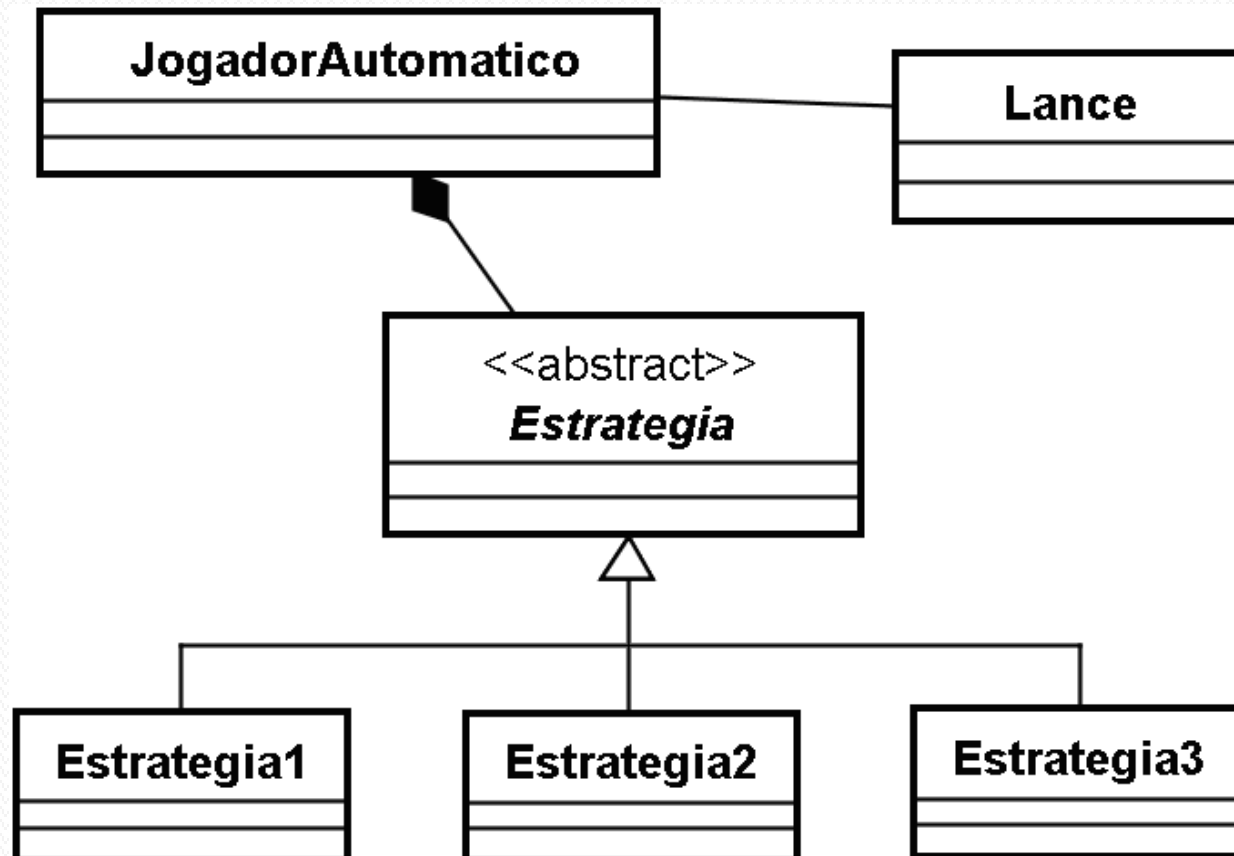
Exemplo – algoritmo de habilitar() do JogadorAutomático do Jogo-da-velha

- Possibilidades
 - Pode ser o primeiro lance do jogador automático, ou não
 - Sendo o primeiro lance do jogador automático, pode ser que ele inicie a partida, ou não
 - Se jogador automático não inicia a partida, adversário pode ter escolhido qualquer das nove posições
- A relação de possibilidades vai muito além
 - Método longo, com muitos *IFs*, *IF-ELSEs* e *SWITCHes*

Exemplo – algoritmo de habilitar() do JogadorAutomático do Jogo-da-velha

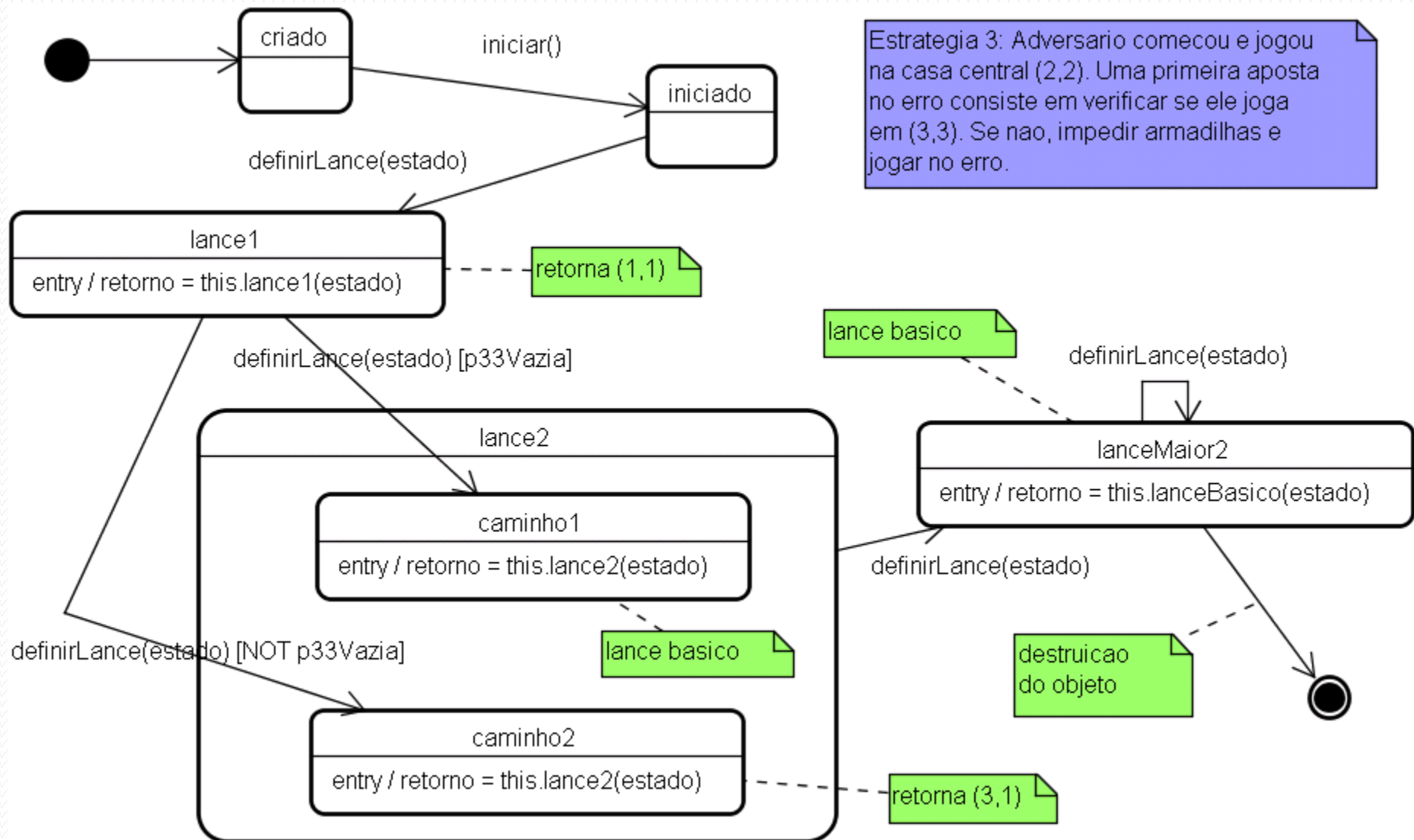
- Solução adotada
 - Não quebrar o método longo em vários e deixar a classe *JogadorAutomatico* com excesso de métodos
 - Cada estratégia (subestado) passa a ser tratada em uma classe diferente
 - Tendência a uma distribuição de métodos mais equilibrada

Exemplo – algoritmo de habilitar() do JogadorAutomático do Jogo-da-velha

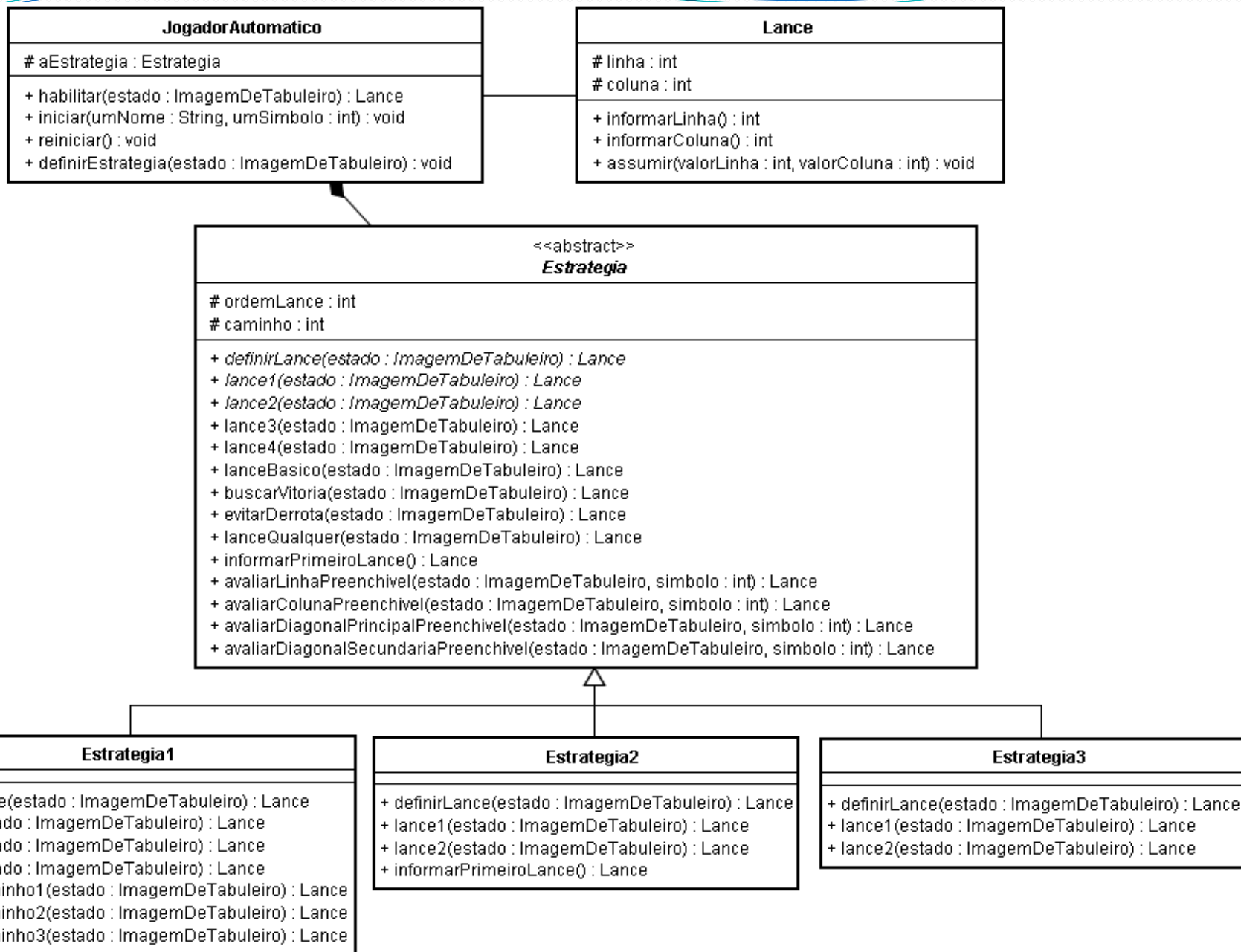


Modelagem de estados da classe

'Estrategia3'



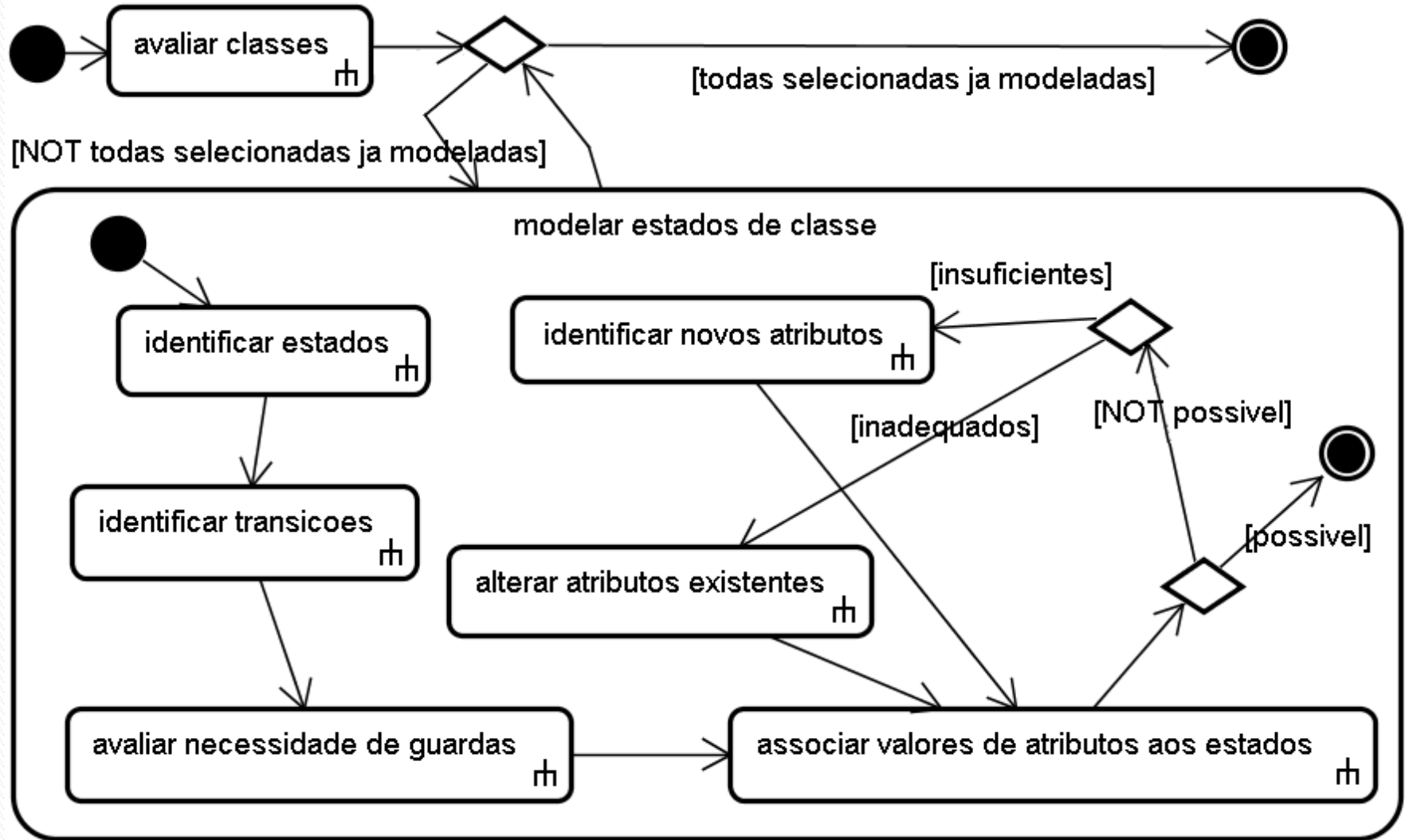
Estrutura obtida com modelagem de estados



Resultados parciais após a quarta etapa do processo de modelagem

- Etapa 1
 - Primeira modelagem estrutural e dinâmica de sistema
- Etapa 2
 - Identificação de atributos
- Etapa 3
 - Refinamento dos casos de uso
- Etapa 4
 - Classes com modelagem de estados, o que pode servir a
 - Descrição da existência de instâncias de classes com comportamento dinâmico relevante;
 - Detalhamento de responsabilidades alocadas a uma única classe

Sumário da quarta etapa do processo de modelagem



Considerações sobre esta aula

- Etapa 4 do processo de modelagem → Modelagem de estados associada a classe
 - Passos da modelagem
- Duas finalidades
 - A avaliação dos atributos até então identificados e identificação de novos
 - Boa estruturação de algoritmos complexos

Referências

Booch, G.; Jacobson, I. e Rumbauch, J. **UML: Guia do Usuário**. Campus, 2006.

Silva, R. P. **UML 2 em modelagem orientada a objetos**. Visual Books, 2007.