



**PCS3413**

# Engenharia de Software e Banco de Dados

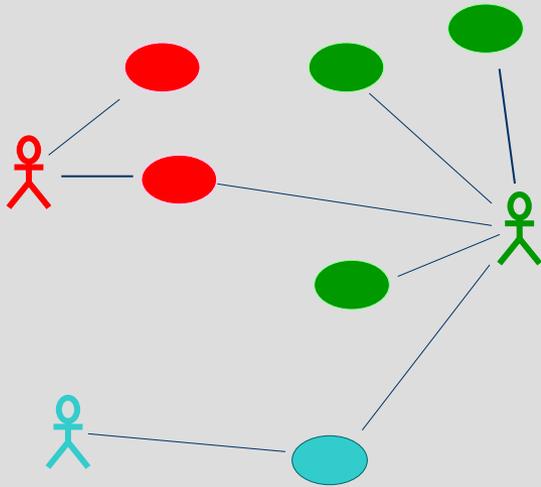
**Aula 7**

# Diagramas de Interação

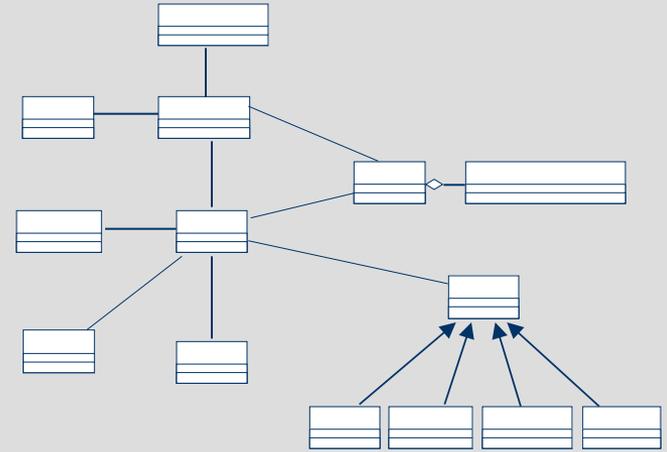
Diagramas de Sequência  
Diagrama de  
Comunicação

# Principais Modelos

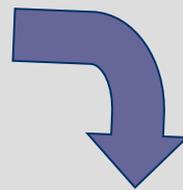
- Modelo de Casos de Uso: descreve as funções do sistema.
  - Diagrama de Casos de Uso
- Modelo Estrutural: descreve a estrutura do sistema.
  - Diagrama de Classes
- Modelo Dinâmico: descreve o comportamento do sistema.
  - Diagrama de Interação
  - Diagrama de Estados



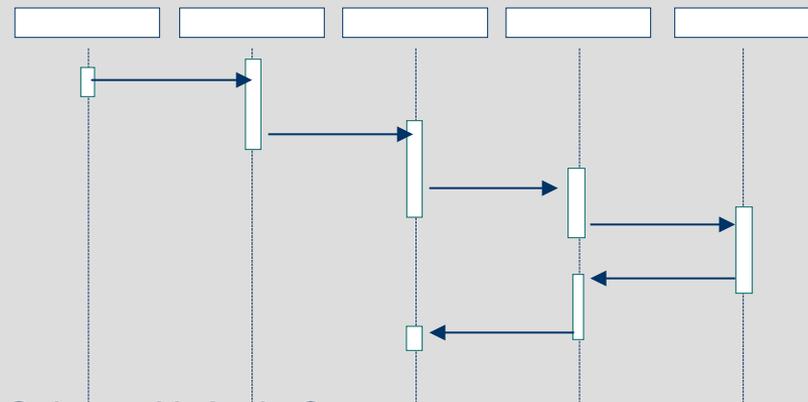
Descrição de casos de uso



Descrição de casos de uso



Classes e relacionamentos



Solange N. A. de Souza

# Diagramas de Interação

## ■ interação:

- “conjunto de mensagens trocadas entre objetos, com o objetivo de alcançar determinado propósito, respeitando-se o contexto do sistema.”

Ana Cristina Melo

# Diagramas de Interação

- diagrama de sequência
  - sequência de mensagens no tempo
- diagrama de comunicação (antigo colaboração)
  - relacionamento estrutural entre objetos
- Diagramas de Sequência e de Comunicação -  
representam: **cenário de caso de uso**

# Perguntas

- No modelo de casos de uso:
  - Quais são as operações que devem ser executadas internamente ao sistema?
  - A que classes estas operações pertencem?
  - Quais objetos participam da realização deste caso de uso?

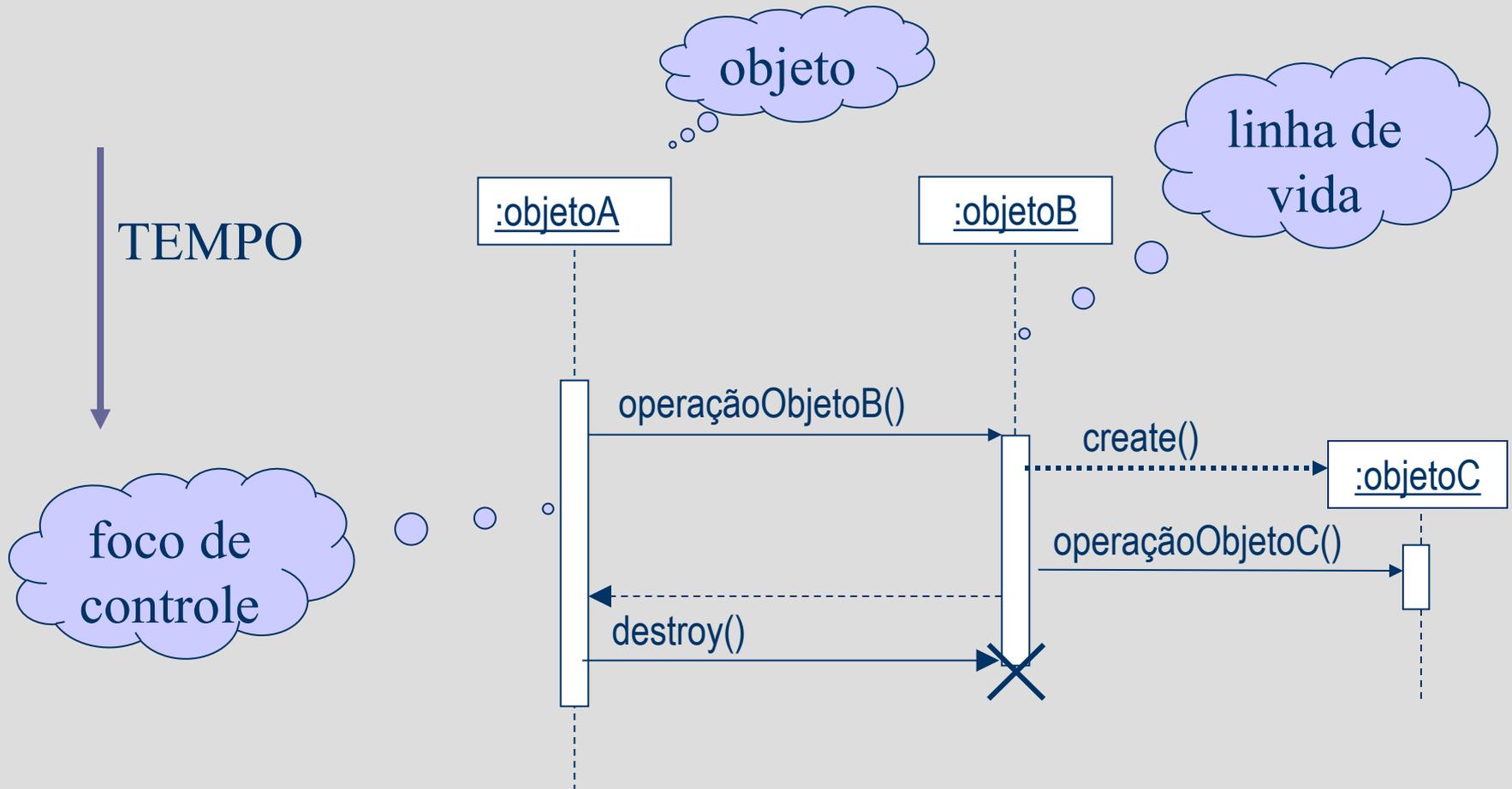


# Perguntas

- No modelo de classes de análise:
  - De que forma os objetos colaboram para que um determinado caso de uso seja realizado?
  - Em que ordem as mensagens são enviadas durante esta realização?
  - Que informações precisam ser enviadas em uma mensagem de um objeto a outro?
  - Como as classes se comportam?



# Elementos do Diagrama



# Mensagens

- Uma mensagem representa a requisição de um objeto remetente para um objeto receptor.
- Uma mensagem é o acionamento de uma operação do objeto receptor por um objeto remetente.
- O envio de uma mensagem corresponde à chamada de uma rotina de linguagens de programação.

# Mensagem

	Mensagem síncrona
	Mensagem assíncrona
	Mensagem de retorno
	Mensagem para criação de objeto
	Mensagem para destruição de objeto

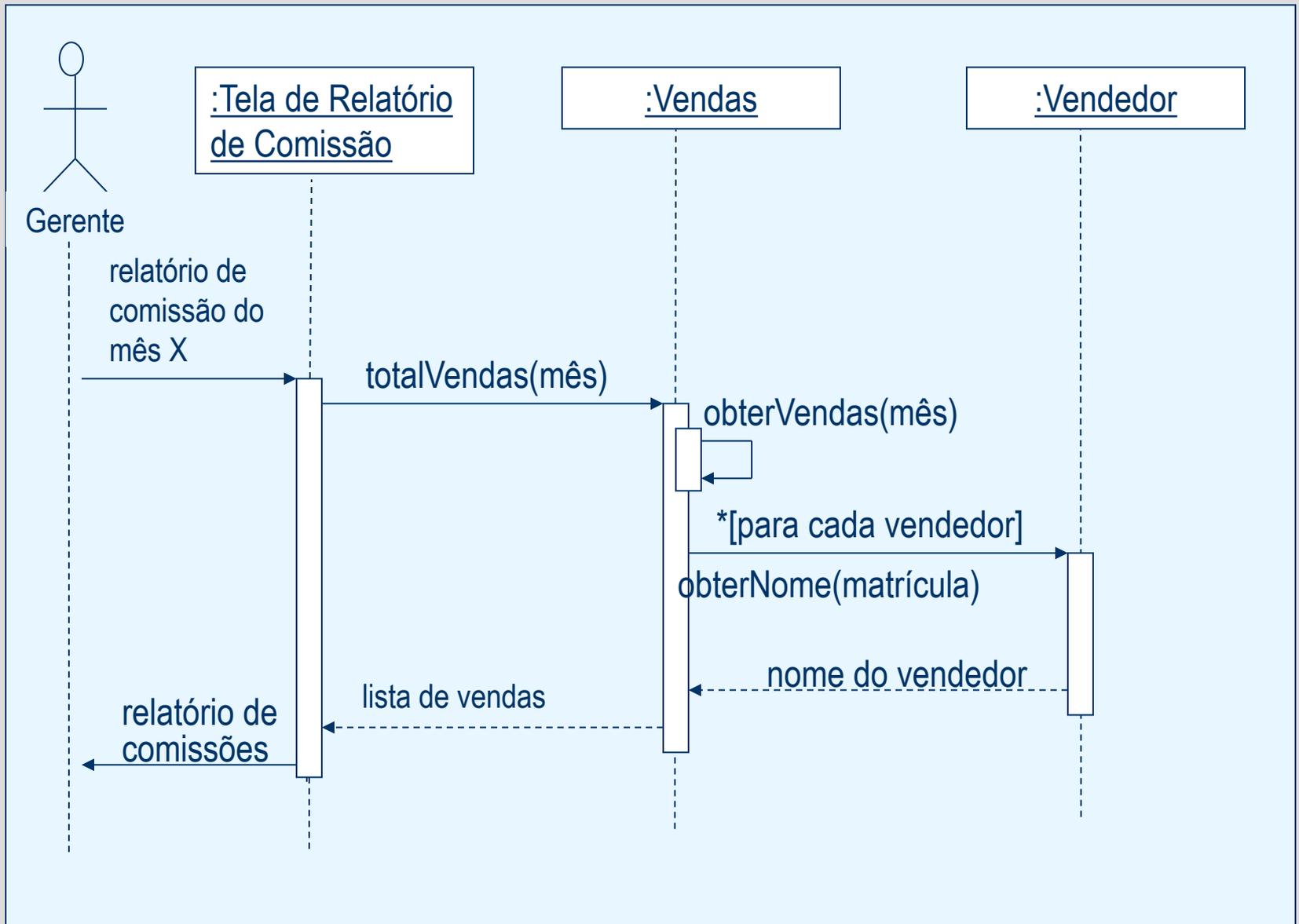
O remetente fica bloqueado, esperando que o receptor termine de atender à requisição.

O remetente não espera a resposta para prosseguir o processamento.

Especifica o término de atendimento de uma mensagem.

# Diagrama de Sequência





# Objetos

objeto  
nomeado

umaDisciplina: Disciplina

objeto  
anônimo

: Disciplina

objeto em  
uma coleção

preRequisitos [i]: Disciplina

faz referência a uma instância  
da classe que está armazenada  
numa coleção de objetos

# Exemplo

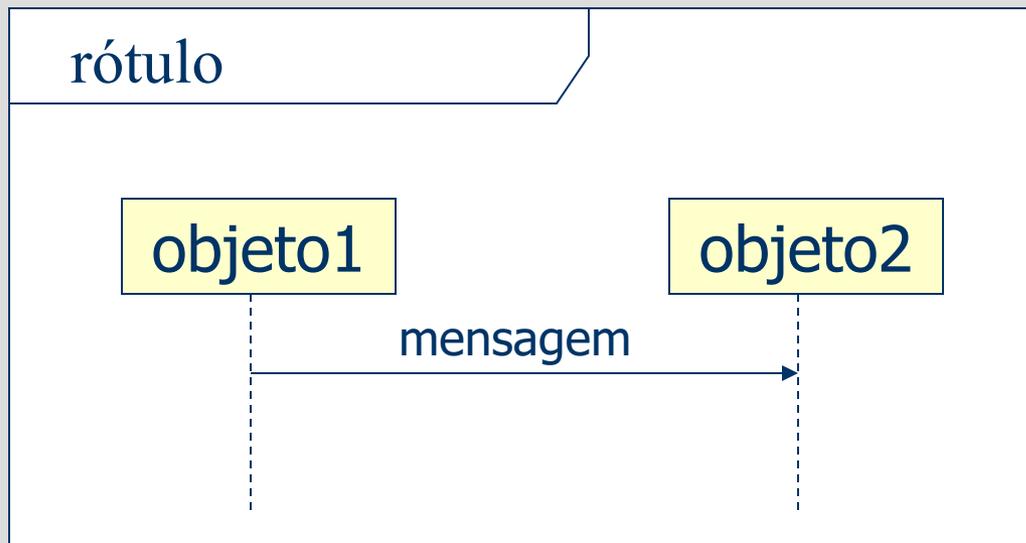


# Modularização de interações (UML2)

- Usado para identificar o diagrama de sequência.
- Usado também para representar diagramas como parte de outros.
- Usado para definir o fluxo de controle da interação.
- Os frames (ou quadros) são opcionais (compatibilidade com modelos antigos).

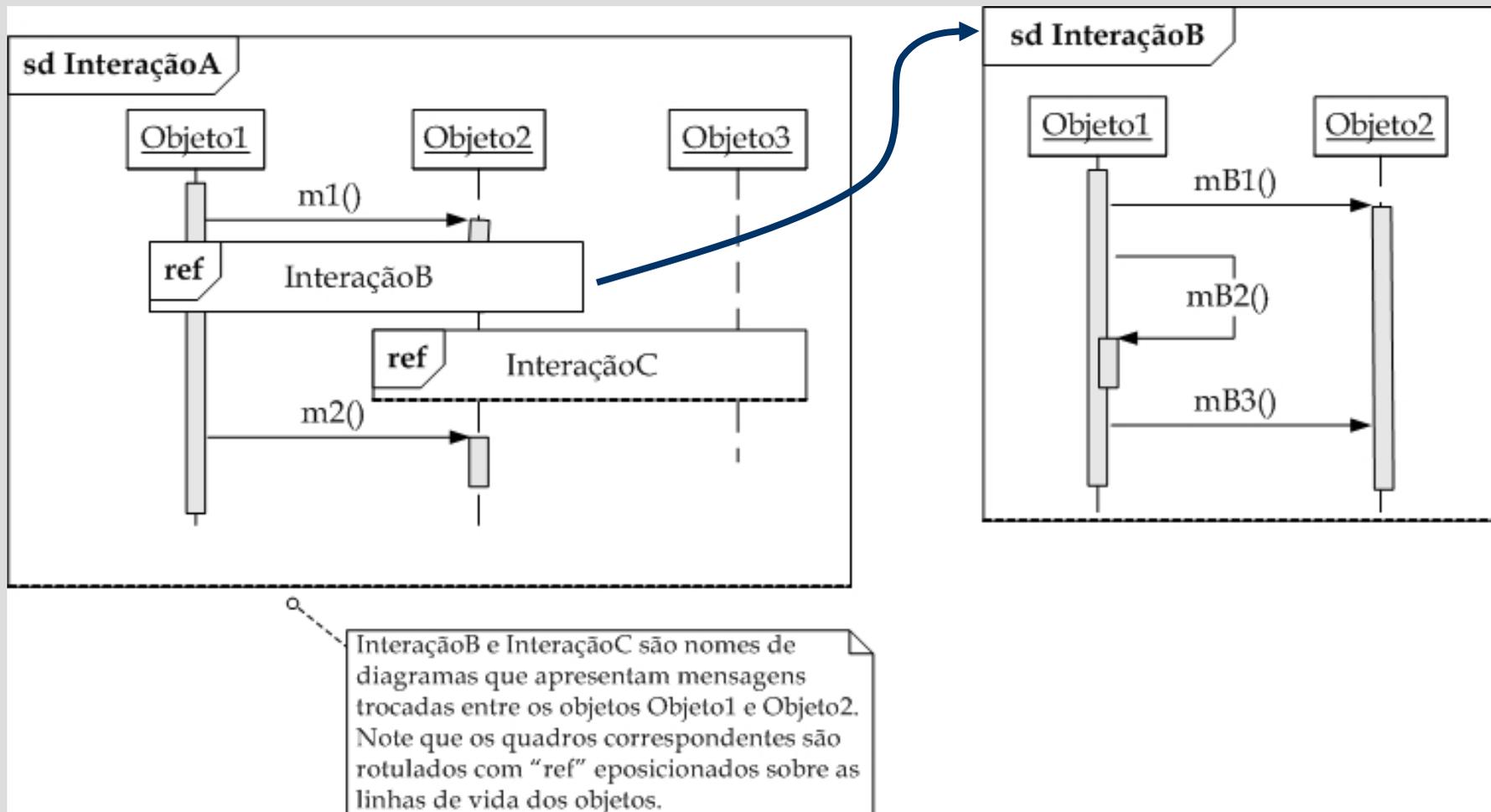
## Frames - continuação

- Serve para encapsular um diagrama de sequência.



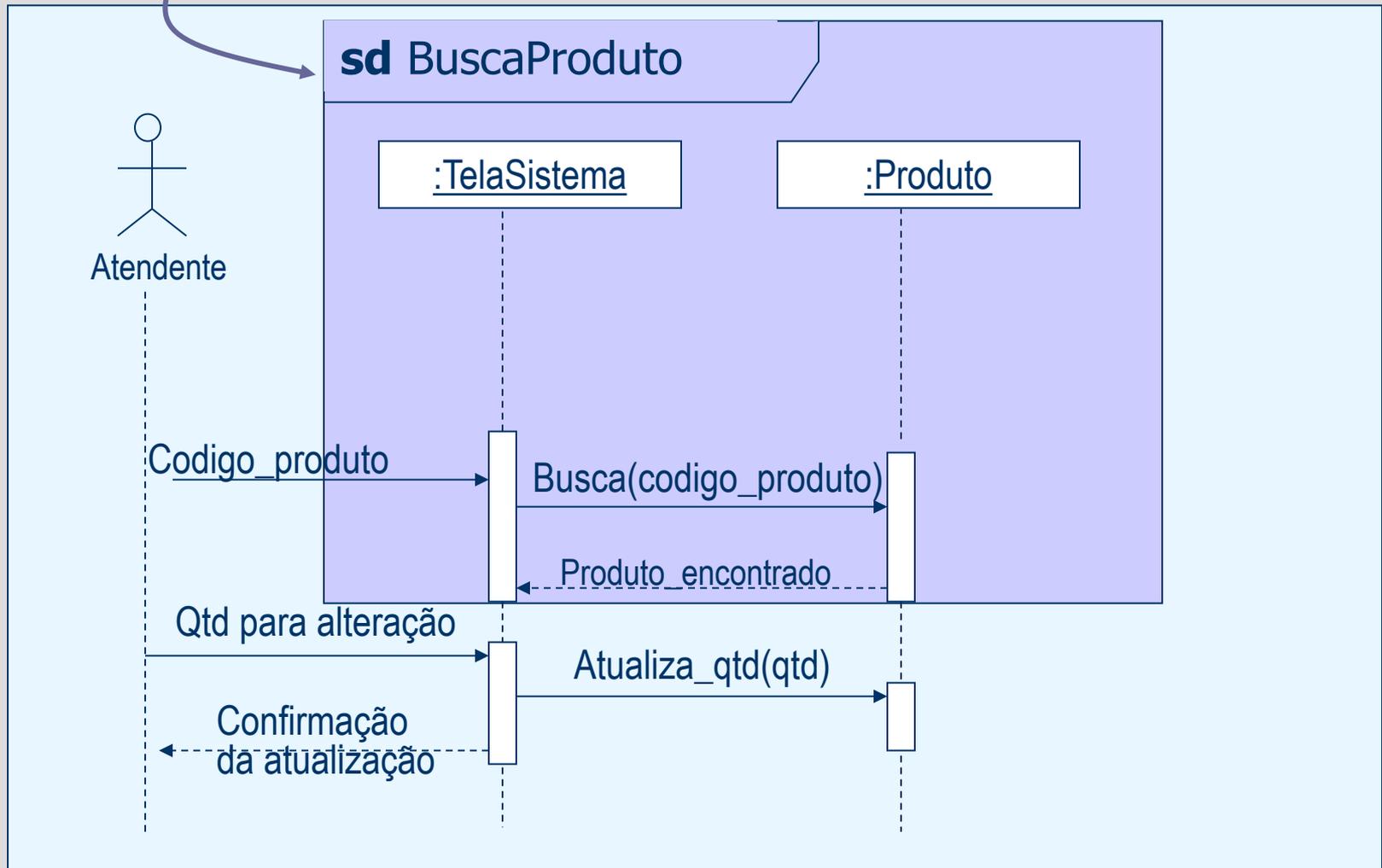
Um diagrama (ou um nome de um diagrama) é posicionado no interior do quadro

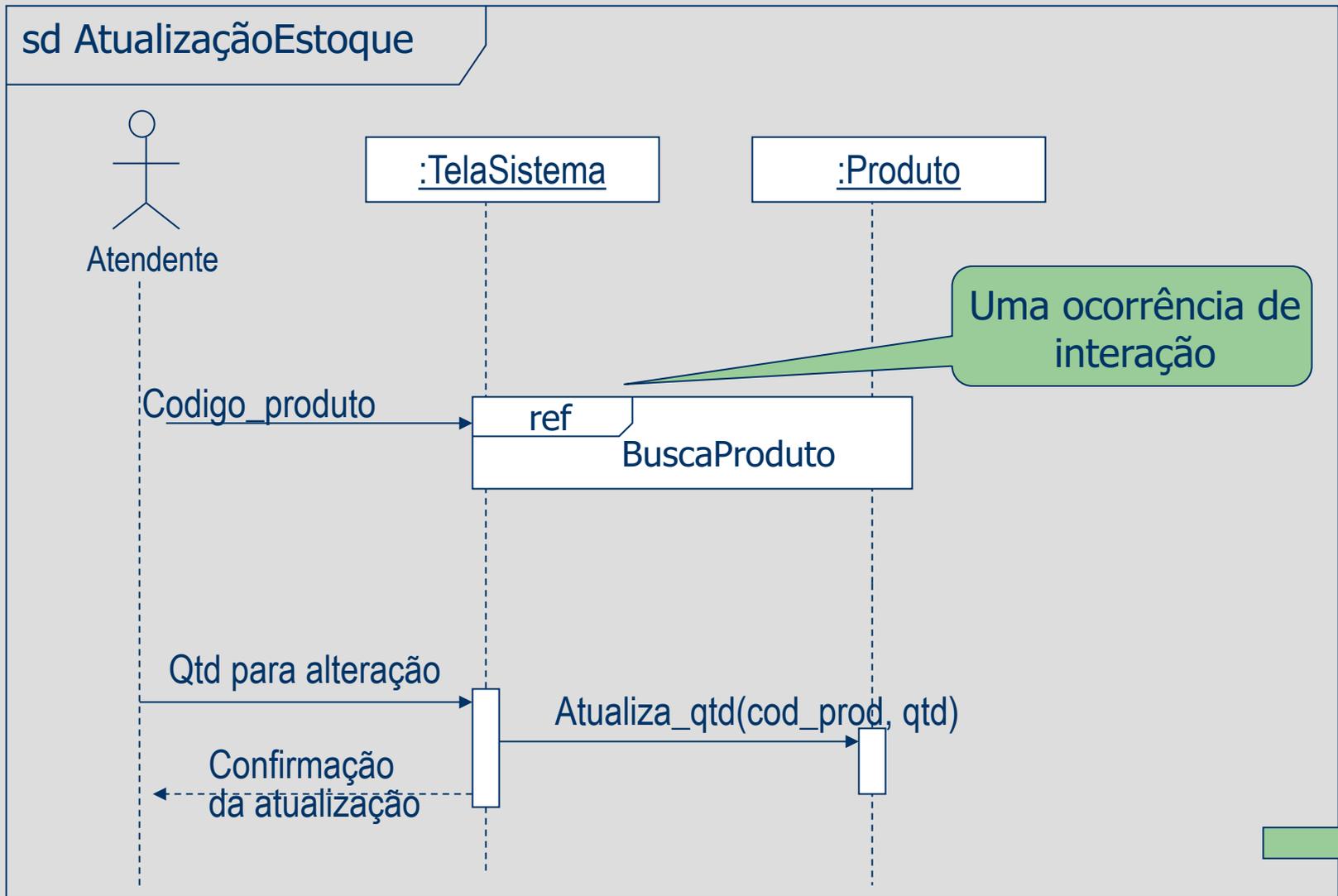
# Diagramas Referenciados



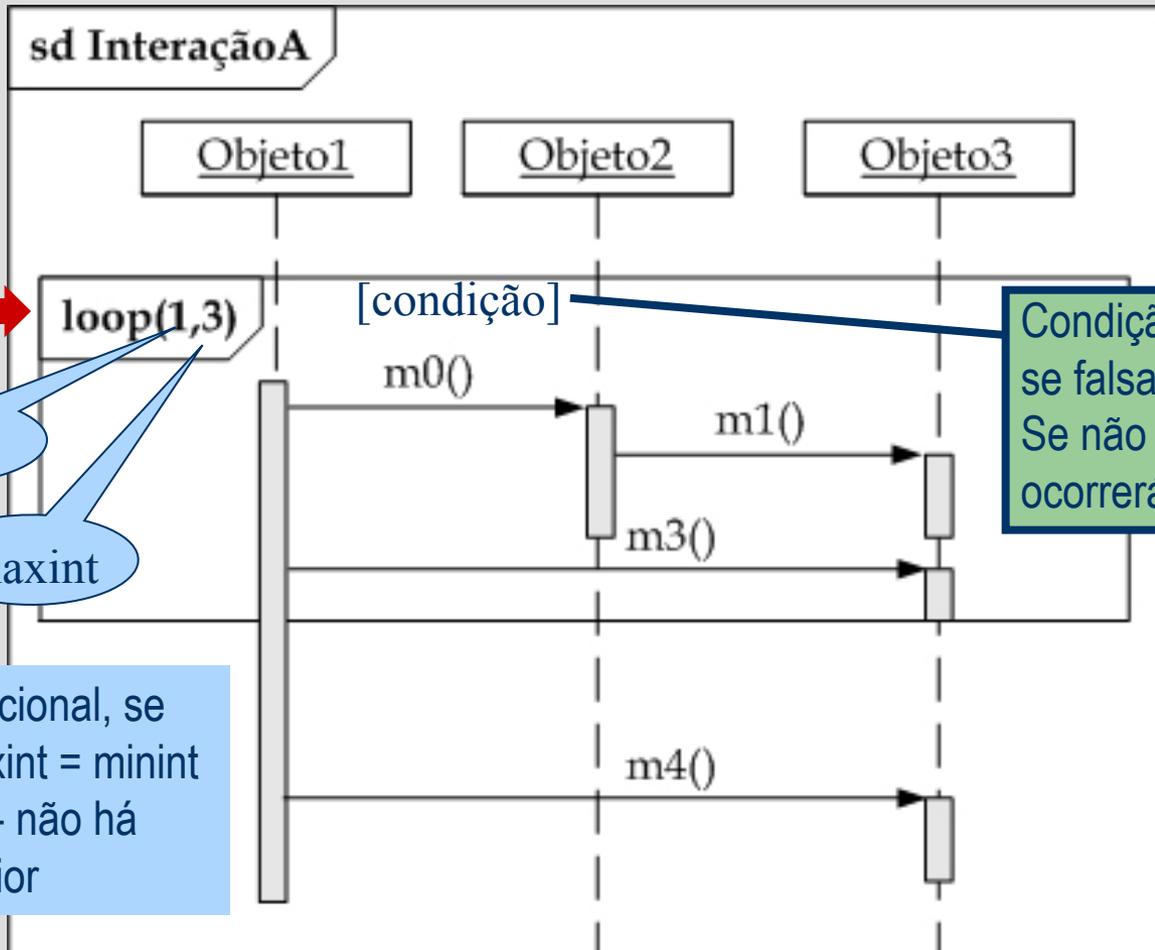
# Frames - continuação

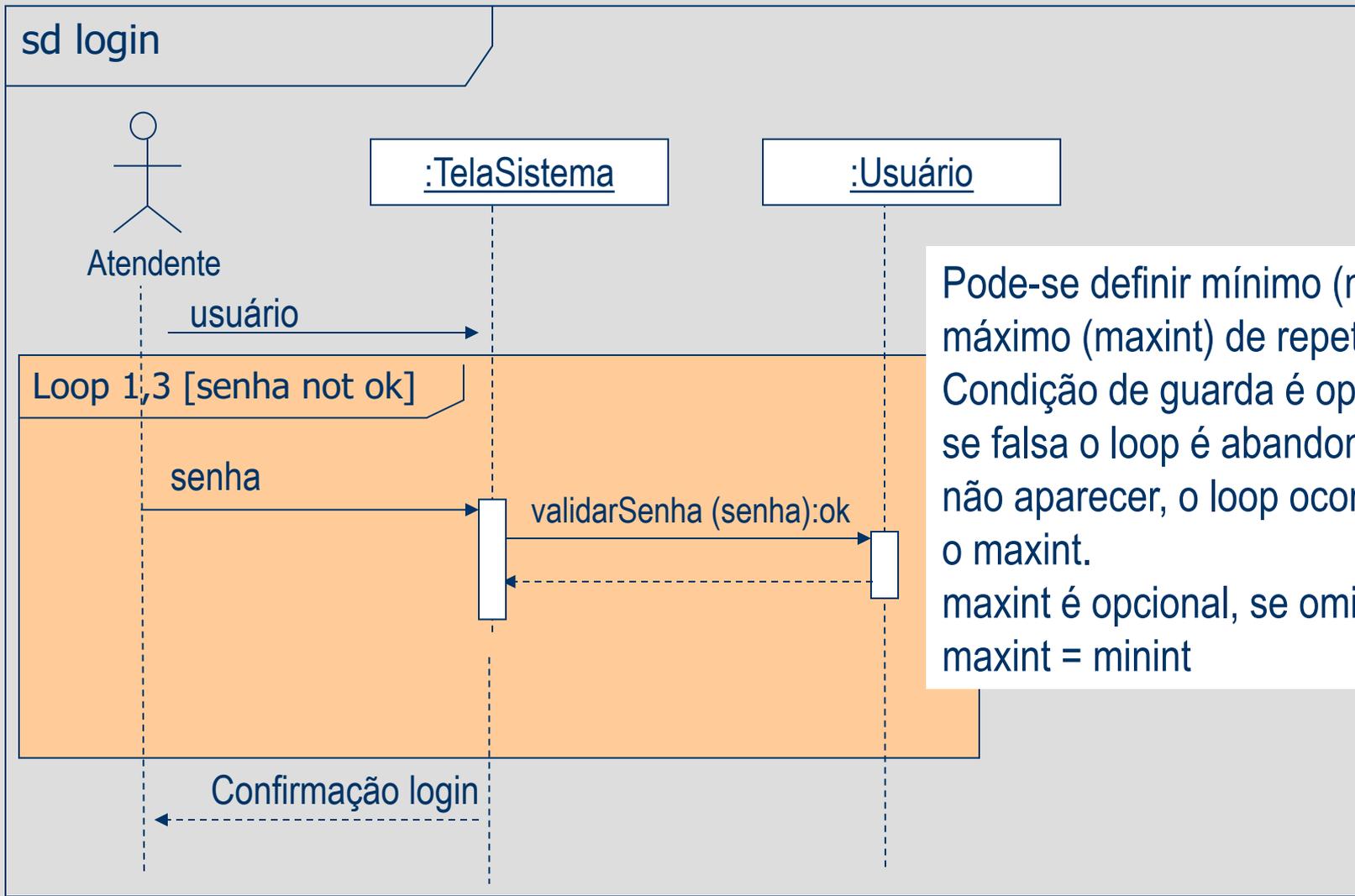
sequence diagram





# Fluxo de Controle: Iteração



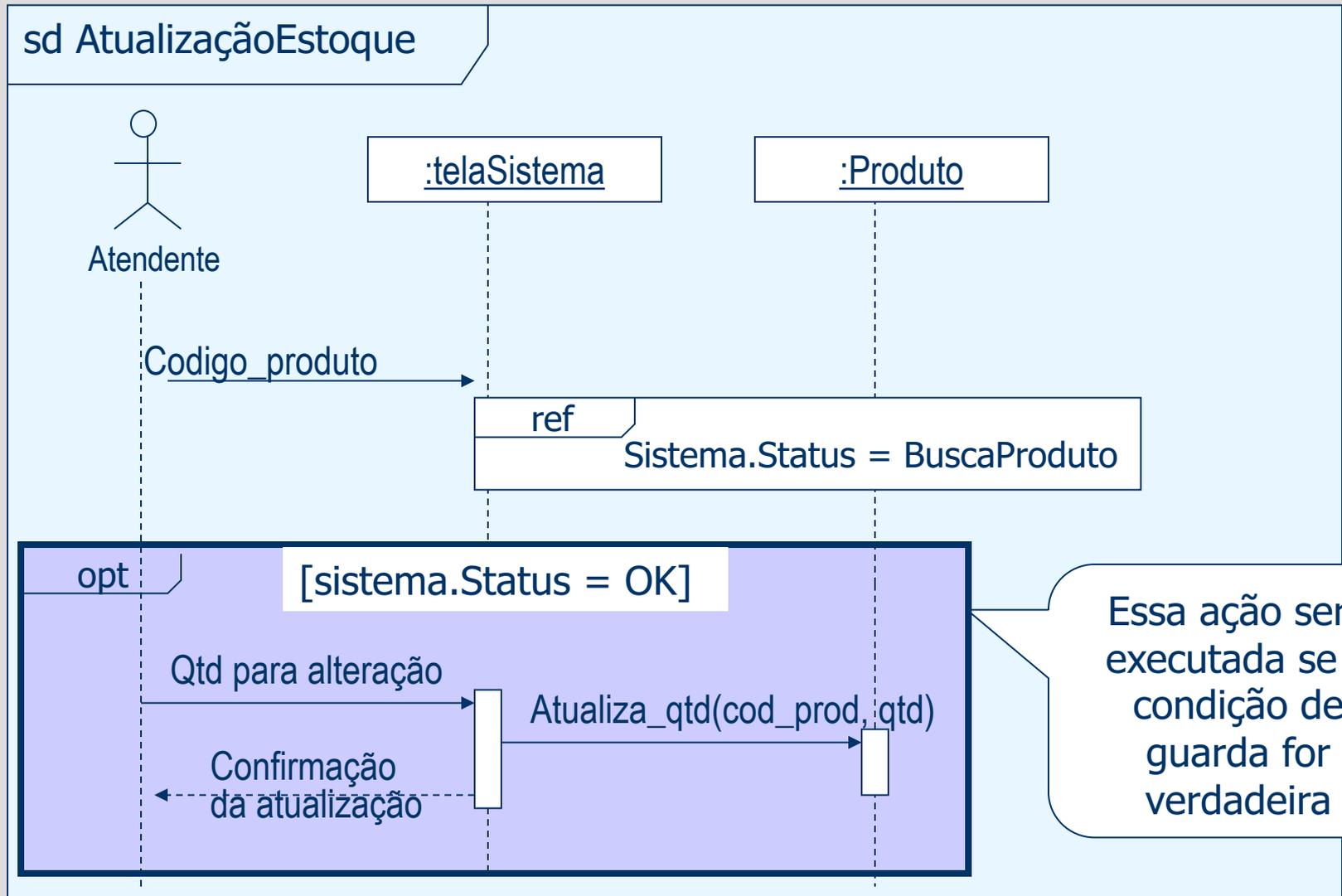


Pode-se definir mínimo (minint) e máximo (maxint) de repetições. Condição de guarda é opcional, se falsa o loop é abandonado, se não aparecer, o loop ocorrerá até o maxint. maxint é opcional, se omitido maxint = minint

# Fluxo de Controle: Opção

- Frame opt: usado para indicar que um trecho interação é opcional.

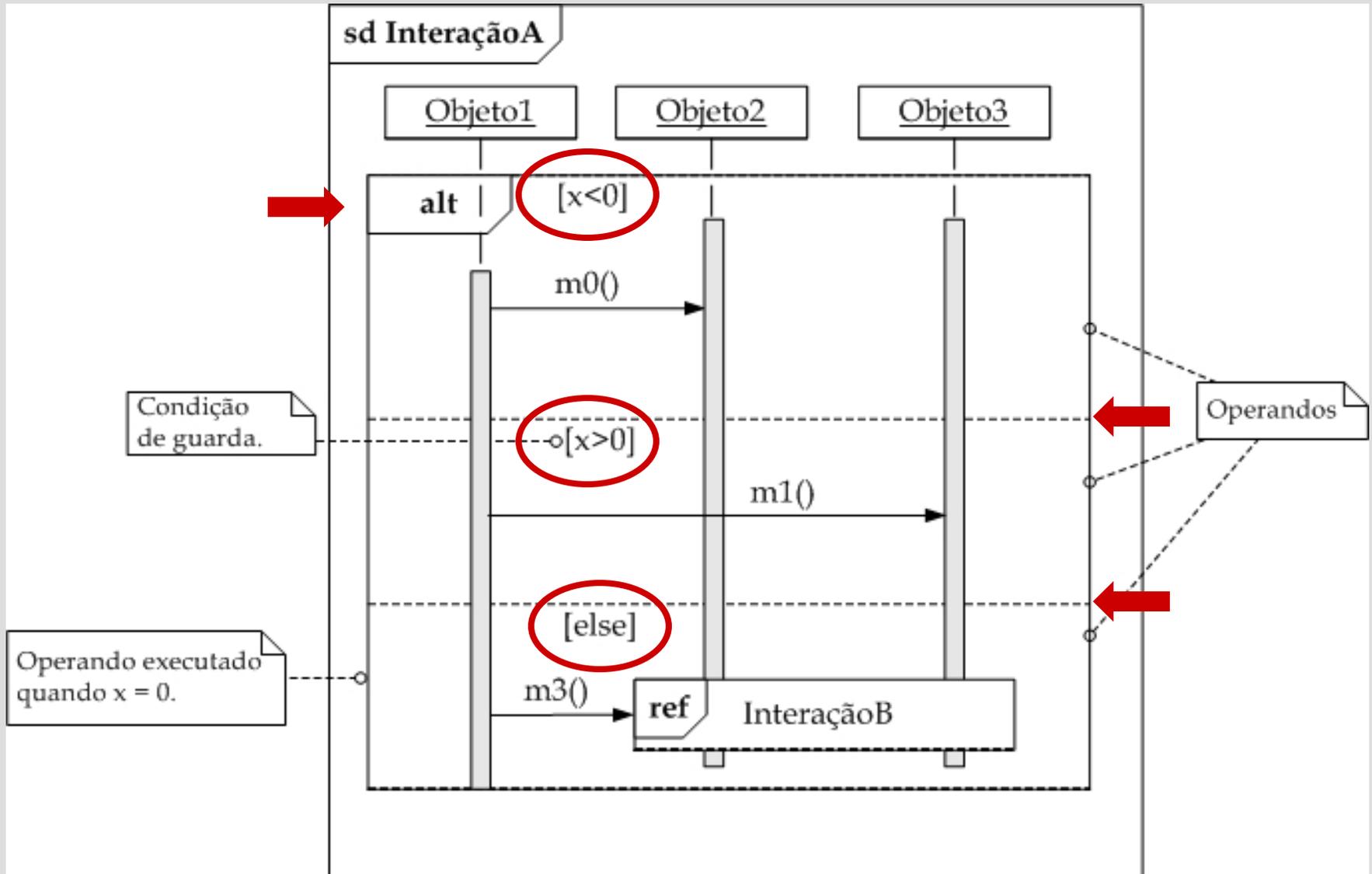
# Interação opt

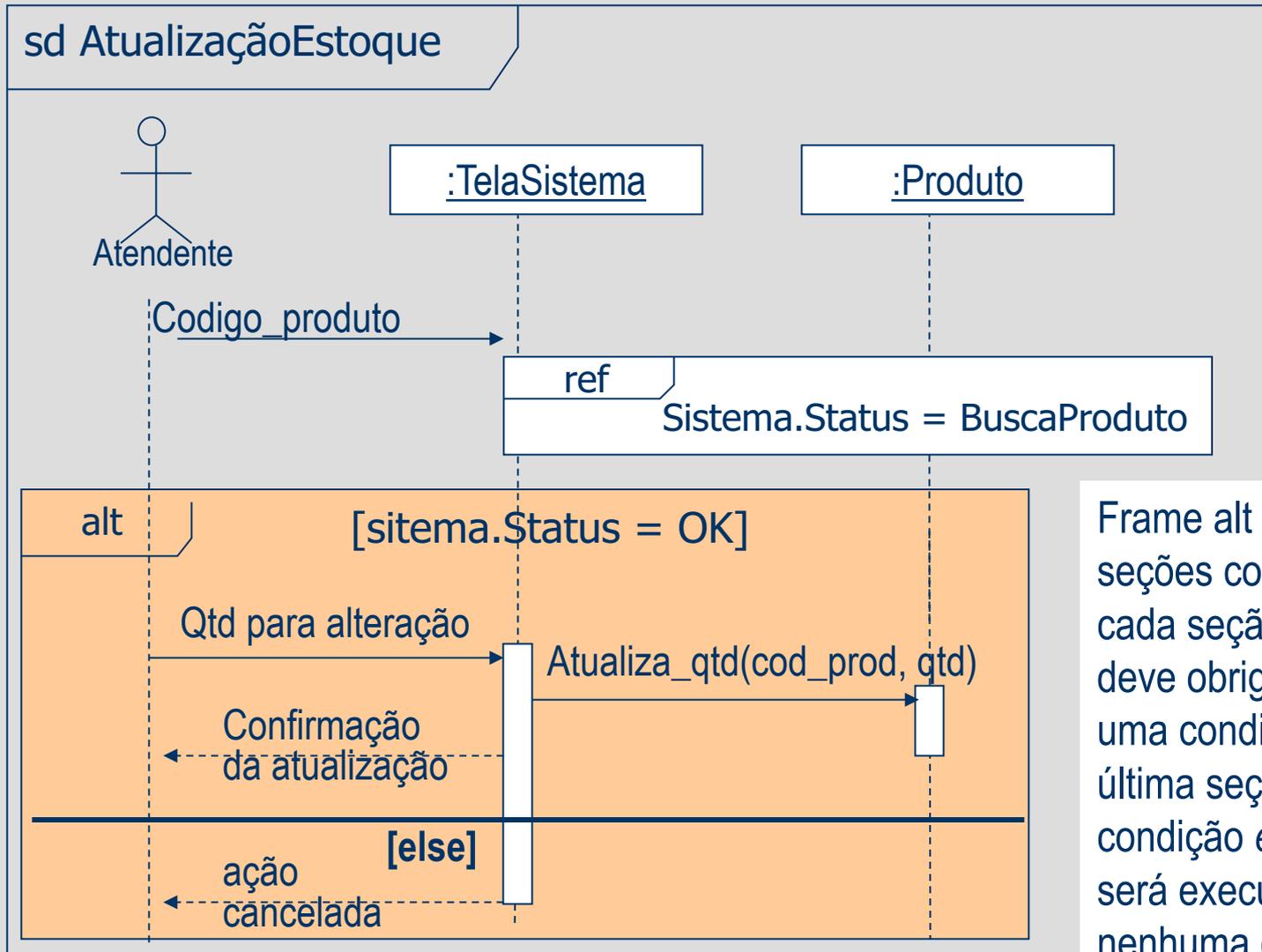


# Fluxo de Controle: Alternativa

Frame alt é dividido em seções com interações em cada seção. Cada seção deve obrigatoriamente ter uma condição de guarda. A última seção pode utilizar a condição *e/se* (a mesma será executada se nenhuma outra tiver sido válida).

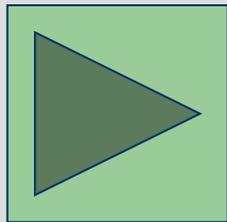
# Fluxo de Controle: Alternativa





Frame alt é dividido em seções com interações em cada seção. Cada seção deve obrigatoriamente ter uma condição de guarda. A última seção pode utilizar a condição e/se (a mesma será executada se nenhuma outra tiver sido válida).

# Exemplo com os frames



# Exemplo - Descrição de Caso de Uso

## Registrar vendas

**objetivo:** permite cadastrar as vendas efetuadas pelos vendedores de uma loja.

**Ator:** Assistente de gerência (usuário)

DiagSeq

**Pré-condição:** usuário na opção de registro de venda

## **Cenário Principal**

Diag.  
Classe

1. O sistema prepara uma lista dos vendedores cadastrados na loja.
2. O usuário seleciona o vendedor que efetuou a venda, a partir da lista já montada pelo sistema
3. O usuário informa o número da venda.

4. O usuário informa, ainda:
  - 4.1. a data da venda;
  - 4.2. o valor da venda.
5. O sistema efetua a gravação da venda e Informa (MSG) sobre venda cadastrada

**Pós-condição:** venda registrada para um determinado vendedor

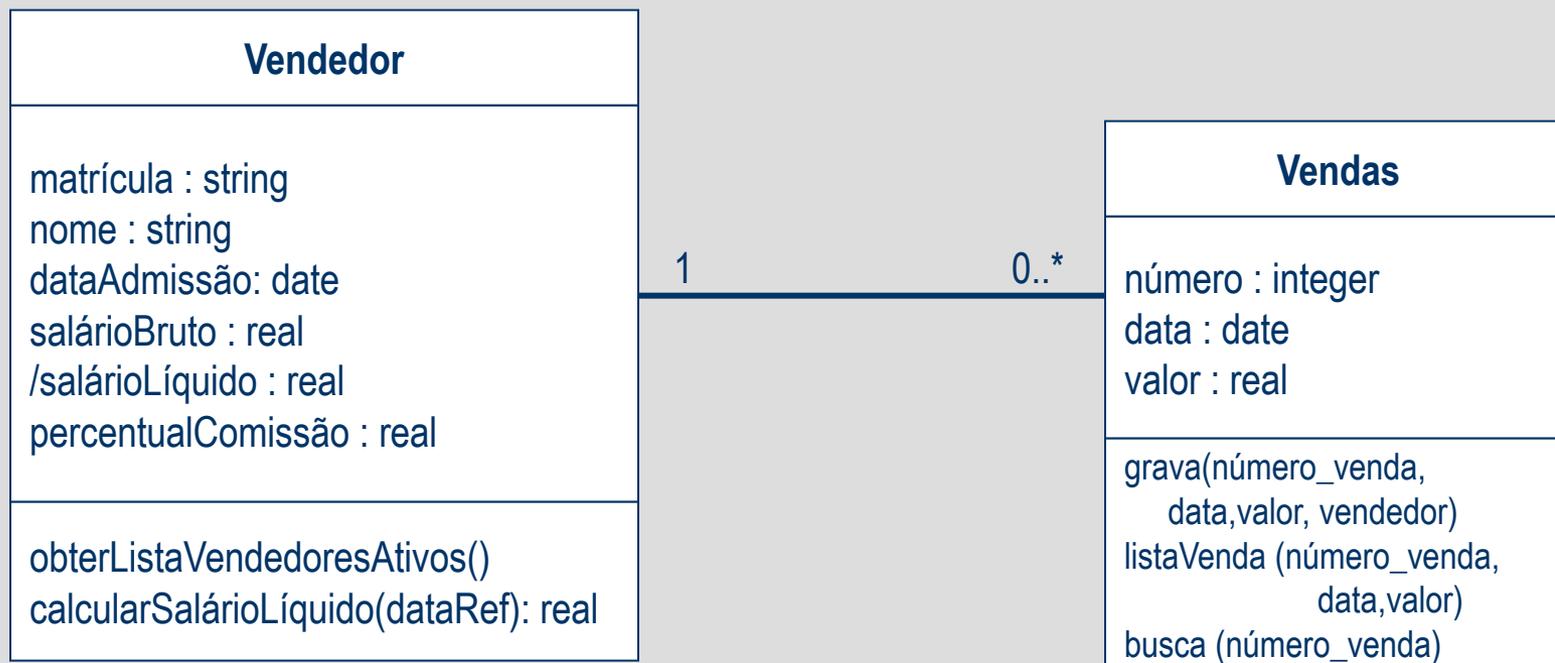
## **Cenário Alternativo**

### Venda já cadastrada

1. Se o número da venda já existir, informar ao usuário, mostrar as informações da venda na tela e entrar em modo de alteração dos dados (passo 3). Inclusão Alterar venda.

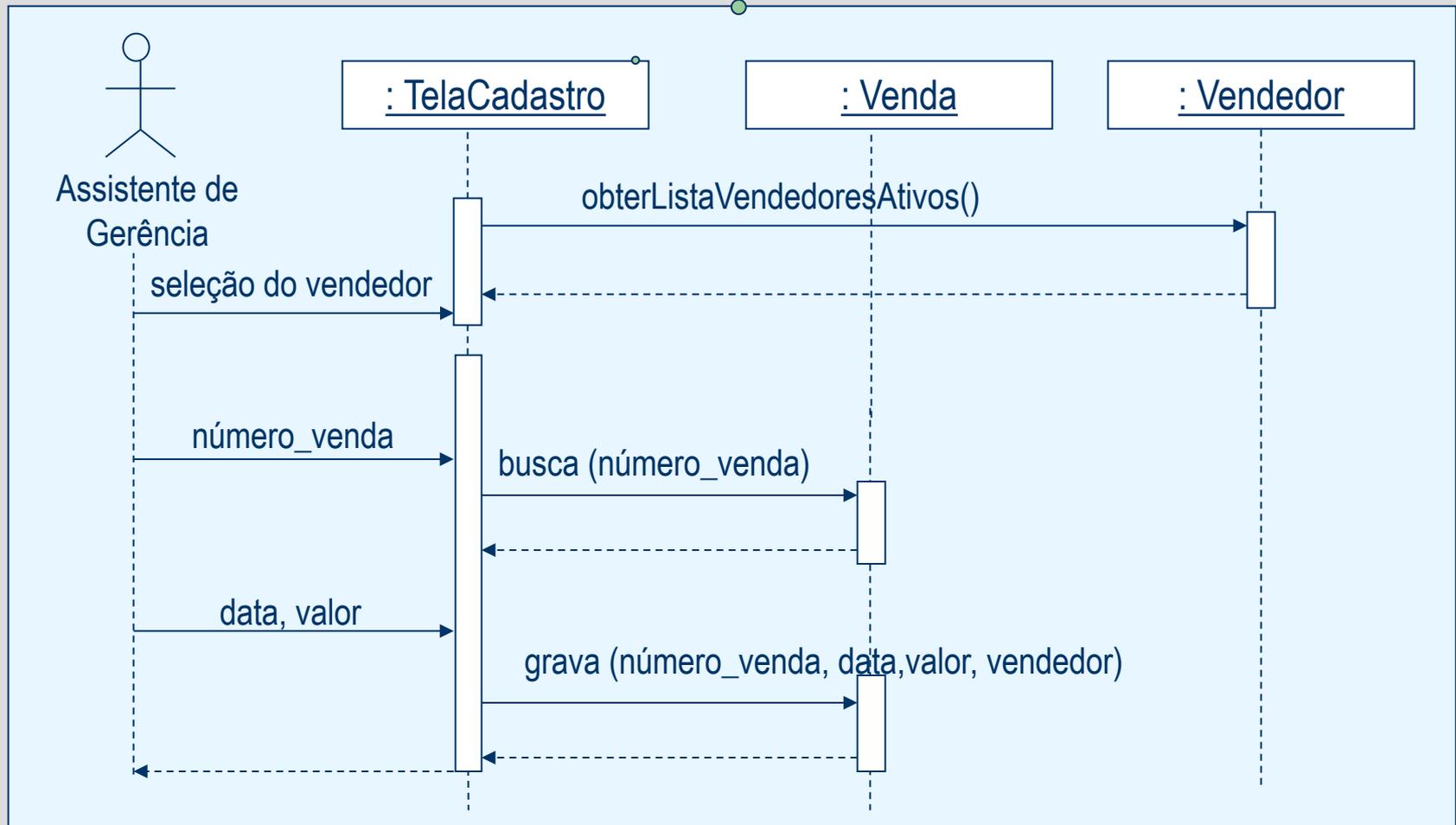
# Diagrama de Classes

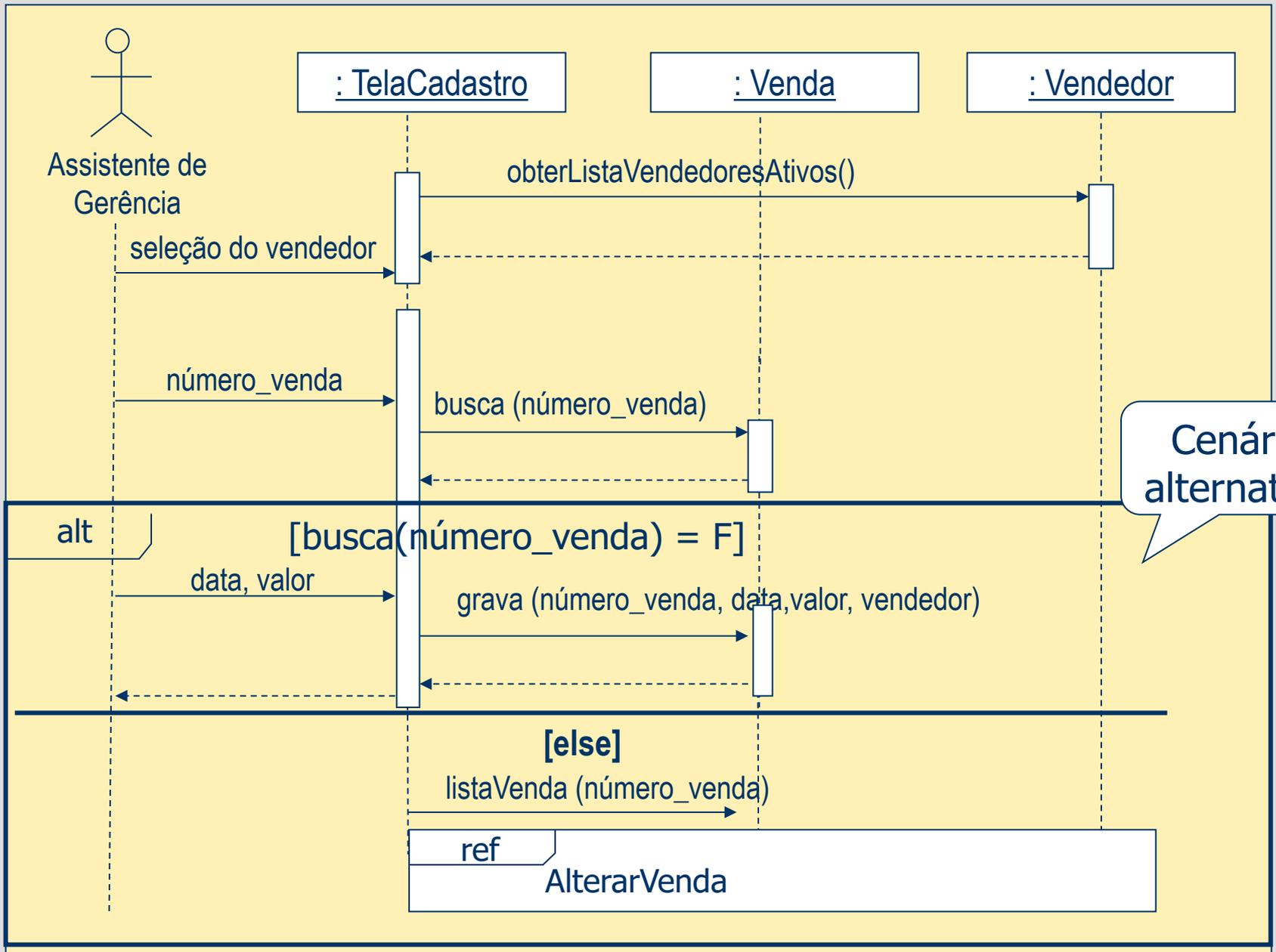
CU





## Diagrama de Sequência

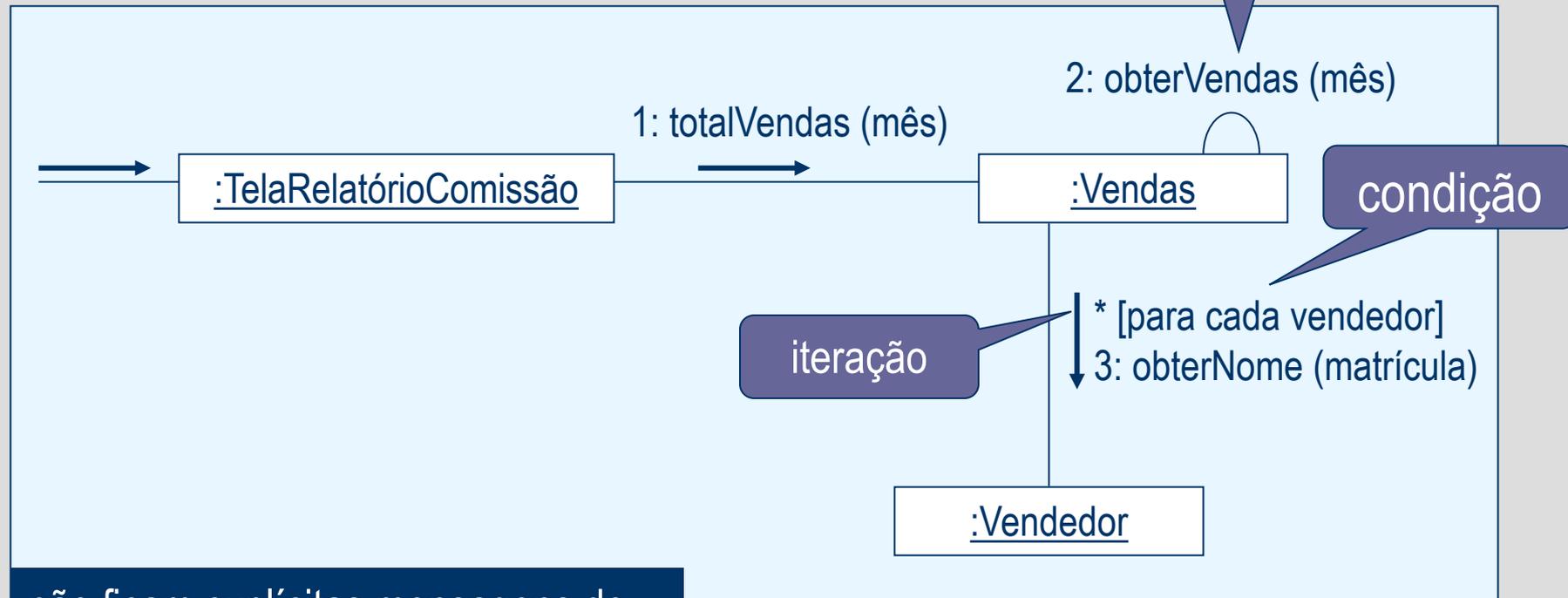




Cenário alternativo

# Diagrama de Comunicação

Para o exemplo inicial:



não ficam explícitas mensagens de retorno, nem quando isso ocorre