

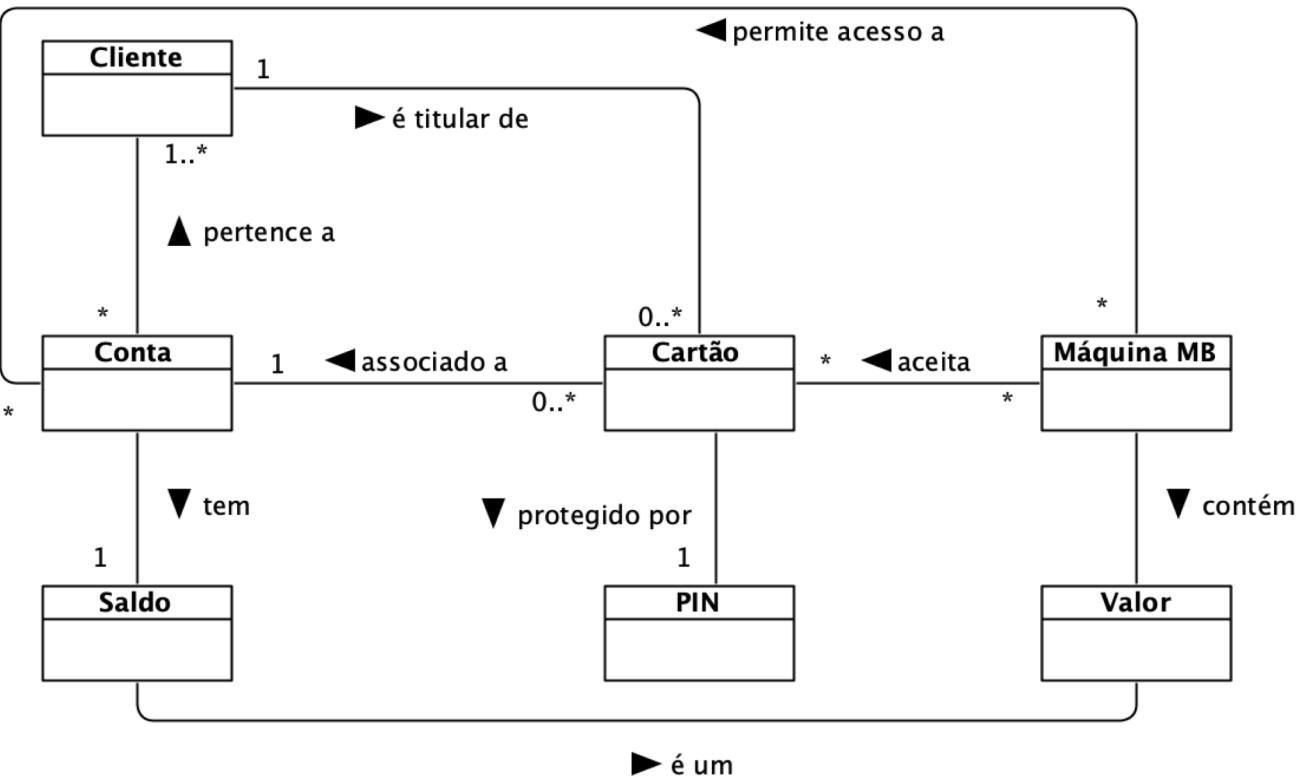


Desenvolvimento de Sistemas Software

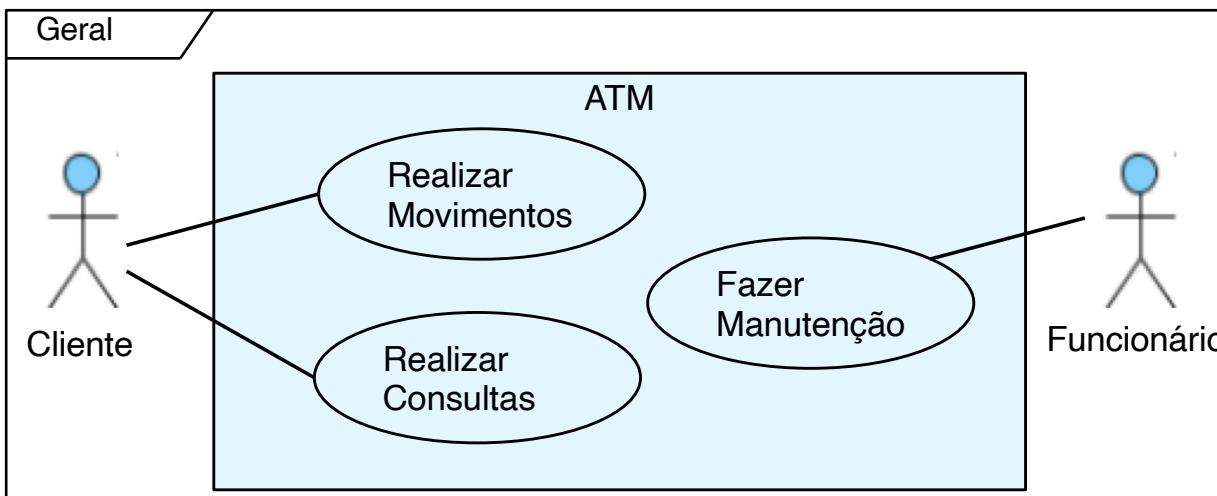
Aula Teórica 08

Diagramas de Sequência I

O que temos...

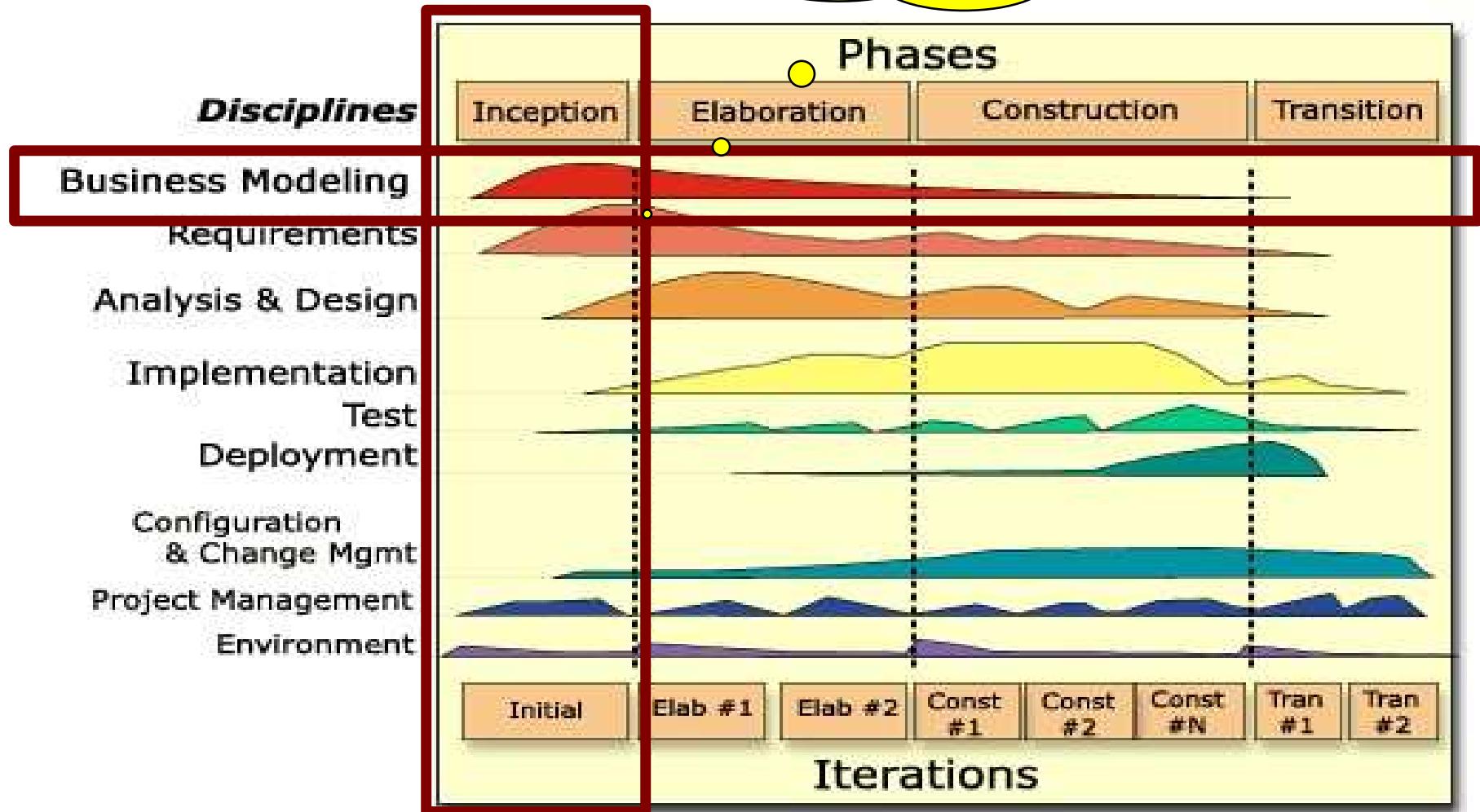


- **Use Case:** Levantar €
- **Descrição:** Cliente levanta quantia da máquina
- **Cenários:** João levanta €60 com cartão; João levanta €10 com MB way
- **Pré-condição:** Sistema tem notas
- **Pós-condição:** Cliente tem quantia desejada e saldo da conta foi actualizado
- **Fluxo normal:**
 1. Cliente apresenta cartão e PIN
 2. Máquina MB valida acesso e pede operação
 3. Cliente indica que pretende levantar dada quantia
 4. Máquina MB pergunta se quer talão
 5. Cliente responde que não
 6. Máquina MB devolve cartão e fornece notas
 7. Cliente retira cartão e notas
- **Fluxo alternativo 1:** [Cliente quer talão] (passo 5)
 - 5.1. Cliente diz que sim
 - 5.2. Máquina MB devolve cartão, notas e talão
 - 5.3. Cliente retira cartão, notas e talão
- **Fluxo alternativo 3:** [Cliente autentica-se com MB way] (passo 1)
 - 5.1. Cliente prima escolhe acesso MB way
 - 5.2. Máquina MB pede Código MB way
 - 5.3. Cliente indica Código MB way
 - 5.4. Regressa a 2
- **Fluxo de exceção 1:** [PIN inválido] (passo 2)
 - 2.1. Máquina MB avisa sobre PIN inválido e fornece cartão
 - 2.2. Cliente retira cartão



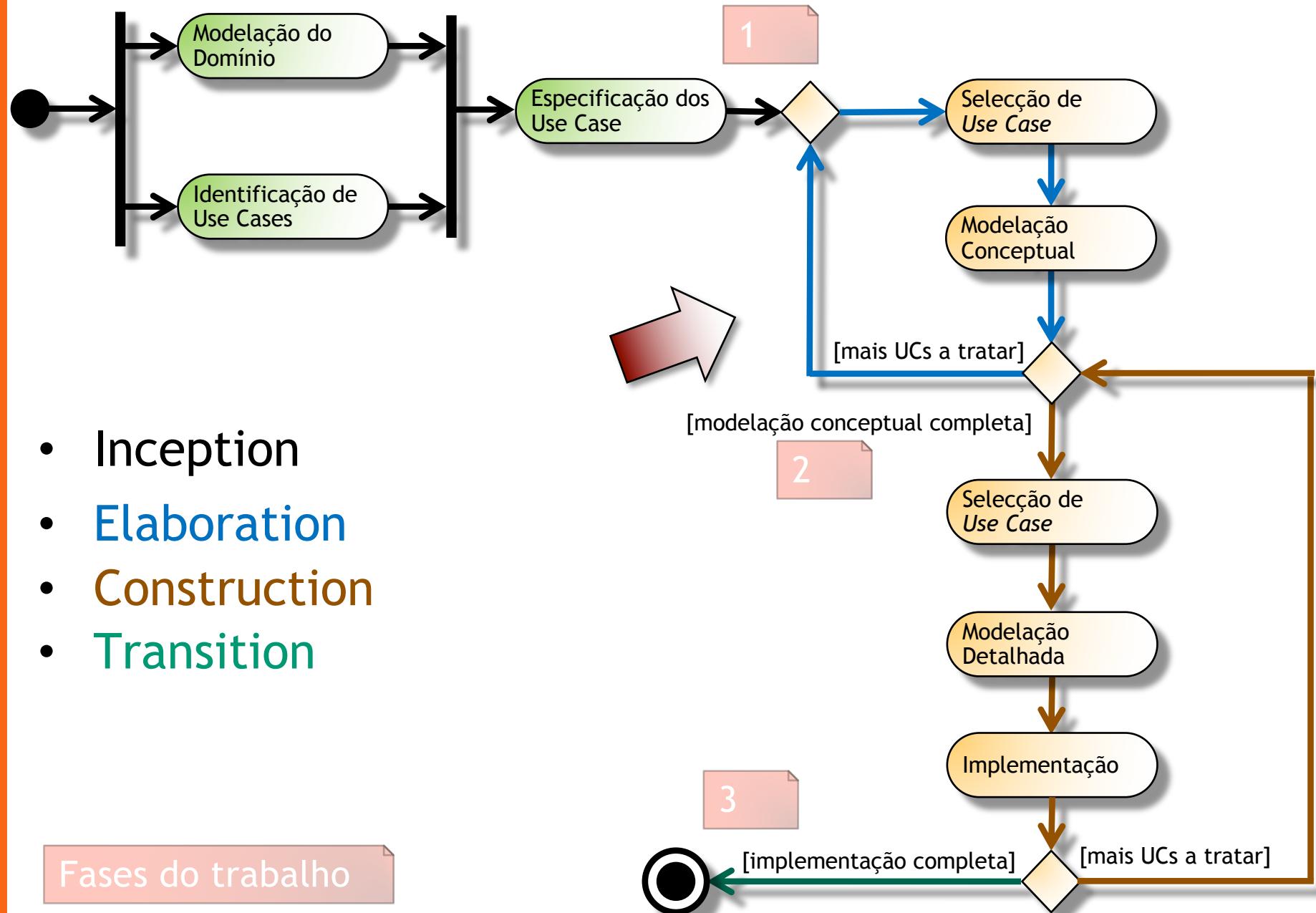
Ponto da situação...

- Modelo de Domínio
- Modelo de Use Case

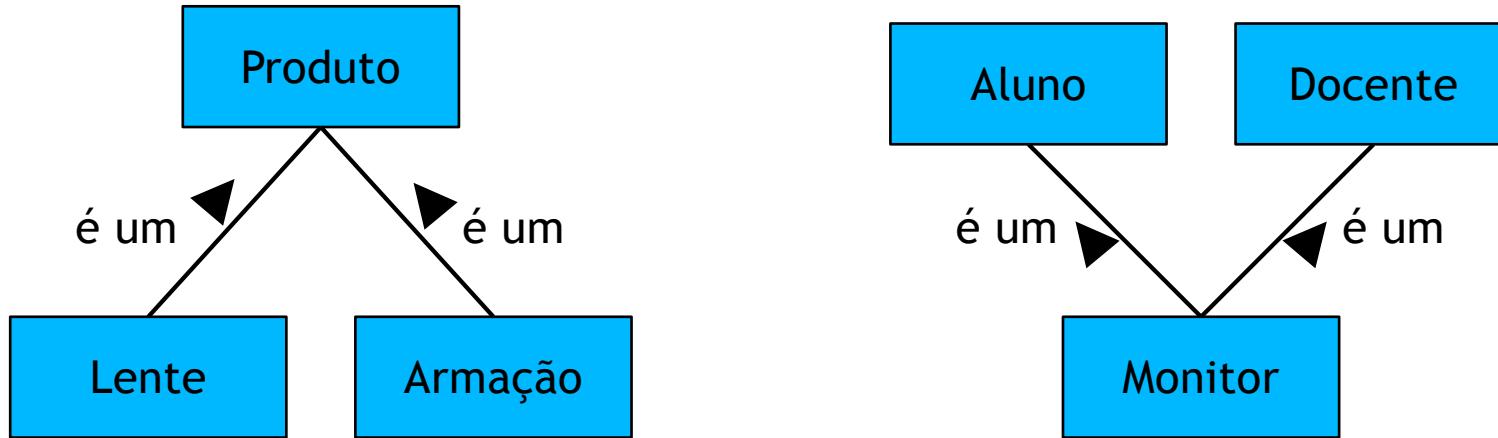


- guiado por casos de uso (use cases)
- centrado na arquitectura do sistema a desenvolver
- iterativo e incremental

Onde estamos...

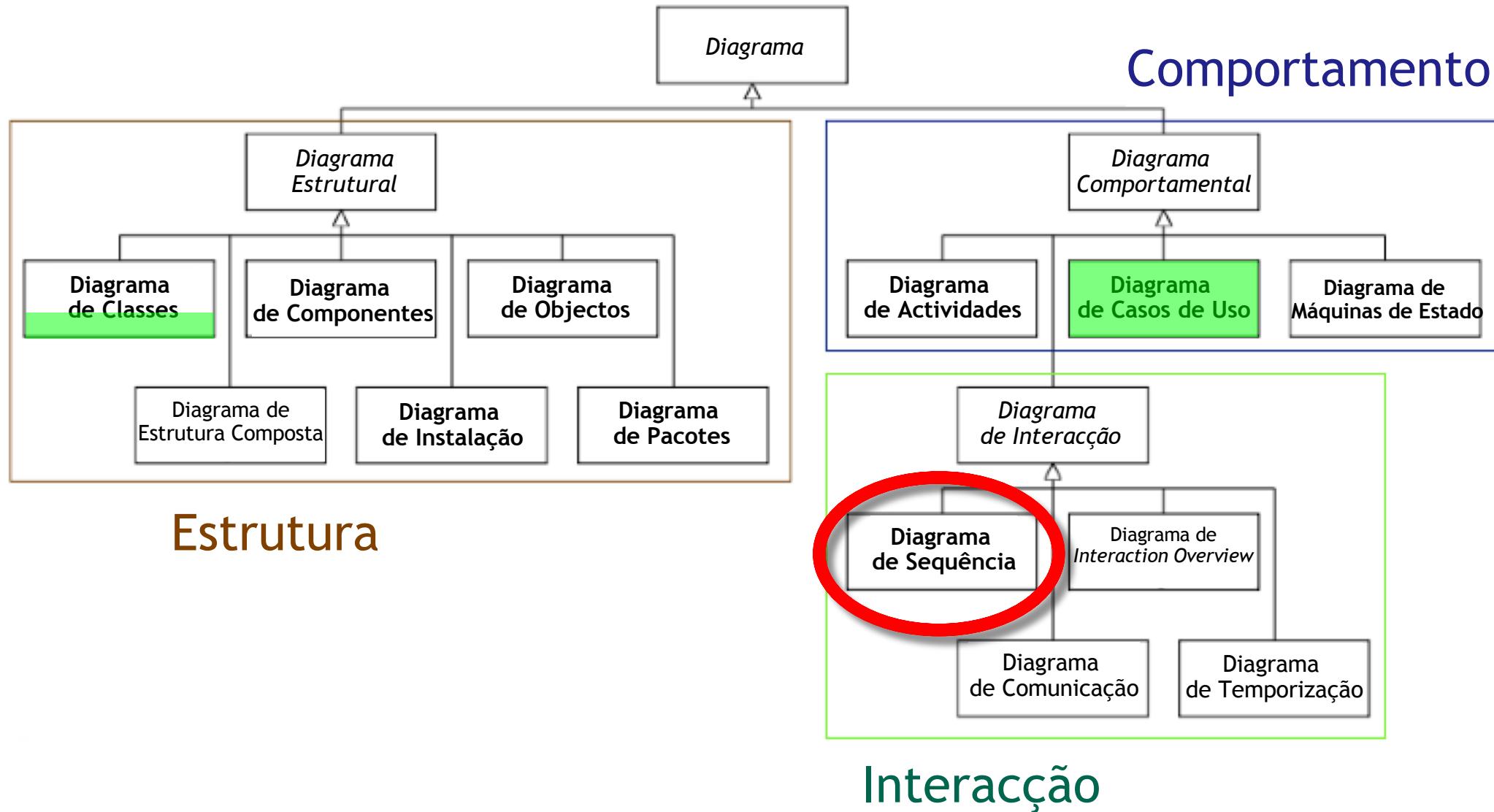


Arquitectura da Camada de Negócio?



- Demasiado cedo para tomar decisões
- É necessário considerar o comportamento

Diagramas da UML 2.x

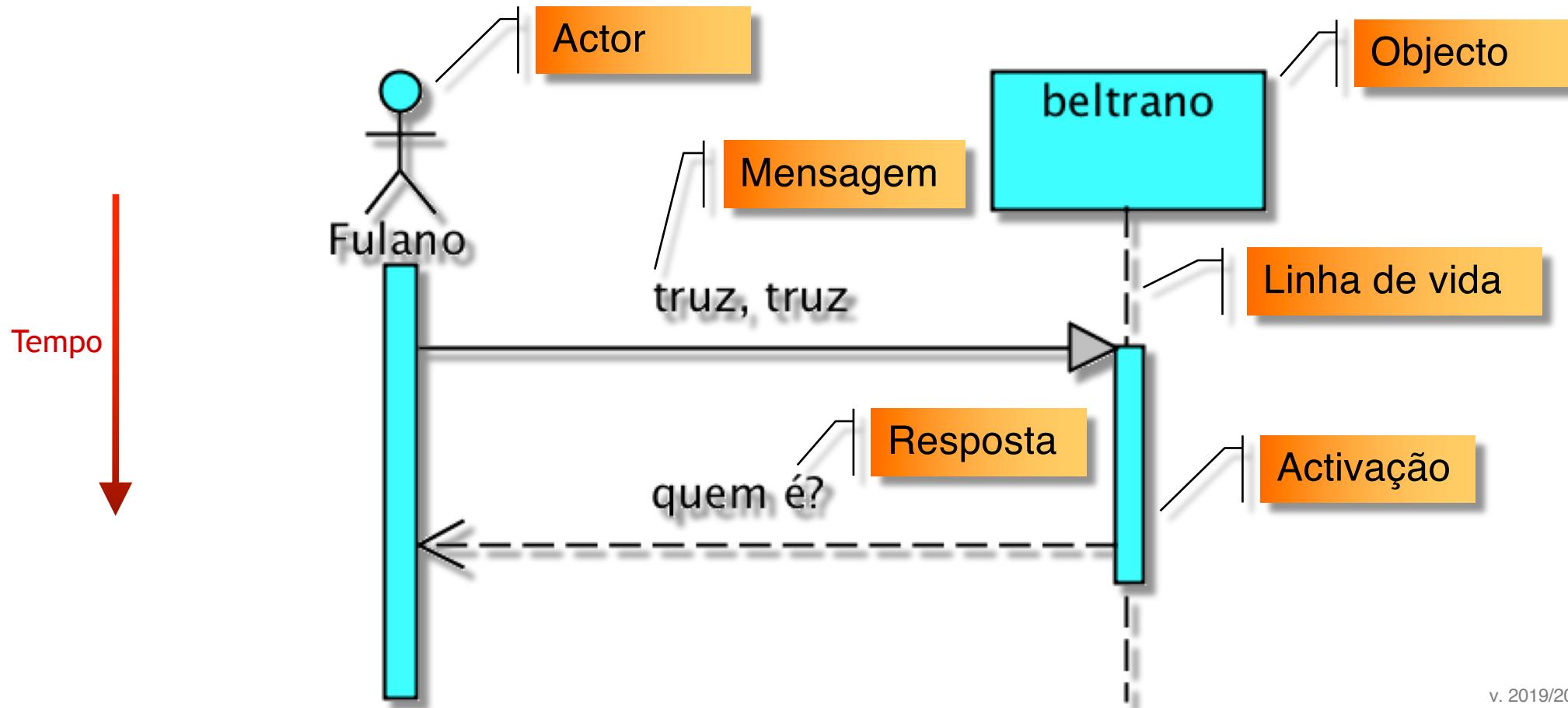


Diagramas de Interacção

- Um tipo de Diagrama Comportamental
- Descrevem como um conjunto de objectos coopera para realizar um dado comportamento
 - modelam as interacções entre os objectos para atingir um objectivo (p.e. realizar um *Use Case*)
- **Diagramas de sequência** 
 - foco no ordenamento temporal das trocas de mensagens
- **Diagramas de comunicação**
 - foco na arquitectura
- **Diagramas de Temporização (*Timing Diagrams*)**
 - foco nos aspectos temporais
- **Diagramas de *Interaction Overview***
 - visão de alto nível que combina os anteriores

Diagramas de Sequência - notação essencial

- representam as interacções entre objectos através das mensagens que são trocadas entre eles
- a ênfase é colocada na ordenação temporal das mensagens
- permitem analisar a distribuição de “responsabilidade” pelas diferentes entidades (analisar onde está a ser efectuado o processamento)



Diagramas de Sequência - notação essencial

Objectos

Um objecto
“beltrano” de tipo
desconhecido

Um objecto
anónimo da
classe Beltrano

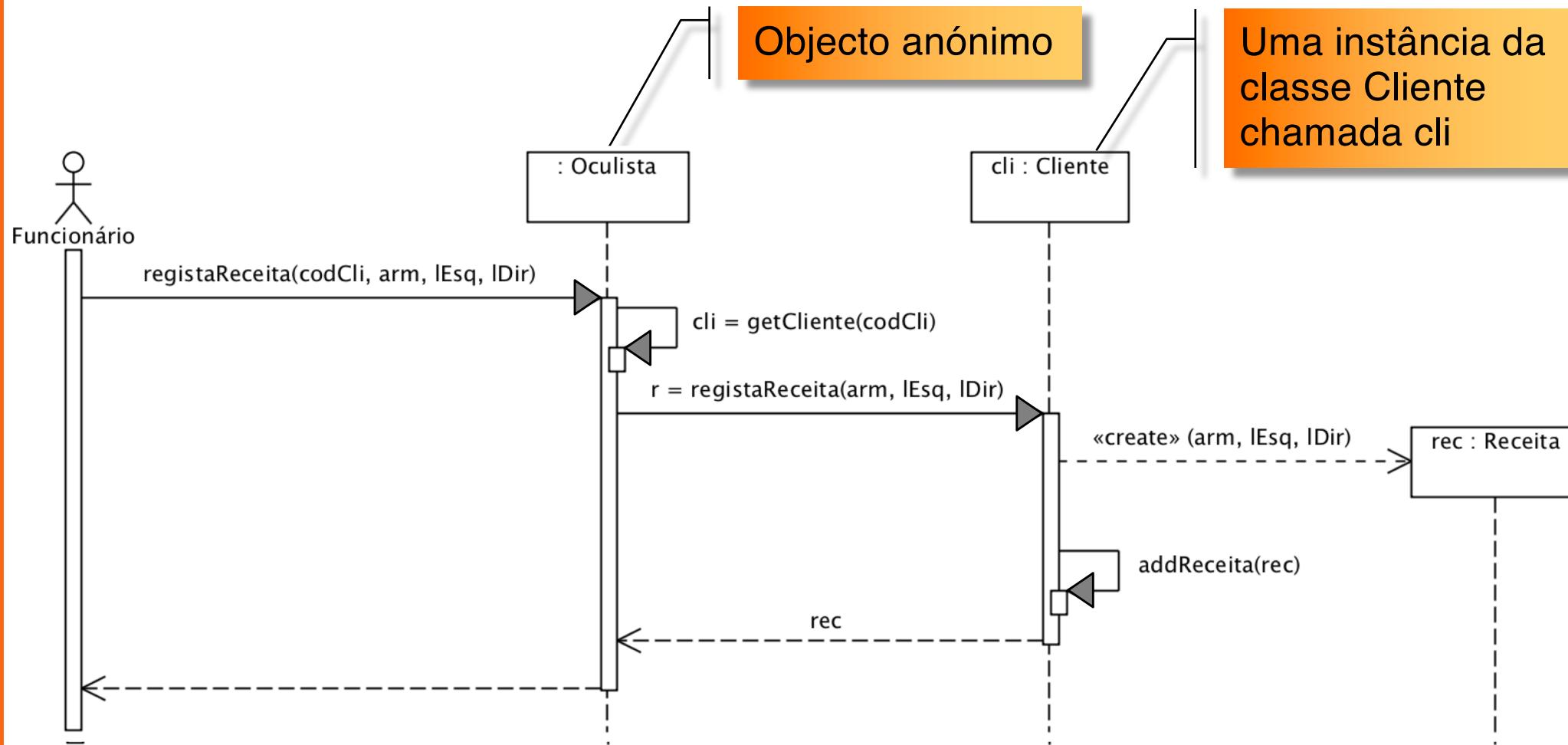
Um objecto
“beltrano” da
classe Beltrano

Uma colecção de
objectos da classe
Beltrano com
nome “grupo”.



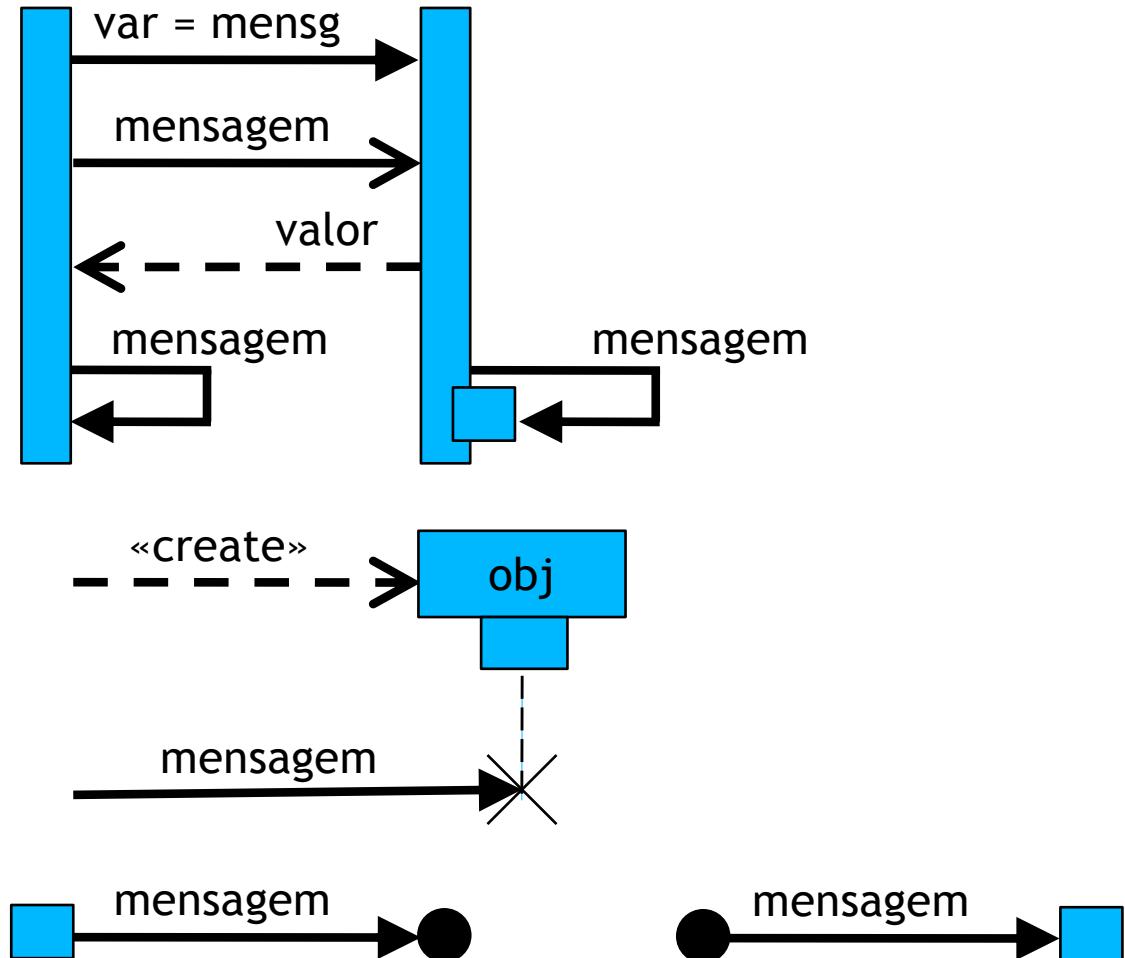
nome_objecto “[“selector”]” : classe

Diagramas de Sequência - notação essencial



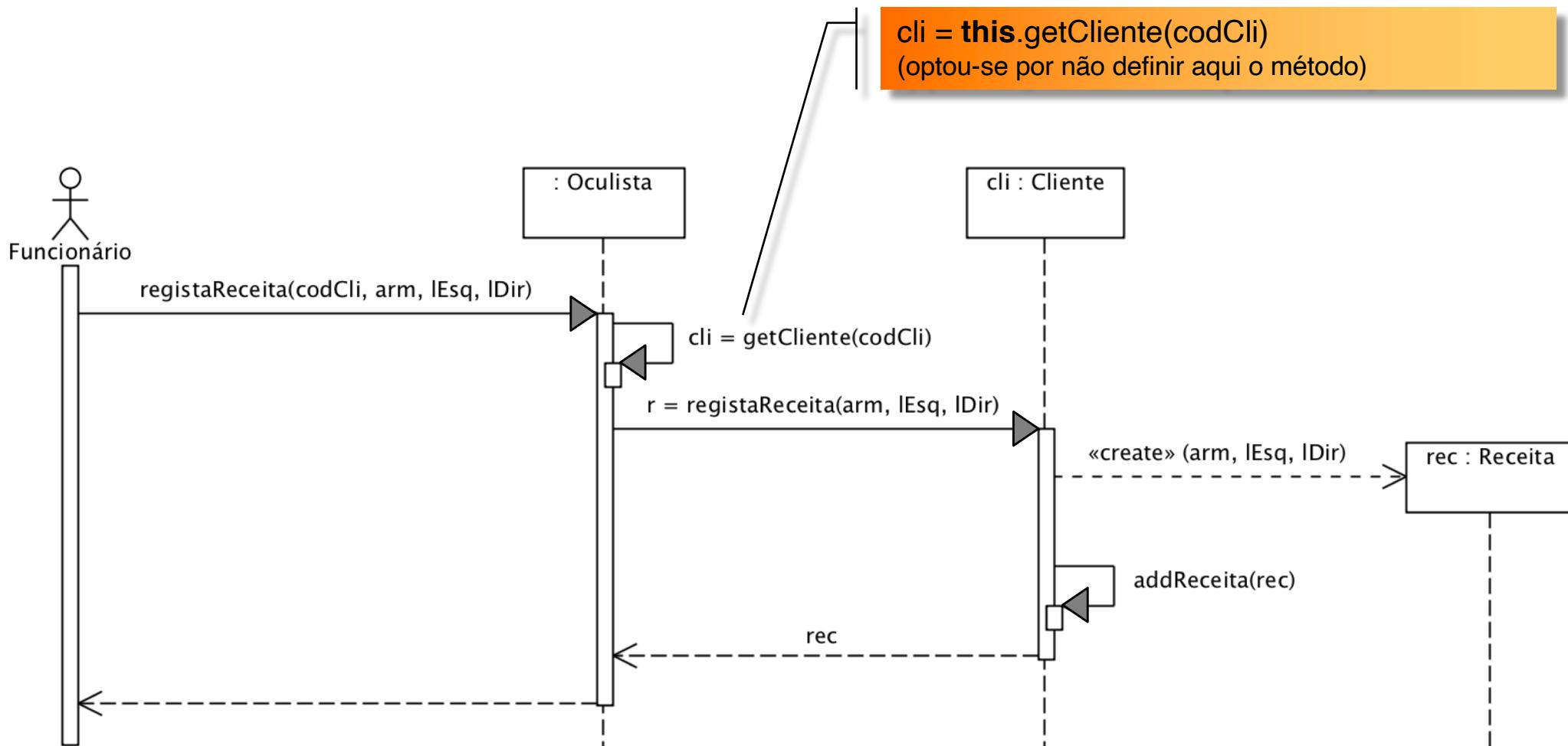
Mensagens

- invocação síncrona
- invocação assíncrona
- return/resultado
- self messages
- criar objectos
- destruir objectos
- lost/found messages



[atributo '='] nome_da_operação_sinal [argumentos] [':' tipo_resultado]

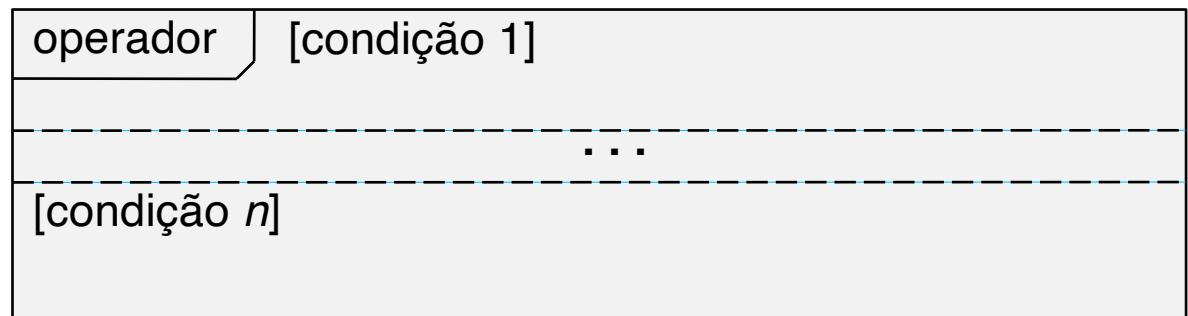
Diagramas de Sequência - notação essencial



- Todas as invocações são síncronas.
- Dois dos métodos não estão aqui definidos (+ construtor).
- **Atenção!** O objecto que envia a mensagem tem que “conhecer” o objecto a quem a envia.

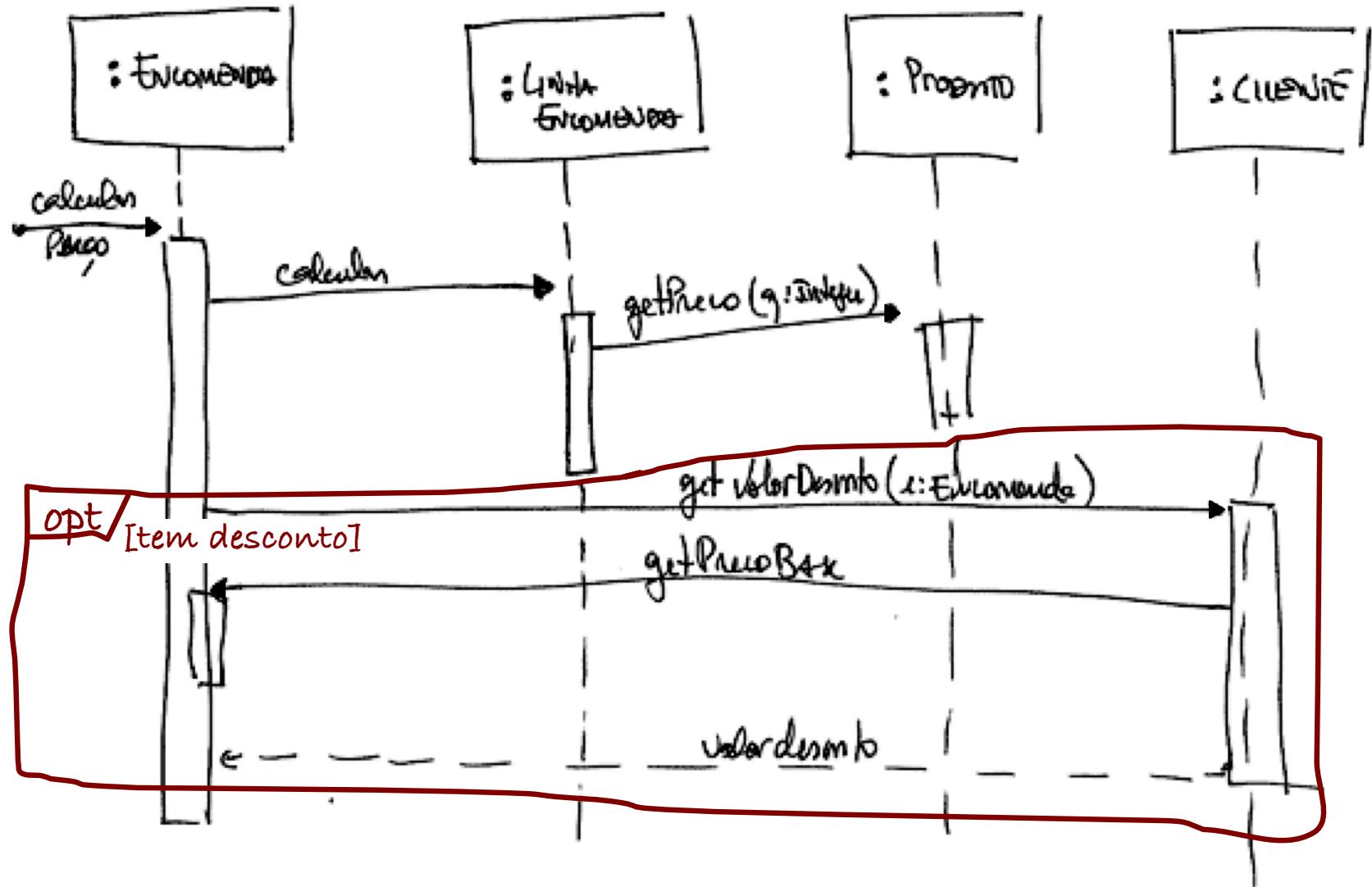
Diagramas de Sequência - fragmentos combinados

- Um fragmento combinado agrupa conjuntos de mensagens
- Permitem expressar fluxos condicionais e estruturar os modelos



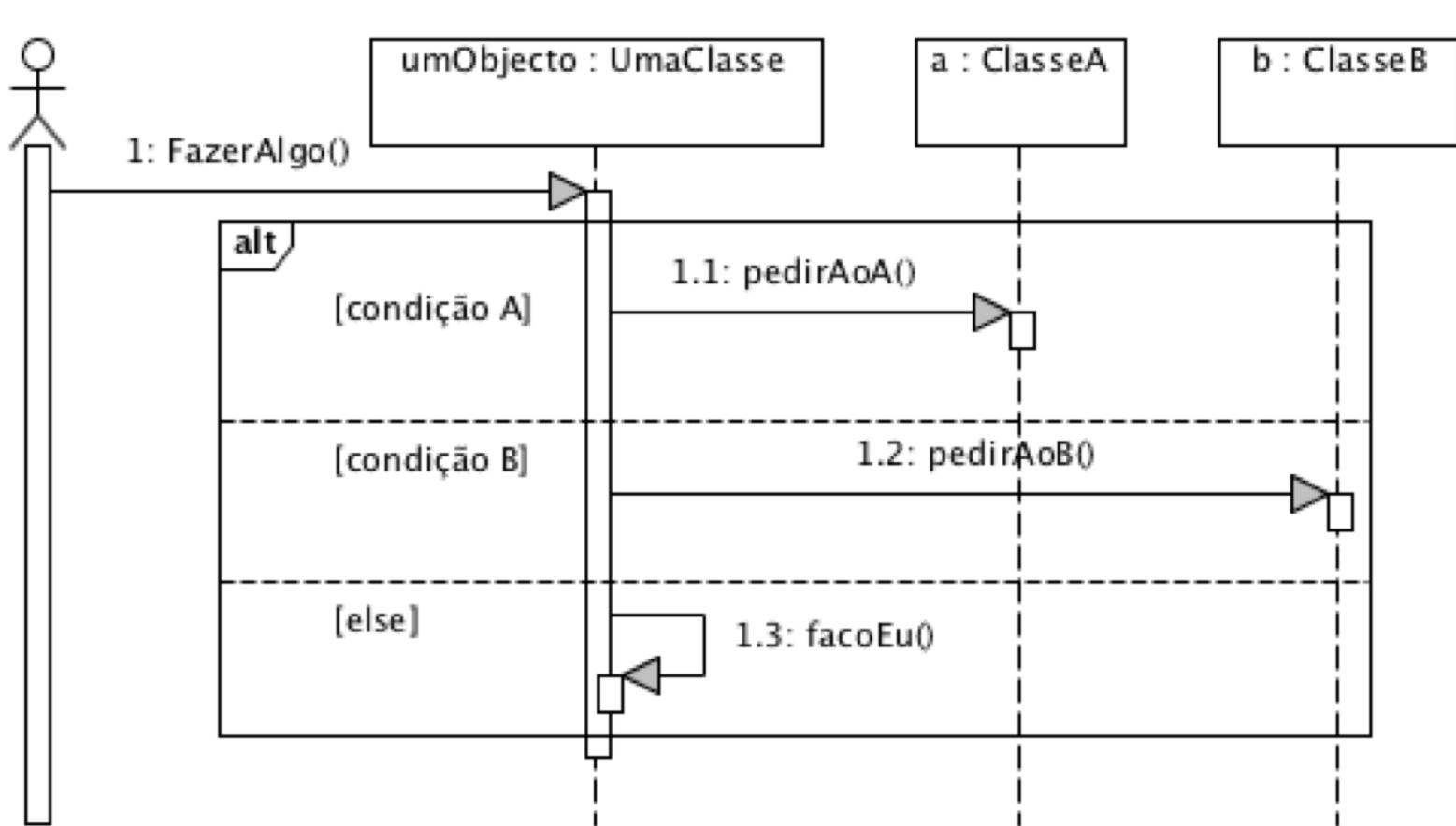
- Operadores mais comuns
 - **alt** - define fragmentos alternativos (mutuamente exclusivos)
 - **loop / loop(*n*)** - fragmento é repetido enquanto a guarda for verdadeira / *n* vezes
 - **opt** - fragmento opcional (ocorre se a guarda for verdadeira)
 - **par** - fragmentos ocorrem em paralelo
 - **break** - termina o fluxo
 - **ref** - referência a outro diagrama

Operador *opt*



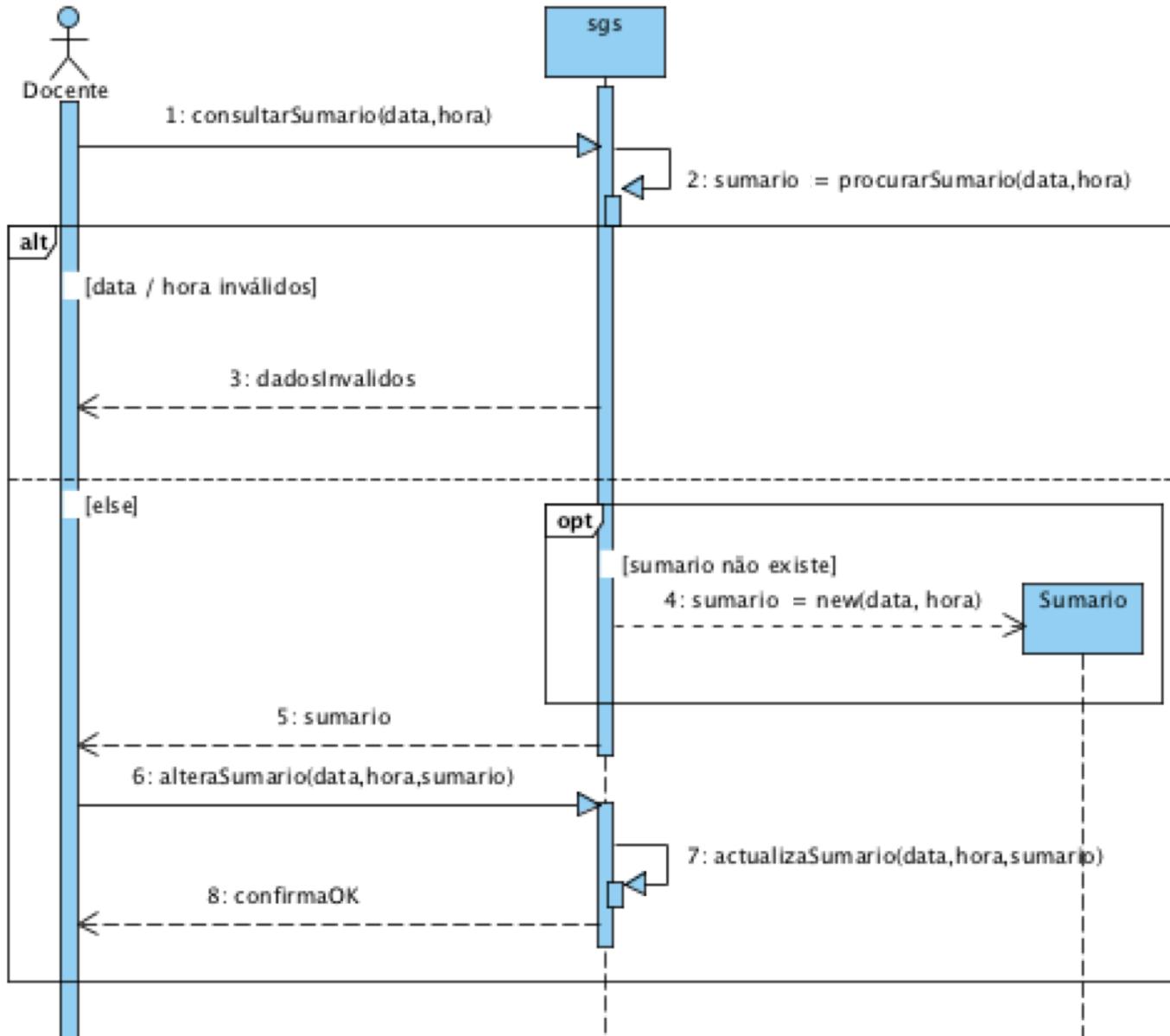
Cálculo do desconto só é efectuado se a guarda **tem desconto** se verificar.

Operador alt



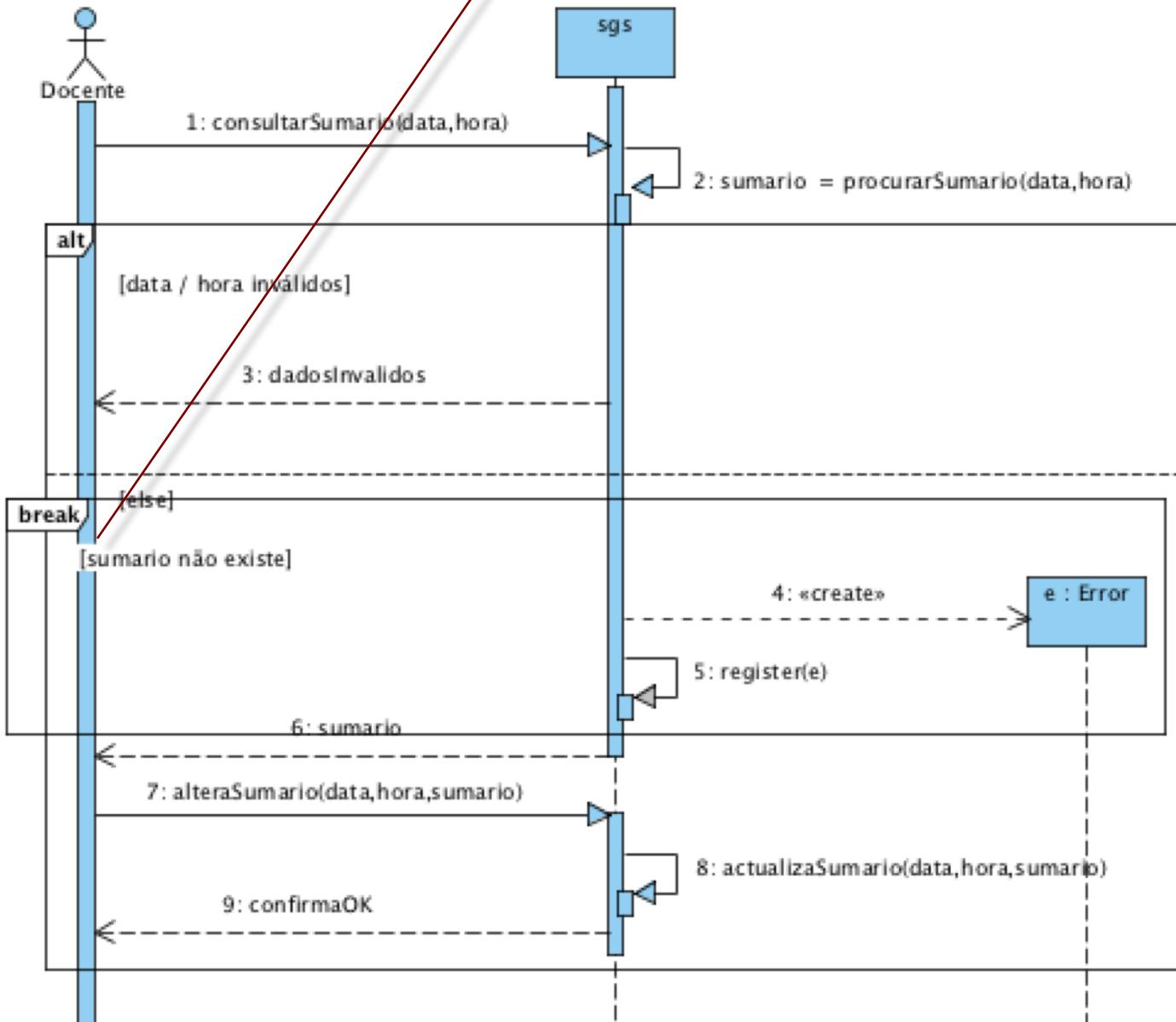
- Os fluxos possíveis são mutuamente exclusivos, pelo que apenas um deles será seguido.
- Se mais que uma condição se verificar, não está definido qual acontece.

Um exemplo...

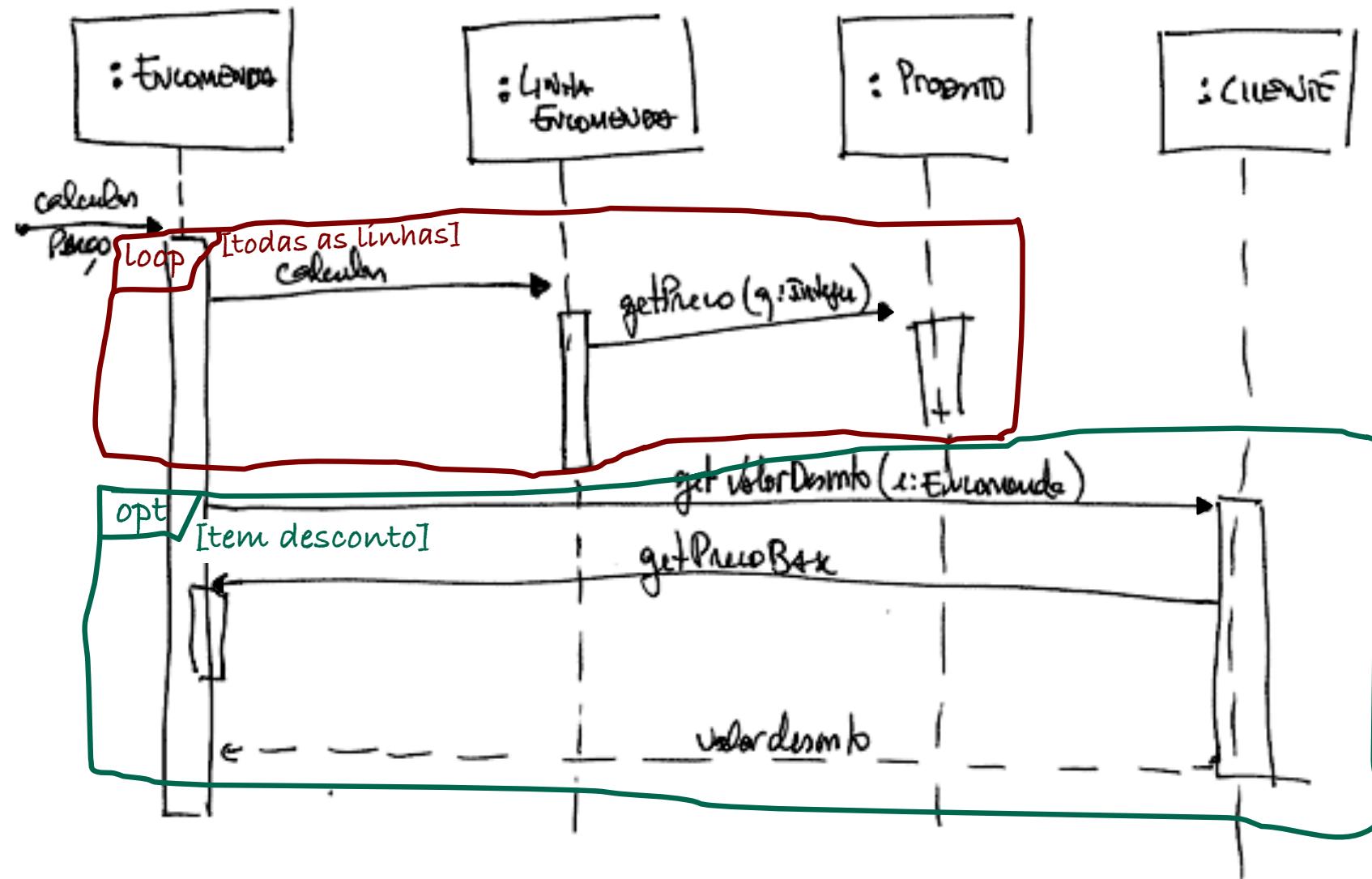


Operador break

Se não existe, regista erro e termina.

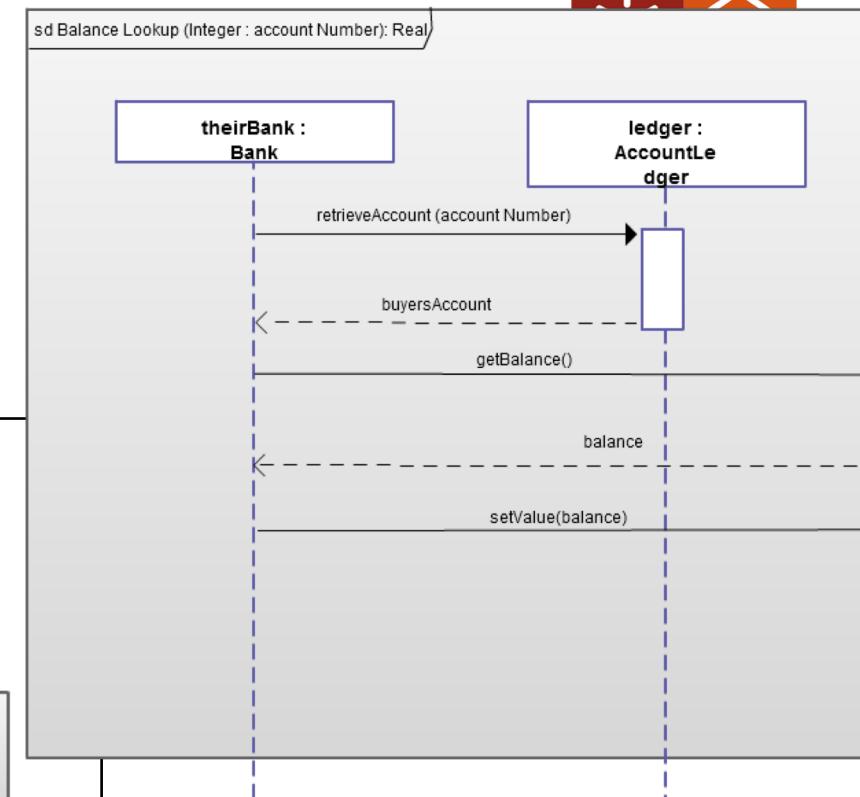
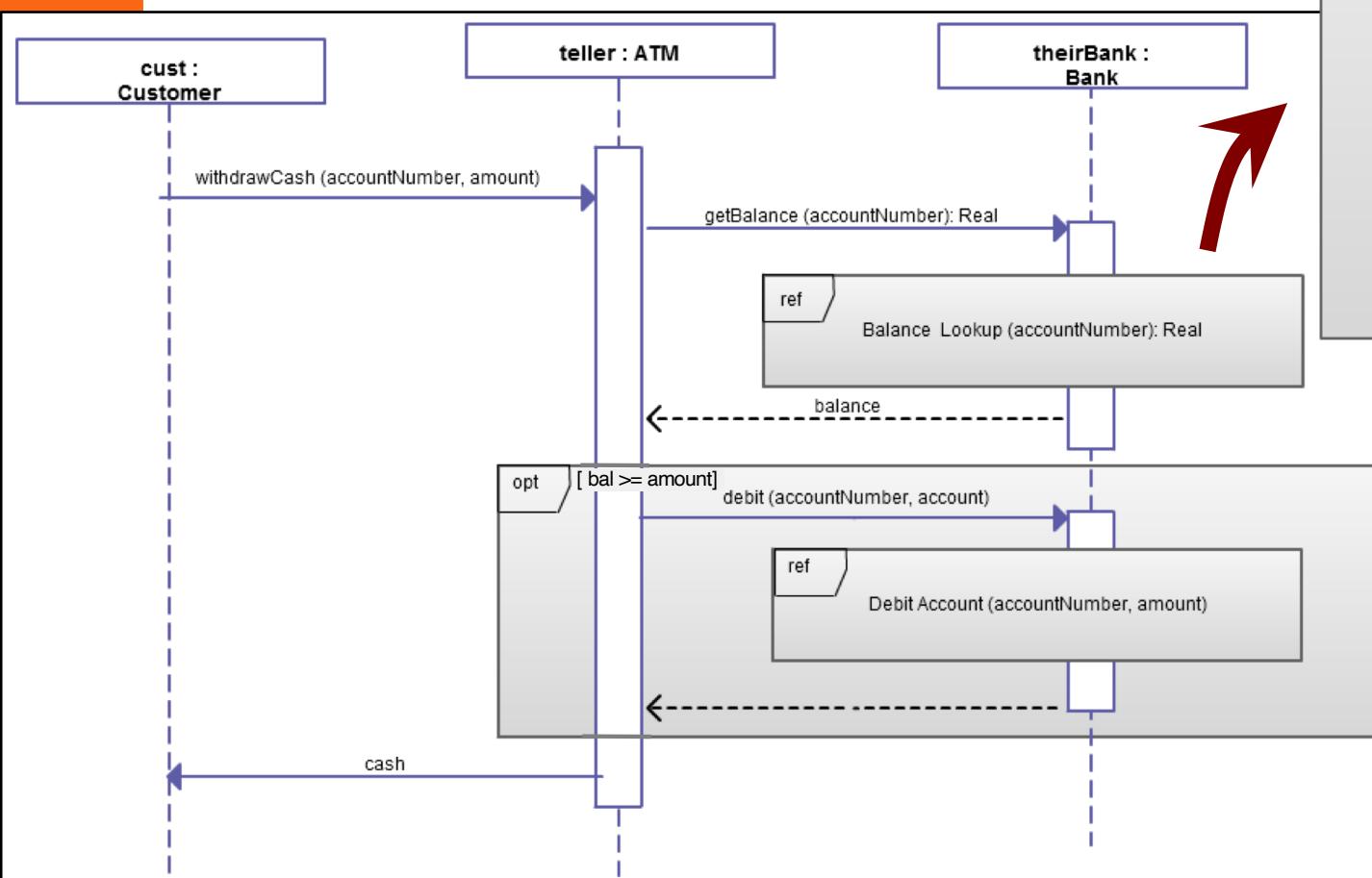


Operador *loop*



Cálculo do preço é efectuado para todas as linhas da encomenda.

Operador ref

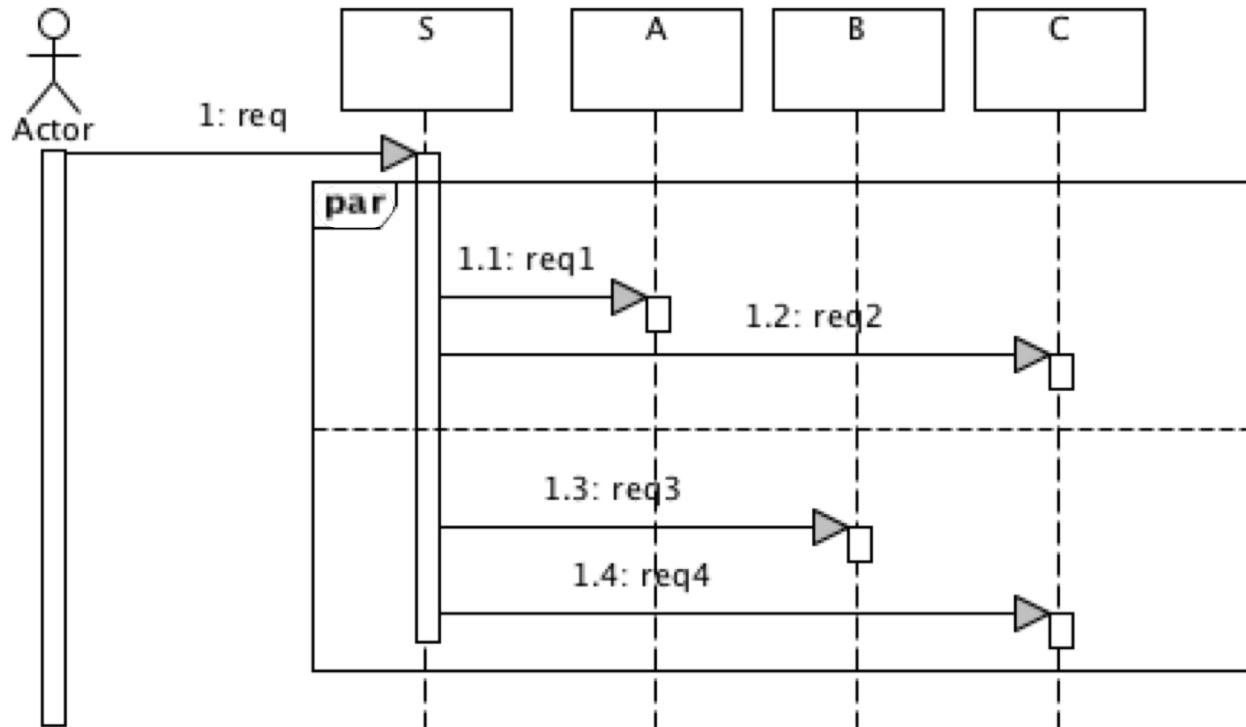


- Um SD pode reutilizar outros SD referenciando-os num fragmento com o operador **ref** – permite estruturar os modelos

Outros operadores

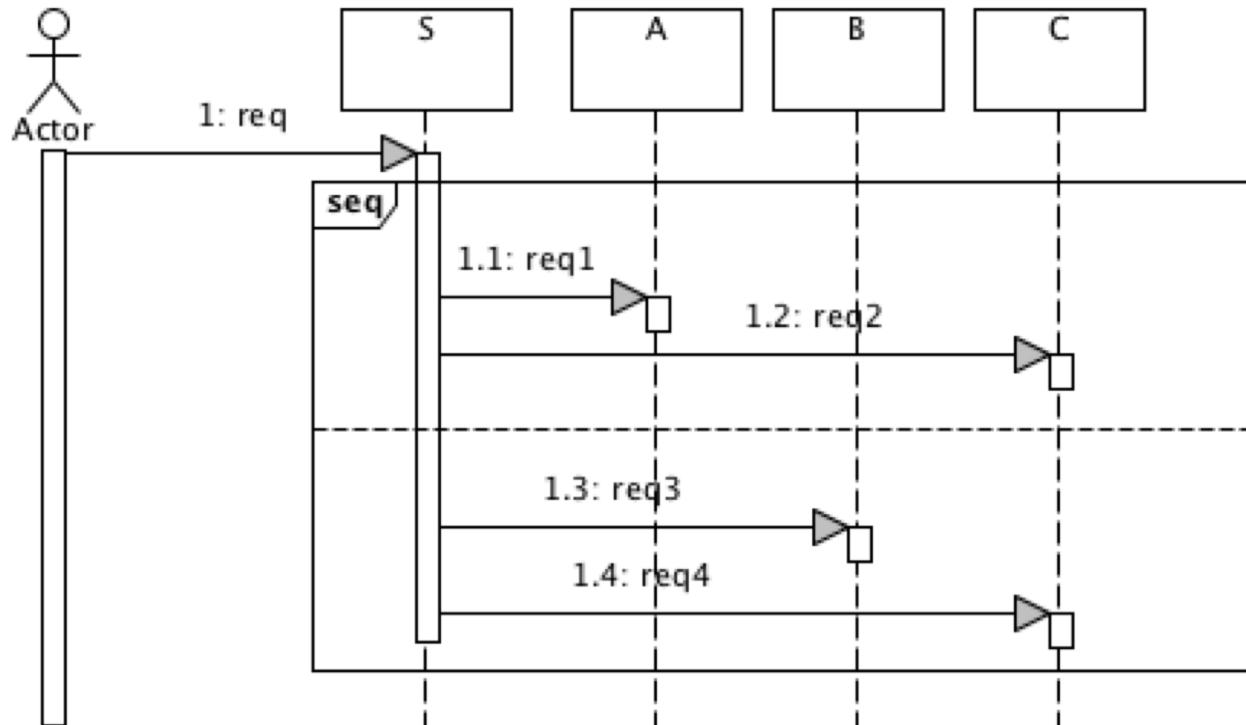
- **critical** - o operando executa de forma atómica
- **par** - os operandos executam em paralelo
- **seq** (sequenciação fraca) - todos os operandos executam em paralelo, mas eventos enviados a uma mesma linha de vida acontecem na mesma sequência dos operandos
- **strict** - os operandos executam em sequência
- **neg** - negação, o operando mostra uma interacção inválida
- **assert** - mostra o único comportamento válido naquele ponto
- **ignore** - indica mensagens intencionalmente omitidas da interacção
 - ignore {m1, m2, ...} - m1, m2 podem acontecer mas não são mostradas
- **consider** - indica mensagens intencionalmente incluídas na interacção (dual de ignore)
 - consider {m1, m2, ...} - outras mensagens, para além de m1, m2, podem acontecer mas não são mostradas

Operador *par*



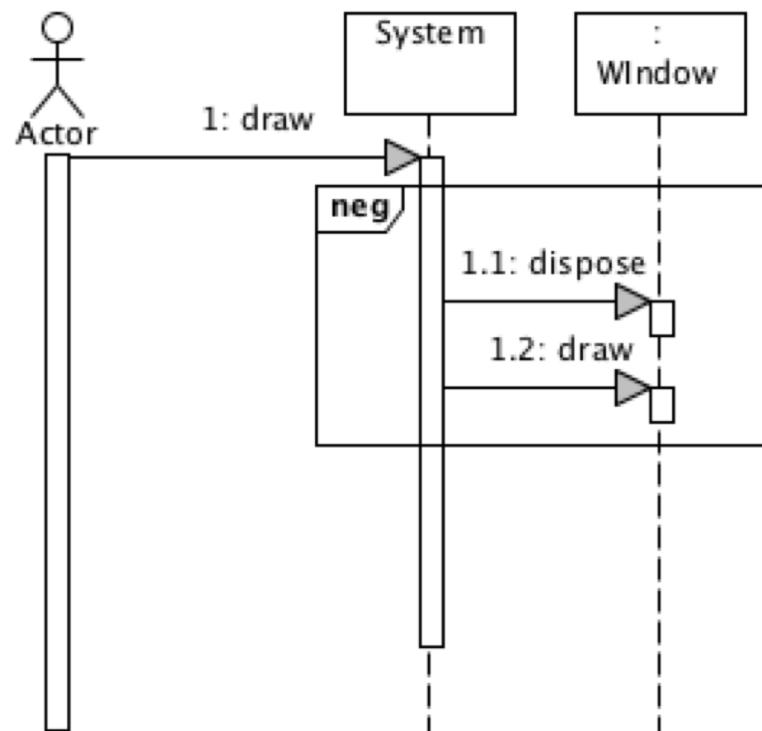
Eventos *req1* e *req2* podem acontecer em paralelo com eventos *req3* e *req4*. Nenhuma ordem é imposta.

Operador seq



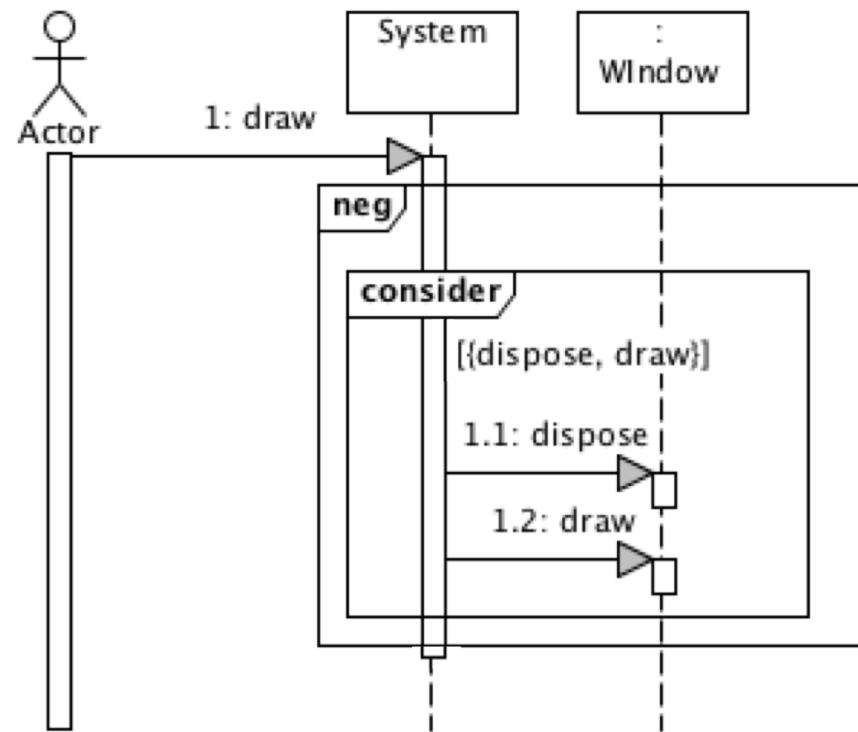
Eventos *req1* e *req3* podem acontecer em paralelo. Evento *req2* acontece antes de evento *req4* (porque ambos vão para C).

Operadro neg



Não é válido desenhar numa janela depois de ela ter sido removida.

Operador consider



Porque podem existir outros eventos pelo meio...



Diagramas de Sequência

Sumário

- Necessidade de modelação comportamental
- Diagramas de Sequência
 - Enquadramento
 - Notação base
 - Notação para representação de objectos
 - Notação para representação de mensagens
 - Fragmentos e operadores