

# Tópico 6:

## Diagrama de Sequência

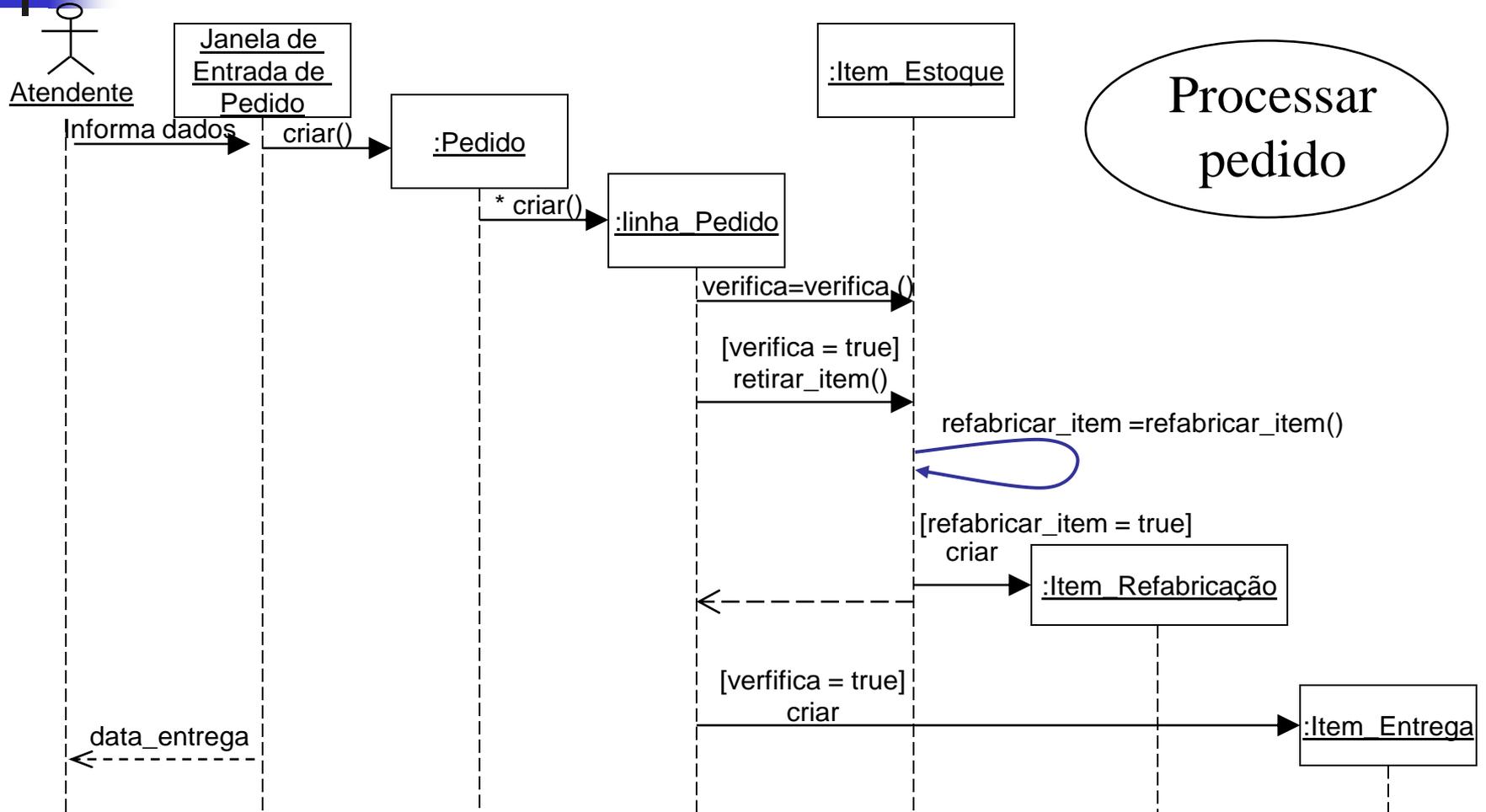


Luiz Antônio M. Pereira

[lpereira@uninet.com.br](mailto:lpereira@uninet.com.br)



# Diagrama de Sequência





# Diagramas de Sequência

- Descrevem como grupos de objetos colaboram em algum comportamento do sistema;
- Tipicamente cada diagrama captura o comportamento de um único cenário de um caso de uso.



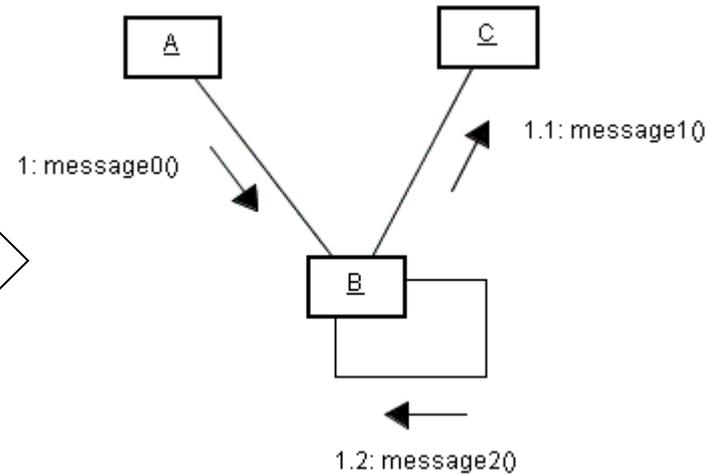
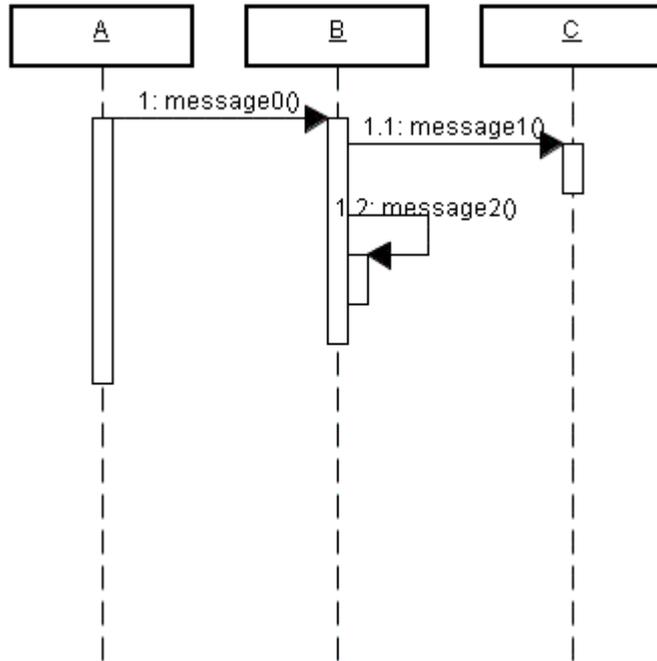
# Diagramas de Sequência

- Usados para a atribuição de responsabilidades aos objetos do sistema;
- Úteis para a descoberta de operações dos objetos e para a modelagem da interação entre eles (DS e de Comunicação são 2 dos diagramas de interação);
- Servem, portanto, para modelar o “funcionamento” do sistema;



# Diagramas de Sequência

- Melhores que o Diagrama de Comunicação para apresentar as responsabilidades de cada objeto, especialmente quando o aspecto da ordenação temporal é relevante.





# Diagramas de Sequência

- Completam o tripé da análise:
    - Casos de Uso - comportamento externo (funcional)
    - Diagramas de Classes - visão estática
    - Diagramas de Sequência - visão dinâmica
- } Internos

# Diagramas de Sequência

---

## **Definição de cenário:**

- Instância de um caso de uso;
- Um caminho único em um caso de uso;
- Um fluxo de informação;
- Um fluxo relacionado com o objetivo com começo, meio e fim;

# Diagramas de Sequência

## **Cenários podem ser:**

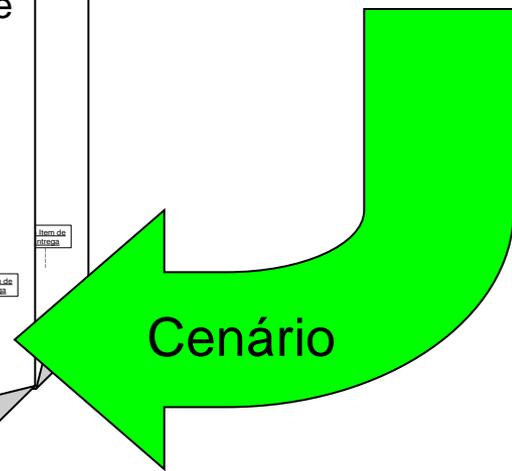
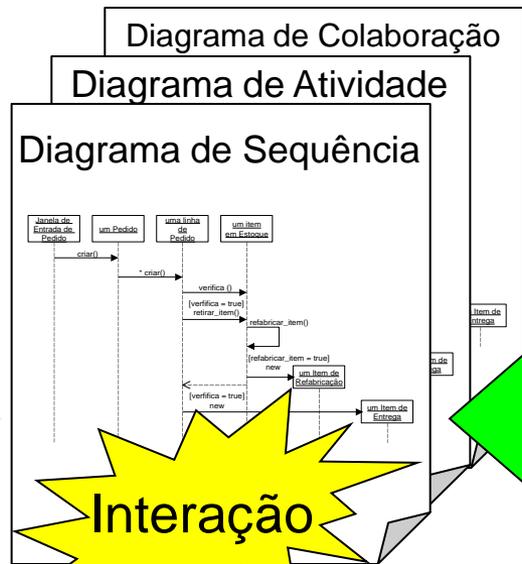
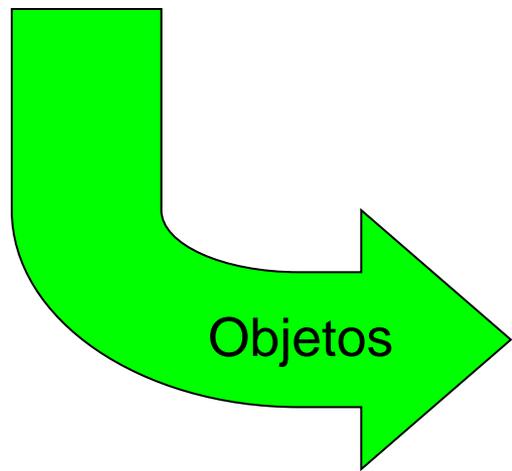
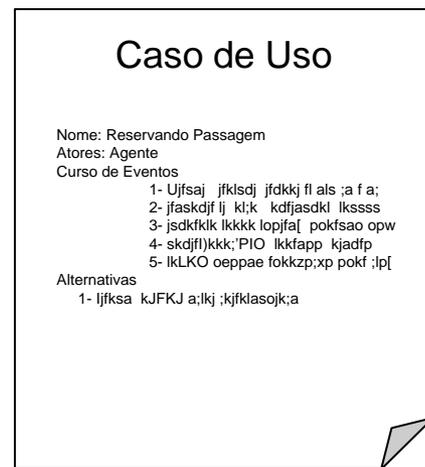
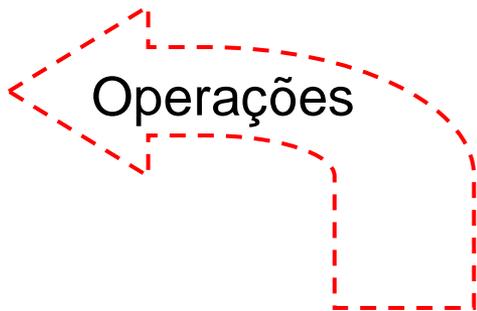
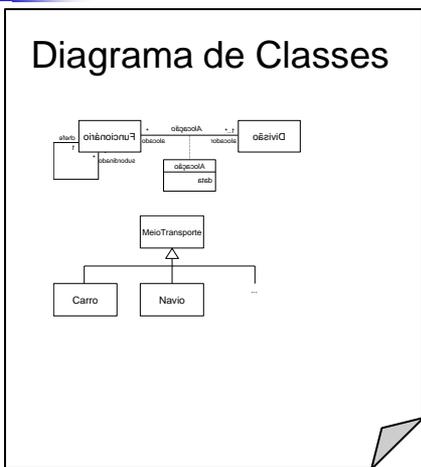
- OTIMISTAS : quando tudo dá certo
- EXCEÇÃO : erros e casos de falha
- ALTERNATIVOS : opções de sequência

# Diagramas de Sequência

Exemplos de cenários:

- Há disponibilidade de apartamentos e hóspede confirma reserva;
- Há disponibilidade de apartamentos, hóspede confirma reserva, mas operadora do cartão de crédito não confirma débito;
- Etc.

# Diagramas de Sequência





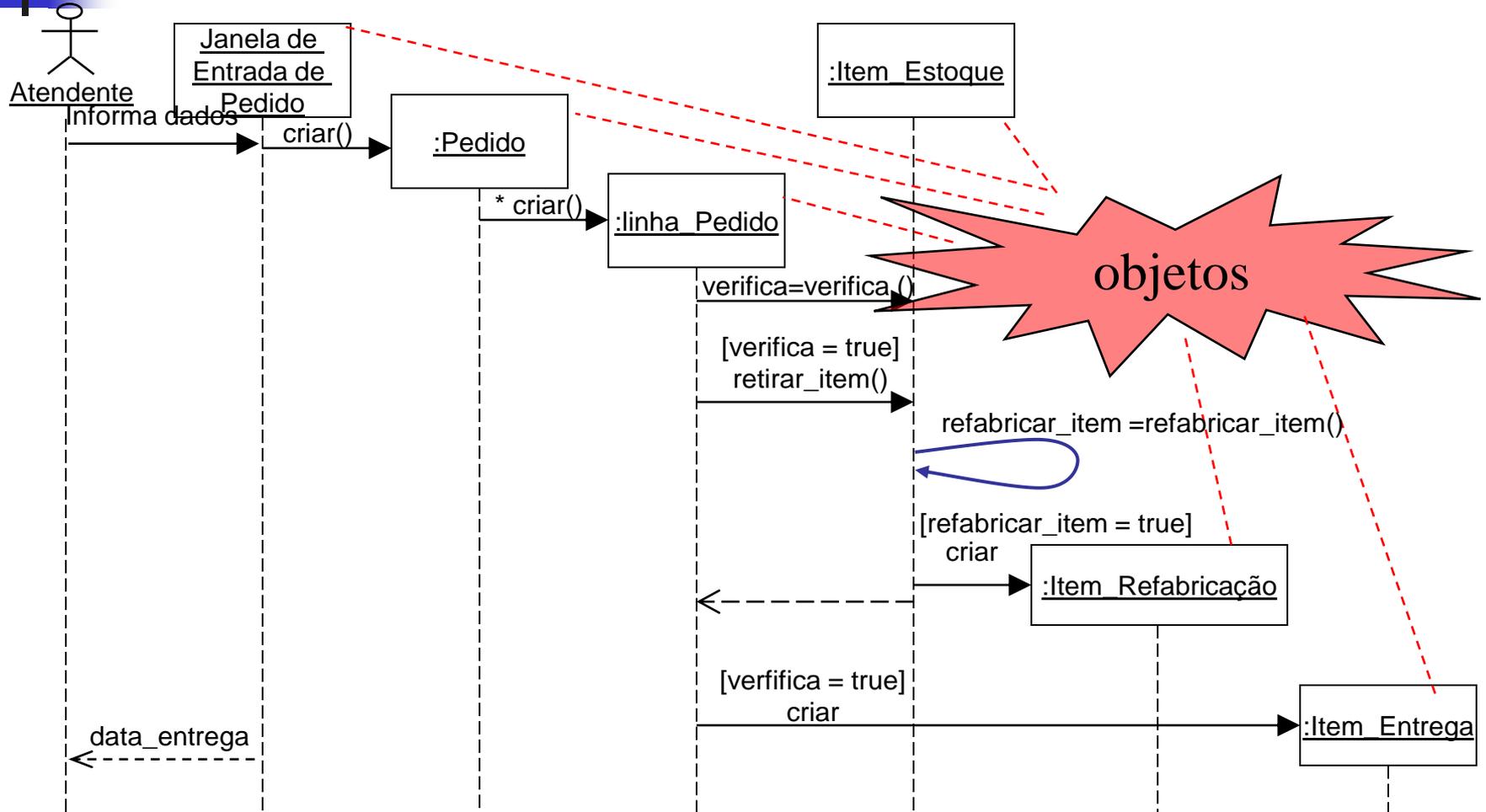
# Diagramas de Sequência

---

## Elementos



# Diagramas de Sequência



# Diagramas de Sequência

- Objetos compõem a *dimensão horizontal* ( $\rightarrow$ ) ;
- Ordem da colocação dos objetos na dimensão horizontal não tem significado;

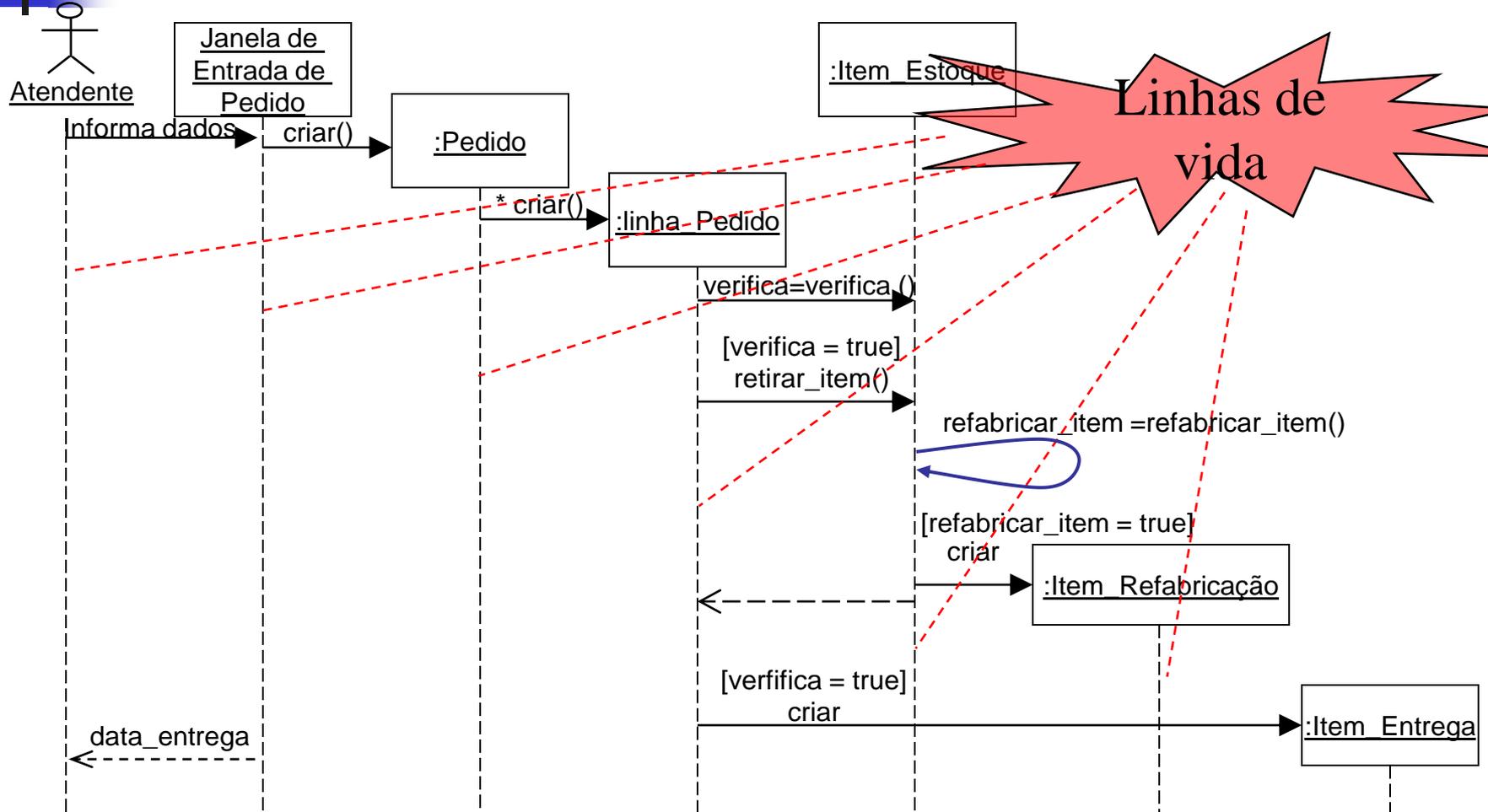


# Diagramas de Sequência

- Objetos possuem nome: obj.classe
  - O sublinhado denota instanciação (objeto = instância de classe) sendo, portanto, obrigatório;
  - *Obj* OU (exclusivo) *classe* pode ser omitido;
  - Tipicamente: um(a)Classe.
  - Ex.: umPedido, umItemPedido, umFuncionário,  
:Funcionário, :Pedido, Joaquim:Funcionário, etc.



# Diagramas de Sequência

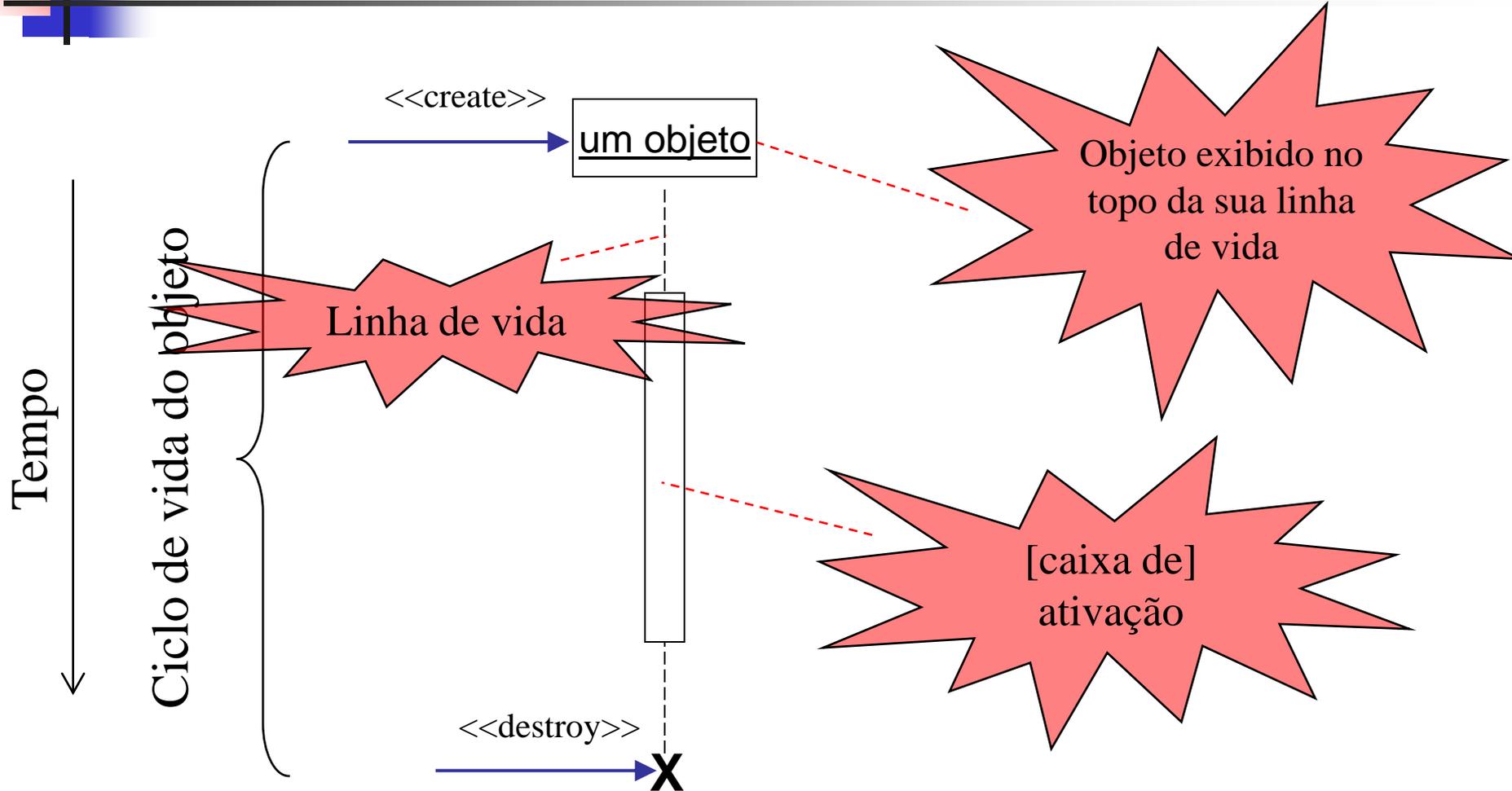




# Diagramas de Sequência

- Linhas de vida compõem a dimensão vertical (tempo);
- Dimensão vertical = sequência ...
- ... onde a vida do objeto durante a interação representada;
- Pode apresentar a ativação e a desativação de objetos (foco de controle);
- Pode representar criação de objetos ...
- ... e a destruição de objetos

# Diagramas de Sequência



# Diagramas de Sequência

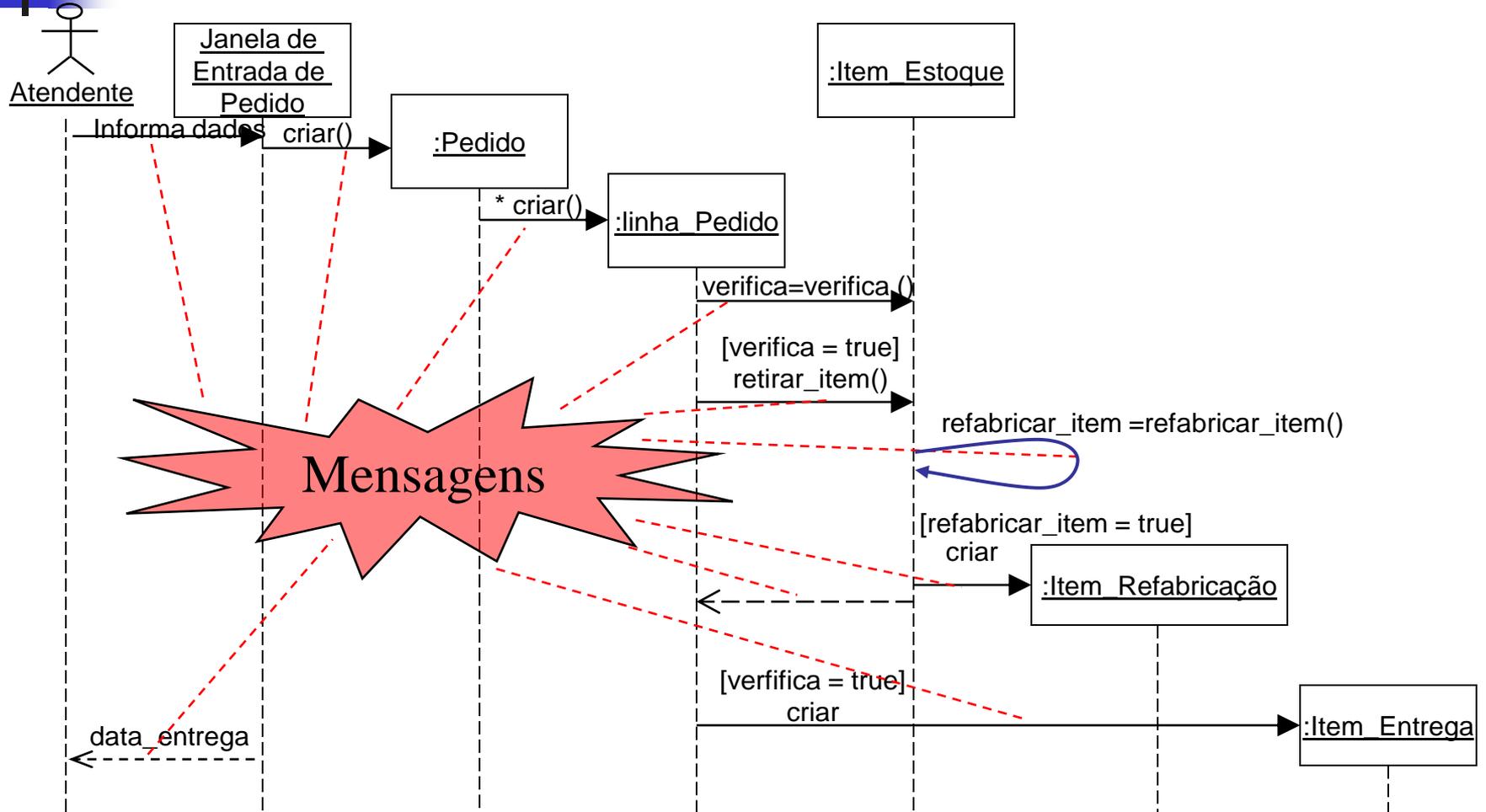
- Linhas de vida podem se subdividir em duas ou mais linhas de vida concorrentes para mostrar condicionalidades (cenários diferentes em um mesmo caso de uso). As linhas podem se unir novamente mais adiante. O diagrama pode, com isso, ficar confuso  $\Rightarrow$  usar mais de um diagrama (um para cada cenário).

# Diagramas de Sequência

- Caixas de ativação denotam que o objeto está executando algo (foco de controle) diretamente ou através de outro objeto;
- Caixas de ativação são opcionais (diagramas ficam mais fáceis de se desenhar), mas a ausência dificulta a leitura do diagrama;
- A atividade sendo executada pode ser descrita em texto na margem esquerda do diagrama., dentro dos limites da ativação;
- Caixas de ativação podem empilhar-se para denotar chamadas recursivas.



# Diagramas de Sequência





# Diagramas de Sequência

- Objetos interagem através de trocas de mensagens;
- Representadas por setas sólidas indo da linha de vida do objeto que envia para a linha de vida do objeto alvo (que pode ser o mesmo que envia: auto-chamada ou auto-delegação);
- As setas são rotuladas com (pelo menos) o nome do estímulo e com os valores ou expressões de seus argumentos. Podem conter, também, as condições em que as mensagens são enviadas entre “[” e “]”.



# Diagramas de Sequência

---

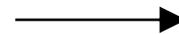
Mensagens:

- Podem corresponder a dois tipos de ações:
  - Chamada;
  - Retorno.

# Diagramas de Sequência

- Chamada:

- Invoca uma operação de um objeto (alvo). Um objeto pode enviar uma mensagem para si mesmo, resultando numa invocação local de uma operação. Notação:



- Retorno:

- Retorna um valor para o chamador. Retornos podem não ser representados, caso não se esteja lidando com concorrência. Notação:



# Diagramas de Sequência

## ■ Chamada de Criação

- Cria um objeto. Notações:  $\xrightarrow{\text{new()}}$   $\xrightarrow{\ll\text{create}\gg}$

## ■ Chamada de Destruição

- Destrói um objeto. Um objeto pode cometer suicídio, se auto-destruindo. Notações:

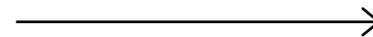
$\xrightarrow{\text{kill()}}$   $\xrightarrow{\ll\text{destroy}\gg}$



# Diagramas de Sequência

Mensagens podem ser ainda:

- Síncronas: objeto que chama espera pelo término do processamento da mensagem pelo objeto chamado. Notação: vide exemplos anteriores.
- Assíncronas:  $\sim$  síncronas. Notação:

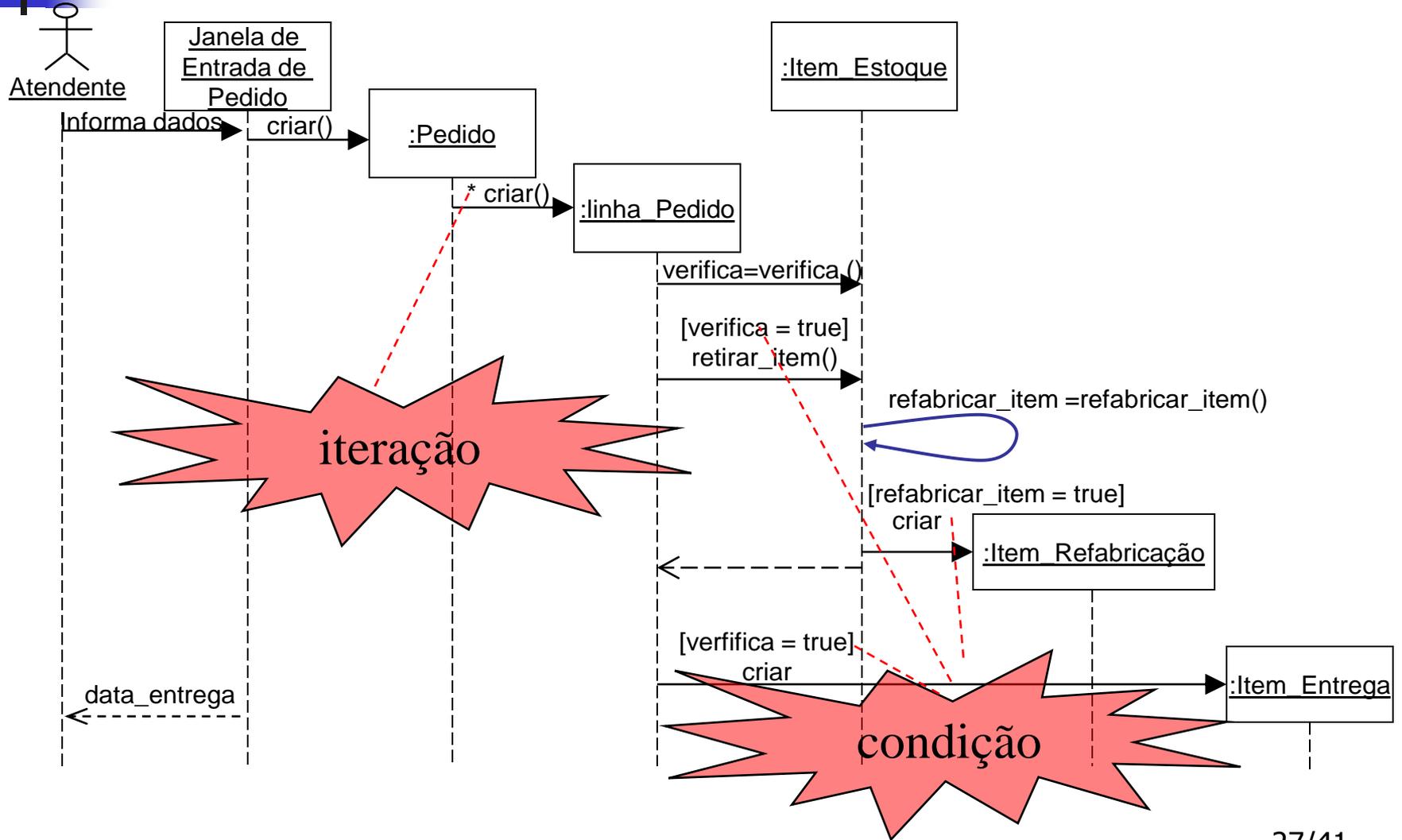


# Diagramas de Sequência

- Uma mensagem também podem conter:
- **Condição:** representada entre “[” e “]”, que indica em que condição a mensagem é enviada.
  - **Iteração:** “\*” no início da mensagem indica que a mesma é enviada repetidas vezes.

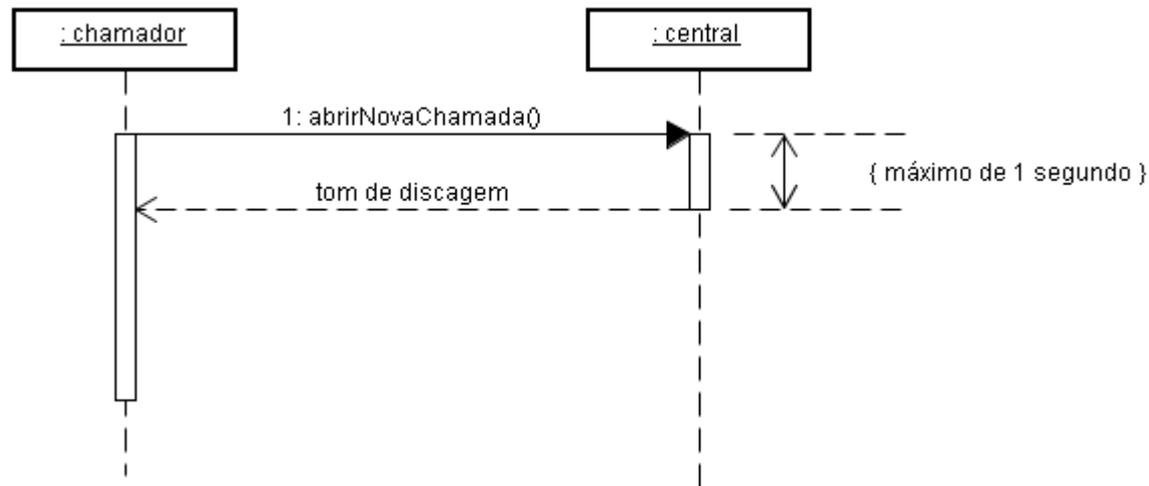


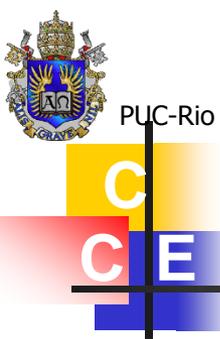
# Diagramas de Sequência



# Diagramas de Sequência

Restrições temporais podem ser representadas.  
Exemplo:





# Diagramas de Sequência

---

- Exercícios:



# Caixa Automático

## ■ Descrição:

... além dos serviços prestados na agências, os clientes podem utilizar os caixas 24h, permanentemente ligados à central bancária para todas as consultas aos dados dos clientes...

... para tal deve-se passar o cartão magnético ... o caixa exibirá as opções disponíveis, que variam com a hora e o tipo do cliente (número de estrelas) ...

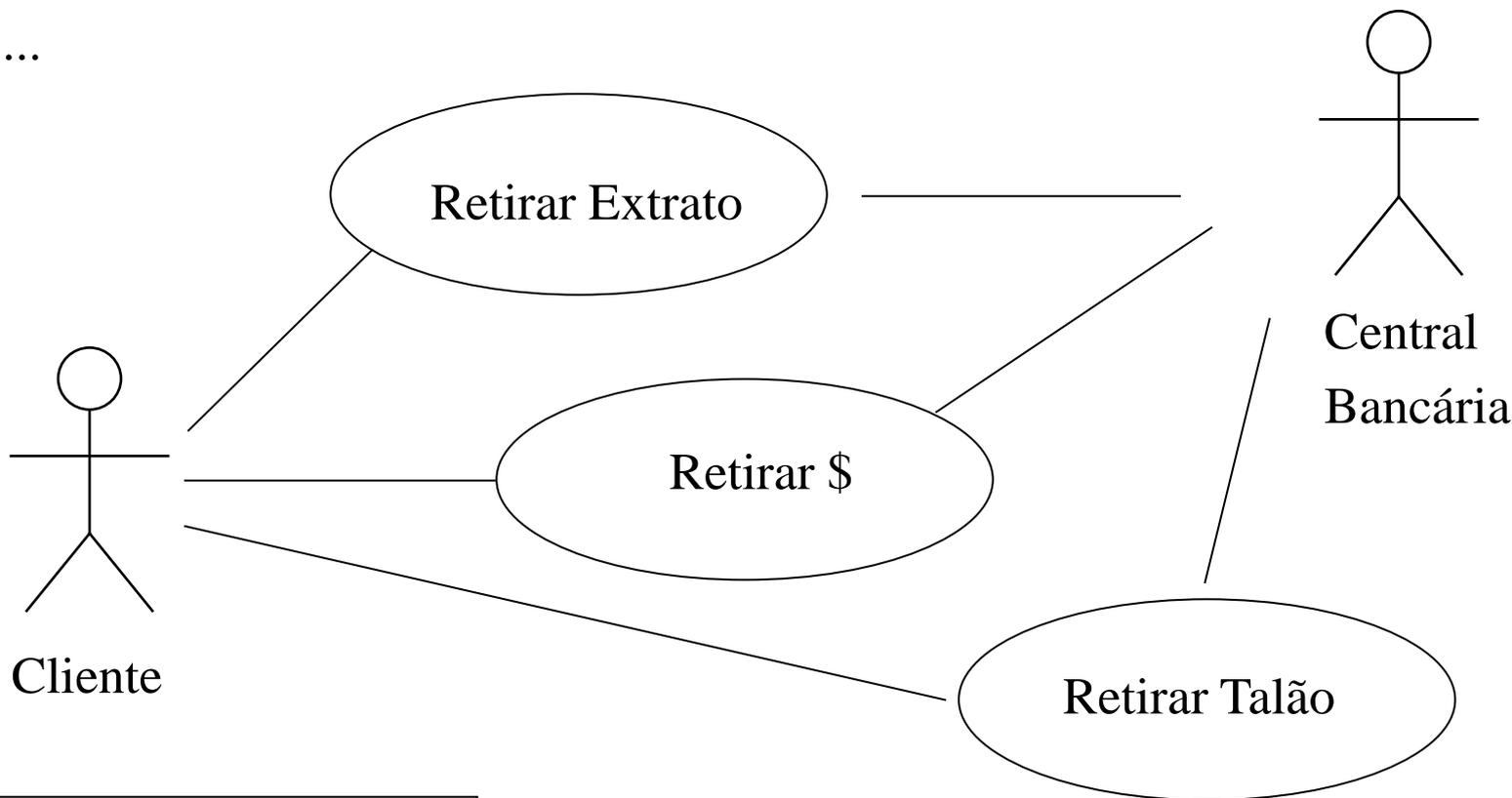
... executando a opção “retirar dinheiro”, o cliente precisará informar o valor e a senha de acesso. O dinheiro será colocado no compartimento ...

... o sistema precisará registrar a operação feita, armazenando o código do cliente, o valor retirado e o horário da operação. Os casos de erro ou desistência após a escolha da opção também deverão ficar registrados. Os registros são enviados à central ao final e, por questões de segurança, nenhum dado de uma transação deve permanecer armazenado no caixa ...



# Caixa Automático

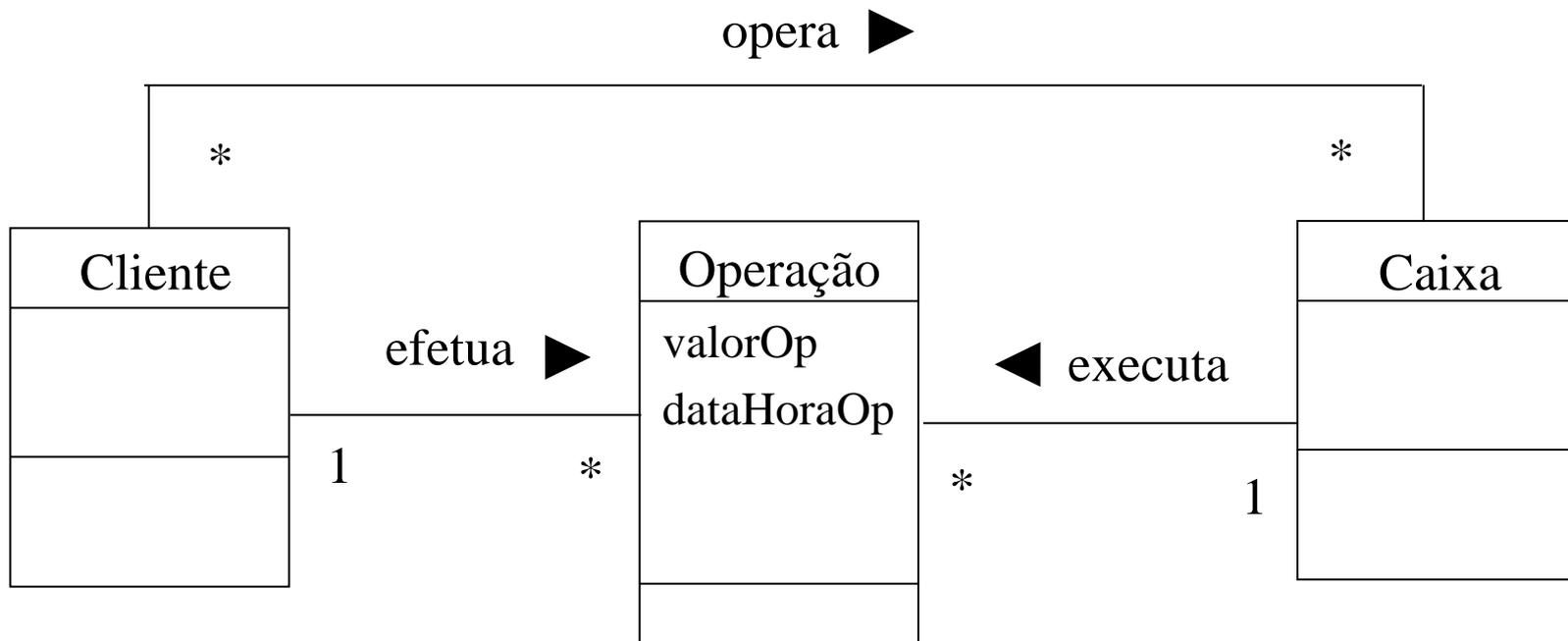
...



Casos de Uso:  
 Sistema Atendimento 24h

...

# Caixa Automático



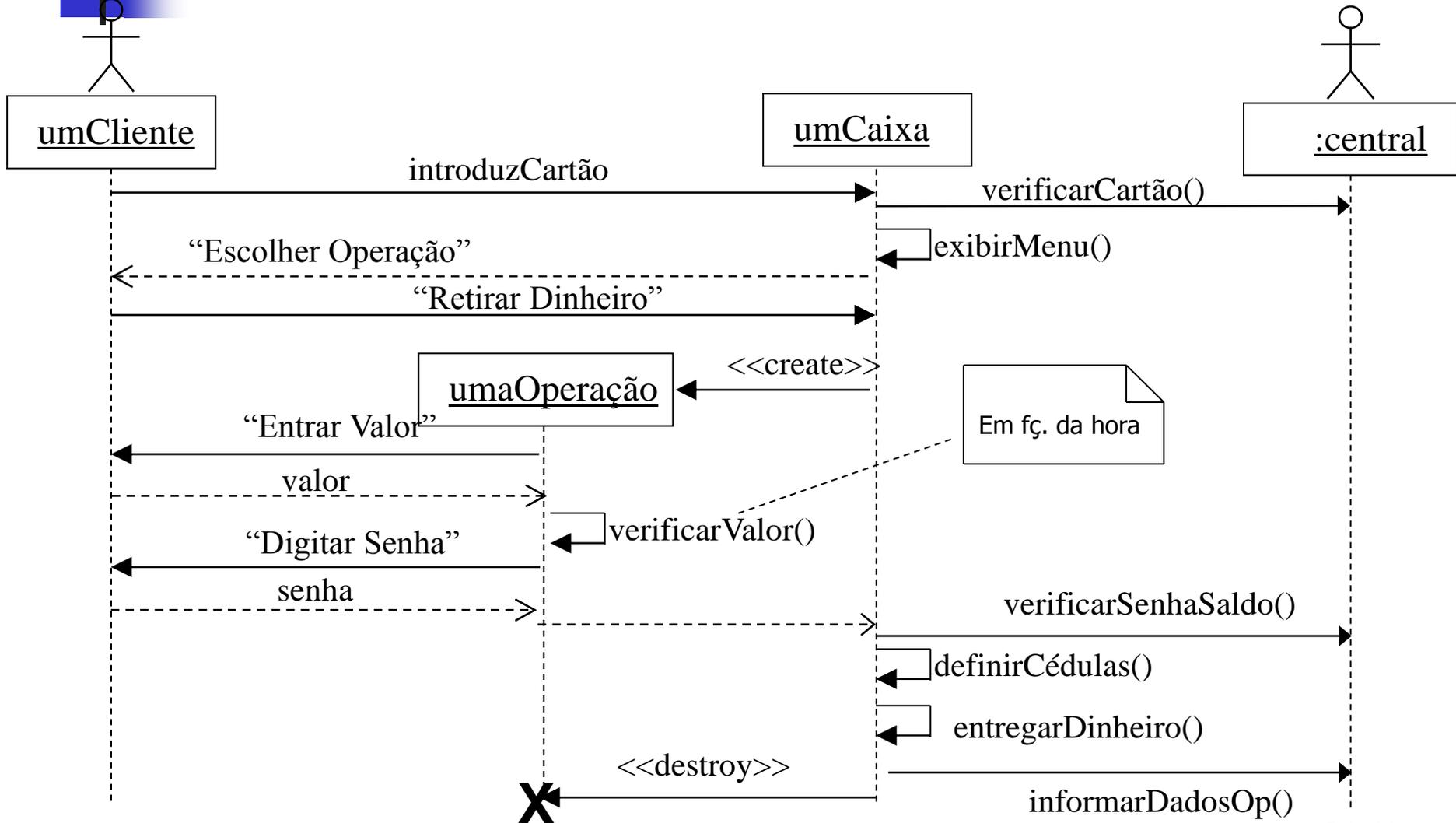
# Caixa Automático

Caso de Uso: Retirar Dinheiro  
 Atores: Cliente, Central Bancária  
 Descrição: Esse caso de uso executa o processo de retirada de dinheiro em um caixa eletrônico 24horas

## Curso Típico dos Eventos

Ações dos Atores	Ações/Respostas do Sistema
1) Cliente introduz cartão magnético	2) Verifica o cartão junto à Central
	3) Apresenta menu de opções e solicita escolha
4) Cliente seleciona “Retirar Dinheiro”	5) Solicita valor
6) Cliente informa Valor	7) Verifica limite de saque/horário
	8) Solicita senha
9) Cliente informa Senha	10) Verifica senha e saldo junto à Central
	11) Libera valor solicitado
	12) Envia info à Central

# Caixa Automático





# Caixa Automático

- Atributos e operações descobertos deverão ser relacionadas no segundo/terceiro compartimento das respectivas classes;
- Atores sistemas em geral disponibilizam APIs para que com eles possamos “interfacear”.

Ex.:

- verificarCartão()
- verificarSenhaSaldo()
- informarDadsOp()



# Caixa Automático

- As classes de nosso modelo de classes ficariam (até o momento):

Caixa
exibirMenu() entregarDinheiro(); definirCedulas()

Operação
codOperação valorOp dataHoraOp
new( ... ) verificarValor() kill()

# Caixa Automático

- Outros cenários deverão ser tratados (da mesma maneira) para o mesmo caso de uso considerando cursos alternativos (erro de leitura de cartão, senha não confere, usuário desiste, usuário não tem saldo suficiente, etc.);
- Durante esse tratamento outros atributos, operações e responsabilidades podem ser detectados;
- Durante esse tratamento, “furos” nos casos de uso ou no diagrama de classes podem ser descobertos!



# DS: Processar Pedido

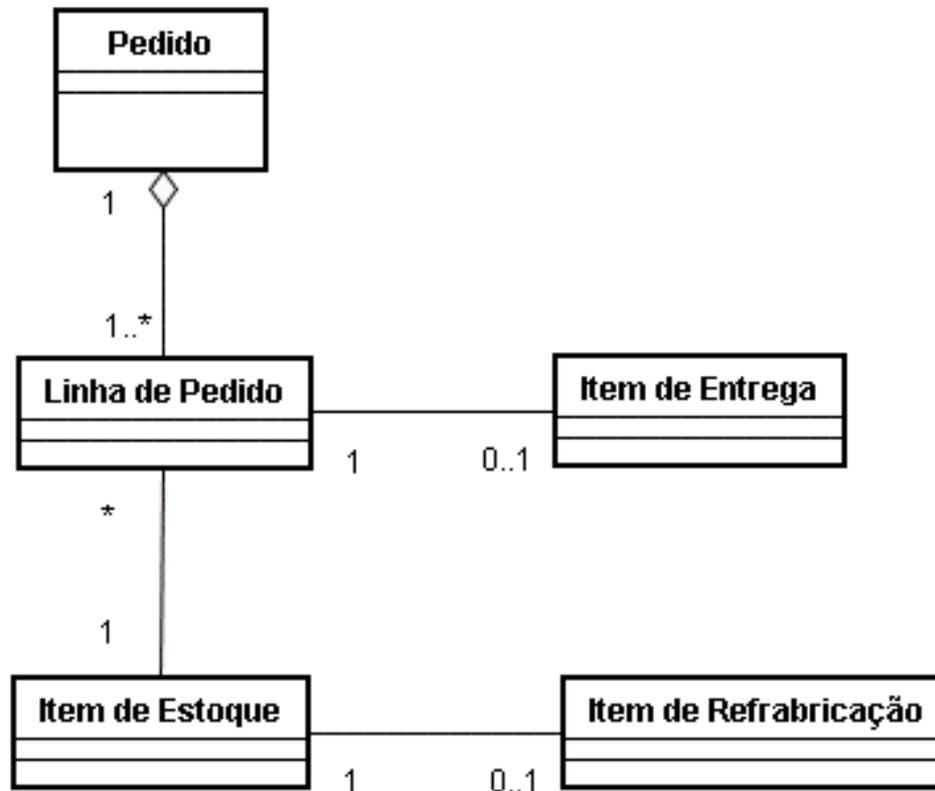
---

- Descrição do Caso de Uso Processar Pedido:
  - Atendente Informa dados do pedido
  - Sistema armazena pedido, retornando a data de entrega



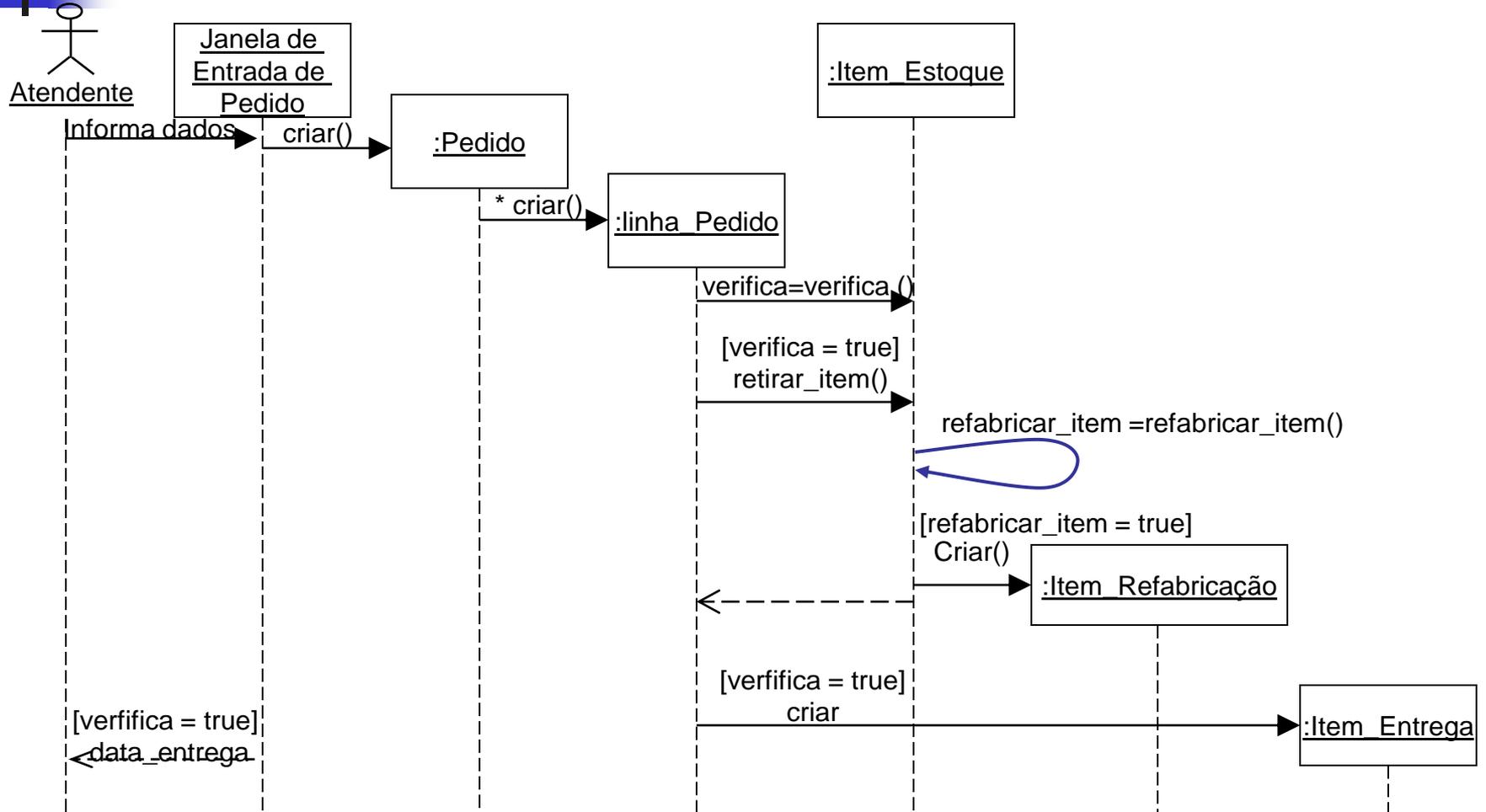
# DS: Processar Pedido

## Diagrama de Classes





# DS: Processar Pedido





# DS: Efetuar Reserva

**Nome:** Efetuar Reserva

**Ator(es):** Cliente

**Descrição:** Esse caso de uso descreve o processo de reserva de um ou mais apartamentos em um dos hotéis da rede.

## Curso Típico de Eventos

ATORES	SISTEMA
1) Cliente informa hotel e período	2) Apresenta os apartamentos disponíveis e preços
3) Cliente seleciona apartamentos	4) Solicita dados cliente
5) Cliente informa dados	6) Apresenta todos os dados reserva, incluindo valor total
7) Cliente confirma reserva	8) Emite voucher