

Computação Orientada a Objetos

Introdução UML - UNIFIED MODELLING LANGUAGE

Slides baseados em:

- Slides Prof. Christian Dannel Paz Trillo

1

Profa. Karina Valdivia Delgado
EACH-USP

REPRESENTAÇÃO DE SISTEMAS DE SOFTWARE

- Implementei um sistema de empréstimo de livros, e preciso que outra pessoa continue trabalhando nele. Como transmitir esse conhecimento?
 - Código?
 - Explicação Verbal?
 - Documento
 - Uma imagem diz mais do que mil palavras
- Hoje em dia a maioria de sistemas de software são representados por um **conjunto de diagramas** e alguns **documentos** associados.
- Mas se cada um utilizar uma representação diferente?

REPRESENTAÇÃO DE SISTEMAS DE SOFTWARE

- Para mim isto é uma classe:



- Para mim isto é uma classe!



POO – UM POUCO DE HISTÓRIA

- No começo da P.O.O. se criaram várias linguagens e metodologias:
 - Yodd
 - Bosch
 - ...
- Após algum tempo perceberam que isso não daria muito certo, cada um “falando” a sua própria linguagem. Assim nasceu **UML** (Unified Modelling Language).

UML – UNIFIED MODELLING LANGUAGE



- **UML** é uma linguagem padrão da OMG (Object Management Group).
- Permite:
 - Visualização
 - A existência de **modelos visuais** facilita a comunicação e faz com que os membros de um grupo tenham a mesma idéia do sistema.
 - Cada **símbolo gráfico** tem uma **semântica** bem definida.
 - Especificação
 - É uma ferramenta poderosa para a especificação de diversos aspectos arquiteturais e internos do sistema.

UML – UNIFIED MODELLING LANGUAGE

- Permite:
 - Construção
 - Geração automática de código a partir do modelo visual.
 - Geração do modelo visual a partir do código
 - Documentação
 - Os diagramas UML junto com a documentação associadas a cada um deles, são parte importante da documentação do software.

UML – VISÃO GERAL

- Tipos de diagramas em UML:
 - Diagramas Estruturais:
 - Descrever **aspectos estáticos** de um sistema:
 - Classes
 - Pacotes (agrupamento de classes)
 - Componentes hardware e software.
 - Diagramas Comportamentais:
 - Descrevem **aspectos dinâmicos** de um sistema:
 - Casos de uso
 - Sequência/colaboração
 - Estados
 - Atividades.

UML – VISÃO GERAL

- Tipos de diagramas em UML:
 - Diagramas Estruturais:
 - Descrever **aspectos estáticos** de um sistema:
 - Classes
 - Pacotes (agrupamento de classes)
 - Componentes hardware e software.
 - Diagramas Comportamentais:
 - Descrevem **aspectos dinâmicos** de um sistema:
 - Casos de uso
 - Sequência/colaboração
 - Estados
 - Atividades.

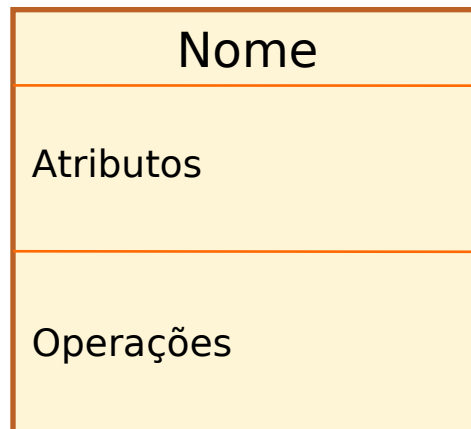
UML – DIAGRAMAS DE CLASSES

- Permitem **modelar o vocabulário** do sistema, distinguindo o que faz parte do sistema e o que não faz.
- **Modelar as associações** entre os elementos constituintes do sistema.
- Modelagem lógica dos dados para o seu armazenamento (banco de dados, etc.)

UML – DIAGRAMA DE CLASSES

○ Notação

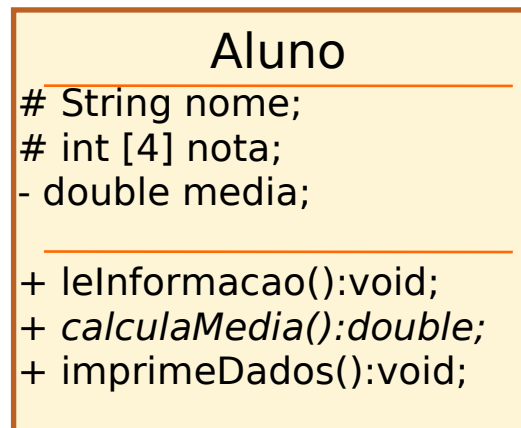
- A classe tem um nome, uma lista de atributos (pode ser vazia) e de métodos (pode ser vazia).



UML – DIAGRAMA DE CLASSES

○ Notação

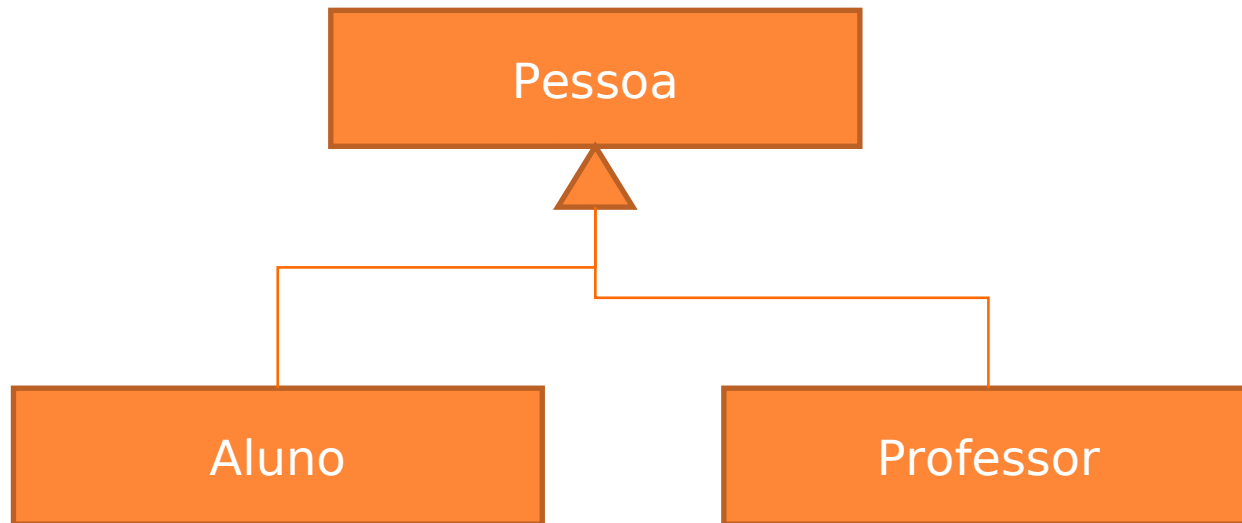
- O tipo de dados dos atributos e de retorno dos métodos, assim como dos parâmetros pode ser especificado.
- Os indicadores **+** (público), **-** (privado) e **#** (protegido) indicam a privacidade dos atributos.



- As letras em *itálico* indicam que a classe ou os métodos são abstratos.

UML – DIAGRAMA DE CLASSES

- Relacionamentos
 - Herança



UML – DIAGRAMA DE CLASSES

- Relacionamentos
 - Associação



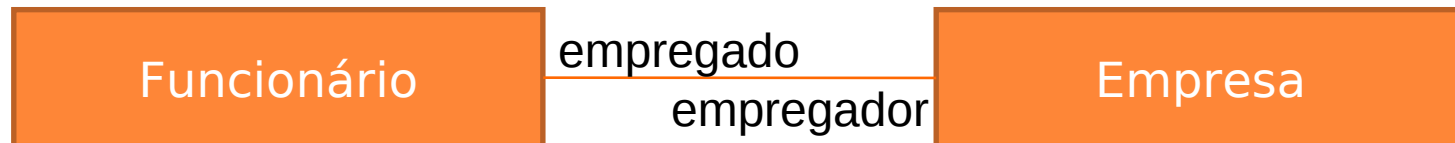
UML – DIAGRAMA DE CLASSES

- Relacionamentos
 - Associação – Ornamentos
 - Nome
 - Direção do relacionamento
 - Classe1 verbo Classe2



UML – DIAGRAMA DE CLASSES

- Relacionamentos
 - Associação – Ornamentos
 - Papel



UML – DIAGRAMA DE CLASSES

○ Relacionamentos

● Associação – Ornamentos

○ Multiplicidade

- Pode ser um valor exato, ou um intervalo, ou * para dizer muitos.



UML – DIAGRAMA DE CLASSES

○ Relacionamentos

● Agregação

- Relação do tipo “todo/parte” ou “tem” ou “possui”.



UML – DIAGRAMA DE CLASSES

- Relacionamentos

- Composição

- As partes são inseparáveis do todo.



UML – DIAGRAMA DE CLASSES

○ Relacionamentos

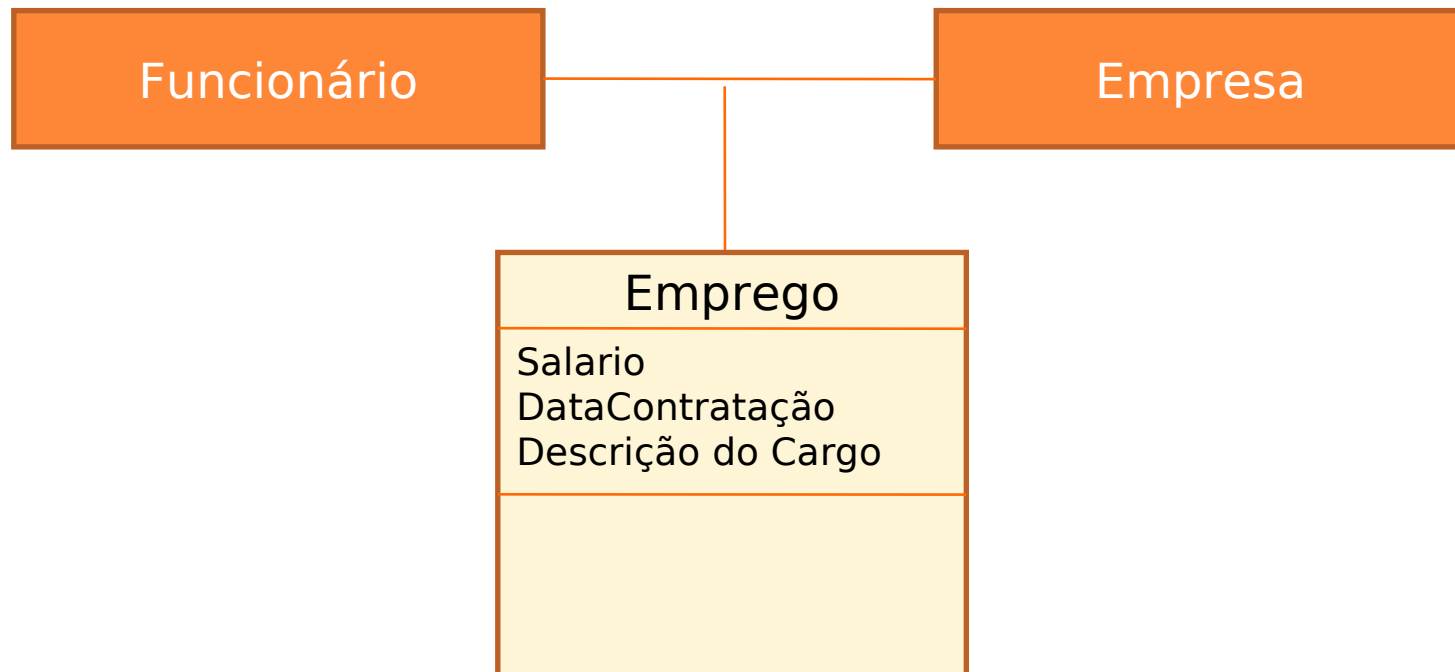
● Usa

- Quando uma classe “utiliza” outra, por exemplo dentro de um método, utiliza-se este tipo de relacionamento.



UML – DIAGRAMA DE CLASSES

- Classes de Associação
 - São classes que representam o relacionamento entre duas classes.



UML – DIAGRAMA DE CLASSES

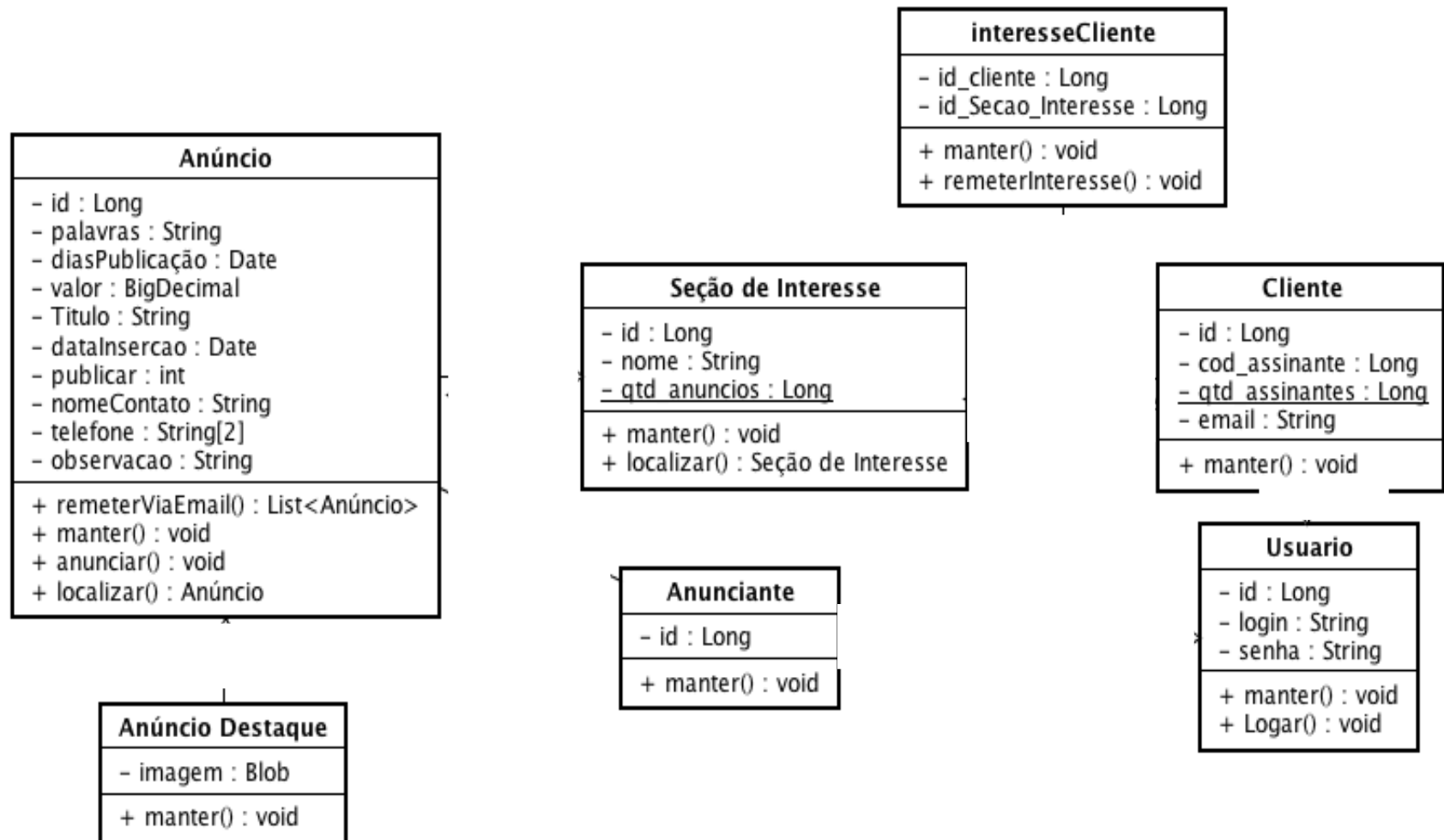
- Notas

- São elementos descritivos que podem ser adicionados a qualquer diagrama UML

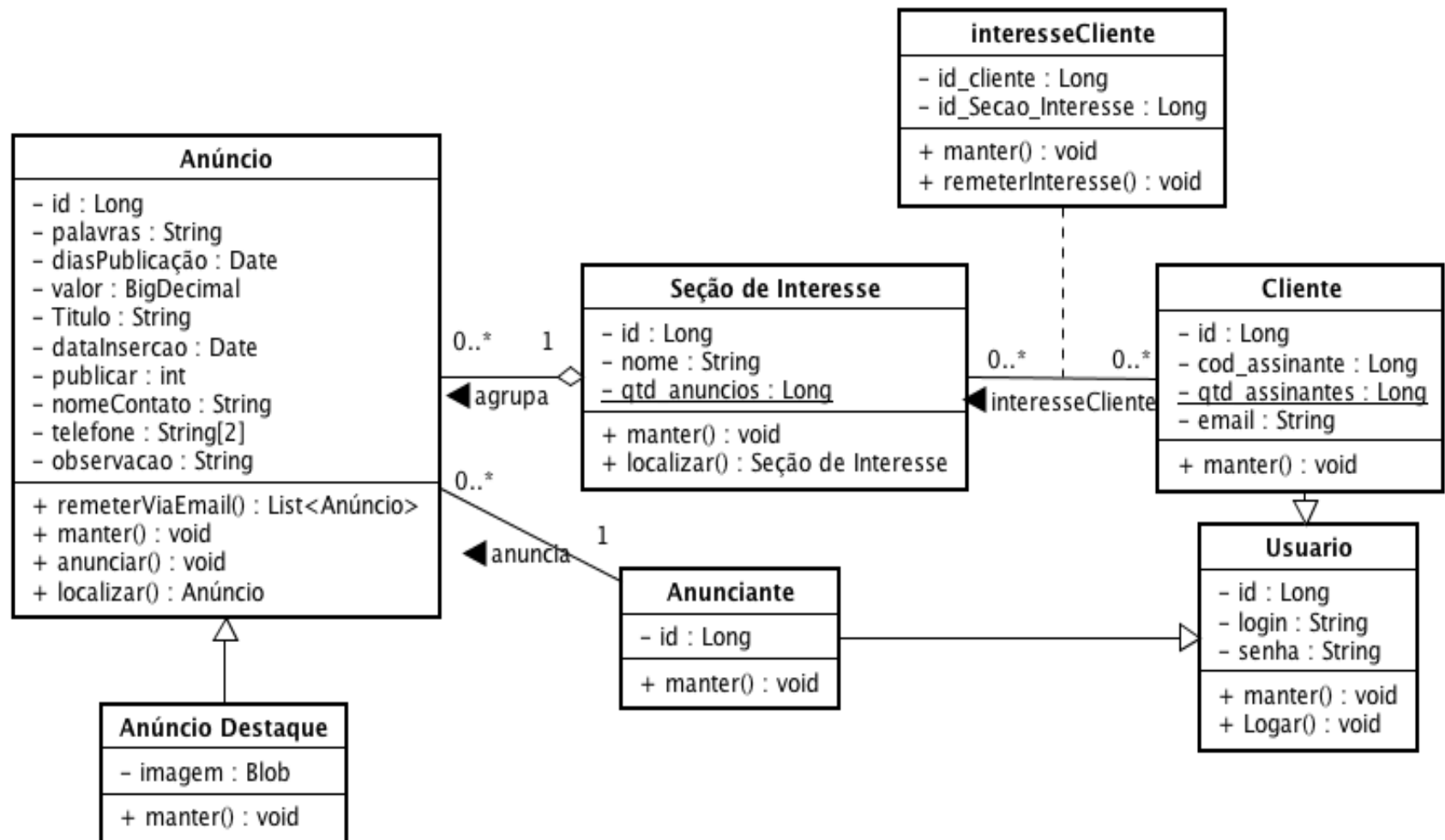
An orange rectangular box with a folded top-right corner, containing text.

O login do aluno é o código de matrícula dele.

UML – EXEMPLO DE DIAGRAMA DE CLASSES



UML – EXEMPLO DE DIAGRAMA DE CLASSES

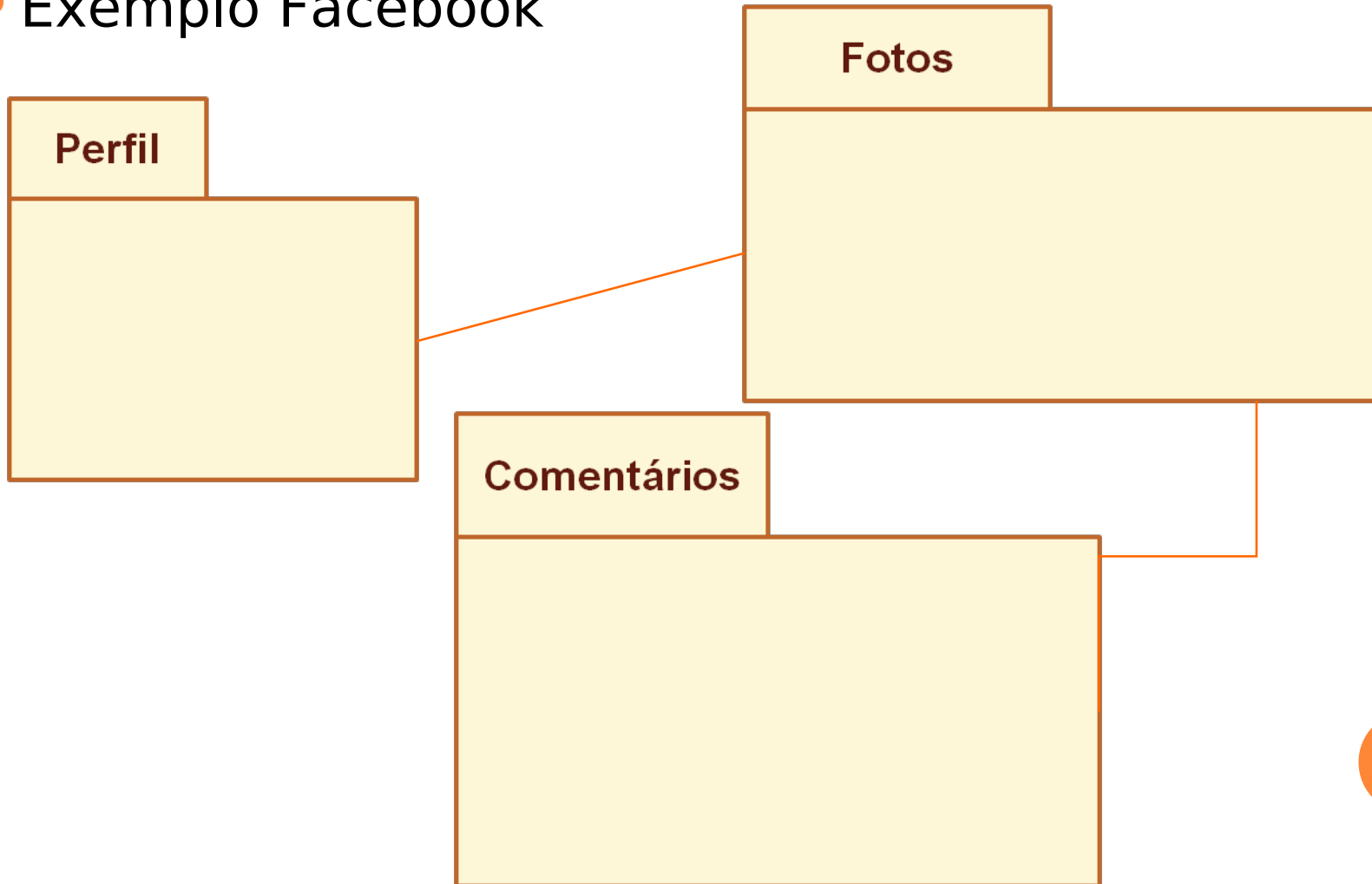


UML – DIAGRAMA DE PACOTES

- Um Pacote em UML (como em Java) permite organizar elementos de um modelo em grupos.
- Cada elemento do modelo pertence a um único pacote.
- O nome do elemento dentro do pacote deve ser único.
- Relacionamento entre pacotes:
 - Os pacotes podem-se relacionar diretamente.
 - Elementos internos de um pacote podem se relacionar com elementos internos de outros pacotes.

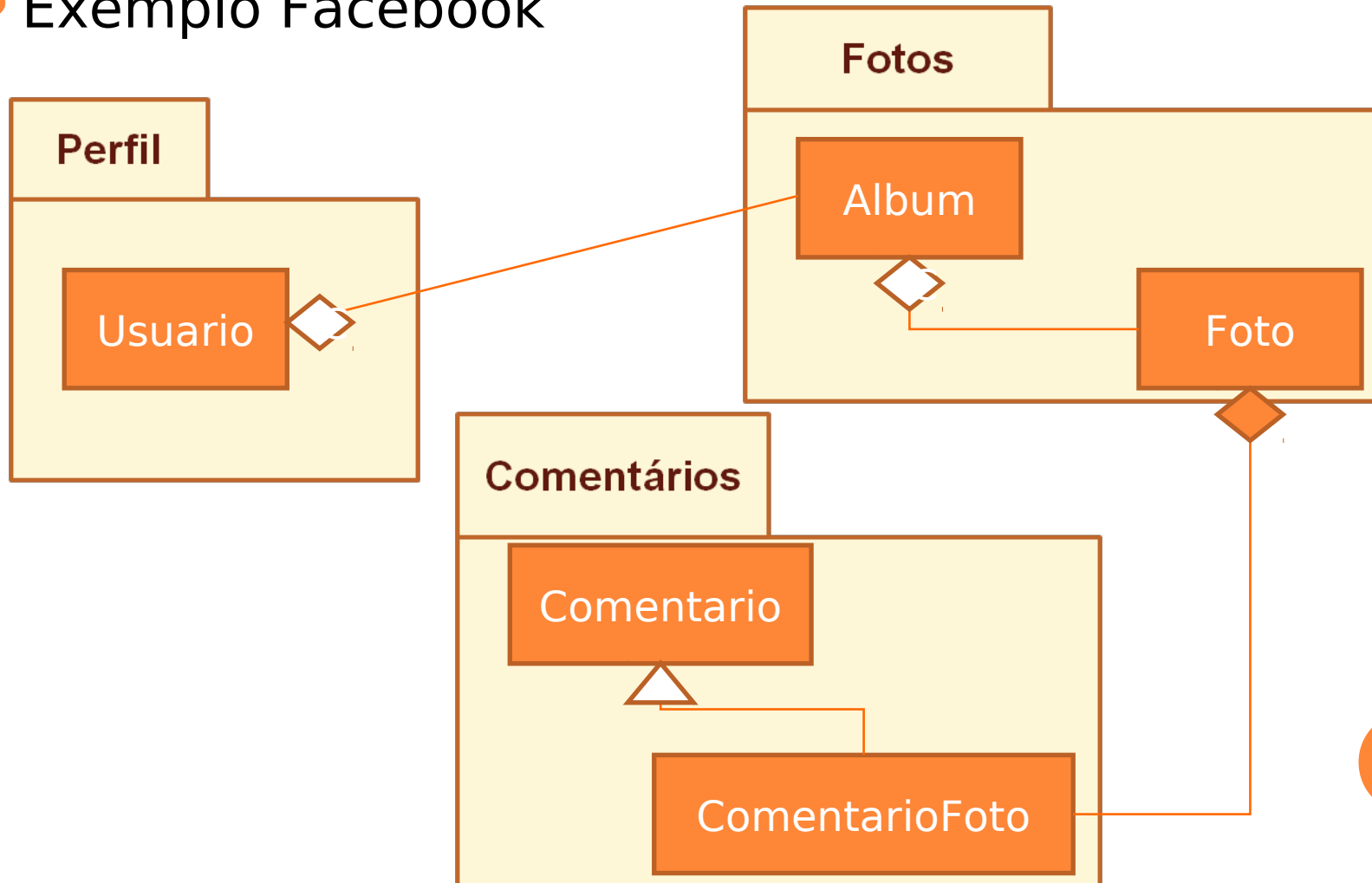
UML – DIAGRAMA DE PACOTES

- Exemplo Facebook



UML – DIAGRAMA DE PACOTES

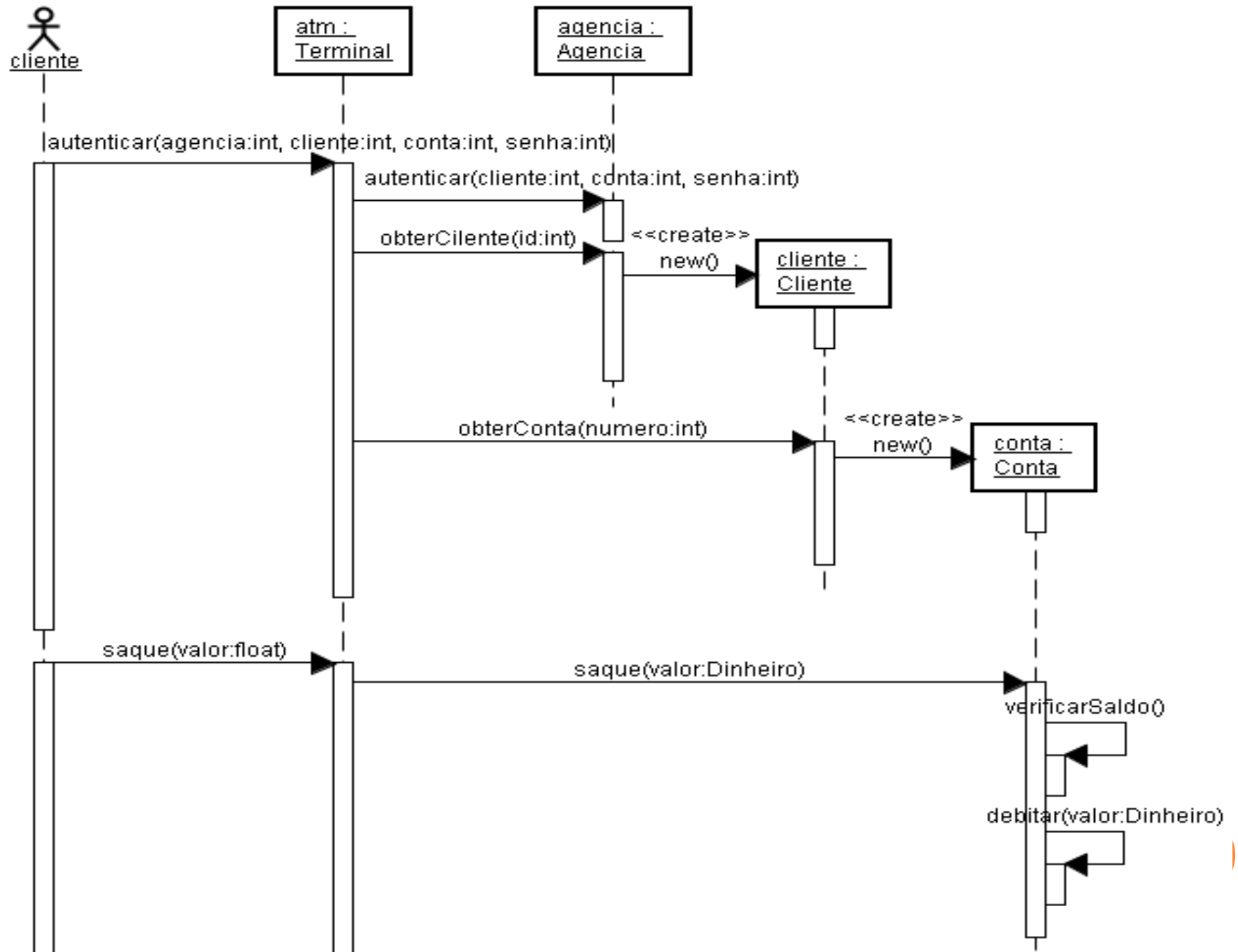
- Exemplo Facebook



UML – DIAGRAMA DE SEQUÊNCIA

○ Definições:

- **Mensagem:** Um objeto chama uma operação (método) de outro objeto.
 - Os parâmetros são a informação enviada do objeto que chama ao objeto chamado.
 - Os valores de retorno compõe a informação enviada do objeto chamado ao objeto que chama.
- **Interação:** Seqüência de mensagens trocadas entre objetos.
- **Linha de Vida:** é uma linha tracejada que representa o tempo de vida de um objeto



UML – DIAGRAMA DE SEQUÊNCIA

- Usuário:
 - Representa um usuário real do sistema.
 - É utilizado para representar o usuário do sistema que inicializa alguma sequência a través de alguma interação com a interface do sistema.

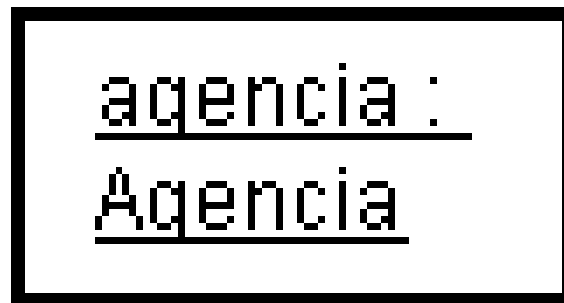


- Usualmente interage com um objeto de interface, por exemplo o ATM (Caixa Automático).

UML – DIAGRAMA DE SEQUÊNCIA

○ Objeto:

- Representa a instância de uma classe: uma variável do tipo da classe.
- Ele é descrito por um nome de variável e o nome da classe.

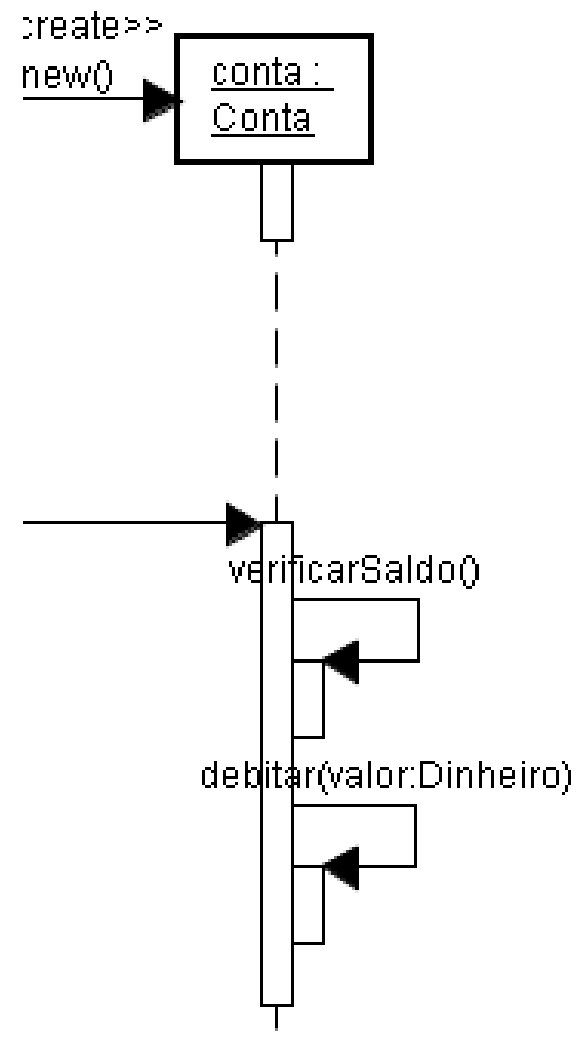


- A linha tracejada embaixo do objeto corresponde ao tempo de vida do objeto: desde que o construtor é chamado até que a variável é destruída.

UML – DIAGRAMA DE SEQUÊNCIA

○ Foco de Controle:

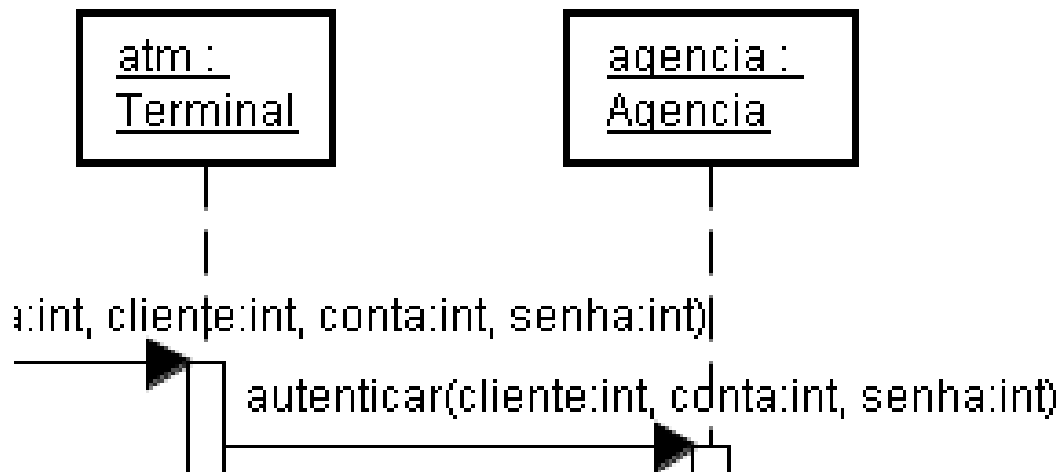
- Representa o tempo que o objeto está ativo na execução do programa.
- Por exemplo:
 - O objeto conta está ativo enquanto é criado e depois somente quando é solicitado o saque.
 - Se entre a autenticação e o saque o usuário faz alguma outra atividade, solicitar informação de empréstimos por exemplo, a conta está fora do foco de controle.

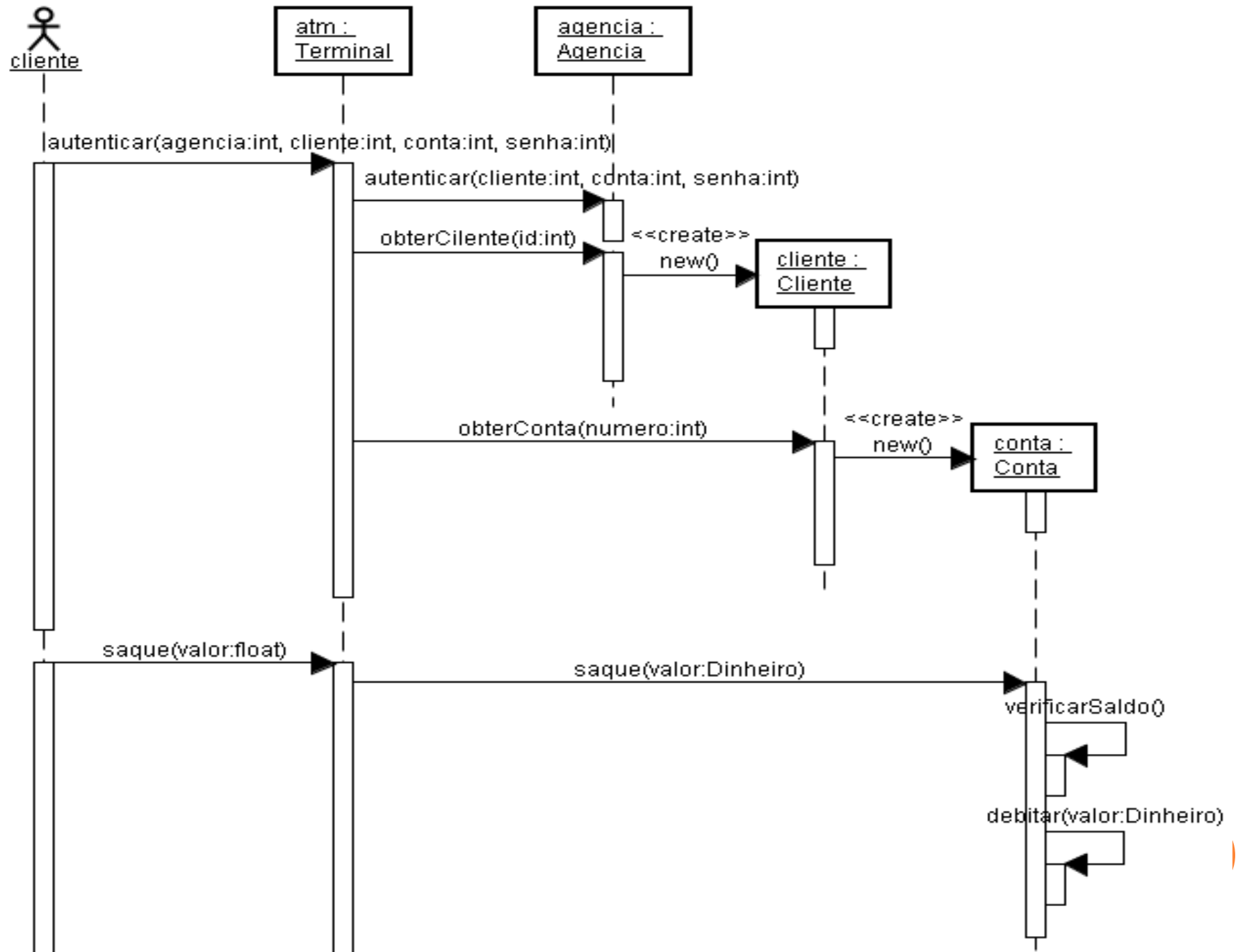


UML – DIAGRAMA DE SEQUÊNCIA

○ Mensagem:

- Representa a chamada de um método de um objeto por parte de outro objeto.
- Pode conter detalhes dos parâmetros e tipos de dados enviados, pois dessa chamada serão extraídos os parâmetros do método a ser implementado.





UML – DIAGRAMA DE SEQUÊNCIA

○ Leitura:

- O Cliente se autentica no Terminal.
 - O Terminal se comunica com a Agência para verificar a senha do Cliente.
 - O objeto Agência cria um objeto Cliente.
 - O Terminal seleciona uma Conta do Cliente.
 - O objeto Cliente cria um objeto Conta associado a ele.
- O Cliente solicita um saque.
 - O Terminal solicita o saque a Conta do Cliente.
 - A Conta verifica o Saldo.
 - A Conta efetua o Débito.