

Seja bem-vindo à

targettrust
treinamento e tecnologia



Apresentação do Instrutor



- Nome: Cássio Trindade
- Quais são suas experiências? Arquiteto Software
- Quais são suas formações?
- Já fez algum outro treinamento na TargetTrust? Como instrutor e/ou como aluno?
- Quais são suas expectativas quanto a este treinamento?

Java Persistence API com MOR

1

Revisão Geral do Mapeamento Objeto Relacional

Objetivos

- Introduzir o conceito de Mapeamento Objeto Relacional
- Demonstrar a utilização do Mapeamento Objeto Relacional no processo de desenvolvimento de software.
- Evidenciar problemas na abordagem MOR manual
- Apresentar possíveis soluções

public 🐼

public.agenda		
🔑 id_agenda	bigint	« pk nn »

rel_consulta_agenda

public.consulta		
🔑 id_consulta	bigint	« pk fk nn »
○ cancelada	boolean	
○ dataconsulta	timestamp	
○ preco	real	
🔑 animal_id_animal	bigint	« fk »

public.animal		
🔑 id_animal	bigint	« pk nn »
○ altura	integer	« nn »
○ apelido	character varying(255)	
○ especie	character varying(255)	
○ peso	real	« nn »
○ vivo	boolean	« nn »

rel_consulta_animal

public.fornecedor		
○ ativo	boolean	« nn »
○ material	character varying(255)	
🔑 id_pessoa	bigint	« pk fk nn »

rel_cliente_animal

public.cliente		
🔑 id_pessoa	bigint	« pk fk nn »
🔑 animal_id_animal	bigint	« fk »

rel_fornecedor_pessoa

public.vendedor		
○ comissao	real	« nn »
🔑 id_pessoa	bigint	« pk fk nn »

rel_cliente_pessoa

rel_vendedor_funcionario

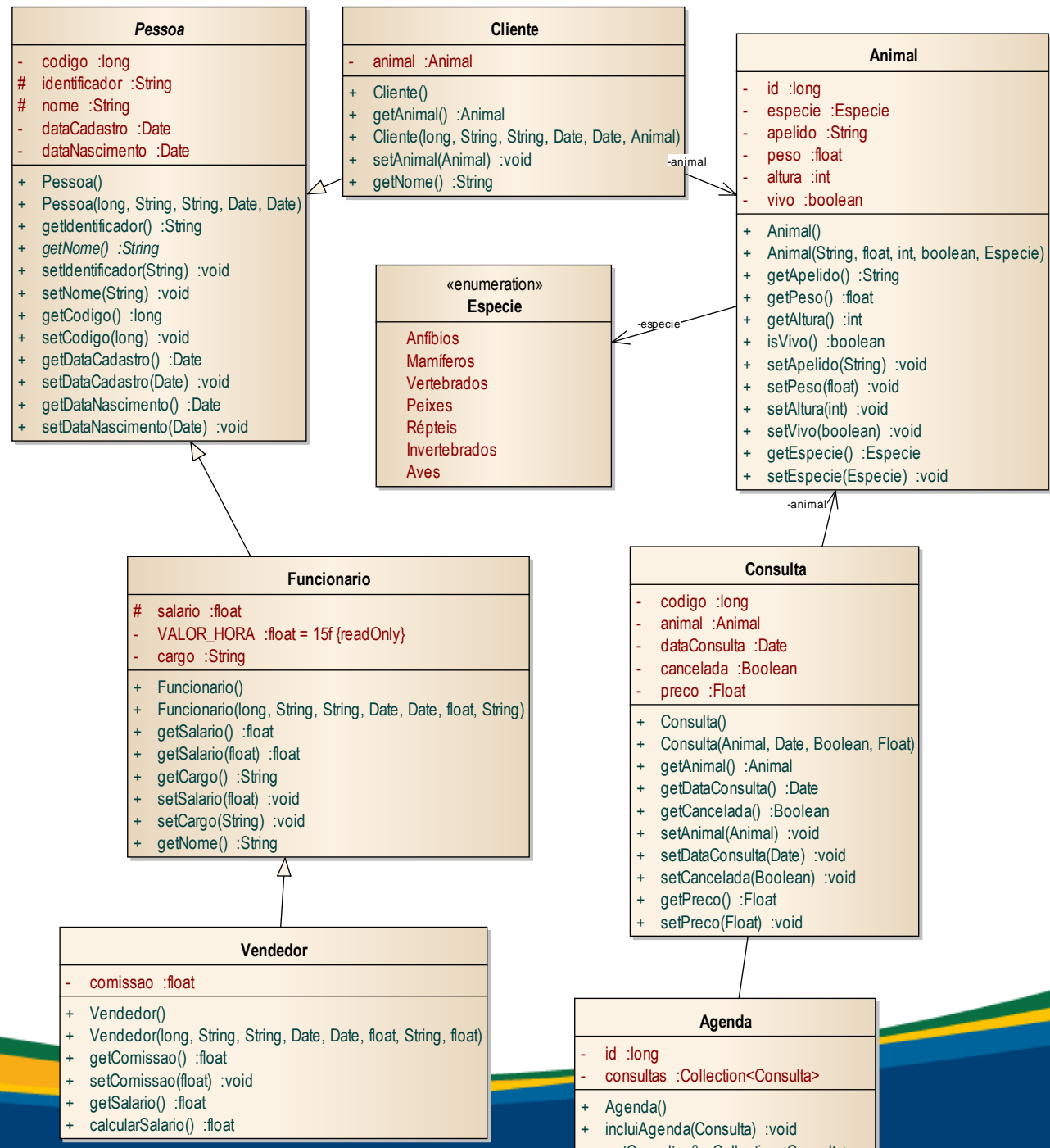
public.pessoa		
🔑 id_pessoa	bigint	« pk nn »
○ datacadastro	date	
○ datanascimento	date	
○ identificador	character varying(255)	
○ nome	character varying(255)	

rel_funcionario_pessoa

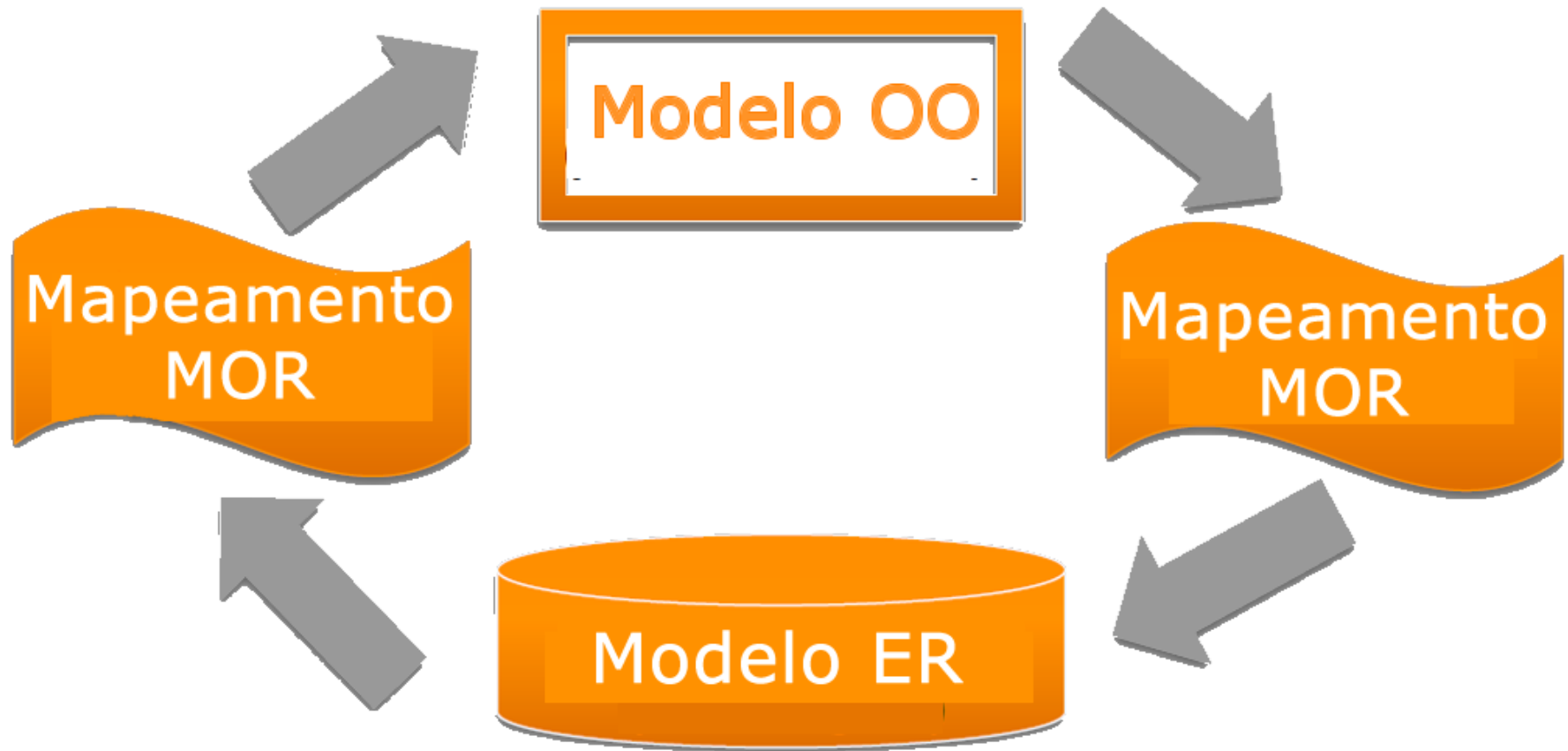
public.funcionario		
○ valor_hora	real	« nn »
○ cargo	character varying(255)	
○ salario	real	« nn »
🔑 id_pessoa	bigint	« pk fk nn »

Classes

class model

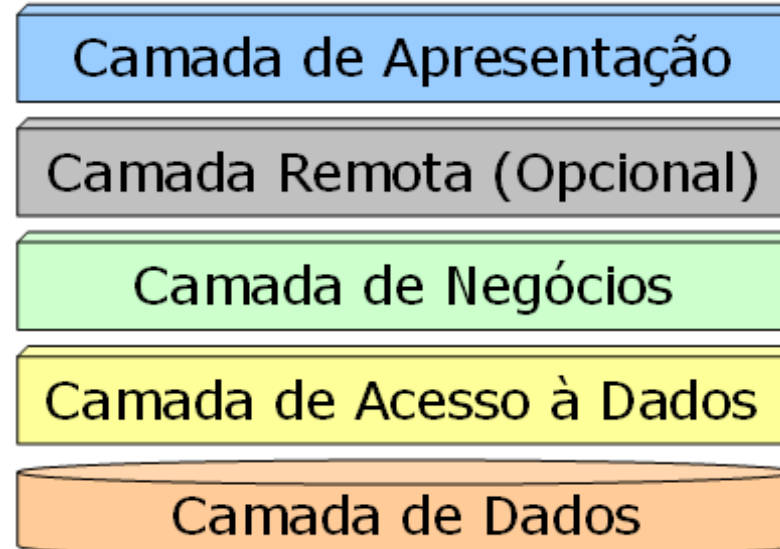


ER \Leftrightarrow OO



Orientação a Objetos e Aplicações Multicamadas

- Classes bastante desacopladas permitem um melhor reuso de código.
- Aplicações N camadas permitem desacoplar dados e funcionalidades.



Soluções

- SGBDOO
- Padrões de Projeto (Design Patterns)
- Framework MOR

Bancos de Dados Orientado a Objetos (SGBDOO)

- Não são padrão de fato
Desempenho, gerenciamento e segurança ainda são aspectos negativos
- Existência de muitos SGBDRs de qualidade;
- Dependência de fornecedor
Não há um padrão estabelecido para Java, por exemplo
- OQL da OMG tem variantes de acordo com o SGBDOO
- Conhecimento amplo em SGDBR.

Padrões de Projeto

- Permitem código evolutivo e corretivo de acordo com as camadas da aplicação;
- Padrões organizam o trabalho, mas não o eliminam (Design Patterns);
- Problemas:
 - aumentam a complexidade do sistema;
 - necessidade de treinamento;
 - ritmo mais lento de desenvolvimento

O Problema nas Camadas de Acesso a Dados e Dados

Dependemos do SGBD, do SQL e das Classes de Domínio

```
/**  
 * Grava um novo Cliente  
 * @param cliente  
 */  
public void gravar(Cliente cliente) {  
    if (cliente != null) {  
        try {  
            if (cliente.getId() == 0) {  
                cliente.setId(getNovoId());  
                PreparedStatement stmt =  
                    PostgreSQL.getConnection().prepareStatement(  
                        "INSERT INTO CLIENTE (id,nome, cpf) VALUES (?, ?, ?)");  
                stmt.setLong(1, cliente.getId());  
                stmt.setString(2, cliente.getNome());  
                stmt.setString(3, cliente.getCpf());  
                stmt.executeUpdate();  
            }  
        }  
    }  
}
```

O Problema nas Camadas de Acesso a Dados e Dados

```
} else {  
    PreparedStatement stmt = PostgreSQL  
        .getConnection().prepareStatement(  
            "UPDATE CLIENTE" SET nome = ?,cpf = ? WHERE id = ?");  
    stmt.setString(1, cliente.getNome());  
    stmt.setString(2, cliente.getCpf());  
    stmt.setLong(3, cliente.getId());  
    stmt.executeUpdate();  
}  
} catch (SQLException e) {  
    e.printStackTrace();  
}  
}  
}
```

O Problema nas Camadas de Acesso a Dados e Dados

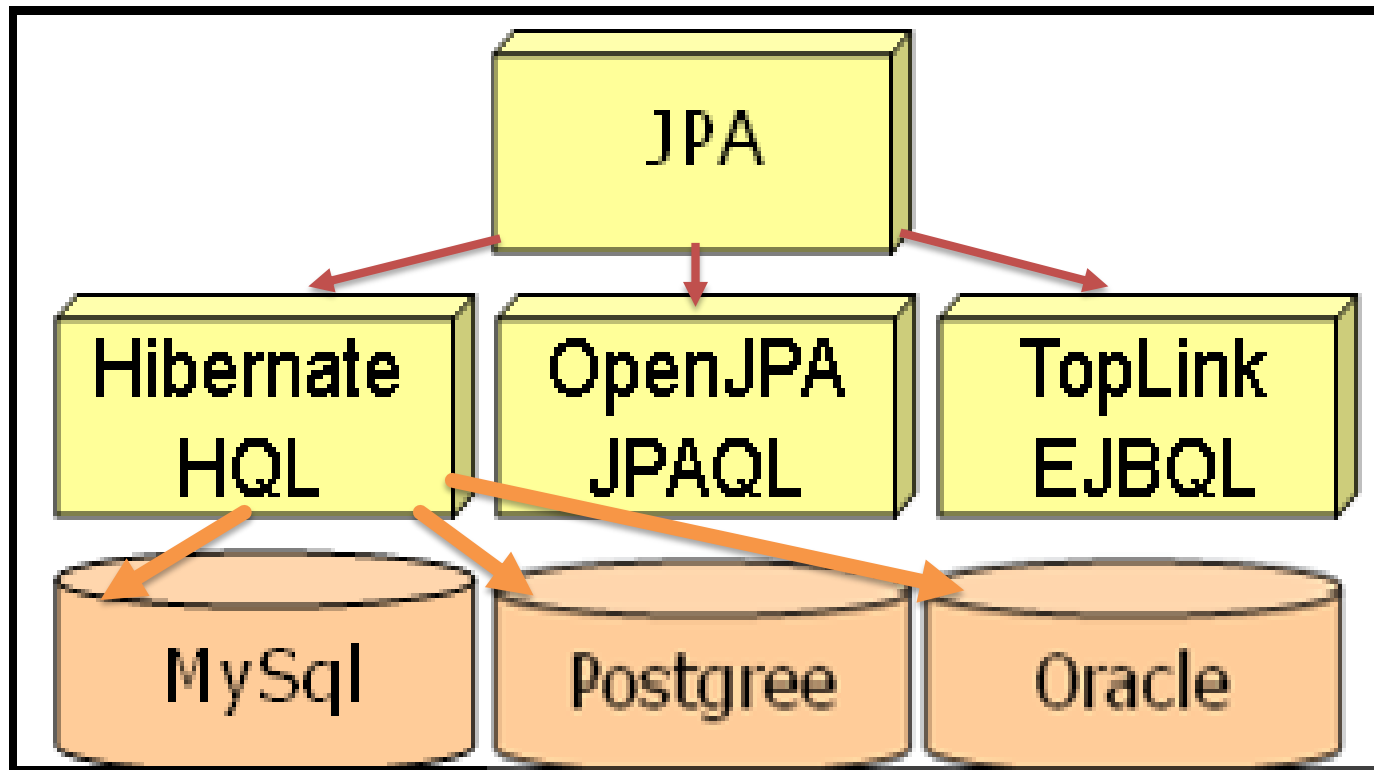
**

* Método que pesquisa o último id utilizado na tabela cliente e retorna
* este id incrementado em 1.
*/

```
private Long getNovold() throws SQLException {  
    java.sql.Statement stmt = PostgreSQL.getConnection().createStatement();  
    ResultSet rs = ((java.sql.Statement) stmt)  
        .executeQuery("SELECT MAX(id) FROM CLIENTE");  
    if (rs.next())  
        return rs.getLong(1) + 1;  
    return new Long(1);  
}
```

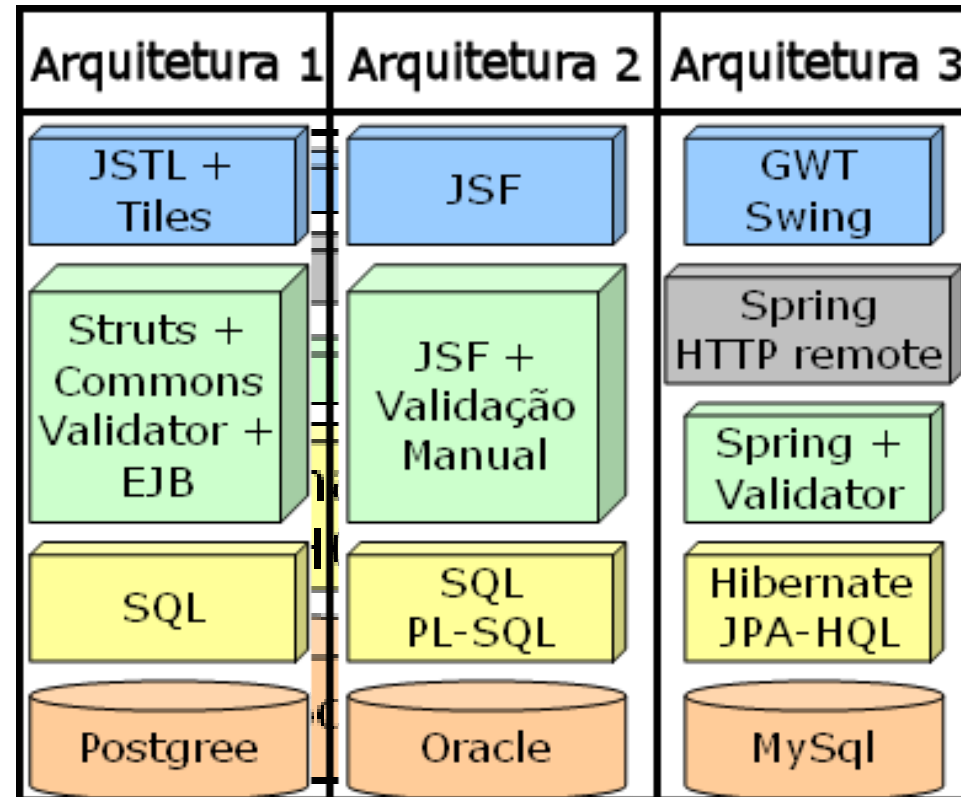
Framework de Mapeamento OO-ER

- Independência de SGBDR
- Flexibilidade de mapeamento
- Mapeamento totalmente OO



Boa Solução: Framework Objeto-Relacional

- Das ilustrações ao lado, a terceira é a mais flexível, no que diz respeito a camada de acesso à dados (JPA) e a camada de gerenciamento de dados (RDBMS)



Introdução ao Mapeamento Objeto-Relacional

Mapeamento de propriedade é a descrição de como um atributo de um objeto deve ser persistido.

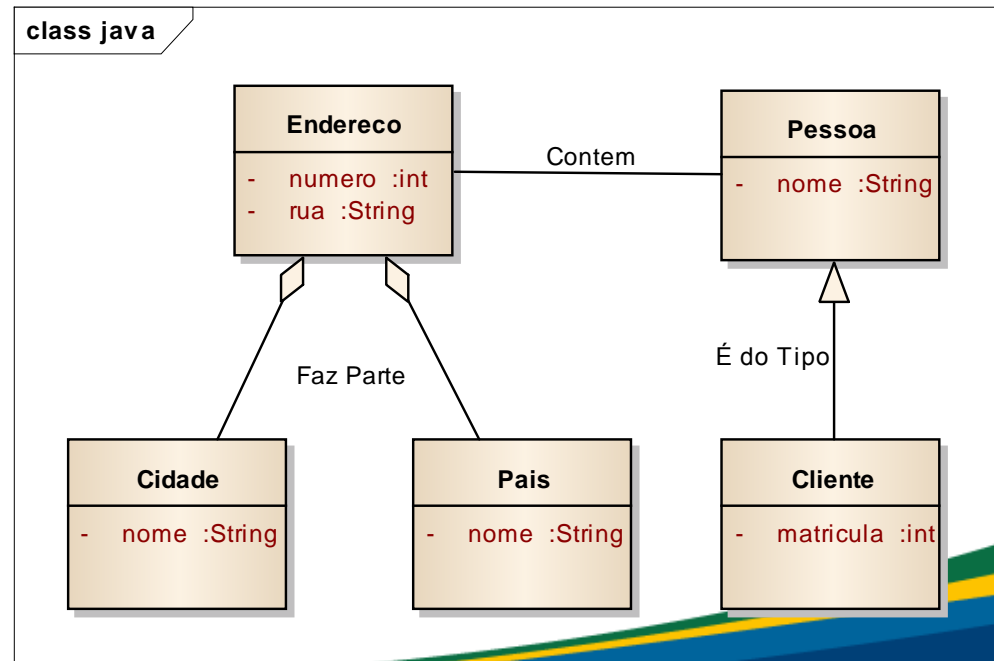
atributo de dado, String nome

atributo virtual, getIdade()



um atributo físico

Mapeamento de relacionamentos é a descrição de como um relacionamento (associação, agregação ou composição) entre dois ou mais objetos deve ser persistido em um BDR



Quem faz o mapeamento?

- A tarefa é dividida entre muitos dos participantes;
- Geralmente o analista fará um mapeamento preliminar;
- Os programadores irão refatorar o mapeamento a medida que os testes forem sendo realizados;
- Os DBAs **modificarão** a base de dados e os programadores precisarão adaptar o modelo de objetos.

MOR e Processos de Desenvolvimento de Software

- XP e Scrum
 - Pequenas equipes
 - Programador arquiteto faz o mapeamento
 - Requisito codificado e testado
- RUP (Unified Process)
 - Agilidade garantida pelo número de participantes no processo
 - Maior colaboração

Exercícios

1. O que é persistência?
2. O que é impedância objeto-relacional?
3. Quais são os tipos de incompatibilidades técnicas entre o modelo de objetos e o modelo relacional?
4. O que é MOR? Qual o seu objetivo?
5. Em poucas palavras, como o MOR é viabilizado no desenvolvimento de sistemas?
6. Defina:
 - Mapeamento;
 - Mapeamento de propriedade;
 - Mapeamento de relacionamento.
7. Quais são os mapeamentos fundamentais? Discuta sobre quem deve fazer o MOR na sua organização
8. Por quais motivos é interessante usarmos frameworks MOR?
9. Na sua opinião, em quais situações você implementaria o MOR na mão, sem o auxílio de um framework?