

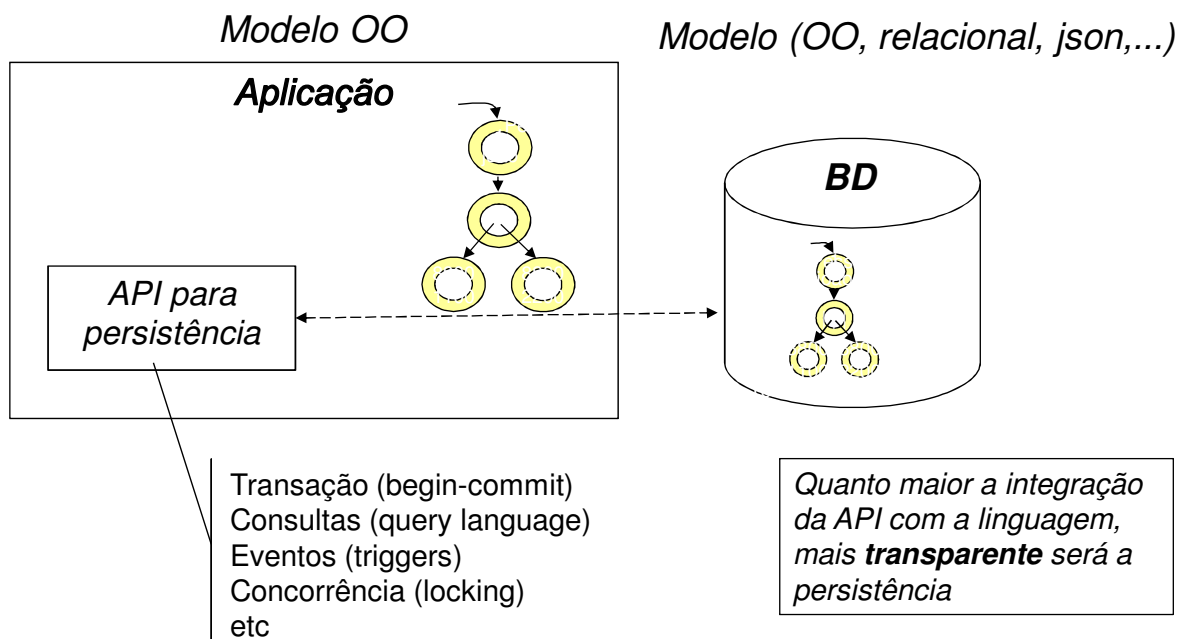
Persistência de Objetos

Prof. Fausto Maranhão Ayres

1 Introdução

Persistência de Objetos

- Objetivo é **sincronizar** os objetos criados pela aplicação Java com banco de dados



Persistência não transparente

- Ex: embutir SQL dentro da linguagem

```
PreparedStatement st = con.prepareStatement(
    "select * from Livro where titulo = 'java8'");

ResultSet cursor = st.executeQuery();

if(cursor.next()){    //encontrou o titulo no banco de dados
    Livro livrol = new Livro(
        cursor.getInt("id") ,
        cursor.getString("titulo") ,
        cursor.getString("autor")    //só um autor
    );
    System.out.println(livrol);
}
```

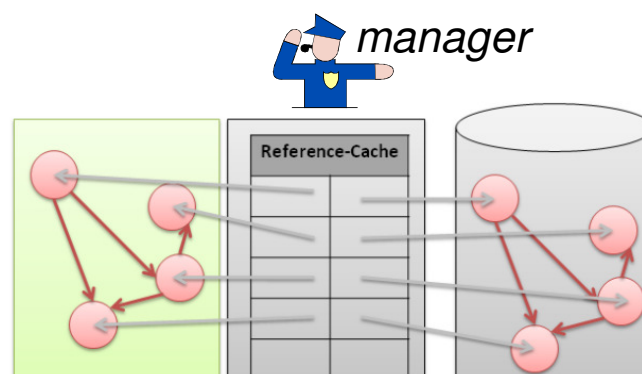
Se o livro tivesse vários autores?

Fausto Maranhão Ayres - IFPB

3

Persistência Transparente

- O programador **não deve se preocupar** com “detalhes” da transferência de objetos entre a aplicação (Memória) e BD.
 - Quem se preocupa ?
O Gerenciador da Persistência (manager) instanciado através de uma API



fausto.ayres@ifpb.edu.br

4

Persistência Transparente

- Definição Histórica:

Um objeto é persistente quando o tempo de vida dele ultrapassa o da aplicação que o criou.

```
persistent int n = 0;  
n++;  
printf(n);          //muda a cada execução
```

Exemplo base

```
public class Pessoa {  
    private int id;  
    private String nome;  
    private List<Telefone> telefones = new ArrayList<Telefone>();  
  
    public class Telefone {  
        private String numero;  
        private Pessoa pessoa;
```

Gerência da persistência

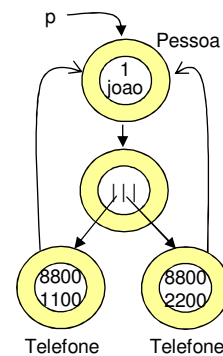
```
public static void main(...) {  
    EntityManager manager = ... ;  
  
    manager.begin();  
    Pessoa p = new Pessoa(1, "joao");  
    p.adicionarTelefone( new Telefone("8800-1100") );  
    p.adicionarTelefone( new Telefone("8800-2200") );  
    manager.persist(p);  
    manager.commit();  
  
    p = manager.find(Pessoa.class, 1);  
    System.out.println(p.getTelefones());  
  
    List<Pessoa> resultados = manager.findAll(Pessoa.class);  
    for (Pessoa p1: resultados)  
        System.out.println(p1);  
}
```

Gravar em cascata

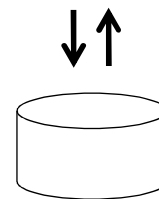
Ler em cascata

fausto.ayres@ifpb.edu.br

memória



manager



7

Gerência da persistência

```
manager.begin();  
p = manager.find(Pessoa.class, 1);  
p.adicionarTelefone( new Telefone("8800-3300") );  
manager.commit();
```

Alterar em cascata

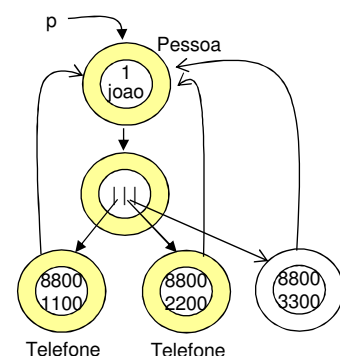
```
manager.begin();  
p = manager.find(Pessoa.class, 1);  
manager.remove(p);  
manager.commit();
```

Excluir em cascata

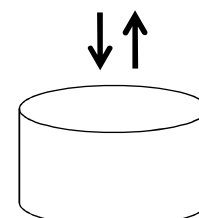
}

fausto.ayres@ifpb.edu.br

memória

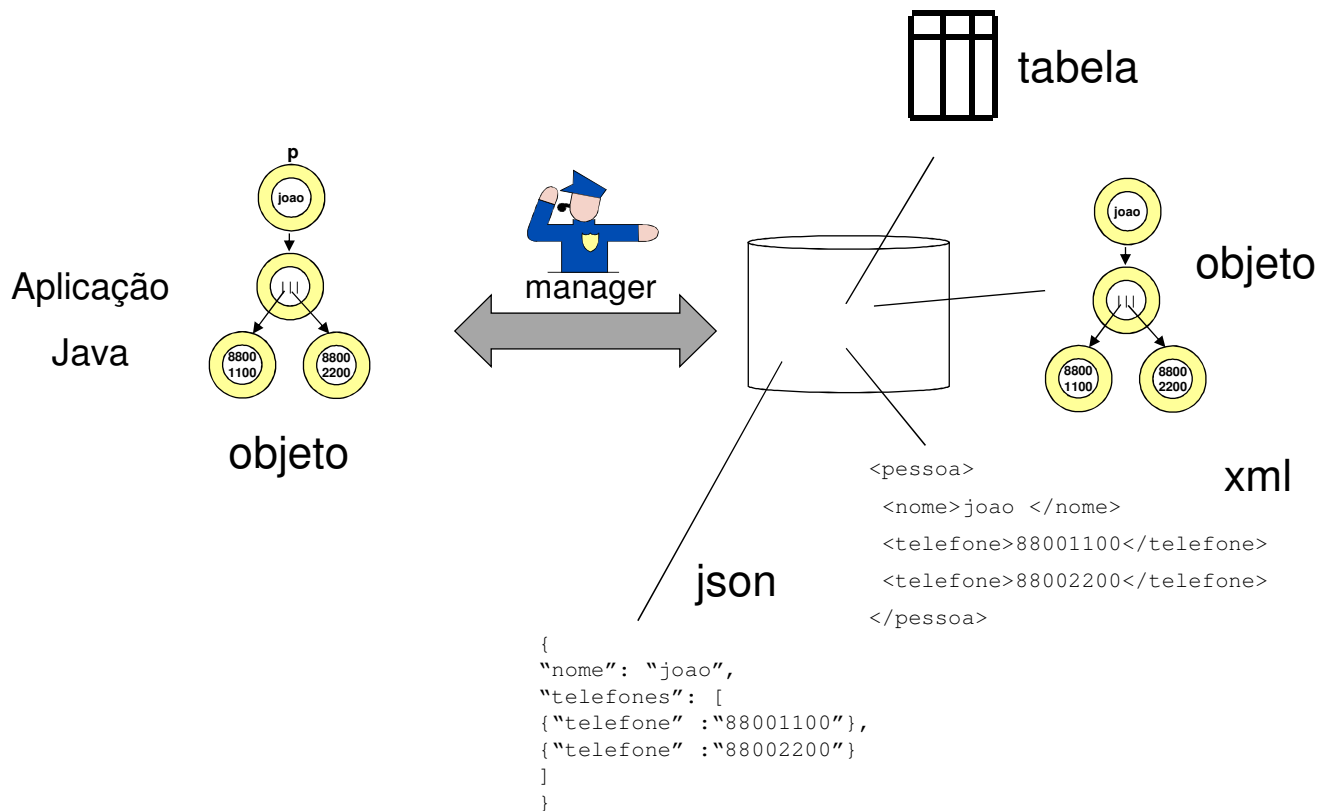


manager



8

Modelo do banco de dados?

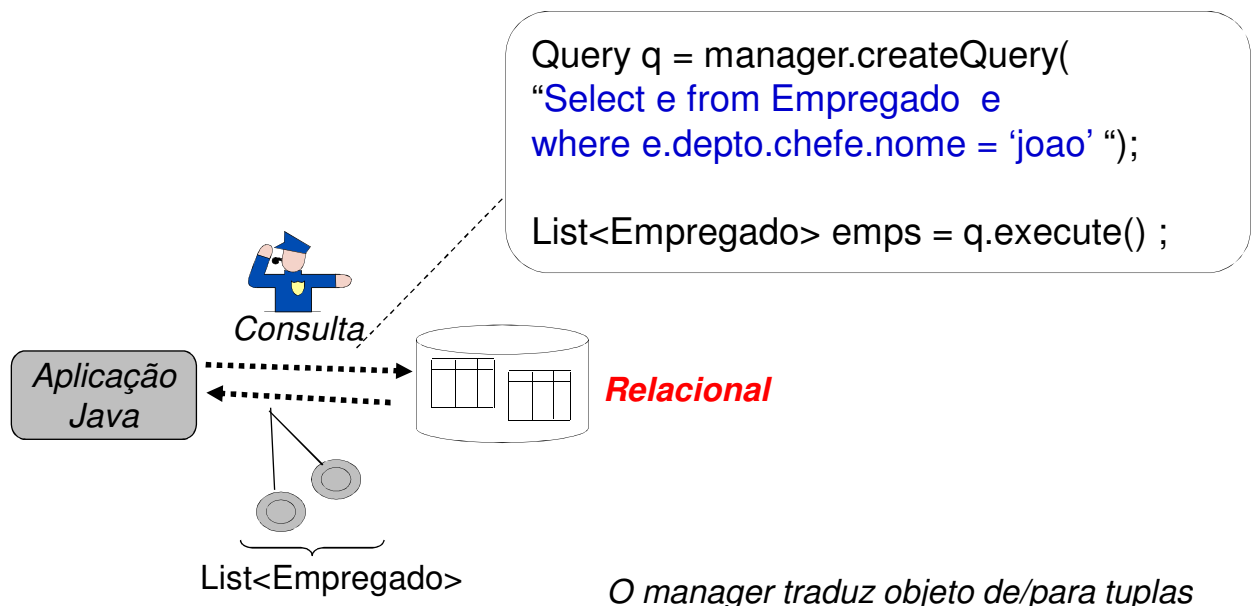


fausto.ayres@ifpb.edu.br

9

Exemplo de Linguagem de Consulta JPQL

Quais os empregados do depto chefiado pelo empregado joao?

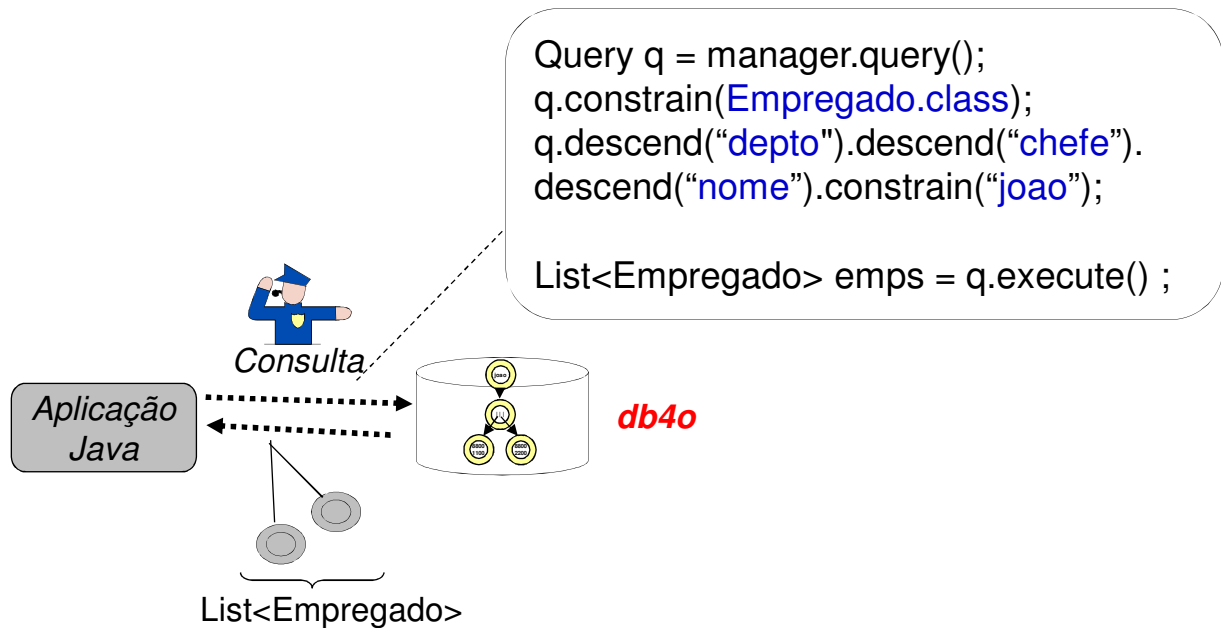


fausto.ayres@ifpb.edu.br

10

Exemplo de Linguagem de Consulta SODA

Quais os empregados do depto chefiado pelo empregado joao?



fausto.ayres@ifpb.edu.br

11

A disciplina

- Conteúdo
 - Conceitos da persistência transparente;
 - Cascata, Carregamento, Transações, Cache, Ciclo de vida, Eventos (trigger), Concorrência (locking)
 - Modelos de dados: Relacional, OO, JSON, XML
 - Linguagem de Consulta (*query language*)
 - Padrão DAO (operações CRUD)
 - Acesso cliente/servidor
 - SGBD
 - DB4o (orientado a objetos)
 - Postgresql (relacional)
 - ObjectDB (orientado a objetos)
 - Firebase (json)
 - MongoDB (json)

fausto.ayres@ifpb.edu.br

12

A disciplina

- Outras Tecnologias
 - Maven
 - JUnit
 - JasperReport

A disciplina

- Avaliação
 - $\text{media} = (1 \text{ prova} + 3 \text{ projetos}) / 4$