

Persistência de Objetos

Fausto Maranhão Ayres

12 JPA – Controle de Concorrência

Conceitos

- Transações concorrentes
 - Quando 2 transações manipulam o mesmo objeto simultaneamente.
- Controle de Concorrência (Bloqueio/Locking)
 - Garantir o **I**solamento entre transações (AC**I**D), evitando que uma não interfira na outra

Tipos de Controle (Bloqueio/Locking)

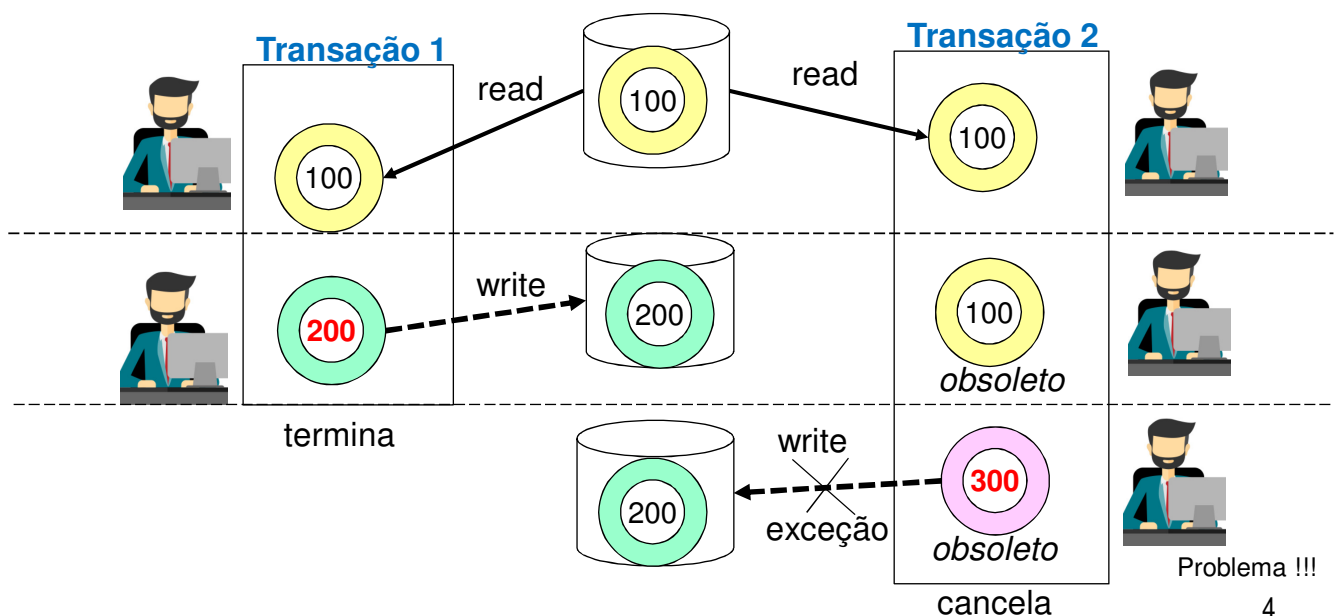
- Bloqueio Otimista (tardio)
 - Assume que existirão poucas transações concorrentes.
 - Garante que nenhuma transação poderá modificar/deletar o objeto gravado na primeira transação
- Bloqueio Pessimista (imediate)
 - Assume que existirão muitas transações concorrentes.
 - Garante que nenhuma transação poderá modificar/deletar o objeto até que a primeira transação tenha terminado

fausto.ayres@ifpb.edu.br

3

Bloqueio Otimista (tardio)

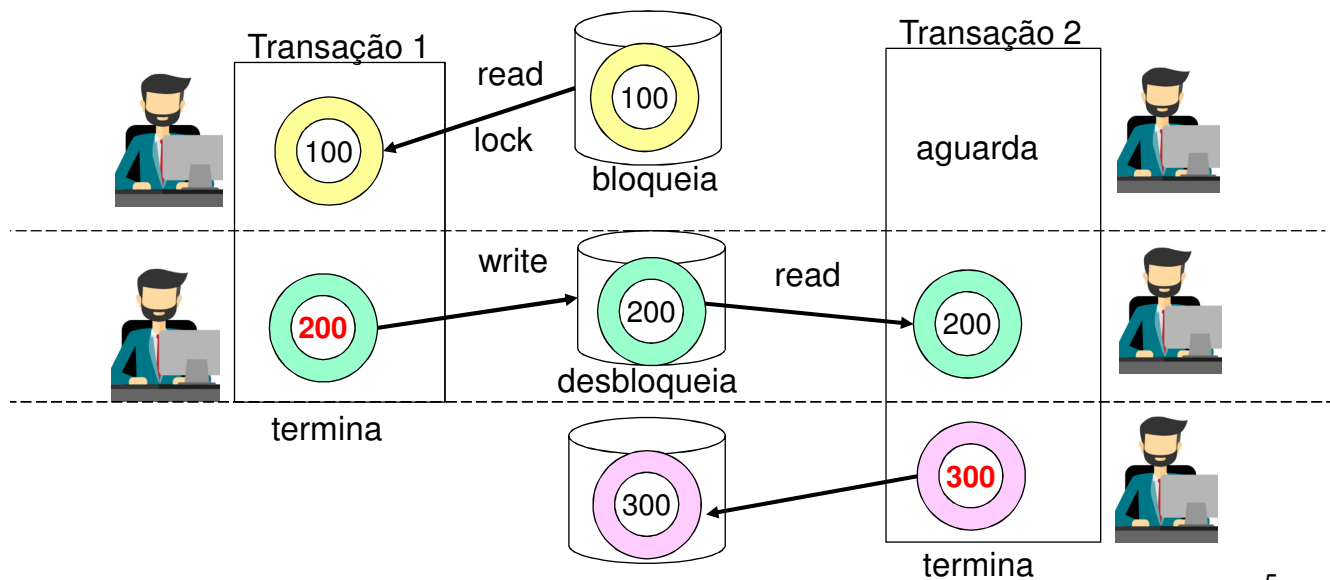
- A primeira transação é efetivada e a segunda é cancelada



4

Bloqueio Pessimista (imediatto)

- A primeira transação bloqueia a segunda até a primeira terminar



5

Implementação no locking otimista no JPA

- **@Version** incrementa a versão do objeto no banco

```
@Entity
public class Produto{...
    @Version
    private int versao;
```

- Cada transação concorrente traz para a memória uma cópia do objeto
- Quando a primeira transação grava a sua cópia, a versão do objeto no banco é incrementada
- Esta versão no banco passa a conflitar com a versão antiga do objeto da segunda transação, causando o cancelamento da segunda transação.

Implementação no locking otimista no JPA

```
manager.getTransaction().begin();
```

```
Produto p = manager.find(10);  
System.out.println(p); //versao=1  
p.setEstoque(p.getEstoque() - 1);
```

A transação que comitar primeiro incrementa a versao. A segunda que tentar gravar, mas será cancelada

```
manager.getTransaction().commit();  
System.out.println(p); // versão=2 (alterada no commit)
```

Implementação no locking pessimista no JPA

```
manager.getTransaction().begin();
```

A primeira transação bloqueia o objeto. A segunda aguarda

```
Produto p = manager.find(10);  
manager.lock( p, LockModeType.PESSIMISTIC_WRITE);  
System.out.println(p);  
p.setEstoque(p.getEstoque() - 1);
```

O commit desbloqueia o objeto. A segunda continua

```
manager.getTransaction().commit();  
System.out.println(p);
```