

Mapeamento Objeto/Relacional

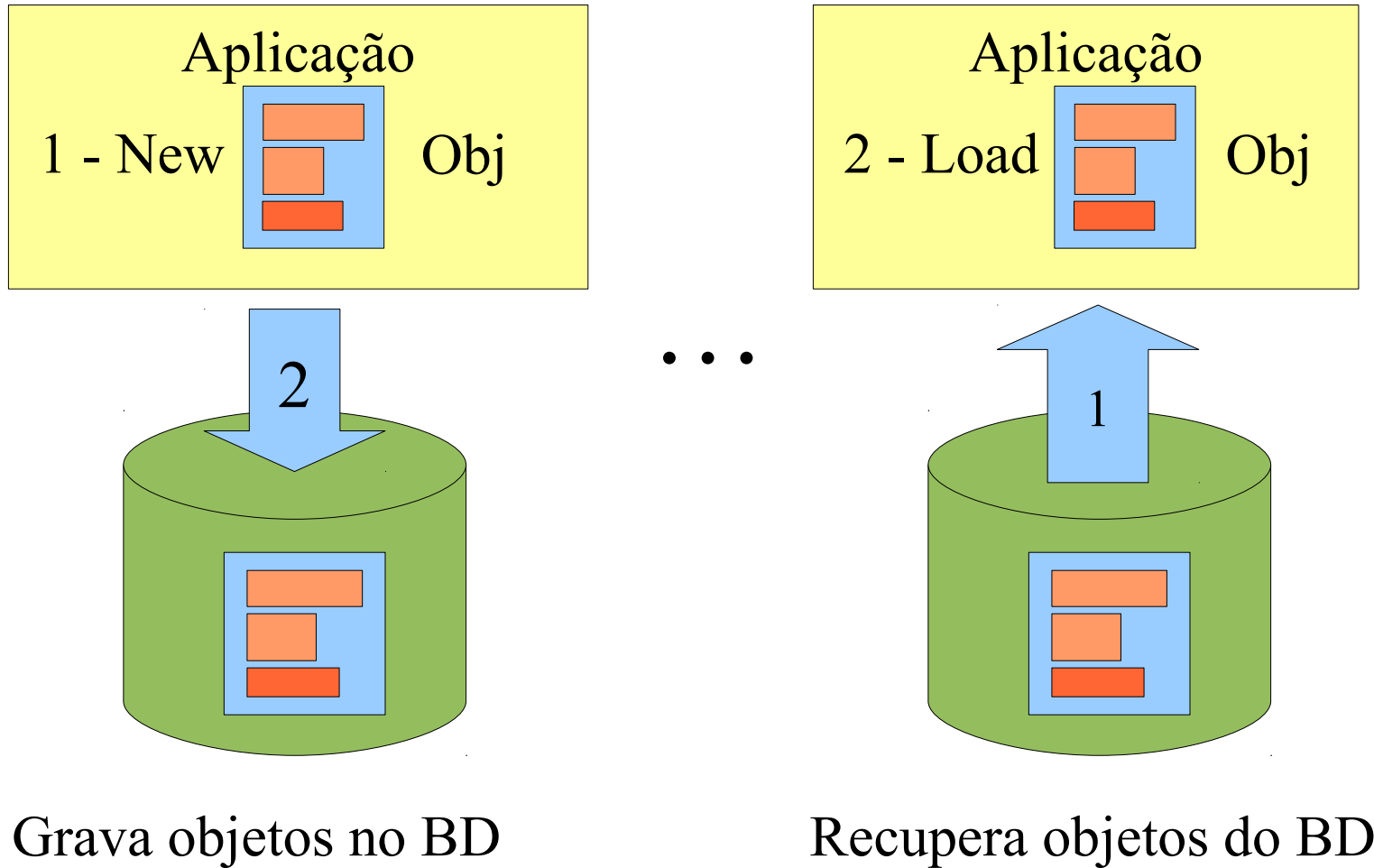
Evandro César Freiburger

`evandrofreiberger@gmail.com`

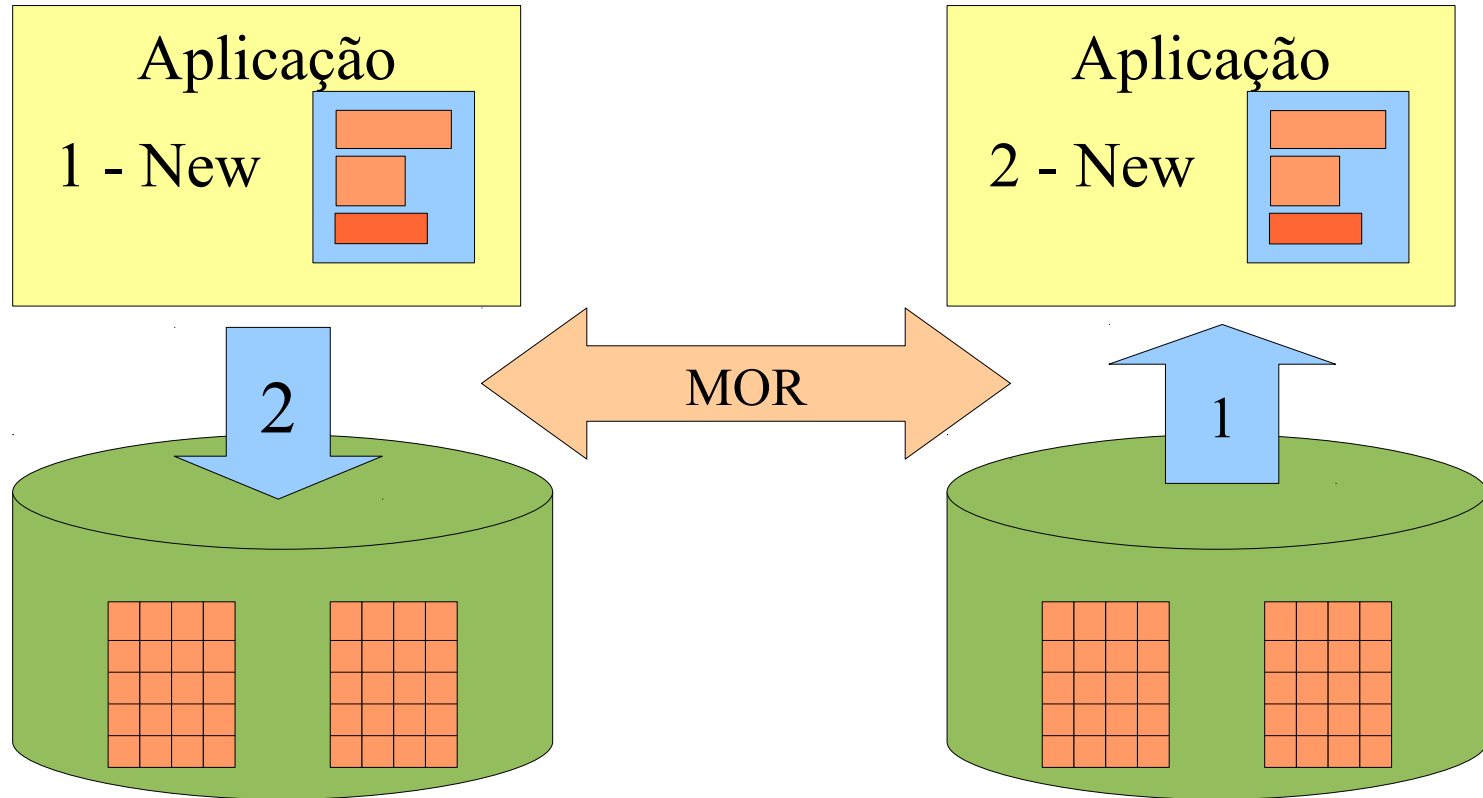
Persistência

- ✓ Persistência em aplicações orientadas a objetos
 - Permite que o objeto sobreviva ao processo que o criou
 - O estado do objeto é armazenado em um meio persistente
 - O estado do objeto possa ser recuperado em algum momento no futuro
 - O estado é recuperado em um novo objeto (Instância) criado a partir dos dados armazenados

Mundo Ideal



Mundo Real



Converte Objeto em Tabela
e grava no BD

Recupera da Tabela e
Converte para Objeto

Persistência em RDB

- ✓ RDB – Banco de Dados Relacional (é bom em)
 - Gerenciar grandes quantidades de dados
 - Pesquisa, ordenação
 - Gerenciar conjuntos de dados
 - Junção, agregação, projeção
 - Compartilhamento
 - Concorrência (Transações)
 - Muitas aplicações
 - Integridade
 - Restrições
 - Isolamento entre as Transações

Persistência em RDB

- ✓ RDB – Banco de Dados Relacional (**não é bom**)
 - Modelagem (ausência de)
 - Polimorfismo
 - Herança
 - Granularidade fina
 - Lógica de Negócio
 - Stored Procedures
 - Distribuição
 - Não tão madura quanto a de aplicações

Aplicação OO

- ✓ Uma aplicação OO possui em sua camada de negócio um modelo de domínio
- ✓ Não manipula diretamente dados tabulares dos bancos RDB
- ✓ Possui seu próprio modelo orientado a objetos das entidades de negócio
- ✓ O modelo orientado a objeto é uma rede de objetos interconectados

Persistência de Objetos

✓ Disparidade entre Modelo OO e o Relacional

- Diferença da granularidade dos elementos
- Diferença de subtipos
- Diferença da identidade dos elementos
- Diferença de associações entre elementos
- Diferença de navegabilidade entre elementos

ORM – Mapeamento Objeto Relacional

- ✓ Pode ser realizado manualmente
 - programando-se todos os detalhes do mapeamento
- ✓ Pode ser realizado de forma automatizada
 - com uso de API's ou Framework

ORM – Mapeamento Objeto Relacional

✓ Forma automatizada

- API para as principais operações CRUD
- Linguagem de Consulta / API
- Facilidade na especificação dos metadados
- Recursos para interagir com objetos transacionais

ORM – Mapeamento Objeto Relacional

- ✓ Os Dados são Importantes
 - Portanto, o modelo relacional é importante
 - Os dados persistirão mais do que as aplicações
- ✓ Utilize as vantagens que um banco de dados relacional proporciona
- ✓ Sem deixar de lado a linguagem orientada a objetos

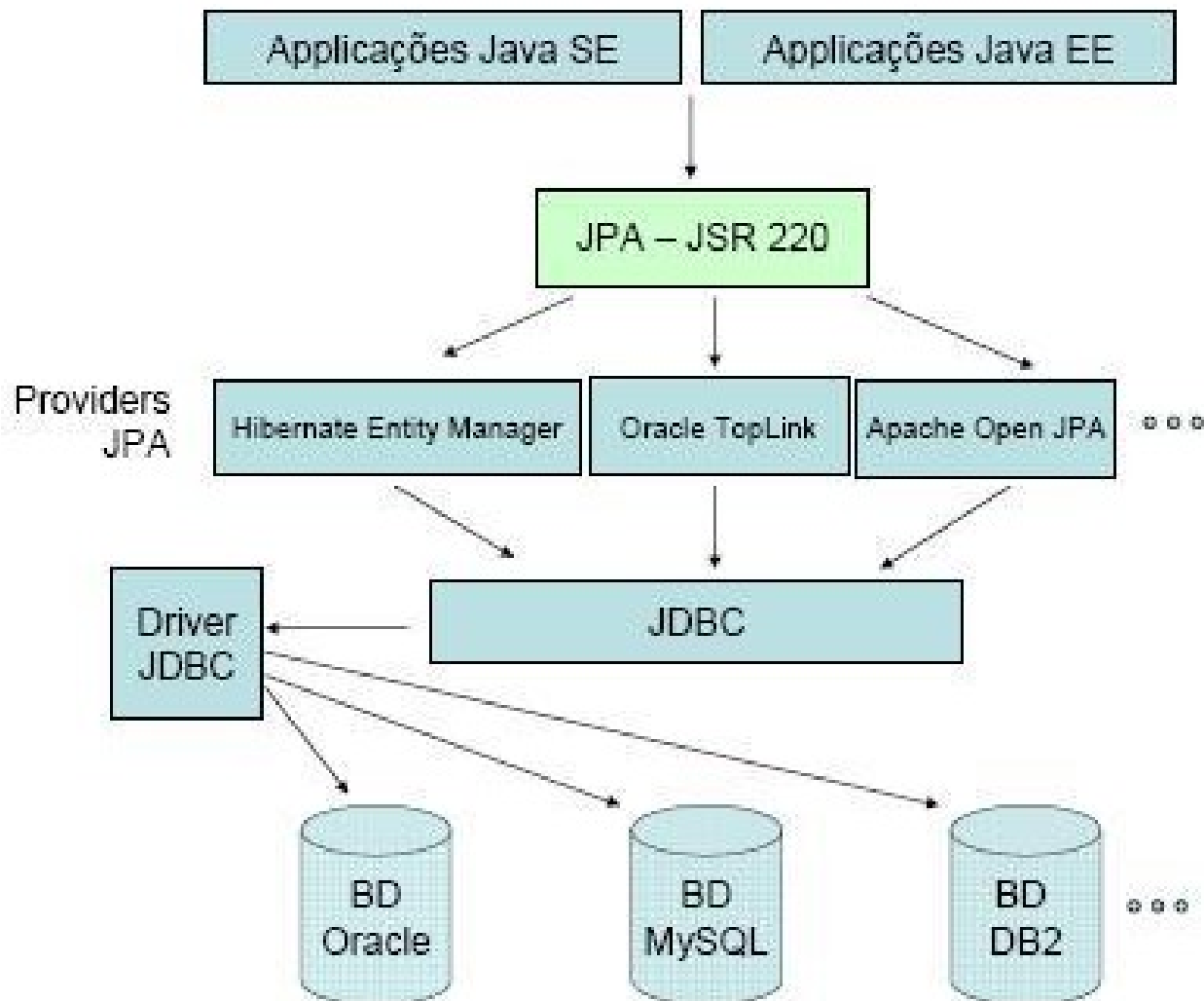
ORM – Mapeamento Objeto Relacional

- ✓ Granularidade Fina
 - Mais classes do que tabelas
- ✓ Granularidade Fina na OO é bom
 - Maior possibilidade de reuso
 - Type Safe (tipo seguro)
 - Melhor encapsulamento

JPA – Java Persistence API

- ✓ Padrão desenvolvido para Mapeamento Objeto/Relacional
- ✓ Tem origem no projeto do Framework Hibernate
- ✓ JPA 1.0 – JSR 220
- ✓ JPA 2.0 – JSR 317

ORM – Arquitetura da JPA



ORM – Exemplos de Frameworks/API's

- ✓ Oracle/Apache JDO - Java Data Objects
- ✓ Hibernate – Framework da JBOSS
- ✓ Apache OJB – Object Relational Bridge
- ✓ Oracle Toplink – JPA da Oracle
- ✓ OpenJPA – JPA do grupo Apache
- ✓ EclipseLink – JPA do Eclipse.org