

Objektově-relační mapování

Miroslav Beneš



Obsah přednášky

- Význam objektově-relačního mapování (ORM)
- Dědičnost
- Strukturovaná data
- Vztahy mezi objekty
- Nástroje pro ORM
 - JDO
 - Hibernate

Objektově-relační mapování

2

Význam ORM

- „Impedance mismatch“ – nesoulad mezi objektovými technologiemi a relačním uložením dat
- Vrstva ORM odděluje logický a fyzický datový model
- Mapování objektového modelu na fyzické databázové schéma
 - manuální
 - automatizované
 - smíšené
- Mapování má velký vliv na efektivitu aplikací

Objektově-relační mapování

3

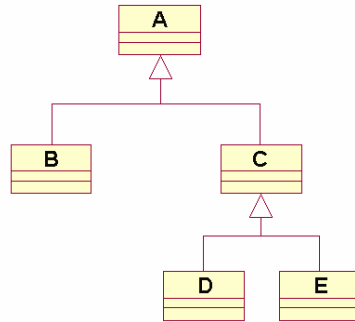
Dědičnost

- Sdílení společné struktury a chování
- Možnost opakovaného využití části implementace společných částí
- Možnost generického přístupu k částem hierarchie objektů
 - Např. provedení akce nad všemi osobami
- Abstraktní / konkrétní třídy

Objektově-relační mapování

4

Příklad hierarchie tříd



Objektově-relační mapování

5

Mapování dědičnosti

- **Horizontální mapování**
 - Tabulka pro každou konkrétní třídu
 - Obsahuje atributy všech bazových tříd
- **Vertikální mapování**
 - Tabulka pro každou konkrétní i abstraktní třídu
 - Získání informace o objektu vyžaduje přístup do více tabulek
- **Mapování pomocí unie**
 - Společná tabulka pro více tříd
 - Neefektivní uložení dat

Objektově-relační mapování

6

Horizontální mapování

B	id	a	b	
D	id	a	c	d
E	id	a	c	e

Vyber všechny hodnoty a (např. login a jméno všech osob)
`(SELECT a FROM B) UNION (SELECT a FROM D) UNION (SELECT a FROM e);`

Vyber konkrétní hodnotu typu D
`SELECT * FROM D WHERE id=100;`

Vytvoř nový objekt typu B
`INSERT INTO B(id, a, b) VALUES(100, 'a', 'b');`

Objektově-relační mapování

7

Vertikální mapování

A	id	a
C	id	c
E	id	e

Vyber všechny hodnoty a (např. login a jméno všech osob)
`SELECT * FROM A;`

Vyber konkrétní hodnotu typu E
`SELECT * FROM (A JOIN C JOIN E) WHERE A.id=100;`

Vytvoř nový objekt typu B
`INSERT INTO A(id,a) VALUES(100, 'a');`
`INSERT INTO B(id,b) VALUES(100, 'b');`

Objektově-relační mapování

8

Mapování pomocí unie

T

id	type	a	b	c	d	e
----	------	---	---	---	---	---

Vyber všechny hodnoty a (např. login a jméno všech osob)
SELECT a FROM T

Vyber konkrétní hodnotu typu E
SELECT a,c,e FROM T WHERE id=100;

Vyber všechny hodnoty typu E
SELECT id,a,c,e FROM T WHERE type='E';

Vytvoř nový objekt typu B
INSERT INTO T(id, type, a, b) VALUES(100, 'B', 'a', 'b');

Objektově-relační mapování

9

Mapování pomocí unie

T1

id	type	a	b
----	------	---	---

T2

id	type	a	c	d	e
----	------	---	---	---	---

Vyber všechny hodnoty a (např. login a jméno všech osob)
(SELECT id,a FROM T1) UNION (SELECT id,a FROM T2)

Vyber konkrétní hodnotu typu E
SELECT a,c,e FROM T2 WHERE id=100;

Vyber všechny hodnoty typu E
SELECT id,a,c,e FROM T2 WHERE type='E';

Vytvoř nový objekt typu B
INSERT INTO T1(id, type, a, b) VALUES(100, 'B', 'a', 'b');

Objektově-relační mapování

10

Dědičnost - hodnocení

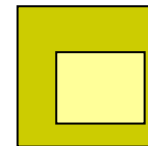
- **Horizontální mapování**
 - Složité dotazy nad abstraktními třídami (union)
- **Vertikální mapování**
 - Složitější výběr konkrétního objektu (join)
 - Příliš mnoho tabulek
 - Flexibilita – vhodné pro návaznost na existující data
- **Mapování pomocí unie**
 - Rozumné množství tabulek
 - Nutnost zavedení informace o typu
 - Větší prostorové nároky

Objektově-relační mapování

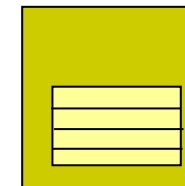
11

Strukturovaná data

- Zanořený objekt (např. adresa osoby)



- Kolekce (např. kontakty)

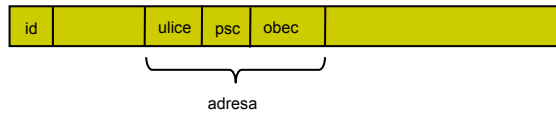


Objektově-relační mapování

12

Zanořené objekty

- Vložení atributů do vnějšího objektu



- Osamostatnění zanořeného objektu



Objektově-relační mapování

13

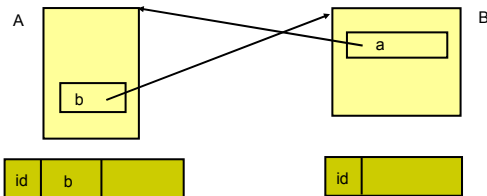
Vztahy

- **1:1**
 - přímá reference na objekt
 - cizí klíč = primární klíč druhého objektu
 - může být obousměrný vztah
- **1:N**
 - reference na straně N
 - vazební tabulka – není nutná
- **M:N**
 - vazební tabulka – cizí klíče do obou svázaných tabulek

Objektově-relační mapování

14

Vztah 1:1



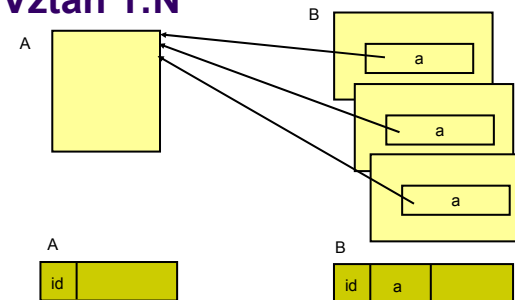
Vyber hodnotu B příslušnou konkrétní hodnotě A
SELECT B.* FROM A,B WHERE A.b=B.id AND A.id=100

Vyber hodnotu A příslušnou konkrétní hodnotě B
SELECT A.* FROM A,B WHERE A.b=B.id AND B.id=100

Objektově-relační mapování

15

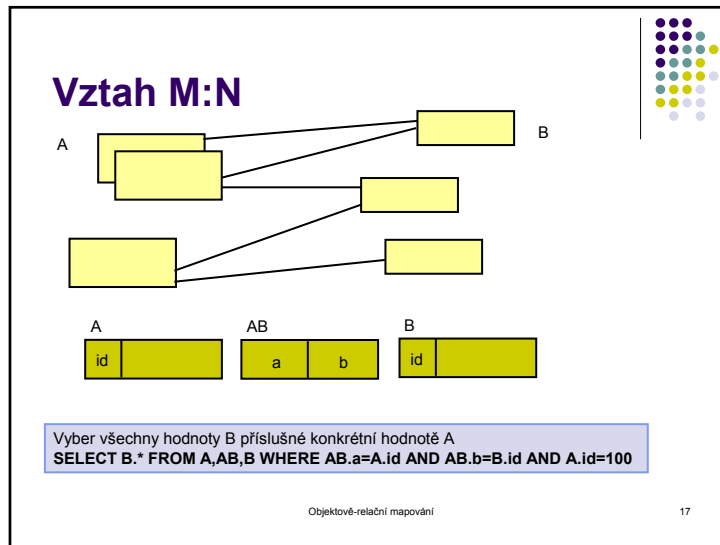
Vztah 1:N



Vyber všechny hodnoty B příslušné konkrétní hodnotě A
SELECT B.* FROM A,B WHERE B.a=A.id AND A.id=100

Objektově-relační mapování

16



Použití nástrojů pro ORM

1. Návrh objektového modelu
 - např. UML
2. Implementace modelu
 - např. Java
3. Doplnění podpůrných metod
 - Perzistence - vytvoření, uložení, načtení, zrušení objektu
 - Transakce
 - Referenční integrita

Objektově-relační mapování 18

Přístupy k automatizaci ORM

- **Modifikace na úrovni zdrojového textu**
 - Dědičnost – rozšíření o metody zajišťující perzistenci
 - J2EE CMP (Container Managed Persistence)
- **Modifikace vygenerovaných tříd**
 - JDO (Java Data Objects) – Sun JSR 12
- **Mapování za běhu pomocí reflexe**
 - Hibernate

Objektově-relační mapování 19

Další informace

- <http://java.sun.com/products/jdo/index.jsp>
- <http://www.hibernate.org/>

Objektově-relační mapování 20