MLDN 魔乐在线
IT电子教育门户

Hibernate

实体层设计

讲师：李兴华

MLDN 魔乐在线
IT电子教育门户

本章内容

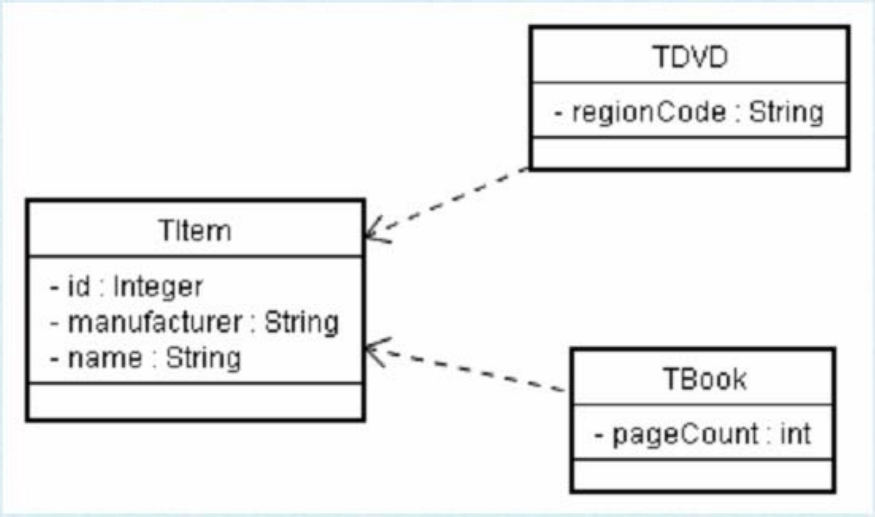
- Hibernate的实体层设计

MLDN 魔乐在线
IT电子教育门户

实体层设计

- Hibernate中支持3种类型的继承形式：
  - **Table per concrete class**
    - 表与子类之间的独立一对一关系
  - **Table per subclass**
    - 每个子类对应一张子表，并与主类共享主表
  - **Table per class hierarchy**
    - 表与类的一对多关系

MLDN 魔乐在线
IT电子教育门户



那么，对于如上的类层次关系，反映到数据库的映射逻辑上是怎样的形式？下面就围绕这个实例，对Hibernate中关于继承关系的3种映射类型进行讨论

MLDN 魔乐在线
IT电子教育门户

Table per concrete class

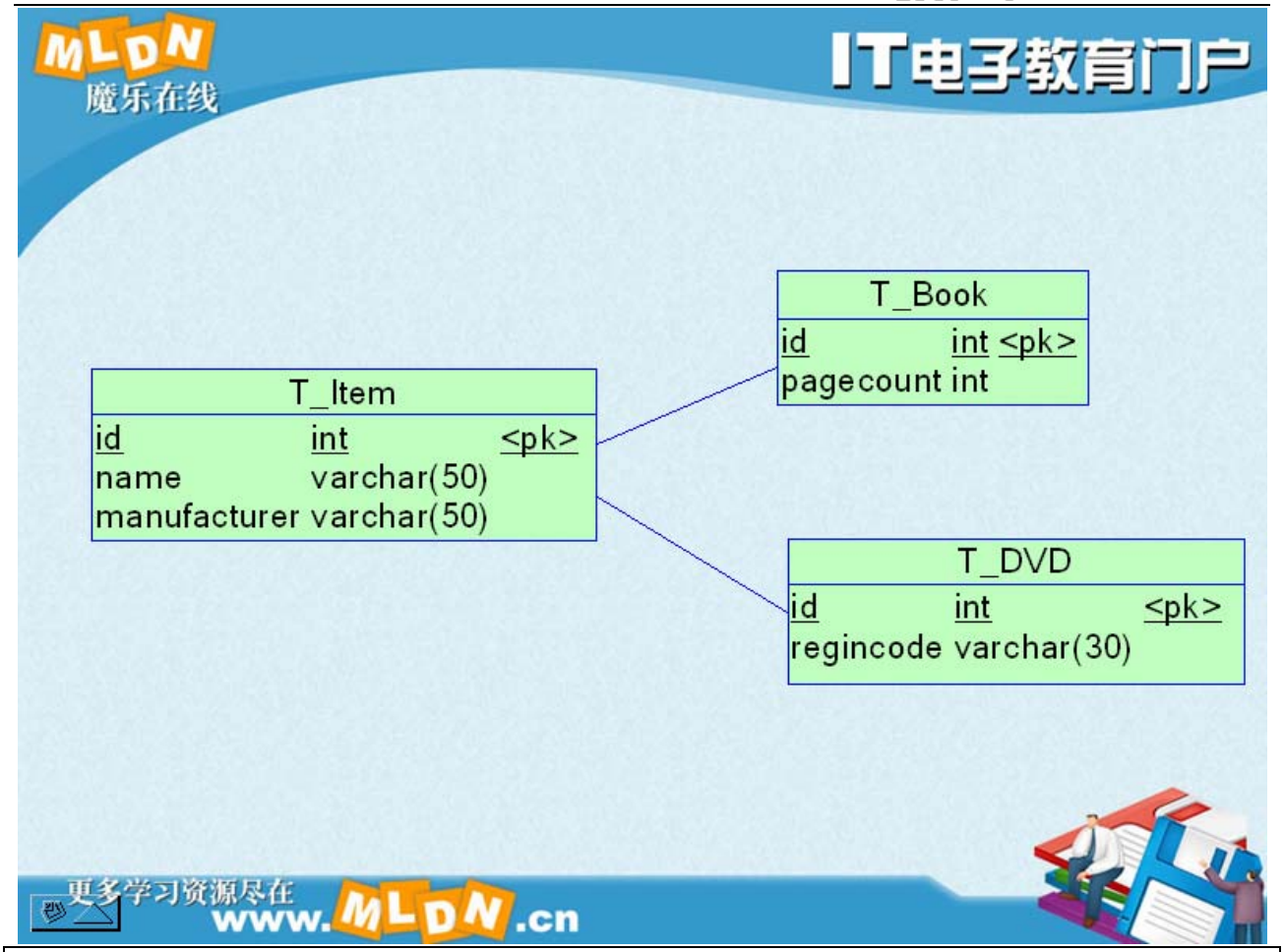
- 在这个实例中，Titem有两个子类：TBook、TDVD。那么，所谓“表与子类之间的独立一对一关系”，也就是每个子类对应一张数据库表。对应TBook和TDVD。有以下库表 。

此种设计与之前的设计相似，需要有两个*.hbm.xml 文件

MLDN 魔乐在线
IT电子教育门户

Table per subclass

- 在Table per concrete class中，提到过一个问题：“由于T_BOOK、T_DVD表中的父类字段必须保持一致，如果父类TItem发生变动，那么所有的子类表必须同时进行修改。”
- 那么自然会想到，是否可以将父类TItem单独映射到一张主表，而为TBook、TDVD表分别单独设立一张子表，子表中只包含子类所扩展的属性，同时，子表和父表通过关系型数据库的外键相关联，如下图所示：




```
package org.lxh.hibernate04;

public class TItem {
	private String id ;
	private String name ;
	private String manufacturer ;
	public String getId() {
		return id;
	}
	public void setId(String id) {
		this.id = id;
	}
	public String getManufacturer() {
		return manufacturer;
	}
	public void setManufacturer(String manufacturer) {
		this.manufacturer = manufacturer;
	}
	public String getName() {
		return name;
```

```
    }
    public void setName(String name) {
        this.name = name;
    }
}
```

```
package org.lxh.hibernate04;

public class TBook extends TItem {
    private int pageCount ;

    public int getPageCount() {
        return pageCount;
    }

    public void setPageCount(int pageCount) {
        this.pageCount = pageCount;
    }
}
```

```
package org.lxh.hibernate04;

public class TDVD extends TItem {
    private String regionCode ;

    public String getRegionCode() {
        return regionCode;
    }

    public void setRegionCode(String regionCode) {
        this.regionCode = regionCode;
    }
}
```

POJO → *.hbm.xml → 表

整个映射文件只有一个，虽然有三个表，但映射文件只需要一个

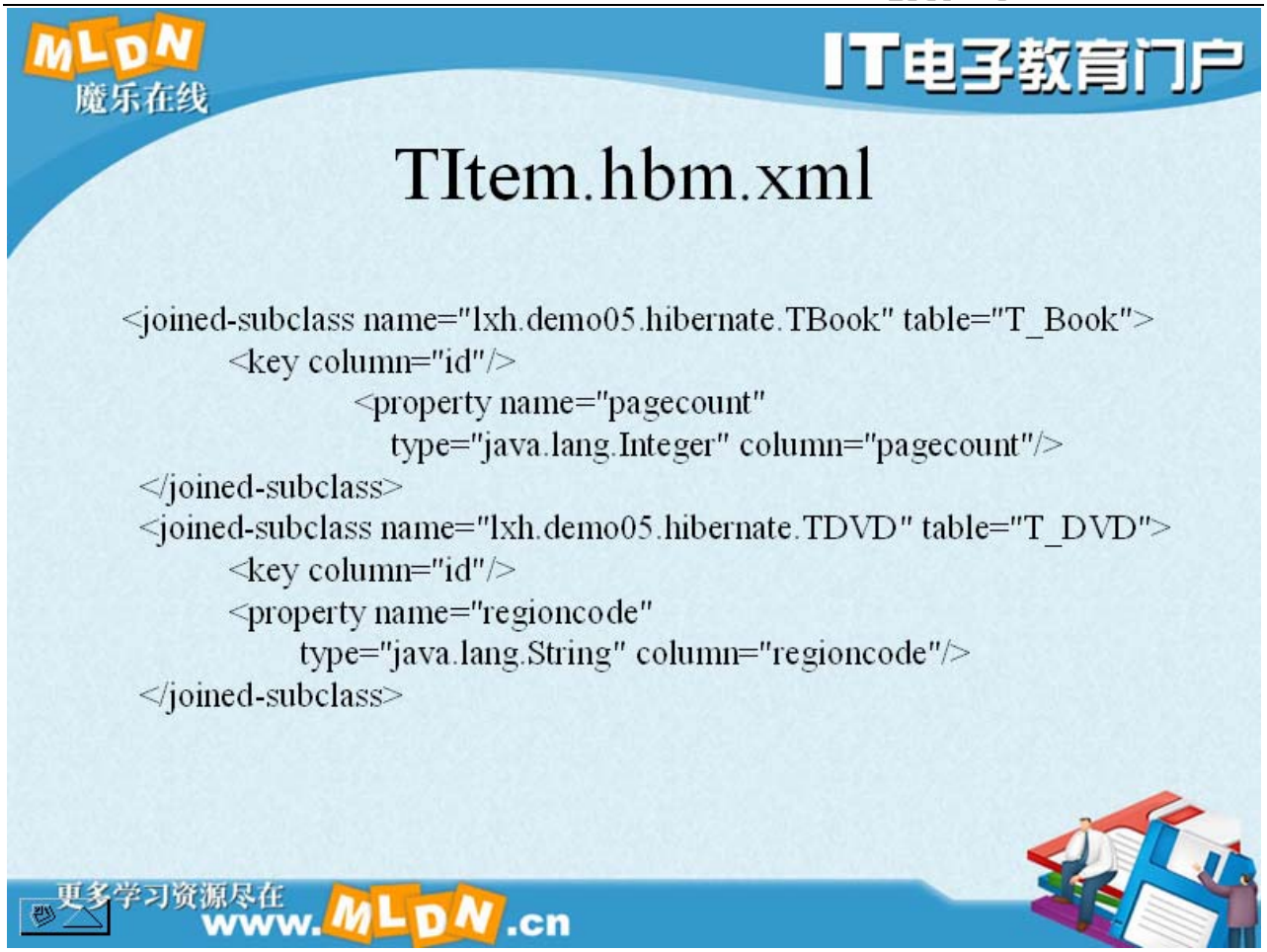
MLDN 魔乐在线
IT电子教育门户

TItem.hbm.xml

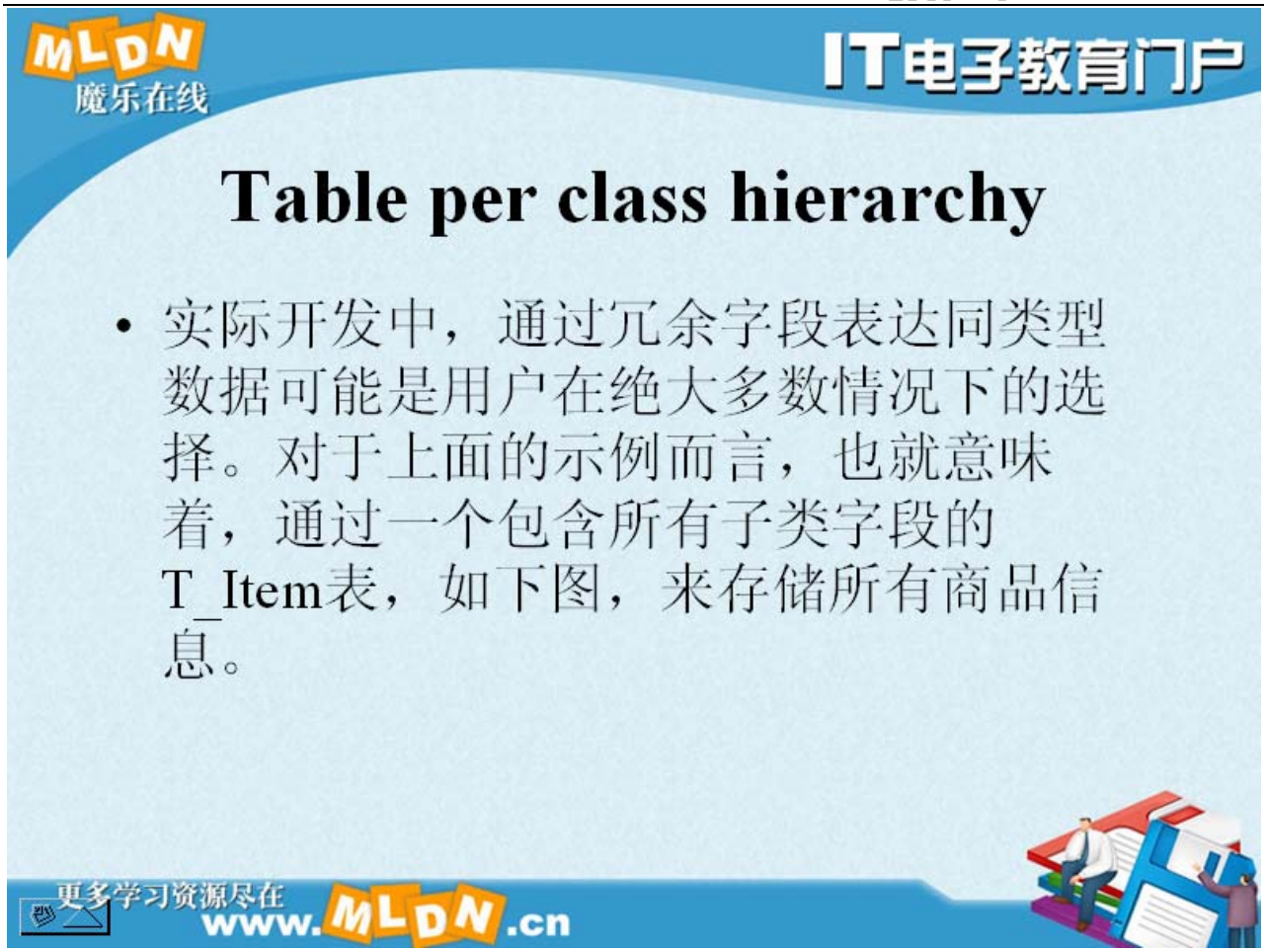
```
<hibernate-mapping>
<class name="lxh.demo05.hibernate.TItem" table="T_Item">
  <id name="id" type="java.lang.Integer" column="id">
    <generator class="assigned" />
  </id>
  <property name="name" type="java.lang.String"
    column="name" length="50"/>
  <property name="manufacturer" type="java.lang.String"
    column="manufacturer" length="50"/>
  <joined-subclass name="lxh.demo05.hibernate.TBook" table="T_Book">
        ...
  </joined-subclass>
  <joined-subclass name="lxh.demo05.hibernate.TDVD" table="T_DVD">
        ...
  </joined-subclass>
</class>
</hibernate-mapping>
```

续。。。

更多学习资源尽在 www.MLDN.cn

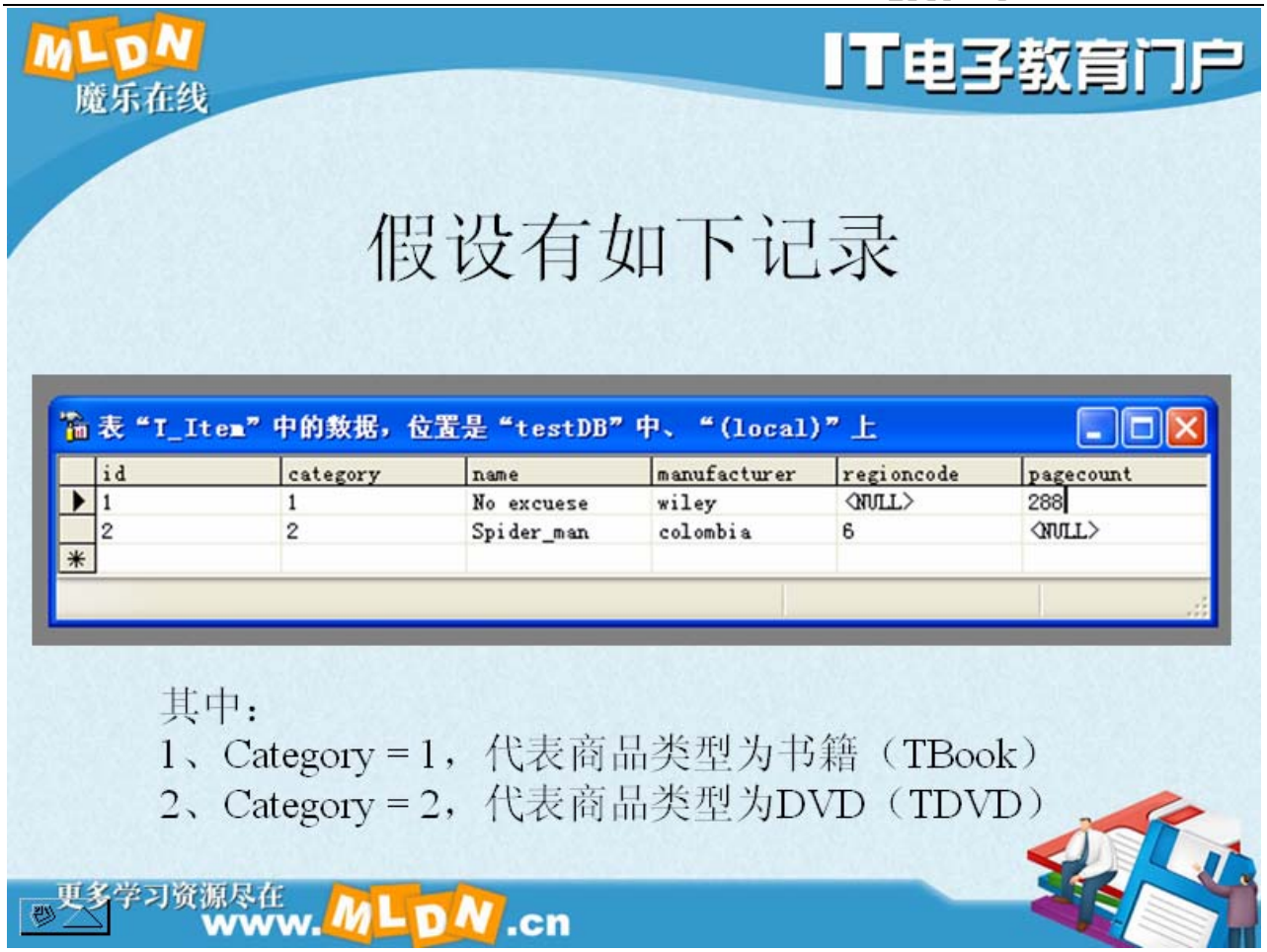


考虑，这样的设计真的是唯一的吗？
此种设计不是解决此类问题的唯一标准



希望只用一张表完成信息存储，这样一来，则意味着在 TItem 表中要存放书和 DVD


MLDN 魔乐在线
IT电子教育门户
T_Item
id int <pk>
category varchar(10)
name varchar(50)
manufacturer varchar(50)
regioncode varchar(30)
pagecount int
更多学习资源尽在 www.MLDN.cn



在 hibernate 中映射文件的编写是较为复杂的

通过 category 字段用来区分不同的操作，但此操作只是在数据库层次上的，与程序无关
此代码需要在*.hbm.xml 文件中进行编写

MLDN 魔乐在线
IT电子教育门户

本章重点

- 掌握Hibernate的如下两种映射策略：
  - Table per subclass
  - Table per class hierarchy