MLDN 魔乐在线
IT电子教育门户

Hibernate

实体映射策略、复合主键

讲师：李兴华

MLDN 魔乐在线
IT电子教育门户

本章内容

- Hibernate实体映射策略
- Hibernate复合主键映射策略

MLDN 魔乐在线
IT电子教育门户

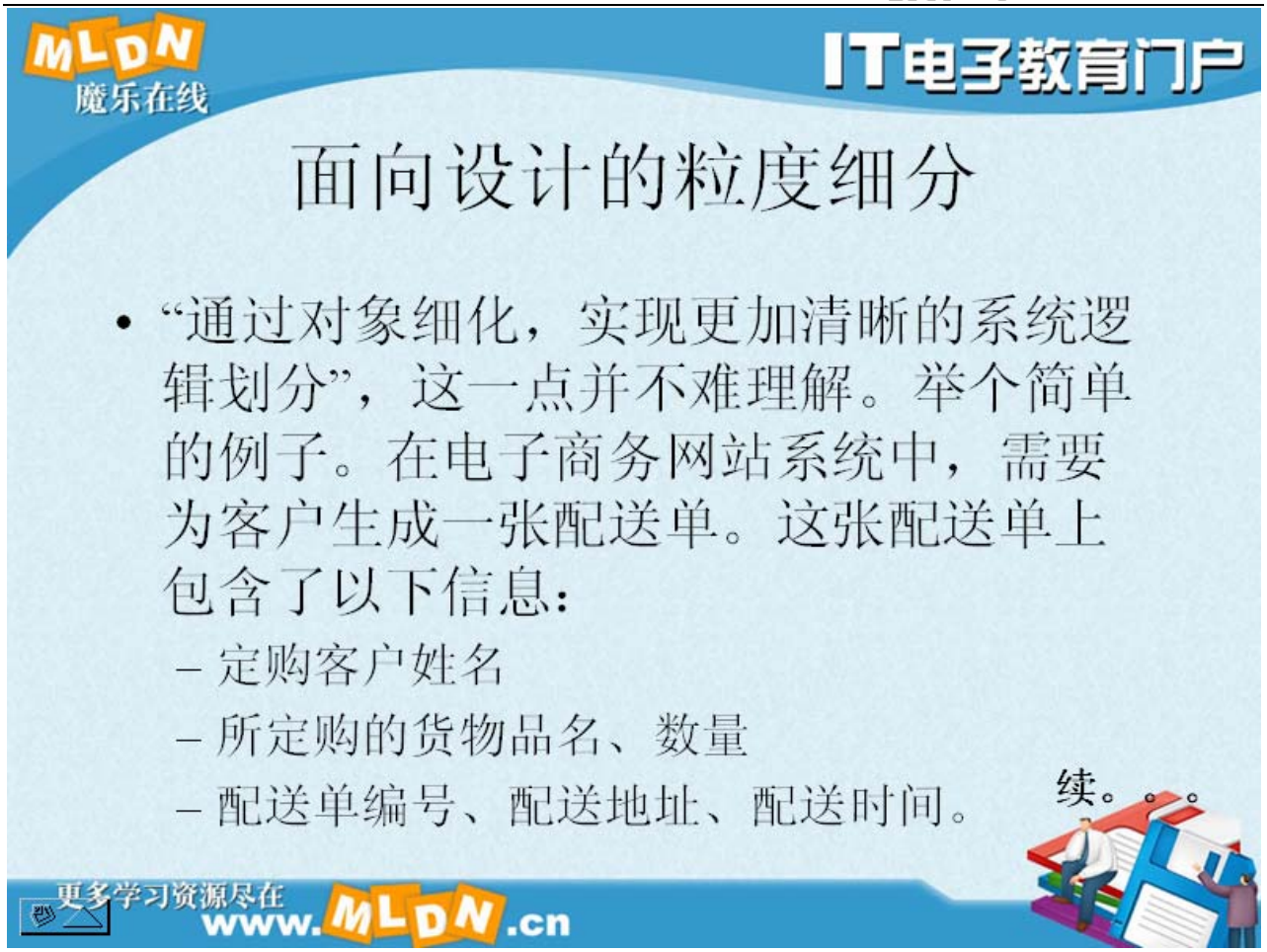
实体粒度设计

- 在Hibernate世界里，经常听到一个术语“fine-grained object model”。什么是find-grained object model，直译比较困难，大致上是“适当的细粒度对象模型”。
- “细粒度”，这就比较好理解。通俗来讲，所谓细粒度模型，就是将原本业务模型中的对象加以细分，从而得到更加精彩的对象模型（直白一点来说，也就是划分出更多的对象）。

MLDN 魔乐在线
IT电子教育门户

细粒度对象模型

- 对于Hibernate而言，所谓细粒度对象模型，主要是针对实体类设计的对象细分。对象的细分主要出于两个目的：
  - 面向设计的粒度细分
    - 通过对象细化，实现更加清晰的系统逻辑。
  - 面向性能的粒度细分
    - 针对业务逻辑，通过合理的细粒度对象，提高系统的能耗比（性能/资源消耗）。



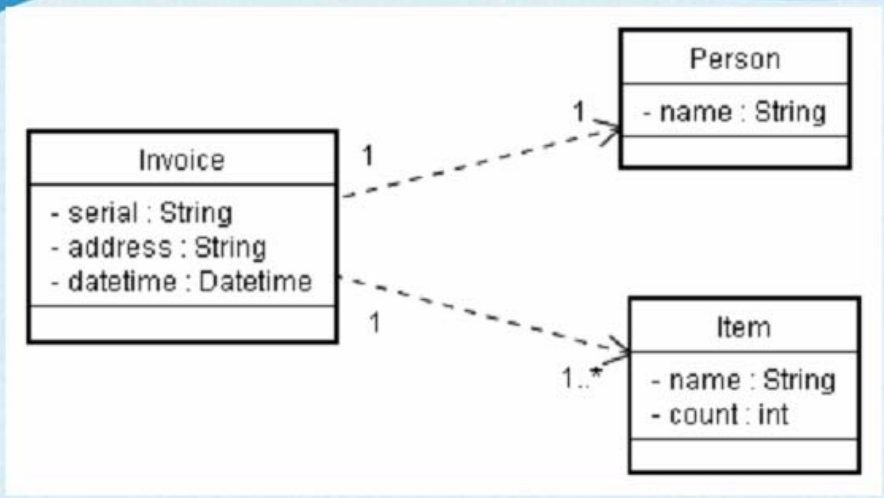
给客户的是一个完整的定单，但是定单会有若干个子模块组成

MLDN 魔乐在线
IT电子教育门户

面向设计的粒度细分

- 对于上面这个例子，通过一个大型的“配送单”对象容纳所有的属性，这个方案在设计的角度上来讲并不可取，可以将其划分为3个对象，如下图所示：
  - Invoice（配送单）
  - Item（定购项目）
  - Person（客户）

MLDN 魔乐在线
IT电子教育门户



可以看到，通过对“配送单对象”的细分，体现出了更加清晰和合理的设计逻辑。

在这个实例中，对象细分更多地体现在实体的关联上，如对于上面的3个类，一般情况下，数据库中会有与之对应的3张表，通过表之间的关联逻辑组合成最终的发货单。

MLDN 魔乐在线
IT电子教育门户

实体映射策略

- 对于表的对象细分，在Hibernate中可借助Component节点的定义完成。
- 何谓Component？从名字上看，既然称之为组件，则其必然是从属于某个整体的一个组成部分。在Hibernate语义中，将某个实例对象中的一个逻辑组成称为Component。
- Component与实体对象的根本差别，就在于Component没有标识（identity），它作为一个逻辑组成，完全从属于实体对象。

更多学习资源尽在 www.MLDN.cn

MLDN 魔乐在线
IT电子教育门户

person		
id	int	<pk>
firstname	varchar(50)	
lastname	varchar(50)	
address	varchar(200)	
zipcode	varchar(10)	
tel	varchar(20)	

更多学习资源尽在 www.MLDN.cn

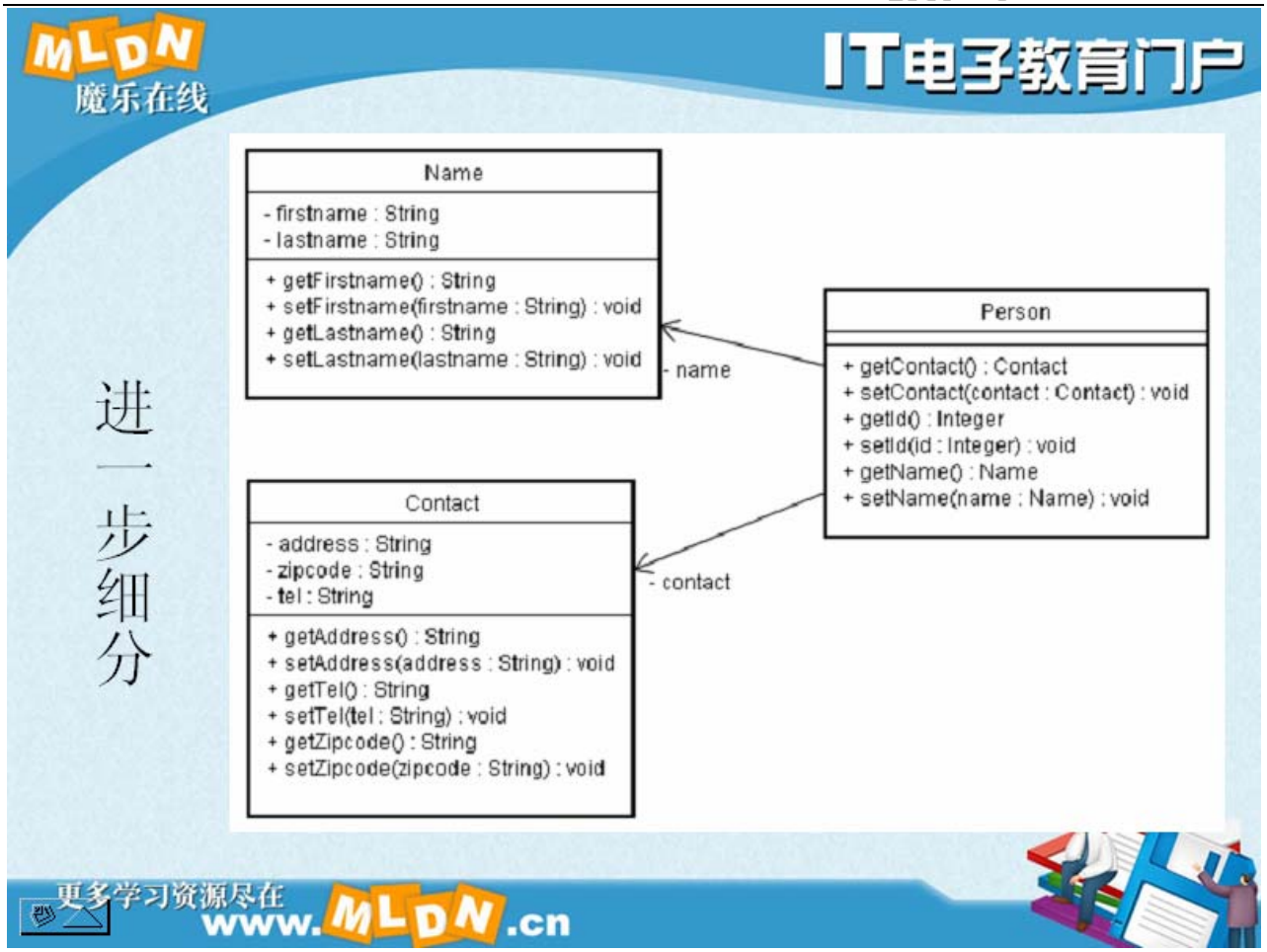
MLDN 魔乐在线
IT电子教育门户

```
public class Person {
        private Integer id;
        // 姓名
        private String firstname;
        private String lastname;
        // 联系方式
        private String address;
        private String zipcode;
        private String tel;
        // setter..getter..
}
```

可以看到，Person中包含了两个逻辑组成部分：

1、姓名

2、联系方式，包括地址，邮编，电话



☆ 所谓的细粒度划分，是指在 POJO 类上的面向对象的划分，而不在于表的划分上

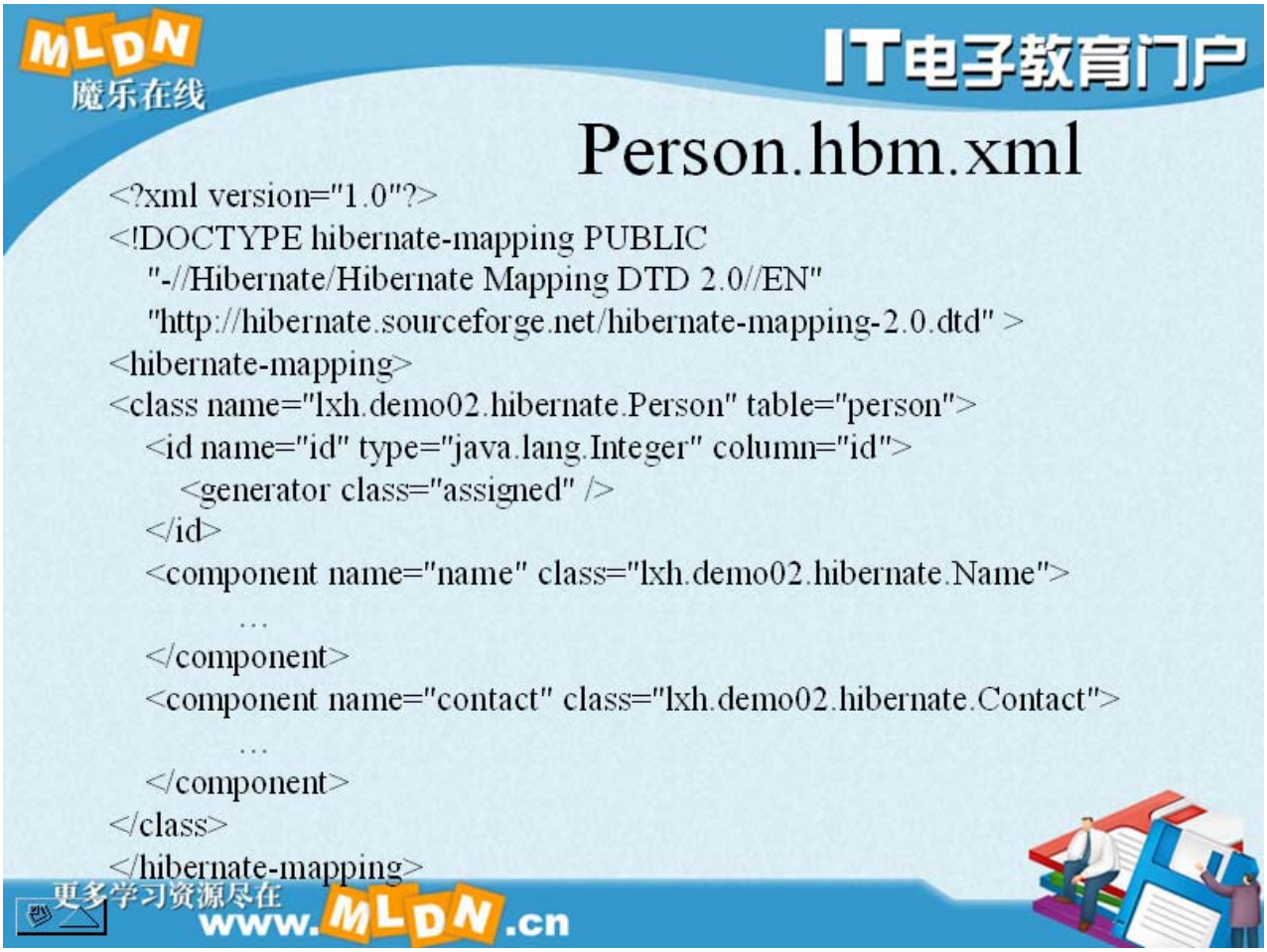
数据库创建脚本：

```
-- 创建 person 表

-- 删除 Person 表
DROP TABLE person ;

-- 创建 Person 表
CREATE TABLE person
(
    id int not null primary key ,
    firstname varchar(20) not null ,
    lastname varchar(20) not null ,
    address varchar(20) not null ,
    zipcode varchar(6) not null ,
    tel varchar(20)
) ;
```

```
-- 事务的提交
commit ;
```



配置文件：

```
<?xml version="1.0" encoding='UTF-8'?>
<!DOCTYPE hibernate-mapping PUBLIC
                            "-//Hibernate/Hibernate Mapping DTD 3.0//EN"
                            "http://hibernate.sourceforge.net/hibernate-mapping-3.0.dtd" >

<!-- DO NOT EDIT: This is a generated file that is synchronized -->
<!-- by MyEclipse Hibernate tool integration.                   -->
<!-- Created Sat Nov 11 20:51:28 CST 2006                       -->
<hibernate-mapping package="org.lxh.hibernate">

    <class name="Person" table="PERSON">
        <id name="id" column="ID" type="long">
            <generator class="assigned" />
        </id>
        <component name="name" class="org.lxh.hibernate.Name">
            <property name="firstname" column="FIRSTNAME" type="string" not-null="true" />
            <property name="lastname" column="LASTNAME" type="string" not-null="true" />
        </component>
```

```
        <component name="contact" class="org.lxh.hibernate.Contact">
            <property name="address" column="ADDRESS" type="string" not-null="true" />
            <property name="zipcode" column="ZIPCODE" type="string" not-null="true" />
            <property name="tel" column="TEL" type="string" />
        </component>
    </class>

</hibernate-mapping>
```