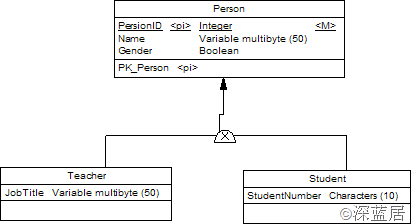
**1. 概述**

在使用PowerDesigner设计RDBMS数据库的时候，发现其中的子类貌似与课程老师所讲的不太一样，查资料发现，关于继承就有三种实现方式。

* Concrete Table Inheritance（具体表继承）
* Single Table Inheritance（单表继承）
* Class Table Inheritance（类表继承）

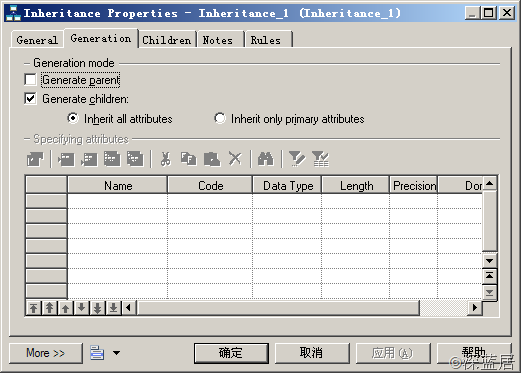
**2.分类**

比如在一个教务系统中，有老师学生2个对象，这两个对象都是“人”对象的子类，所以我们可以建立一个Person表，该表有人的公共属性：姓名、性别等，还有就是数据的唯一标识，一个ID。而教师对象有教师的特有属性，比如职称，学生有学生的特有属性，比如学号。所以我们可以建立Person、Teacher、Student3个表，其关系在PowerDesigner中如图所示：



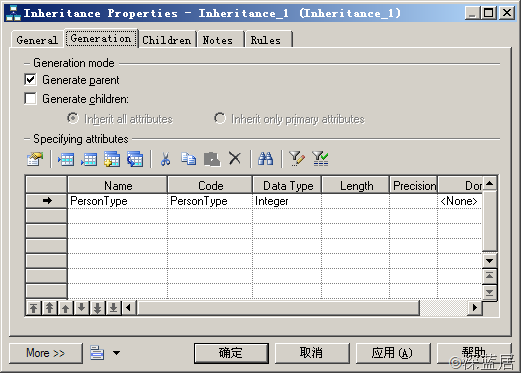
**2.1 Concrete Table Inheritance**

不建立父对象，将父对象的所有属性转移到子对象中，为每个子对象建立对于的表。如果使用这种方法，那么就只需要建立Teacher表和Student表，不需要Person表，在PowerDesigner中，双击继承节点，打开属性窗口，取消“Generate Parent”选项，选中“Generate children”并选择“Inherit all attributes”，如图所示：

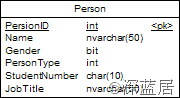


**2.2 Single Table Inheritance**

在一个宽表中列出所有父对象和子对象的属性，同时用一个标识列表示该行数据存储的是哪个子类的数据。在PowerDesigner中，修改继承节点的属性，取消“Generate children”，选中“Generate parent”，然后在下面添加一个标识列，叫PersonType，如图所示：



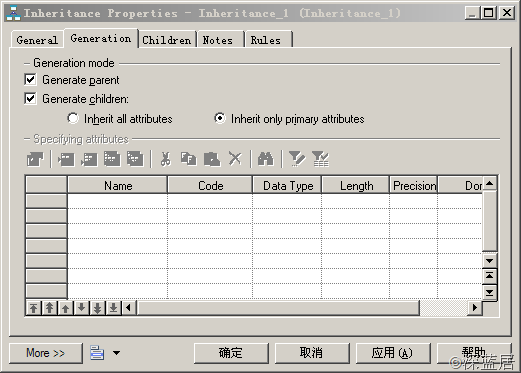
生成的数据库表，在一个宽表中表示如图所示：



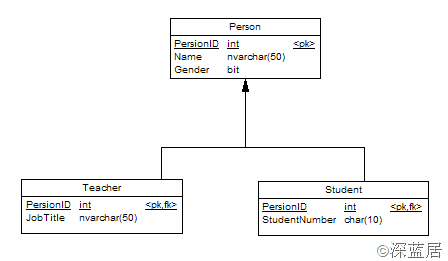
可以看到Person中的列集成了Person、Teacher、Student这3个表的所有列，同时还多了一个列PersonType，这个列就是用来区分这行数据到底表示的是一个学生还是一个老师。

**2.3 Class Table Inheritance**

对父对象和每个子对象建立一个对应的表，然后在子表中设置该子表的主键为与父表关联的外键。在PowerDesigner中，对于继承节点的属性，允许生成父和子表，另外，将“Inherit only primary attributes”选中。如图所示：



生成的数据库表如图所示：



这里可以看到，Teacher的主键和Student的主键同时又是该表的外键，连接到Person表。

**3.分析**

* Concrete Table Inheritance方式遵循了面向对象的思想，父表只是作为一个接口，并没有具体的实现方式。
* Single Table Inheritance方式是将本应放在多个子表的属性都放在父表中，个人不是很推荐这种方式，觉得这样的话一方面不是很有利于拓展，另一方面的话也导致数据有一定的冗余。
* Class Table Inheritance方式是课程老师所讲的继承方式，通过这种方式相当于把具有相同属性的数据都存在一张父表中，不同属性的数据再存在不同的子表中。通过这种方式可以方便的查看各个子表之间的相同属性和不同属性，同时如果要添加一些公有属性的话，直接在父表中添加即可，我很钟意，非常方便。