

E-R模型 和 E-R分析

1- ER图分为实体、属性、关系三个核心部分。实体是长方形体现，而属性则是椭圆形，关系为菱形。一般来讲实体都至少有一个唯一属性。

2- ER图中关联关系有三种：

1对1 (1:1) : 1对1关系是指对于实体集A与实体集B，A中的每一个实体至多与B中一个实体有关系；反之，在实体集B中的每个实体至多与实体集A中一个实体有关系。

1对多 (1:N) : 1对多关系是指实体集A与实体集B中至少有N(N>0)个实体有关系；并且实体集B中每一个实体至多与实体集A中一个实体有关系。

多对多 (M:N) : 多对多关系是指实体集A中的每一个实体与实体集B中至少有M(M>0)个实体有关系，并且实体集B中的每一个实体与实体集A中的至少N (N>0) 个实体有关系。

3- 弱实体：(双线菱形表示)

弱实体和强实体的联系必然只有1: N或者1: 1，这是由于弱实体完全依赖于强实体，强实体不存在，那么弱实体就不存在，所以弱实体是完全参与联系的，因此弱实体与联系之间的联系也是用的双线菱形。

学生 ---- 成绩单 [没有学生就没有成绩单]

4- 复合实体(桥接实体)

用户 ---- 订单(没有用户，没有订单；同理，没有商品，也没有订单) -----商品

5- 属性：

- 复合属性：一个属性，由多个属性组合。比如名字属性(= 姓 + 名)

- 多值属性：一个属性有多个取值。比如书，它有多重分类， [双线椭圆]

- 派生属性：一个值由其它值计算出来，一般不在数据库中，如果在数据库中，那么隔一段时间就得更新值。

如，兴趣小组的人数 【虚线椭圆】

- 可选属性：有些值不是必须的。如地址。 【在后面加 (0) 】

- 联系属性：比如，学生-----兴趣小组，中间的联系属性：参加，参加关系上衍生出一种 "职能" 属性。

 职能
 /
学生---参加--兴趣小组

一般来讲M:N的两个实体的联系具有联系属性。 1: 1 和 1: M 中联系属性不是那么重要

8- 1 : N 可以将1的属性合并到里面去

ER图分为实体、属性、关系三个核心部分。实体是长方形体现，而属性则是椭圆形，关系为菱形。

ER图的实体 (entity) 即数据模型中的数据对象，例如人、学生、音乐都可以作为一个数据对象，用长方体来表示，每个实体都有自己的实体成员 (entity member) 或者说实体对象 (entity instance)，例如学生实体里包括张三、李四等，实体成员 (entity member) / 实体实例 (entity instance) 不需要出现在ER图中。

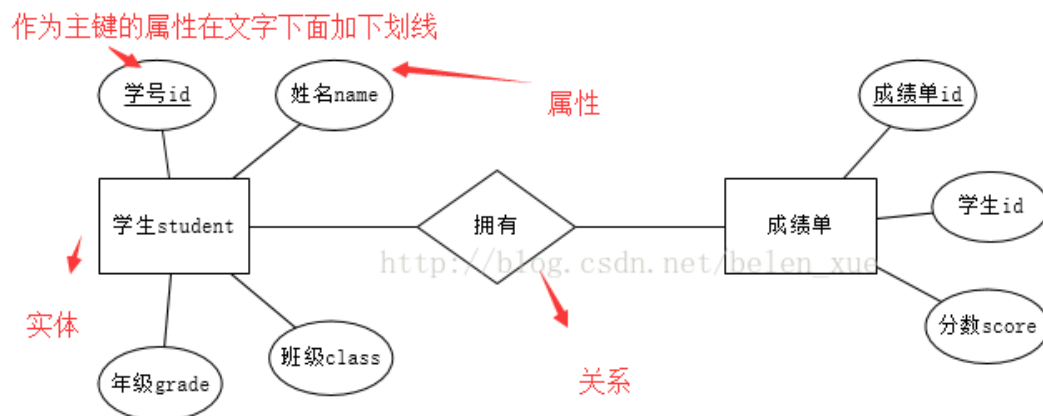
ER图的属性 (attribute) 即数据对象所具有的属性，例如学生具有姓名、学号、年级等属性，用椭圆形表示，属性分为唯一属性 (unique attribute) 和非唯一属性，唯一属性指的是唯一可用来标识该实体实例或者成员的属性，用下划线表示，一般来讲实体都至少有一个唯一属性。

ER图的关系 (relationship) 用来表现数据对象与数据对象之间的联系，例如学生的实体和成绩表的实体之间有一定的联系，每个学生都有自己的成绩表，这就是一种关系，关系用菱形来表示。

ER图中关联关系有三种：

1对1 (1:1) : 1对1关系是指对于实体集A与实体集B，A中的每一个实体至多与B中一个实体有关系；反之，在实体集B中的每个实体至多与实体集A中一个实体有关系。**1对多 (1:N) :** 1对多关系是指实体集A与实体集B中至少有N(N>0)个实体有关系；并且实体集B中每一个实体至多与实体集A中一个实体有关系。**多对多 (M:N) :** 多对多关系是指实体集A中的每一个实体与实体集B中至少有M(M>0)个实体有关系，并且实体集B中的每一个实体与实体集A中的至少N (N>0) 个实体有关系。

下面是个简单的例子：



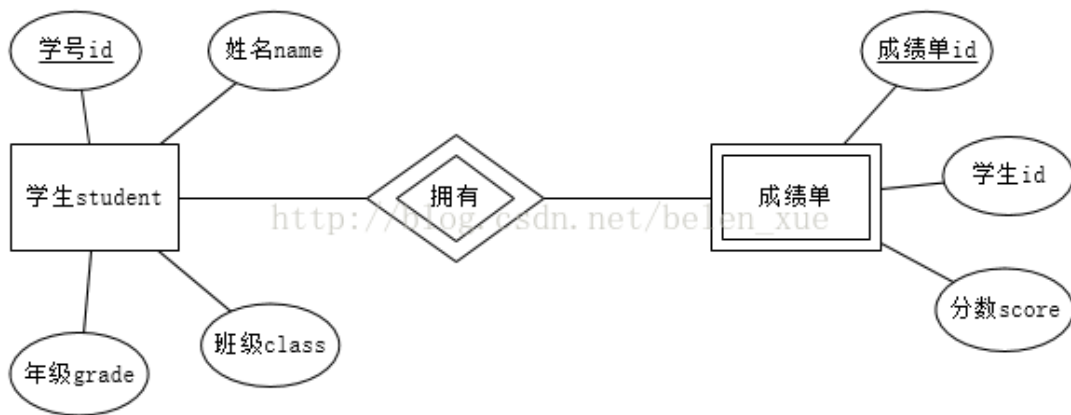
ER实体补充讲解：

ER的实体还会细分为弱实体和复合实体：

弱实体： 一个实体必须依赖于另一个实体存在，那么前者是弱实体，后者是强实体，弱实体必须依赖强实体存在，例如上图的学生实体和成绩单实体，成绩单依赖于学生实体而存在，因此**学生是强实体，而成绩单是弱实体**。

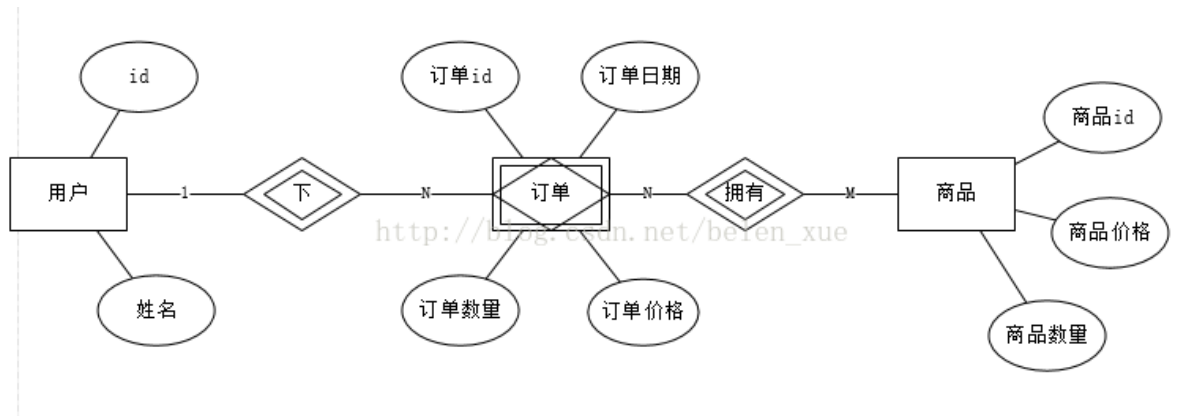
弱实体和强实体的联系必然只有1：N或者1：1，这是由于弱实体完全依赖于强实体，强实体不存在，那么弱实体就不存在，所以弱实体是完全参与联系的，因此弱实体与联系之间的联系也是用的双线菱形。

上面实例根据弱实体的情况更改如下图：



复合实体：复合实体也称联合实体或桥接实体，常常用于实现两个或多个实体间的M：N联系，它由每个关联实体的主码组成，用长方体内加一个菱形来表示。

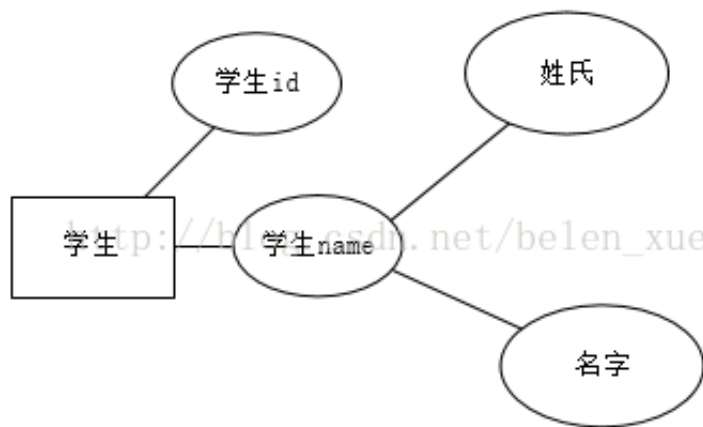
下图就是一个典型的复合实体，因为只是举例，相对粗糙，用户和商品两个实体是M：N的关系，中间又订单这个实体联系，因此订单这个实体是一个复合实体，同时如果用户实体不存在，就没有订单实体的存在，因此对于用户实体来讲订单是弱实体，同理商品实体如果不存在，同样不存在订单实体，因此对商品实体而言订单是弱实体，具体如图：



ER属性补充讲解：

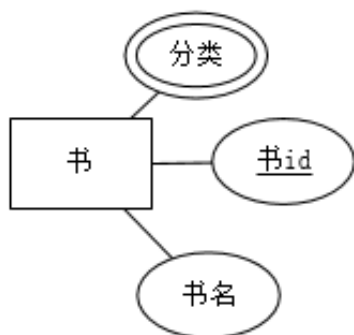
er图的属性还细分为复合属性、多值属性和派生属性、可选属性，同时还有用来表示联系的属性，称为联系属性。

复合属性(composite attribute)：复合属性是指具有多个属性的组合，例如名字属性，它可以包含姓氏属性和名字属性，如下图：



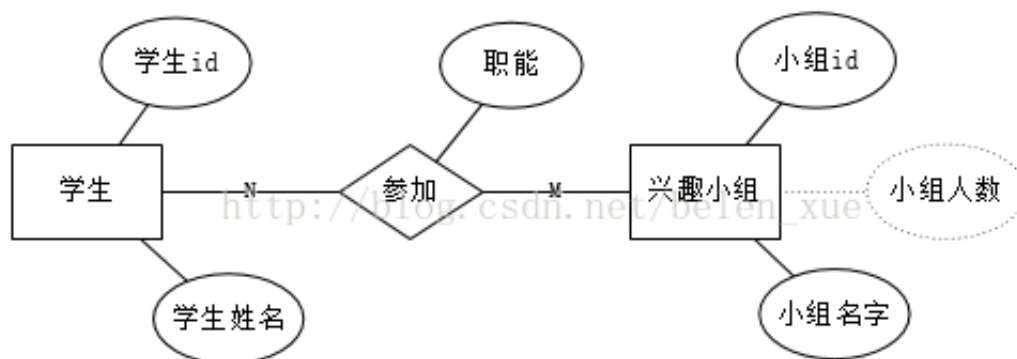
复合属性也有唯一属性，例如学生的所在班级属性，由于多个年级都有班级，所以单单班级属性是不唯一的，但是和年级组成的复合属性后则可以匹配成唯一属性。

多值属性 (multivalued attribute)： 一个实体的某个属性可以有多个不同的取值，例如一本书的分类属性，这本书有多个分类，例如科学、医学等，这个分类就是多值属性，用双线椭圆表示。【多分类属性】

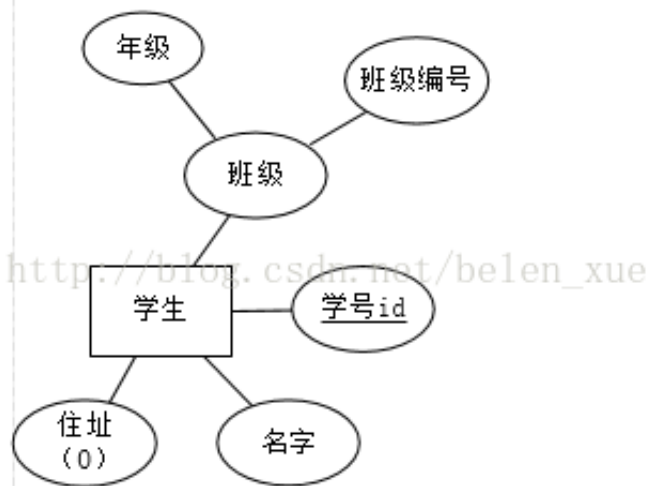


派生属性(derived attribute)： 是非永久性存于数据库的属性。派生属性的值可以从别的属性值或其他数据（如当前日期）派生出来，用虚线椭圆表示，如下图。

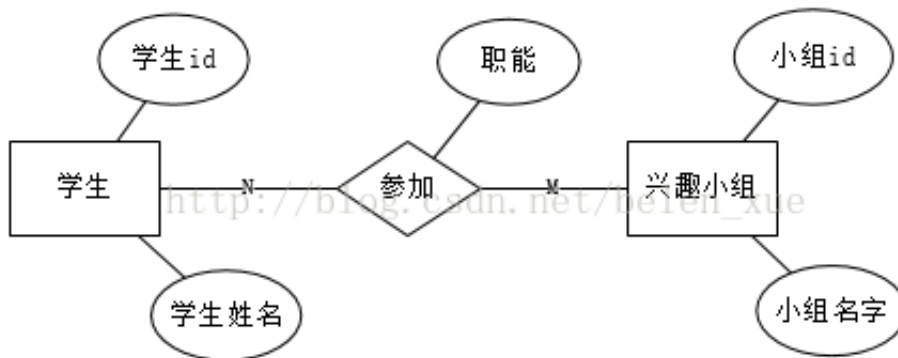
下面的小组人数就是典型的派生属性，随着学生实例的参加的兴趣小组变化，小组人数属性也会变化，一般来讲派生属性不存在于数据库中，而是通过相应的公式进行计算得到，如果要放到数据库中，那么隔一段时间就要进行更新，否则会出现数据错误。



可选属性(optional attribute)： 并不是所有的属性都必须有值，有些属性的可以没有值，这就是可选属性，在椭圆的文字后用 (O) 来表示，如下图的地址就是一个可选属性。



联系属性：联系属于用户表示多个实体之间联系所具有的属性，一般来讲M:N的两个实体的联系具有联系属性，在1:1和1:M的实体联系中联系属性并不必要。



er图的联系内容较多，以后单独补充！

1 简介

1.1 ER模型的作用

以建筑房屋的例子，说明建筑师、建筑工人和原材料商的作用，强调数据库规划师、数据库操作人员和数据库管理系统的作用，指出ER模型是种基于关系模型的数据库整体规划方法。

1.2 ER模型的概念

模型来源于数据概念模型，说明ER模型是一种概念模型设计方法。

ER来源于英文单词“实体”和“联系”，说明它主要是利用这两个图例的绘制进行表达和设计。由美籍华人陈平山设计。Codd的范式方法不甚适用于全局性设计，除此以外，还有其他一些概念模型设计方法。

1.3 ER模型的设计方法

从用户需求调研开始，结合学籍信息系统的案例说明。

- 1) 调研用户需求，得到几点基本要求，如“查找和修改学生的基本信息”，“查找和修改学生成绩信息”。
- 2) 识别用户需求中的数据内容，忽略功能性内容，并再次调研以确定数据的含义，如什么是学生基本信息。最终得到如下数据：学号，姓名，性别，生日，身高、课程名、成绩。判断这些数据是否合格，只需判断它们是否可以再次细分（准则1）。这些最后得到的数据被称为属性。
- 3) 将全部信息放入一个关系，虽然功能可行，但是冗余太大，易带来不一致的危险。
- 4) 使用“拆”的方法，将关系按照语义联系细分。

将“学号，姓名，性别，生日，身高”五个属性识别出“学生”语义，并新建关系，学号作为主键。

将“课程名”一个属性识别出“课程”语义，并新建关系，并添加课程号主键，同时体现出信息建设所具有的变革意味。

说明“成绩”属性不能建立成绩实体，因为它不能独立存在（准则2）。反过来说，它与现有关系存在联系。

说明“成绩”属性不能归入上述两个关系中任一关系，因为对于上述实体集中的每一实体，该项属性取值不惟一（准则3）。

强调“成绩”属性是学生和课程联系的属性，即将两个实体主键拿过来新建关系，并将“成绩”属性放入该关系。

最后得到关系为：

学生（学号，姓名，性别，生日，身高）

课程（课程号、课程名）

选修（学号、课程号，成绩）

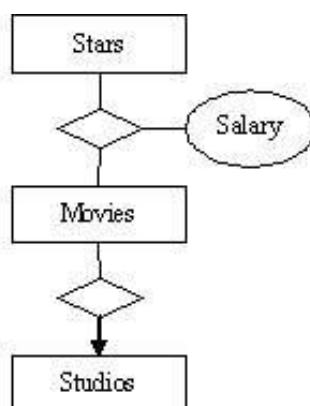
2 ER图的绘制

结合学籍信息系统案例简述建模方法，并描述相应的ER图。

说明实体、属性（包括主属性）和联系的画法。

强调ER模型设计的两大内容：一是将需求通过ER表达，另一个是将ER转换为关系模型。ER图是从实践归纳出来的理论方法，可以重新指导实践行为，既可以作为理解和表达用户数据需求的手段，同时也构成如何生成关系模型的依据。

结合电影公司案例说明ER图的绘制。



3 ER图转换为关系模型的方法

从学籍信息系统案例解释ER图和生成关系模型的对应关系，方法为：

- 1) 一个实体直接对应一个关系，实体的所有属性成为关系的属性，实体的主属性成为关系的主属性
- 2) 一个联系直接对应一个关系，此关系首先将联系所关联的所有实体主属性拿来一齐构成自己的主属性，并添加自己的属性。

强调实体和联系对应的关系名称命名问题，如不用汉字，联系关系要表达语义等。

按照上述方法将电影公司案例ER图转换为关系模型（属性自己添加）：

```
Movies (mid, title, year, length, type)
Stars (stid, name, address)
Studios (sdid, name, address)
MovieStar (mid, stid, salary)
MovieStudio (mid, sdid)
```

说明上述做法中MovieStudio关系可以合并到Movies中，指出原因在于此种联系为**1：N**。??

结合学生和班级的问题再次说明这种合并的用法，并说明1：N的联系如何生成关系模型，方法为：在联系值为多的实体后添加联系值为1的实体的主属性。

说明联系的三种类型，并强调涵盖关系。说明多对多联系之所以新建关系的原因在于将其拆成两个一对多，而一对多联系作为多对多联系的特例，可以按照多对多联系处理，不过合并更为简单。

ER图转换为关系模型的课堂练习之一（属性自己添加）：

结果为：

```
学生（学号，姓名，性别）
课程（课程号，课程名称，教师号）
教师（教师号，教师名）
参考书（书号，书名，课程号）
学生课程（学号，课程号，成绩）
```

说明一对一联系转换方法为在任一关联实体后添加另一实体的主属性。强调必要时要结合查询要求等功能决定选择。

ER图转换为关系模型的课堂练习之二（属性自己添加）：

结果为：

学生（学号，姓名，性别）
课程（课程号，课程名称）
教师（教师号，教师名）
参考书（书号，书名）
课程教师参考书（课程号，教师号，书号）
学生课程（学号，课程号，成绩）
学生管理（学号，班委号）

（注：如学生管理为1：N，则应在学生关系后添加“班委号”即可）

4 ERWin使用简介

介绍ERWin等常用数据库建模工具（还有Visio，PowerDesigner，Rational等）的好处，绘图简单、转换方便。

4. 1 安装注册

4. 2 ER的绘制，注意联系实体（Associate Entity）的生成

4. 3 正向工程的使用，即从逻辑模型转换为物理模型

4. 4 逆向工程的使用，即从物理模型转换为逻辑模型

5 ER图的规划方法

将所得到的属性分门归类，利用以前讲述的几条简单准则即可将其并入实体或者联系。

按照的准则主要有两个：

- 1) 属性如果不能具有独立存在的意义，则不能成为一个新建立实体。
- 2) 属性归入一个实体的依据是对实体的每个实例而言具有唯一值。

结合合同案例讲解设计方法，如：

相应的ER图结构可以描述为：

合同（合同号，订货日期，厂商名称，厂商地址）
货物（货物编号，货物分类）
合同货物（合同号，货物编号，订货数量）

除此以外，还有一种方法，将厂商视为实体，相应的ER图结构可以描述为：

合同（合同号，订货日期，厂商号）
货物（货物编号，货物分类）
厂商（厂商号，厂商名称，厂商地址）
合同货物（合同号，货物编号，订货数量）

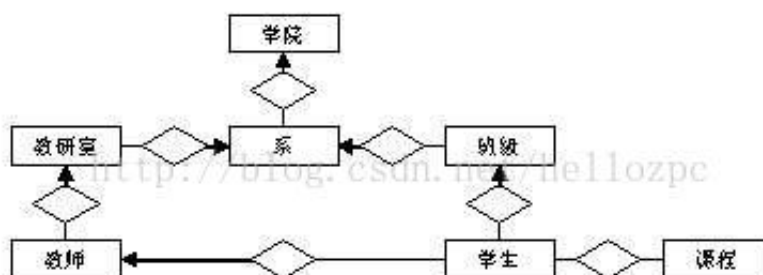
两种方法的比较：

1) 第一种方法存在冗余（相同的厂商具有相同的地址），但是实体少，处理简单，以后利用范式分析方法可以得到标准结构。所以，实体能少则少。

2) 第二种方法需要考察厂商实体与合同、货物的关系，发现与合同存在一对多的关系，但与货物存在多对多的关系，明显增加处理的复杂度。事实上，可以考虑去除厂商与货物的关系（可以从剩下的关系中推断出来，但是其他的关系不能省略）。

结合学校信息系统案例讲解设计方法，如：

学院有很多不同的系，每个系下属若干个教研室，每个教研室下属若干个教师，其中的有些教授能够带若干个研究生，但是每个研究生只能有一个导师；每个系下属若干个班级，每个班有很多学生，每个学生选修很多门课程，每门课也可以被多个学生选修。

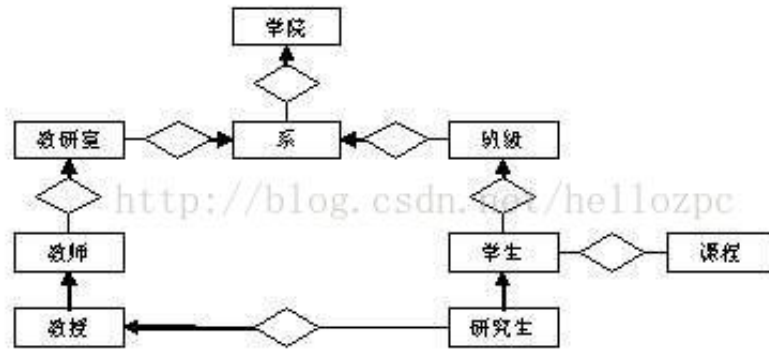


相应的ER图结构可以描述为：

学院（学院号，学院名）
系（系号，系名，学院号）
教研室（教研室号，教研室名，系号）
教师（教师号，教师名，职称，教研室号）
班级（班级号，班级名，系号）
学生（学号，名称，班级号，学历，导师号）
课程（课程号，课程名）
学生课程（学号，课程号）

除此以外，还有一种方法，将教授和研究生单独视为实体，相应的ER图结构可以描述为：

学院（学院号，学院名）
系（系号，系名，学院号）
教研室（教研室号，教研室名，系号）
教师（教师号，教师名，教研室号）
教授（教授号）
班级（班级号，班级名，系号）
学生（学号，名称，班级号，学历）
研究生（研究生号，导师号）
课程（课程号，课程名）
学生课程（学号，课程号）



两种方法的比较：

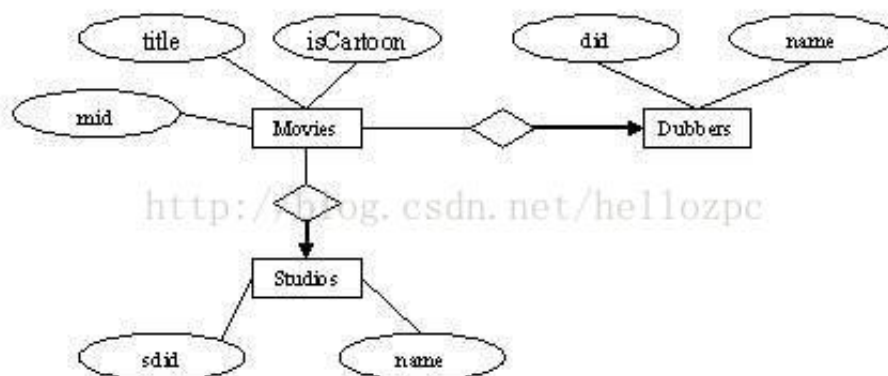
- 1) 第一种方法实体较少，但是教师实体的职称、学生实体的学历和导师号存在较多的空值可能，在空值较少的情况下是个较好的选择。
- 2) 第二种方法虽然实体较多，没有上述的空值问题。对于此类子类问题，要注意相应的处理方法，即子类中只留有必须的主键信息，全部信息都可以直接从父类中获取，这样查询更为简单易行。

6 子类问题

在电影公司数据库中表达电影的实体信息中，表达卡通片的主角配音演员信息

这是个子类问题，因为卡通片电影是电影的一种子类。

处理方式一：将子类视为父类的内容来处理



上述做法对应的关系模型为：

```
Movies(mid,title,isCartoon,sdid,did)
Studios(sdid,name)
Dubbers(did,name)
```

说明：

- 1) 可以表达查询要求
 - 2) 对于非卡通片，did多为空
 - 3) 不适用于子类较多的情况，如引入爱情片（affectional film）的结局（finale）查询要求
- 对应的关系模型为：

```
Movies(mid,title,isCartoon,sdid,did, isAffectional, finale)
Studios(sdid,name)
Dubbers(did,name)
```

可以看出大部分电影的did和did信息皆为空

处理方式二：将子类单独表达

相应的关系模型为：

```
Movies(mid,title,sdid)
Studios(sdid,name)
Dubbers(did,name)
AffectionalFilms(mid,title,sdid,finale)
Cartoons(mid,title,sdid,did)
```

说明：

1) 没有较多的空值问题

2) 爱情片、卡通片和一般电影分开存储不便于那些不区分电影类型的常见检索（需要更多的表合并操作）

处理方式三：将子类单独表达，但是父类拥有全部记录信息，子类只有主键信息和自己的特殊信息。这种方式一般是最为有效的。

改进的关系模型为：

```
Movies(mid,title,sdid)
Studios(sdid,name)
Dubbers(did,name)
AffectionalFilms(mid,finale)
Cartoons(mid,did)
```

总结：

ER模型的设计一般步骤：

①客户调研，了解需求，收集数据

②根据用户需求来把数据细分到不可再分的基本数据项

③可以先将全部信息(或者称属性、字段)放入一个关系，虽然功能可行，但是冗余太大，易带来不一致的危险。使用“拆”的方法，将关系按照语义联系(实体-关系)细分。根据从客户那得到的原始信息识别出实体，依次将属性放入实体中(原则是只有该属性对于该实体来说是唯一确定的才能将这个属性归到这个实体里，否则看看能不能放到其他实体里)。注意有的属性是联系的属性，这时就要建立实体间的联系，并给联系加个属性。

从E-R模型到关系模型的变换规则：

①每个实体建一张表，实体的属性变为表的一个字段，要设个主属性。

②所有的多对多的联系也要转变为一张表，如果是多对一的联系就不要新建表了，在联系值为多的实体后面添加联系值为1的实体的主键，如果是一对一的联系那就随便在哪个实体后面加上另外一个实体的主键。

参考文献：

1.<http://www.njcie.com/bbs/dispbbs.asp?boardid=19&Id=412&page=8>

2.<http://www.njcie.com/bbs/dispbbs.asp?boardid=19&Id=1163&page=2>

3.<http://www.njcie.com/bbs/dispbbs.asp?boardid=19&Id=1166&page=2>

4.<http://www.njcie.com/bbs/dispbbs.asp?boardid=19&Id=1173>

视频下载地址（迅雷输入地址即可）

<http://www.njcie.com/DB/files/41.wmv>

<http://www.njcie.com/DB/files/42.wmv>

<http://www.njcie.com/DB/files/43.wmv>