

# Ch02 高级（概念）数据模型

---

## 1. 数据库的设计过程

- 需求分析 ( requirements analysis )
- 概念数据库设计 ( conceptual DB design )
- 逻辑数据库设计 ( logical DB design )
- 模式优化 ( schema refinement )
- 物理数据库设计 ( physical DB design )
- 安全设计 ( security design )

## 2. 在数据库的设计中，E-R图产生于 逻辑设计 概念设计阶段

## 3. 数据库的概念模型独立于 具体的机器与DBMS

## 4. 将概念模型转换为逻辑数据模型，属于数据库的 逻辑设计

## 5. 数据库三级模式体系结构的划分，主要有利于保持数据库的 数据独立性

## 6. 实体与实体型的关系

实体型是同类实体的集合。实体型就像C++中的类( class )，实体就是类的实例( instance )。

## 7. 实体的属性 ( attribute )

- 按结构分：简单属性和复合属性（可被划分为更小的子属性）
- 按取值分：单值属性、多值属性、导出属性和空值属性
  - 多值属性：对于某个属性，可以取多个值，如学位为多值属性，可以取值为学士、硕士、博士
  - 导出属性：根据其他属性可以推断或简单计算得到的
  - 空值属性：值不确定或无值的属性，如书籍的附件属性

## 8. 实体的键 ( key )

键——具有唯一标识特性的一个或一组属性。

由一个属性构成的键是简单键，由多个属性构成的键是复合键

- 候选键 ( candidate key ): 最小属性集合的键，可以有一个属性或多个属性
- 主键 ( primary key ): 存在多个候选键时，选定一个候选键作为主键

## 9. 键约束

简单说来，就是给定一个实体，若通过联系可以唯一确定另一方实体的实例，则两实体间存在键约束，要注意键约束的方向。例如：实体A与B存在联系X，若给定A，通过X可唯一确定B，则E-R图中，A指向X（有箭头）。

只有一对一联系和一对多联系才存在键约束。

## 10. 弱实体约束

弱实体需要依附于其他实体才能存在，该弱实体与其他实体的这种“依附关系”称为识别联系，“其他实体”称为“父实体”。父实体与其他实体的联系只可能是1: 1或1: n.

## 11. E-R图

- E-R图的各种元素图示如下：
  - 实体——矩形
  - 联系——菱形
  - 属性——椭圆
  - 导出属性——虚线椭圆
  - 多值属性——双椭圆
  - 复合属性——分支椭圆
  - 键属性——主键下加下划线
  - 键约束——箭头，由实体指向联系
  - 完全参与约束——粗实线
  - 弱实体——粗实线矩形
  - 识别联系——粗实线菱形

## 12. 扩展实体联系模型

用三角形表示，三角形内书写“ISA”。如学生是一个“超类”，有本科生和研究生两个“子类”。

**13. 联系为动词，实体为名词。**对于同一名称的联系，也许不一定是三元联系，有时需要单独写出实体A与C和B与C的联系，如演员和导演属于电影公司，此处的“属于”应是二元联系，因为导演和演员不是“捆绑式”地属于电影公司，而售货员、顾客和商品应当“捆绑式”地设计为三元联系。