

6. 高阶数据库模型（数据库概念设计）

实体-联系模型 (entity-relationship model, ER model): 可通过画图得到 ER 图。

- **实体 (entity):** 类似于面向对象中的对象；
- **实体集 (entity set, ES):** 许多相似实体的集合，图中用矩形表示；
- **属性 (attribute):** 实体集中实体的性质，图中用椭圆表示；
- **联系 (relationship):** 联系两个或多个实体集的，用于描述若干实体集的实体之间存在联系，图中用菱形表示。
- 图中的边连接 ES 及其属性、关系以及其连接的 ES；ER 图描述数据库的模式。
- **二元联系 (binary relationship):** 连接两个 ES 的联系，分为如下若干中，在 ER 图中，“多”用不带箭头的连线表示；“一”代表着最多一个，用箭头表示；恰好一个用圆箭头表示。
 - 多对多：many-to-many；
 - 一对多/多对一：one-to-many, many-to-one；
 - 一对一：one-to-one。
- **多元联系:** 连接多个 ES 的联系（共同关联）；如老师、课程以及开课院系形成了一个三元联系。
 - 三元联系与三个二元联系不等价！
 - 如何转化为二元联系？人为增加一个虚构的抽象实体集 (connecting ES)，将这个实体集与原有的每个实体集中分别创建联系。
- 一个 ES 可能参与不止一个关系，给边带上标签以区分不同的意义。
- **子类 (subclass):** 实体集的子集（特例），其可能满足一些特殊性质。
- **isa 关系 (isa relationship):** 联系两个实体集，表示一个是另一个的子类，用三角形表示，是一个一对一的关系，只允许树形结构，同时不允许多重继承（如助教既是老师，又是学生）。
 - isa 关系组成了一棵树；
 - 子类具有父类具有的所有属性，同时具有祖先类具有的所有属性（但是不在子类中显示，在祖先类中显示）；
 - 子类参与父类参与的所有关系，同时参与祖先类参与的所有关系（但是不在子类中参与，在祖先类中参与）；
 - 与面向对象不同：面向对象的子类中具有祖先类的所有信息，不需要到祖先类中进行查询。
- **ER 模型要求实体集必须有键值 (key),** 即没有两个元素有着相同的键值，当然有时候一个实体集的键值可能在另一个实体集中。如果有超过一个键值，则用下划线表示主键值。子类需要继承父类的键值。
- **引用完整性 (Referential Integrity):** 利用圆箭头来限制多对一/一对一关系（必须存在）。
- **度限制 (Degree Constraint):** 限制使用这个连接的实体数量，用标签标注在边上。如每个院系不能有超过 200 个学生，可以利用联系边上的标签“ ≤ 200 ”来标注。
- **弱实体集 (Weak ES, WES):** 某个实体集的所有属性都不足以构成其自身的键值，需要当前实体集连接的其他 ES 的某些属性来构成自身的键值，那么当前实体集称为弱实体集。
 - 来源：unit-of 或 belong-to 结构（如班级与学校，摄制组与电影制片厂等），或是由于消除多元联系引入的虚拟的抽象实体集。

- **键值**：来自本身实体集的零个或多个属性，以及来自于**支撑实体集** (supporting ES) 的若干属性。
- **要求**：
 - 弱实体集必须与其支撑实体集中有支撑联系 (supporting relationship) R ；且 R 是多对（恰好）一 (many-to-exactly-one) 的二元关系；
 - 支撑实体集将其键值提供给当前弱实体集；当然支撑实体集可能本身也是 WES，那么其键值可能由其他实体集提供；
 - 一个弱实体集可能由多个支撑实体集支撑，他们中的任何之一都可以支持当前弱实体集；他们任何之一的键值都可能出现在弱实体集的键值中。
- 在 ER 图中，弱实体集用双框矩形表示；支撑关系用双框菱形表示；键值用下划线表示。