



数据库系统概论

An Introduction to Database System

第七章 数据库设计 (续 2)

中国人民大学信息学院

第七章 数据库设计



- 7.1 数据库设计概述
- 7.2 需求分析
- 7.3 概念结构设计
- 7.4 逻辑结构设计
- 7.5 数据库的物理设计
- 7.6 数据库的实施和维护
- 7.7 小结

7.4 逻辑结构设计



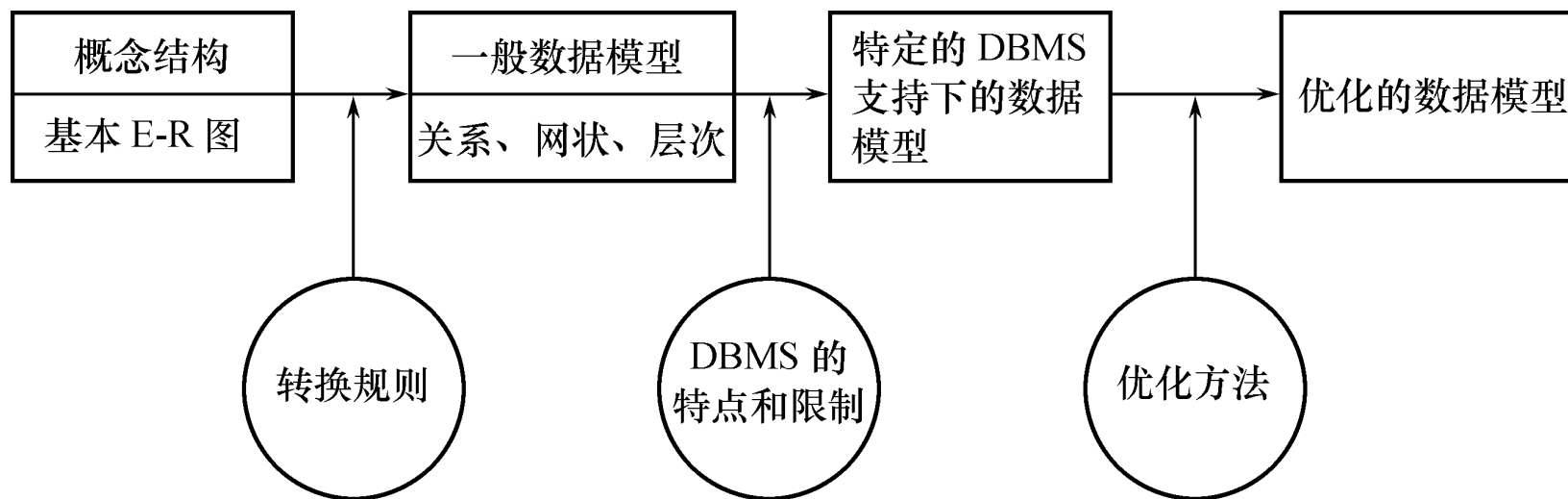
❖ 逻辑结构设计任务

- 把概念结构设计阶段设计好的基本 E-R 图转换为与选用 DBMS 产品所支持的数据模型相符合的逻辑结构

❖ 逻辑结构设计的步骤

- 将概念结构转化为一般的关系、网状、层次模型
- 将转换来的关系、网状、层次模型向特定 DBMS 支持下的数据模型转换
- 对数据模型进行优化

逻辑结构设计（续）



逻辑结构设计时的 3 个步骤

7.4 逻辑结构设计

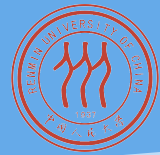


7.4.1 E-R 图向关系模型的转换

7.4.2 数据模型的优化

7.4.3 设计用户子模式

7.4.1 E-R 图向关系模型的转换



❖ 转换内容

❖ 转换原则

E-R 图向关系模型的转换（续）



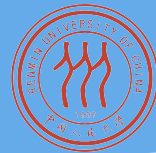
❖ E-R 图向关系模型的转换要解决的问题

- 如何将实体型和实体间的联系转换为关系模式
- 如何确定这些关系模式的属性和码

❖ 转换内容

- 将 E-R 图转换为关系模型：将实体、实体的属性和实体之间的联系转换为关系模式。

E-R 图向关系模型的转换（续）



实体型间的联系有以下不同情况：

(1) 一个 1:1 联系可以转换为一个独立的关系模式，也可以与任意一端对应的关系模式合并。

- 转换为一个独立的关系模式
- 与某一端实体对应的关系模式合并

(2) 一个 1:n 联系可以转换为一个独立的关系模式，也可以与 n 端对应的关系模式合并。

- 转换为一个独立的关系模式
- 与 n 端对应的关系模式合并

E-R 图向关系模型的转换（续）



(3) 一个 $m:n$ 联系转换为一个关系模式。

例，“选修”联系是一个 $m:n$ 联系，可以将它转换为如下关系模式，其中学号与课程号为关系的组合码：

选修（学号，课程号，成绩）

E-R 图向关系模型的转换（续）



(4) 三个或三个以上实体间的一个多元联系转换为一个关系模式。

例，“讲授”联系是一个三元联系，可以将它转换为如下关系模式，其中课程号、职工号和书号为关系的组合码：

讲授（课程号，职工号，书号）

E-R 图向关系模型的转换（续）



(5) 具有相同码的关系模式可合并

- 目的：减少系统中的关系个数
- 合并方法：将其中一个关系模式的全部属性加入到另一个关系模式中，然后去掉其中的同义属性（可能同名也可能不同名），并适当调整属性的次序

E-R 图向关系模型的转换（续）



注意：

- ❖ 从理论上讲，1:1 联系可以与任意一端对应的关系模式合并
- ❖ 但在一些情况下，与不同的关系模式合并效率会大不一样。因此究竟应该与哪端的关系模式合并需要依应用的具体情况而定。
- ❖ 由于连接操作是最费时的操作，所以一般应以尽量减少连接操作为目标。
例如，如果经常要查询某个班级的班主任姓名，则将管理联系与教师关系合并更好些。

E-R 图向关系模型的转换（续）



[例] 把图 7.30 中虚线上部的 E-R 图转换为关系模型

- 部门实体对应的关系模式

部门（部门号，部门名，经理的职工号，…）

- 此关系模式已包含了联系“领导”所对应的关系模式

- 经理的职工号是关系的候选码

- 职工实体对应的关系模式

职工（职工号、部门号，职工名，职务，…）

- 该关系模式已包含了联系“属于”所对应的关系模式

E-R 图向关系模型的转换（续）



[例] 把图 7.30 中虚线上部的 E-R 图转换为关系模型（续）

- 产品实体对应的关系模式

产品（产品号，产品名，产品组长的职工号，…）

- 供应商实体对应的关系模式

供应商（供应商号，姓名，…）

- 零件实体对应的关系模式

零件（零件号，零件名，…）

E-R 图向关系模型的转换（续）



[例] 把图 7.30 中虚线上部的 E-R 图转换为关系模型（续）

- 联系“参加”所对应的关系模式

职工工作 (职工号, 产品号, 工作天数, ...)

- 联系“供应”所对应的关系模式

供应 (产品号, 供应商号, 零件号, 供应量)

7.4 逻辑结构设计

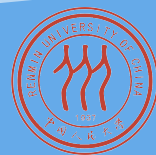


7.4.1 E-R 图向关系模型的转换

7.4.2 数据模型的优化

7.4.3 设计用户子模式

7.4.2 数据模型的优化



- ❖ 得到初步数据模型后，还应该适当地修改、调整数据模型的结构，以进一步提高数据库应用系统的性能，这就是数据模型的优化
- ❖ 关系数据模型的优化通常以规范化理论为指导

数据模型的优化（续）



❖ 优化数据模型的方法

1. 确定数据依赖

按需求分析阶段所得到的语义，分别写出每个关系模式内部各属性之间的数据依赖以及不同关系模式属性之间数据依赖

2. 消除 冗余的联系

对于各个关系模式之间的数据依赖进行极小化处理，消除 冗余的联系。

3. 确定所属范式

- 按照数据依赖的理论对关系模式逐一进行分析
- 考查是否存在部分函数依赖、传递函数依赖、多值依赖等
- 确定各关系模式分别属于第几范式

数据模型的优化（续）



4. 按照需求分析阶段得到的各种应用对数据处理的要求，分析对于这样的应用环境这些模式是否合适，确定是否要对它们进行合并或分解。

注意：并不是规范化程度越高的关系就越优，一般说来，第三范式就足够了

数据模型的优化（续）



例：在关系模式

学生成绩单（学号，英语，数学，语文，平均成绩）

中存在下列函数依赖：

学号 \rightarrow 英语

学号 \rightarrow 数学

学号 \rightarrow 语文

学号 \rightarrow 平均成绩

（英语，数学，语文） \rightarrow 平均成绩

数据模型的优化（续）



显然有：

学号 \rightarrow (英语 , 数学 , 语文)

因此该关系模式中存在传递函数信赖，是 2NF 关系

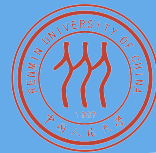
虽然平均成绩可以由其他属性推算出来，但如果应用中需要经常查询学生的平均成绩，为提高效率，仍然可保留该冗余数据，对关系模式不再做进一步分解

数据模型的优化（续）



5. 按照需求分析阶段得到的各种应用对数据处理的要求，对关系模式进行必要的分解，以提高数据操作的效率和存储空间的利用率
 - 常用分解方法
 - 水平分解
 - 垂直分解

数据模型的优化（续）



■ 水平分解

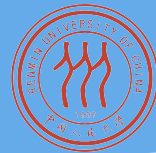
➤ 什么是水平分解

- 把（基本）关系的元组分为若干子集合，定义每个子集合为一个子关系，以提高系统的效率

➤ 水平分解的适用范围

- 满足“80/20 原则”的应用
- 并发事务经常存取不相交的数据

数据模型的优化（续）



- 垂直分解

- 什么是垂直分解

- 把关系模式 R 的属性分解为若干子集合，形成若干子关系模式

- 垂直分解的适用范围

- 取决于分解后 R 上的所有事务的总效率是否得到了提高

7.4 逻辑结构设计



7.4.1 E-R 图向关系模型的转换

7.4.2 数据模型的优化

7.4.3 设计用户子模式

7.4.3 设计用户子模式



❖ 定义用户外模式时应该注重的问題

包括三个方面：

(1) 使用更符合用户习惯的别名

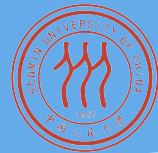
(2) 针对不同级别的用户定义不同的 View ，

以

满足系统对安全性的要求。

(3) 简化用户对系统的使用

设计用户子模式（续）



【例】 关系模式产品（产品号，产品名，规格，单价，生产车间，生产负责人，产品成本，产品合格率，质量等级），可以在产品关系上建立两个视图：

为一般顾客建立视图：

产品 1 （产品号，产品名，规格，单价）

为产品销售部门建立视图：

产品 2 （产品号，产品名，规格，单价，车间，生产负责人）

- 顾客视图中只包含允许顾客查询的属性
- 销售部门视图中只包含允许销售部门查询的属性
- 生产领导部门则可以查询全部产品数据
- 可以防止用户非法访问不允许他们查询的数据，保证系统的安全性

逻辑结构设计小结



❖ 任务

- 将概念结构转化为具体的数据模型

❖ 逻辑结构设计的步骤

- 将概念结构转化为一般的关系、网状、层次模型
- 将转化来的关系、网状、层次模型向特定 DBMS 支持下的数据模型转换
- 对数据模型进行优化
- 设计用户子模式

逻辑结构设计小结



- ❖ E-R 图向关系模型的转换内容
- ❖ E-R 图向关系模型的转换原则

逻辑结构设计小结



❖ 优化数据模型的方法

1. 确定数据依赖
2. 对于各个关系模式之间的数据依赖进行极小化处理，消除冗余的联系。
3. 确定各关系模式分别属于第几范式。
4. 分析对于应用环境这些模式是否合适，确定是否要对它们进行合并或分解。
5. 对关系模式进行必要的分解或合并

逻辑结构设计小结



❖ 设计用户子模式

1. 使用更符合用户习惯的别名
2. 针对不同级别的用户定义不同的外模式，以满足系统对安全性的要求。
3. 简化用户对系统的使用