第七章 数据库设计

- 7.1 数据库设计概述
- 7.2 数据库设计过程
- 7.3 小结

7.2 数据库设计过程

7.2.1 需求分析

7.2.2 概念结构设计 ==> 建立概念模型 E-R图

7.2.3 逻辑结构设计 ==> 建立逻辑模型 关系模式

7.2.4 物理结构设计 ==> 建立物理模型 CREATE TABLE等

7.2.1 需求分析

- •需求分析就是分析用户的要求
 - 数据库设计的起点
 - 结果是否准确地反映了用户的实际要求,将直接影响到后面各个阶段的设计,并影响到设计结果是否合理和实用

- •需求分析的任务
 - •详细调查现实世界要处理的对象(组织、部门、企业等)
 - 充分了解原系统(手工系统或计算机系统)工作概况
 - •明确用户的各种需求
 - 在此基础上确定新系统的功能
 - 新系统必须充分考虑今后可能的扩充和改变

- •调查的重点是"数据"和"处理",获得用户对数据库的要求
 - (1) 信息要求
 - 用户需要从数据库中获得信息的内容与性质
 - 由信息要求可以导出数据要求, 即在数据库中需要存储哪些数据
 - (2) 处理要求
 - 用户要完成的处理功能
 - 对处理性能的要求
 - (3) 安全性与完整性要求

- 确定用户最终需求的难点
 - 用户缺少计算机知识,不能准确地表达自己的需求,他们所提出的需求往往不断地变化。
 - 设计人员缺少用户的专业知识,不易理解用户的真正需求, 甚至误解用户的需求
- •解决方法
 - 设计人员必须不断深入地与用户进行交流,才能逐步确定用户的实际需求

- •需求分析的方法
 - 调查清楚用户的实际需求并进行初步分析
 - 与用户达成共识
 - 分析与表达这些需求

• 分析方法

- 结构化分析方法(Structured Analysis,简称SA方法)
 - SA方法从最上层的系统组织机构入手
 - 采用自顶向下、逐层分解的方式分析系统
- •对用户需求进行分析与表达后,需求分析报告必须提交给用户,征得用户的认可

需求分析小结

- 把需求收集和分析作为数据库设计的第一阶段是十分重要的。
- •强调两点
 - (1) 设计人员应充分考虑到可能的扩充和改变,使设计易于更改,系统易于扩充
 - (2) 必须强调用户的参与

7.2.2 概念结构设计

- 将需求分析得到的用户需求抽象为信息结构(即概念模型) 的过程就是概念结构设计
- 概念模型的特点
 - (1) 真实、充分地反映现实世界,是现实世界的一个真实模型
 - (2) 易于理解,从而可以用它和不熟悉计算机的用户交换意见
 - (3) 易于更改, 当应用环境和应用要求改变时, 容易对概念模型修改和扩充
- 描述概念模型的工具
 - E-R模型

E-R模型

- 1. 实体之间的联系
 - (1) 两个实体型之间的联系:
 - 1)一对一联系(1:1)
 - ②一对多联系 (1:n)
 - ③多对多联系 (*m*: *n*)

①一对一联系 (1:1)

- ➤如果对于实体集A中的每一个实体,实体集B中至多有一个 (也可以没有)实体与之联系,反之亦然,则称实体集A与 实体集B具有一对一联系,记为1:1。
- ▶例如, 学校里一个班级只有一个正班长, 而一个班长只在一个班中任职, 则班级与班长之间具有一对一联系。

②一对多联系 (1: n)

- ▶如果对于实体集A中的每一个实体,实体集B中有n个实体 (n>0)与之联系,反之,对于实体集B中的每一个实体,实体集A中至多只有一个实体与之联系,则称实体集A与实体集B有一对多联系,记为1:n。
- ▶例如,一个班级中有若干名学生,而每个学生只在一个班级中学习,则班级与学生之间具有一对多联系。

③多对多联系 (m:n)

- 》如果对于实体集A中的每一个实体,实体集B中有n个实体($n \ge 0$)与之联系,反之,对于实体集B中的每一个实体,实体集A中也有m个实体($m \ge 0$)与之联系,则称实体集A与实体集B具有多对多联系,记为m:n。
- ▶例如,一门课程同时有若干个学生选修,而一个学生可以同时选修多门课程,则课程与学生之间具有多对多联系。

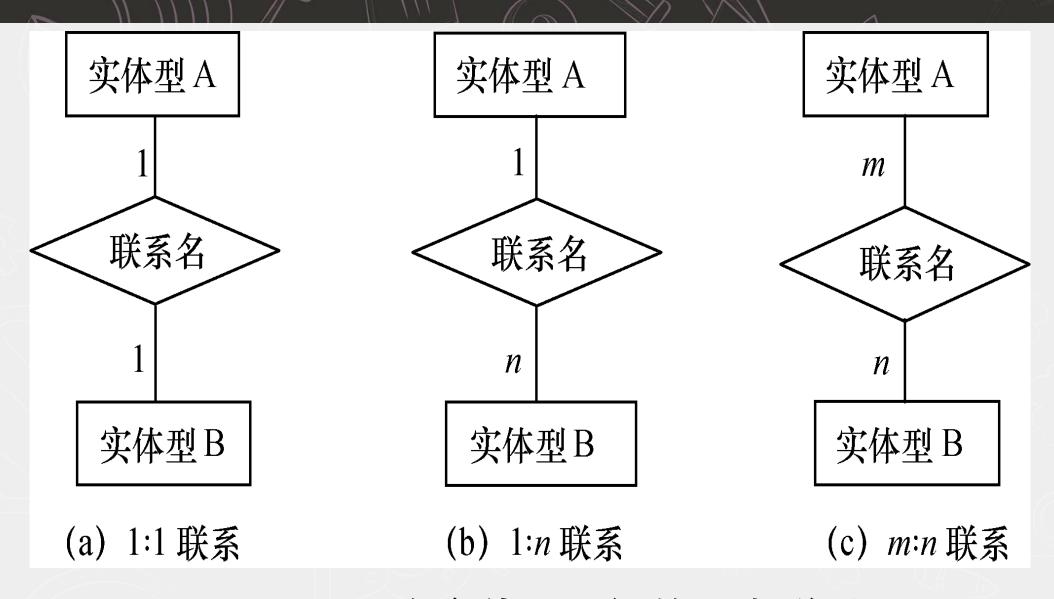


图7.6 两个实体型之间的三类联系

(2) 两个以上的实体型之间的联系

- ●一般地,两个以上的实体型之间也存在着一对一、一对多、多对多联系。
- ●对于课程、教师与参考书3个实体型,如果一门课程可以有若干个教师讲授,使用若干本参考书,而每一个教师只讲授一门课程,每一本参考书只供一门课程使用,则课程与教师、参考书之间的联系是一对多的,如图7.7(a)所示。

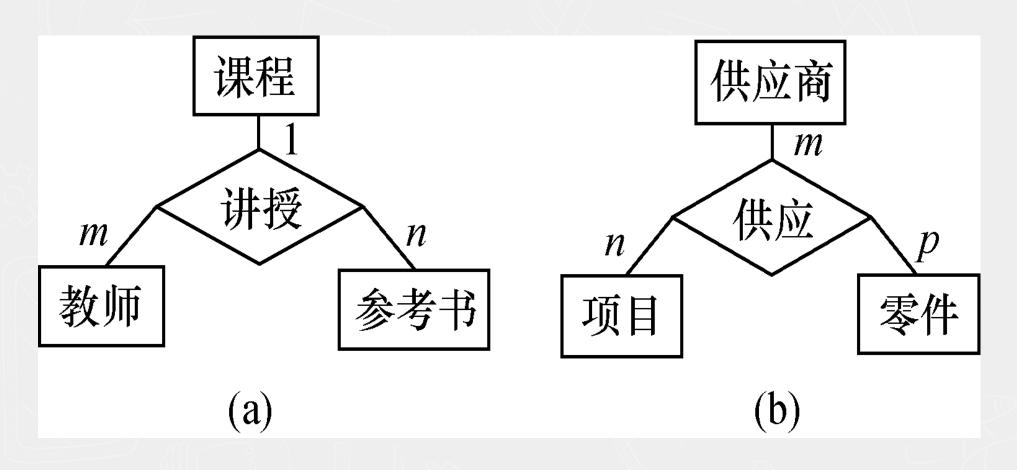


图7.7 三个实体型之间的联系示例

(3) 单个实体型内的联系

- ●同一个实体集内的各实体之间也可以存在一对一、一对多、多 对多的联系。
- ●例如,职工实体型内部具有领导与被领导的联系,即某一职工 (干部)"领导"若干名职工,而一个职工仅被另外一个职工直接 领导,因此这是一对多的联系,如图7.8所示。

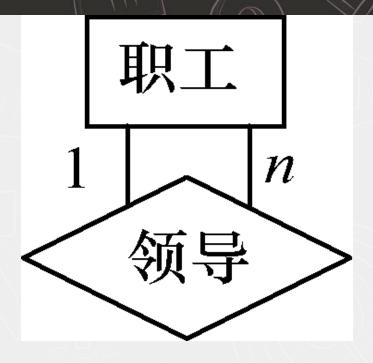


图7.8 单个实体型内的一对多联系示例

- ■联系的度:参与联系的实体型的数目
 - ●2个实体型之间的联系度为2,也称为二元联系;
 - ●3个实体型之间的联系度为3,称为三元联系;
 - ●N个实体型之间的联系度为N,也称为N元联系

2. E-R图

• E-R图提供了表示实体型、属性和联系的方法:

实体型:用矩形表示,矩形框内写明实体名。

属性:用椭圆形表示,并用无向边将其与相应的实体型连接起来。

▶例如, 学生实体具有学号、姓名、性别、出生年份、系、入学时间等属性, 用E-R图表示如图7.9所示

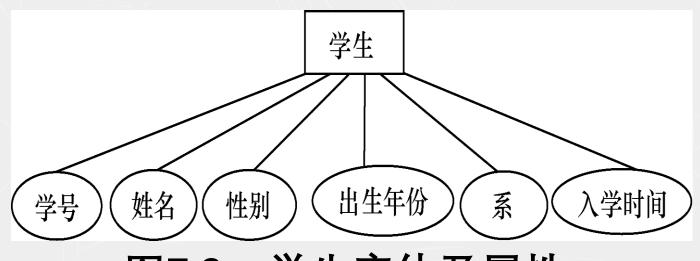


图7.9 学生实体及属性

- ●联系:用菱形表示,菱形框内写明联系名,并用无向边分别与有关实体型连接起来,同时在无向边旁标上联系的类型(1:1, n或 m:n)。
- ●联系可以具有属性

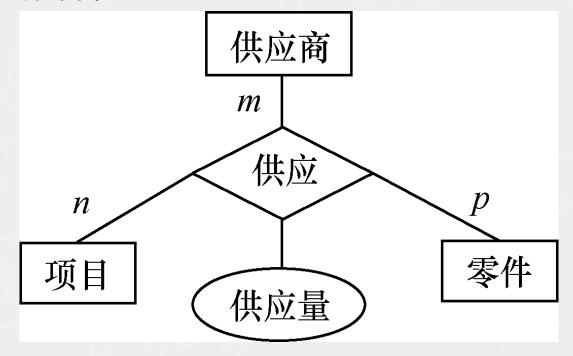
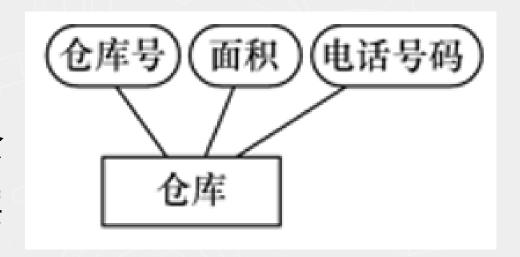
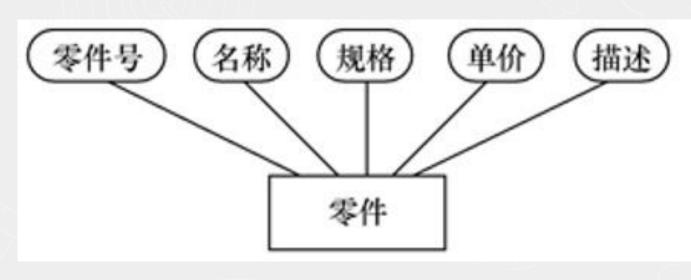


图7.10 联系的属性

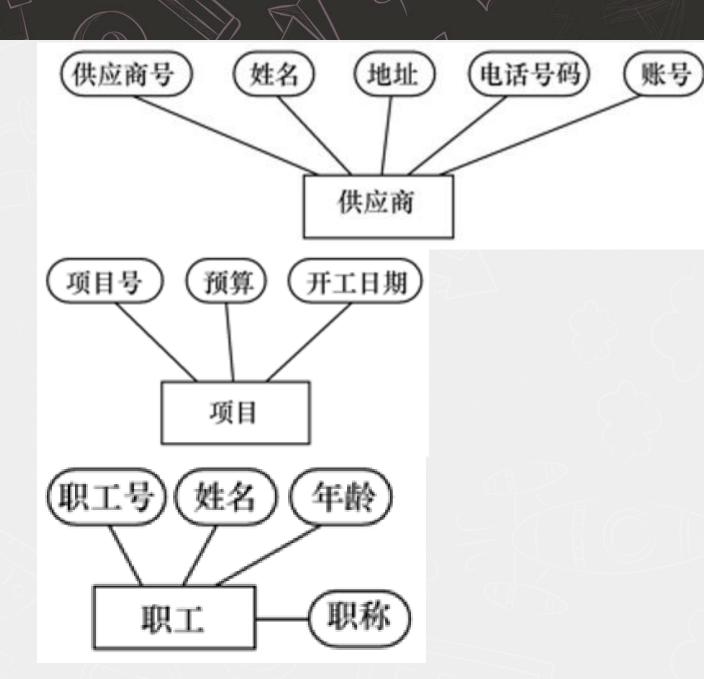
• 3. 一个实例

- 某个工厂物资管理的概念 模型。物资管理涉及的实 体有:
 - ●仓库:属性有仓库号、面 积、电话号码
 - ●零件:属性有零件号、名 称、规格、单价、描述

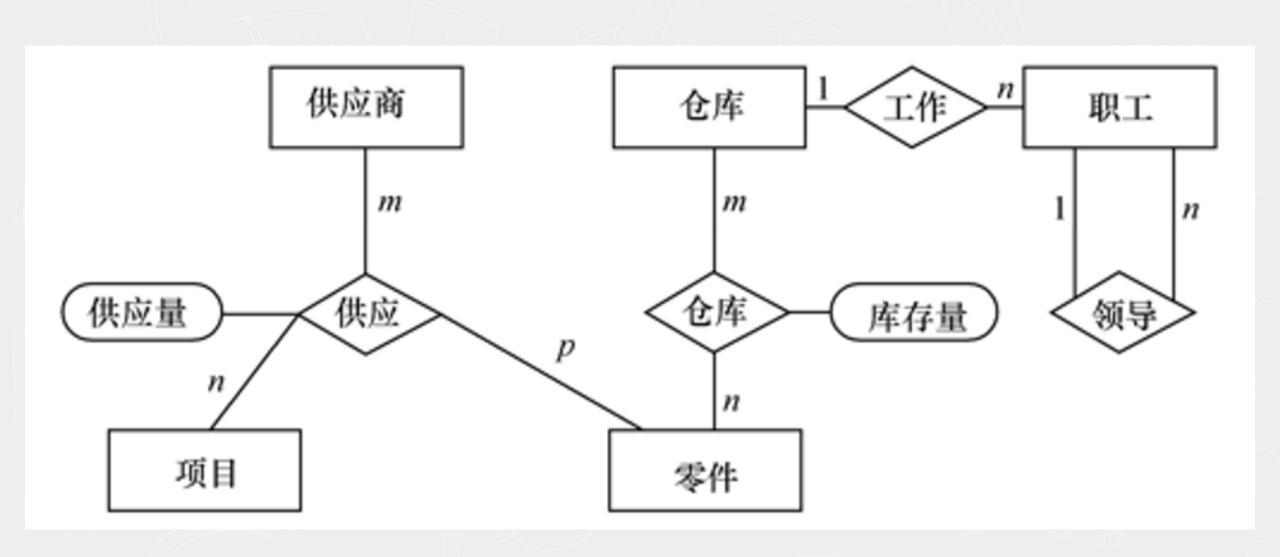


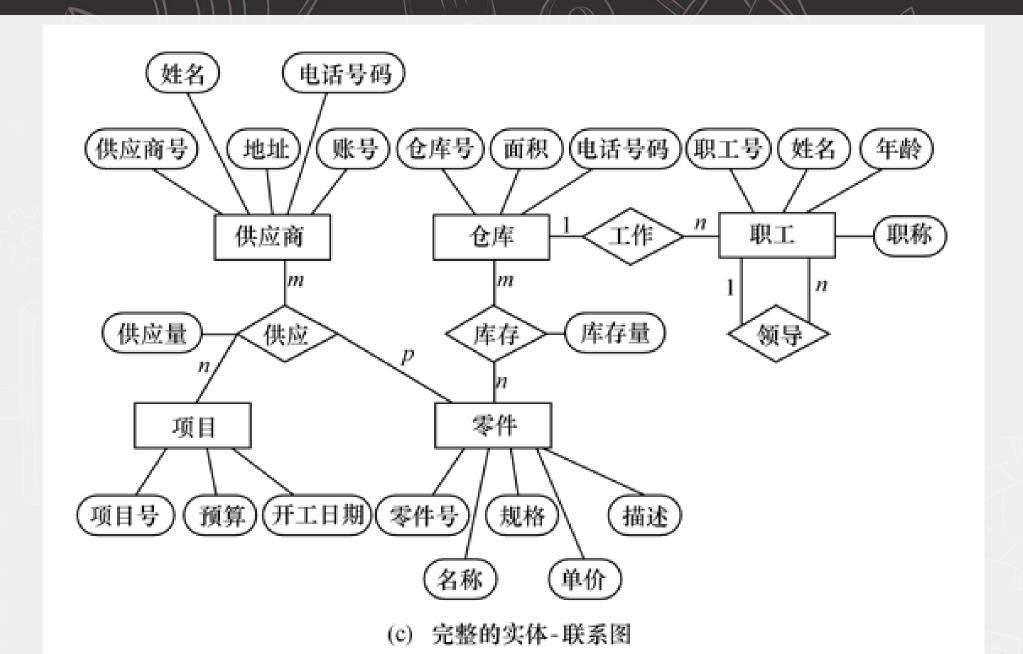


- ●供应商:属性有供应商 号、姓名、地址、电话 号码、账号
- ●项目:属性有项目号、 预算、开工日期
- ●职工:属性有职工号、 姓名、年龄、职称



- 这些实体之间的联系如下:
 - (1) 一个仓库可以存放多种零件,一种零件可以存放在多个仓库中,因此仓库和零件具有多对多的联系。用库存量来表示某种零件在某个仓库中的数量。
 - (2) 一个仓库有多个职工当仓库保管员,一个职工只能在一个仓库工作,因此仓库和职工之间是一对多的联系。
 - (3) 职工之间具有领导与被领导关系。即仓库主任领导若干保管员,因此职工实体型中具有一对多的联系。
 - (4) 供应商、项目和零件三者之间具有多对多的联系。即一个供应商可以供给若干项目多种零件,每个项目可以使用不同供应商供应的零件,每种零件可由不同供应商供给。



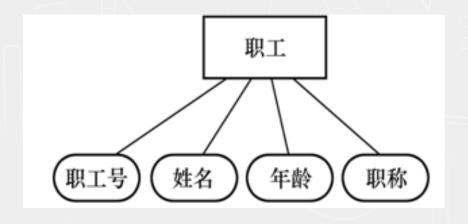


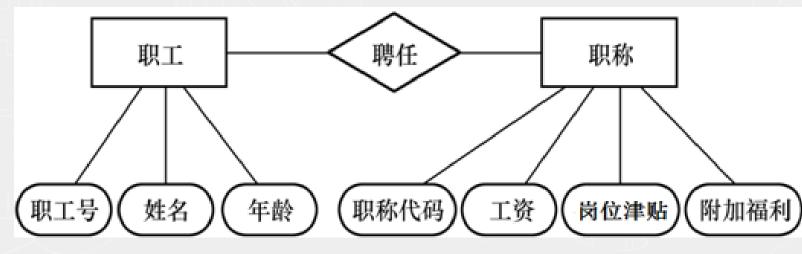
1. 实体与属性的划分原则

- 为了简化E-R图的处置,现实世界的事物能作为属性对待的, 尽量作为属性对待。
- •两条准则:
 - (1) 作为属性,不能再具有需要描述的性质。属性必须是不可分的数据项,不能包含其他属性。
 - (2) 属性不能与其他实体具有联系,即E-R图中所表示的联系是实体之间的联系。

[例1] 职工是一个实体,职工号、姓名、年龄是职工的属性。

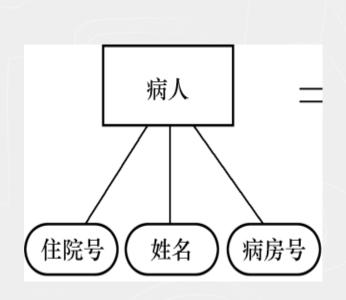
- 职称如果没有与工资、福利挂钩、根据准则(1)可以作为职工实体的属性
- 如果不同的职称有不同的工资、住房标准和不同的附加福利,则职 称作为一个实体更恰当

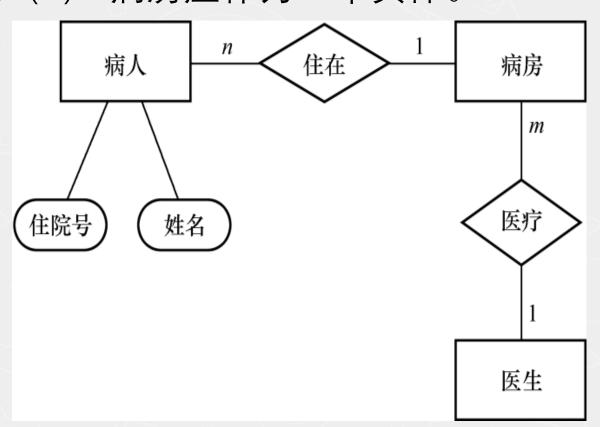




[例2] 在医院中,一个病人只能住在一个病房,病房号可以作为病人实体的一个属性;

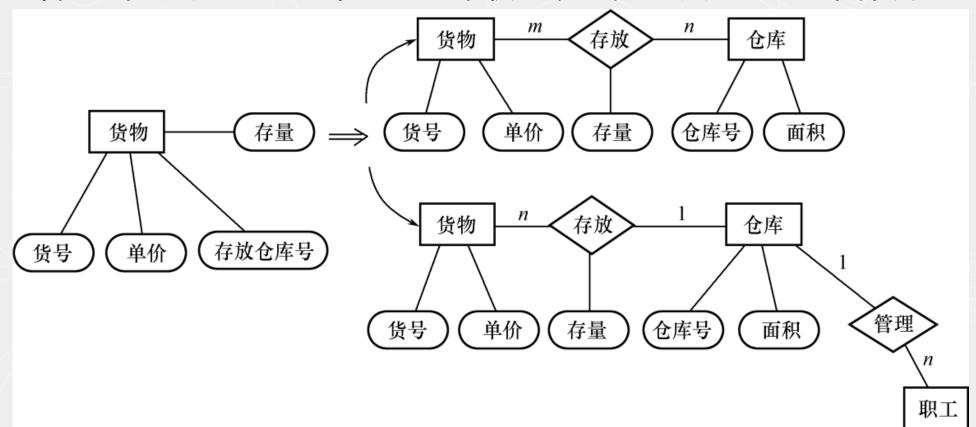
如果病房还要与医生实体发生联系,即一个医生负责几个病房的病人的医疗工作,则根据准则(2)病房应作为一个实体。





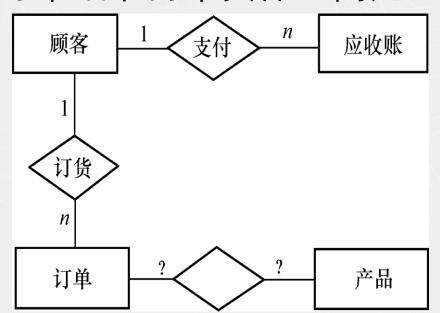
[例3] 如果一种货物只存放在一个仓库,那么就可以把存放货物的仓库的仓库号作为描述货物存放地点的属性。

如果一种货物可以存放在多个仓库中,或者仓库本身又用面积作为属性,或者仓库与职工发生管理上的联系,那么就应把仓库作为一个实体。



- •[例7.1] 销售管理子系统E-R图的设计。
 - 该子系统的主要功能是:
 - ●处理顾客和销售员送来的订单
 - ●工厂是根据订货安排生产的
 - ●交出货物同时开出发票
 - ●收到顾客付款后, 根据发票存根和信贷情况进行应收款处理

- 参照需求分析和数据字典中的详尽描述, 遵循前面给出的两个准则, 进行了如下调整:
 - (1)每张订单由订单号、若干头信息和订单细节组成。订单细节又有订货的零件号、数量等来描述。按照准则(2)订单细节就不能作订单的属性处理而应该上升为实体。一张订单可以订若干产品,所以订单与订单细节两个实体之间是1:n的联系。



- (2) 原订单和产品的联系实际上是订单细节和产品的联系。 每条订货细节对应一个产品描述,订单处理时从中获得当 前单价、产品重量等信息。
- (3) 工厂对大宗订货给予优惠。每种产品都规定了不同订货数量的折扣,应增加一个"折扣规则"实体存放这些信息,而不应把它们放在产品实体中。

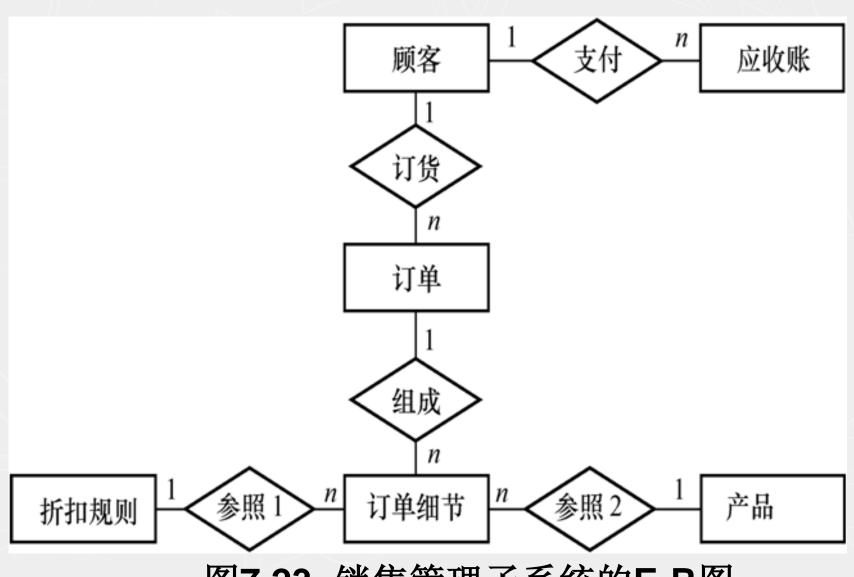


图7.23 销售管理子系统的E-R图

- 对每个实体定义的属性如下:
 - •顾客:{顾客号,顾客名,地址,电话,信贷状况,账目余额}
 - •订单:{<u>订单号</u>,顾客号,订货项数,订货日期,交货日期,工种号,生产地点}
 - 订单细则: {订单号, 细则号, 零件号, 订货数, 金额}
 - 应收账款: {顾客号,订单号,发票号,应收金额,支付日期,支付金额,当前余额,货款限额}
 - •产品:{产品号,产品名,单价,重量}
 - 折扣规则: {产品号, 订货量, 折扣}

2. E-R图的集成

- E-R图的集成一般需要分两步
 - 合并。解决各分E-R图之间的冲突,将分E-R图合并起来生成初步E-R图
 - 修改和重构。消除不必要的冗余, 生成基本E-R图

合并E-R图, 生成初步E-R图

- ●各个局部应用所面向的问题不同,各个子系统的E-R图之间必定 会存在许多不一致的地方,称之为冲突。
- ●子系统E-R图之间的冲突主要有三类:
 - 1属性冲突
 - 2命名冲突
 - ③结构冲突

1属性冲突

- ●属性域冲突, 即属性值的类型、取值范围或取值集合不同。
 - ▶例如零件号,有的部门把它定义为整数,有的部门把它定义 为字符型。
 - ▶年龄,某些部门以出生日期形式表示职工的年龄,而另一些部门用整数表示职工的年龄。
- ●属性取值单位冲突。
 - ▶例如,零件的重量有的以公斤为单位,有的以斤为单位,有 的以克为单位。

2命名冲突

- ●同名异义,即不同意义的对象在不同的局部应用中具有相同的 名字。
- ●异名同义(一义多名),即同一意义的对象在不同的局部应用中具有不同的名字。
 - ▶如对科研项目,财务科称为项目,科研处称为课题,生产管理处称为工程。
- ●命名冲突
 - ▶可能发生在实体、联系一级上
 - ▶也可能发生在属性一级上
 - ▶通过讨论、协商等行政手段加以解决

3结构冲突

- ●同一对象在不同应用中具有不同的抽象。
 - ▶例如,职工在某一局部应用中被当作实体,而在另一局部应 用中则被当作属性。
 - ▶解决方法:把属性变换为实体或把实体变换为属性,使同一对象具有相同的抽象。
- ●同一实体在不同子系统的E-R图中所包含的属性个数和属性排列 次序不完全相同。
 - ▶解决方法:使该实体的属性取各子系统的E-R图中属性的并 集,再适当调整属性的次序。

③结构冲突(续)

- ●实体间的联系在不同的E-R图中为不同的类型。
 - ▶实体E1与E2在一个E-R图中是多对多联系,在另一个E-R图中是一对多联系
 - ▶解决方法是根据应用的语义对实体联系的类型进行综合或调整

图7.25(a)中零件与产品之间 存在多对多的联系"构成"

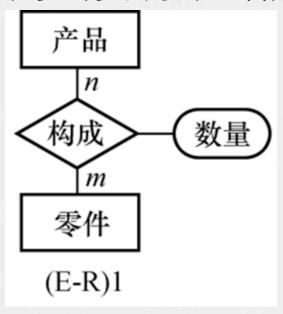
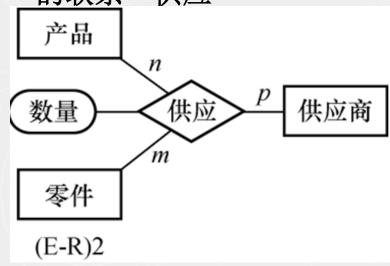
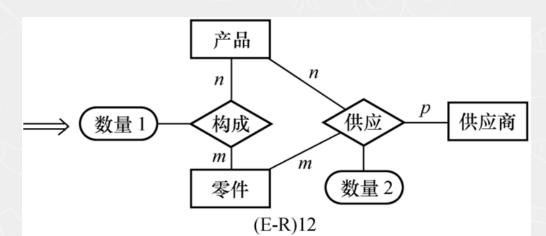


图7.25(b)中产品、零件与供应商三者之间还存在多对多的联系"供应"



合并两个E-R图, 如图7.25(c)



7.2.3 逻辑结构设计

- •逻辑结构设计的任务
 - 把概念结构设计阶段设计好的基本E-R图转换为与选用数据库管理系统产品所支持的数据模型相符合的逻辑结构

E-R图向关系模型的转换

•转换内容

- E-R图由实体型、实体的属性和实体型之间的联系三个要素组成
- 关系模型的逻辑结构是一组关系模式的集合
- 将E-R图转换为关系模型:将实体型、实体的属性和实体型之间的联系转化为关系模式

转换原则

- 1. 一个实体型转换为一个关系模式。
 - 关系的属性:实体的属性
 - •关系的码:实体的码

2. 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。
 - ① 转换为一个独立的关系模式
 - >关系的属性:与该联系相连的各实体的码以及联系本身的属性
 - ▶关系的候选码:每个实体的码均是该关系的候选码

- (1) 一个1:1联系的转换(续)
 - ②与某一端实体对应的关系模式合并
 - ▶合并后关系的属性:加入对应关系的码和联系本身的属性
 - >合并后关系的码:不变

- (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。
 - ①转换为一个独立的关系模式
 - >关系的属性:与该联系相连的各实体的码以及联系本身的属性
 - >关系的码:n端实体的码

②与n端对应的关系模式合并

- ▶ 合并后关系的属性:在n端关系中加入1端关系的码和联系本身的属性
- >合并后关系的码:不变
- >可以减少系统中的关系个数,一般情况更倾向于采用这种方法

- (3) 一个m:n联系转换为一个关系模式
 - ●关系的属性:与该联系相连的各实体的码以及联系本身的属性
 - ●关系的码:各实体码的组合

[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(<u>学号,课程号</u>,成绩)

- (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。
 - ●关系的属性:与该多元联系相连的各实体的码以及联系本身的 属性
 - ●关系的码:各实体码的组合

- (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)
 - > 适当调整属性的次序

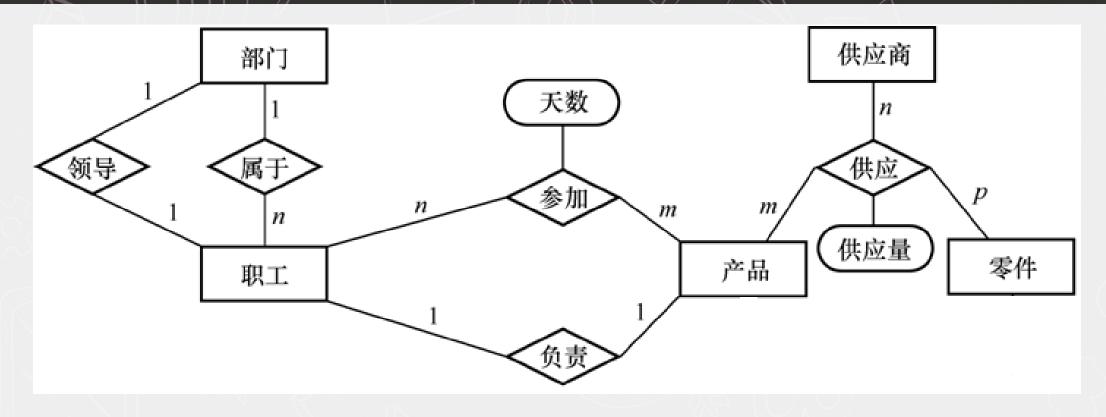


图7.28中虚线上部的E-R图转换为关系模型。 关系的码用下横线标出。

- ■部门(部门号,部门名,经理的职工号,…)
- ■职工(职工号、部门号,职工名,职务,…)
- ■产品(产品号,产品名,产品组长的职工号,…)
- ■供应商(供应商号,姓名,…)
- ■零件(零件号,零件名,…)
- ■职工工作(职工号,产品号,工作天数,…)
- ■供应(<u>产品号,供应商号,零件号</u>,供应量)

- •数据库逻辑设计的结果不是唯一的。
- 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。

7.2.4 数据库的物理设计

- •什么是数据库的物理设计
 - 数据库在物理设备上的存储结构与存取方法称为数据库的物理结构,它依赖于选定的数据库管理系统。
 - 为一个给定的逻辑数据模型选取一个最适合应用要求的物理结构的过程,就是数据库的物理设计。

- •数据库物理设计的步骤
 - 确定数据库的物理结构
 - •在关系数据库中主要指存取方法和存储结构;
 - 对物理结构进行评价
 - •评价的重点是时间和空间效率
 - 若评价结果满足原设计要求,则可进入到物理实施阶段。否则,就需要重新设计或修改物理结构,有时甚至要返回逻辑设计阶段修改数据模型。