

数据库系统课程教案

第 18 单元

学时：2

教材内容	第 7 章 数据库设计 (7.3~7.4)
基本知识点	视图的集成和用户子模式的设计，逻辑结构设计的概念和步骤，E-R 图向关系模型转换的原则、数据模型的优化
教学重点	理解逻辑结构设计的概念和步骤，掌握 E-R 图向关系模型转换的原则、数据模型的优化
教学难点	E-R 图向关系模型转换的原则、数据模型的优化
要求掌握内容	视图的集成和用户子模式的设计，逻辑结构设计的概念和步骤，E-R 图向关系模型转换的原则、数据模型的优化
教学思路，采用的教学方法和辅助手段，板书设计，重点如何突出，难点如何解决，师生互动等	<p>教学思路：</p> <p>一、复习旧课，巩固上次授课主要内容</p> <ol style="list-style-type: none"> 1、简述需求分析的任务和方法。 2、简述概念结构设计的方法和任务。 <p>二、导入新课，明确本次授课的目的与要求</p> <ol style="list-style-type: none"> 1、掌握视图的集成。 2、理解逻辑结构设计的方法和步骤。 3、掌握 E-R 图向关系模型转换的原则。 4、掌握数据模型的优化。 5、理解用户子模式的设计。 <p>三、讲解本次授课的具体内容</p> <p>教学方法：整合教学内容，强化基础训练；努力营造生动活泼的课堂气氛，搭建师生间良好的沟通渠道；采用多媒体教学与传统的板书设计相结合的方式，教学手段灵活多变。</p> <p>辅助手段：通过 PPT 幻灯片演示结合板书设计和例题加以阐述。</p>
本章思考题和作业	P241 第 5、6、7、10 题
主要教材参考资料	<ol style="list-style-type: none"> 1. 《数据库系统概论》，萨师煊，王珊，高等教育出版社，2014.9 2. 《数据库系统概论学习指导与习题解答》，王珊，张俊，高等教育出版社，2015.7
备 注	

本次授课具体内容

第 7 章 数据库设计

7.3 概念结构设计

(一)视图的集成：将各分 E-R 图合并，集成为一个整体的数据概念结构即总 E-R 图，分如下两步进行：

1、合并分 E-R 图，生成初步 E-R 图

由于各个局部应用面向的问题不同，且由不同设计人员设计，这就导致了各个分 E-R 图之间可能存在冲突。合并分 E-R 图的主要工作与关键所在就是合理消除各分 E-R 图的冲突。冲突主要分为如下三类：

(1)属性冲突：通常用讨论、协商等手段加以解决。

①属性域冲突：即类型、取值范围不同。如零件号，有的部门将它定义为整型，有的将它定义为字符型。又如年龄，有些部门以出生日期形式，而另一些部门用整数形式表示职工年龄。

②属性取值单位冲突：如零件重量可能以公斤、斤、以克为单位。

(2)命名冲突：可能发生在属性级、实体级、联系级上，也通常用讨论、协商等手段解决。

①同名异义：不同意义的对象在不同局部应用中具有相同名字。

②异名同义：同一意义的对象在不同局部应用中具有不同名字。如对科研项目，财务科称为项目，科研处称为课题。

(3)结构冲突

①同一对象在不同应用中具有不同的抽象。如职工在某一局部应用中被当作实体，在另一局部应用中被当作属性。解决方法：通常是把属性变为实体或把实体变为属性，使同一对象具有相同的抽象。

②同一实体在不同分 E-R 图所包含的属性个数和排列次序不完全相同：这是由于不同局部应用关心的是该实体的不同侧面。解决方法：使该实体的属性取各分 E-R 图中属性的并集，再适当设计属性的次序。

③实体之间的联系在不同局部视图中呈现不同的类型：如实体 E1 与 E2 在局部应用 A 中是多对多联系，而在局部应用 B 中是一对多联系；又如在局部应用 X 中 E1 与 E2 发生联系，而在局部应用 Y 中 E1、E2、E3 三者之间有联系。解决方法：根据应用语义对实体联系的类型进行综合或调整。

2、消除不必要的冗余，设计生成基本 E-R 图

初步 E-R 图中可能存在冗余数据和冗余实体间联系。冗余数据是指可由基本数据导出的数据，冗余联系是指可由其他联系导出的联系。冗余数据和冗余联系容易破坏数据库的完整性，给数据库维护增加困难。消除不必要冗余后的初步 E-R 图称为基本 E-R 图。

消除冗余的分析法：以数据字典和数据流图为依据，根据数据字典中数据项间逻辑关系的说明来消除冗余。如职工工资单中包括基本工资、补贴、应扣除水电费及实发工资，而实发工资可由前面各项推算，因此可以去掉，在需要查询实发工资时根据基本工资、补贴、应扣除水电费数据临时生成。

7.4 逻辑结构设计

(一)逻辑结构设计的任务：概念结构是独立于任何一种数据模型的信息结构。逻辑结构设计的任务就是把概念结构设计阶段设计好的基本 E-R 图转换为选用 DBMS 产品所支持的逻辑结构。

(二)逻辑结构设计的步骤

- 1、将概念结构转化为一般的关系、网状、层次模型。
- 2、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。
- 3、对数据模型进行优化。(三)E-R 图向关系模型的转换

1、转换要解决的问题：如何将实体、实体的属性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。

2、转换应遵循的原则

(1)一个实体型转换为一个关系模式，实体的属性就是关系的属性，实体的码就是关系的码。

(2)对于实体间的联系则有以下不同的情况：

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：若转换为一个独立的关系模式，则与该联系相连的各实体的码以及联系本身的属性均是关系的属性，每个实体的码均是关系的候选码；若与某一端实体对应的关系模式合并，则需在该关系模式中加入另一个关系模式的码和联系本身的属性。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。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：若转换为一个独立的关系模式，则与该联系相连的各实体的码以及联系本身的属性均为关系的属性，关系的码为 n 端实体的码；若与 n 端对应的关系模式合并，则合并后关系的属性是在 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。第二种方法可以减少系统中的关系个数，一般情况下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：与该联系相连的各实体的码以及联系本身的属性均是关系的属性，各实体码的组合为关系的码。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：与该多元联系相连的各实体的码以及联系本身的属性均是关系的属性，各实体码的组合是关系的码。(3)具有相同码的关系模式可合并。目的是减少系统中的关系个数，合并方法是将其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。(4)形成一般的数据模型后，还需向特定 DBMS 规定的模型进行转换，转换的主要依据是所选用 DBMS 的功能及限制，没有通用规则。对于关系模型来说，这种转换通常都比较简单。

(四)数据模型的优化：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。优化数据模型的方法是：

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。

如学生关系模式中存在数据依赖：学号→姓名，学号→性别，学号→年龄，学号→所在系选修关系模式中存在数据依赖：(学号, 课程号)→成绩

课程关系模式内部存在数据依赖：课程号→课程名，课程号→学分

学生与选修关系模式的学号之间存在数据依赖：学生. 学号→选修. 学号

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、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。如分析可知课程关系模式属于 BC 范式。

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。

(1)并不是规范化程度越高的关系就越优。

(2)非 BCNF 的关系模式虽从理论上分析会存在不同程度的更新异常，但若在实际应用中对此关系模式只是查询，并不执行更新操作，则就不会产生实际影响。对一个具体应用来说，到底规范到什么程度，需权衡响应时间和潜在问题两者的利弊。一般说来，第三范式就足够了。如在关系模式学生成绩(学号, 英语, 数学, 语文, 平均分)中存在函数依赖：学号→英语，学号→数学，学号→语文，学号→平均分，(英语, 数学, 语文)→平均分。显然有学号→(英语, 数学, 语文)，所以该关系模式中存在传递函数依赖，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。

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或合并，以提高数据操作的效率和存储空间利用率。常用的分解方法是：

(1)水平分解：把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。

(2)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。分解原则是经常在一起使用的属性从 R 中分解出来形成一个子关系模式。垂直分解必须确保无损连接性和保持函数依赖。

(五)设计用户子模式：将概念模型转换为全局逻辑模型后，还应根据局部应用需求，结合具体 DBMS 特点，设计用户的外模式。定义数据库模式主要从系统时间效率、空间效率、易维护等角度出发，而定义用户外模式更应注重考虑用户的习惯与方便。

1、使用更符合用户习惯的别名：对于某些局部应用，由于改用了不符合用户习惯的属性名，可能会使他们感到不方便，因此在设计用户的子模式时可以重新定义某些属性名，使其与用户习惯一致。

2、针对不同级别的用户定义不同的视图，以满足系统对安全性的要求。如对于关系模式中产品(产品号, 产品名, 规格, 单价, 车间, 生产负责人, 成本, 合格率, 质量等级)，可以建立两个视图：

一般顾客视图：产品 1(产品号, 产品名, 规格, 单价)

销售部门视图：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。这样就可以防止用户非法访问本来不允许他们查询的数据，保证了系统的安全性。

3、简化用户对系统的使用：若某些局部应用中经常使用某些很复杂的查询，为方便用户，可将这些复杂查询定义为视图，用户每次只对定义好的视图进行查询，大大简化了用户的使用。

本次授课小结

本次授课讲述了视图的集成、逻辑结构设计的概念和步骤、E-R 图向关系模型转换的原则、数据模型的优化、用户子模式的设计。

学生课后复习时应着重于其中的第 1、3、4 点内容，为进一步学习后续章节打好基础。

实验

无