

陈其明 编著

工程数据库原理

测绘出版社

工程数据库原理

陈其明 编著

测绘出版社

内容简介

本书针对工程数据库的特点,对复杂对象的模拟、操作和推理进行了比较全面的论述。本书不受现有规范数据库原理的局限,从更广义的角度建立工程数据库的理论体系。主要内容包括:结构层次及语义层次的数学模型,面向对象数据库原理,演绎数据库原理,复杂对象及类型的数据模型,对象的标识与符号对象模型,广义对象操作以及涉及复杂对象的推理。这些都是在建立各种工程数据库中所面临的具有共性的课题。本书所述内容一方面扩充了经典数据库原理,另一方面可为工程数据库的开发实行提供概念上和方法上的指导。

本书可供有关专业的科研工作者、工程技术人员、大专院校教师、研究生和大学生等参考。

工程数据库原理

陈其明 编著

*

测绘出版社出版

北京市昌平印刷厂印刷

新华书店总店科技发行所发行

*

开本 850×1168 1/32·印张13.75·字数303千字

1991年3月第一版·1991年3月第一次印刷

印数0,001—1,900册·定价15.00元

ISBN 7-5030-0432-0/TP·6

目 录

绪论.....	(1)
第一章 结构层次的数据模型.....	(14)
§ 1.1 层次数据模型.....	(19)
§ 1.1.1 层次数据库的内涵.....	(20)
§ 1.1.2 层次数据库的外延.....	(24)
§ 1.1.3 层次数据库的视图.....	(26)
§ 1.1.4 层次数据库的存取途径.....	(29)
§ 1.1.5 层次数据库的系统结构.....	(32)
§ 1.1.6 层次数据库的操作.....	(33)
§ 1.2 网状数据模型.....	(36)
§ 1.2.1 网状数据库的内涵.....	(36)
§ 1.2.2 网状数据库的外延.....	(47)
§ 1.2.3 网状数据库的视图.....	(47)
§ 1.2.4 网状数据库的存取途径.....	(49)
§ 1.2.5 网状数据库的系统结构.....	(52)
§ 1.2.6 网状数据库的操作.....	(54)
§ 1.3 关系数据模型.....	(60)
§ 1.3.1 关系数据库的内涵.....	(61)
§ 1.3.2 关系数据库的外延.....	(66)
§ 1.3.3 关系数据库的视图.....	(67)
§ 1.3.4 关系数据库的存取途径.....	(70)
§ 1.3.5 关系数据库的系统结构.....	(71)
§ 1.3.6 关系数据库的操作.....	(73)
§ 1.3.7 规范关系数据库理论.....	(91)

§ 1.4	本章小结	(102)
第二章	语义层次的数据模型	(105)
§ 2.1	实体—联系数据模型	(106)
§ 2.1.1	基于实体—联系的数据模拟	(106)
§ 2.1.2	数据完整性	(112)
§ 2.1.3	数据操作的特点	(112)
§ 2.1.4	实体—联系方法的扩充	(113)
§ 2.2	函数数据模型	(116)
§ 2.2.1	对象与函数	(118)
§ 2.2.2	泛函数	(122)
§ 2.2.3	数据操作语言	(123)
§ 2.2.4	与关系数据模型的联系	(124)
§ 2.2.5	与抽象数据型的联系	(126)
§ 2.2.6	在工程数据库中的应用前景	(127)
§ 2.3	语义网络数据模型	(128)
§ 2.3.1	语义的网络表达	(128)
§ 2.3.2	语义约束	(134)
§ 2.3.3	数据操作的特点	(135)
§ 2.4	超图数据模型	(138)
§ 2.4.1	超图数据结构	(138)
§ 2.4.2	基于超图的数据模拟	(139)
§ 2.4.3	数据操作的特点	(147)
§ 2.5	本章小结	(147)
第三章	演绎数据库	(149)
§ 3.1	一阶逻辑	(153)
§ 3.1.1	命题逻辑	(154)
§ 3.1.2	一阶谓词逻辑	(159)

§ 3.2	逻辑程序	(165)
§ 3.2.1	Horn 子句逻辑	(165)
§ 3.2.2	Herbrand 解释与模型	(170)
§ 3.2.3	合一与归结	(174)
§ 3.2.4	极小不动点	(178)
§ 3.3	数据库与逻辑	(182)
§ 3.3.1	数据库的逻辑模型论	(182)
§ 3.3.2	数据库的逻辑证明论	(184)
§ 3.4	演绎数据库系统	(188)
§ 3.5	基于逻辑的数据库语言	(192)
§ 3.5.1	PROLOG——不适当的选择	(192)
§ 3.5.2	DATALOG	(193)
§ 3.5.3	函数项的引入	(194)
§ 3.5.4	对否定的处理	(195)
§ 3.6	本章小结	(198)
第四章	面向对象数据库	(200)
§ 4.1	问题的提出	(200)
§ 4.2	面向对象程序系统	(209)
§ 4.2.1	主要概念	(212)
§ 4.2.2	smalltalk —80	(217)
§ 4.2.3	C + +	(220)
§ 4.3	面向对象数据库系统	(225)
§ 4.3.1	对象标识	(226)
§ 4.3.2	封闭	(229)
§ 4.3.3	类型层次与继承	(235)
§ 4.3.4	算子重叠与后结合	(245)
§ 4.3.5	面向对象数据库的可扩充性	(247)
§ 4.4	本章小结	(248)

第五章 对象结构的数据模型	(250)
§ 5.1 复杂对象	(250)
§ 5.2 对象的半序格模型	(253)
§ 5.3 部分对象	(266)
§ 5.4 对象的图示	(268)
§ 5.5 本章小结	(269)
 第六章 对象的标识	(271)
§ 6.1 对象标识的表达和时间	(271)
§ 6.2 对象标识方法的评价	(273)
 第七章 符号对象模型	(279)
§ 7.1 符号对象	(281)
§ 7.2 符号对象的操作	(286)
§ 7.2.1 对象的相等比较	(286)
§ 7.2.2 对象的构造	(288)
§ 7.2.3 对象的复制	(290)
§ 7.3 逻辑数据结构	(292)
§ 7.4 本章小结	(294)
 第八章 类型的结构和符号模型	(295)
§ 8.1 型的结构模型	(296)
§ 8.2 型的结构交换	(301)
§ 8.2.1 型的重构	(301)
§ 8.2.2 保息重构	(308)
§ 8.3 类的符号对象模型	(311)
§ 8.4 封闭与继承	(313)
§ 8.5 符号类模型的逻辑数据结构	(318)
§ 8.6 统一的符号对象模型	(319)

§ 8.7	本章小结	(323)
第九章	元类	(324)
§ 9.1	元类的基本概念	(325)
§ 9.2	元类的性质	(328)
§ 9.2.1	元实例化	(328)
§ 9.2.2	元类复盖	(329)
§ 9.2.3	元类的部分实例化	(330)
§ 9.3	元类与面向对象系统	(331)
§ 9.4	本章小结	(332)
第十章	广义对象操作	(334)
§ 10.1	对象代数	(334)
§ 10.1.1	子对象和部分对象在操作上的区别	(335)
§ 10.1.2	广义并与交	(336)
§ 10.1.3	广义投影	(341)
§ 10.1.4	广义选择	(348)
§ 10.1.5	广义联接	(349)
§ 10.2	对象演算	(361)
§ 10.2.1	虚型	(361)
§ 10.2.2	重构表达式作为对象演算法则	(363)
§ 10.2.3	附加限制条件的相容重构	(364)
§ 10.2.4	对象演算的优点和限制	(365)
§ 10.3	关于类的操作	(366)
§ 10.4	本章小结	(370)
第十一章	复杂对象推理	(372)
§ 11.1	以函数项表达复杂对象的逻辑程序系统	(373)
§ 11.1.1	LDL	(373)

§ 11.1.2	L P S	(375)
§ 11.1.3	C O L	(375)
§ 11.2	扩展的逻辑程序系统HILOG	(376)
§ 11.2.1	HILOG 的语法	(378)
§ 11.2.2	HILOG 的语义	(380)
§ 11.2.3	HILOG 的应用	(394)
§ 11.3	复杂对象推理的符号过渡法	(396)
§ 11.3.1	L P T 系统	(397)
§ 11.3.2	由 HILOG 到 LPT 的映射	(403)
§ 11.3.3	关于映射的进一步讨论	(410)
§ 11.4	HILOG 与一阶逻辑程序模型理论的关联	(415)
§ 11.5	本章小结	(417)
附录 1	拆卸 σ 的定义	(419)
附录 2	证明概要	(420)
参考文献		(422)

绪 论

作为当代计算机系统的重要组成部分，数据库系统的发展与程序语言，软件工程和人工智能的发展并驾齐驱且相互融合，其应用范围正在从统计、管理等领域迅速扩大到实际工程应用。然而，面临工程界“非经典”应用领域内巨大挑战的数据库技术却表现出明显不足。几乎所有商品化数据库管理系统都无法有效支持诸如CAD/CAM等工程应用。这一现象的出现并非偶然。跟所有其它学科一样，数据库也具有简单到复杂的发展过程。以关系模型为代表的数据库模拟主要针对简单对象；以PROLOG为典型工具的演绎技术也只有支持涉及简单对象的数据库推理。而工程对象的主要特点在于具有复杂的结构和内涵，工程数据库系统必须具备对于复杂对象模拟与推理的功能。一方面可将工程数据库技术视作经典数据库技术的扩充；另一方面，由于经典数据库理论是以规范关系为基础的，而复杂对象的模拟、操作和推理主要面向非规范关系，工程数据研究的实质性进展已不能依赖经典数据库技术，而有待于理论上的进一步发展。

工程应用五花八门，但工程数据的模拟组织、控制管理和操作推理仍有其共性。本书的目的在于：讨论如何针对复杂对象建立统一的数学模型，定义统一的广义操作，奠定形式化的演绎推理基础，贯通各种相关的概念，构成关于工程对象模拟、操作、推理的理论体系，以作为发展工程数据库系统的根据。

六十年代计算机操作系统问世以后，在可随机存取的磁盘系统和以分时、共享为特点的资源管理技术支持下，数据库系统获得了长足的进步。在数据库技术发展的初期，人们致力于数据文件的组织、管理、检索和更新等方面的研究，这些研究为物理层次的数据管理奠定了基础。随着数据库技术向较高层次发展，在

逻辑层次对数据模拟和操作的研究愈加深入，逻辑层次和物理层次的界面也愈加分明。为了建立统一的数据组织的操作环境，数据模型的概念应运而生。最初出现的两个典型数据模型是层次模型和网状模型，它们各自反映了一种数据组织的结构约束，以及针对各自结构所定义的一组通用数据操作。随后出现的一个重要数据模型就是人们所熟知的关系模型，它具有较高的数据独立性和较严密的数学基础。随着对关系数据库研究的深入，人们提出了一系列有关理论，如函数依赖、多值依赖，Armstrong公理系统等等。关系数据库理论的形成是数据库技术走向成熟的重要标志。

网状数据模型、层次数据模型、关系数据模型都属于结构层次的数据模型，尽管它们彼此之间还存在着表达层次上的差异。结构层次的数据模型侧重于描述数据模拟的结构约束，而非数据对象本身的内涵。例如，无论何种数据，如果存在于网状数据库，就必须以网状数据库所特有的关系为单位组织；如果存在于关系数据库，就必须以关系数据库所特有的关系为单位组织。在结构层次之上的是语义层次的数学模型，如实体—联系数据模型、语义网络数据模型等等，它们的特点是直接表达数据对象本身的内涵和其间联系，而不具体描述数据的组织结构。由于舍弃了低层次的信息而着重于高层次的数据说明，语义层次的数据模型对于客观系统具有更直接的表达能力，基于这些模型构成的数据库系统也更便于操作。应当指出，数据模型由低级到高级的发展是一步一步实现的，前者往往成为后者的基础。关系数据库的用户只须使用关系操作来处理数据而无须涉及关系数据存贮的文件结构，然而，关系数据库正是建立在文件系统基础之上的。按照语义，层次数据模型组成的数据库往往建立于某些结构层次数据模型的基础之上。例如一个数据库系统配备了函数数据操作语言，但支持它的是一关系数据库，只是二者之间的接口对用户透明而已。

除数据模拟与操作之外，数据库技术的另一引人注目的发展

方向是推理，即所谓演绎数据库技术。

数据库和逻辑的关系在近年来已日趋明朗。当前，数据库系统和程序系统融合成统一计算机应用系统的需求，以及以逻辑程序作为新一代计算机基本程序语言的设想，更进一步推动了数据库与逻辑程序的结合，并形成了数据库技术的一个重要分支：演绎数据库技术。这一学科的建立，始于H·Gallaire, R.Reiter等人的工作和将PROLOG引入关系演算。

PROLOG是一种利用模式匹配的符号式语言，其基础是一阶谓词逻辑的一个分支，Horn子句逻辑。PROLOG具有事实语句和法则语句。分别用以表达关系外延和演绎法则。例如关系person（人）和parent（父母）：

person		parent	
name		parent	son
a		a	b
b		b	c
c		d	c
d		c	e
e			

可用以下PROLOG语句表达(a, b, ...为常量)：

person(a).	parent(a, b).
person(b).	parent(b, c).
person(c).	parent(d, c).
person(d).	parent(c, e).
person(e).	

而演绎法则（即产生关系视图的法则）可以PROLOG法则语句表达如下(grand-parent为祖父母，X，Y，Z为变量)：

person(X) :- parent(X, Y)
parent(Y) :- parent(X, Y)

$\text{grant-parent}(X, Z); \neg \text{parent}(X, Y), \text{parent}(Y, Z)。$

法则右端表示条件的取舍，分隔符“，”代表“ λ ”；法则的左端表示在右端条件满足的情况下获得的结论。法则也可表示成递归的形式，如(ancestor为祖先)：

$\text{ancestor}(X, Y); \neg \text{parent}(X, Y),$
 $\text{ancestor}(X, Z); \neg \text{parent}(X, Y), \text{ancestor}(Y, Z),$

在一般关系数据库中，用户只能针对实际存在的关系数据提问，例如“c是不是e的父或母？”。然而，在演绎数据库中，由于引入法则系统可对于诸如“找出e的祖父母”或“找出e的全部长辈”之类咨询作回答。显然这是一般数据库系统所不及的。演绎功能已为现代数据库系统所不可缺少，数据库推理也成为数据库理论的一个重要组成部分。

然而，以关系数据库理论为代表的经典数据库原理是面向“简单对象”的，这一理论只适用于规范化的关系，即每一属性值为不可再分解的原子项的关系。相应的演绎数据库理论也只基于一阶逻辑，不适合于含高阶项的系统。从本质上讲，关系理论和一阶逻辑程序理论均源于广义的一阶理论。这一理论上的局限性使得经典数据库系统缺乏直接处理“复杂对象”的能力，而工程对象往往具有复杂的结构和内涵，无法为规范关系和一阶公式所直接描述。一般而言，复杂对象泛指在结构（语法）和操作“语义”方面均具有嵌套层次关系的对象。对于复杂对象模拟和推理的研究和实行，是建立工程数据库系统的核心工作。

下面我们就数据模拟、操作和推理分析复杂对象静态方面和动态方面的主要特点，以展示工程数据库与传统数据库的区别，以揭示各种工程数据库的某些共性。

1. 复杂对象模拟

一数据库为客观世界一系统之模型。数据模拟是对数据结构及数据涵义的形式化描述，前者体现在如关系模型等各种基

本数据模型：后者体现在各种语义模型，从软件理论角度而言二者被统一于型或类的概念之下。现在基本数据模型，如关系模型等，均有明显的局限性：关系DBMS不直接支持结构聚合（aggregation）和综合（generalization）；关系操作对于许多工程应用对象如影像毫无意义以至非法，等等。简言之，面向简单对象的传统数据模型在数据描述、数据操作，检索途径、组织方式和系统控制方面，都无法满足处理复杂对象的要求。

绝大多数工程对象属于复杂对象。复杂对象是指具有复杂结构（因而也具有复杂的操作语义）的对象。我们通过下列例子来说明。

（a）一个由若干关系所构成的聚合（aggregation）是一个复杂对象。例如，一个电路模块（MODULE）由若干部分（PART）构成，每个部分具有一定的功能（FUNCTION），以下列关系的聚合来表示：

MODULE [M_1 D, NUMBER, PRICE]

PART [P_1 D, M_1 D, QUALITY]

FUNCTION [F_1 D, P_1 D, CODE]

按照一般的理解，上述聚合被视为一个复杂对象，而其中每个关系则视为一个简单对象。

（b）一个具有嵌套断言作为变量的断言是一个复杂对象。

例如断言

employee(lee, degree(ms, 1981))。

可视为一个复杂对象，因为它的一个变量本身又是一个断言。反之，断言

degree(ms, 1981)。

可视为一个简单对象，它无须涉及其它断言即可完全定义。

（c）一个由多种不同类型的对象所构成的对象是一个复杂对象。例如在图0.1中，省作为一个复杂对象具有若干属性和部件，包括一项省内城市的集合和一项分省地图；每个城市和分省

地图本身又是复杂对象；地图对应着一个图形，为拓扑数据结构所描述。整个对象由不同抽象层次的部分构成，这些部分分别存放于关系数据库和图形数据库中。图中给出了这一复杂对象型的示意。

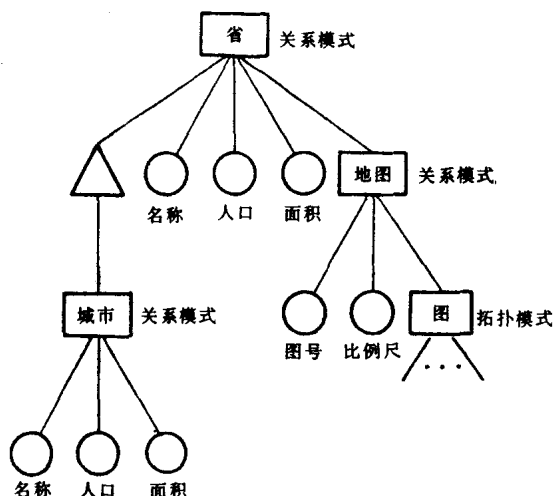


图 0.1 复杂对象

一般而论，复杂对象具有以下特点：

(1) 可逐层分解为低级的对象，一个复杂对象由不同的部件构成；每一个部件对象又可参与其它对象的构成，从而可为多方共享。

(2) 具有多种数据结构，如元组(tuple)、集合(set)、串(list)，以及这些结构的各种组合。

(3) 一个复杂对象的部件对象可具有各自不同的型，从而归入不同的类。

(4) 一个复杂对象不同层次的描述，或不同的部件对象，可为不同的数据模型所支持。换言之，一个复杂对象的各个部分可能分布于不同的数据库。

以上特点对数据库系统提出了巨大的挑战。仅从数据模拟的

角度而言,对复杂对象的描述可划分为多个彼此正交的空间以反映各种结构关系,至少包括如下内容:

(1) 分类(Classification)

对象按其公共特性而归入不同的类,从而具备不同的型以及相应的合法操作——方法,这一处理和数据库设计中的局部化原则相符合,并构成一个分类层次结构。

(2) 概括(generalization)

概括强调从某些具有共性的类中提取出更加综合的特性以构成高层次的类。反之,由一个高层次的类可衍生出低层次的具体化的类,并允许后者自动继承前者的结构与操作特性。继承关系是传递的,从而构成一个对象型的层次结构,或类的层次结构。如前所述,鉴于一个类可能具有非单系祖先类,实际上存在于一个继承网络。

(3) 聚合(aggregation)

一个对象可从结构上划分为多个部分,或称具有多个部件对象,这种划分可逐层细化从而形成一个构造的层次结构。聚合的概念由[Smi77]首先提出,但未涉及聚合链的内部关系而被称之为原子聚合(atomic aggregation)。进一步的划分还可包括合成、版本、表达的层次。

虽然工程对象千差万别,所需数据结构也各不相同,但这些对象结构是具有一定规律的。探讨这一规律,建立对象结构的数学模型,是工程数据库理论的基本课题之一。

2. 对象标识的稳定性

在一般程序系统中,数据标识包括在程序说明之中,一旦程序运行结束,这些数据的标识随之失去意义,或者说,它们对其它程序不起作用。在数据库系统中,数据和程序是分离的。数据的标识在数据库系统中确定,而不是由数据库应用程序给出。因此,同一数据的标识跨越程序,具有时间上的稳定性。然而,传统数据库系统在标识数据的稳定性方面并没有严格的要求。例如,

关系模式

零件编号	名 称	材 料	价 格	

给出组装某种机器所需零件的记录，并以零件编号为该关系的关键字及无组标识。如果由于某种原因对零件编号进行了修改，则在确定组装上述机器所需何种零件的查询中就可能发生错误。这个例子向我们提示，对象的标识应当独立于数据，即相对于数据更新保持稳定。这一点对于工程数据库环境尤为重要，原因可归纳如下：

(a) 对象的高度共享。同一对象的标识符可出现于多个以其为部件的高层次对象的记录中，面向于多种应用或多个用户集团，其标识必须对于数据独立。

(b) 数据库应用环境的分布。同一对象可在分布式系统中，或在主系统和分系统之间来回传递，因此其标识必须对于地址独立。

(c) 对象的时序性。同一对象可具有多个版本（可具有相同的关键字），必须分别标识。

讨论标识的另一意义在于标识符本身可用作对象的“替身”，如本书后面所述，这一概念不仅支持复杂对象的管理，而且支持复杂对象的推理。

3. 复杂对象操作

在经典关系数据系统中，对关系的操作是建立在关系代数和关系演算基础上的，更确切地说，是建立在关系数学模型基础上的。为了定义针对复杂对象的广义操作，则有必要建立关于对象的广义数学模型，这正是本书要讨论的内容之一。

4. 复杂对象推理

数据库推理是数据库和人工智能的共同目标，也是第五代计