

NF²关系数据模型及其在工程数据建模中的应用

聂培尧 (山东财政学院计算机系)

摘要 目前关于支持层次结构的复杂工程对象的研究,大多是扩充原有的关系DBMS,但这种方法在概念上和执行等方面都存在许多缺点。本文介绍一种嵌套关系数据库系统中的NF² (Non First Normal Form) 数据模型。该模型可按一种规范的方法处理标准关系(1NF)、列表和层次结构,最后讨论NF²数据模型在工程数据建模中的应用。

在CAD/CAM环境中,常常需要嵌套层次较深的结构和多种不同的元组类型来表示复杂工程对象^[1,2]。由于数据聚类和避免不必要联结之类的语义(概念)与执行上的原因,这种复杂对象^[3]不能简单地作为普通的标准关系(1NF)存储。这就要求DBMS能提供一种直接支持层次结构的功能。

文献[3,4]论述了一种扩充关系DBMS来更有效地支持层次结构的方法。该方法的实现是加入了一些新定义的属性类型和系统产生的标识码(以表示层次联系和加快联结速度)并且提供了适当的新DML语言(如对复杂对象的插入、检索、更新、删除等操作)的专用接口。这种方法的优点是对原有DBMS的修改部分较少,但缺点是对于底层的DBMS来说复杂对象仅是一种特殊的对象,并且不能按照通常的标准关系进行处理。也就是说,这种方法不能像标准关系那样支持复杂对象的投影、选择以及聚类等操作。为了真正实现标准关系与复杂对象的统一表示,应当开发一种能体现这两种数据类

型的数据模型。

一种较好的方法是扩充数据模型,使之能在数据模型中表示标准关系与复杂对象,又不放弃高层关系查询语言。这种扩充方法的主要思想是允许在一关系中关系本身也可以作为属性值出现。这就意味着应放弃关系必须满足1NF的要求。我们称这种关系为非第一范式(NF²)关系^[4,5,6]亦即带关系值属性的关系^[7]。

一、工程DBMS中的数据模型

数据模型是构成DBMS的基础,根据目前的工程DBMS,我们可把它的数据模型分为三大类。

1. 传统的数据模型

现在很多用于工程设计、制造等应用领域的DBMS仍是使用60年代末,70年代初开发的关系、网状、层次之类的传统数据模型^[8]。基于网状模型的一个DBMS的例子是由phil-

SHM+模型最重要的方法就是它对动态数据建模和有关动态及静态模式的一致建模方法所作的贡献,图2和图17中结构的相似性说明了SHM+中数据和事务建模方法的共同性。由于数据库对象的语义总是对数据库行为指定了约束,所以在数据库建模中定

义事务(动态对象定义了数据库行为)是合理的。

(未完待续)

[纪岳、何未艾、王红、赖维生、陆强、钟颖译自、姚舞达校自ACM Computing Surveys, 1988, vol.20 No3]

ip开发的PHIDAS^[9]。虽然,使用这种数据模型的DBMS也具有一定的优点^[10],但由于这种数据模型最初是出于商用需要而开发的,自然就不能真正适合工程应用的需求。事实上,很多研究人员^[11,12]都指出了这种基于记录的数据模型不能反映高级CAD/CAM应用需求的一些弱点。

2. 扩充的数据模型

这方面的研究工作主要是对现有的数据模型进行改造,使其成为适合工程应用的数据模型。这种扩充工作大都是在关系模型上进行的,如ADT-INGRES^[13]系统,它在标准关系模型中使用了一种新的用户定义的抽象数据类型ADT。RAD^[14]在扩充标准关系模型时也使用了类似的方法。嵌套关系数据模型NF^[15],扩充了关系模型,允许嵌套关系,支持层次结构。R³D²^[16,18]中的NF²^[16,17,18]数据模型也使用了抽象数据类型的概念。XS-QL^[1]则使用了层次结构的概念。这里的层次(复杂)对象是由不同的标准关系中的元组通过聚集,即系统产生标识码的方法构成的。

3. 新的数据模型

近年来已有很多研究人员把注意力集中到开发一种新的数据模型,以支持高级工程应用。Maier等人^[19]提出了一种新的数据模型。在这种模型中使用了一种面向对象的程序设计语言Smalltalk以及对象管理系统之间的接口。Zdonik和Wegner^[20]描述了一种从面向对象的程序设计语言环境到数据库系统的耦合方法。Batory和Buchmann^[21],Batory和Kim^[22]及Dittrich等人^[23]描述了一种基于扩充的E-R模型^[24]的系统。这些系统的主要目标之一是支持复杂对象的表示及几何数据的建模。

二、工程数据管理中的数据库系统及其适应性

1. 基于传统的关系模型的数据库系统

传统的关系数据库系统在事务处理中得到了十分广泛的应用,因为事务处理中的信

息结构非常简单。

然而,传统的关系模型中所有的关系表均满足1NF的要求,即每个属性均为原子数据类型,所以对复杂对象的处理变得十分困难,因为应用对象建模时属性不得不存储于若干关系中,这些关系与其前驱关系必须通过联结属性来联系。为了说明这一点,本文中引用了文献^[10]中一个机器人模拟的例子。一个机器人有名字(name)及手臂(arm)。在描述机器人的手臂(含有若干个轴及一个抓柄)的属性时,使用传统的关系数据库系统(如INGRES或DB2)只允许表示原子属性。如在DB2中可定义两个关系表如下:

[例1]

```
CREATE TABLE robots (name char (10),
                        arm char(10))
CREATE TABLE arms(arm_id char(10),
                    axis...,
                    gripper...)
```

这里robots关系中的属性arm及arms关系中的arm_id构成了联结属性。我们可以通过使用以下的查询来构造出整个结构:

[例2]

```
SELECT name, arm, axis, gripper
FROM robots, arms
WHERE robots.arm=arms.arm_id
```

当前,在工程数据库系统研究领域已提出了这样一种方法,在关系数据库系统之上增加一个软件前端处理机制,解决不同结构上的对象分解问题。

2. 扩充的关系数据库系统

传统的关系数据库系统不允许定义非原子属性的表。为了克服这一局限性,可在关系数据库系统之上开发一附加层,以便把数据库系统的逻辑视图转换成用户接口的概念视图。这种方法的优点是:

(1) 系统可继承关系数据模型的优点,如非过程化的用户接口,实体间的任意联系以及一些完整的理论基础等;

(ii) 关系DBMS仍是有效的,无需再开发其他的系统。

这种方法同时也存在很多缺点,如:

(i) 关系DBMS不是专门支持面向结构的聚类,所以对分散存贮于若干关系中的对象进行选择操作时,其效率非常低。

(ii) 此方法还需要一种逻辑模型到概念模型之间的变换过程。

(iii) 概念视图常常是若干关系表之间联结操作的结果,这种联结可能产生不可更新的对象结构。

(iv) 为了把概念模型中给出的对象结构转换为1NF的逻辑模型,结构不得不进行分割。当这种分割不适当时,将会由于破坏了原有的函数依赖或完整性约束而导致修改及插入异常。

3. NF²数据模型

以上所述缺点可以说均归咎于逻辑及概念视图的处理不一致,导致数据库结构不能反映出对象的语义。解决这一问题的唯一方法是避免不同层之间的转换过程,对数据模型进行扩充。这实质上是扩充数据模型的基本概念,支持层次结构,如嵌套关系模型(即非1NF关系或称NF²关系)等。

Schek和Pistor^[20]曾提出了一种扩充关系数据模型,其中属性既可以是原子类型也可以是关系。为了支持层次结构对象的定义及操纵,他们使用了扩充的SQL语言HPDL(Heidelberg Data Base Language)^[13],于是前面的例1可定义如下:

[例3]

```
CREATE robots SET (TUPLE (
    name: STRING(10),
    arm: SET(TUPLE(
        arm_id, STRING
            (10),
        axis: ...,
        gripper: ...
    ))
))
END
```

以上定义中的*SET(TUPLE(*相当于1NF中的关系表TABLE。

在定义了嵌套关系对象结构之后,若想检索机器人的全部信息,则可简单地使用以下查询:

```
SELECT r
FROM r IN robots
WHERE r.name='R2D2'
```

这一检索过程隐含了对机器人对象的每一子关系的存取操作,而不需要象例2中那样通过SELECT语句重新构造对象结构。这是因为层次结构已不再是分割存贮于1NF的关系表中了。

此外,NF²数据模型的另一优点是支持隐含的向前引用。这就意味着:若一个对象是另一对象的子关系,则在选择所有的子结构信息时不再需要进行联结操作。同样,NF²也支持后向引用。这可根据对每一选择的子结构简单地抽取路径信息来完成。以下是用户从不同的层次结构中查询信息的例子:

[例4]

```
SELECT ar.arm_id, ax.axis_id
FROM r IN robots,
    ar IN r.arm,
    ax IN ar.axis,
WHERE r.name='R2D2'
```

另一种可能的操作是类属函数father-of的使用。该函数的执行结果是返回相应的高层对象。如father_of(arm)返回的是与arm相对应的机器人类型的对象。

三、NF²模型在工程实体建模中的应用

1. 概念模型

实体类的建模是CAD/CAM系统中的基本任务。工程实体的建模属性是根据具体的应用而确定的。例如在机器人模拟系统中,装配体零部件的基本描述可包括这样一些属性:标识符identifier,重量weight,材料material,参考坐标系中的位置及方向ref-frame,参考坐标系的数目ref-coord以及几何描述的引用ref-geom等。

下面的例子是装配体零部件的NF²模式描述:

[例5]

```
Objects: SET(TUPLE (
    obj_id : INTEGER,
    weight : REAL,
```

```

material : STRING(10)
ref_fram : TUPLE(
    row1 : LIST(4 FIX REAL),
    row2 : LIST(4 FIX REAL),
    row3 : LIST(4 FIX REAL),
    row4 : LIST(4 FIX REAL),
),
ref_coord : INTEGER
ref_gcom : INTEGER
))
    
```

2. 几何建模

CAD/CAM系统中的另一问题是主要用于几何造形的几何信息的建模。几何形状数据的表示法可分为CSG与B_reps两种^[10]。

边界表示法B_reps (Boundary representation) 是一种用组成几何对象的面以及与面之间的拓扑结构关系表述几何体形状的方法。图1示出了一个立方体的边界表示法。图中虚线以上表示的是拓扑信息，下面表示的是度量信息。根据这种方法，我们可对每一几何对象进行抽象的层次建模。

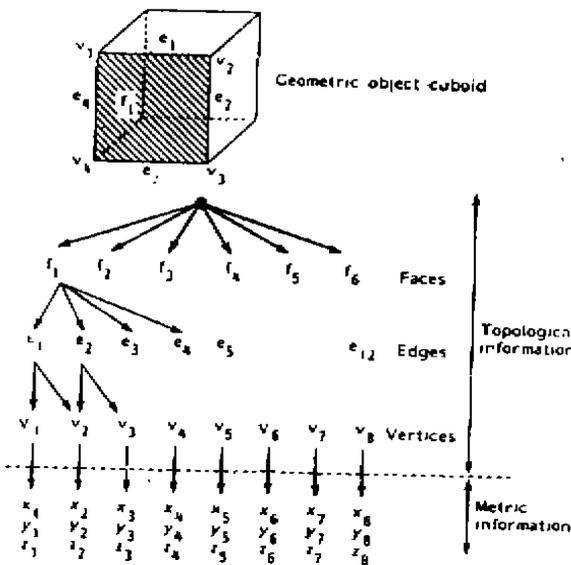


图1 立方体的边界表示 (引自文献[10])

构造实体几何法CSG (Constructive Solid Geometry) 中使用了一些称为体素的基本形体,通过体素的集合运算,可使它们组合表示出复杂的立体形状。如设一形体P,它可由几个体素构成的部分图形 P_i ($i=1,2,\dots,n$)的并集表示: $P = \bigcup_{i=1}^n P_i$ 。式中的各部分图形 P_i 是图形元素 P_{ij} ($j=1,2,\dots,m$)的共同部分,而每个图形元素 P_{ij} 可

由 m_j 个不等式表示 $G_{ij}(X) \geq 0$,它可由交集表示: $P_{ij} = \bigcap_{k=1}^{m_j} P_{ijk}$, $P_{ijk} \equiv \{X | G_{ijk}(X) \geq 0\}$ 。根据以上式子可得出描述形状P的表达式为: $P = \bigcup_{i=1}^n \bigcap_{j=1}^{m_i} P_{ij}$ 。

实体的CSG描述是一包含中间结点(子对象)的树状结构(称为CSG树),在一CSG树中,每一非终结点表示一运算,如刚体旋转或集合运算。终结点则可表示一运动变量,也可表示一基本对象。每个基本对象可由其自身的参数,如长、宽、高及它们的相对位置来描述。以下是在NF²模型中使用B_reps方法描述的几何信息的一数据库模式。

[例6]

```

Geometry : SET(TUPLE(
    geo_id : INTEGER,
    faces : SET(TUPLE(
        face_id : INTEGER,
        edges : SET(TUPLE(
            edge_id : INTEGER,
            start_vertex : INTEGER,
            end_vertex : INTEGER
        ))
    ))
)
    
```

```

Metric : SET(TUPLE(
    vertex_id : INTEGER,
    x : REAL,
    y : REAL,
    z : REAL,
))
    
```

B_reps表示在层次结构中具有很好的聚类效果,使得对拓扑信息的检索不需要对不同的关系表作联结操作,而仅使用单步操作即可。然而,如下所示,若使用标准的数据库模型描述几何信息的B_reps表示时,则必须分割几何信息并离散地存储于不同的关系表中。

当想从如下数据库检索一几何对象时,如同前面的例2一样,必须在所有关系表中进行联结操作,这显然既麻烦又费时。

```

geo_object [geo_id, ...]
face [face_id, geo_obj, ...]
face_to_edge [face, edge]
edge [edge_id, v1, v2, ...]
vertices [vertex_id, x, y, z]
    
```

四、结束语

总之,传统的数据模型(如标准关系型)不适合工程应用。在这种情况下,工程应用中的外部实体不能以一种自然的方式在数据库中建模,从而导致了外部实体与内部实体之间不能进行自然的映射,因此外部实体不得不分割存储于若干离散的关系表中。对实体的整体操作需要对数据库中的若干相关的关系表进行联结操作。然而,使用NF²模型可克服以上的缺点。NF²模型可允许层次结构对象作为一个整体存储于数据库中。工程应用中的外部对象在数据库中被表示为一种嵌套的NF²结构形式,这为把复杂的工程对象作为整体进行操作提供了十分方便的功能。

参考文献

- [1] Lorie, R. A., Plouffe, W., "Complex Objects and Their Use in Design Transactions", "IEEE Engineering Design Applications, 1983.
- [2] Kalz, R. H., Information Management for Engineering Design, Springer-Verlag, 1985.
- [3] Haskin, R. L., Lorie, R. A., "On Extending the Functions of a Relational Database System", Proc. SIGMOD'82.
- [4] Jaeshke, G. et al., "Remarks on the Algebra of Non First Normal Form Relations", Proc. ACM SIGACT-SIGMOD'82.
- [5] Schek, H. J., Pistor, P., "Data Structures for an Integrated Database Management and Information Retrieval System", Proc. VLDB'82.
- [6] Schek, H. J., "Towards a Basic Relational NF² Algebra Processor", Proc. Inter. Conf. on Foundations of Data Organization, 1985.
- [7] Schek, H. J. et al., "An Algebra for the Relational Model with Relation-Valued Attributes", Information Syst., Vol.11, No.2, 1986.
- [8] Date, C. J., An Introduction to Database Management Systems, Addison Wesley, 4th edition, 1986.
- [9] Fisher, W. E., "PHIDAS: A Database Management for CAD/CAM Application Software", CAD, Vol.11, No.3, 1979.
- [10] M. Dürr et al., "Using Conventional and Nested Relational Database Systems for Modelling CIM Data", CAD, Vol.21, No.6, 1989.
- [11] Kemper, A., et al., "An Analysis of Geometric Modelling in Database Systems", ACM. Comput. Surv. Vol.19, No.1, 1979.
- [12] Kent, W., "Limitations of Record-Based Information Models", ACM TODS, Vol.4, No.1, 1979.
- [13] Stonebraker, M. et al., "Application of Abstract Data Type and Abstract Indices to CAD Database", "Proc. Database Week, 1983.
- [14] Osborn, S. L. et al., "The Design of a Relational Database System with Abstract Data Types for Domains", ACM TODS, Vol.11, No.3, 1986.
- [15] Kemper, A., "Abstract Data Types in Geometrical Database", Proc. 20th Hawaii Inter. Conf. System Science, 1987.
- [16] Kemper, A. et al., "An Object-oriented Database System for Engineering Applications", Proc. ACM SIGMOD'87.
- [17] Dadam, P. et al., "A DBMS Prototype to Support Extended NF² Relations: An Integrated View on Flat Tables and Hierarchies", Proc. ACM SIGMOD'86.
- [18] Pistor, P. et al., "Designing a Generalized NF² Data Model with an SQL-type Language Interface", Proc. VLDB'86.
- [19] Maies, D. et al., "Object-oriented Database Development at Servio Logic", IEEE Database Eng. Vol.8, No.4, 1985.
- [20] Zdonik, S. B. and Wegner, P.,

- "Language and Methodology for Object-oriented Database Environments". Proc. 19th Hawaii Inter. Conf. System Science, 1988.
- [21] Batory, D. S. and Buchmann, A. P., "Modular Objects, Abstract Data Types and Data Models: A Framework", Proc. VLDB'84.
- [22] Batory, D. S. and Kim, W., "Modeling Concepts for VLSI CAD Objects", ACM TODS, Vol.10, No.3, 1985.
- [23] Dittrich, K. R. et al., "Complex Entities for Engineering Applications", Proc. 5th E-R Conf. 1987.
- [24] Chen, P.P.S., "The Entity-Relationship Model: toward a Unified View of Data", ACM TODS, Vol.1, No.1, 1976.
- [25] Nie Peiyao (聂培尧), "SNDM: A Semantic Network Data Model for CAD DBMS", Proc. Inter. Conf. on Japanese-Sino Sapporo Computer Applications, Japan, 1990.
- [26] 聂培尧等, "工程DBMS中几何实体的数据描述及操作", 计算机工程与应用, No.1, 1990.
- [27] 聂培尧, 徐秋元等, "一种几何数据库的建模方法", 计算机工程 (待发表), 1990.
- [28] 聂培尧, 徐秋元, "一种基于CODM的工程数据库的建模方法", 计算科学技术与应用, No.1, 1990.
- [29] 聂培尧, 蒋泽军, "工程数据库中复合对象研究概况评述", 软件产业, No.1, 1990.

投稿须知

致尊敬的作者

《计算机科学》系全国性科技情报刊物之一, 以其新颖、准确、及时为特色。

为了繁荣我国计算机科学领域, 本刊坚持以“百花齐放, 百家争鸣”方针为指导思想, 恳切希望广大从事计算机科学事业的仁人志士给以支持与合作。本刊报导范围包括: 以国内外计算机科学的发展动态为主, 兼顾涉及面广的方法论与技术, 和反映新苗头、能起承先启后作用的研究成果和实践原则。内容涉及程序理论、软件工程、操作系统、程序语言和编译、文件管理和数据库、人工智能、应用等。本刊也报导有具体学术内容的国际会议见闻、出国参观和访问、工作之感想等。

来稿经录用刊出后, 即酌付稿酬。投稿者须知如下:

1. 内容充实、重点突出、顺理成章、立论确凿;
2. 文字精炼、言简意赅, 一般在5000字左右; 并附100—200字中、英文摘要及题目、作者和工作单位英译名;
3. 译稿应附原文;
4. 来稿请用文稿纸书写, 字迹要清楚, 上角或下角、英文大、小写字母须分明; 附图可用草图, 但必须正确无误;
5. 一律不采用复印稿, 计算机打印稿件须单面、隔行印刷;
6. 文末请列出主要参考文献;
7. 请勿一稿两投。

《计算机科学》编辑部