维普资讯 http://www.cqvip.com

计算机科学 1996 Vol. 23№. 6

时态数据库基本框架的研究*)

严小卫
(广西师范大学数学与计算机科学系 桂林 541004)

樊明武 方锦清
(中国原子能科学研究院)

摘 要 We discuss an infrastructure for temporal databases, list some consensus on the infrastructure, show our viewpoints on some unsolved problems, and give some guiding recommendation. We believe that our work would provide a helpful base for the further studies of temporal databases.

关键词 Temporal database, Data model, DBMS.

1. 引 言

近二十年的时态数据库(TDB)的研究和发展，远没有给商业上带来什么巨大的影响，究其原因主要是，缺少一个共同接受的数据模型，缺少一个时态数据库的基本框架。由于在各种应用中对时间的语义有不同的要求，使 TDB 的研究领域呈现出一派"繁荣"景象。这种"繁荣"使得 DBMS 产家仍在等待，以免自己过早决策的产品将来会被其它竞争产品所代替。在 TDB 的理论研究上也是如此。在暴雨般的查询语言和数据模型的面前，是零星的时态数据库设计和时态查询优化等方面的结果，其部分是由于缺少一个后者赖以生存的基础，即一个普遍认可的数据模型的确定；更基本地，我们甚至缺少一个全面而标准的术语集。例如，同样是描述某事件的发生时间这一概念，在不同的模型中就有可能被称为固有时间、逻辑时间、现实世界时间、或有效时间等等。可见，我们很有必要建立一个共同的时态数据库基本框架，以提供一个对时态数据进行标准化、实现和研究的基础。93 年 6 月来自各国 44 名著名学者在美国德克萨斯就召开了这样一个研讨会。目前，已取得了一些一致意见，建立了一套标准术语，但在某些方面仍未有统一的结果。由于关系模型的重要性，本文先给出若干分析时态关系模型的要素，借此介绍一些现有的时态关系模型；其次指出目前时态关系数据模型的最主要分歧。因为 SQL 是对应关系数据模型的标准化了的查询语言，所以这种分歧也具体体现在对时态 SQL 的态度上；然后我们将分析时态关系模型应具有的性质，接着提出一个时态数据库的基本框架；最后考虑 TDBMS 的结构定义及实现问题等。

2. 基本时态关系数据模型及其查询语言

2.1 现状

自从 Codd 在 70 年提出关系数据模型以后，关系数据库获得极大成功。因此，目前的时态数据模型大都是以关系数据模型为基础。78 年以来已经提出

[4]David W. Cheung et al., Maintenance of Discovered Association Rules in Large Databases: An Incremental Updating Technique

[5]David W. Cheung et al., A Fast Distributed Algorithm for Mining Association Rules

[6]R. Agrawal and R. Srikant, Fast algorithms for mining association rules. In Proc. 1995 Int. Conf. Data Engineering, Taipei, March1995

[7]R. Srikant and R. Agrawal, Mining generalized association rules. In Proc. 1995 Int. Conf. VLDB, Zurich, Switzerland, Sept. 1995

[8]R. Agrawal and J. C. Shafer, Parallel Mining of association rules: Design, implementation, and experience. In IBM Research Report, 1996

[9]J. Han, and U. Fu. Discovery of multiple-level association rules from large databases, Same to[7]

[10]J. S. Park et al., An effective hash-based algorithm for mining association rules, Same to[2]

*)本文由国家自然科学基金和国家 863 计划项目资助

维普资讯 http://www.cqvip.com

了超过30种的关系数据模型的时态扩展模型。其中超过一半的模型仅支持有效时间，少数几个仅支持处理时间，有的则是双时态的。我们可以从下列四个方面来考虑一个时态数据模型的类型。

(a)有效时间的表示。一般通过标记在每个属性或元组上的事件、区间、或者时态元素来表示有效时间。

(b)处理时间的表示方式。通过标记在每个属性、元组、或者若干元组上的事件、区间、三元事件、或者时态元素来表示。

(c)属性值的表示方式。有可能是原子的、或者是有序对的、或者是三元的、或者用集合表示等等。

(d)模型的齐次性及合并性。其可能组合自然使模型对应分成四种类型。

下面我们简要分析几个例子。

* 历史数据库模型[2]：在这个模型中，每一个关系模式都附加一个取值为时间元的属性 STATE 来表示时间，以及一个布尔属性 EXISTS 来表示元组是否在该时间元上存在。因此，该模型用有效时间事件来标记元组，从而是齐次的。

* TQeul 模型：每一个关系都增加四个隐含属性，包括插入和删除元组的处理时间，元组开始和停止有效的时间。

* Tansel 的模型：支持基于演算的查询语言 HQuel 以及例示时间语言。该模型仅允许一种类型的对象，即有效时间关系。但支持四种属性：时间变化的或非时间变化的、原子的或集合的。Tansel 还把这个模型推广到 N1NF 的关系，给出了一个时态关系代数和一个等价的时态关系演算的形式定义，并研究了嵌套的时态关系的范式[3]。

* Gadia 的模型[4]：首次提出时态元素的概念。开始时模型是齐次的，导致在卡氏乘积时丢失时态信息，所以后来取消了齐次性的限制。通过把时态元素多维化还可以表示各种时间。属性值仍是从时态元素到属性值域的函数。笔者认为该模型值得研究，它不仅通用，而且表达能力强，表示方法"漂亮"，具有很好的性质，如时态元素的封闭性等等。

* DM/T 模型：对每一个用户定义的关系，都有一个特殊的、系统产生和维护的处理时间关系，称为储备(backlog)。其元组包含一个处理时间值以指出请求时间，以及一个属性值以指出请求类型(如插入、删除、或者修改等)。

* M_{TG} 模型[7]：定义了两个规范的数据模型，分别称为 M_{TU} 和 M_{TG}。M_{TU} 代表的是元组时标的一类模型，称之为时态未分组的。而 M_{TG} 代表另一类具有时态维的复杂属性值的模型，称之为时态分组的。每一个模型又分别给出一个基于逻辑的查询语言，作为标准来衡量各种模型对应查询语言的表达能力。在 M_{TG} 中，定义每一个属性域为从时态域到通常值域的函数。其对应语言 L_h 是 Codd 的关系演算的一个多方面扩展，增加了许多变量类型，包括历史值域上的和时间上的变量。有以下结论：分组模型的表达能力比未分组的强，但通过一个组代表(surrogate)可以把一个时态未分组模型扩展成时态分组模型；若干代数和若干基于演算的查询语言是未分组完备的；L_h 是分组完备的；已知代数还未有一个是分组完备的，等等。M_{TG} 模型与 Gadia 模型有类似的地方，且作为一个更高层次的评判标准提出。感兴趣的读者可参考有关文献。我们在后面还要提到它。

绝大多数模型都分别提出了与其对应的时态查询语言。有些是扩展传统关系代数的，有些是扩展 Quel 的，有些则是扩展 SQL 的。例如：

* SQL+T 语言：它基于元组时标模型，其中每个元组都有四个时标，用来标记有效时间和处理时间。它在 SQL 基础上增加了 *valid*、*when* 和 *as of* 子句。

* TempSQL：基于 Gadia 的属性时标的齐次数据模型，增加了 while 子句，采用时态表达式来说明时态关系。

* SQL^T：基于属性时标的非齐次嵌套关系，在 SELECT 语句中增加时态特性[5]。

* IXSQL[6]：基于区间扩展的关系代数，除了传统的五种基本关系操作外，还支持另两种关系代数操作，支持新的比较运算符和新的标度函数。不区分用户定义时间和有效时间。这也是一种较吸引人的语言，希望引起读者注意。

2.2 分歧

传统数据库系统对时间的支持是在用户定义时间级上的，这种实现被限制在某个范围内。目前普遍认为应该整理各种数据模型，整理基于代数的或演算的各种查询语言，因为与现有众多模型和语言思想相重复的文章仍在不断出现。但是，时态数据库界却是一个相当松散的群体，每一个研究项目都是基于某一组特别的假设和方法。例如，在查询优化中，研究必须假定某一数据模式和某一查询语言。在这种情况下，各研究项目选择的语言应该相同。不幸的是，不存在这样一个能吸引足够多人们的语言。对于如何看待 TSQL 有下列三种最基本观点。

维普资讯 http://www.cqvip.com

(1)SQL 数据模型已经相当接近于可以支持时态应用中的需求,所需增加的主要是在代数运算符中和语法上。这一观点的具体体现之一就是 IXSQL 模型。

(2)时间是数据的基本属性,应该以基本的方式结合进数据模型和查询语言。它不赞同扩展现已被接受的 SQL92 标准,而应该直接努力将时间添加到 SQL3 的设计中,争取在 95 至 96 年度产生一个时态查询语言标准。

(3)第三种观点对时间的看法与第二种观点相同,只是赞同上述二分叉方法,即短期地定义一个时态扩展的同时,也长期地定义一个广泛的扩展。

这三种观点显然是相互冲突的,但每一个都有它自己的支持者,都在组织强有力的技术上的论点论据来支持他们的立场。此外,在每一种观点下,同样也存在着许多的、次级上的分歧。很清楚,在这基本问题都未达成共识的情况下,很难选择一个标准的数据模型。

2.3 要求

为方便起见,我们把 SQL 的一个时态扩展称为 TSQL。我们将讨论一个时态关系模型应该具有的功能和要求。

(1)时态数据库中有三种类型的时间,即用户定义时间、有效时间、处理时间。大多数时态应用需要的是一个 SQL 的双时态扩展。因此,在 CREATE TABLE 语句中应能说明快照关系、有效时间关系、处理时间关系、以及双时态关系。同时在 FROM 子句中应能包含多种关系并进行运算。

(2)SQL 的扩展应该是向上兼容的。它还应满足可简化性,即一个 SQL 查询,如果作为一个 TSQL 查询应该产生一个时态关系,该时态关系在某一时间的切片产生一个快照关系;另一方面,在相同时间处,对时态数据库切片产生一个快照关系,作为一个 SQL 查询在这个快照关系上所计算出来的结果应该与前面产生的快照关系相同。

(3)一些应用如规划中需要对将来时间的支持。因此,TSQL 应该支持将来时间。注意,处理时间应该不允许超过 NOW。

(4)关于离散时间和连续时间的问题已被哲学家、数学家和物理学家们讨论了几千年还仍未得出结论。不过在时态数据库领域,普遍都同意在表示模型上采用离散时间。因此,TSQL 中的时标也应该是离散的。

(5)开区间[a,b)和闭区间[a,b]的选择,其中 a、b 为时标。在边疆时间模型,两者不可比较。在离散时间模型中,[a,b]=[a,b+1)。因此,TSQL 在表示时态值时应该支持两种区间。

(6)未分组的和分组的完备性。一般认为应该要求 TSQL 具有未分组完备的表达能力。至于是否要求分组完备性,由此分出两大阵营。一种意见认为现在缺少任何分组完备数据模型的实现经验,在 TSQL 中引入它是很危险的。另一种意见是,由于方便用户是我们最终的目标,所以即便是复杂的实现,我们也应该把分组完备性作为模型的一个要求。问题的焦点是实现。虽然通过采用系统维护的"代表",可给出一个从分组完备数据模型到未分组完备模型的映射,但该映射仍未被实现。直观地说,实现 M_{TG} 方法中的联接运算似乎也是相当困难的。不管怎么说,分组完备性确实能支持"属性历史"这么一个相当强的概念。

2.4 框架

前面提到的传统关系模型的扩展都希望能保持关系模型的简单性,希望在一个元组中表示一个对象的所有信息,还希望容易实现和有效的查询计算。大多数模型的目标都包括使自己适合于数据显示、数据存储和时态语义获取。然而,在一个数据模型中同时满足所有的目标是困难的。因此,我们提出了一个分离特性的基本框架,其主要思想是采用一个简单的概念性数据模型来获取时间变化对象的基本语义,并且适合于显示、存储和查询计算等,如图 1 所示。逻辑数据库设计产生概念性模式,然后在进行物理数据库设计时提炼成某一表示数据模型的关系模式。查询语言本身就可基于该概念性数据模型。查询优化则可以在逻辑代数上进行,由表示的代价模型参数化。当然也可以在表示模型的代数上进行。显示应该与存储表示无关,并利用现存的具有方便显示格式的若干数据模型。

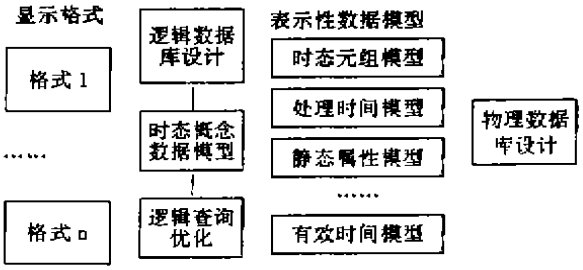

图 1 概念性和表示性数据模型的相互关系

这一基本框架依赖于各种数据模型的语义等价性,即必须能够在概念性模型和各种表示模型之间

维普资讯 http://www.cqvip.com

相互映射。一个适当的概念性模型应该能证明自己与许多现存的表示模型等价。这种等价性可以基于快照等价性。但可以证明一个分组关系能快照等价于大量未分组关系，其中仅有一个关系具有相同的信息内容。一种更合适比较异类表示的方法是采用强等价，也称为历史等价。一个适合的概念性数据模型的例子就是我们提出的 TER 模型。

在开发概念性时态数据模型中需要验证许多问题，例如：齐次性、合并性、属性值时标与元组时标、嵌套层数、吸除(vacuuming)、周期时间、制造时间(manufactured time)、重构等等。

3. 高级关系模型及非关系时态数据库

目前有许多复杂数据库应用对对象之间的时空关系有较强的需求，如：多媒体、计算机集成生产(CIM)，地理信息系统(GIS)等。而关系数据模型对复杂对象的表达能力几乎还不够。因此，也出现了一些基于语义数据模型和对象数据模型的方法。我们将借助这些模型进一步讨论下一代时态数据库的基本框架。这些模型或者扩展已有模型，或者在已有模型上建立一新模型，或者自己新创方法。例如：Info-logical 数据模型、时间扩展实体关系模型(TERM)、时态 EER 模型、ERT 模型[8]、时态对象数据库模型、时态面向对象数据模型(TOODM)、时态知识表示模型(OSAM-/T)、OODAPLEX 模型、TIGUKAT 模型。另外，时态主动数据库是目前较为引人注目的模型之一[5]。

那么，下一代时态数据模型(NGTDM)应该是怎样的呢？我们认为它不应该是一个受限于目前时态模型的扩展模型，更不应该是一个扩展的关系模型。时态模型的作用就是构造和刻画对象的时态数据、时态信息和时态关系。通常其主要方法之一是采用不同粒度上的抽象。这样，时态模型可分成结构、操作和约束三个不同部分讨论。其中结构为对象及其相互关系，操作(也称为行为)是对象的处理，约束为对象结构和操作在时间上的完整性的规则。一个时态对象被定义为一个或多个时态属性的集合。时态属性描述对象在时间上的结构、操作和约束特征。一个时态约束被定义为一条带有效期的数据库规则，它由三部分组成：事件、条件和动作，都可引用时间点，并至少事件和条件之一引用时间点，以使规则为时态的。一个 NGTDM 应具有下列性质。

+ 其时态查询语言应考虑到传统语言设计的问题，如方便使用、可优化性、表达能力和可实现性；

+ 应支持满足对象一致性的模式演化，这取决于模型对时间的支持；

* 时间的刻画应独立于数据模型，时间应建立起自己的方法学。基本的时间概念包括：双时态区间、双时态片刻、双时态元、双时态元素、双时态时间点、区间操作、片刻操作、时间点操作、以及它们之间的转换；

* 除了定义 NGTDM 基本框架的概念外，还应能定义一些附加构造来说明条件即约束。

一个 TDBMS 还应考虑下列应用需求：多时间性，如一张支票至少有三个日期(签发日期、兑现日期、实际兑现日期)；多时间粒度及其转换；取消操作；时态数据的不精确性；绝对和相对时间；时态关系；在不同时间粒度上定义的不同数据库的合并等等。注意：时态推理已超出时态数据库的范围。由于不同应用领域具有对时间的不同要求，如地理数据的时间元为地理年代，所以数据定义语言应提供用户定义时间元的功能。在同一数据库中也可能有多种时间元，应区分时刻和时间元。应支持时态聚集和时态联接操作，支持变化数据类型的定义、复合事件的定义、数据的不相交区间和周期数据的定义。

目前 NGTDM 的基本框架大都是给出一些概念而不是技术，因为现阶段还未能在技术上达成一致意见。较成功的一个例子为主动时态数据库的基本框架[5]。

4. 实现

前面已经阐明，实现问题很关键。目前仅有少数几个通用时态 DBMS 被实现，其中最好的也许是 TQuel 原型[1]。当然存在许多建立在现有商业 DBMS 上的或文件系统上的专用时态数据库，但在这些众多实现中，时间的含义和解释由用户应用程序实现，而不是由 DBMS 本身所理解。我们的目标是建立一个通用的 TDBMS。

现已提出若干索引技术，其中一些是原空间索引技术的扩展，而另一些则是专门为时态数据库设计的。设计 TDBMS 体系结构的一个关键部分是为时态数据库操作定义一个标准代数。这些操作将作为时态查询和更新的内部目标表示，并作为查询处理和优化等系统模块的基础。只可惜至今没有一个充分接受的 TDB 代数，也几乎没什么利用时态数据库特征的关于并发控制、恢复、安全和其它技术的研究。

一个 TDBMS 不应该与非时态 DBMS 相差太

维普资讯 http://www.cqvip.com

远，特别是在物理存储层，数据要仍能作为字节流存储。有四个主要模块：在最低层，存在一个存储系统负责存储一致的数据。该层能使用一个现有的存储系统，如UNIX文件系统，被存储对象作为字节流被检索，并在系统高层根据系统目录信息被解释为对象。数据记录可以分簇以使存取更有效，更高一层的模块之一是现有执行算法库。另一个可被执行算法存取的模块是现有索引结构库。还有一个模块就是查询优化模块。它在执行算法库中为一个时态查询适当挑选算法来产生一个执行策略。其它的标准模块，例如查询语法分析器，将产生一个内部查询表示，然后由查询优化器进行优化。问题是应选择哪些形式的操作来表示和优化查询。这又回到了我们最初提出的问题。最自然的折衷方法是前面介绍的第二种观点。

对于如何评价各种时态索引结构和查找技术，不是一两个典型TDB应用就能说明问题的，它依赖于下列几个具体应用中的因素：

* 数据库大小即数据容量：可由快照对象的平均数据和存储对象的总数给出；

* 更新频率：参数包括平均版本区间(生命期)、每个对象的平均版本数，很长期对象的百分率，等等；

* 应用的清除、存档特性；

* 应用是否要求向后更新；

* 查询特征：包括大多数查询是否纯时态的、是否纯基于属性的，对于时态查询，是点查询还是区间查询，等等。我们在低层可用一个初始矩阵来刻画这些查询特征。

此外，还应考虑索引或存储结构对不同类型查询的空间消耗、更新时间、存储/迁移时间、存取时间。TDB的查询处理结构应该支持各层上的可扩展性。在代数层，应能使用不同的代数。执行算法库也应该是可扩展的。通过在该库增加算法的实现和描述，以及被优化器使用的代价估算公式，就能扩展新的查找技术。优化算法本身也可以改变，即可基于不同的方法，如动态规划或分枝定界等。这种可扩展性应是朝着开放结构的方向的。

作为短期任务，我们应该更详细地检验所提出的框架，进一步说明各系统模块及其相互关系。应该开发出一个内部表示用的、基于现有关系代数时态扩展的代数，在此基础上进行时态查询优化并开发一个执行算法库和索引方法库。作为长期任务，我们应该继续研究是否在时态和非时态数据库之间存在更大的差异，研究更独立于关系代数的时态代数，以及TDB对并发控制、恢复、安全和其它系统模块的影响。

总结 本文重点讨论了时态数据库的基本框架，给出了一些现已达成的一致意见，主要包括：

(1)TDB应用对时间要求范围之广，使得我们面临从时标到相对时间的时态联接、再到自成体系的时态推理等诸多方面的问题，不可能在一个模型中就提供所有支持。因此，TDB的基本框架必须是也只能是基于一组主要的并被充分理解的需求特征，使得大多数时态应用能从TDBMS中得到至少是部分的支持；

(2)术语的充分定义工作还应继续。由于时间随处可见，必然导致对同一概念在不同的活动领域产生不同的术语，也有可能是相同的术语描述不同的概念。

(3)概念性模型必须与表示模型分开，从而使得执行问题与语义完整性问题可分开考虑；

(4)基本结构必须能识别概念性与表示性模型的不同，必须具有存储模型和索引库的可扩展性，以保证可选择适合于应用的存储模型和时态索引而获得高效的执行。

目前已有工作包括一个包含87个术语及其定义的术语集，提出了一个用于评价时态查询语言用户友好性的"语言标准"，TSQL2和TSQL3的设计仍在继续。TSQL3的一个语言说明草案已被提出，等等。

参考文献

[1]I. Ahn and R. Snodgrass, Proc. SIGMOD Intl. Conf. Washington, DC, pp. 96-107, May 1986

[2]J. Clifford, Proc. Workshop on Logical Bases for Data Bases. Toulouse, France, December 1982

[3]A. Tansel and L. Garnett, Proc. ACM Intl. Conf. on Management of Data, pp. 284-293, May 1989

[4]S. Gadia and C. Yeung, Proc. ACM Intl. Conf. on Management of Data, pp. 251-259, Chicago, IL, June 1988

[5]O. Etzion, et al., Proc. 2nd Intl. Workshop on Technologies and Systems, Dallas, TX, December 1992

[6]N. Lorentzos et al., Proc. IWITDB, Arlington, TX, June 1993

[7]J. Clifford et al., Technical report STERN IS-93-8, New York University Stern School of Business, 1993

[8]C. Theodoulidis et al., Proc. 10th Conf. on the Entity Relationship Approach. San Mateo, CA. October 1991