

# 一种支持时态数据的实时数据模型

张晓芳 刘云生

(华中科技大学计算机科学与技术学院 武汉 430074)

**摘要** 实时数据库系统不但包括事务的定时限制,还包括数据的定时限制。本文讨论了实时数据库中数据的时间特性,给出了数据的绝对时态一致性和相对时态一致性的定义;针对时态数据的特征,提出了一种实时关系数据模型。

**关键词** 实时数据库,时态数据,实时数据模型

## A Real-time Data Model Supporting Temporal Data

ZHANG Xiao-Fang LIU Yun-Sheng

(Institute of Computer Science and Technology, Huazhong University of Science and Technology, Wuhan 430074)

**Abstract** Real-time database systems contain not only transaction timing constraints, but also data timing constraints. This paper discusses the temporal characteristics of data in real-time database, and offers a definition of absolute and relative temporal consistency. A policy is given by considering both the deadlines of transactions and the "data deadline" to schedule real-time transactions. A real-time relational data model and a real-time relational algebra based on the characteristics of temporal data are also put forward.

**Keywords** Real-time database, Temporal data, Real-time data model

## 1 引言

时间是自然界无所不在的客观属性,所有信息都具有相应的时态属性。随着数据库与信息技术的深入和发展,信息系统面临许多新的应用和新的需求,对时态信息处理的需求越来越迫切。在某些系统中,如数据通信、电力调度、电子商务交换与电子商务、证券与股票交易、交通控制、雷达跟踪、作战指挥自动化等信息的时态性还起着关键性作用<sup>[1]</sup>。

实时数据库系统(RTDBS)中事务和数据都可以具有显式的定时限制,系统的正确性不仅依赖于事务执行的逻辑结果,还依赖于逻辑结果产生的时间<sup>[2]</sup>。到目前为止,研究实时数据库的文献大多集中在研究实时事务的定时特性,而对数据的时间特性讨论得较少<sup>[3,4]</sup>。关于时态数据的建模问题,在研究时态数据库的文献中有较多的讨论,而实时数据库中鲜有专门讨论数据建模问题的<sup>[5~7]</sup>。大多数文献,尤其是关于实时事务处理的文献,都假定具有有变化颗粒的数据项的数据模型。但这种方法有局限性,因为它没有使用一般的及时间的语义知识,而这对系统满足事务截止时间是很有用的。一般RTDBS都使用传统的数据模型,还没有引入时间维,而即使是引入了时间维的“时态数据模型”与“时态查询语言”也没有提供事务定时限制的说明机制。

本文主要讨论实时数据库中数据的时间特性以及时态数据对事务调度策略带来的影响,并在此基础上提出一种实时数据模型。

## 2 数据库中数据的时间特性

### 2.1 时间的表示和推理

有两种基本的时间形式模型:时间点和时间区间。在时

间点模型中,一个时间段可表示为时间点的偏序集: $I = \langle P, <_I \rangle$ 。其中 $P = \{P_1, P_2, \dots, P_n\}$ 为时间点的有限集, $<_I$ 表示 $P$ 上的时序。在时间区间模型中,基本的时间单位为时间区间。此时,一个时间点可定义为两个相邻时间区间的交。

在本文中,我们认为时间是离散、线性的全序集,并用非负整数 $0, 1, \dots$ 来表示时间的值。有个特殊的时间量NOW,表示当前的时间值。 $[t_i, t_{i+n}]$ 表示一个由时间 $t_i$ 开始、到时间 $t_{i+n}$ 结束的时间区间。

关于时态数据的推理包括:处理不同时态数据间的相关性;推出数据值的有效期;处理实际或近似的 inconsistency,识别不正确数据;处理不完全的时态数据。在支持时态数据的RTDBS中,这些功能应由RTDBS来实现。

### 2.2 数据的时间特性

在RTDBS中,数据随外部环境状态的变化而快速变化,其值只在一定的时间内是“有效”的,过时则不能反映外部环境的真实状态。此外,许多应用除了需要了解某些数据对象的当前值之外,常常还需要了解数据对象在以前某个特定时间的值。由此可知,RTDBS为了维护数据库中数据的时间特性,需做到:1)保证系统在需要了解外部环境当前真实状态时能得到足够“新鲜”的数据,系统除了维护数据库内部状态(数据值)的正确性、相容性外,还必须同时维护内部状态与外部环境实际状态的一致性,以及数据用来决策或推导新数据时在时间上的相互一致性。2)系统能够保留过去的历史数据。

在RTDBS中,现实世界的对象由传感器监视,其值周期性地被采样并写入数据库中,这类对象被称为外部对象(external objects)。数据库中的数据对象是系统对外部对象的描述,称为内部对象(internal objects)。在内部数据对象中,我们将数据分为时态数据对象(temporal data objects)和非时

张晓芳 讲师,博士研究生,主要研究方向为现代数据库技术;刘云生

教授,博士生导师,主要研究领域为数据库与信息系统、高级应用的数据

库(主动、实时、时序)技术。

态数据对象(non-temporal data objects)。顾名思义,时态数据对象是其值随时间变化而变化的数据对象,而非时态数据对象是其值不随时间变化的数据对象。时态数据对象中的一类数据,其值为一个外部对象在特定时刻的一个映像,称之为映像对象(image objects),它是由传感器监视,其值被周期性地采样,并写入数据库。另一类时态数据对象为导出对象(derived objects),其值是由一组时态数据对象/非时态数据对象计算而得。

**定义 1** 一个 RTDBS 的时态数据对象为一个二元组  $\langle v, evi \rangle$ 。其中分量  $v$  为数据值;分量  $evi$  为数据的有效期,其形式为时间区间  $[evi_b, evi_e]$ 。数据对象的有效期长度为  $evi_e - evi_b$ 。

时态数据的有效期指明了数据对象的值的有效时间。在此时间范围之外,此值是无效的。对于映像对象而言, $evi_b$  的取值为数据的采样时间,也称为时标。对于导出对象而言, $evi_b$  和  $evi_e$  的取值由导出此数据的数据集的  $evi_b$  和  $evi_e$  综合确定。

### 3 时态一致性

数据库中的事务在读取时态数据时,通常要求所得到的数据值要能够反映现实世界的当前状态,即数据值不能过“老”。这种一致性要求被称做绝对时态一致性。事务的另一种可能的要求是一组时态数据的时标要相对接近,以保证这组数据的导出数据的一致。这种一致性要求被称为相对时态一致性。

**定义 2** 事务  $t$  在读取时态数据对象和提交时,所读时态数据都在其有效期范围之内,即

$$\forall x \in t.in; T_x \in x, evi, T_{commit} \in x, evi$$

则称事务  $t$  是绝对时态一致的。其中  $t.in$  为事务  $t$  的读数据集,  $T_x$  为  $t$  读数据  $x$  的时间,  $T_{commit}$  为  $t$  提交的时间。

**定义 3** 对于用来导出一个视图或数据的数据集  $R$ , 当  $R$  满足下面的条件时,被称作相互一致集:

$$\forall x, y \in R, |x, evi_b - y, evi_b| \leq R_{rel}$$

即  $R$  中的任意两个数据的时标之差不大于  $R_{rel}$ 。  $R_{rel}$  称为  $R$  的相对有效期,它的值是由应用决定的。

**定义 4** 对于事务  $t$  所读取的每个导出数据对象,用于计算出该数据对象的数据集都为相互一致集,则称事务  $t$  是相对时态一致的。

由于实时数据库的数据具有时态性,这种数据的值只在一定范围内是有效的,这就要求实时事务处理考虑数据的定时限制。一个 RTDBS 不但要维护数据和事务的逻辑一致性,还要维护事务的定时限制和数据时态一致性。根据事务正确提交的条件,事务完成时数据应是时态一致的<sup>[8]</sup>。如何在事务调度过程中满足数据的时态一致性要求,在较多文献中都有讨论<sup>[8~10]</sup>。较为普遍的做法是根据事务自身的截止期和“数据截止期”来确定事务的截止期。

**定义 5** 一个事务  $t$  结束时刻  $t_p$  访问过的所有时态数据对象的有效期终点的最小者为事务  $t$  在时刻  $t_p$  的“数据截止期”,记为  $dd_p(t)$ , 即  $dd_p(t) = \min(evi_e(x)), x \in t.in$ 。

**定义 6** 事务  $t$  的截止期  $d(t) = \min(d', dd_p(t))$ 。其中  $d'$  为事务自身的截止期,  $dd_p(t)$  为事务的数据截止期。

### 4 实时数据模型

在传统的关系数据模型基础之上,我们提出一种新的实

时关系数据模型。为了直观地表达时态数据的二维(值,有效期)性,该数据模型采用非第一范式( $\rightarrow 1NF$ )的方法。在此模型中,属性可分为时态属性和非时态属性。时态属性不但包含有数据值,还包含有数据值的有效期。另外,为了表达时态属性随着时间变化而不断变化的特性,一个元组中的时态属性允许有多个值,但这些值的有效期应不相交。

**定义 7** 一个实时关系模式  $R$  是一个四元组  $R = \langle A, D, ALS, dom \rangle$ 。其中:

(1)  $A = NTA \cup TA \cup EVI$ , 是  $R$  的属性集。  $NTA = \{NTA_1, NTA_2, \dots, NTA_k\}$  是  $R$  的非时态属性集;  $TA = \{TA_1, TA_2, \dots, TA_l\}$  是  $R$  的时态属性集;  $EVI = \{evi\}$  为隐式的元组有效期属性。

(2)  $D$  为属性集  $A$  中属性所来自的域。

(3)  $ALS: R(A) \rightarrow T$  是属性集  $A$  向时间域的映射函数集,即时态属性的有效期的取值范围。

(4)  $dom: A \times ALS \rightarrow D$  是从属性集  $A$ 、有效期  $ALS$  到值域  $D$  的映射函数集。

**定义 8** 实时关系模式  $R$  上的一个关系实例  $r$  为有限个元组的集合:

$$r = \{x | x = (a_1, a_2, \dots, a_n | evi)\}$$

其中: $evi$  是元组  $x$  的有效期,  $evi \in ALS$ ;  $a_1, a_2, \dots, a_n$  是元组  $x$  的各属性值,它是一个由属性集、属性的有效期到属性值域的映射函数。

**定义 9** 实时关系模式  $R$  中的时态属性为时态数据对象(见定义 1),由值和有效期两个分量组成。一个元组中的时态属性允许有多个值,但这些值的有效期应不相交。一个元组中该属性的有效期为该属性各个值的有效期的并。设元组  $x$  的时态属性  $ta_i$  由  $w$  个值组成,则

$$x, ta_i, evi = x, ta_i [1], evi \cup x, ta_i [2], evi \cup \dots \cup x, ta_i [w], evi$$

**定义 10** 实时数据模型中一个元组的各个时态属性的有效期应一致,时态属性的有效期也就是元组的有效期,即

$$x, evi = ta_1, evi = ta_2, evi = \dots = ta_l, evi$$

下面举一个实时数据模型的例子。有一空中交通控制系统,需定时对管理范围内飞机的位置、速度等状态进行采样。如表 1 所示, AIR-TRAFIC 是采用上述实时数据模型的关系表。 PNO 为飞机号, 它为非时态属性; 速度和高度为时态属性, 其值由传感器定时采样写入数据库; 元组有效期为关系的隐式属性。

表 1 关系表 AIR-TRAFIC

PNO	速度	高度	有效期
001	1200 [100,102]	2000 [100,101]	[100,105]
	1500 [103,105]	1989 [102,103]	
		2002 [103,104]	
		2000 [104,105]	
002	1800 [110,113]	2400 [110,113]	[110,116]
	1760 [114,116]	2500 [114,116]	

### 5 实时关系代数

基于实时数据模型的关系代数是传统关系代数的扩展。当各元组的有效期的都为  $[NOW, NOW]$  时,则转化为传统数据模型。传统关系代数由集合运算和专门的关系运算组成,实时关系代数同样需要进行这些运算。在以下的讨论中,

我们用  $R$  和  $S$  表示关系模式,  $r, s$  表示关系模式的实例;  $x, y, z$  用来表示元组;  $A_i, B_i, C_i$  用来表示属性, 并用  $A$  来表示属性  $A_i$  的集合; 符号  $x[A_i]$  表示元组  $x$  的属性  $A_i$  的值,  $x[evi]$  表示元组的有效期属性。

### 5.1 并运算

设  $r_1$  和  $r_2$  都为模式  $R$  的实例, 其并操作  $\cup^T$  可表示为:  
 $r_1 \cup^T r_2 = \{z[A|evi] | (\exists x \in r_1, y \in r_2 (z[A] = x[A] = y[A] \wedge z[evi] = x[evi] \cup y[evi])) \vee (\exists x \in r_1 (z[A] = x[A] \wedge (\neg \exists y \in r_2 (y[A] = x[A])) \wedge z[evi] = x[evi])) \vee (\exists y \in r_2 (z[A] = y[A] \wedge (\neg \exists x \in r_1 (x[A] = y[A] = y[A])) \wedge z[evi] = y[evi]))\}$

### 5.2 差运算

$r_1$  和  $r_2$  的差操作  $-^T$  可表示为:  
 $r_1 -^T r_2 = \{z[A|evi] | \exists x \in r_1 (z[A] = x[A]) \wedge ((\exists y \in r_2 (z[A] = y[A] \wedge z[evi] = x[evi] - y[evi])) \vee (\neg \exists y \in r_2 (z[A] = y[A] \wedge z[evi] = x[evi])))\}$

### 5.3 笛卡尔积运算

设  $r$  和  $s$  分别为实时关系模式  $R$  和  $S$  的一个实例,  $R$  和  $S$  的定义如下:

$$R = (A_1, A_2, \dots, A_n | evi)$$

$$S = (B_1, B_2, \dots, B_m | evi)$$

则  $r$  和  $s$  的笛卡尔积  $\times^T$  可表示为:

$$r \times^T s = \{z[A, B|evi] = \{\exists x \in r, \exists y \in s (z[evi] = x[evi] \cap y[evi] \wedge \forall A_i \in A, \forall zevi \in z[evi] (z[A_i, zevi] = x[(A_i, zevi)]) \wedge \forall B_j \in B, \forall zevi \in z[evi] (z[B_j, zevi] = y[B_j, zevi]))\} \vee (z[evi] = x[evi] - y[evi] \wedge \forall A_i \in A, \forall zevi \in z[evi] (z[A_i, zevi] = x[(A_i, zevi)]) \wedge \forall B_j \in B, \forall zevi \in z[evi] (z[(B_j, zevi)] = \emptyset)) \vee (z[evi] = y[evi] - x[evi] \wedge \forall A_i \in A, \forall zevi \in z[evi] (z[(A_i, zevi)] = \emptyset) \wedge \forall B_j \in B, \forall zevi \in z[evi] (z[(B_j, zevi)] = y[(B_j, zevi)]))\}$$

$\emptyset$  表示空值。关系  $r$  和  $s$  在进行笛卡尔积运算时, 其中的每个属性的有效期在新的关系中保持不变。

### 5.4 选择运算

设  $r(a_1, a_2, \dots, a_n | evi)$  为实时关系模式  $R$  的一个实例,  $P$  为限定条件的时态布尔表达式, 则对  $r$  的选择运算  $\sigma^T$  可表示为:

$$a_P^T(r) = \{z[A|evi] | z \in r \wedge P(z[A|evi])\}$$

时态布尔表达式  $P$  不但包括传统的运算符, 还包括对时间区间的运算。

### 5.5 投影运算

实时关系代数中的投影运算与传统关系代数中的投影运算类似, 且新生成的元组的有效期与源元组的有效期相同。设  $D$  为  $r$  的属性集  $A$  的子集, 则  $r$  在  $D$  上的投影  $\pi^T$  可表示为:

$$\pi_D^T(r) = \{z[D|evi] | \exists x \in r (Z[D] = x[D] \wedge z[evi] = x[evi])\}$$

### 5.6 连接运算

设  $r$  和  $s$  分别为实时关系模式  $R$  和  $S$  的一个实例,  $R$  和  $S$  的定义如下:

$$R = (A_1, \dots, A_n, B_1, \dots, B_l | evi)$$

$$S = (A_1, \dots, A_n, C_1, \dots, C_m | evi)$$

则  $r$  和  $s$  的自然连接操作  $\infty^T$  可表示为:

$$r \infty^T s = \{z[A, B, C|evi] | \exists x \in r, \exists y \in s (x[A] = y[A] \wedge x[evi] \cap y[evi] \neq \emptyset \wedge z[A] = x[A] \wedge z[B] = x[B] \wedge z[C] = y[C] \wedge z[evi] = x[evi] \cap y[evi])\}$$

由公式可知, 连接所产生的目的元组的有效期为两个源元组的有效期的交。

**结束语** 基于实时数据库中数据的时间特性, 本文提出了一种支持时态数据的实时数据模型。在此模型中, 时态数据不但有相应的数据值, 还有与数据值相对应的有效期。同时, 该数据模型也可对历史数据值进行保留。RTDBS 除了要维护传统数据库的内部状态一致性, 还必须考虑外部世界与内部状态的一致性, 同时必须考虑不同数据间的相互时间一致性。本文给出了时态数据的绝对时态一致性和相对时态一致性的形式化定义。

实时数据库系统不但要满足事务的定时限制要求, 也要满足数据的时间特性要求。如何解决大量历史数据需占用较大的存储空间与实时数据库中事务和数据的定时限制需较快的时间之间的矛盾, 如何处理非 1NF 数据结构对查询处理和存储管理带来的影响、事务调度策略如何适应数据的时间特性等问题都值得我们做深入的研究。

## 参考文献

- 1 刘云生. 现代数据库技术[M]. 北京: 国防工业出版社, 2001
- 2 刘云生. 实时数据库系统[J]. 计算机科学, 1994, 21(3): 42~46
- 3 Song Xiaohui, Liu Jane W S. Maintaining Temporal Consistency: Pessimistic vs. Optimistic Concurrency Control [J]. IEEE Transactions on Knowledge and Data Engineering, 1995, 7(5): 786~796
- 4 Kao Ben, Lam Ksam-Yin. Maintaining Temporal Consistency of Discrete Objects in Soft Real-Time Database Systems [J]. IEEE Transactions on Computers, 2003, 52(3): 373~389
- 5 Ozsoyoglu G, Snodgrass R T. Temporal and Real-Time Databases: A Survey [J]. IEEE Transactions on Knowledge and Data Engineering, 1995, 7(4): 513~532
- 6 Uz Tansel A. Temporal Relational Data Model [J]. IEEE Transactions on Knowledge and Data Engineering, 1997, 9(3): 464~479
- 7 Uz Tansel A, Tin E. The Expressive Power of Temporal Relational Query Languages [J]. IEEE Transactions on Knowledge and Data Engineering, 1997, 9(1): 120~134
- 8 刘云生, 李国徽. 实时数据库数据特征对事务处理的影响[J]. 计算机研究与发展, 1999, 36(3): 364~368
- 9 Xiong Ming, Ramamritham K. Scheduling transactions with temporal constraints: exploiting data semantics [J]. IEEE Transactions on Knowledge and Data Engineering, 2002, 14(5): 1155~1166
- 10 Kim Young-Kuk, Son S H. Supporting Predictability in Real-Time Database Systems [J]. Real-Time Technology and Applications Symposium Proceedings, 1996. 38~48
- 11 Jensen C S, Snodgrass R T. Unification of Temporal Data Models [J]. In: Data Engineering Proceedings, Ninth International Conference, 1993. 262~271