

时态数据完整性约束研究与实现

刘 海^{1,2} 汤 庸^{1,2} 郭 欢¹ 叶小平²

(中山大学信息科学与技术学院 广州 510275)¹ (华南师范大学计算机学院 广州 510631)²

摘 要 时态数据库的完整性是为了保证时态数据库中存储的时态数据的正确性。为了防止时态数据库中出现不符合时态模型语义的数据,需要研究时态数据库的时态完整性相关理论,为时态更新操作的正确执行提供理论支持。在时态关系数据模型的基础上,通过将传统的数据完整性约束机制扩展到时态数据库领域,给出了时态完整性约束的完整定义和在不同的情况下违反时态完整性约束的具体处理机制,为时态数据处理构件——TempDB 实现时态数据完整性约束提供了理论基础和实现依据,这对时态数据库理论的进一步完善和相关的实现技术具有重要的参考价值。

关键词 时态数据库,时态完整性,时态实体完整性,时态参照完整性

中图法分类号 TP311 **文献标识码** A

Research and Realization of Temporal Data Integrity Constraints

LIU Hai^{1,2} TANG Yong^{1,2} GUO Huan¹ YE Xiao-ping²

(School of Information Science and Technology, Sun Yat-Sen University, Guangzhou 510275, China)¹

(School of Computer Science, South China Normal University, Guangzhou 510631, China)²

Abstract Temporal integrity constraints are used to ensure the correctness of the data in temporal database. To avoid the appearance of inconsistent data with temporal semantics model, relevant theories of temporal integrity should be studied, which also provide theoretical support for implementation of temporal updating operations. Based on temporal relational model (TRM), this paper gave an integral definition of temporal integrity constraints in the temporal extensions to traditional integrity constraints, and studied the concrete processing mechanism in detail when some manipulating operations violate temporal integrity constraints. Some of the proposed processing mechanisms for temporal integrity constraints were realized in TempDB. This study not only provides the theory and implementation basis for temporal data integrity constraints in TempDB, but also has important reference value for the further perfection of the theories and relevant implementation techniques of temporal database.

Keywords Temporal database, Temporal integrity, Temporal entity integrity, Temporal referential integrity

1 引言

时间是客观事物最基本属性之一,时态数据库是数据库系统中的基本研究开发领域。虽然时态数据库技术的研究已经有多年历史,但随着各种数据库新技术的兴起与发展(例如时空数据库^[1]、移动对象数据库^[2]、传感器数据库^[3]和 XML 数据库等主流技术^[4]),对时态数据管理提出了众多新的挑战性和更为深入强大的应用驱动。由于上述数据库系统更注重对数据的动态管理,因此,基于时间因素的数据更新问题更为突出和迫切。据我们所掌握的资料,有关时态数据库的研究主要集中在时态数据模型^[5]、时态查询语言^[6,7]和时态存取方法^[8]等方面,关于数据的时态更新的工作并不多见。数据更新的前提是保证数据的完整性约束,时态数据更新也需要满足相应时态完整性约束,否则,将会造成大量的垃圾数

据和违反现实建模语义的数据,导致数据系统的不可使用。因此,时态完整性近年来成为学术界关注的基本课题^[9-13],其主要研究工作可归纳为 4 个方面:①时态完整性包含数据和数据对应的时间;②时态完整性包括时态参照完整性以及时态实体完整性;③时态完整性包括静态的完整性和动态的完整性,也称为静态完整性和动态完整性;④传统数据库的完整性是静态的完整性,而时态数据库的完整性是动态的完整性。但上述工作实际上没有给出时态完整性的确切定义,更为重要的是没有具体讨论在时态更新时关于时态完整性约束的处理实现机制。

TempDB^[14]是国内第一个时态数据处理构件,是基于现有非时态商业数据库管理系统(如 My SQL)的前端应用软件,它能够在非时态性的商业关系数据库上支持时态功能,通过将时态语言转换成符合时态语义的后台数据库支持的标准

到稿日期:2009-12-21 返修日期:2010-02-20 本文受国家自然科学基金(60673135, 60970044),广东省自然科学基金(7003721, 05003348, 9151027501000054)资助。

刘 海(1974-),男,博士生,主要研究方向为时态数据库与协同软件, E-mail: liuhai@senu.edu.cn; 汤 庸(1964-),男,博士,教授,博士生导师,主要研究方向为数据库、知识工程与协同软件; 郭 欢(1984-),女,博士生,主要研究方向为时态数据库与协同软件; 叶小平(1955-),男,教授,主要研究方向为数据库、时态信息处理、知识发现。

SQL 语言来实现相关时态功能。本文重点研究 TempDB 在时态数据更新过程中的时态完整性约束处理机制,首先给出时态完整性的确切定义,同时从原理和技术层面研究时态完整性约束处理机制,为时态数据处理构件 TempDB 提供理论基础与实现依据。本文第 2 节给出时态完整性确切定义;第 3 节研究在时态更新时违反时态完整性约束时的处理机制;第 4 节对时态完整性约束处理机制在 TempDB 的实现进行说明并进行下一步工作的展望。

2 时态完整性

本文讨论的时态完整性约束是基于时态关系数据模型的,有效时间表示的是元组时间,各属性值的有效时间为其所在元组的有效时间,即同一元组的属性值具有相同的有效时间,在不引起混淆的情况下元组的有效时间和属性的有效时间可以混用。元组的有效时间称为元组的生命周期(Tuple's Life cycle),记为 $LC(t)$;对象的有效时间称为对象的生命周期(Object's life cycle),记为 $LC(o)$ 。对象在时态数据库中是由表示该对象的一个或多个元组记录的,所以元组的 $LC(t)$ 与对象的 $LC(o)$ 满足 $LC(t) \subseteq LC(o)$ 。

时态数据库的完整性是为了保证时态数据库中存储的数据是正确的,所谓正确是指符合现实世界的语义。与传统关系数据库的完整性约束一样,时态数据库的完整性约束可以分成 3 类:时态实体完整性、时态参照完整性和用户定义的完整性。其中时态实体完整性和时态参照完整性是时态数据库必须支持的完整性约束。时态完整性由两部分组成——数据和时间,其中时间指定了相应数据值的有效时间。本文着重讨论了时态实体完整性和时态参照完整性。为了简明清晰地定义时态实体完整性和时态参照完整性,首先在 Allen 定义的 13 种时间区间关系^[15]的基础上引入如下两个概念。

定义 1(属于关系) 给定两个时间区间 t_1 和 t_2 ,如果 $During(t_1, t_2)$, $Starts(t_1, t_2)$, $Finishes(t_1, t_2)$ 和 $Equals(t_1, t_2)$ 关系中的任一个成立,则称 t_1 属于 t_2 ,记为 $Belonging(t_1, t_2)$,否则,称 t_1 不属于 t_2 ,记为 $Unbelonging(t_1, t_2)$ 。

定义 2(不相关关系) 给定两个时间区间 t_1 和 t_2 ,如果 $Before(t_1, t_2)$, $After(t_1, t_2)$, $Meets(t_1, t_2)$ 和 $MetBy(t_1, t_2)$ 关系中的任一个成立,则称 t_1 与 t_2 不相关,记为 $Uncorrelated(t_1, t_2)$,否则,称 t_1 与 t_2 相关,记为 $Correlated(t_1, t_2)$ 。

2.1 时态实体完整性

数据库中的关系表表示一个对象集,对象之间通过其唯一的标识相互区分。在关系数据模型中,基本关系的唯一标识就是该关系的主键。例如,学生的学号在整个关系数据表内是唯一的。但是在时态数据库或时态关系模型中,标识的唯一性是指,该标识在某个时间区间内是唯一的。例如,在时态数据库中(‘S001’, [2005-01, 2006-08])和(‘S001’, [2007-02, 2009-05])可以同时存在于一个数据库表中。但是(‘S001’, [2005-01, 2006-08])和(‘S002’, [2004-02, 2005-10])不能同时存在一个基本表中,因为在时间区间[2005-01, 2005-10]内学号有两个不同的赋值。

定义 3(时态关系模型中的主键) 在时态关系数据模型中,时态关系 S 的主键定义为: (PK_1, \dots, PK_n, PT) ,其中 PK_i 为时态关系 S 的一个主键属性, PT 为主键属性 (PK_1, \dots, PK_n) 的有效时间。当 $i=1$ 时,表示 S 中只有一个主键属性。

定义 4(时态实体完整性规则) 若属性 A 是时态关系 S 的一个主键,则属性 A 的取值满足如下要求:

- (1) A 不能取空值;
- (2) A 的取值在其有效时间范围内唯一。

其中,属性 A 和其有效时间称为时态关系 S 的一个联合主键,记为 $CPK(A)$,即 $CPK(A) = (PK(A), PT(A))$,且 $CPK(A)$ 在时态关系 S 中具有唯一取值。

例 1 给出学生、课程和选课 3 个时态关系表,如表 1—表 3 所列,其中学号、课程号、学号和课程号分别为 3 个表的主键属性。

表 1 学生时态关系

学号	姓名	有效时间
S001	杨海	[2002-09, 2005-07]
S003	张洋	[2001-09, 2003-07]
S001	汪涛	[2006-09, 2008-07]

表 2 课程时态关系

课程号	课程名	有效时间
C001	操作系统	[2007-09, 2009-07]
C002	组成原理	[2002-09, 2004-07]
C002	数据结构	[2004-09, 2005-07]

表 3 选课时态关系

学号	课程名	成绩	有效时间
S001	C001	85	[2007-09, 2008-07]
S003	C002	90	[2002-09, 2003-07]

由定义 3 可知,3 个表的主键分别为(学号,有效时间)、(课程号,有效时间)和(学号,课程号,有效时间)。虽然表 1 中学号“S001”出现在两个不同的元组中,但两个元组的有效时间不相关,即学号“S001”在其有效时间范围内唯一。表 2 中的课程号“C002”也满足同样的约束。所以这 3 个时态关系表均满足时态实体完整性要求。

2.2 时态参照完整性

现实世界中的实体之间会存在一定的联系,在关系模型中,实体之间的联系通过关系之间的相互“引用”实现。而在时态关系模型中,则需要研究实体之间的“时态引用”。

定义 5 设 A 是时态关系 R 的一个或者一组属性,但不是时态关系 R 的主键。如果 A 与时态关系 S 的主键 PK 相对应,则称 A 为外键属性,记为 FK (Foreign Key Attribute)。 A 对应的时间区间称为外键时间区间,记为 FT (Foreign Key Time Interval)。并称时态关系 R 为时态参照关系(Temporal Referencing Relation),时态关系 S 为时态被参照关系(Temporal Referenced Relation)。

由定义 5 可以看出,时态关系数据模型中,时态参照关系和时态被参照关系之间不仅存在属性值的引用,还存在该属性值对应的时间区间的引用。因此定义时态关系数据模型中的外键如下:

定义 6(时态关系模型中的外键) 在时态关系数据模型中,参照关系 R 中的外键定义为: (FK_1, \dots, FK_n, FT) ,其中 $FK_i (1 \leq i \leq n)$ 是参照关系 R 中的一个外键属性, FT 为外键属性 (FK_1, \dots, FK_n) 的时间区间。当 $i=1$ 时,表示 R 中只有一个外键属性。

定义 7(时态参照完整性规则) 若 (FK_1, \dots, FK_n, FT) 是时态关系 R 的外键, FK_i 与时态关系 S 的主键 PK 相对应,则对于 R 中每个元组对应的 $FK_i (1 \leq i \leq n)$ 必须满足下列条件:

- (1) FK_i 的取值或者取空值,或者等于 S 中某元组的主键

值;

(2) 如果 FK_i 不取空值, 则 FT 必须属于 PK 的有效时间, 即 $Belonging(FT, PT)$ 。

同时称外键属性 FK_i 存在对时态关系 S 的主键 PK 的时态引用。

由定义 7 可以看出, 在时态关系数据模型中, 为了正确地反映出现实世界建模的语义, 必须考虑时态参照表与时态被参照表之间的时态引用关系。

例 2 学生(S)、课程(C)和选课(CS)3 个时态关系表当前存储的数据即表 1—表 3, 其中选课表为时态参照关系, 而学生和课程表则为相应的时态被参照表。选课关系中的第一个元组 {'S001', 'C001', 85, [2007-09, 2008-07]} 中, 学号 'S001' 是学生关系表中真正存在的学号, 且与元组 {'S001', '汪涛', [2006-09, 2008-07]} 的有效时间之间满足 $Belonging([2007-09, 2008-07], [2006-09, 2008-07])$; 同时课程号 'C001' 与课程时态关系之间也满足时态参照完整性约束。注意, 选课时态关系元组的生命周期 $LC([学号, 课程号])$ 与学生和课程时态关系元组的生命周期 $LC(学号)$ 、 $LC(课程号)$ 满足: $LC([学号, 课程号]) \subseteq LC(学号) \cap LC(课程号)$ 。

3 时态完整性处理机制

在时态数据库中, 时态实体完整性和时态参照完整性是最重要的完整性约束, 其他的完整性约束都可以归结为用户定义完整性。在实际应用中, 主要是对时态实体完整性和时态参照完整性进行控制。对于违反时态实体完整性规则 and 用户定义完整性规则的操作, 一般都采用拒绝执行的方式进行处理; 对于违反时态参照完整性的操作, 并不是简单地拒绝执行, 为保证数据的正确性, 有时会在执行必要的附加操作之后接受该操作。本节将从保持时态参照完整性方面考虑时态插入、删除和修改的处理机制。

3.1 时态插入处理机制

对于时态插入操作, 只有时态参照表中新元组的插入可能会违反完整性约束机制。因此, 下面讨论的完整性处理机制都是在时态参照表中插入元组时所需要考虑的情况。

(1) 时态受限插入

向时态参照关系表 R 插入元组 T 时, 如果 T 的外键值 $FK_i (1 \leq i \leq n)$ 不全为空值, 当且仅当时态被参照关系 S 中存在相应的元组, 其主键值 PK 与 FK_i 相同且满足 $Belonging(PK, PT)$, 系统才执行插入操作, 否则拒绝执行此操作。

(2) 时态递归插入

向时态参照关系表 R 插入元组 T 时, 如果 T 的外键值 $FK_i (1 \leq i \leq n)$ 或有效时间 FT 不满足时态参照完整性约束, 则向其对应的时态被参照表中插入相应的元组, 其主键值等于 FK_i , 有效时间等于 FT , 直到待插入元组 T 的每个外键值 $FK_i (1 \leq i \leq n)$ 和 FT 都满足时态参照完整性约束, 再向参照关系中插入元组 T 。

例 3 给定 3 个时态数据关系表, 如表 1—表 3 所列, 向选课时态关系表中插入下面两个元组: $T_1 ('S001', 'C002', 90, [2002-09, 2004-07])$; $T_2 ('S002', 'C001', 88, [2004-09, 2005-07])$ 。

元组 T_1 满足时态参照完整性约束, 可以直接插入选课时态关系中; 元组 T_2 不满足时态参照完整性, 其中, 学生时

态关系中不存在学号为 'S002' 的元组, 课程时态关系中学号为 'C001' 的元组的有效时间与待插入元组 T_2 的有效时间不相关。在时态受限插入处理机制下, 元组 T_2 被拒绝插入; 在时态递归插入处理机制下, 需要往学生时态关系表中插入元组 {'S002', ' ', [2004-09, 2005-07]}, 在课程时态关系表中插入元组 {'C001', ' ', [2004-09, 2005-07]}, 然后才可以插入元组 T_2 。

在时态递归插入处理机制下, 插入元组 T_1 和 T_2 后 3 个时态关系表存储的数据如表 4—表 6 所列。

表 4 时态递归插入后的学生时态关系表

学号	姓名	有效时间
S001	杨海	[2002-09, 2005-07]
S003	张洋	[2001-09, 2003-07]
S001	汪涛	[2006-09, 2008-07]
S002		[2004-09, 2005-07]

表 5 时态递归插入后的课程时态关系表

课程号	课程名	有效时间
C001	操作系统	[2007-09, 2009-07]
C002	组成原理	[2002-09, 2004-07]
C002	数据结构	[2004-09, 2005-07]
C001		[2004-09, 2005-07]

表 6 时态递归插入后的选课时态关系表

学号	课程名	成绩	有效时间
S001	C001	85	[2007-09, 2008-07]
S003	C002	90	[2002-09, 2003-07]
S001	C002	90	[2002-09, 2004-07]
S002	C001	88	[2004-09, 2005-07]

3.2 时态删除处理机制

在时态关系数据模型中, 时态参照表中元组的删除是不会违反完整性约束的。因此, 下面讨论的完整性处理机制都是在删除时态被参照表中的元组时需要考虑的情况。

(1) 时态受限删除

当且仅当相应的时态参照关系的外键值不存在对待删除元组 T 的主键值的时态引用时, 系统才执行删除操作, 否则拒绝执行此操作。

也就是说, 当时态参照关系中没有任何元组的外键值 FK_i 与时态被参照关系中要删除元组 T 的主键值相同; 或者存在这样的主键值, 但待删元组的时间区间与时态参照关系表中外键值的时间区间不相关。

(2) 时态置空值删除

删除时态被参照关系表中指定的元组, 及将时态参照关系中相应元组的外键值在被删除元组的时间区间内设置为空值。

(3) 时态级联删除

删除时态被参照关系表中指定的元组, 及将时态参照关系中所有外键值与删除元组的主键值相同的元组, 并属于被删元组的时间区间中的值全部删除。如果时态参照关系同时又是另一个关系的时态被参照关系, 则这种删除操作会一直继续级联下去。

例 4 给定 3 个时态数据关系表, 表 1—表 3, 在学生时态关系表中删除元组: $T_1 ('S001', '汪涛', [2006-09, 2007-11])$; 在课程时态关系表中删除元组 $T_2 ('C002', '组成原理', [2002-09, 2003-01])$ 。按照 T_1, T_2 的删除顺序:

在时态置空值删除处理机制下, 删除操作后的时态关系

表如表 7—表 9 所列。在时态级联删除处理机制下,删除操作后的学生和课程时态表同表 7—表 8,选课时态关系表如表 10 所列。如果选课时态关系表还是数据库中某个时态关系 R 的时态被参照表,则时态关系 R 也需要同样的级联删除操作。

表 7 时态置空值删除后的学生时态关系表

学号	姓名	有效时间
S001	杨海	[2002-09, 2005-07]
S003	张洋	[2001-09, 2003-07]
S001	汪涛	[2007-11, 2008-07]

表 8 时态置空值删除后的课程关系表

课程号	课程名	有效时间
C001	操作系统	[2007-09, 2009-07]
C002	数据结构	[2004-09, 2005-07]
C002	组成原理	[2003-01, 2003-07]

表 9 时态置空值删除后的选课时态关系表

学号	课程名	成绩	有效时间
S001	C001	85	[2007-12, 2008-07]
S003	C002	90	[2003-02, 2003-07]
	C001	85	[2007-09, 2007-11]
S003		90	[2002-09, 2003-01]

表 10 时态级联删除后的选课时态关系表

学号	课程名	成绩	有效时间
S001	C001	85	[2007-12, 2008-07]
S003	C002	90	[2003-02, 2003-07]

3.3 时态修改处理机制

时态修改处理机制最基本的目的就是控制时态修改操作,使时态参照表的外键值和其对应的时间区间不能被修改为被时态参照表中没有出现的值。可以从两个角度来考虑该问题:时态参照表被修改和时态被参照表被修改。

(I) 修改时态被参照关系表

分两种情况考虑:(1)不允许时态关系表中的主码被修改。当修改被参照关系中的元组时,不会违反时态参照完整性约束的语义;(2)允许修改时态关系的主码。当修改时态被参照关系中的元组时,必须检查时态参照关系中是否存在对待删除元组有时态引用的元组。下面讨论的 3 个修改规则的前提条件都是:数据库中允许修改时态参照关系的主码值。

(1) 时态拒绝修改 当且仅当时态参照关系中所有的元组都不存在对待修改元组的时态引用时,系统才执行修改操作,否则拒绝执行此操作。

(2) 时态置空值修改 修改时态被参照关系的元组 T ,并将时态参照关系中存在对 T 时态引用的元组的外键值在修改后的元组 T 的时间区间内置空值。

(3) 时态级联修改 修改时态被参照关系的元组 T ,并将时态参照关系中存在对元组 T 时态引用的所有元组的外键值,及属于 T 修改后的时间区间内的值修改为 T 修改后的新值。如果时态参照关系同时又是另一个关系的时态被参照关系,则这种修改操作会一直继续级联下去。

也就是说,当主码的值被修改时,所有对修改之前主码值存在时态引用的时态被参照表都会被修改。

例 5 给定 3 个时态数据关系表,即表 1—表 3,现分别执行以下 2 个语句:

S_1 : UPDATE 选课 SET 学号='S002',姓名='孙文'

WHERE 学号='S001' VALID PERIOD '[2006-09, 2007-11]'

S_2 : UPDATE 课程 SET 课程号='C005' 课程名='组成原理' WHERE 课程号='C002' VALID PERIOD '[2002-09, 2003-01]'

在时态拒绝修改处理机制下, S_1 和 S_2 是不能够执行的。在时态置空值修改的处理机制下,修改过后的结果如表 11—表 13 所列。

表 11 时态置空值修改后的学生时态关系表

学号	姓名	有效时间
S001	杨海	[2002-09, 2005-07]
S003	张洋	[2001-09, 2003-07]
S001	汪涛	[2007-11, 2008-07]
S002	孙文	[2006-09, 2007-11]

表 12 时态置空值修改后的课程时态关系表

课程号	课程名	有效时间
C001	操作系统	[2007-09, 2009-07]
C002	组成原理	[2003-01, 2004-07]
C002	数据结构	[2004-09, 2005-07]
C005	算法设计	[2002-09, 2003-01]

表 13 时态置空值修改后的选课时态关系表

学号	课程名	成绩	有效时间
S001	C001	85	[2007-11, 2008-07]
S003	C002	90	[2003-01, 2003-07]
	C001	85	[2007-09, 2007-11]
S003		90	[2002-09, 2003-01]

在时态级联修改的处理机制下,两个被参照关系的内容和时态置空值修改的结果一样,因为这两个概念的不同之处只是对时态参照表的处理不同。选课时态关系(时态参照表)在时态级联修改后的内容如表 14 所列。

表 14 时态级联修改后的选课时态关系表

学号	课程名	成绩	有效时间
S001	C001	85	[2007-12, 2008-07]
S003	C002	90	[2003-01, 2003-07]
S002	C001	85	[2007-09, 2007-12]
S003	C005	90	[2002-09, 2003-01]

(II) 修改时态参照关系表

必须检查时态被参照关系中,是否存在这样的元组,其主键值在属于时态参照关系要修改的时间区间内是否等于要修改的外键值。其处理规则如下:

(1) 时态受限修改 当且仅当时态被参照关系中存在相应的元组,其主键值与时态参照关系中要修改元组的外键属性 FK 的新值相同,并且要修改元组的新的有效时间 FT 属于被参照关系中该主键值的有效时间,系统才执行修改操作,否则拒绝执行此修改操作。

具体地说,即要修改元组中指定外键属性的新值不存在于被时态参照表中;或者外键属性的新值存在于被时态参照表中,但是待修改元组的新的有效时间区间不属于该时态被参照表中相应主键值的有效时间。都不能对原有的元组进行修改。

(2) 时态递归插入修改 修改时态参照关系中某元组时,如果该元组的外键值(FK_1, \dots, FK_n, FT)中某个 FK_i 值或者 FT 值不满足时态参照完整性约束,则向对应的被时态参照表中插入如下元组:其主键值等于时态参照关系要修改

元组的新外键值,其有效时间区间等于时态参照关系要修改元组的新的有效时间区间。直到 (FK_1, \dots, FK_n, FT) 中的每个元素都满足时态参照完整性约束,再修改时态参照关系中元组的元组为新值。

例6 给定3个时态数据关系表,即表1—表3,执行下面的修改语句:

S: UPDATE 课程 SET 学号='S002',课程号='C005'
WHERE 学号='S001' 课程号='C001' VALID PERIOD
['2007-09,2008-01']

在时态拒绝修改处理机制下,S是不能够执行的。在时态递归插入修改的处理机制下,修改过后的结果如表15—表17所列。

表15 时态递归插入修改后的学生时态关系表

学号	姓名	有效时间
S001	杨海	[2002-09, 2005-07]
S003	张洋	[2001-09, 2003-07]
S001	汪涛	[2006-09, 2008-07]
S002		[2007-09, 2008-01]

表16 时态递归插入修改后的课程时态关系表

课程号	课程名	有效时间
C001	操作系统	[2007-09, 2009-07]
C002	组成原理	[2002-09, 2004-07]
C002	数据结构	[2004-09, 2005-07]
C005		[2007-09, 2008-01]

表17 时态递归插入修改后的选课时态关系表

学号	课程名	成绩	有效时间
S001	C001	85	[2008-01, 2008-07]
S003	C002	90	[2002-09, 2003-07]
S002	C005	85	[2007-09, 2008-01]

(III) 修改的时态关系表既是时态参照关系又是时态被参照关系表

此时需要考虑两种情况:如果不能修改主码值,则只需考虑该时态关系作为参照关系的情况;如果允许修改主码值,从理论上讲,该时态关系将可能有 $3 \times 2 = 6$ 种组合的处理机制,这里不再详述。

结束语 本文主要从时态参照完整性的定义和在进行时态更新违反时态参照完整性约束时,引入不同机制来维护时态数据库数据的正确性。根据掌握的文献,本文首次对违反时态完整性约束处理机制进行研究,将传统关系数据库对参照完整性的处理机制扩展到时态关系数据库,同时在时态中间件TempDB中实现了时态参照完整性的处理机制。下一步的研究工作将是双时态数据库的时态完整性处理机制和实现。

参考文献

- [1] Floris G, Sofie H, Bart K. First-order complete and computationally complete query languages for spatio-temporal databases[J]. ACM Transactions on Computational Logic, 2008, 9(2): 1-51
- [2] Pouri W. Research issues in moving objects databases (tutorial session)[C]// Maggie Dunham, ed. Proceedings of the 2000 ACM SIGMOD international conference. NY USA: ACM, 2005: 581-581
- [3] Ma Z M, Li Yan. Querying Imprecise Data in Sensor Databases [C]// Hong Va Leong, The Ninth International Conference on Mobile Data Management (mdm 2008). NW Washington, DC USA: IEEE Computer Society, 2008: 205-210
- [4] Andrei A, Angela B, Ioana M, et al. Path Summaries and Path Partitioning in Modern XML Databases[J]. World Wide Web, 2008, 11(1): 117-151
- [5] Noh Seo-Young, Gadia Shashi K. Benchmarking temporal database models with interval-based and temporal element-based timestamping[J]. Journal of Systems and Software, 2008, 81(11): 1931-1943
- [6] Sabás G R, Carlos A J. Expressiveness of temporal query languages: on the modelling of intervals, interval relationships and states[J]. Artificial Intelligence Review, 2006, 26(4): 269-289
- [7] Böhlen M H, Jensen C S, Snodgrass R T. Evaluating and Enhancing the Completeness of TSQL2[R]. TR 95-5. Computer Science Department, University of Arizona, 1995
- [8] Salzberg B, Tsotras V. A Comparison of Access Methods for Temporal Data[J]. ACM Computing Surveys, 1997, 31(2): 158-221
- [9] Wes C, Dimitris P. Temporal Integrity Constraints with Indeterminacy[C]// Amr El Abbadi, PMichael L. Brodie, eds. Proceedings of the 26th Very Large Database Conference. CA USA: Morgan Kaufmann Publishers Inc., 2000: 441-450
- [10] Tansel Abdullah U. Integrity Constraints in Temporal Relational Databases. Extended Abstract[C]// Deeber Azada, Proceedings of the International Conference and Information Technology: Coding and Computing (ITCC'04). Washington, DC, USA: IEEE Computer Society, 2004, 2: 460-464
- [11] Tansel Abdullah U. Temporal Data Modeling and Integrity Constraints in Relational Databases[M]. ISCS, 2004: 459-469
- [12] Jan C P. Efficient Checking of Temporal Integrity Constraints Using Bounded History Encoding[J]. ACM Transactions on Database Systems, 1995, 20(2): 149-186
- [13] Dimitris P. Integrity Constraint and Rule Maintenance in Temporal Deductive Knowledge Bases [C]// PRakesh Agrawal, PSean Baker, eds. Porceeding of the 19th VLDB Conference. CA USA: Morgan Kaufmann Publishers Inc., 1993: 146-157
- [14] TempDB: <http://www.cosoft.sysu.edu.cn/TempDB/index.asp>
- [15] Allen James F. Maintaining knowledge about temporal intervals [J]. Communications of the ACM, 1983, 26(11): 832-843
- [16] Andreas S. A Generalization Approach to Temporal Data Models and Their Implementations[D]. Switzerland: Federal Institute of Technology, 1998
- [17] James C P, Curtis D, Tomas I, et al. On the semantics of "now" in databases [J]. ACM Transactions on Database Systems (TODS), 1997, 22(2): 171-214
- [18] 叶小平, 汤庸. 时态变量"Now"语义及相应时态关系运算[J]. 软件学报, 2005, 16(05): 838-845
- [19] Nascimento Mário A. Efficient Indexing of Temporal Databases via B+-Trees[D]. USA: School of Engineering and Applied Science Southern Methodist University, 1996
- [20] 汤娜, 刘瑞君, 陈罗武, 等. XML 文档中时态信息存储方法的研究与比较[J]. 计算机科学, 2008, 35(05): 226-228