

25-27

基于谓词的历史数据库<sup>\*</sup>

张师超

(广西师范大学数学与计算机系 桂林 541004)

**摘要** A model of predicate-based historical database is established in this paper. And some properties of historical database are discussed. It presents the corresponding query language for historical database as well.

**关键词** Temporal database, Historical database, Historical relation, Temporal relation, Query language.

可操作数据的时间方面的时态数据库和历史数据库系统对国内学者来说已不再陌生。国内象唐常杰等研究了多媒体中信息的时态处理方法;复旦大学招兆铨等在时态数据库上建立了一种基于归结的演绎方式;我们建立了 N1NF 和 1NF 式的时态关系代数 and 相应的元组演算,并首先提出在时态数据库中对“NOW”进行了单独讨论;等等。

与国外一样,要形成可商品化的时态数据库或历史数据库方面的软件,必须要建立一个达成共识的时态框架。由于附加时间的对象更为复杂并且时间有其特有性质,因此其理论研究仍处于探索和实践过程中。

本文建立了一个基于谓词的历史数据库模型,讨论了时间的性质。这些性质可用于数据库设计的完整性和一致性检查。也建立了这种历史数据库的查询语言。

## 1 谓词形式的历史数据库

一个对象的一个或多个属性值的一次改变(即从发生到结束)称为一个事件。一个事件可用原子谓词表示成:  $P(a_1, a_2, \dots, a_n, \mu)$ 。其中,  $a_i$  是事件发生的一个属性值,  $i=1, 2, \dots, n$ ,  $\mu$  是该事件生存期的时间区间。 $\mu$  也可以用它的两个端点  $t_1$  和  $t_2$  来代替。通常,我们用  $first(\mu)$  和  $last(\mu)$  分别表示  $\mu$  的左、右端点。一个对象的整个生存期可分为多个事件。多个对象同类型的事件可用这种原子谓词表示,其形式为:

$$r = \{(a_1, a_2, \dots, a_n, \mu) | P(a_1, a_2, \dots, a_n, \mu)\}$$

$r$  也称为一个谓词式关系,或称为谓词式历史关系。谓词形式的历史数据库是  $m$  个这种原子谓词形式的关系的集合。

## 2 时态性质

为简化描述,设对象仅有三个属性  $a_1, a_2$  和  $a_3$ , 其中标识对象的键集中仅有属性  $a_1$ 。那么,谓词式关系为:

$$r = \{(a_1, a_2, a_3, \mu) | P(a_1, a_2, a_3, \mu)\}$$

这里,  $P$  是一个原子谓词,  $\mu$  是时间区间。

对于谓词式关系  $r$ , 有如下一些性质。

$$(I1) \forall x \forall y_1 \forall y_2 \forall z_1 \forall z_2 \forall \mu ((P(x, y_1, z_1, \mu) \wedge P(x, y_2, z_2, \mu)) \rightarrow (y_1 = y_2 \wedge z_1 = z_2))$$

$$(I2) \forall x \forall y \forall z \forall \mu_1 \forall \mu_2 ((P(x, y, z, \mu_1) \wedge P(x, y, z, \mu_2)) \rightarrow \exists (x, y, z, \mu) \in r (\mu_1 \subseteq \mu \wedge \mu_2 \subseteq \mu))$$

$$(I3) \forall x \forall y \forall z \forall \mu (P(x, y, z, \mu) \rightarrow \forall \mu_1 (\mu_1 \subseteq \mu \wedge P(x, y, z, \mu_1)))$$

$$(I4) \forall (x, y, z, \mu) \in r \wedge \forall (x_1, y_1, z_1, \mu_1) \in r \wedge ((x, y, z, \mu) \neq (x_1, y_1, z_1, \mu_1)) \rightarrow (x \neq x_1 \vee y \neq y_1 \vee z \neq z_1)$$

$$(I5) \forall x \forall y \forall z \forall x_1 \forall y_1 \forall z_1 \forall \mu \forall \mu_1 (x = x_1 \wedge (y \neq y_1 \vee z \neq z_1) \wedge P(x, y, z, \mu) \wedge P(x_1, y_1, z_1, \mu_1)) \rightarrow \mu \cap \mu_1 = \Phi$$

规则(I1)表明第一个属性值标识(唯一地)对象的事件。规则(I2)说明事件均有唯一的左、右端点(假定事件的时区是凸区间)。规则(I3)表明了一个事件在其生存期的任何子区间上为真。规则(I4)说明  $r$  中任何两个不同元组是有区别的。规则(I5)蕴含

\* 国家自然科学基金和 863 计划资助课题,并得到广西教委跨世纪人才基金资助。张师超 教授,主要兴趣为时态信息处理和不确定性推理。

同一个对象的不同事件的发生时期是互斥的(注意,这里指同类型的事件)。这5个规则是历史数据库应具有的最基本的性质。不过,(I5)可由(I1)推出。因为若 $\mu_2 = \mu \cap \mu_1 \neq \Phi$ ,则 $P(x, y, z, \mu_2) \wedge P(x_1, y_1, z_1, \mu_2)$ ,所以由 $x = x_1$ 及(I1)可知, $y = y_1 \wedge z = z_1$ ,这与 $y \neq y_1 \vee z \neq z_1$ 假设矛盾,故应该有 $\mu_2 = \Phi$ 。□

另外,若任何对象的属性 $a_2$ 的值均满足某个限制C(即 $a_2 \text{ s. t. } C$ ),则表示成规则形式为:

(I6)  $a_2 \text{ s. t. } C \rightarrow \forall x \forall y \forall z \forall \mu (P(x, y, z, \mu) \wedge y \text{ s. t. } C)$

规则(I6)可用于描述完整性约束。

一般,一个事件现在仍为真,那么该事件的时间区间 $\mu$ 的右端点 $\text{last}(\mu)$ 为“NOW”这个不确定的时间值。于是应有:

(I7)  $P(x, y, z, \mu) \wedge \text{last}(\mu) = \text{"NOW"} \rightarrow \forall y_1 \forall z_1 \forall \mu_1 ((y_1 \neq y \vee z \neq z_1) \wedge P(x, y_1, z_1, \mu_1) \wedge \text{last}(\mu_1) \neq \text{"NOW"})$

注意,以上提供的性质并不是完全的,仍需进一步在实际中总结。

### 3 历史数据库的询问语言

历史数据库的查询语言可表示成这样的形式:

$a : \varphi$ .

其中 $\varphi$ 是一个谓词演算表达式, $a$ 是要查找的对象,它是一个出现在 $\varphi$ 中的自由变量表。首先,我们递归地定义 $\varphi$ 如下:

- ①一个时间关系式是一个谓词演算表达式;
- ②一个非时间的关系式是一个谓词演算表达式;
- ③ $P(x_1, x_2, \dots, x_n, \mu)$ 是一个谓词演算表达式;
- ④若 $\varphi_1$ 和 $\varphi_2$ 是谓词演算表达式,则 $\varphi_1 \wedge \varphi_2, \varphi_1 \vee \varphi_2, \neg \varphi_1, \varphi_1 \rightarrow \varphi_2$ 都谓词演算表达式;
- ⑤若 $\varphi$ 是谓词演算表达式, $\mu$ 是时间变量, $x$ 是非时间变量,则 $\exists \mu(\varphi), \exists x(\varphi), \forall \mu(\varphi), \forall x(\varphi)$ 是谓词演算表达式;
- ⑥除此外无其它谓词演算表达式。

现在,举例说明查询语言的使用。设我们的对象是职工档案(注意,退休后不算职工)。 $a_1$ 是职工号, $a_2$ 是姓名, $a_3$ 是性别, $a_4$ 是工龄, $a_5$ 是出生日期, $a_6$ 是部门, $a_7$ 是工作类别, $a_8$ 是工资。将它表示成原子谓词形式为:

$P(a_1, a_2, a_3, a_4, a_5, a_6, a_7, a_8, \mu)$

这里, $a_i$ 是标识对象的键集中的唯一属性。我们也假设 $t_0$ 是职工档案的开始时间。下面是多个询问语句。

[Q1]职工号为0829的职工工龄是多少?

$x : P(0829, u, v, x, y, z, s, t, \mu) \wedge \text{last}(\mu) = \text{"NOW"}$

[Q2]1994年参加工作的有哪些人?

$y : P(x, y, z, u, v, w, s, t, \mu) \wedge (\text{first}(\mu) = 1994) \wedge (\forall y_1, \forall z_1 \forall u_1 \forall v_1 \forall w_1 \forall s_1 \forall t_1 \forall \mu_1 P(x, y_1, z_1, u_1, v_1, w_1, s_1, t_1, \mu_1) \wedge (\text{first}(\mu_1) > 1994))$

[Q3]1992年张三的工资是多少?他当时年龄是多少?

$a, w : P(x, \text{张三}, y, z, s, u, v, w, \mu) \wedge \text{first}(\mu) \leq 1992 \wedge \text{last}(\mu) \geq 1992 \wedge (a = 1992 - s)$

[Q4]1976年前,哪些人是打字员?

$y : P(x, y, z, s, t, u, \text{打字员}, v, \mu) \wedge \text{last}(\mu) < 1976$

询问[Q1]的结果是唯一一个人(职工号为0829)的工龄,而其它几个询问的结果要涉及几个人的档案。包括[Q3]查询的对象“张三”可能有几个人同名。这些均是对非时间数据的查询,下面我们来看一下对时间方面的询问。

[Q5]李四是什么时候做打字员的?

$t : P(n, \text{李四}, x, y, z, u, \text{打字员}, w, \mu) \wedge \forall x_1 \forall y_1 \forall z_1 \forall u_1 \forall w_1, \forall \mu_1 P(n, \text{李四}, x_1, y_1, z_1, u_1, \text{打字员}, w_1, \mu_1) \wedge \text{first}(\mu) < \text{first}(\mu_1) \wedge t = \text{first}(\mu)$

[Q6]王二是什么时候退休?

$t : P(n, \text{王二}, x, y, z, u, \text{退休}, w, \mu) \wedge (\forall x_1 \forall y_1 \forall z_1 \forall u_1 \forall w_1, \forall \mu_1 P(n, \text{王二}, x_1, y_1, z_1, u_1, \text{退休}, w_1, \mu_1) \rightarrow \text{first}(\mu) < \text{first}(\mu_1)) \wedge t = \text{first}(\mu)$

[Q7]钱五是什么时候去逝的?他去逝前工资是多少?曾在哪些部门工作过?

$t, w, d : P(n, \text{钱五}, x, y, z, u, v, w, \mu) \wedge \text{last}(\mu) \neq \text{"NOW"} \wedge ((\forall x_1 \forall y_1 \forall z_1 \forall u_1 \forall v_1 \forall w_1 \forall \mu_1 P(n, \text{钱五}, x_1, y_1, z_1, u_1, v_1, w_1, \mu_1) \rightarrow \text{first}(\mu) < \text{first}(\mu_1)) \wedge t = \text{last}(\mu)) \wedge (\forall x_2 \forall y_2 \forall z_2 \forall u_2 \forall v_2 \forall w_2 \forall \mu_2 P(n, \text{钱五}, x_2, y_2, z_2, u_2, v_2, w_2, \mu_2) \rightarrow v_2 \neq \text{退休}))$

[Q8]邓六什么时候开始工作?他什么时候做程序员?

$t_1, t_2 : (P(n, \text{邓六}, x, y, z, u, v, w, \mu) \wedge (\forall x_1 \forall y_1 \forall z_1 \forall u_1 \forall v_1 \forall w_1 \forall \mu_1 P(n, \text{邓六}, x_1, y_1, z_1, u_1, v_1, w_1, \mu_1) \rightarrow \text{first}(\mu) < \text{first}(\mu_1)) \wedge t_1 = \text{first}(\mu)) \wedge (P(n, \text{邓六}, x_2, y_2, z_2, u_2, \text{程序员}, w_2, \mu_2) \wedge (\forall x_3 \forall y_3 \forall z_3 \forall u_3 \forall v_3 \forall w_3 \forall \mu_3 P(n, \text{邓六}, x_3, y_3, z_3, u_3, v_3, w_3, \mu_3) \rightarrow \text{first}(\mu_2) < \text{first}(\mu_3)) \wedge t_2 = \text{first}(\mu_2)))$

询问[Q5]和[Q6]是对单个时间值的操作。询问[Q7]是对时间和非时间数据同时处理。询问[Q8]是对多个时间值进行处理。

当然,询问也可以是模糊的,也可能要经过一定的推理才能回答。例如,“年轻的程序员是哪些人?”、“一般需要几年才可能涨一次工资?”等。这需要系统具有处理模糊信息及演绎功能才能回答上述问题。这方面的内容将另文介绍。

### 4 相关问题

前面的(I1), (I2), (I3), (I4), (I5), (I6), (I7)规

则描述了历史数据库的一些功能和性质。这里再对一些具体问题作简单的分析并给出它们的规则。

上一节中属性“工龄”是不会出现负增长的,它可以描述成:

$$\forall \mu_1 \forall \mu_2 (\text{last}(\mu_1) < \text{last}(\mu_2) \rightarrow (\forall n \forall x \forall y \forall z \forall u \forall v \forall w \forall s \forall x_1 \forall y_1 \forall z_1 \forall u_1 \forall v_1 \forall w_1 \forall s_1 ((P(n, x, y, z, u, v, w, s, \mu_1) \wedge P(n, x_1, y_1, z_1, u_1, v_1, w_1, s_1, \mu_2)) \rightarrow z < z_1)))$$

当然,“工龄”在职工退休前均逐年增加1年,这对数据库设计来说是困难的,可将“工龄”改成“工作时间”,那么计算工龄可用指定时间减去“工作时间”属性值得到。

前面的规则(I6)描述某个属性满足某个性质C,这里可将它具体化。例如,一个人的工资不会是负数,它可以描述成:

$$\forall n \forall x \forall y \forall z \forall s \forall u \forall v \forall w \forall \mu (P(n, x, y, z, s, u, v, w, \mu) \rightarrow w \geq 0)$$

另外,1945年以前没有“程序员”这个工作,它可描述成:

$$\forall n \forall x \forall y \forall z \forall s \forall u \forall v \forall w \forall \mu (P(n, x, y, z, s, u, v, w, \mu) \rightarrow v \neq \text{程序员})$$

以上这类约束条件是因对象不同而具有不同的内容及形式,它们应作为历史数据库的一部分加入到系统中,便于对数据库中数据进行完整性和一致性检查。

现在,我们可以来讨论这种历史数据库对不确定时间“NOW”的处理。需要这种处理的操作有:插入、删除、修改、查询等。

插入一个记录(或元组) $(n, x, y, z, s, u, v, w, \mu)$ 到关系 $r$ 中时,该记录的 $\mu$ 可能含有右端点“NOW”,即 $\text{last}(\mu) = \text{“NOW”}$ 。 $r$ 中职员 $n$ 的某个记录的时间区间的右端点也可能是“NOW”,因此,要使得 $r$ 在插入完成后仍满足规则(I7),就必须对“NOW”作特殊处理。它可描述成如下形式:

$$(P(n, x, y, z, s, u, v, w, \mu) \wedge \text{last}(\mu) = \text{“NOW”}) \rightarrow (\forall x_1 \forall y_1 \forall z_1 \forall s_1 \forall u_1 \forall v_1 \forall w_1 \forall \mu_1 ((P(n, x_1, y_1, z_1, s_1, u_1, v_1, w_1, \mu_1) \wedge \text{last}(\mu_1) = \text{“NOW”}) \rightarrow \text{last}(\mu_1) = \text{first}(\mu) - 1))$$

上面规则表明,当记录 $(n, x, y, z, s, u, v, w, \mu)$ 插

入后, $(n, x_1, y_1, z_1, s_1, u_1, v_1, w_1, \mu_1)$ 记录的 $\mu_1$ 若含有“NOW”右端点,则 $\text{last}(\mu_1)$ 应改成 $\text{first}(\mu) - 1$ ,这种改动是假定时间粒度为1,也要求原先的 $r$ 满足规则(I7)。

删除一个记录可能是因为某人调离该单位引起的。这个操作仅需将该职员含有右端“NOW”为时间区间的记录的时间区间的右端点改为指定时间即可。

修改操作可能是将一个记录的时间区间的右端点改成“NOW”,这也需要协调已含有右端点为“NOW”的记录和修改的记录的时间关系。这种操作应满足一定的要求。例如,设 $\mu_1, \mu_2, \mu_3$ 分别为某个职员的三个不同记录,且 $\text{last}(\mu_1) < \text{first}(\mu_2) \wedge \text{last}(\mu_2) < \text{first}(\mu_3)$ ,则一般情况下, $\text{last}(\mu_1)$ 是不能改成“NOW”的。

查询操作可能是这样的,“找出王二为程序员时张三所有的工资情况。”。如果王二为程序的一个记录 $(n_1, \text{王二}, y_1, z_1, s_1, u_1, \text{程序员}, w_1, \mu_1)$ 且 $\text{last}(\mu_1) = \text{“NOW”}$ ,而张三有一个记录为 $(n_2, \text{张三}, y_2, z_2, s_2, u_2, v_2, w_2, \mu_2)$ 且 $\text{last}(\mu_2) = \text{“NOW”}$ ,那么查询的结果是不完全的,因为“NOW”是一个不确定值,所以,在不同时间同样使用该语句查询,其结果可能不一样。

上面建立了一个基于谓词的历史数据库模型,并讨论了一些性质。它实际上与我们建立的1NF式时态关系代数和元组演算语言是等价的。由于附加时间的数据是一个复杂的对象,应从不同角度,不同程度加以讨论。另外,我们也想以此为基础建立一个演绎历史数据库,使之逐步实用化。

## 参 考 文 献

- [1]张师超等,基于间断区间的1NF时态关系代数,高技术通讯,1995. 增刊
- [2]Mckenzie L., Snodgrass R., Evaluation of relational algebras in incorporating the time dimension in databases, ACM Computing Surveys, 23(4), 1991