

多媒体数据库

数据库

时态概念模型

(16)

多媒体

计算机科学 1996 Vol. 23 No. 2

66-69

一种多媒体数据库时态概念模型

周学海 赵振西

(中国科技大学计算机系 合肥 230027)

TP311.13

TP391

摘要 When the multimedia objects in the MDBMS are represented, the temporal relationships between intra-medias or between inter-medias must be maintained in order to keep the synchronizations. In the paper, based on the Conceptual Temporal Models given by T. D. C. Little, we present a new approach to select the temporal parameters for supporting the user's interactive operations, and give a new storage model for the temporal parameters.

关键词 MDBMS, Conceptual temporal model, Interactive operation.

在多媒体数据库(MDBMS)中,所要展示的媒体可分为三类:静态、动态和混合类型。静态类型是指来自不同(或相同)存储位置,但无时间变化的静态图像、文本等;动态类型是指时间、空间都变化的视频、音频、动画等媒体;混合类型是静态和动态类型的综合。在展示混合类型和动态类型的媒体时,必须考虑媒体间以及媒体内部在时间上的同步关系。例如,在同一界面上说话者的口形应与其声音保持同步(lip-synchronization),在同一屏幕的不同窗口显示动画时,应保持各窗口的画面在时间上的同步。因此多媒体同步是多媒体系统中的一个关键问题。

在MDBMS中,有关同步问题应满足如下要求:1)提供有效的工具来方便地说明复杂媒体间以及媒体内部的时间约束关系。2)支持与时间相关的数据检索。3)对描述媒体的时间约束参数,能有效地存储、检索,并能方便地将时间约束关系与媒体源数据联系起来。4)具有高效的展示算法。5)不仅支持顺序展示,还应支持反向展示、快进、快退、暂停及重放功能。为此,需要有较好的时态概念模型来有效地表达多媒体对象的时态语义。

本文在 Thomas. D. C. Little 等人提出的基于时间片的概念模型基础上,对该模型时态参数的选择进行了适当调整,并改进了时态参数的存储模型,使得在展示过程中,能更好地支持用户的交互式操作。

一、时态概念模型

为了从MDBMS中检索时间相关的数据,既要建立描述展示所必须的时态语义的时态概念模型,又要建立存储时态语义的数据模型。目前在描述时

态语义方面,经常使用的方法是基于时间片的方法。

1.1 基本的时态关系

一个时间片由两个时刻定义,表示的是一段期间。任意给定两个时间片,它们之间的相对关系有13种^[2](before, meets, overlaps, starts, finishes, during, equals, before⁻¹, meets⁻¹, overlaps⁻¹, starts⁻¹, finishes⁻¹, during⁻¹),这些关系指出了时间片在时间上的相对关系,表示了七种情况,而有六种关系是这七种关系的逆关系。

为了对多媒体展示过程模型化,可以用时间片来代表媒体数据。多媒体对象可分为组合的多媒体对象和基本的多媒体对象,后者是指这些对象是不能再分割的对象,从展示的角度考虑是最基本的展示单元,它对应于基本时间片,而组合的多媒体对象由基本的对象或组合的多媒体对象组成,从展示的角度考虑,是可以再分步展示的对象,它对应于组合时间片。可以证明,对于任意复杂的时态关系,都可以用这十三种关系表示出来。

1.2 n元时态关系

n元时态关系是对二元时态关系的扩充,它引入了多个时间片,定义出多个时间片的时态关系。

定义1 给定一组有序的时间片集合 $P = \{P_1, P_2, P_3, \dots, P_n\}$ 和二元关系 TR, TR^{-1} 称为 n 元时态关系,当且仅当 $P_i TR P_{i+1}, \forall i (1 \leq i \leq n-1)$ ^[3]。

图1是十三种时态关系中的七种。对每一关系中的 n 个时间片,其起始时间按递增序排列,这样定义是为了使展示算法描述简单,而并不影响模型的表达能力。

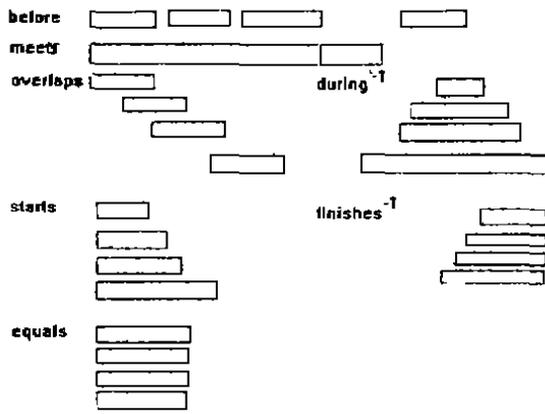


图1 七种 n 元时态关系

在 n 元时态关系中,我们选择如下时态参数: $T[i]$: 第 i 块媒体的展示时间; $D[i-1]$: 第 i 块媒体开始展示需经过的延时; T_{TR} : 多媒体对象的整个展示时间。那么 n 元时态约束关系如下:

关系	时态约束	T_i
before	$T(i)+D(i-1)<D(i)$	$T(n)+D(n-1)$
meets	$T(i)+D(i-1)=D(i)$	$T(n)-D(n-1)$
overlaps	$D(i)>D(i-1), D(i-1)+T(i)<D(i)+T(i-1)$	$T(n)-D(n-1)$
during ⁻¹	$D(i)<D(i+1), D(i-1)+T(i)>D(i)+T(i+1)$	$T(i)$
starts	$T(i)<T(i-1), D(i)=0$	$T(n)$
finishes ⁻¹	$T(i)+D(i-1)=T(i-1)+D(i)$	$T(i)$
equals	$T(i+1)=T(i), D(i)=0$	$T(i)$

1.3 n 元逆关系

在 MDBMS 中,除了能够支持按顺序展示多媒体对象外,还应能够支持其它方式的信息存取。如反向展示、中间点暂停和重放等操作。

首先我们考查一下二元逆关系,其定义如图 2 所示^[1]。

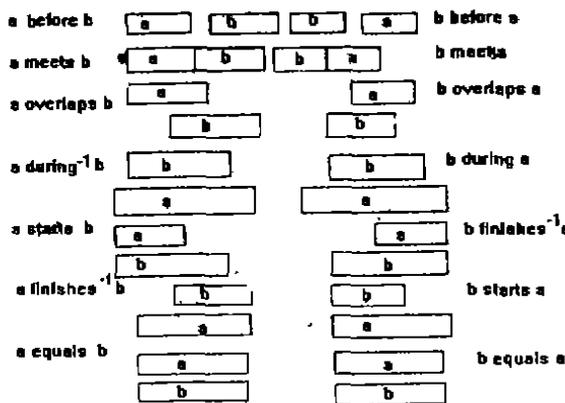


图 2

由向前展示的时态参数导出反向展示的时态参

数如下:

	Forward	Reverse	T_{a-r}	T_{b-r}	D_r
before	before	before	T_b	T_a	$D+T_b-T_a$
meets	meets	meets	T_b	T_a	T_b
overlaps	overlaps	overlaps	T_b	T_a	$D+T_b-T_a$
during ⁻¹	during	during	T_b	T_a	T_a-T_b-D
starts	finishes ⁻¹	finishes ⁻¹	T_b	T_a	T_b-T_a
finishes ⁻¹	starts	starts	T_a	T_b	T_a-T_b
equals	equals	equals	T_b	T_a	0

类似地我们可以定义 n 元逆关系如下:

定义 2 给定一组有序的 n 个时间片集合 $P = \{P_1, P_2, P_3, \dots, P_n\}$, 设二元关系 TR 的反关系为 TR^{-1} , 如果 TR^{-1} 是 p 集合上的 n 元时态关系, 那么当 $P_i TR, P_{i-1} TR^{-1}$ 成立时, 则 TR^{-1} 称为 n 元逆关系^[1]。

给定一组 n 元时态关系的时态参数, 我们可以得出相应的逆关系时态参数, 如下:

	Forward	Reverse	$T_r[i]$	$D_r[i]$
before	before	before	$T[n+P-i]$	$T_{TR}-(T[n-1]+D[n-i])$
meets	meets	meets	$T[n+1-i]$	$T_{TR}-(T[n-1]+D[n-i])$
overlaps	overlaps	overlaps	$T[n+1-i]$	$T_{TR}-(T[n-1]+D[n-i])$
during ⁻¹	during	during	$T[n+1-i]$	$T_{TR}-(T[n-1]+D[n-i])$
starts	finishes ⁻¹	finishes ⁻¹	$T[n+1-i]$	$T_{TR}-(T[n-1]+D[n-i])$
finishes ⁻¹	starts	starts	$T[n+1-i]$	$T_{TR}-(T[n-1]+D[n-i])$
equals	equals	equals	$T[n+1-i]$	$T_{TR}-(T[n-1]+D[n-i])$

二、时态参数的存储模型

对于多媒体数据库系统而言,我们希望时态参数能以比较归整化的数据结构表示,以利于存储和检索。如前面所述的 n 元时态参数,需要存储的时态参数为 $D[i], T[i]$ 以及关系类型(结点类型),可用如下结构表示:

TYPE	$D[i-1]$	$T[i]$	Child	Next
------	----------	--------	-------	------

TYPE: 指出结点类型。我们将七种关系分为两类。一类为串行关系 $TYPE = SEQ$ (before, meets); 一类为并行关系 $TYPE = PAR$ (overlaps, during⁻¹, starts, finishes⁻¹, equals)。当 $TYPE = TERM$ 时, 表示该结点所示时间片为基本时间片; 而 $TYPE = FERM$ 时, 表示该结点所示时间片为组合时间片。

$D[i-1]$: 该结点开始展示需经历的延时。

$T[i]$: 该结点的展示时间。

Child: 指向其左子树。若该时间片是可分解的组合时间片, 则 Child 非空, 指向第一个子时间片。

否则指向媒体流数据的存贮地址。

Next: 将同级时间片连接在一起, 即组合时间片的所有子时间片由该指针连接。

例如, 假定一多媒体对象 media 由 media1、media2 和 media3 组成, 而 media1 由 media11 和 media12 组成, media2 由 media21 和 media22 组成, media3 由 media31 和 media32 组成, 且对应的时态参数分别为 (PAR, D, T), (SEQ, D[1], T[1])... 则其时态概念模型的层次结构 (时态语义树) 如图 3 所示:

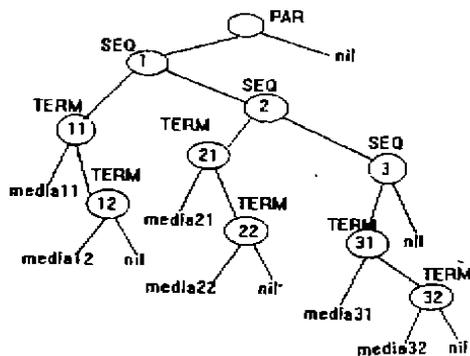


图 3

采用上述的数据结构, 选择什么类型的数据库来存贮多媒体对象的时态语义? 比较直观的选择是层次数据库模型, 但是该结构比较归整, 显然采用关系模型也是可行的, 采用面向对象的数据库模型也完全可以。

三、展示算法中时态参数的确定

复杂多媒体对象的展示, 涉及到计算机硬件本身的性能, 特别是对于动态数据的展示, 数据供给是多媒体展示的瓶颈问题。如果在网络环境下多媒体数据又不在本地工作站上, 获取数据又不得不考虑网络传输的延时, 这种延时与网络负载、数据链路的路由选择有密切关系。这些问题是设计实用的展示算法时必须考虑的问题。在高层同步模型的基础上, 我们着重研究随着用户的交互式操作, 在展示算法中时态参数的更新算法, 以便为低层展示算法提供时态参数, 而忽略属于低层同步所需考虑的问题。

3.1 确定媒体的开始展示时刻

多媒体对象的时态语义可按照前面所示的表示方法表示成一树型结构, 对该树进行遍历可得出每种媒体的开始展示时刻 T_{start} , 算法描述如下:

1. 对该多媒体对象, 给出其开始展示时刻 T_{start} ;
2. 先序遍历该时态语义树。

(1) 若结点为非终结点, 以 $T_{start} = T_{start} + D[i-1]$ 为参数, 递归调用该算法。

(2) 若结点为终结点, 则 $T_{start} = T_{start} + D[i-1]$

3.2 确定反向展示时态参数算法

用户希望能反向展示有些多媒体对象, 要求时态概念模型能支持用户的反向展示操作。根据 1.3 节逆关系, 可以很方便地由正向时态参数求出反向时态参数, 然后根据新的时态参数, 确定每一媒体的开始展示时刻, 算法描述如下:

先序遍历该时态语义树。

(1) 若结点为非终结点, 则其子结点的 $D_{new}[i] = T_{TR} - (T[n-i] + D[n-i])$, $T_{new}[i] = T[n+1-i]$, 其中 T_{TR} 为该结点的 T , n 为该非终结点所含子时间片个数, 递归调用该算法对该非终结点的子结点处理。

(2) 若结点为终结点, 考虑下一结点。

3.3 中间点暂停时时态参数的确定

多媒体对象展示过程中, 用户可能会在中途暂停, 尔后继续向前展示或反向展示。

对中间点暂停重播算法设计, 需考虑组合时间片和基本时间片两种情况, 若用户暂停时刻落在基本时间片内, 系统记录的暂停时刻应等待该时间片展示完, 而对于组合时间片, 则需重新计算它所包含的基本时间片的时态参数。算法描述如下:

设暂停时刻为 T_{pause} , 则考查时态语义树中每一结点。

1. 若其时态关系为串行, 则考查其子结点。

(1) 若 $D[i-1] \leq T_{pause} < D[i-1] + T[i]$, 则 $T_{new}[i] = T[i] + D[i-1] - T_{pause}$
 $T_{sub-pause} = T[i] - T_{new}[i]$
 $D_{new}[i] = D[i] - T_{pause}$

以 $T_{sub-pause}$ 为参数, 递归调用该算法对其子结点处理。

(2) 若 $D[i-1] + T[i] \leq T_{pause} < D[i]$, 则 $D_{new}[i] = D[i] - T_{pause}$

(3) 若 $T_{pause} \geq D[i]$, 考虑下一结点。

2. 若该结点时态关系为并行关系, 则考查其子结点。

(1) 若 $D[i-1] \leq T_{\text{pause}} < D[i-1] + T[i]$, 则

$$D_{\text{new}}[i] = T[i] + D[i-1] - T_{\text{pause}}$$

$$T_{\text{sub-pause}} = T[i] - T_{\text{new}}[i]$$

$$D_{\text{new}}[i] = D[i] - T_{\text{pause}}$$

以 $T_{\text{sub-pause}}$ 为参数递归调用该算法对其子结点进行处理。

(2) 若 $D[i-1] + T[i] \leq T_{\text{pause}} < D[i]$, 则

$$D_{\text{new}}[i] = D[i] - T_{\text{pause}}$$

(3) 若 $T_{\text{pause}} \geq D[i]$, 则考虑下一结点。

对于暂停后反向重放, 确定时态参数的算法描述如下:

1. 若该结点为非终结点, 且时态关系为串行 (SEQ), 则考查其子结点。

(1) 若 $D[i-1] \leq T_{\text{pause}} < D[i-1] + T[i]$, 则

$$T_{\text{new}}[1] = T_{\text{pause}} - D[i-1]$$

$$D_{\text{new}}[1] = T_{\text{pause}} - (D[i-1] + T[i-1])$$

以 $T_{\text{new}}[1]$ 为 T_{pause} 递归调用该算法对 $T[i]$ 结点处理

$$T[i] = T_{\text{new}}[1]$$

FOR $j = i-1$ DOWNTO 2

以 $T[j]$ 为 T_{pause} 递归调用该算法对

$T[j]$ 内部处理

$$T_{\text{new}}[j] = T[i-j+1]$$

$$D_{\text{new}}[j] = T_{\text{pause}} - (T[i-j] + D[i-j])$$

ENDFOR

(2) 若 $D[i-1] + T[i] \leq T_{\text{pause}} < D[i]$, 则

$$D[0] = T_{\text{pause}} - (D[i-1] + T[i])$$

FOR $j = i$ DOWNTO 1

以 $T[j]$ 为 T_{pause} 递归调用该算法对

$T[j]$ 内部处理

$$T_{\text{new}}[j] = T[i-j+1]$$

$$D_{\text{new}}[j] = T_{\text{pause}} - (T[i-j] + D[i-j])$$

ENDFOR

2. 若该结点为非终结点, 且时态关系为并行关系时, 考查其每一子结点。

(1) 若 $D[i-1] \leq T_{\text{pause}} < D[i-1] + T[i]$, 则

$$T_{\text{new}}[i] = T_{\text{pause}} - D[i-1]$$

以 $T_{\text{new}}[i]$ 为 T_{pause} 递归调用该算法对该子结点处理。

(2) 若 $D[i-1] + T[i] \leq T_{\text{pause}}$, 则

$$T_{\text{new}}[i] = T[i]$$

以 $T_{\text{new}}[i]$ 为 T_{pause} 递归调用该算法对该子结点处理。

(3) 找出满足以上两条件的最大 i 值 MAXI, 然后确定其逆关系参数

FOR $j = 1$ TO MAXI

$$T_{\text{new}}[i] = T[i-j+1]$$

$$D_{\text{new}}[i] = T_{\text{pause}} - (T[i-j] + D[i-j])$$

ENDFOR

3.4 展示算法中快进、快退操作的处理

在交互式展示存储的多媒体对象时, 用户可能会对某些类型的多媒体对象进行快进、快退展示。为了处理快进、快退操作, 在我们的模型中引入展示速度因子 V_i , 可以有两种策略来处理用户的这种操作。

方法一: 在高层同步机制中, 仅对展示速度因子 V_i 修改, 而具体到对展示时间、延迟时间的修改在低层同步机制中完成。

方法二: 在高层同步机制中, 不仅对展示速度因子 V_i 修改, 而且对展示时间、延迟时间直接修改。

在低层同步展示算法中, 可以通过改变展示速度或通过略去部分数据的方法实现快进、快退展示。

参考文献

- [1] Thomas D. C. Little 等, Interval-based conceptual models for time-dependent multimedia data, IEEE Trans. on Know. and Data Eng., Vol. 5, No. 4, 1993
- [2] J. F. Allen, Maintaining knowledge about temporal intervals, CACM, Vol. 26, Nov. 1983
- [3] Neil Williams and Gordon S. Blair, Distributed multimedia applications: A review, Computer Commun. Vol. 17, No. 2, 1994
- [4] T. D. C. Little 等, Scheduling of bandwidth-constrained multimedia traffic, Computer Commun. Vol. 15, No. 6, 1992