CN4321258/ TP
ISSN 10072130X
计算机工程与科学
COMPUTER ENGINEERING & SCIENCE
2004 年第 26 卷第 3 期
Vol126 ,No13 ,2004

文章编号:10072130X(2004)0320065204

多数据库系统中的模式映射方法X

Schema Mapping in Multidatabase Systems

李瑞轩,卢正鼎,肖卫军,李 兵
LI Rui2xuan,LU Zheng2ding,XIAO Wei2jun ,LI Bing
(华中科技大学计算机学院,湖北 武汉 430074)
(School of Computer Science and Technology, Huazhong University of Science and Technology, Wuhan 430074, China)

摘 要:多数据库系统一般具有四级模式结构,全局用户只能访问全局模式,而最终的数据必须从各局部数据库系统中获得,因此必须建立多数据库系统的模式映射,它表示了局部模式通过输出模式集成为全局模式的相应转换。本文给出了一种多数据库系统中的模式映射方法,并使用模式映射树来存储和表达这种模式映射。

Abstract: There are four levels of schema structures in multidatabase systems. Multidatabase global users can only access global schemas. But the final answer should be obtained from local database systems. So, schema mappings, which represent the transformation and integration from local schemas to export schemas and from export schemas to global schemas, should be established. This paper gives an approach of schema mapping in multidatabase systems and uses a schema mapping tree to store and express these mappings.

关键词:多数据库系统;模式结构;模式映射;模式映射树

Key words: multidatabase system; schema architecture; schema mapping; schema mapping tree

中图分类号:TP311.13 **文献标识码**:A

1 引言

多数据库系统研究的目的主要是解决多个成员数据库之间数据共享和集成的问题[1]。各成员数据库的局部模式可能是由不同的用户基于不同的数据模型独立设计的,它们之间可能存在着各种差异和冲突。为了给多数据库用户提供透明的访问接口,需要在多数据库全局层屏蔽这些差异,但又不能简单地通过修改局部模式来解决,因为多数据库要保证各成员数据库的自治性,以保证那些建立在各自数据库之上的原有应用程序仍然能够继续运行。通常的办法是在多数据库系统中构造一个全局模式,这一全局模式是由各参与的成员数据库中的局部模式经过一定的模式变换得到的[2]。

多数据库系统的模式结构决定了模式映射的层次和查询处理的流程。现有的多数据库系统大多采用四级模式结构或类似的模式结构,它包括四种模式:局部模式、输出模式(也叫中间模式)、全局模式(也叫联邦模式)和外模式[3]。多数据库用户只能访问全局模式,它的实际数据必须从各

X 收稿日期:2002210222;修订日期:2003203212
基金项目:国家高性能计算基金资助项目(99319)
作者简介:李瑞轩(1974 -),男,湖北宜昌人,博士生,讲师,研究方向为分布异构系统集成;卢正鼎,教授,博士生导师,研究方向为CIMS及分布异构系统集成。
通讯地址:430074 湖北省武汉市华中科技大学计算机学院;Tel:(027)87544285;E2mail:rxli@public.wh.hb.cn
Address: School of Computer Science and Technology, Huazhong University of Science and Technology, Wuhan, Hubei 430074, P. R. China

局部数据库系统中获得。这里,使用模式映射来表示局部模式通过输出模式集成为全局模式的相应转换。经过模式集成后,全局模式中存储的信息实际上是全局模式到输出模式、输出模式到局部模式的映射。本文给出了一种多数据库系统中的模式映射方法,并使用模式映射树来存储和表达这种模式映射。

2 多数据库系统中的类模式

多数据库全局模式中包含一组全局类和模式映射信息。下面用类来定义模式中的信息。

定义 1 多数据库的类模式是一个多元组 $C(U, D, I, Q, M, Parent)$,其中 C 是类名,U 是组成 C 的有限属性集,D 是 U 中属性的值域,I 是类 C 的对象所响应的消息的集合,Q 是类 C 的对象所满足的限定条件集,M 是类 C 的模式映射信息的集合,$Parent$ 是类 C 继承的父类的集合。类 C 有主键 $K \subseteq U$。

全局类模式描述了全局类的结构及语义约束,它可以按一定的条件 Q 转换成中间类模式。类是类模式在某一时刻的当前值。为了简单起见,本文不讨论属性的值域、消息和继承情况,将类模式简化为 $C(U, Q, M)$;有时也用 $C(U, Q, M)$ 或类名 C 表示类。

在多数据库系统中存在三种类:全局类、中间类和局部类。

定义 2 全局类是指对多数据库全局用户可见的类。全局类是虚拟的,它并不具有实际的对象,而是由若干中间类和局部类按模式映射 M 组成的,其中 $M \neq$ “ 。

定义 3 中间类是指全局类在某个输出模式上的映射,其中 $M \neq$ “ 。中间类也是虚拟的。全局类与中间类通过中间映射进行联系,它们具有 $1:n$ 的联系。

定义 4 局部类是指中间类映射到某个局部模式上的基本类,其中 $M =$ “ 。中间类与局部类通过局部映射进行联系,它们具有 $1:1$ 或 $1:n$ 的联系。

3 模式映射

多数据库系统呈现给用户的是一组全局类,这组全局类实际上是由若干个输出模式中的中间类所组成的,而中间类又是由若干个局部类组成的。那么,在这些模式之间必定存在一种将这些类维系起来的映射机制,这就是模式映射。

定义 5 类 $C(U, Q, M)$ 的模式映射 M 是全局类与中间类、中间类与局部类之间联系的集合。

模式映射描述了多数据库全局类模式是如何从局部数据库中获取数据的。它又可分为中间映射(EM)和局部映射(LM)。其中,中间映射是全局类与中间类之间联系的集合,局部映射是中间类与局部类之间联系的集合。

3.1 中间映射

根据全局类与中间类之间的关系,可将中间映射分为水平映射、垂直映射和混合映射。

定义 6 全局类 $C(U, Q, M)$ 上的水平映射 HM 是一操作,它将 C 按照一组给定的条件 $P_1, \cdots, P_n$ 映射成一组中间类模式 $C_1(U_1, Q_1, M_1), \cdots, C_n(U_n, Q_n, M_n)$,满足:(1) $K_1 = K_2 = \cdots = K_n = K$;(2) $U_1 = U_2 = \cdots = U_n = U$;(3) $Q_i = Q \wedge P_i$;(4) $M_i = LM_i$。记为 $C(HM) < P > = \{C_1, \cdots, C_n\}$。其中,$K$,$K_1, \cdots, K_n$ 分别为 C,$C_1, \cdots, C_n$ 的主键(下同);$P_i \wedge P_j =$ “ ,$\vee P_i =$ True,$P = \{P_1, \cdots, P_n\}$,$i, j \in \{1, \cdots, n\}$。

由定义知,水平映射将全局类的对象按对象标识(OID)横向以某些条件映射成中间类的对象。

定义 7 全局类 $C(U, Q, M)$ 上的垂直映射 VM 是一操作,它将 C 按照一组给定的属性 $A_1, \cdots, A_n$ 映射成一组中间类模式 $C_1(U_1, Q_1, M_1), \cdots, C_n(U_n, Q_n, M_n)$,满足:(1) $K_1 = K_2 = \cdots = K_n = K$;(2) $U_i = A_i$;(3) $Q_1 = Q_2 = \cdots = Q_n = Q$;(4) $\Pi_{K_1}(C_1) = \cdots = \Pi_{K_n}(C_n) = \Pi_K(C)$;(5) $M_i = LM_i$。记为 $C(VM) < A > = \{C_1, \cdots, C_n\}$。其中,$A \subseteq U$,$A_i \cap A_j = K$,$\cup A_i = U$,$A = \{A_1, \cdots, A_n\}$,$i, j \in \{1, \cdots, n\}$。

由定义知,垂直映射将全局类的对象按类的属性纵向以属性组映射成中间类的对象。为了保证全局类的可重构性,应将键属性映射到各个中间类中。

定义 8 全局类 $C(U, Q, M)$ 上的混合映射 MM 是一操作,它将 C 按照一组属性 $A_1, \cdots, A_n$ 和一组条件 $P_1, \cdots, P_n$ 映射成一组中间类模式 $C_1(U_1, Q_1, M_1), \cdots, C_n(U_n, Q_n, M_n)$,满足:

(1) $K_1 = K_2 = \cdots = K_n = K$; (2) $U_i = A_i$; (3) $Q_i = Q \wedge P_i$; (4) $M_i = LM_i$。记为 $C(MM) < AP > = \{C_1, \cdots, C_n\}$。其中，$A_i \subset U$，$A_i \cap A_j = K$，$\cup A_i = U$，$P_i \wedge P_j =$ “”，$\vee P_i =$ True，$AP = \{ < A_i, P_i > \mid i = 1, \cdots, n\}$，$i, j \in \{1, \cdots, n\}$。

由定义知，混合映射是水平映射和垂直映射的混合操作。视应用需要，可先水平映射后垂直映射，也可先垂直映射后水平映射。在查询处理由中间结果合并得全局结果时，可按相应次序进行并操作和连接操作。多数据库系统中常见的中间映射多为混合映射。

定义 9 全局类 $C(U, Q, M)$ 与另一全局类 H 已水平映射为 $H_1(U'_1, Q'_1, M'_1), \cdots, H_n(U'_n, Q'_n, M'_n))$ 在公共属性 A 上的相关映射 CM 是一操作，它将 C 映射成一组中间类模式 $C_1(U_1, Q_1, M_1), \cdots, C_n(U_n, Q_n, M_n)$，满足：(1) $K_1 = K_2 = \cdots = K_n = K$; (2) $U_1 = U_2 = \cdots = U_n = U$; (3) $\Pi_A(C_i) \subset \Pi_A(H_i)$; (4) $Q_i = Q \wedge Q'_i$; (5) $M_i = LM_i$。记为 $C(CM) < H > = \{C_1, \cdots, C_n\}$。其中，$H = \{H_1, \cdots, H_n\}$，$i \in \{1, \cdots, n\}$，公共属性 A 称为相关属性。

由定义知，相关映射是指一个全局类根据另一个已水平映射的全局类的相关属性来进行映射。在查询结果合并处理过程中，对于相关映射需要在两个类的相关属性上作半连接操作，以减少全局连接和数据传输的开销。

3.2 局部映射

局部映射有 1∶1 和 1∶n 两种，这里主要针对 1∶1 的局部映射进行处理。对于 1∶n 的映射，处理方法与中间映射类似。

定义 10 中间类 $C(U, Q, M)$ 的局部映射 LM 是一操作，它将 C 按照一种转换函数 f 映射成一局部类模式 $C_1(U_1, Q_1, M_1)$，满足：$K_1 = f(K)$; $U_1 = f(U)$; $Q_1 = f(Q)$; $M_1 =$ “”。记为 $C(LM) = C_1$。

局部映射实际上是定义了一种中间类到局部类的转换关系，例如属性名、属性类型的转换等。

由此，我们可以给出模式映射的形式化定义及其命题如下：

定义 11 多数据库的模式映射可以描述为：$C(O) < S > = \{C_1, \cdots, C_n\}$，其中 $O \in \{HM, VM, MM, CM, LM\}$，$S = \{S_1, \cdots, S_n\}$，$i = 1, \cdots, n$。当 $O \in \{HM, VM, MM, CM\}$ 时，$S_i = \{ < A_i, P_i > \mid$ 如果 $O = HM$，则 $A_i = U$；如果 $O = VM$，则 $P_i =$ True；如果 $O = CM$，则 $A_i = U$，$P_i = Q'_i\}$；当 $O = LM$ 时，$n = 1$，$S = S_1$ 表示转换函数 f。

命题 1 模式映射操作的结果是将全局类或中间类 $C(U, Q, M)$ 按操作 O 映射为一组满足条件的子类 $C_1(U_1, Q_1, M_1), \cdots, C_n(U_n, Q_n, M_n)$，且当 $O \in \{HM, VM, MM, CM\}$ 时，$K_i = K$，$U_i = A_i$，$Q_i = Q \wedge P_i$，$M_i = LM_i$；当 $O = LM$ 时，$K_1 = f(K)$，$U_1 = f(U)$，$Q_1 = f(Q)$，$M_1 =$ “”。

值得指出的是，为了应用和优化的需要，有时也可将全局类映射为其它的全局类，将中间类映射为其它的中间类。

4 模式映射树

多数据库模式映射定义了全局类、中间类及局部类之间的关联，它主要用于多数据库全局查询的分解处理。可以用多种方式来存储和表达模式映射信息。由于多数据库全局查询处理过程中需要将全局查询转换为查询树，为了便于模式映射信息在查询树中的使用，我们使用模式映射树来存储和管理多数据库模式映射，它存储了用于查询分解的模式信息。

定义 12 模式映射树是一棵树 $T = (V, E)$，其中：

(1) V 是节点集。每个节点用 $C(U, Q, M)(O)$ 表示，其中 $C(U, Q, M)$ 表示类，O 表示该节点的操作，且 $O \in \{HM, VM, MM, CM, LM\}$。如果该节点是根，则 $Q =$ True，$M = T$。

(2) E 是边集。对于任意操作 O，若有 $C(O) < S > = \{C_1, \cdots, C_n\}$，其中 $S = \{S_1, \cdots, S_n\}$，$i = 1, \cdots, n$，则：

①如果 $O \in \{HM, VM, MM, CM\}$，那么 $S_i = < A_i, P_i >$，在模式映射树中有从节点 $C(U, Q, M)(O)$ 到 $C_i(U_i, Q_i, M_i)(O_i)$ 的边，且 $K_i = K$，$U_i = A_i$，$Q_i = Q \wedge P_i$，M_i 是以 $C_i(U_i, Q_i, M_i)(O_i)$ 为根节点的子树；

②如果 $O = LM$，那么 S_i 是转换函数 f，在模式映射树中有从节点 $C(U, Q, M)(O)$ 到 $C_i(U_i, Q_i, M_i)(O_i)$ 的边，且 $K_1 = f(K)$，$U_1 = f(U)$，$Q_1 = f(Q)$，$M_1 =$ “”。

例 1 在一个多数据库系统中，全局类模式 $C_0(U_0, Q_0, M_0)$ 经水平映射 (O_0) 为 $C_1(U_1, Q_1, M_1)$ 和 $C_2(U_2, Q_2, M_2)$，而 C_1 经垂直映射 (O_1)

为 $C_{11}(U_{11}, Q_{11}, M_{11})$ 和 $C_{12}(U_{12}, Q_{12}, M_{12})$，$C_2$ 经垂直映射 (O_2) 为 $C_{21}(U_{21}, Q_{21}, M_{21})$ 和 $C_{22}(U_{22}, Q_{22}, M_{22})$，其模式映射树如图 1 所示。

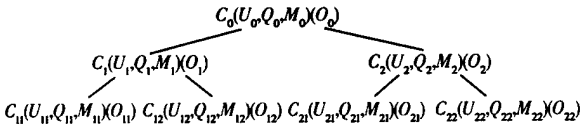

图 1　模式映射树示例

由定义 12 可得模式映射树的性质：

性质 1　在模式映射树 T 中，对于任一节点 $C(U, Q, M)(O)$，如果它是根节点，那么它所表示的类一定是全局类，且 Q = True，M 是整个模式映射树 T；如果它是叶节点，那么它所表示的类一定是局部类，有 M = “；如果它既不是根节点也不是叶节点，那么它一定是中间类，且 M 是以该节点为根的子树。

关于模式映射树有如下定理：

定理 1　在模式映射树 T 中，根节点 $C_0(U_0, Q_0, M_0)(O_0)$ 的类模式为 $C_0(U_0, Q_0, M_0)$，设从根节点到任一非叶节点 $C_n(U_n, Q_n, M_n)(O_n)$ 的所有节点为 $C_0(U_0, Q_0, M_0)(O_0)$，$C_1(U_1, Q_1, M_1)(O_1)$，…，$C_n(U_n, Q_n, M_n)(O_n)$，其映射条件依次为 P_1，P_2，…，P_n，则有：(1) $K_n = K_0$；(2) $U_n \wedge U_{n-1} \wedge \cdots \wedge U_1 \wedge U_0$；(3) $Q_n = Q_0 \wedge P_1 \wedge \cdots \wedge P_n$；(4) M_n 是以 $C_n(U_n, Q_n, M_n)(O_n)$ 为根的子树。

证明　用归纳法证明。$n = 1$ 时，表示 $C_n(U_n, Q_n, M_n)(O_n)$ 与根节点 $C_0(U_0, Q_0, M_0)(O_0)$ 直接相连，由命题 1 知定理 1 显然成立。

假设 $n = k$ 时定理 1 成立，即 $K_k = K_0$；$U_k \wedge U_{k-1} \wedge \cdots \wedge U_1 \wedge U_0$；$Q_k = Q_0 \wedge P_1 \wedge P_2 \wedge \cdots \wedge P_k$。那么，当 $n = k + 1$ 时，$C_k(U_k, Q_k, M_k)(O_k) < S_{k+1} > = C_{k+1}(U_{k+1}, Q_{k+1}, M_{k+1})$，因 $C_{k+1}(U_{k+1}, Q_{k+1}, M_{k+1})$ 不是叶节点，故 $O_k \in \{HM, VM, MM, CM\}$，$S_{k+1} = < A_{k+1}, P_{k+1} >$。由命题 1 和定义 12 有，$K_{k+1} = K_k = K_0$；$U_{k+1} = A_{k+1} \wedge U_k \wedge \cdots \wedge U_1 \wedge U_0$；$Q_{k+1} = Q_k \wedge P_{k+1} = Q_0 \wedge P_1 \wedge P_2 \wedge \cdots \wedge P_k \wedge P_{k+1}$；$M_{k+1}$ 是以 $C_{k+1}(U_{k+1}, Q_{k+1}, M_{k+1})(O_{k+1})$ 为根的子树。因此，$k + 1$ 时定理也成立。定理得证。 □

在多数据库查询分解的过程中，首先将全局查询转换成内部结构表示的查询树，然后将全局查询树与模式映射树合并转换成部分优化的中间查询树，并最终转换为相应局部数据库上的局部查询，并在分解处理的过程中逐步实现查询优化。

5　结束语

多数据库系统的模式集成是一个比较复杂的问题。模式映射是模式集成之后模式信息在多数据库系统中的表现形式，它构成了多数据库系统的全局数据字典。通过模式映射树这一结构，模式映射将全局类、中间类和局部类有机地结合起来。一方面，模式映射表明了全局类分解成中间类和局部类的过程；另一方面，它指出了如何从局部类、中间类组合成全局类。使用模式映射为多数据库的查询分解处理提供了方便，同时也指出了全局子查询从局部数据库获得结果之后进行中间结果合并的策略。在我们自行研制的多数据库系统 Panorama 中，对本文所提出的模式映射进行了实现，并实现了基于模式映射的查询分解[4,5]。随着应用需求的不断发展，多数据库系统需要集成越来越多的数据源，在未来的工作中，我们将对集成文件系统、Web 信息、XML 数据等的模式映射方法进行深入的研究。

参考文献：

[1] A Sheth, J Larson. Federated Database System for Managing Distributed, Heterogeneous, and Autonomous Database [J]. ACM Computing Surveys, 1990, 22(3): 183 - 236.

[2] Shirley A Becker, Rick Gibson, Nancy L Leist. A Study of a Generic Schema for Management of Multidatabase Systems[J]. Journal of Database Management, 1996, 7(4): 14 - 20.

[3] Martin P, Powley W. Database Integration Using Multidatabase Views [A]. Proc Of CASCON '93, IBM Center for Advanced Studies 1993 Conf[C]. 1993. 779 - 788.

[4] Li Bing, Lu Zheng-Ding, Xiao Wei-Jun, et al. An Architecture for Multidatabase Systems Based on CORBA and XML[A]. IEEE Proc of 12th Int'l Workshop on Database and Expert Systems Applications (DEXA '2001)[C]. 2001. 32 - 37.

[5] 卢正鼎，李兵，肖卫军，等. 基于 CORBA/XML 的多数据库系统研究与实现[J]. 计算机研究与发展，2002，9(4)：443 - 449.