CIMS 信息集成 数据库

计算机科学1995Vol.22№.4

31-35

CIMS中的信息集成技术*)

王国仁 于戈 张斌 单吉弟 郑怀远

(东北大学计算机科学与工程系 沈阳110006)

TH166

摘 要 This paper attempts to expound information integration technologies employed by Computer Integrated Manufacturing System(CIMS), mainly discusses serveral key techniques in Information Integration System(IIS), namely, integration architecture, information integration mechanism, the procedure of information integration, integration data model, heterogeneous translation and conflicts of information integration systems.

关键词 Computer Integration Manufacturing System (CIMS), Information integration, Integration data model, Integration mechanism.

一、引 言

计算机集成制造系统(CIMS)是一个非常复杂的过程,将涉及产品的设计、加工制造到产品的销售等一系列活动,各个设计与制造环节既独立工作又相互协作,而且有大量的信息需要进行共享与交换。因此如何组织CIMS环境中的各种数据源,以利于信息的交换与共享是实施CIMS战略目标的关键所在。CIMS环境中其数据成份非常复杂,既有结构化数据,又有大量的非结构化数据,而且CIMS环境中的数据源是相当异构的,既有关系型数据库(如ORACLE, SYBASE 和 INGRES 等),又有非关系型数据库(如IMAGE 和 IMS 等),并且数据库在物理上是分散的。另外,CIMS环境中各个子系统在系统集成前就已经建立了大量的数据库并开发了许多应用系统,各个系统之间的设计是独立进行的,这就容易产生所谓的"数据库孤岛"问题。因此,解决CIMS环境中的信息集成问题以实现各个系统之间的信息共享与交换是非常迫切和必须的。

目前在大多数集成系统中主要采用三类集成体系结构:一是全局模式结构[1-7];二是场地自治的联邦结构[8,9,11,12];三是松散结构[13,15]。集成系统中的集成机制也是一个非常重要的研究问题,就全局模式结构而言主要有三种常用的集成机制:一是源标签机制[1,2,6];二是虚拟机制[3,4,5];三是静态机制[14]。集成模型的设计与选择是集成系统的核心问题,常用的集成模型有四种:一是关系模型,主要用于关系类型的数据库集成[1];二是嵌套关系模型,有利于传统的数据库系统的集成[2];三是语义数据模型,能够集成更多的存在于应用之中的语义,便于应用的集成[5,6];四是面向对象的数据模型,其系统有文[7][14]。由于被集成的模式可能来自各个不同的模式设计者,所以可能存在各类模式冲突问题。模式冲突概括起来主要有两大类:一类是命名冲突,一类是结构冲突。另外,数据之间也可能发生冲突,例如在一个关系中用"IBM"来表示IBM的公司名称,而在另一个关系中用"IBM."来表示IBM的公司名称,完全靠系统自动解决冲突是比较困难的。因此如何解决模式冲突和数据冲突问题在集成系统中非常重要,不提供解决冲突问题的集成系统的集成能力将是很弱的。

本文试图从数据集成的角度来阐述CIMS环境中进行信息集成时产生的各类问题,详细论述了其中几项关键技术,主要包括信息集成的体系结构、集成机制、集成的一般过程、集成数据模型以及信息集成过程中的异构转换和模式冲突等问题。

二、集成体系结构

信息集成系统的体系结构概括起来有如下三种:

*)国家863自动化领域CIMS主题资助项目。

1. 强调全局模式的集成体系结构

在这种体系结构中，除了有全局数据模型以外还必须强调全局数据模式的存在，全局数据模式又分为如下两种：

(1)自上而下地建立全局数据模式。系统自顶向下提供统一的全局视图和数据操纵/定义语言，建立统一的全局字典，由同化器实现从全局数据库到局部数据库的映射。这种系统数据透明性好，用户操作方便，但欲自底向上建立全局模式比较困难，不易提供良好的场地自治性。这种体系结构是一种传统的分布式数据库体系结构，不利于数据库自下而上地进行数据的集成。

(2)自下而上地建立全局集成模式。在这种体系结构中，既可自上而下地建立新的数据模式，也可自下而上地集成已有的数据库模式。系统在自下而上地集成已有的数据库模式时，首先进行模式转换与模式同化，消除因各种局部数据模式之间的差异所带来的影响，解决各种局部模式之间的命名冲突和部分结构冲突。然后在同化后的模式基础上进行模式的集成，其主要手段是模式的合并与重构，以解决部分结构冲突问题。在操作阶段集成系统按照集成模式自顶向下操作各个局部数据库，对来自各低层数据库的异构数据进行异构转换与集成：首先将来自各个局部数据库的局部查询结构转换为符合全局数据模型要求的全局数据表示形式，然后再根据模式的合并与重构结果来进行全局数据结果的集成，以解决相关的数据冲突问题。自底向上的集成体系结构是 CIMS 环境下多数据库(源)集成的一种较好的体系结构。

2. 强调场地自治的联邦体系结构

联邦式数据库系统是在各个成员数据库系统保持局部自治的前提下进行协商合作的数据库系统[12]，它具有三个基本特性：(1)场地自治性。联邦数据库系统的各个成员均独立于联邦系统，各个成员有权决定自己的数据模式，数据操作语言和与其它成员间的协商过程等；(2)不坚持建立全局模式，但可以有全局模型，也可以没有全局模型。如果有全局模型，则各个联邦成员的输入输出模式所体现的数据模型与全局数据模型是一致的。如果没有全局模型，则各个联邦成员的输入输出模式所体现的数据模型是各个联邦成员的局部数据模型；(3)各个联邦成员间通过协商合作来进行数据的交换与共享。

在联邦体系结构中，由各个底层的局部库提供输入和输出模式，通过联邦成员间的协商机制，实现各场地之间的数据共享。其优点是系统组成较灵活，场地的自治性好，缺点是数据透明性较低，协议复杂，所需的转换器和翻译器数量较大，不适合多库共享。

3. 松散结构的集成体系结构

这种集成体系结构通过扩充网络系统提供的功能，在发出请求的源场地和接受请求的目标场地上，分别建立顾客单元和服务器单元，为终端用户提供一定的全局查询功能。松散体系结构有三个基本特点：(1)松散耦合。各个集成单元之间通过网络提供的手段相联系，其间的联系比较松散。每个集成结点由两部分组成：一是客户单元，一是服务器单元。客户单元主要接收来自本地的操作请求，并将处理后的操作请求发送给本地的服务器单元或远程服务器单元。服务器单元主要完成来自本地或远程数据库结点的操作请求；(2)有统一的数据模型和数据语言，但不要求建立统一的全局模式，这一点与联邦体系结构类同；(3)高自治性。各个服务器单元具有很高的局部自治能力，要求集成单元和局部 DBMS 之间的并发控制必须能够协调一致。

这种方法实现上最简单、但数据透明性非常差，用户使用起来很困难。由于篇幅的限制，后面主要讨论自下而上的集成体系结构。

三、集成机制

在自下而上的集成体系结构中，目前主要有以下几种信息集成机制：

1. 源标签集成机制

将要集成数据模式中的各种数据项贴上标签以标识集成模式中的数据来源。优点是在源标签集成机制中能够较好地解决各种冲突问题，模式设计者和终端用户容易理解，便于进行查询优化与查询分解。如文[1][2]采用的就是源标签集成机制。

2. 虚拟集成机制

采用视图集成的基本方法来进行信息集成。优点是消除了数据重复，避免了多副本的修改问题，减少了应用之间的数据不一致性。如文[3][4][5]采用的就是虚拟集成机制。

3. 静态集成机制

源标签和虚拟集成机制都是首先进行模式集

成，然后在对集成模式进行数据操作时再进行数据集成；而静态集成机制则是在模式集成时就进行数据的集成，即数据的集成是静态进行的，这样数据就存在多副本，产生了数据冗余与数据不一致性，但系统实现起来比较简单。如文[14]采用的就是静态集成机制。另外，由IBM公司研制的系统使能器CDF也是采用的静态集成机制。

四、信息集成的过程

一般来讲，一个集成过程应该包括下面四个阶段：(1)预集成阶段；(2)模式比较阶段；(3)模式冲突解决阶段；(4)模式的合并与重构阶段。但是对一个集成系统并不一定要求它完全具有上述四个集成阶段。下面分别讨论各个集成阶段应包含的内容。

1. 预集成。主要要确定集成规则、选择要集成的模式、集成优先次序、被集成的模式数目、全局集成策略等。

2. 模式比较。分析并比较要集成的模式并确定各概念之间的相似之处和各种可能的冲突。在信息集成过程中发生的各类冲突问题归纳起来有三类：一类是命名冲突，包括异物同名和同物异名两种；第二类是结构冲突，主要包括格式冲突、类型冲突、关键字冲突、依赖冲突和行为冲突等；第三类冲突是数据冲突。

3. 解决冲突。主要是解决第2步所确立出的各种冲突。冲突的解决方法取决于系统采用的集成数据模型和施加在系统中的冲突仲裁机制，有的冲突系统能够较好地解决，而有的冲突完全由系统来解决非常困难，必要时需要人工调整。事实上，模式同化与冲突解决是很困难的，其中包含有大量的人为因素，完全做到自动解决冲突在目前来说是不太可能的。

4. 模式合并与重构。建立最后的集成模式，有二元法和多元法两种途径。二元法是每次集成两个模式，像爬梯子一样，逐次得到全局模式。多元法是一次可合并两个以上的模式，这样，可一次建成集成模式，也可分为多步完成。

模式合并与重构的原则是：①完备与正确原则。集成模式必须完全正确地包含出现在所有被集成模式中的所有概念。②最小原则。如果同一概念出现在不同的组成模式中，则这一概念在集成模式中仅能出现一次。③易理解原则。集成模式对于设计者和终端用户来说应该易于理解，即在几种可能的集成结果表示中应该选择最容易理解的一种。

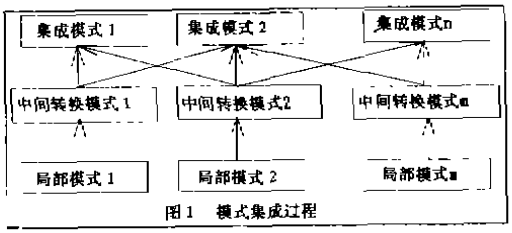

图1 模式集成过程

模式集成的过程如图1所示。在模式转换阶段，需要将被集成的m个局部模式一一转换为中间转换模式，支持中间转换模式的数据模型为全局集成数据模型。在这一阶段要选择被集成的局部数据模式，确定集成策略，同时还要进行局部模式之间的比较与分析，找出集成模式之间各类冲突问题，而且一部分冲突问题(如命名冲突和部分结构冲突)应该在这一阶段加以解决。

在模式集成阶段主要是进行模式的合并与重构，生成所需要的全局集成模式。集成模式可以一步生成，也可以分步生成。另外，部分结构冲突问题应该在这一阶段加以解决。

五、信息集成模型

目前在集成系统中最常用的数据模型有四种：面向对象的数据模型，语义数据模型，关系数据模型和嵌套关系模型。

1. 面向对象的数据模型

这种数据模型的模型化能力非常强，具有丰富的语义表达能力，其模型结构为有向图。因此采用面向对象的数据模型作为集成数据模型便于集成非传统数据库系统的数据源，同时也有利于集成存在于原有模式之间的语义，但系统的实现相对要复杂一些。

2. 非一范式的嵌套关系模型

非一范式是一个嵌套关系模型，一个关系属性既可以是简单属性，也可以是另外一个关系，即形成关系嵌套。非一范式的基本模型结构是树结构，因此采用嵌套关系模型来集成传统的数据库系统(尤其是关系数据库系统和层次数据库系统)较为方便，而且系统的实现相对简单。

3. OSAM * 模型及语义数据模型

用语义数据模型作为集成数据模型的集成系统有很多[5,6,7]，另外IBM公司研制的系统使能器CDF所采用的数据模型也是一个语义数据模型OERA，文[6]采用的是扩展的ER模型，文[5][7]采用的是面向对象和语义关联的数据模型OSAM *。

OSAM *模型是一个面向对象和语义关联的数据模型，它具有如下特征：①支持O-O的部分特征，如对象标识(符)，超类/子类联系，属性继承等；②支持丰富的语义特征。在OSAM *数据模型中，共支持六种语义关联：聚合(A)关联，概括(G)关联，互联(I)关联，组合(C)关联，叉积(X)关联，累计(S)关联。这几种关联基本上可以表示工程应用中的所有语义。③支持复杂对象的描述。通过A关联层次可以表示和描述一些非常复杂的对象。④支持递归结构的定义。当一个实体类的A关联属性的某个组成对象类是这个实体类本身时，就形成了递归结构。⑤支持复杂数据类型的表示。在OSAM *模型中，除了支持简单的数据类以外，还支持复杂数据类型，如集合(SET)、有序集(Ordered SET)、向量(Vertor)、矩阵(Matrix)。

采用这类模型作集成数据模型的考虑是语义数据模型能够方便地集成存在于原有系统之间的语义，有利于模式之间冲突问题的解决。

4. 关系数据模型

文[1]采用的集成数据模型是关系数据模型加上源标签。关系数据模型作为集成数据模型的好处是便于集成各种关系型数据库系统，系统的实现非常简单，但它的缺点是，其模型能力非常弱，不能很好地集成非关系数据源。

还有许多系统以上述四种模型的综合模型(如非一范式模型与面向对象的数据模型相结合)作为集成数据模型。

六、转换与冲突问题

1. 异构转换

在信息集成过程中，必然需要进行各种异构转换以实现信息的集成与共享。总的来讲异构转换可按以下层次来进行：

(1)数据模型转换。主要研究被集成的各种数据库系统所支持的数据模型表达和描述的各种模型概念和结构到集成系统所支持的数据模型的转换问题。被集成的数据模型的概念和结构必须能等价地转换成集成数据模型的概念和结构，集成模型所支持的概念、结构和行为(包括语义)必须完全包容被集成系统所支持的各种概念、结构和行为(包括语义)。

(2)数据模式转换。实际上是自下而上进行的，这是信息(数据)集成的初步，这一步主要用一种集成机制和相应手段(包括模式集成语言)将各个被集成的子系统中希望参加集成的各种数据模式等价地转换为一个或多个集成数据模式。从局部数据模式到集成数据模式的转换为信息(数据)的共享与交换打下了基础。

(3)数据语言转换。与数据模式的转换过程相反，是自上而下的。当完成了数据模式的转换以后便产生了全局数据集成模式，用户可以通过集成系统提供的数据操作语言来共享集成数据。在这个过程中集成系统应该进行数据语言的转换，即应将用户发来的操作请求(操作语言)根据集成数据模式进行转换和分解，等价地转换为与被操作的集成模式相关的若干种局部操作语句的集合。数据语言的转换是实现数据(信息)交换与共享的主要步骤。

(4)数据表达转换。是实现信息(数据)共享的终结步骤。在这个过程中，各个局部操作语句的集合被提交给各个子(分)系统后就可以得到以各局部模型概念为基础所表示的若干个局部数据的集合。这时必须根据与该操作请求相关的集成数据模式信息对各局部数据集合进行转换和综合，以得到一个最终的与全局集成模式相一致的数据集合。

2. 冲突问题及相应的解决策略

在信息集成过程中可能会遇到各类模式冲突问题，主要有以下几种：

(1)命名冲突。在不同的模式中用相同的名字来表示不同的概念及同一概念在不同的模式中用不同的名字来命名；命名冲突的解决比较容易，一般的解决策略是提供一条换名语句或采用源标签集成机制。

(2)格式冲突。主要表现在数据类型、定义域、量纲和精度的不一致性。当两个数据项的数据类型或定义域发生冲突时其解决冲突的方法与类型冲突的解决方法大致类同；当两个数据项的量纲发生冲突时一般是提供一个量纲转换公式；当两个数据项的精度不一致时将低精度数据项的精度提高至与高精

度数据项一样高的精度。

(3)数据冲突。是指描述同一现实世界对象的数据在实际保存的值上也会发生冲突。数据冲突的一种情况是两个数据库中有关同一数据项的取值不同。例如在一个关系中用"IBM"表示 IBM 公司的名称,而在另一个关系中用"IBM."来表示。这类冲突问题的解决是比较困难的,一般有两种方法:一是由人工解决;一是采用建立同义名的方法来解决数据间的冲突问题。

(4)类型冲突。对于同一种对象,可能用不同的数据结构表示。例如,一个实体在一个数据库中可能表示成一个属性,而在另一个数据库中表示成一个关系。一个数据项在一个数据库中是单值的,而在另一个数据库中是多值的。当两个模式发生类型冲突时,一般的解决方法是将这两个模式的共同部分概括为一个共同的超类,而将不同的部分(包括冲突部分)保留在子类当中。

(5)关键字冲突。在不同的模式中用不同的关键字来标识同一概念。例如 SS# 和 Emp_id 分别是两个组成模式中 Employee 的关键字。

(6)依赖冲突。同一组概念在不同的模式中有不同的依赖关系。例如,在 Man 和 Woman 之间联系 Marriage 在一个模式中是1:1,而在另一个婚姻历史模式中可能是 m:n。依赖冲突的解决办法是将低级别的依赖关系转换为高级别的依赖关系,其转化的顺序为:1:1-->1:N-->N:M。

(7)行为冲突。当对不同模式中的同一对象采用不同的插入/删除策略时发生行为冲突。例如在一个模式中允许没有雇员的部门存在,而在另一个模式中,当删除某个部门的最后一个雇员时将导致该部门的删除。不过这种行为冲突仅当数据模型能够表示对象行为特征时才可能发生。行为冲突的一般解决办法是通过行为的提炼来解决行为冲突问题,这就要求系统的数据模型应该有操作或行为的提炼功能。

参考文献

[1] Wang Y. R. et al., A Polygen Model for Heterogeneous Database Systems: The Source Tagging Perspective, Proc. of the 16th Intl. Conf. on VLDB, Brisbane, Austrilia, 1990

[2] 王国仁、于戈等,CIMS 环境下多数据库的集成技术,第二届中国计算机集成制造系统学术会议,深圳,1992

[3] Kaul M. et al., View System: Integration Heterogeneous Information Bases by Object-Oriented Views, IEEE, 1990

[4] Day Y. et al., View definition and generalization for database integration in multidatabase system, IEEE Trans. on Software Enigeering, SE-10(6), 1984

[5] 石晶,郑怀远,一个基于 CIMBASE 的异构数据库集成系统,第十二届全国数据库学术会议论文集,武汉,1994

[6] 王国仁,郑怀远,基于 EER 数据库集成方法的研究,《计算机研究与发展》,1993.12

[7] 王国仁等,面向对象和关系数据库系统的集成方法,第九届全国数据库学术会议论文,上海,1990.9

[8] Elmasri R. et al., Schema Integration algorithms for federated database and logical database design, Submitted for publication (1987)

[9] Heimbigner D. et al., A Federated architecture for information management, ACM Trans. on Office Information Systems, 3(3), 1985

[10] Deen S. M. et al., Data Integrated in distributed databases, IEEE Trans. on Software Engineering, SE-13(7), 1987

[11] 孙志挥等,SU-FDBS--一个联邦数据库系统原型的设计与实现,同[2]

[12] 孙志挥等,一种解决联邦数据库系统嵌套查询的处理方法,同[5]

[13] Wolski A., LINDA: A System for Loosely Integrated Database, IEEE Computer, 22(5), 1988

[14] 于戈等,一个 CIMS 信息集成平台系统的设计,同[5]

[15] 刘烽,郑怀远,松散结构的异构数据库集成技术的研究,第十届全国数据库学术会议论文集,沈阳,1992