一个基于CORBA的异构数据源集成系统的设计*

王 宁 陈 瀅 俞本权 徐宏炳 王能斌

(东南大学计算机系 南京 210096)

摘要 提出一个基于CORBA(common object request broker architecture)的即插即用的异构多数据源集成系统的设计方案.由于采用具有较强描述能力的OIM(object model for integration)对象模型作为集成系统的公共数据模型,该系统不仅能集成各种异构数据源,包括数据库系统、文件系统、WWW上HTML文件中的数据,而且能集成随时插入的新数据源中的数据.着重讨论系统的总体结构、OIM对象模型、查询处理及界面设计.

关键词 异构数据源,数据集成,半结构化数据,互操作性,对象模型,查询语言.

中图法分类号 TP311

目前,随着通信网络和计算机网络的普及,数据资源的共享已经成为一个热门话题.传统的数据集成技术,例如多库方法,已无法适应人们获取更多更新数据的需要.人们要求数据集成系统不仅能集成数据库系统中的数据,而且能集成非数据库系统中的数据;不仅能集成传统数据,而且能集成多媒体数据;不仅能集成已有数据源中的数据,而且能集成随时加入的新数据源中的数据.也就是说,数据集成系统必须具有可扩展性,可以实现数据源的"即插即用".这是传统的数据集成技术难以实现的.

以往一些数据库产品用转换器(Gateway)实现互操作,例如Oracle 7的Dedicated Transparent Gateway和Sybase的OmniSQL Gateway,但转换器只能连接两个数据库系统,局限性较大.目前研究较多的是多库系统.[1,2]多数据库的集成问题,早在70年代中期即被提出.开始采用全局模式的集成方法,后来Mcleod等人提出了联邦式数据库系统的概念.由于缺乏必要的标准,联邦数据库系统只能在一定的限制条件(如对加入联邦系统的DBMS和各DBMS间的互操作加一定的限制)下实现,难以实现各种数据源的灵活的数据集成.因此,联邦式数据库目前还不能成为一种通用的数据集成方法.

面向对象技术的发展为异构数据源的集成提供了新的途径.对象管理集团OMG于1991年提出了一个对象管理结构的基准结构OMA(object management architecture),包括描述互操作机制的CORBA(common object request broker architecture)[3,4]以及对象服务规范COSS(common object services specification).在CORBA系统中,所有的应用程序都封装成对象,其界面定义了对象可提供的操作,客户方只需知道目标对象及其界面,就可获得目标对象提供的服务.CORBA的核心部分是ORB(object request broker),它是对象访问的中介,对象间的任何访问都要通过ORB.ORB的动态激活界面DII(dynamic invocation interface)使得客户不用修改程序就可以访问所需对象,并接受它提供的服务.从某种意义上说,CORBA提供了一个集成框架,应用程序只要给出用界面定义语言IDL(interface definition language)书写的界面,就可插入框架,与其他对象互操作,为实现数据源的"即插即用"式集成提供了可能.

Galaxy是作者研制的一个基于CORBA的分布式异构数据源集成系统.与多库系统(如Multibase[5],MIND[6],IRO-DB[7]等)相比,Galaxy除可以集成DBMS管理的数据外,还可以集成非数据库系统管理的数据,例如文件系统、WWW上的数据.与其他异构数据源集成系统(如Galic[8],OLE DB[9])相比,它最大的特点在于数据源可以"即插即用".新的数据源加入时,只要经过一定的包装,就可插入系统,无需修改Galaxy系统程序.

目前,Galaxy正对微软公司的SQL Server、自行研制的面向对象数据库系统FOOD(friendly object-oriented database system)、文件系统、超文本数据(即WWW中的数据)进行包装和集成.通过Galaxy的可视化界面,用户可以

* 本文研究得到国家自然科学基金资助.作者王宁,女,1967年生,博士生,讲师,主要研究领域为数据库和分布对象技术.陈瀅,1973年生,博士生,主要研究领域为数据库和计算机网络.俞本权,1973年生,硕士生,主要研究领域为数据库和分布对象技术.徐宏炳,1947年生,副教授,主要研究领域为数据库应用.王能斌,1929年生,教授,博士导师,主要研究领域为数据库及信息系统.

本文通讯联系人:王宁,南京 210096,东南大学计算机系

本文1997-04-08收到原稿,1997-06-09收到修改稿

任意存取来自上述各数据源的数据,并可根据需要加入新数据源.有关HTML文件的集成以及"即插即用"的集成方式尚未见文献报道.

本文第1节给出Galaxy系统总体结构,第2节提出OIM对象模型作为Galaxy系统的公共数据模型,第3节介绍Galaxy的查询处理,第4节介绍Galaxy的界面设计,最后是结束语.

1 Galaxy体系结构

图1描述了Galaxy系统的结构,Galaxy系统由可视化界面GUI、查询服务与系统集成QI、元数据仓库、各数据源及其包装器构成.Galaxy将各个部件封装成对象,依次插在ORB这根"软件总线"上.

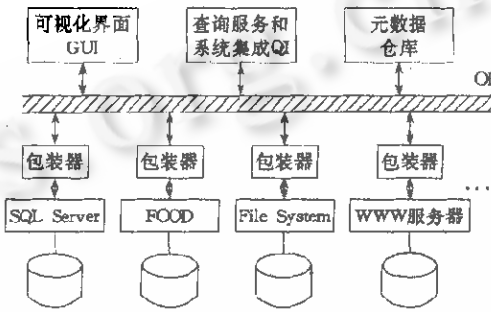

图1　Galaxy总体结构

Galaxy系统采用OIM (object model for integration)对象模型作为数据集成的公共模型.该模型的每个数据项都含有描述符,因而能描述来自各种数据源的数据,包括数据库系统中的数据、文件系统中的数据、多媒体数据、WWW上HTML文件中的数据.灵活的OIM对象模型以及ORB的支持,使Galaxy系统的可扩展性得到了充分保证.

Galaxy的每个数据源都配有包装器,包装器将各数据源包装成对象.例如,SQL Server管理的关系数据库经包装器包装后可作为ORB的一个数据源对象.该对象的界面定义了对象可提供的服务,由统一的IDL语言描述.而对象服务的具体实现则由包装器完成.包装器将来自客户方的用Galaxy对象集成查询语言OIQL(object integration query language)表达的查询转换成SQL语句,交给SQL Server执行,再将返回的表格式数据转换成OIM对象模型表示的对象,以便与来自其他数据源的对象集成.任何数据源,只要配备合适的包装器,并向Galaxy提供用IDL语言写成的界面,就可加入系统.

数据集成时生成的对象定义、视图定义以及安全性管理所需的授权等信息由元数据仓库管理.

Galaxy的核心部分是查询服务与系统集成部件QI.QI为不同用户提供不同的对象视图.用户使用Galaxy查询语言OIQL可以对视图中的对象进行各种查询.

GUI是Galaxy的可视化用户界面,它不仅能浏览对象及其子对象之间的关系,而且集浏览和查询于一体,使用户不必另外书写查询语句.

2 OIM对象模型

Galaxy是一个可扩展的异构数据源集成系统,它面向各种各样的数据源.各种数据源都有自身的特点,除数据模型不同外,有些数据源没有一个稳定的模式,例如生物学方面的一些数据,由于实验技术的快速发展,数据模式需要经常调整.在CAD中,数据模式随着设计的进展而不断地发展、修改.有些数据源包含一些非结构化的和半结构化的数据,例如图象等.这些数据本身除了可具有少数属性外,很难用数据模式来详细描述.另外,还有些数据源包含自描述的数据[5],例如WWW上的HTML文件等.要在一个系统中集成上述各种数据源,是传统的数据集成技术难以胜任的.

传统的数据集成技术主要通过"模式集成",为用户提供统一的数据模式.这种方法常用于多库系统,因为数据模式是数据库的一个重要组成部分.而非数据库系统管理的数据,数据模式往往隐含于数据类型说明中,一般无显式的数据模式.因而,对于多数据源的集成,需要另辟蹊径.

Microsoft公司利用DBMS部件化[10]的思想,正在研制OLE DB.OLE DB旨在为存储于数据库系统及非数据库系统中的数据提供统一的存取界面.它将表格式数据(包括关系数据)看成是"行集"(Rowset)对象.但"行集"是针对表格式数据提出的,不适宜于表示非表格式的或自描述的数据.

Galaxy系统采用对象集成技术,把每个需集成的数据单位,例如关系数据库的一张表、文件系统的一个文件、WWW上的一个结点均看成对象.系统采用OIM对象模型作为数据集成的公共模型.在这种对象模型中,每个对象由四元组(OID,d,t,v)表示,其中OID表示对象标识符,d表示对象名,t表示对象类型,v表示对象值.t除了可表示基本数据类型(如integer,char,float,string等)外,还可表示集合数据类型(如set,list,bag等)、可变长数据类型(text,

BLOB)和引用类型(ref).

例 1:关系数据库中的表 teacher(如图 2 所示)可用 OIM 对象模型表示成如图 3 所示的结构.

name	sex	age
'Wang'	'f'	25
'Chang'	'm'	48

图 2 teacher 表

⟨&O_1,teacher,ref,{&O_{11},&O_{12}}⟩
⟨&O_{11},tuple,ref,{&O_{111},&O_{112},&O_{113}}⟩
⟨&O_{111},name,string,'Wang'⟩
⟨&O_{112},sex,string,'f'⟩
⟨&O_{113},age,integer,25⟩
⟨&O_{12},tuple,ref,{&O_{121},&O_{122},&O_{123}}⟩
⟨&O_{121},name,string,'Chang'⟩
⟨&O_{122},sex,string,'m'⟩
⟨&O_{123},age,integer,48⟩

图 3 teacher 表的对象结构

图 3 中,第 1 个四元组表示了 teacher 表对象.它是引用(ref)类型的,引用名为 tuple 的两个子对象(即 teacher 表的两个元组),每个 tuple 对象又引用名为 name,sex,age 的 3 个子对象(即 teacher 表的 3 个属性).本文以下用"&"开始的字母数字串表示对象标识符,用缩进格式表示对象之间的引用关系.当表结构转换成对象结构后,模式信息就成为对象的一部分.在包装有模式结构的数据源时,可以利用模式进行适当的数据压缩.但概念上,OIM 对象模型的每个对象都含有描述符,以便表示来自异构数据源的数据,尤其是自描述的数据.

例 2:设 WWW 上有一个结点 http://www.seu.edu.cn,若需检索的 HTML 文件结构简单表示成如图 4 所示的有向图,有向图中的边表示超文本链,结点表示超文本内容.该结构描述结点 //www.seu.edu.cn/pub/papers 中有一引向 database 的超文本链,database 结点有两条引向 data-mining 和 data-warehouse 的超文本链,用 OIM 对象模型可表示成如图 5 所示的对象结构.

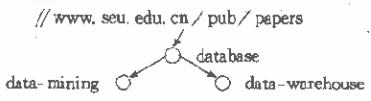

图4 HTML文件的链结构

⟨&O_2,//www.seu.edu.cn/pub/papers,ref,{&O_{21}}⟩
⟨&O_{21},database,ref,{&O_{211},&O_{212}}⟩
⟨&O_{211},data-mining,text,(text description)⟩
⟨&O_{212},data-warehouse,text,(text description)⟩

图 5 超文本链的对象结构

图 4 仅是一个极其简单的 HTML 文件结构,真正的 HTML 文件不仅有复杂的内容,而且超文本链可能循环引用或交叉引用.由于 OIM 对象模型具有自描述的特性,它完全可以描述这样的数据.

OIM 对象模型为异构数据源集成而设计,它与传统的 O-O 模型[11]相比,具有以下特点:

(1) OIM 对象模型具有自描述的特性,它的每个对象都含有描述符,因而特别适合于描述那些没有显式模式或者模式无法预知的数据对象.

(2) OIM 对象模型支持对象标识和对象之间的引用关系,却并不强调分类.这一特点使得集成系统能够方便地处理来自各数据源的异构数据.事实上,传统 O-O 模型支持的分类在 OIM 对象模型中通过对象之间的引用关系也是可以实现的.

(3) OIM 对象模型是集成系统的公共数据模型,它仅在逻辑上统一地表示来自各数据源的异构数据,至于对象在各数据源如何存储,与 OIM 对象模型无关.

OIM 对象模型是一种极其灵活的数据模型,它可以方便地描述各种数据源中的数据,尤其是自描述的数据,这是其他数据模型无法比拟的.在这种模型中,需集成的数据都抽象成对象,对象集成方式代替了传统的数据集成方式,便于集成那些非结构化的、半结构化的以及自描述的数据.

3 Galaxy 的查询处理

Galaxy 的对象集成查询语言 OIQL 是 SQL(structured query language)语言的扩充.由于 Galaxy 系统的公共数据模型是一种对象模型,OIQL 在 SQL 语言的基础上增加了一些构造符,以便完成对象之间导航式查询以及对一些集合类型对象的查询.

Galaxy 系统需要集成一些没有固定模式或模式无法预知的数据源的数据,如 WWW 上的 HTML 文件,一般人很难了解其超文本链的详细结构.传统的面向对象的查询语言,需要描述完整的搜索路径,显然并不适合超文本数据的搜索.为此,Galaxy 借用了文件系统中"通配符"的表示方式,将通用路径与准确路径结合起来,大大提高了 OIQL 的

表达能力.

例3:为在图5所示超文本链的对象结构中查找所有data-mining文件,OIQL提供3种表达方式.

(1) 准确路径表达方式

```
select  //www.seu.edu.cn/pub/papers:database:data-mining
from    //www.seu.edu.cn/pub/papers
```

用":"分隔的一串对象名为路径表达式.为清楚起见,本例采用全路径表达方式.在语义明确的情况下,select后路径表达式的第1个对象名可省去.

(2) '?'通用路径表达方式

```
select  //www.seu.edu.cn/pub/papers:?:data-mining
from    //www.seu.edu.cn/pub/papers
```

路径表达式中,"?"可以匹配任一个对象名.该语句表示,从//www.seu.edu.cn/pub/papers对象的任一个子对象α中,查找α的对象名为data-mining的子对象.

(3) '*'通用路径表达方式

```
select  //www.seu.edu.cn/pub/papers:*:data-mining
from    //www.seu.edu.cn/pub/papers
```

路径表达式中,"*"可以匹配任意零到多个对象名.该语句表示,从//www.seu.edu.cn/pub/papers对象开始,沿任意路径寻找data-mining对象.

当HTML文件具有非常复杂的链结构时,通用路径为用户提供了极大的方便,用户可以在对象结构无法预知的情况下搜索超文本数据.

上例(2)(3)两条OIQL语句在执行时,一旦遇到data-mining对象,搜索立即停止.当用户要在一个复杂的HTML文件链结构中查找所有的data-mining对象时,可以在"select"的后面加上关键词"all",这样,仅当搜索完所有路径后,查找才会停止.

OIQL的"通用路径"表示方法给用户带来了很大的方便,然而,从搜索效率考虑,用户也不能滥用通用路径,应遵循"先准确路径,再'?'通用路径,后'*'通用路径"的原则,尽量避免盲目搜索.

顺便指出,对网上结点的HTML文件链结构的包装以及通用路径的搜索都是基于我们正在研制的WWW数据管理系统(WWWDMS),WWWDMS为WWW建立统一的数据模型和查询语言,将WWW由用户手工式"导航"变为基于内容的导航方式,并提供与其他系统集成的接口.有关WWWDMS的详细情况,将另文介绍.

用户通过GUI发出查询,Galaxy的查询服务与系统集成部件,即QI,将查询分解成若干子查询,发送到相应数据源,并形成一定的执行计划.各数据源的包装器负责将这些子查询映射成局部数据源认可的查询语言.有些检索能力较弱的数据源,例如文件系统、WWW等,需要功能较强的包装器.为了减轻包装数据源所需的工作量,Galaxy只要求各数据源在包装界面上描述其检索能力,系统自动根据各数据源的检索能力安排执行计划.因此,一个数据源在作一定程度的包装并描述其查询能力后,就可轻松地加入Galaxy系统.

灵活的数据模型和查询模型,为Galaxy的可扩展性以及"即插即用"的实现奠定了基础.

4 Galaxy的界面设计

界面是人机交互的接口,对于Galaxy这样一个异构数据源集成系统而言,方便、统一的界面尤为重要.

由于Galaxy采用OIM对象模型作为公共数据模型,因而它的GUI必须能处理复杂的对象结构,方便用户基于复杂结构的查询.具体地说,GUI应能支持路径表达式、集合类型对象的查询以及用户指定的联想查询.

传统的图形化查询工具面向具有显式模式的数据库,通常将查询和浏览分开.然而,Galaxy是一个异构数据源集成系统,它不仅能集成数据库系统中具有一定模式的数据,而且能集成非结构化、半结构化以及自描述的数据.由于要集成的对象的结构很难事先确定,因而需要将浏览和查询功能融为一体,边浏览边查询.

5 结束语

Galaxy是一个基于CORBA的可扩展的分布式数据集成系统.Galaxy的对象模型具有很强的描述能力,因而能同时集成来自多个异构数据源中的数据,新的数据源只要经过包装并描述其检索能力后,就可插入Galaxy系统.Galaxy系统的对象集成查询语言OIQL不仅能完成对象之间导航式查询以及对一些集合类型对象的查询,而且采用

通用路径与准确路径相结合的表示方法,方便用户对超文本数据的搜索.为了适应异构数据源集成的需要,Galaxy还设计了浏览、查询于一体的可视化界面,便于用户进行各种操作.

Galaxy是我们探索通用数据集成途径的一个尝试.目前我们在IONA公司的OrbixWeb产品上,对关系数据库、面向对象数据库、文件系统、HTML文件的集成进行了实验研究,初步结果表明,这种集成方法是可行的,是有希望的.要使这种方法成为实用技术,还要进行更大规模的试验,并研制相应的配套工具,方便包装器的书写或自动生成.

参考文献

1 Bright M W *et al*. A taxonomy and current issues in multidatabase systems. IEEE Computer, 1992, 25(3): 50～59

2 Amit P Sheth. Federated database systems for managing distributed, heterogeneous, and autonomous databases. ACM Computing Surveys, 1990, 22(8): 182～236

3 Jon siegel. CORBA fundamentals and programming. New York: John Wiley & Sons, Inc., 1996

4 Randy Otte, Paul Patrik, Mark Roy. Understanding CORBA. Prentice Hall, 1996

5 John Grant *et al*. Query languages for relational multidatabases. The VLDB Journal, 1993, 2(2): 153～171

6 Dogac A *et al*. METU interoprable database system. SIGMOD RECORD, 1995, 24(3): 56～61

7 Georges Gardarin *et al*. Calibrating the query optimizer cost model of IRO-DB, an object-oriented federated database system. In: Vijayaraman T M *et al* eds. Proceedings of the 22nd VLDB Conference. Mumbai (Bombay), India. San Fransisco, USA: Morgan Kaufmann Publishers, Inc., 1996. 378～389

8 Carey M J *et al*. Towards heterogeneous multimedia information systems: the garlic approach. RIDE-DOM'95, Taiwan, 1995

9 José A Blakeley. Data access for masses through OLE DB. In: Jagadish H V, Inderpal Singh Mumick eds. Proceedings of the ACM SIGMOD Conference'96. Montreal, Canada, New York: the Association for Computing Machinery, Inc., 1996. 161～172

10 Batory D S *et al*. Genesis: an extensible database management system. IEEE Transactions on Software Engineering, 1988, 14(11): 1711～1730

11 Cattell R G G. The object database standard: ODMG-93. San Mateo, California: Morgan Kaufmann Publishers, 1994. 11～43

Design of a Heterogeneous Data Integration System Based on CORBA

WANG Ning CHEN Ying YU Ben-quan XU Hong-bing WANG Neng-bin

(*Department of Computer Science Southeast University Nanjing* 210096)

Abstract A scheme, designed for integration of heterogeneous data sources based on CORBA (common object request broker architecture) in a plug & play way, is put forward in this paper. An object model, named OIM (object model for integration), is proposed as the common data model for integration. The system can integrate various heterogeneous data sources, ranging from database systems, file systems, to hypertext files in World-Wide-Web. Any other data source, if wrapped according to the specified way can be integrated into the system, whenever necessary. The architecture, OIM, query processing, user interface design are introduced in some detail in the paper.

Key words Heterogeneous data sources, data integration, semistructured data, interoperability, object model, query language.