

●陈 远 周朴雄

数据管理性能的折中方案 ——后关系型数据库

摘 要 全面分析了关系型数据库和面向对象型数据库各自的长处和局限,阐明基于两者结合与网络环境的后关系型数据库是数据管理性能的一种折中方案,并谈到了有关后关系型数据库的研究现状和热点课题。参考文献7。

关键词 后关系型数据库 关系型数据库 面向对象型的数据库 数据库模式

分类号 G250.74

ABSTRACT In this paper, the authors thoroughly analyze the advantages and shortcomings of relational database system and object-oriented database system, think that the post-relational database system based on the combination of these two types and the Web environment form the compromise proposal of data management function. Then, the authors discuss the present status and focus in the study of this field. 7 refs.

KEY WORDS Post-relational database system. Relational database system. Object-oriented database system. Database schema.

CLASS NUMBER G250.74

数据库技术从60年代出现以来,支持的数据模型由层次型、网状型、关系型到面向对象型,数据的分布由一台计算机上的集中式发展到计算机网络上场地自治的分布式。几十年来,关系模型和面向对象模型一直是数据库系统研究的热点,研究成果丰硕,产品应用也最广泛。尽管关系模型先起于对象模型,但关系模型一直没有退出IT舞台,在目前兴起的后关系型数据库时代中,它们都找到了自己的位置。这两种模型具有许多互补的特性,也有一些共同的困境。

1 成熟而“固化”的关系型数据库系统

关系数据库模型(Relational Database)是1970年由IBM公司的E. F. Codd首先提出的。80年代,其实用的系统开始流行,1986年被美国国家标准协会(ANSI)和国际标准化组织(ISO)确定结构化查询语言(SQL)标准。关系型数据库发展到现在,技术已相当成熟,拥有大量的用户,商业化产品非常丰富,它自身的特点显而易见。

1.1 关系型数据库的长处

(1)数据模型简单。关系模型在实体和联系之间不作区别,用关系(二维表)描述实体,用带有叫做外

关键字属性的关系(同样是二维表)来表示联系。通过关系运算的操作,属性之间的联系特性就可被主动挖掘出来,也可用一个带相关属性关键字的中介关系来表示属性之间的联系。

(2)数据独立性高。表示关系的二维表中的每一列(属性)都有不同的命名,每列中的数据类型相同,表中的属性都是一维的基本数据类型,即每个属性值都是原子的,不可再细分,表中没有完全相同的行(元组)。

(3)理论基础严格。关系数据库基于数学的关系理论。关系是笛卡尔乘积的子集,有严格的数学定义。1971年Codd提出关系的规范化理论,其规范化形式从1NF, 2NF, 3NF,一直演化到BCNF, 4NF, 5NF。

(4)操作方便灵活。关系数据库的操作基于集合运算与谓词演算,非常灵活。在数据库关系上执行的所有操作最终也形成一个由关系组合派生出来的新关系,以表的形式给用户提供服务。

(5)管理安全性好。关系数据库管理系统对每次数据的访问都有中介,这种狭窄的接口有助于安全。

(6)对用户要求不高。关系模型简洁、易理解、灵

活性很强且容易使用,可被非数据库专业人士使用,导致它被广泛地应用和接受。

1.2 关系型数据库的缺陷

关系数据库以数据表示的单一性、数据操作的统一性、数据语言的非过程化及面向集合的存取方式等特点得以长期流行,但它对因特网网络环境下复杂并多样化的应用日渐暴露出许多不适应性。

(1) 关系模型只能处理整数、实数、字符串等这样简单的结构化数据类型。对于一本书、一张图片等非结构化数据,或包、队列之类本身带有插入或从中取出不同类型数据项操作的复杂数据,或者因特网上具有一定描述层次与结构的页面这类半结构化数据的表达,以及信息与组织形式都不一样,且本身具有自描述性和动态可变性的各网站之间的异构型数据的集成之类的问题,显得无能为力。

(2) 关系模型不支持复杂的数据嵌套。每个表项中只能存放一个不可分的数据值,不能通过指针链接。关系模型不可能捕获像事务处理应用中典型的多维数据这种复杂对象的语义。为了模拟复杂对象,常常需要把信息分开成几个不同的表,表之间的连接常隐藏在应用程序里,而不是在更易于管理的数据库中,每次必须连接一批表才能得到对象的信息,存取速度变慢,且这种方式也很难模仿出数据的现实关系。

(3) 关系型数据库在其性能及其伸缩性方面有很大的限制。它只能进行短小的处理,不能进行要求操作临时变量、历史记录和数据版本的长处理,适合于保存记录的应用和报表输出的设计。关系数据库只处理被动的数据,只能通过读和写传递信息,对因特网上巨大、分布、异构的数据,依靠原有的静态结构化查询语言很难有效地得到数据,查询效率低,编程工作量大且很难优化,引入新的存储结构、索引结构、数据的缓冲、并发控制和恢复机制及算法比较困难。

1.3 关系型数据库自身已没有改进余地

关系模型中对表示实体之间的联系、层次和不同实体具有相同属性值的问题,是用额外表和重复列的方法解决的,对部分属性值相同的元组必须重复存贮,更复杂的数据表示需要在关系模型内部作大量的调整才能支持。这样,增加了额外的冗余、空间和访问开销,使数据库变得笨重与缓慢。

在关系型及一些传统数据库语言中,,基本操作是不能被修改或改变的,每个操作都需要有一个对应的程序来完成,任何扩充的功能都必须来自用户自己

编写的代码,从数据库中抽取必要的信息来完成扩展的有关功能。这种有限的功能主要是由语言的结构和基本设计决定的。

关系数据库是被动的,即数据对数据库周围的条件或数据库内部的值不进行主动反应,数据和数据库的状态只能被改变、查看,被数据操纵原语操作,现在则要求数据库成为一种更完整、更主动的应用环境。

仅靠关系数据库自身的调节不能解决所有问题,而且硬件与软件的成本代价过高。

2 弹性和“自由化”的面向对象型数据库系统

80年代初就引起人们广泛关注的面向对象程序设计语言与随之发展起来的面向对象软件设计方法是一种公认的成功技术。在关系数据库面临一些困境时,人们发现对象技术的特性正好能解决数据库面临的某些问题。

2.1 面向对象型数据库的优势

面向对象数据库(Object-Oriented Database)是面向对象设计思想与数据库技术相结合的产物。面向对象数据库系统扩展了面向对象编程语言的基本特性。

(1) 可以存贮复杂结构的数据。例如有关地图之类的抽象数据类型在计算上是不完备的,与关系模型中有规则的结构本身就不相容。面向对象数据模型以客观自然方式表现客观世界中的各种实体及其联系。它既可表示格式也可表示非规则数据(声、图、形)。面向对象模型的“聚合”与“概括”概念可更好地处理非规则数据复杂的结构语义。

(2) 可以存贮可变长的数据。当把所有的异构数据库转换成关系数据库时,所有可变长度的记录都变成固定长度的元组,不足部分用空值填补。空值占存贮空间,不能计算,也不能表明它所占据位置的值的特性,且要专门写算法对它进行管理。面向对象数据库模型中常见的集合类型是数组,由一组一维的可变长的类组成,数组大小可修改。数组允许把对象以索引列表的形式进行组织和访问,这样,允许数据库系统针对特定的对象实例构造逆序的索引值,或用作对象集合的索引,数组把位置索引值作为访问指针来依次进行访问。

(3) 可以处理具有相同值的不同对象。与其他数据模型不同,面向对象环境中的每个对象实例都赋予一个惟一的标识符。作为对象的标识,就像内存中某个数据项的指针一样,这样就可以让两个不同对象具

有相同的属性值。而关系模型中两个元组具有相同的值是不可能的事,即便它们表示的是现实世界中不同的实体。

(4) 相同的属性值不必重复存贮。面向对象数据库的引入目的之一是为了满足一再出现的复杂信息的共享。对象标识的概念可用来对数据存贮和访问进行优化,相同的属性值不必重复存贮,之间的关联也很容易通过标识(指针)表明,易于表达继承性。面向对象数据库可以提供对象的连续存贮及维护。对象、类、层次、继承特点减少了冗余等问题,利于数据库的版本控制。

(5) 数据库语言与应用主体语言的集成。面向对象数据库较重视与编程语言集成和提高编程速度。面向对象数据库具有编程语言的天然特性,如 small Talk, C++。对象(指方法的执行)的主动特性以及封装特性(增加了完全性和处理选项),使得分布处理比关系模型更为容易。面向对象数据库系统可把数据库对象和数据管理放在应用程序软件中,性能大大提高。把对象处理的负载放在客户端,可解放服务器,以支持更多的客户机。

(6) 提供更丰富的数据操作。面向对象数据库管理系统的语言支持一种无边界范围的数据处理元语,是根据系统和应用程序如何使用和表示数据来定义的,支持用户定义方法。这些方法可在将来通过对象的修改和重编加以改变,以至于成为数据库管理系统的基本操作,增加了数据处理的灵活性。面向对象的查询语言常是沿着系统提供的内部固有联系进行,避免了大量查询优化工作。对象还可以继承事务特性、并发控制特性、安全特性、约束、触发等等许多扩展功能,而不需要重写或复制代码和数据。

(7) 有助于分布计算系统的实现。对象所有的封装、继承、多态等特征,会大大有助于解决分布计算系统的设计中所遇到的如命名、地址空间转换、传送协议、接口描述等复杂问题。尤其是,对象作为一个数据与行为的联合体,并严格分开接口与实现,是在分布计算中把数据和处理分配到最终用户的应用程序中去的一个最合适的包装体。

(8) 系统的可维护性好。封闭性使对象独立,数据差异封闭,很容易实现并行处理。对象是模块化的,任何对象的内部变化都不影响外部的其他编码。对象的这种模块化能力大大简化了应用程序的升级和维护处理,便于系统模式的扩充与修改。此外,对象还是内部可操作的,它可在需用时才动态安装,并

被其他应用程序(不管是哪种程序设计语言)识别、使用或修改。

2.2 面向对象型数据库的局限

面向对象数据库在引入封装、继承、多态、对象标识、复杂对象、计算完整性和可扩展性等面向对象特性的同时,也伴随了一些负作用。

(1) 系统算法复杂。由于对象可以是多态的,多态还允许跨节点存在,这给查询处理算法增加了复杂度。还要小心处理对象模型的继承特性,以及构造和使用对象时使用的层次特性,如果想删除一个对象,则必须处理所有与该对象有联系的对象。经过数据库和事务转换,面向对象数据库系统将会更加复杂。

(2) 执行效率不高。面向对象数据库中,数据通过对象标识(指针)的获取和移动来访问,如果不提供对象集群的方法,面向对象数据库在处理查询时,其响应时间就不能令人满意,需要在数据库内部增加一些结构来提高查询效率。面向对象数据库系统希望把所有的数据模型、数据语言、数据系统及其应用都考虑进去。这种应用的普适性会造成对于每个特定的应用有效性不强,并需要开销更多的硬件和软件。

(3) 安全性能较差。对象模型中一个对象可以收集现实世界中事物的不同信息,这种对象不太可能在一个安全级别上存在,而且面向对象数据库系统在运行时可不经中介而直接调用代码,造成安全设计复杂化。另外,对象模型在定义与操作中的动态性对系统的安全性也构成威胁。

(4) 尚未制定标准。相对而言,面向对象数据库模型仍是一个较新的概念,尚未达到成熟的阶段,其本身的复杂性给实现造成很大困难。尽管相关产品开发了不少,但到目前为止还没有一个为大家所公认的标准或实际应用系统,面向对象数据库模型的基本概念还需进一步精练,某些技术还需进一步开发,只有在其达到稳定之后,才能制定出明确而相对持久的标准。

3 审“时”度“势”的选择——后关系型数据库

关系型数据库与面向对象型数据库在很多方面的特性是互补的,更何况计算机技术的发展一直遵循在兼容已有的产品基础上创新的原则。因此,结合两者功能的后关系型数据库代表了数据库技术研究的一种趋势。

3.1 由来已久的后关系型数据库

后关系数据库早在 70 年代后期就开始有人研

究。1976年P. Chen提出的实体——关系数据模型就是它的一个先例。当时是希望它作为网状型、层次型、关系型数据库三者统一的中间模型。伴随多媒体数据处理的迫切需要,一方面许多厂家在扩展现有关系数据库的性能,ANSI授权的X3H2工作组一直在进行关系数据库查询语言SQL的开发和维护工作;另一方面有些公司、组织借鉴面向对象软件设计技术,研制开发面向对象的数据库系统,如工业界的ODMG委员会旨在组织、发展并统一面向对象数据库模型。

10多年的研究证明,面向对象数据库与关系型数据库的结合是新一代多媒体数据模型的基础。1994年以来,X3H2委员会和ODMG委员会的一些会员举行了一系列联合会议来讨论SQL3(结构查询语言)和OQL(对象查询语言)的融合,以求形成一种单一的数据库查询语言,从而避免分裂市场,也避免使厂商负担过重。

这几年,几个著名的数据库公司的产品都不同程度地体现了对象与关系的结合趋势。如Borland公司的dBASE 5.0 for Windows;Oracle的Oracle 81系列;IBM的DB2的Universal Database 5.0;Sybase的Adaptive Server Anywhere 6.0等产品都符合这种思路,且各具特色。

后关系型数据库是以关系型数据库为基础,以事务型多维数据模型和集成面向对象技术为特征、结合传统数据库(如网状、层次和关系数据库)特点,利用因特网开发背景,面向Internet Web应用的新时代的关系数据库。它能够提​​供事务处理应用开发所需的高性能和伸缩性,支持应用和数据的复杂性,具有独特的可兼容性,正成为事务交易处理的优先选择之一。它的最近一个比较成功的产品是美国InterSystems公司的Caché系统。Caché的高速、可靠、跨平台、可重用已在有几千用户的实际网络环境中得到证明。

3.2 我国数据库技术发展任重道远

我国数据界从70年代后期开始跟踪国际先进的研究趋势,在集中式数据库系统、分布式数据库系统、主动数据库系统、面向对象数据库系统、并行数据库系统、数据库应用等方面都具有有一些成果。但直到90年代后才有自己的商品化数据库系统。而目前数

据库市场仍被国际数据库大公司垄断,并且很多用户已使用或正在开发的数据库系统是国际上已淘汰的产品,以致付出了某些不必要的代价。

后关系数据库90年代后在国内学术界有过一些介绍,但对它的性能及应用前景认识远远不够,了解的人也很少,目前北京只有几家研发医疗系统的单位开始在后关系数据库的基础上开发以医疗信息为中心的医院信息管理系统。其实,除了医疗系统,在金融、证券、交通、制造、电子商务等应用领域,后关系数据库技术也大有用武之地。

目前,数据库技术的研究呈百花齐放局面,缺乏统一的规范与标准。信息技术行业的人士都知道以违反标准换取性能的代价。但在后关系数据库标准还未出现的情况下,遵循已有的关系数据库标准,跟踪数据库技术与市场的发展动态,注意技术创新,在保证性能指标的同时,选择可靠、成熟及移植性好的技术和开发工具,是一种明智的做法。

参考文献

- 1 Bertino. E., and L. Martino. Object-oriented Database Systems: Concepts and Architectures. Reading, MA: Addison - Wesley, 1993.
- 2 Doherty, M., J. Peckham, and V. F. Wolfe. Implementing Relationships and Constraints in an Object-Oriented Database Using Monitors. RIDS93, 1993. 347 ~ 363
- 3 Gallagher, L. Object SQL: Language Extensions for Object Data Management. In Proceeding of the International Society for Miniand Microcomputers, CIKM - 92, 1992
- 4 Simovici, D., and R. Tenney. Relational Database Systems. New York: Academic Press. 1995
- 5 程渝荣. 吹响后关系数据库时代的号角. 微电脑世界. 2000(12)
- 6 李洪等. 多媒体信息的组织和管理技术. 现代图书情报技术. 2000(2)
- 7 周宁. 情报数据库系统. 武汉: 武汉大学出版社. 1994

注 本文由国家社会科学基金项目《数字图书馆的相关关键技术研究》(OOBTQ004)资助。

陈 远 女, 副教授, 发表论文20余篇, 参编著作3部。
通讯地址: 武汉大学传播与信息管理学院. 邮编 430072。

周朴雄 武汉大学传播与信息管理学院研究生。

(来稿时间: 2000-05-11)