

分布式数据库系统中的一些问题

郭江 (北京航空航天大学软件工程研究所)

摘要: 本文围绕着数据管理的透明性、数据可靠性、执行效率、系统的扩充等问题进行了探讨。分布式数据库的透明性虽然增加了设计上的困难,但却方便了用户的使用。由于数据的分布,不仅使得数据的可靠性得到了改善,而且使得内部查询和内在查询的并行化得以提高,从而提高了执行效率;另外,数据的分布还使得系统的扩充变得更加容易了。以上是本文所要讨论的主要问题。

分布式数据库在过去几年中已经有了长足的发展,而且极大地影响着计算机的数据处理方式,在出现了第一代分布式数据库的产品之后达到了顶峰。一些研究人员[1]指出今后几年中大多数企业都将趋向于分布式数据库的管理方式,而集中式数据库的管理方式将遭淘汰。

现在,无论是商业数据处理,还是计算机工程环境(SEE)都迫切需要分布式数据库系统的支持。但就目前的技术状况来说,关键问题还是怎样使分布式数据库成为产品。因此下面这些问题的解决就变得很重要了:

- (1) 分布式数据库中数据管理的透明性
- (2) 分布式数据库中数据的可靠性
- (3) 分布式数据库系统的执行效率
- (4) 分布式数据库系统的扩充

以上这些问题的解决不仅影响着分布式数据库技术的发展,而且还影响着分布式数据库技术向产品的转化。

1. 什么是分布式数据库系统

一个分布式数据库就是一个分布在计算机网络上的多个在逻辑内在联系的数据库的集合[2]。分布式数据库管理系统(DDBMS)是管理分布式数据库并使分布式数据库对用户透明的一个软件系统。一般分布式数据库系统是分布式数据库和分布式数据库管理系统构成的整个系统。下面的假设是一些值得注意的方面:

(1) 数据存放在多个节点上,每个节点在逻辑上是由单个的处理器组成的。尽管一些节点可能是多处理机,但 DDBMS 与这些多处理机上的数据存储和管理是无

关的。

(2) 在各个节点上的处理器是由计算机网络来连接起来的,而不是按多个处理器来配置的。这里关键的一点是处理器的松散耦合,即它们有自己的操作系统和独立的操作。尽管“无任何共享”的多处理器体系结构类似于松散连接的分布式系统,但是它们所考虑的是另一些问题。如负载分配、负载转移、负载平衡等,这些本文中不讨论。

(3) 分布式数据库是一个数据库而不是存放在网络节点上的文件集合。这一特点标志着分布式数据库和分布式文件系统的区别。在分布式数据库中,分布式数据应该使用某种结构的形式化方法来进行逻辑上的相关,而对数据的访问则是在一个高层次上通过共同的接口来实现。这里建立逻辑关系的典型的形式化方法是关系模型。实际上,目前对分布式数据库系统的大多数研究都是在一个关系型系统中进行的。

(4) 该系统具有全面的 DBMS 功能。它既不是一个分布式文件系统也不是事务处理系统。这里所指的事务处理不仅是一种类型的分布式应用,而且也是由 DDBMS 所提供的功能。但是 DDBMS 还提供了其它功能,如查询处理和结构化数据的组织等,而这些却并非是一个事务处理系统必须支持的。

这些是根据目前的技术状况作出的假设,大多数当前的分布式系统是建立在局域网上的,在网上每个节点是一个单个计算机,这样将数据库分布在每个节点上以便形成若干个局部数据库。如图 1。

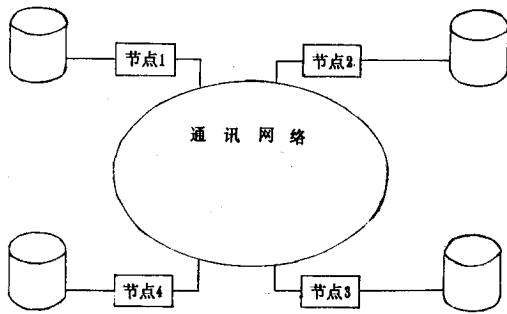


图 1 分布式数据库环境

但是下一代的 DDBMS 是用不同的方法来设计的 [3], 是可负担多处理器网和高速网等技术发展的结果。系统设计还将受到数据库技术应用领域不断扩大的影响, 这些应用领域比商业数据处理更为复杂, 更广泛地采用用户 / 服务器的计算模式及其接口标准。因此, DDBMS 环境将包括与高速网相连的多处理器形式的数据库服务器以及运行应用程序、执行数据库请求的用户机器上的数据库仓库。现在这种类型的分布式关系型 DBMS 已经出现了, 而且若干面向对象的系统也满足这种描述。

此处所定义的 DDBMS 仅是为分布式计算环境提供数据库管理支持的一种方法。现在给出设计选择的一个三维分类。

(1) 自治性。自治性是指单个的 DBMS 可以独立操作的程序以及控制分布的程度。它涉及到许多因素, 包括是否系统的组成成份需要交换信息、是否系统的组成成份可以独立执行事务处理、以及是否允许用户对系统组成成份进行修改。在这里, “交换信息”并不是指所涉及网络的信息交换, 而且指是否需要 DBMS 来交换信息以及是否需要协调用户请求的执行。三种自治的类型是紧耦合、半自治、全自治。在紧耦合系统中, 想共享多个数据库信息的用户可以使用整个数据库的单一映象。半自治系统是由多个 DBMS 组成。这些 DBMS 通常可以独立操作, 但是一般设计成松散耦合的形式以便共享局部数据。在全自治系统中, 各个组成成份是独立的 DBMS, 它们之间互不往来, 也不通讯。

(2) 分布性。分布性是指数据存放的离散程度, 一般

是两种情况, 一种是数据在物理上分布于多个节点而它们之间通过某种通讯媒质的形式来通讯; 另一种是数据在物理上仅存放在一个节点上。

(3) 异质性。异质性是指分布式系统中的各种差异程序。范围是从硬件异质、网络协议异质到数据管理的异质。从数据库系统的角度来看异质的主要形式是: 数据模型、查询语言、接口以及事务处理管理协议等的异质。以这种分类方式可将 DBMS 分为同质的和异质的。

基于以上分类的系统结构划分如图 2 所示。

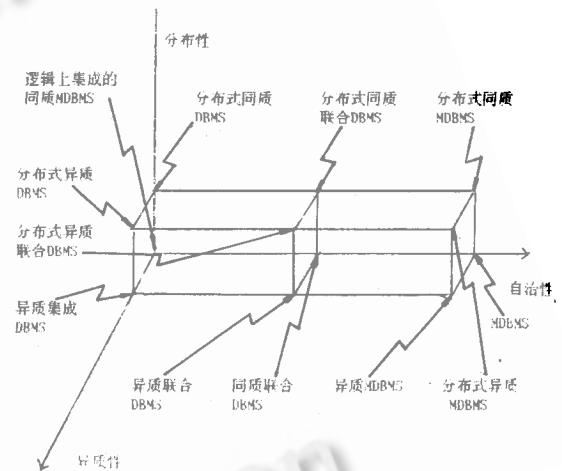


图 2 DBMS 实现的选择方向

其中轴上的箭头并不是指一个无限的选择数目而仅是标志简单的分类维数。本文主要讨论的是紧耦合、分布式、同质的数据库系统。

2. 数据管理的透明性

集中式的数据库系统使用户离开了具体的数据处理结构, 在这种具体的数据处理结构中将数据定义和维护嵌入到了每个应用程序之中, 而这些功能则从应用中抽象出来并置于 DBMS 的服务器控制之下。这种发展方向的结果是数据的独立性增强了, 即一方面应用程序对数据的逻辑或物理组织的变化不太敏感, 另一方面数据组织对程序的变化也不敏感了。分布式数据库的数据独立性是指数据分布在网络节点机器环境中的独立性。

数据独立性是由几种透明的形式来提供的:网络(即分布式)透明性、数据冗余透明性以及局部透明性。将数据的访问设计成透明的方式就会把系统的较高层次的语义和较低层次的实现问题分隔开来。因此,用户将看到一个逻辑上集成的、单一映象的数据库。这样尽管在物理上是分布的,但能使用户象访问集中式数据库一样来访问分布式数据库。就理想的情况来说,全透明性的含义是指一个 DDBMS 将与一个集中式 DBMS 有完全一样的查询语言接口。

但是,目前的大多数商品化 DDBMS 并不能提供一个充分透明的层次支持。部分原因是冗余数据管理的支持。一些系统不允许数据在多个数据库中冗余;而一些允许冗余的系统则要求用户在特定的时间内在一个数据库上进行注册。一个 DDBMS 可以有自己的命名计划,但通常产生的结果不令人满意,而且需要用户确定数据的全部路径名并通过建立别名来避免过长的路径名。这个问题的关键就是透明性缺少合适的操作系统的支持。网络透明性可以由操作系统中透明的命名机制来支持;而冗余透明性也可以由操作系统支持;但是局部透明性的问题却应留给 DDBMS 来解决。

全透明性并不是一个公认的目标。Gray[4]认为全透明性使得分布式数据管理变得更为困难而且“对物理上分布的数据为进行透明性访问的应用程序在可管理性、模块性、以及执行效果方面都不好。”他提出将一个远程过程调用(RPC)机制放置在用户和服务器 DBMS 之间,这样将把用户的查询引导到特定的 DBMS 上。

现在可以认为将全透明性访问提供给用户会使得分布式数据库的管理变得更加困难,而用 RPC 作为通讯基础的用户/服务器的体系结构可以看作是一个正确的解决方法。实际上,一些商品化的 DDBMS 也正是以这种方式来组织的(如, Sybase)。但是,对分布和冗余数据的透明性访问的目标却不能因困难而放弃。这个问题就是:是用户的应用还是 DDBMS 来负责分布的和冗余的数据?目前一般认为[4]应该是 DDBMS,它的组成成份可以以用户/服务器的方式来加以组织。这当然有一些技术性问题需要解决。

3.可靠性

DDBMS 应能改善数据库的可靠性。这是由于 DDBMS 有重复的组成部分,因而消除整个系统崩溃的

可能性。任何单个节点上的崩溃、或通讯连接上的问题都只能影响到一个或几个节点,而不足以毁坏整个系统。当然,通讯连接上的问题可能引起网络的分隔,并因此而更难于处理。在一个分布式数据库中,这样的问题意味着不能访问某些数据,但是用户可以访问数据库的其它部分,这就是分布式处理改善可靠性的一种形式。

一次事务处理是由一系列的数据库操作组成,这样就要求将数据库从一个一致的状态转换到另一个一致的状态,但是当许多这样的事务处理同时进行时就出现了并发执行。因而要考虑实现系统的并发透明性;同时还要考虑可能发生操作失败的情况。也就是一个提供全部处理功能的 DBMS 要确保用户事务处理的并发执行将不破坏数据库的一致性,而且在系统的操作失败时也将维护数据库的一致性和完整性。

分布式处理在多个节点上执行,分别访问各自的局部数据库。由于有对分布式事务处理的全部支持,因而用户的应用可以访问的是数据库的单一逻辑映象,而且依赖于 DDBMS 来确保正确地执行应用的请求而不管系统中发生了何种问题。这就意味着用户的应用不需要涉及协调对局部数据库的访问,也不需要处理在事务处理执行期间节点或通讯连接发生问题的情况。在分布式事务处理和透明性之间存在一个衔接的问题,这是因为二者均涉及到分布式命名和字典的管理。

为了给事务处理提供所需的支持,就需要实现分布式并发控制和分布式可靠性协议。这些都要比集中式数据库系统中的相应部分要复杂。典型的分布式并发控制算法是人所熟知的两阶段上锁(2PL)协议的某个变形,依赖于锁表的放置和锁的管理。分布式的可靠性协议由分布式的提交协议和恢复过程组成。提交协议为了保证任何给定的事务处理有同样的效果(提交或放弃),强加了分布式事务处理的不可分性,而恢复协议确定了全局数据库的一致性是如何从失败中恢复过来的。在分布式环境中,提交协议是两阶段(2PC)协议,在第一阶段,各种节点之间权衡并综合考虑事务处理的命运;在第二阶段,在第一阶段的基础上采取动作。

数据的冗余性增加了数据库的可靠性。一些在某一个节点不能访问的数据如果也存放在别的节点上,那么就还可以对这些数据进行访问。但是,对这种数据冗余即数据拷贝的支持需要往控制协议的实现中加入特定的

拷贝访问语义。最直接的一个语义是拷贝的等价性,这可以通过 ROWA(只能读一个拷贝,必须写所有的拷贝)协议来实现。在 ROWA 之中,对重复数据顶上的一个逻辑读操作可以转换成对它的任何一个拷贝的物理读操作,但是一个逻辑写操作则被转换成对它的所有拷贝的物理写操作。还有一种更为复杂的、有较少限制的拷贝控制协议,这主要是推迟在一些拷贝上的写操作;但目前还没有在什么系统中实现。

并发控制和提交协议是分布式数据库研究中最热门的两个话题,但是它们还没有在商品化的系统中得以广泛的实现。一些系统(如 Oracle_Corp. 的 Oracle)需要用户在一个给定的时间提供一个数据库开口,这样就消除了对分布式事务处理的需求。而另一些系统(如 Sybase_Inc. 的 Sybase_SQL 服务器)则实现了 2PC 协议的基本需求,但是需要用户的应用去协调提交动作。换句话说, DDBMS 并不施加分布或事务处理的可行性而是仅提供基本的功能,又由用户的应用来进行处理。但是,还有一些系统全面实现了 2PC 协议(如 ASK / Ingres 的 Ingres 和 Tandem Computer 的 Non Stop SQL)。

4. 执行效率

DDBMS 较好的执行效率通常是以下面两条为基础的。第一, DDBMS 分裂成概念数据库,使得数据存放到离其应用点最近的节点。这个特征称作数据的局部性,这有两个潜在的优势:由于每个节点只处理数据库的一个部分,这样 CPU 和 I/O 服务的竞争就不象集中式数据库那样严重;而且局部性还减少了远程访问的延迟,这种延迟常发生在广域网中(如,最小的环形信息传播延迟在有通讯卫星的系统中接近 1 秒)。大多数分布式 DBMS 是结构化的,这样就便于从数据的分布中获得最大收益。减少竞争和减少通讯延迟的好处只有通过数据库的合理分布才能得到。

支持 DDBMS 执行效率的第二条,是分布式系统的固有并行机制可用于内部查询和内在查询的并行化。内部查询并行化是多个查询同时执行的结果。内在查询并行化是将单个查询分解成多个子查询以便每个在不同的节点来执行的结果,这样即可同时访问分布式数据库的不同部分。如果用户对分布式数据库的访问仅是查询,即仅是只读访问,那么内部查询和内在查询的并行化就

意味着这样的数据库应尽可能地重复。但是,大多数数据库的访问却不是只读性的,一般是读和修改的混合操作,这就需要实现特定的并发控制和提交协议。

目前的商品化系统主要使用两个执行模型来改善执行效率而不是实现全部的分布式处理支持。第一个模型是在正常的操作时间内仅允许对数据库进行查询访问(即只读性访问),而修改访问则是批处理进行的。通常的情况是当数据库顺序进行批处理的修改时,关闭查询活动,如图 3 所示。

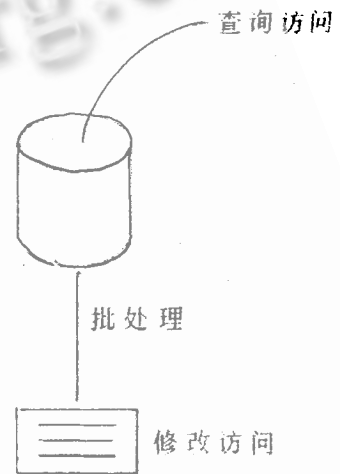


图 3 执行模型 I

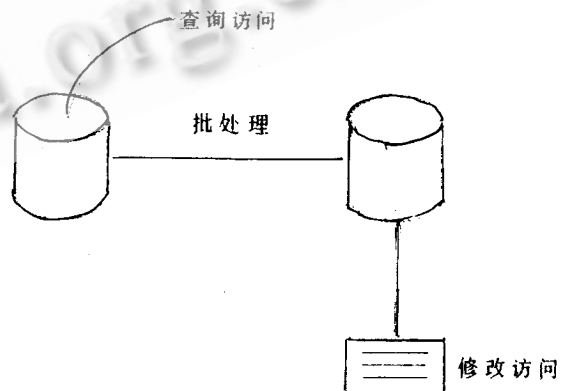


图 4 执行模型 II

第二个模型是建立在多重数据库基础之上;即维护数据库的两个拷贝,一个用于特定的查询(称作查询数据

库),而另一个则用于由应用程序来进行修改(称作生产数据库)。并且定期将生产数据库拷贝到查询数据库中。这种模型仍然需要对生产数据库实现并发控制和可靠性协议,因为这些功能对同步同一数据上的写操作是非常必要的;但是,它却改善了查询的执行,因为这种方式可以不处理操作延迟就能执行查询,如图4所示。

但就目前的实际情况来说,由于没有足够多的实际分布式数据库应用来提供一个实际判断的基础,因此分布式数据库系统的执行特征并不好理解,而且也没有完全开发出来执行模型。通常执行的结果来自于商品化的DBMS,并且考虑了一些测试基础,但目前仍没有公开发表的结论[5]。

5.系统的扩充

在分布式系统环境中,适应数据库的不断增加还是比较容易的。通常很少重新配置系统,一般总是通过给系统增加处理和存储能力的方式来进行扩充,这称作数据库大小规模的扩充。相对而言的是网络规模的扩充。由于不可能获得系统能力的线性增加,而且这种能力的增加还依赖于分布的延伸,因而改进的幅度就是有限的了,但还是存在有重大改进的可能性[6]。

微处理器和 workstation 技术在改善系统的经济性方面已经扮演了一个重要的角色。许多商品化的 DDBMS 是运行在小型机和 workstation 上,以便利用其较好的性能/价格比。而且大多数商品化的 DDBMS 是建立在局域网中,因而 workstation 技术就更适用了。在广域网上运行的 DDBMS 的出现毫无疑问,会增加主机所扮演角色的重要性。就目前的发展状况看,需要 DDBMS 去支持层次结构化的组织形式,在这种组织形式中,一节点是由局域网进行通讯连接的各种计算机组成的,然后再由一个高速的主干广域网来将各局域网连接起来。

另一个经济性因素是数据通讯和远程通讯代价之间的权衡。在前面的部分里,数据的局部化由于减少了延迟而改善了执行效率,因而也减少了代价。在考虑一个需要在多个节点上运行的应用(如订单控制)系统时,将数据进行分布化和局部化处理,就比在一个节点上运行

应用而对存储在另一个节点上的集中式数据库进行远程访问更为经济。换句话说,将数据分布并定期将它们从一个节点传送到另一个节点来执行分布式查询的代价将比经常用通讯来访问远程数据库的代价低。这种经济主张仍是理论上的推测。正如我们已经看到的,大多数分布式 DBMS 是局域网产品,而怎样才能将它们扩展到广域网上运行,仍是一个值得探讨的问题。

参考文献:

- [1] Stonebrader, M., *Reading in Database Systems*, Morgan Kaufmann, San Mateo, California, 1988, PP. 189.
- [2] Ozsu, M.T., and P. Valduriez, *Principles of Distributed Database Systems*, Prentice Hall, Englewood Cliffs, New Jersey, 1991.
- [3] Garcia mobina, H., and B. Lindsay, "Research Directions for Distributed Databases," *IEEE Q. Bull. Database Eng.*, 13(4), PP. 12-17, Dec., 1990.
- [4] Gray, J., "Transparency in Its Place-The Case Against Transparent Access to Geographically Distributed Data" *Tech. Report TR 89. 1*, Tandem Computers, Cupertino, California, 1989.
- [5] Ozsu, M.T., and P. Valduriez, "Distributed Database Systems", *IEEE*, vol.4, No. 8, PP. 68-78, August, 1991.
- [6] Stonebraker, M., "Future Trends in Database Systems," *IEEE Trans. Knowledge and Data 1(1)*, PP. 33-44, Mar., 1989.

· 投稿须知 ·

- 1.内容开门见山,文字简练通顺,书写工整。
- 2.图形正规,程序一律用5宋打印清楚。