

多数据库中间件中分布异构数据缓冲区系统的关键技术

赵 洁^{1,2}, 张 鹏², 齐德昱²

(1 广东工业大学 经济管理学院, 广东 广州 510520; 2 华南理工大学 计算机科学与工程学院, 广东 广州 510640)

摘要: 对实现系统所需要的关键技术进行讨论, 对数据的一致性管理方法进行分析并对传统方法进行改进, 对系统的数据复制策略、局部数据源的即插即用机制、缓冲区系统中的数据名空间和系统的安全管理等关键技术进行深入探讨。

关键词: 多数据库中间件; 分布异构数据缓冲区系统; 数据一致性管理; 复制策略; 局部数据源即插即用

中图分类号: TP311.138

文献标识码: A

文章编号: 1007-7162(2007)03-0064-05

1 研究背景 and 基础

“多数据库中间件”项目主要研究如何构建新型意义上的多数据库中间件, 其实质是构造“DBMS Over DBMS”, 是多数据库的一种兼容与普通 DBMS 的模式化解决方案, 从而解决异构数据库集成问题。模型主要包括 4 个逻辑部分, 如图 1 所示。

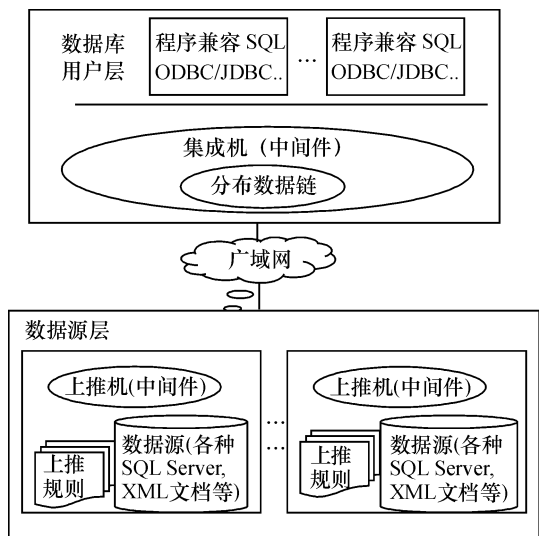


图 1 多数据库中间件架构

数据上推规则: 使用数据上推描述语言 DPDL (Data Push Description Language), 对欲上推出去供第三方访问的数据进行描述。

上推机 (中间件): 为中间件形式, 负责接收“集成机”发来的数据访问消息, 根据“上推规则”访问

数据源, 并将结果回送到相应的“集成机”。

数据缓冲区系统 (中间件): 又称集成机, 为中间件形式, 负责将上层应用的访问请求 (形式为 SQL 或 ODBC/JDBC 等), 转化为“上推机”消息, 发送到“上推机”; 或者, 接收“上推机”的响应消息, 转化为应用程序兼容的数据形式。这一层又分为两大层, 最上面一层作用类似于 ODBC/JDBC 等数据库编程接口, 下面一层是分布数据链接集。

分布数据链接集: 该集合相当于多数据库的各应用视图。

2 分布异构数据缓冲区模型

根据对多数据库中间件的缓冲区的需求分析, 设计了分布异构数据缓冲区系统逻辑模型。缓冲层管理层中的主要组件包括: 应用访问接口 (APIs)、全局数据对象模型管理器、元数据管理器、数据缓存区 (Cache)、事件动作触发管理器、全局事务管理器、XML 转换器、数据收发器 (Server 端), 分布异构数据缓冲区模型, 如图 2 所示。

3 分布异构数据缓冲区中的关键技术

本文将重点讨论该分布异构数据缓冲区模型中的几项关键技术, 包括数据的一致性管理方法、数据的复制策略、局部数据源的即插即用机制、缓冲区系统中的数据名空间、系统的安全管理技术等。

3.1 数据一致性管理方法的分析及改进

目前在多数据库领域, 人们提出许多方法来维

收稿日期: 2006-07-14

基金项目: 广东省重大科技专项资助项目 (2003A1030404)

作者简介: 赵 洁 (1979-), 女, 讲师, 主要研究方向为计算机应用。

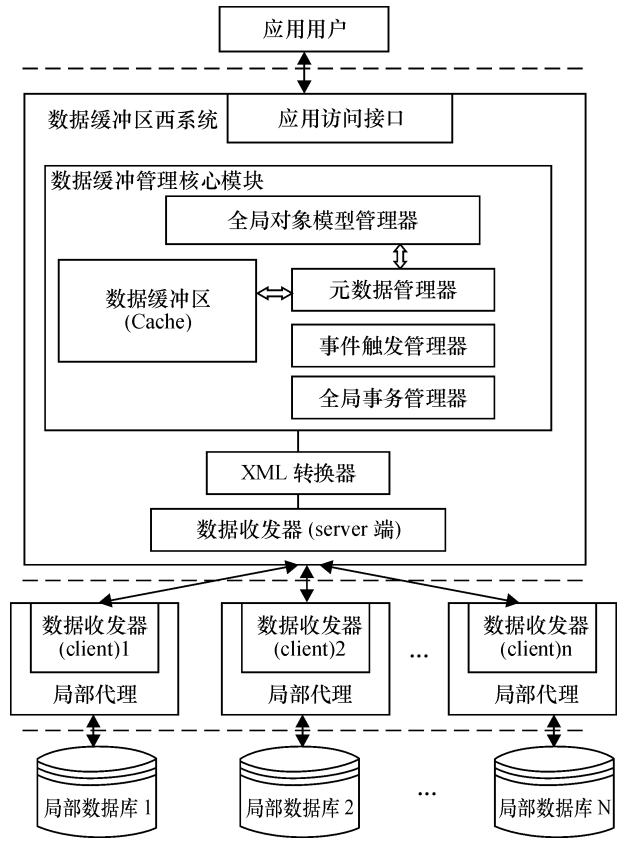


图 2 分布异构数据缓冲区模型

护数据一致性, 包括: 事务控制法、复制控制法以及消息队列法等^[1-2], 但这些方法都有一定的适用局限性. 这些方法成熟的部分可直接应用于本缓冲区系统 DHDB, 在此基础上再进行改进.

首先从新值的传播者和传播时机两方面来论述系统应该采用哪种方法来维护缓存区系统中的数据和各个局部数据库中的数据一致性, 然后给出缓冲区系统针对数据一致性的维护应该有哪些策略供用户选择使用.

1) 就新值传播者而言, push 方式比 pull 方式更适合于本系统^[3]. push 方式中, 只有当数据被更新时才传送信息, 而不至于每次使用数据时都要向服务器申请. 实现数据库信息动态发布的关键问题是, 当数据库的数据发生改变时, 如何及时地将信息发送给关心它的用户, 使信息的推送既具有好的针对性又具有及时的特点. 动态数据的更新机制在解决问题的过程中非常重要, 更新机制的不同形成不同的系统驱动方式.

(1) 用户驱动机制 (pull 方式), 也成为客户轮询方式. 数据更新是通过用户以手动方式提出刷新请求来实现的. 客户端按照预先指定的时间间隔发出查询请求, 应用服务器响应请求, 并访问数据库服务器, 不论数据是否发生变化, 都将当前的数据库查

询结果返回到客户端. 由一系列连续的数据拉去动作实现数据更新.

这种方式的特点是数据的动态更新依靠从客户端的定期主动请求. 时间间隔是预先指定的. 对于数据更新频率较稳定的应用, 可以通过合理的时间间隔设定来达到较好的实时性.

而对于数据更新的频率较大的应用, 时间间隔的设定较困难. 当时间间隔设定过大时, 会降低数据的实时性; 时间间隔设定较小时, 又增加了网络通讯的负担. 通过分析可知, 该方法不能很好地解决数据更新频率不稳定应用的实时性问题, 因此希望有一种新的解决模型, 可以较好地解决该类问题.

(2) 数据驱动机制, 也称为触发式信息推送. 数据更新是利用数据库的触发机制来实现的. 通过触发器和扩展存储过程来探测数据库中的信息的变化, 当数据库中数据发生变化时, 出现插入 (Insert)、删除 (Delete)、修改 (Update) 操作时, 由触发器启动推送过程 (通常是一个扩展存储过程), 将新数据推送给用户, 完成数据更新的任务.

这种方法的特点是只有当数据发生变化时, 才传输数据, 既减少了网络流量, 又减轻了用户的负担, 同时具有较高的用户针对性和数据实时性.

2) 就新值传播时机而言, 本系统根据用户自定义来选择急切更新释放一致性方式或懒惰更新释放一致性方式. 对于数据的数据量比较小时, 采用前者造成的网络通讯量并不大; 而后者只有在需要时才由客户端向服务器索取该数据的最新备份, 这就要求服务器记录所有数据的更新信息, 极大地增加了服务器的负担.

在缓冲区系统中, 要提供多项维护数据一致性的方案供用户按实际需求选择. 用户可以根据数据的更新操作频繁度、系统通信量大小、网络质量状况选择利用同步复制使得各数据副本保持紧一致性; 也可以利用异步复制使得各数据副本保持松一致性. 由于很多实际的应用并不要求严格一致性, 而允许在一定时间内 (容忍的范围有实际的应用决定) 各站点上的数据库内容存在差异, 所以可以利用数据库的同步更新技术, 采用适时更新的方法, 首先在某一站点提交, 然后在一定的时间内系统把所做的更新传送到其它站点, 从而完成数据的一致.

因此, 缓冲区系统根据用户定义的数据更新规则来选择缓冲区中的数据变化是立即更新传播或者是延迟更新传播. 在选择延迟更新传播数据变化时, 将被传播的更新信息都存放到待发送消息队列中, 这样到某一时刻把消息队列中的全部消息发送出

去,大量数据更新就可以完成,这种采取批处理方式操作数据可以充分地提高网络使用率。

在 DHDB 系统中,采用数据双向同步更新技术,保证源数据和副本数据的一致性。局部数据库中的数据发生变化时,局部代理(LA)可以把变化的数据发送给系统缓冲区,使其中的数据副本更新。同样,系统缓冲的数据被全局用户进行更新变化时,也要把数据的变化 diff 发送具体的局部数据源使其数据一致更新。全局用户每一次对缓冲区中的数据操作,都会由全局事务管理器启动一个全局事务来完成此次用户的数据操作,以保证缓冲区数据副本和各个局部数据库系统的源数据的逻辑一致性。

3.2 数据的复制策略

复制本质上是对数据的缓存,缓冲区系统的数据都是来自于局部数据库数据的复制。虽然网络速度提高很快,但要达到高性能地频繁访问和处理大量远程数据仍然很困难。复制技术为用户应用提供一个能够快速访问和处理远程数据的局部缓冲数据拷贝,避免大量数据远程传输到应用端。一般情况下,复制管理应具有以下一些功能:建一个完整的或部分的数据集拷贝;提供选择数据复制策略^[4]、复制方式和复制地点的能力;在复制目录中注册新的数据拷贝;允许用户和应用查询复制目录,以便找到某个文件或数据集已存在的数据拷贝;根据用户和应用的执行要求以及存储、网络的性能预测,有方法和能力选择“最好”的数据副本进行访问和处理。数据复制之间的数据一致性和更新一直是分布式数据管理的难点,它与应用数据访问、产生、操作特性紧密相关。本系统采用的复制策略有以下几种:

1) 可更新快照方式

将只读快照机制扩展为一种允许快照可更新的对称复制机制。快照更新的传播方式和如何应用到快照主节点采用了延迟远程过程调用机制(deferred remote procedure calls RPCs)作为主要的传播和应用变化的机制。各节点之间变化的传播,既可以以基于事件的方式立即传播,也可以在某个特定的时间点,如在网络空闲时(如晚上)传播。在传播变化时,如果其中的一个远端系统没有准备好,传播变化的延迟远程过程调用(RPCs)就会保存在其本地队列中,等到系统准备好以后再执行。

对于快照的定义,既可以是包含一个主表的完全拷贝,也可以是为满足基于值的选择标准的主表中行的子集或者是从主表中经过计算后获得的表视图。快照制技术采用把某一时刻源数据的表的内容通过网络发送到复制的副本,多长时间进行一次表

快照的复制,要根据实际的需求和环境决定。快照在主节点的刷新是按照一定的时间间隔或用户单独请求进行的。最后一次刷新后主表的任何变化也同样被传播并应用到快照。多个快照的刷新是在一个一致的事务中完成的,以确保数据和引用的完整性。

2) 数据链接集方式

缓冲区系统中保存大量局部数据源数据的链接集,而不是仅仅保存局部数据源的某个表的快照。有时候,局部数据库系统的发布者不希望在缓冲区系统保持数据的副本,而是只提供数据访问的链接。多数数据库用户访问缓冲区中的某个数据对象时,若只有此数据的链接方式,那么缓冲区负责根据链接方式即时地去访问局部数据源。这样对用户的数据请求响应可能要受到网络状况的影响,若网络通讯状况较差,可能造成用户等待的现象。此方式的好处是多数数据库用户访问操作的数据都是来自局部数据库的即时数据快照,不存在数据的不一致问题,减轻系统的数据一致性维护的同步/异步更新负担。

3) 其他复制策略

除了前面讨论的两种复制机制以外,还提供另外两种复制机制:过程级复制和同步复制。

过程级复制:这种复制方案主要应用在存在大量数据更新以及采取批处理方式操作数据时需要复制数据的情况。例如,下面这个应用,一个以年为单位管理客户订单的公司,对于老的订单数据不再需要在线保存,而需要卸载到某存储介质中。这时,如果在多个节点间复制每一独立的行级变化效率将是非常低的。对于这种情况,可以直接在每个复制节点执行一个过程来直接进行这种更新操作。

同步复制:一个采用同步复制方案的表发生变化时,会确保这种变化能够成功地作用在本地表和其他节点的复制表,如果失败则整个事务会被成功回滚。同步复制在网络的稳定性比较高的情况下是可行的,可以保证复制节点之间的复制数据一直保持同步。

在本系统中,不仅采用快照可更新的数据复制机制,也使用数据链接集方式,就是在缓冲区系统中不是保存数据副本本身,而是保存源数据的链接。如果用户只想读取数据(而不更新数据),则不必将它存储在数据缓冲区内。相反,可以直接从数据库读出数据并读入应用程序。这样当用户访问数据时,通过数据链接去访问局部数据库中的数据。

3.3 局部数据源的即插即用性研究

DHDB系统必须具备可扩充性,使局部数据源“即插即用”,就是系统能够做到数据源的“热插

拔”。本系统实现了方便的数据源的即插即用机制, 用户可以通过应用程序远程操作后台的 XML 文件, 进行数据源的相关配置如安全认证信息、交互方式、数据连接参数等。

局部数据源系统的即插即用指在不影响整个系统的正常运行的前提下, 可以随时插入和删除数据源系统, 其基本实现思想是局部代理利用 XML 文件和 JDBC 的配合使用, 通过从 XML 文件读出参数, 例如需要的驱动程序、URL 地址、数据源用户名和密码, 来完成对数据源的连接操作。然后局部代理向缓冲区的服务器发送连接请求, 服务器响应局部代理的请求并在缓冲区端初始化有关该局部代理一些必要的网络连接、网络状态、用户注册信息等。

下面论述数据库 (LDBMS) 注册到多数据库系统的过程。当一个数据库新加入系统, 或者某一个数据库因为需要维护而撤出系统, 都需要一个数据库注册机制, 使得客户能够动态地访问多数据库。在缓冲区端, 由数据库注册模块 (元数据管理器一个子模块) 来负责发布数据库注册信息, 当一个数据库系统加入或退出时, 通过修改注册信息的状态来描述某局部数据库是否在多数据库系统中。当收到加入的注册信息时, 缓冲区系统一方面为用户程序提供当前有哪些数据库在线可用, 另一方面通知全局数据模型管理器将已经注册的数据库端的数据字典取出, 以方便用户程序使用。同时, 在本地临时数据库为新库建立相应的镜像库, 这样用户访问系统缓冲区时就可以操作刚加入多数据库系统的局部数据源的那些数据或者撤出的数据源的数据被禁止访问。如此机制实现局部数据源的即插即用。

3.4 缓冲区系统中的数据名空间

缓冲区中的命名空间用于帮助用户如何发现一个存在于特定位置的数据副本 (快照)。名字空间通常有两种实现方法: 一是建立一个完全全局统一的名字空间, 所有用户可以看到系统中的所有数据文件; 二是多名字空间系统, 不同的用户或不同位置使用的名字空间不同, 可以访问数据资源也有所不同。DHDB 系统中存在大量分散的用户和大量分散的数据, 如何建立一个数据命名机制是一个重要问题。它必须实现数据定位的高效率、位置透明性和访问透明性。

DHDB 系统的名字空间结合了单一名字空间和多名字空间的特点, 建立一个全局统一的文件命名空间, 对所有用户提供相同的文件命名、定位和访问机制。同时针对不同的用户的兴趣, 为他们提供不同的数据视图。这样, 系统针对特定用户要维护的元数

据信息也相应减少从而提高了数据定位的效率。整个 DHDB 系统的物理资源由分散的多个局部数据库 LDB 构成, 为了便于管理和存储各个 LDB 的数据, 笔者提供的一种命名机制, 也就是针对数据来源分布的分散性, 提供一种名字空间使得数据在逻辑上构成一个统一数据视图。给用户提供的名字空间是逻辑的, 以屏蔽数据物理位置的分散性, 实现数据的透明访问。就是说, 用户对数据的各种访问。

在缓冲区系统存在着表映射的问题。默认情况下, 当从数据库表生成数据集时, 数据集内表和列的名称与数据库中的相同。但这可能并不实用。例如, 可能会发现数据库中使用的名称太简洁或太罗嗦了, 或者名称使用的是外语。如果使用的是现有架构, 则还可能发现架构中定义的名称与数据库中使用的名称不匹配。因此, 数据库中的名称和数据集内的名称不必匹配。相反, 可以在数据集命令中创建新的表名和列名, 然后将它们映射到数据库中所使用的名称。缓冲区可以使用 Table Mappings 集合维持数据集结构 (数据表和数据列) 与数据存储区结构 (表和列) 之间的对应。

3.5 系统的安全管理概述

下面对系统容错功能进行研究。多数据系统中, 由多个结点组成, 若有结点从系统中退出 (可能突然断电或者软件故障等原因) 或加入, 要保证系统的正常运行, 必须对发生的变化做适应性调整。由安全认证机制保证访问系统的用户是授权用户。

DHDB 系统提供相应的安全保密措施, 以保护存储在各个局部数据库中的各类数据, 防止对这些数据的非授权访问、修改和破坏, 并保证被授权的用户能在所授权的范围内对数据进行访问和操作。系统的安全管理包括 3 个方面^[5]。

1) 身份验证

对于登录 DHDB 系统的用户, 验证其真实身份, 确定该用户确实是他所声称的用户; 对于用户要连接的服务器, 验证远程服务器的身份, 确定该服务器确实是用户所要连接的服务器。DHDB 系统具有分布性、异构性, 数据存放分散。通信的双方面临着如何信任对方就是所声称的用户、服务器, 以及通信过程中如何保证通信数据的安全性的问题。在 DHDB 系统中, 身份认证指的并不仅仅是用户的身份认证, 也包括服务器之间的相互认证。

DHDB 系统面临着被窃听、被重放攻击或假冒攻击这些安全问题, 这使得身份认证成为 DHDB 安全管理的一个重要组成部分。DHDB 系统中, 在执行真正的数据访问操作之前, 用户与数据库服务器之

间要进行双向的身份认证,防止假冒攻击;在用户登陆系统之后,数据库服务器之间传输数据之前也要进行相互的身份认证;确认了身份之后,在通信双方之间建立起保密通道,数据处理的过程中采用加密措施,这样就能有效地防备窃听。

2) 访问控制

对主体(用户)访问客体(数据库对象)的操作实施控制。验证了用户的身份后,能根据用户所具有的权限、所扮演的角色来确定该用户能够做什么和不能够做什么;确定用户在哪些数据对象可执行什么样的操作,这可以保证用户只能存取他有权存取的数据,不能存取他无权存取的数据。访问控制(Access Control)就是通过某种途径显式地准许或限制访问能力及范围的一种方法。通过访问控制服务,可以限制对关键资源的访问,防止非法用户的侵入或者因合法用户的不慎操作所造成的破坏。

DHDB系统的访问控制基于角色授权和直接授权两种方式。在DHDB系统中,定义了不同的授权单位(用户和角色),不同的客体保护粒度(系统权限和数据库表权限),对不同的访问权限能够进行合并和冲突的消解。

3) 服务器间通信的安全

服务器之间发送的信息都经过加密,第三者即使拦截了数据流,也无法得知真实的数据信息,可以抵御外界的恶意数据攻击。

4 结语

多数据库中间件中分布异构数据缓冲区系统的实现是一项复杂的工作,需要综合使用多种技术,本

文对其中几项关键技术进行深入研究:分析系统的数据一致性管理方法,对传统方法进行改进,提供多项维护数据一致性的方案;详细介绍系统采用快照可复制方式与数据连接集方式综合的数据复制策略;说明了局部数据源的即插即用机制如何实现;介绍系统所建立的全局统一的名空间结合单一名空间及多名字空间;并对系统的安全管理机制所包含的3方面措施进行介绍。

上述技术构成该缓冲区系统实现的理论基石,也是对一些传统技术的改进或创新。下一步将进行具体的实现,但系统的实现还涉及很多其他的技术问题,还需要深入研究与探讨,以便实现系统所应该具备的功能。

参考文献:

- [1] 陈珉,喻丹丹,涂国庆. 分布式数据库系统中数据一致性维护方法研究[J]. 国防科技大学学报, 2002(3): 76-80
- [2] 宋宝燕,张斌,王国仁,等. 物理集成的多数据库系统中数据一致性维护方法[J]. 小型微型计算机系统, 2000(8): 858-861.
- [3] 彭路,胡越明. Push技术及其在Internet上的应用[J]. 计算机应用研究, 1999(9): 42-43
- [4] Lewis C. Oracle Advanced Replication: A Definition in Plain English [EB/OL] [2005-03-24].
<http://blogs.ittoolbox.com/oracle/guide/archives/oracle-advanced-replication-a-definition-in-plain-english-3611>
- [5] 林志斌. 数据库安全性若干问题的探讨[J]. 微型机与应用, 1998(3): 8-9

Research on Key Technologies of the Distributed and Heterogeneous Data Buffer System in Multidatabase Middleware

ZHAO Jie^{1,2}, ZHANG Peng², QI De-yu²

(1. Faculty of Economics and Management Guangdong University of Technology, Guangzhou 510520, China

2. School of Computer Science and Engineering South China University of Technology, Guangzhou 510640, China)

Abstract The Distributed and Heterogeneous Data Buffer (BHDB) is an important part of multi-database middleware. This paper discusses data consistency management of the system, improves the traditional technology, and incubates the data replication strategy, local data source's plug-and-play mechanism, data namespace and system security management, which are the key technologies and the basis of the system realization.

Key words multi-database middleware; distributed and heterogeneous data buffer system; data consistency management; data replication strategy; local data source's plug-and-play