第 11 卷 第 11 期
2006 年 11 月

中国图象图形学报
Journal of Image and Graphics

Vol. 11, No. 11
Nov., 2006

网格环境下多数据源并发控制的研究

张 伟 赵正德 刘宜宁 全卫新
(上海大学计算机工程与科学学院, 上海 200072)

摘 要 为了有效地控制和实现网格用户的并发操作,提出了一种基于可变周期的任务序列重建的并发控制算法,用来保证网格用户访问的并发性和并行性;同时提出服务一致法和服务跟踪法,用来维护数据源数据的一致性和客户端显示信息与数据源数据的一致性,这也体现了网格数据的易变性和系统的可适应性。

关键词 任务序列重建法 服务一致法 服务跟踪法 网格技术
中图法分类号: TP393.07 **文献标识码**: A **文章编号**: 1006-8961(2006)11-1543-04

The Research on the Technology of Concurrency Control Under More Data Sources Based on GRID

ZHANG Wei, ZHAO Zheng-de, LIU Yi-ning, QUAN Wei-xin
(*School of Computer Engineering and Science, Shanghai University, Shanghai* 200072)

Abstract In this paper, we propose a concurrent control algorithm, which can realize the parallel and concurrent visiting of users, on the basis of the work regenerate of alterable period; And then the consistency service and tracking service are raised to maintain the consistency of data source and client perform information.

Keywords task list regenerating, service consistency, service tracking, grid technology

1 引 言

网格(grid),又称为虚拟计算环境,是近年来兴起的一种重要的网络信息技术。伴随网格技术的日益发展,作为对数据应用需求的快速回应,在数据网格之后又出现了网格数据库[1]的概念。因此在网格环境下如何访问已有的数据源,如何有效地控制和实现网格用户的并发操作,并在此基础上将已有的多个数据源动态地组成一个虚拟数据库来完成特定任务,是一个非常值得研究的问题。

本文在同构或异构的网格环境下,针对多个网格用户对同一数据的操作进行了并发控制的研究。

2 并发控制与数据一致性

2.1 并发控制

网格环境下,由于用户数量巨大,致使信息资源剧增,因而往往存在许多用户同时并发地存取数据库中相同数据的现象,这时就需要数据库服务器对这些并发执行的事务进行处理和控制,否则就有可能造成不正确的结果和数据源不一致的状态。并发控制的作用就是协调在同一时间里访问数据库的用户的相互作用,其目标是使得多个用户能以多道程序设计的方式访问数据库,使每个用户就像在一个专门的系统上单独执行一样。

并发控制一直是多数据库系统所要解决的一个关键的问题,它既是个热点问题,也是个难点问题。管理数据库中的并发有以下 3 种常见的方法[2]:

(1) 保守式并发控制,即在从获取记录直到记录在数据库中更新的这段时间内,该行对用户不可用;

(2) 开放式并发控制,即只有当实际更新数据时,该行才对其他用户不可用;

(3) 最后的更新生效,即只有当实际更新数据时,该行才对其他用户不可用,但是,不会将更新与

收稿日期:2006-07-28;**改回日期**:2006-09-05

第一作者简介:张伟(1982 ~),男。2005 年获安徽建筑工业学院理学学士学位,现为上海大学硕士研究生。主要研究方向为 CSCW、多媒体及网络技术等。E-mail:zhangwei0945@163.com

初始记录进行比较，而只是写出记录，这可能就改写了自上次刷新记录后其他用户所进行的更改。

在以上的基础上，本文提出基于可变周期的任务序列重建并发控制算法。

基于可变周期的任务序列重建法是指系统的并发调度程序每隔一个周期就扫描任务队列，并分析任务队列中没被执行的各个任务，同时判断这些任务有无读写冲突，然后将发生冲突的任务放入一个队列（队列 1），而将没有冲突的任务放入另一队列（队列 2）。队列 1 中的任务插入链表，并按照先写后读的顺序来重建任务序列后依次执行，而队列 2 中的任务并行执行。这里的周期是可变的，其随任务队列中任务的执行情况而变化。

2.2 数据一致性

如何保证数据库系统的数据一致性，长期以来一直是人们所关心的问题，现在有很多种方法可以用来解决数据一致性[3]，例如：

（1）数据库间的复制方案，数据库的复制是将一组数据从一个数据源拷贝到多个数据源的技术，也是将一份数据发布到多个存储站点上的有效方式。

（2）订阅-通知方案，即目的方根据自己关心的事件指定订阅表达式，发给源方，源方接收表达式后，即产生订阅的相关信息，并由源方控制订阅的生命周期，当订阅服务的生命周期到了，便将其停止并销毁。

在以上两种方法的基础上，本文提出了服务一致法和服务跟踪法。

服务一致法：该方法在开发 Web 服务时就考虑相关数据的一致性维护问题，即当对数据源中某个数据修改时，就同时将数据源中和系统服务器的结果数据库中与此数据相关的内容进行修改。

服务跟踪法：该方法是在修改数据源的同时，将相应的已读取表信息的标志位置为 1。在客户端进行循环探测时，若标志位为 1，则通知网格用户通过刷新页面来得到更新数据。

3 任务序列重建并发控制

3.1 数据结构

在介绍基于可变周期的任务序列重建并发控制方法之前，先引入实现该算法的数据结构，也就是几张重要的表。

tasklist 表：用于存放的是任务队列的信息。字段如下：id（自动编号），servername（用户提交的服务名），finished（服务完成标识，初始值为 0，表示尚未被处理），urladr（服务所在的地址），checked（服务读到的数据有无改变）；

realtasklist 表：用于存放分解后的子服务。字段如下：id（用户提交的任务编号），servername（用户提交的服务名），finished（服务完成标识），linklistname2（服务操作的数据库表名），pid（自动编号），checked（服务读到的数据有无改变），caozuo（读写操作标志），wlist（是否插入冲突队列）；

realtasklist1 表：用于存放冲突判断后需要调用的子服务。字段如下：id（用户提交的任务编号），servername（用户提交的服务名），finished（服务完成标识），linklistname2（服务操作的数据库表名），pid（子任务编号），checked（服务读到的数据有无改变），caozuo（读写操作标志）；

realtasklist2 表：字段同 realtasklist1 表；

multitabl 表：用于存放多表操作的服务信息。字段如下：servername（多表操作的服务名），subservername（子表操作的服务名）。

3.2 任务序列重建并发控制的设计

并发所引起的问题来自对同一数据对象的写-写冲突或读-写冲突，而且问题出在“写”上，只读事务并发执行不会发生问题。并发控制的任务旨在避免访问冲突所引起的数据不一致。

本系统把不同的数据源所提供的网格服务（单表操作）作为原子，当网格用户提出一个请求时，就可先根据客户提交服务的 Web 服务名在 UDDI（universal description discovery and integration）注册中心中查找这个服务所在的位置，然后向 UDDI 数据库中的 tasklist 表插入一条记录。若 tasklist 表中存放的服务是多表操作，则把它们分解成针对单表的原子操作，分解后的子服务再插入表 realtasklist 中。当并发程序扫描 realtasklist 表时，则先分析其中 finished 标志为 0 的服务，并判断服务之间是否存在读写冲突。对于有冲突的服务，其可能是多表操作，若其可以分成多个子服务，则把分解后的子服务插入 realtasklist1 表中，否则把该服务直接插入 realtasklist1 表中。对于没有冲突的服务，若其可以分成多个子服务，则把分解后的子服务插入 realtasklist2 表中，否则将其直接插入 realtasklist2 表中。此时表 realtasklist1 和表 realtasklist2 中存放的

是实际需要执行的所有任务。这样一个服务提交的先期操作就完成了,之后就可由调度程序来调度任务序列。

具体的并发控制流程如图 1 所示。

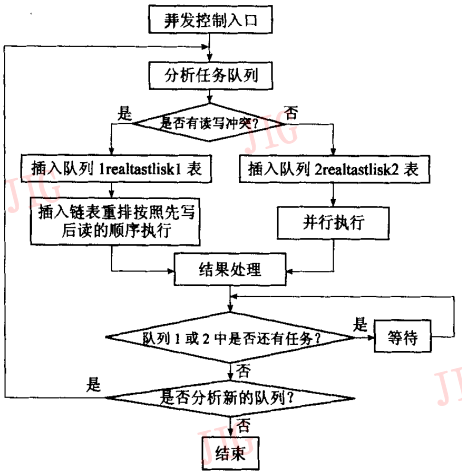

图 1　并发控制流程

Fig. 1　Flow chart of concurrency control

3.3　任务序列重建并发控制的实现

任务序列重建并发控制的执行过程是:先对用户提交的请求进行分析,同时将相应的 Web 服务插入 tasklist 表中,并将其分解为原子操作插入 realtasklist 表中,再判断读写冲突,并将发生冲突的操作插入 realtasklist1 表,而将不冲突的插入 realtasklist2 表;realtasklist1 表中的操作任务是按照先写后读的顺序执行,而 realtasklist2 表中的操作任务则并行执行。

其具体的代码省略,主要步骤如下:

(1) 接受用户请求,先将请求插入 tasklist 表任务队列,然后分解多表服务,并将原子操作插入 realtasklist 表。

(2) 分析 realtasklist 表,同时形成 realtasklist1 表和 realtasklist2 表。

(3) 通过分析冲突队列 realtasklist1 表来重建任务序列。

4　数据一致性控制

本文采用服务一致法和服务跟踪法来解决各个数据源中数据的一致性问题和客户端显示信息和数据源中数据的一致性问题。

4.1　服务一致法

数据源的一致性涉及到以下两个层次,首先是网格服务的数据源部分;其次是系统服务器中保存的信息。该一致性是用服务一致法来实现,即在开发 Web 服务的时候就要考虑相关数据的一致性维护问题。

4.2　服务跟踪法

对于客户端显示信息和数据源中数据的一致性问题,本文借鉴订阅 - 通知方式,利用服务跟踪法来解决的,即通过在页面中嵌入一个 iframe 页面,并不断地探测数据库中的数据有没有改变,如果改变了,就通知客户数据已变化,然后客户刷新页面就可看到最新的数据。具体分为两个步骤,数据修改和用户页面探测。

(1) 数据修改

当调用 Web 服务执行写操作修改了数据源某表(设为 tableA)中的数据时,则可将 realtasklist 表中所有 linklistname2 字段值为 tableA 的记录中 checked 字段置为 1。同样根据修改记录在 realtasklist 表中的 id 号(设为 idA),先将 tasklist 表中 id 号为 idA 的记录中 checked 字段置为 1;然后根据被修改记录在 tasklist 表中的服务名,将临时表 temptable 中相同服务名记录的 checked 字段置为 1。

(2)用户界面探测

用户界面探测时,先查询询所请求的服务中标志位 checked,并判断此服务请求已读到的数据有没有改变,并在一个调用页面中用 iframe 隐含一个页面,然后每隔时间 T,自动探测一下 temptable 表中 checked 标志位是否为 1,若为 1,则说明所访问的数据已经改变了,此时就为用户弹出数据改变通知。最后根据用户的 IP 地址(设为 ipA)和所请求的服务名(设为 snA)来探测 temptable 表中 checked 标志位。

这样通过嵌入探测页面的方式,就可在数据发生改变时自动通知用户,用户只需刷新页面就可得到最新数据。这就实现了客户端显示的信息与数据源中数据的一致性。

5　并发控制实例

某时刻系统接收了 4 个用户提交的请求后,其 tasklist 表及 realtastlisk 表如图 2 和图 3 所示。

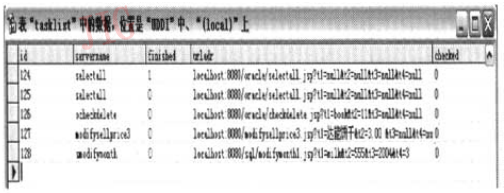

图 2 4 个请求提交后的 tasklist 表

Fig. 2 Table tasklist with four requests

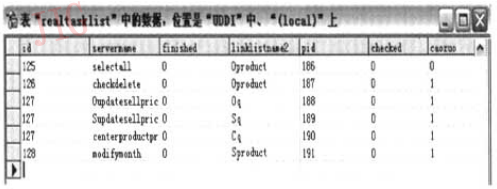

图 3 4 个请求提交后的 realtasklist 表

Fig. 3 Table realtask with four requests

并发程序扫描后，先对 id 编号为 125 ~ 128 的服务进行分析（周期开始），由于得知 id 号 125 和 126 分别对 Oproduct 表进行读写操作会发生冲突，故把 pid 号为 186 和 187 的两条记录放入 realtasklist1 表中。因 id 号 127（分解为 3 个子服务）和 128 不冲突，故先把 pid 号为 188 ~ 191 的 4 条记录放入 realtasklist2 表中；然后对 realtasklist1 表中的子任务进行重排，以形成包含两个节点的链表，187 号写操作在前，186 号在后，并按先写后读顺序执行。realtasklist2 表中各子任务并行执行，各子任务执行后再将相应的 finished 标志置为 1，若是写操作，就将 checked 置为 1，同时根据 pid 号修改 realtasklist 表中的 finished 和 checked 标志；当 realtasklist 表中与 id 号相同的所有记录的 finished 标志均为 1 时，则根据 id 号将 tasklist 表中的 finished 置为 1；若 realtasklist 表中相同 id 号的服务只要有一条记录的 checked 标志为 1，则根据 id 号将 tasklist 表中的 checked 置为 1，同时还需根据 tasklist 表中的服务名将临时表 temptable 中相应记录的 checked 置为 1。这时 tasklist 表中 125 ~ 128 号记录的 finished 都为 1，则 4 个请求执行完毕，也即一个周期结束，然后并发程序继续扫描新的任务队列，开始下一个新的周期。

6 结 论

本文首先概述了常用的并发控制方法和一致性方法，之后借鉴这些方法，提出了基于可变周期的任务序列重建并发控制方法，有效地控制多用户的并发访问；为了保证数据的一致性，又结合面向服务的特点，设计了服务一致法来维护数据源中的数据一致性，又借鉴订阅-通知方法设计了服务跟踪法来实现客户端显示信息与数据源的数据一致性。

参考文献（References）

1 Malaika Susan, Eisenberg Andrew, Jim Melton. Standards for databases on the grid[J]. ACM SIGMOD Record, 2003, **32**(3): 92 ~ 100.

2 MAO Xin-yu, WANG Hua-wen. A New Algorithm For Concurrency Control in Distributed Database System[J]. 2002, **21**(4): 12 ~ 13, 22. [毛新宇，王化文. 一种用于分布式数据库系统的并发控制新算法[J]. 微型机与应用, 2002, **21**(4): 12 ~ 13, 22.]

3 LU Zheng-ding, YANG Yu-ping, LI Chang-lei, *et al*. Maintaining consistency in multidatabase systems [J]. Journal of Computer Research and Development, 2001, **38**(2): 157 ~ 162. [卢正鼎，杨玉萍，李长磊等. 多数据库系统中的一致性维护[J]. 计算机研究与发展, 2001, **38**(2): 157 ~ 162.]