计算机科学 1999Vol. 26№ 11

基于 WWW 的分布式图书资料信息系统中并发控制研究

On Concurrency Controlling of Distributed Book & Data Information System Based on WWW

舒忠梅　胡金柱　　左亚尧

（华中师范大学计科系　武汉 430079）（华南师范大学　广州 510613）

Abstracts　In this paper, we introduce the function module and the architecture of DBDIS(Distributed Book & Data Information System), present query concurrency controlling based on improved time-stamp and update concurrency controlling based on lock. Furthermore, we give the implementation methods.

Keywords　Concurrency, Lock, Time-stamp, DBDIS, Petri net

目前，各企事业单位的图书资料管理多数还停留在效率比较低下的手工作业的水平上；另一方面，大部分单位内部已建立了局域网环境，有的甚至联上了 Internet。因此，开发一个分布式 BDIS 是一项具有重要现实意义和实用价值的研究课题。DBDIS 作为一个分布式数据库管理系统，它面临的是多任务分布环境，可能会有多个用户在同一时刻对同一数据进行读或写操作，因此需要并发控制功能保证数据的一致性，提高系统的并发能力。

1　分布式 BDIS 的体系结构

分布式 BDIS 系统是基于现代网络环境下的分布式 MIS，根据各单位的部门层次性特点，DBDIS 采用多层分布式 B/S 体系结构。其主要功能包括图书资料的管理、流通、查询、统计打印及读者的管理等。其中，图书资料包括图书、连续出版物、电子出版物、缩微出版物及会议论文集等。系统的主要功能模块如图 1 所示，系统的体系结构如图 2 所示。

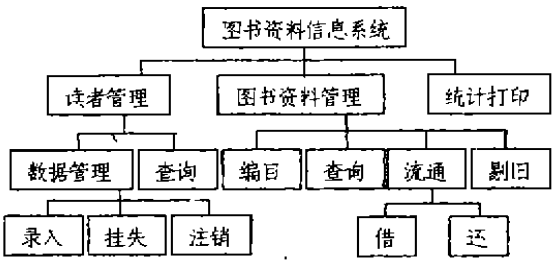


图 1　DBDIS 功能模块结构图

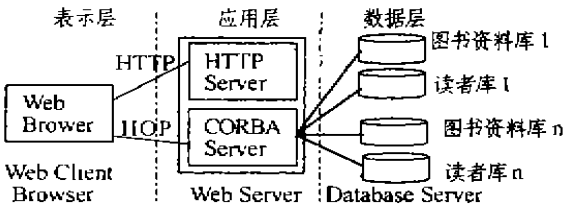


图 2　DBDIS 的体系结构图

2　DBDIS 的并发控制模型

在数据库系统中，事务以发出读操作和写操作原语的方式存取数据库。事务是包含一组操作（查询、插入、更新和删除）的逻辑单位，具有 ACID 四个特性，DBDIS 中的事务操作分为两类：只读和只写，即查询和修改。任一事务 T_i 的只读操作记为 $OR_i(x)$，只写操作记为 $OW_i(x)$，x 是一数据项。一般合法用户都可通过网络查询图书资料或读者的有关信息，而修改操作（包括借、还、录入、编目、挂失、注销、统计打印等）有一定的权限，必须由资料管理员来处理。为了保证分布式数据库系统中多个事务的高效正确执行，必须采取一定的并发控制策略来管理、调度分布式事务。

2.1　基于锁的修改操作并发模型

锁方式的基本思想是：事务对任务数据的操作必须先申请该数据项的锁，待加锁成功以后才可以对数据项进行操作。操作完以后，要释放已申请的锁。通过锁的共享与互斥的特性，实现事务的可串行化调度，从而保证事务的并发正确执行[1]。DBDIS 中的锁分为两种：只读锁与只写锁。只读锁是对数据项进行只读操作时要加的锁，只写锁是对数据项进行只写操作时要加的锁。只读锁与只写锁的相容矩阵如下：

	只读锁	只写锁
只读	共享	共享
只写	共享	互斥

事务间出现的冲突操作，通过加锁的原则及锁的相容机制实现冲突操作的可串行化调度。在DBDIS中，查询操作是共享的，其事务并行执行；修改操作是互斥的，其事务串行执行。对于系统中的修改操作，采用锁模型并发机制，即某事务要对一数据项执行借、还、录入、编目、挂失、注销、统计打印等操作前必须申请该数据的只写锁。

2.2 基于改进的时间印查询操作并发模型

改进时间印模型所基于的原则是：对于每个事务在激活时系统分配给其时间印和方向，它们可以唯一地标识一个事务及事务激活的次序，事务中的操作拥有事务的时间印和方向，事务对数据项进行操作时把自己的时间印和方向赋给该数据项，当事务间存在冲突操作时，冲突操作间的执行次序由时间印和方向来决定。在DBDIS中，对于只读性质的查询操作，采用改进时间印并发模型。某事务对读者或图书资料信息进行查询，当它启动时，系统将分配给该事务一全局唯一的时间印和一方向，此时间印的大小和方向决定该事务何时被执行。

3 并发控制功能实现

在DBDIS中，对于查询事务采用改进时间印方法；对于其他事务采用两段锁(2PL)方法来实现锁模型功能。即使一个数据项正被其他事务封锁，查询事务也可以访问该数据项在加时间印之前的前映像。

3.1 基于修改操作的锁并发模型实现

在DBDIS中，采用分布式两段锁协议(2PL)来实现修改操作的并发。2PL协议是指并发执行的多个事务中事务对数据进行操作以前要进行加锁，且每个事务中的所有加锁操作在第一个解锁操作以前执行。分布式2PL协议指的是：对于几个分布事务$T_1,T_2,\cdots,T_n$，在m个场地上执行，其全局调度F可以由一组局部调度序列$S_1,S_2,\cdots,S_m$来描述，每一个局部调度S_k $(k=1,\cdots,m)$均遵守2PL协议，且对于冲突操作事务T_i,T_j的冲突操作对O_i,O_j，有$T_i<T_j$，则全局调度F满足分布式2PL协议。

考虑以下三个事务T_1,T_2,T_3，设x,y为两个读者记录，T_1,T_2为在一场地对读者x进行查询，然后进行还书操作；在另一场地对读者y进行查询，然后进行借书操用。T_3在一场地对读者y进行查询，然后进行借书操作；在另一场地对读者x进行查询，然后进行还书操作。它们的操作序列如下：

T_1:$OR_1(x)OW_1(x)OR_1(y)OW_1(y)$
T_2:$OR_2(x)OW_2(x)OR_2(y)OW_2(y)$
T_3:$OR_3(y)OW_3(y)OR_3(x)OW_3(x)$

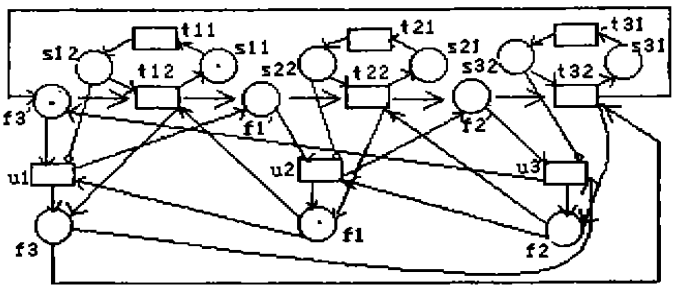
若三个事务同时被激活，2PL协议保证它们的可串行性，设$T_1<T_2<T_3$，将T_1,T_2,T_3分解执行得到以下两个局部场地的调度：

S_1:$OR_1(x)OW_1(x)OR_2(x)OW_2(x)OR_2(x)OW_3(x)$
S_2:$OR_1(y)OW_1(y)OR_2(y)OW_2(y)OR_3(y)OW_3(y)$

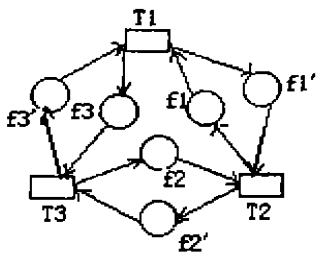
按照可串行化理论，S_1中的$OR_2(x)OW_2(x)$和S_2中的$OR_1(y)OW_1(y)$，S_1中的$OR_3(x)OW_3(x)$和S_2中的$OR_2(y)OW_2(y)$并发执行，但在2PL协议中这是不允许的，因为T_1在获得对y的锁之前不会释放对x的锁。根据2PL协议，事务只有在提交时才释放锁，则T_2,T_3被迫等待。这样虽然降低了并发执行的速度，但对保证分布式事务的正确调度是必不可少的。否则，一些不可串行化的调度也将执行，从而无法保证正确的并发调度。例如下面两个调度是不可串行化的：

S_1:$OR_1(x)OW_1(x)OR_3(x)OW_3(x)$
S_2:$OR_2(y)OW_2(y)OR_1(y)OW_1(y)$

根据2PL协议，正确的事务T_1,T_2,T_3的并发执行模型用Petri网表示如图3所示，每个事务有空闲(S_{i1})和请求锁(S_{i2})两种状态，t_{i1}表示等待锁，t_{i2}表示释放锁。



(a) 锁的并发控制详细图



(b) 等价图

图3 事务T_1,T_2,T_3的并发执行模型图

3.2 基于查询操作的时间印并发模型实现

在DBDIS中，对于所有查询操作中的并发冲突问题，采用时间印和事务的方向相结合的方法来考虑冲突操作间事务的等待与重启[2]。事务在激活时，系统将自动分配其时间印和方向。事务在整个操作过程中，时间印的大小将保持不变，但方向会发生改变。事务的时间印表示为TS(T)，事务方向表示为Ot(T)。事务的Ot(T)可以取值为中立(n)、向前(f)、向后(b)，其初始值为'n'。当T_1向T_2发出请求，如果$TS(T_1)>TS(T_2)$且T_1可等待T_2(事务是否可等待取决于事务决策表[2])，则$Ot(T_1):=Ot(T_2):='f'$，即向前等待；如果$TS(T_1)<TS(T_2)$且T_1可等待T_2，则$Ot(T_1):=Ot(T_2):='b'$，即向后等待；但若不允许T_1等待T_2，则年轻的事务回滚并重启，并将其方向置为'n'。

在此协议中，具体的等待方法为：(1)如果$TS(T_1)<TS(T_2)$，$Ot(T_1)$、$Ot(T_2)$同为'f'或同为'n'，亦或其中一个为'f'，另一个为'n'，则允许T_1向前等待T_2。(2)如果$TS(T_1)>TS(T_2)$，$Ot(T_1)$、$Ot(T_2)$同为'f'或同为'n'，亦或其中一个为'f'，另一个为'n'，则允许T_1向后等待T_2。其实现算法为：

步1：如果T_1申请到一有效锁，则不必查看决策表。

步2：如果T_1请求的锁已被另一个事务T_2占用(T_1，T_2事务冲突)，则需查看决策表并做出决定。

下面考虑两个查询事务之间的通信。事务占有资源的信息是其时间印和方向，由数据管理器(DM)来负责处理。当有事务请示资源时，DM将决定该资源是否有效。如有效，DM将产生一个包含这些信息的对象，否则根据决策表决定事务是否等待。假设T_1占有资源x，$Ot(T_1)='n'$，T_2申请T_1所占有的x，且$TS(T_1)<TS(T_2)$，根据以上规则和算法，有以下三种情况(情况(1)与(2)如图4所示)：

(1)若$Ot(T_2)='n'$，则允许T_2等待T_1，并让$Ot(T_1)='b'$，$Ot(T_2)='b'$。

(2)若$Ot(T_2)='b'$，则允许T_2等待T_1，并让$Ot(T_1)='b'$。

(3)若$Ot(T_2)='f'$，则不允许T_2等待T_1，T_2回滚，并且$Ot(T_2)='n'$。

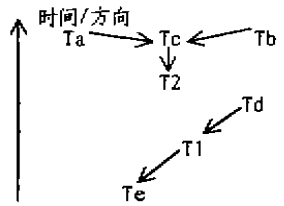


图4 T_2向后等待T_1图

结束语 本文主要针对DBDIS中的修改和查询操作并发冲突，讨论了其并发执行模型与具体的实现方法与步骤。在保证事务正确执行前提下，如何尽可能地提高系统的并发执行速度，是一个具有研究价值和实用价值的课题，值得我们进一步的去探讨。

参考文献

1 郑振楣，于戈，郭敏，分布式数据库．科学出版社，1998

2 Yao W，Perrizo W，He X. An Improved Algorithm for Concurrency Control in Distributed Database System. Information Science，1997，103

3 袁崇义．Petri网原理．电子工业出版社，1998

(上接第73页)

(B,E;F)为简单有向网，称为Σ的基网。

令Cp(B)是丛上的完全情态集，C中的丛称为Σ的情态；

b B c1,c2 C:b c1 b c2，即B中每个事件都有机会成真，也有机会成假；

e E c1,c2 C:c1[e>c2，即E中每个事件都有机会发生。

因而，四元组Σ=(B,E;F,C)能构成条件/事件系统(C/E-系统)。

在这个C/E-系统中，由于要求b1,b2,…,bn可以并发，因而Sb1,Sb2,…,Sbn中都应事先设有托肯。设条件集S1的容量函数K为i(0<i≤n+1)，也就是事件b1,b2,…,bn中有i个事件可以并发，i越大，能并发的事件越多，i越小，能并发的事件越少。

由于网模型中$\sigma(a,b1)=0(a,b2)=\cdots=\sigma(a,bn)=1$，因此满足要求(3)，事件a与事件b1必须交替，事件a与事件b2必须交替……事件a与事件bn必须交替。

结束语 本文运用通用网论的同步论，针对基于Internet的远程教育中的远程交互问题建立了几个远程交互的网模型，并分析了这些网模型中关键事件的同步距离与其系统行为之间的关系。进一步的研究工作将是如何在我们正在开发的远程教育软件中实现。

参考文献

1 袁崇义．Petri网原理，电子工业出版社，1998

2 刘立人．并发程序设计．上海科技文献出版社，1996

3 杨文龙，姚淑珍，Petri网在FMS中的应用．计算机世界，1997年7月

4 徐心平．远程教育及其分类．微电脑世界，1998年6月