

文章编号:1001-9081(2007)05-1092-03

移动计算中基于权值的自适应事务处理模型

李智超,吕晓峰,郗晨霞,何 涛

(中共中央党校 信息管理部,北京 100091)

(lizhch78@yahoo.com.cn)

摘 要:在综合分析现有事务模型的基础上,提出了一种根据事务提交所处的网络环境和事务是否访问热点数据,自动选择子事务处理模型的基于权值的自适应移动事务处理模型——AMTMBW。实验证明,新的事务处理模型在移动事务的撤销率,移动事务的执行时间等方面与其他模型相比,其性能都有较大的提高。

关键词:移动计算;移动事务处理模型;预写事务模型;AMTMBW 模型

中图分类号: TP393 **文献标识码:** A

Adaptable mobile transaction model based on weight in mobile computing

LI Zhi-chao, LÜ Xiao-feng, QIE Chen-xia, HE Tao

(Department of Information Management, China Central Party School, Beijing 100091, China)

Abstract: Mobile transaction model is one of the key technologies. Based on the analysis of the present transaction models, this paper proposed an adaptable mobile transaction model based on weight, which can itself choose sub-transaction model according to the transaction's network and whether the transaction visits the hot data. The results of experiment show that new model has better performance at mobile transaction redo ratio and executing time than other models.

Key words: mobile computing; mobile transaction model; pre-write transaction model; Adaptable Mobile Transaction Model Based on Weight model(AMTMBW)

0 引言

移动计算中,由移动客户机发出的事务称为移动事务。移动事务实际上是一种分布式事务,其中部分计算工作放在客户机上完成,而其他部分则交给固定节点(服务器)完成。移动事务处理是移动数据库系统的关键技术,主要用于维护数据的一致性,支持多用户的并发访问,使用户可以可靠地查询和更新数据库。移动环境的特殊性决定了移动事务处理具有以下几个特点:移动性、长事务、易错性、异构性。总之,移动事务是不同于传统事务的一类新事务,且要访问更加复杂的数据资源,因此需要研究适应移动计算环境的新事务处理模型^[1]。

在移动事务处理方面,人们已经提出了很多处理模型,大体上可以分为两类:一类模型如集群事务模型^[2],Kangaroo 事务模型等,对事务的 ACID 性质做了一些折中或者重新定义,因此不能完全保证事务的可串行性和数据库的一致性;另外一类模型如 MDSTPM 事务处理模型^[3]以及在乐观两段锁协议基础上提出的 O2PC-MT 移动事务处理模型,能够保证事务的 ACID 性质。但是上述事务处理模型都是针对某一种具体情况的,比如 O2PL、预写事务模型^[4]等针对的是移动计算环境中断接情况下的事务处理机制。对于现实环境中的移动终端,每个移动用户所提交事务的网络状态,即可能处在断接状态,也可能处在连接状态,单一应用上述模型都不能取得很好的效果。所以本文提出一种综合现有模型优点的事务处理模型——基于权值的自适应移动事务处理模型(AMTMBW)。

1 AMTMBW 模型

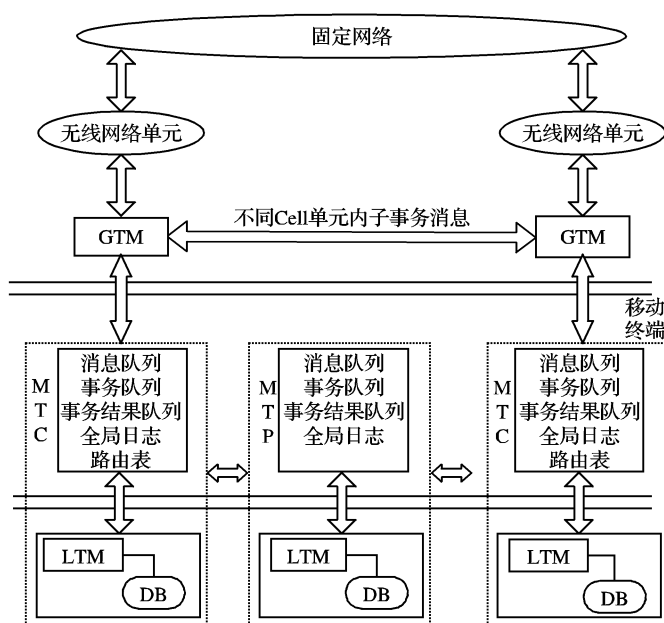


图1 AMTMBW 移动事务处理模型

AMTMBW 模型的体系结构是一种基于支持移动终端的数据库系统。AMTMBW 模型在各个移动终端的事务基础上,增加了一层全局事务管理层,用来管理当移动事务出现跨区情况时,对一个移动事务的各个子事务的协调处理,其体系结构如图1所示。

收稿日期:2006-11-08;修订日期:2007-01-19

作者简介:李智超(1978-),男,工程师,博士,主要研究方向:移动计算、数据挖掘;吕晓峰(1978-),男,工程师,主要研究方向:卫星通信;郗晨霞(1974-),女,高级工程师,硕士,主要研究方向:卫星通信;何涛(1966-),男,高级工程师,硕士,主要研究方向:卫星通信。

图1中各模块的作用如下:

GTM:全局事务管理器,负责对每个事务进行全局管理,当一个事务发生跨区分解时,该模块负责同不同Cell内的GTM互相通信,协调各个子事务的提交和结果返回。

MTC:移动事务控制器,负责对移动终端的事务进行管理,包括对移动终端消息队列管理、各个事务按权值排队、事务返回的结果队列管理,并且负责发生事务跨区情况下的事务分解,对于不同区域的子事务可以通过路由表进行转发处理。

MTP:移动事务参与控制器,具有除了MTC的路由功能外的其他功能。

LTM:本地缓存事务管理器,负责本地事务或者锁分配的全局子事务的实际执行以及故障恢复。

AMTMBW模型的设计思想是根据移动网络连接好坏和所访问的数据是否是热点数据,将移动事务分为四类,分别是连接网络下的高权值事务、连接网络下的普通权值事务、断接网络下的高权值事务、断接网络下的普通权值事务。移动事务提交的时候,判断自己事务的类型,自动选择相应类型的处理模式进行事务处理,这样提高了事务执行的成功率并且减少了移动用户的通信代价。根据各种模型的特点和上述四类事务提交情况的需要,这里设定三种子事务处理模式,分别对应上述四种事务类型:

1) 预写事务处理模式:该模型主要针对连接网络下的高权值事务处理,因为在网络连接的情况下,移动终端对热点数据的访问比较频繁,移动终端一旦修改了该数据,通过预写事务处理模式,对该数据进行预提交后,其他移动终端就可以看到该数据,通过这种方式可以提高数据处理的实时性。当然用这种模式处理热点数据,也带来了一个问题,就是热点数据在预提交状态下,如果又被其他终端修改,如何来维护其数据修改的顺序和有效性?我们采用时间戳的概念来处理不同MSS下对同一个数据项的修改提交,具体过程如下:给每一个事务加上一个时间戳,这样MSS的GTM在预写处理模式下,如果要对该事务进行预提交,首先该TM先向服务器发送一个带有该数据项时间戳的修改请求消息,服务器查找排队的事务日志中是否有对该数据项的修改请求消息,如果有,比较两个消息的时间戳,服务器发送取消修改消息给时间戳较早的GTM,同时发送确认修改消息给时间戳较晚的GTM,这样收到取消修改消息的GTM就地取消对该数据项的预提交,而收到确认修改消息的GTM可以对数据项进行预提交。通过时间戳机制,就可以避免不同GTM对同一数据项修改预提交所带来的移动事务数据冲突,减少事务回滚。

2) 乐观两阶段提交移动事务处理模式:O2PC-MT事务处理模式主要针对网络连接下的普通权值事务处理和断接

好的性能,但是这里需要对事务提交进行读、写冲突的检测。为此定义四种锁类型,分别是:预读锁、预写锁、读锁、写锁。其锁相容矩阵如表1所示。

3) 就地夭折事务模式:该模式主要针对断连网络下高权值事务处理,对于这类事务,移动终端在网络连接后,不需要对本事务进行提交,而是在本地就地夭折。这是因为高权值事务操作的是热点数据,而热点数据的修改是极为频繁的,在移动终端断连时间内,其他移动终端对该数据将进行多次的操作,所以网络连接后,再次提交该事务已经没有意义。该处理方式可以有效的减少重新连接后移动事务的回滚率,从而降低网络通信代价。

移动终端发送给MSS的读写操作规定以BEGIN(MTID, MODE, WEIGHT, TS)开始,以COMMIT或ABORT结束。这里MTID为每个移动终端对每个事务的移动事务标识(MTID),规定MTID的格式为 $MTID = MCID + N$,这个N是移动终端对所发送的事务操作依次从0到N进行编号,MCID为移动终端在Cell中的终端标识。MODE为上面讨论的三种子模式,这里定义0代表预写事务处理模式,1代表O2PC-MT处理模式,2代表就地夭折事务处理模式。TS为该移动事务的时间戳,用来记录该事务的时间。WEIGHT代表每个移动事务开始的权值,该权值的获得应该根据数据在数据库中的访问频率来决定。数据库数据在初始的状态下,每个数据的权值都是相等的,但是随着数据库数据的使用,将有20%的热点数据被频繁访问。为了计算出数据的权值,在服务器上对每个数据项的更新进行计数,记为W,每更新一次数据项,对应的计数器数值加1。为了能更好地反映近期内它们更新的频率,周期性的对计数器数值减1。权值在事务处理分类上,有高低之分,这里可以根据数据库实际情况,设定一个阈值W,大于这个阈值的事务定义为高权值事务,低于这个阈值的为普通权值事务。阈值可以根据需要实时修改,其数值通过广播的形式发送给各个移动终端。移动终端在从服务器下载数据的同时,也下载了相应数据项的权值。

通过上面的分析,AMTMBW模型的移动事务处理模型可以以下算法进行描述。

1. 产生一个移动事务T,自动赋予标识并交MU处理;
2. MU分析该事务,确定其进行的操作(如读、写),并分析其操作数据项的权值,确定相应的处理模式和时间戳;
3. BEGIN(MTID, MODE, TS, WEIGHT)
4. Case MODE = 0; //使用预写模式进行处理
5. 对操作事务加锁,进行本地操作;
6. 初始提交给GTM,并使操作数据项立刻全局可见,GTM协调所有子事务完成后,根据权值进行队列排队,进行预提交;
7. GTM发送修改请求AR(Alter Request)给服务器;
8. Server根据TS查找排队事务日志log,发出相应消息;
9. if 消息为CAR(Conform Alter Request)
10. 事务预提交成功,将该事务加入Server排队日志;
11. Server处理结束后,发送确认消息;
12. 事务Commit结束;
13. if 消息为AAR(Abort Alter Request)
14. 事务在GTM取消预提交,Abort结束;
15. Case MODE = 1; //使用O2PC-MT模式处理
16. 对操作事务加锁,进行本地操作;
17. 初始提交给GTM,GTM协调所有子事务完成后,并根据权值进行队列排队,进行预提交;

表1 AMTMBW锁矩阵

	Pre-read	Pre-write	Read	Write
Pre-read	是	否	是	是
Pre-write	否	否	是	是
Read	是	否	是	否
Write	是	是	否	否

情况下的普通权值事务处理,因为该模型采用了乐观并发控制和两阶段提交结合的方法,可以很好地支持弱事务处理,允许多次发送事务和事务跨区的操作。该模式比O2PL具有更

18. 进行读写和执行时冲突检测;
19. if 无冲突
20. 事务预提交成功;
21. Server 处理结束后, 发送确认消息;
22. 事务 Commit 结束;
23. if 有冲突
24. 事务在 GTM 取消预提交, Abort 结束;
25. Case MODE = 2; //使用就地夭折处理模式
26. 对操作事务加锁, 进行本地操作;
27. 重新网络连接后, 直接对该事务取消提交, 以 Abort 结束;

2 AMTMBW 模型性能分析

AMTMBW 模型结合了 O2PC-MT 模型和预写处理模型优点: 使用两阶段提交算法, 对热点数据项的修改使用时间戳检测冲突, 对普通数据项的修改使用严格的读写冲突和执行时冲突检测, 即可以充分提高热点数据的实时性, 也可以保证数据项的 ACID 性质, 保证了事务的可串行化; 引入了权值和时间戳概念, 保证热点数据处理的优先性; 引入队列机制和消息机制, 保证了预提交的顺利进行和事务发送方式的灵活性。为了对上述几种事务模型进行性能比较, 本文设计了详细的模拟实验, 具体模块包括: 一个普通事务发生器 (记为 PTG); 一个移动事务发生器 (记为 MTG); 一个普通事务处理器 (记为 PTP); 三个移动事务处理器: AMTMBW 事务处理器 (记为 AMTP)、O2PC-MT 事务处理器 (记为 O2TP)、预写事务处理器 (记为 PWTP)。整个实验的系统参数主要如表 2 所列。

表 2 模拟实验中主要系统参数

参数	意义	数值
CPU	模拟计算机的处理能力(Mips)	2.8GHz
CacheSize	模拟计算机的内存大小	1GB
U/Dlinkband	无线网络上下行带宽	19.2/144kbps
FixMsgIns	发送和接收一条消息的固定部分协议开销	64B
DBSIZE	数据库大小	2000 个元组
TupleSize	数据库元组的大小	128B
Tsize	平均每个事务操作数	4~8 个
MTIDSize	每个事务标识的大小	6B
Pr	只读事务所占的比例	0.3~0.6
Pu	在更新类事务中, 更新操作所占比例	0.3~0.6
MPL	服务器上同时并发的任务数	1~100
Delay	移动事务引入的延迟	10~500ms

MTG 通过随机数发生器生成 2000 个移动事务, 其中 80% 的移动事务访问数据库 20% 的热点数据。首先 1000 个移动事务作为系统的预热数据, 不记入实验统计数据, 其中 Pr 和 Pu 所占移动事务比例为 50%, 并且定义 Delay > 250ms 的事务为长事务。移动事务撤销率的实验中, 长事务在移动事务中的比例从 10% 到 80% 之间变化, 其实验结果如图 2 所示。

从实验结果可以看出, 经过预热的 AMTMBW 处理模型, 随着长事务比重的增加, 事务的撤销率近似线性减少, 这说明 AMTMBW 具有更好的事务执行成功率, 可以避免较多事务的回滚所带来的网络资源的浪费和系统资源的占用。当长事务比重达到 100% 的时候, 系统的撤销率会近似为 0, 这是因为所有的移动事务都在本地执行, 连接网络后就地夭折。

下面的实验验证移动事务响应时间与事务更新率之间的关系, Pu 的取值为 20% 到 80% 之间, 主要验证的是移动事务响应时间与 Pu 变化之间的关系, 结果如图 3 所示。

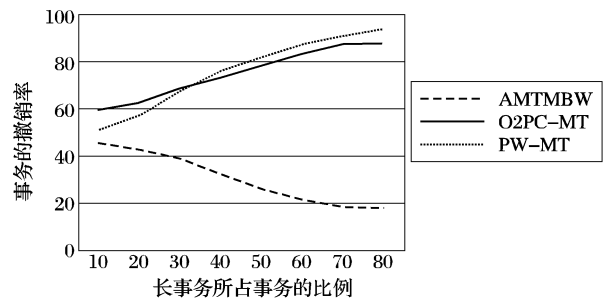


图2 三种模型事务撤销率

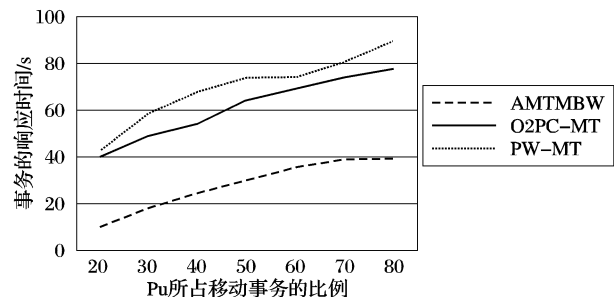


图3 三种模型事务响应时间与 Pu 关系图

图3与图2比较可以看出, 在对具有热点数据项特性的数据库事务处理情况下, AMTMBW 比其他两种模型具有较大的性能提高。这是因为在处理移动事务过程中, AMTMBW 模型考虑了热点数据项事务的实时性, 使用预写模型提高了数据的可用性, 并且通过时间戳协议和消息机制, 很好地控制了事务并发时的冲突, 并且提前了热点数据事务的处理顺序, 使得修改频繁的数据事务能得到优先处理, 从而提高了整个系统的性能。

3 结语

本文提出了一种新的自适应移动事务处理模型, AMTMBW 模型在对移动事务灵活性的支持、移动事务的吞吐率、移动事务的撤销率, 移动事务的执行时间等方面的性能相比其他两种模型都有较大的提高。但是该模型也存在一些不足, 主要是增加了 GTM 模块后, 需要增加不同 Cell 之间子事务的同步消息开销, 并且在进行预提交方面, 增加了服务器和 GTM 之间消息的开销。

参考文献:

- [1] CHRYSANTHIS PK. Transaction Processing in Mobile Computing Environment[A]. IEEE Workshop on Advances in Parallel and Distributed Systems[C]. 1993. 77-82.
- [2] PITOURA E, BHARGAVA B. Maintaining Consistency of Data in Mobile Distributed Environments[A]. Proceedings of 15th International Conference on Distributed Computing Systems[C]. 1995.
- [3] YEO LH, ZASLAVSKY A. Submission of Transactions from Mobile Workstations in a Cooperative Multidatabase Processing Environment [A]. Proceedings of the 14th IEEE CS International Conference on Distributed Computing Systems[C]. 1994. 372-379.
- [4] MADRIA SK, BHARGAVA B. A Transaction Model to Improve Data Availability in Mobile Computing[J]. Distributed and Parallel Databases, 2001, 10(2): 127-160.