一种基于乐观方法的安全实时并发控制协议

肖迎元[1] 陈向阳[2] 刘小峰[3] 邓华锋[3]

(天津理工大学计算机科学与工程系 天津 300191)[1]

(武汉工程大学计算机科学与工程系 武汉 430073)[2] (华中科技大学计算机学院 武汉 430074)[3]

摘 要 实时数据库通常应用在一些安全关键类应用中,如电子商务、股票交易、军事指挥系统等。在这样一些应用中,实时数据库系统需同时满足两方面的需求:确保数据安全和尽可能减低实时事务错过截止期的比率。然而,通常这两方面需求是相互冲突的,满足一方面是以牺牲另一方面为代价。本文提出了一种基于乐观方法的安全实时并发控制协议,该协议将安全约束整合到实时乐观并发控制协议中,并能根据应用的需求在安全性和实时性方面进行了适当的折中。性能测试结果显示,该协议在确保数据安全的同时并未明显地降低实时性能。

关键词 实时数据库,并发控制,安全,隐通道

A Secure Real-time Concurrency Control Protocol Based on Optimistic

XIAO Ying-Yuan[1] CHEN Xiang-Yang[2] LIU Xiao-Feng[3] DENG Hua-Feng[3]

(Department of Computer Science and Engineering, Tianjin University of Technology, Tianjin 300191)[1]

(School of Computer Science and Engineering, Wuhan Institute of Technology, Wuhan 430073)[2]

(School of Computer Science and Technology, Huazhong University of Science and Technology, Wuhan 430073)[3]

Abstract Real-time database are usually needed in some safety-critical applications such as e-commerce, stock trading systems and military systems. In these applications, real-time database systems must simultaneously satisfy two requirements in guaranteeing data security and minimizing the missing deadlines ratio of transactions. However, these two requirements can conflict with each other and to achieve one requirement is to sacrifice the other. This paper presents a secure real-time concurrency control protocol based on optimistic method. The concurrency control protocol incorporates security constraints in a real-time optimistic concurrency control protocol, and can make a suitable tradeoff between security and real-time requirements according to the needs of applications. Performance tests show that the concurrency control protocol achieves data security without sacrificing real-time performance significantly.

Keywords Real-time database, Concurrency control, Security, Covert channel

1 引言

在实时数据库中,事务和数据都可能具有定时限制,系统的正确性不仅依赖于事务产生的逻辑结果,还依赖逻辑结果产生的时间。事务的定时限制典型地表现为事务的截止期,一个实时事务若在规定的截止期后完成,结果将变得毫无价值(对固实时事务而言),甚至还可能带来灾难性的后果(对硬实时事务而言),因此,针对实时数据库,在并发控制策略上必须优先确保截止期紧迫事务(高优先级事务)获得所需数据资源,以尽可能减低实时事务错过截止期的比率,来确保系统实时性能。

实时数据库通常应用在一些安全关键类应用中,如电子商务、股票交易、军事指挥系统等。对这类应用而言,除了确保实时性能,阻止不同事务间非法的信息流也是至关重要的。传统的实时并发控制协议,如 HP2PL(高优先级两段锁)[1]、PC(优先级顶)[2]、OPT-SACRIFICE [3]、OPT-WAIT[3]等,都着重于如何减低实时事务错过截止期的比率,并没有考虑到安全性需求,因而不能避免事务间利用并发控制机制绕过系统安全检查进行隐密通信,即隐通道问题。而大多数安全数据库系统又都未考虑到事务和数据可能存在的时间约束,因而无法确保实时性能。因此,研究既能达到应用所需的安全需求又能确保系统实时性能的安全实时并发控制协议具有重要的理论与现实意义。

一些安全实时并发控制协议已经被提出[4~7],但是这些协议都未能在安全与实时性两方面达成很好的折中,通常都是强调一方面而忽略了另一方面。本文在现有的乐观实时并发控制协议的基础上,通过引入安全违背因子和实时影响因子提出了一种整合了安全约束的基于乐观方法的安全实时并发控制协议(OSRTCC),OSRTCC 同时考虑了安全性与实时性的需求,并能在安全性与实时性方面进行适当的折中来更好地满足应用的需求。

2 隐通道问题

大多数安全数据库系统采用了基于 Bell-LaPadula 模型的存取控制机制。Bell-LaPadula 模型将系统中元素按其特性区分为主体(Subject)和客体(Object)。主体是系统中的主动元素,能执行一系列的动作,如进程、事务。客体是系统中包含信息的被动元素,如关系、元组等。系统中每一客体被赋予一安全级别(Security Level),客体的安全级别反映了存储在客体内信息的敏感度,也反映了未经授权向不允许存取该信息的用户泄露这些信息造成的潜在损害度。每一主体根据其置信度(被系统信任的程度)被授予一许可级(Clearance level)。Bell-LaPadula 的存取限制("向下读"和"向上写")能阻止不同许可级事务间直接的非法信息流,但不能阻止并发事务间利用并发控制机制合谋进行隐秘的非法信息传输(本文称之为并发控制隐通道)。比如一个高许可级的事务持有一

肖迎元 博士,主要研究方向为现代数据库理论与集成实现,信息系统设计与开发等;陈向阳 讲师,主要研究方向为计算机网络;刘小锋 博士,主要研究方向为时空数据库。

数据对象上的排它锁，另一与它合谋的低许可级的事务同时申请该数据对象的锁，由于数据冲突，低许可级的事务被阻塞，那么低许可级事务可以根据被阻塞时间，从高许可级事务获取信息。这样在高许可级的事务和低许可级的事务间就形成了并发控制隐通道。

一个安全实时并发控制协议应该能避免并发控制隐通道。现有的实时并发控制协议都没有考虑可能存在的并发控制隐通道问题，因而也无法避免并发控制隐通道。由于仅当低许可级事务被高许可级事务阻塞时才可能导致并发控制隐通道，因此，对上述并发控制协议的一个改进的策略是：当并发的不同许可级事务发生数据冲突时优先低许可级事务。比如当一个低许可级事务申请被某一高许可级事务以排它方式持有的数据资源时，改进策略采用夭折高许可级事务，以确保低许可级获得所需数据资源，从而避免了可能产生的并发控制隐通道。上述改进策略的问题是可能导致高优先级事务被低优先级事务夭折(优先级倒置)，从而严重影响了系统实时性能。因此，这种以牺牲实时性能为代价的改进策略并不能有效地同时满足应用在安全性和实时性两方面的需求。

3 安全实时并发控制策略

3.1 实时乐观并发控制协议

类似于传统的乐观并发控制协议，实时乐观并发控制协议也将事务的执行过程分为三个阶段：读阶段、验证阶段和写阶段。实时乐观并发控制协议不同于传统的乐观并发控制协议的是在验证阶段引入了优先级牺牲机制。事务抵达它的验证阶段，进行冲突检测，若未检测到冲突，验证事务被允许提交。若冲突被检测到，如果冲突集中存在一个或多个比验证事务具有更高优先级的事务，则验证事务被夭折，即以牺牲验证事务来保证高优先级事务截止期的满足；如果冲突集中不存在比验证事务更高优先级的事务，则验证事务被允许提交，冲突集中事务被夭折。OPT-WAIT 协议通过引入优先级等待机制对上述实时乐观并发控制协议进行了改进，根据 OPT-WAIT 协议，当验证事务发现冲突集中存在更高优先级的事务，该验证事务并不立即被夭折，还是被限制进入阻塞状态(等待状态)，若冲突集中某个更高优先级事务被允许提交，则处在阻塞状态的验证事务被夭折；反之，若冲突集中不再存在比验证事务更高优先级的事务，则验证事务被允许提交。显然，OPT-WAIT 给较高优先级事务一个优先满足其截止期的机会，有利于提高实时性能。但是由于整合了优先级等待机制，OPT-WAIT 不能避免并发控制隐通道。我们在 OPT-WAIT 的基础上，通过引入安全检查机制，提出了 SRTCCO。

3.2 安全违背因子和实时影响因子

我们通过定义安全违背因子和实时影响因子来分别描述违背安全性的严重程度和影响实时性的严重程度。在下面的定义中，我们用 $CL(T)$ 表示事务 T 的许可级；$P(T)$ 表示 T 的优先级；ST 表示进入系统的事务的集合。

定义 1 $\forall T_i, T_j \in ST$，若 T_i 和 T_j 存在一对冲突操作，则称 T_i 和 T_j 为冲突事务对，记为：$T_i\ CF\ T_j$。

定义 2 假定 $T_i \in ST$，$ST_1 \subseteq ST$，如果条件：$\forall T_j \in ST_1(T_i\ \mathrm{CF}\ T_j)$ 成立，则称 ST_1 为 T_i 的冲突集，记为：$CS(T_i)$。

定义 3 假定 $T_i \in ST$，$ST_2 \subseteq ST$，如果条件：$(\forall T_j \in ST_2(T_i\ \mathrm{CF}\ T_j)) \wedge (T_j \in (ST - ST_2)\ (\neg(T_i\ \mathrm{CF}\ T_j)))$ 成立，则称 ST_2 为 T_i 的最大冲突集，记为：$MCS(T_i)$。

定义 4 定义 $|f(CL(T_i)) - (f(CL(T_j))|$ 为 T_i 和 T_j 的许可级差异度，记为：$CDD(T_i, T_j)$。这里，f 是许可级的取值域到自然数集的一个映射。

定义 5 定义 $|P(T_i) - (P(T_j)|$ 为 T_i 和 T_j 的优先级差异度，记为：$PDD(T_i, T_j)$。

定义 6 假定 $T_i \in ST$，$MCS1(T_i) \subseteq MCS(T_i)$，若条件：$(\forall T_j \in MCS_1(T_i)(CL(T_j) > CL(T_i))) \wedge (\forall T_j \in (MCS(T_i) - (MCS_1(T_i))(CL(T_j) \leqslant CL(T_i)))$ 成立，则称 $MCS_1(T_i)$ 为 T_i 的高许可级最大冲突集，记为：$HCMCS(T_i)$。

定义 7 假定 $T_i \in ST$，$MCS_2(T_i) \subseteq MCS(T_i)$，若条件：$(\forall T_j \in MCS_2(T_i)\ (P(T_j) < P(T_i))) \wedge (T_j \in (MCS(T_i) - (MCS_2(T_i))(P(T_j) \geqslant P(T_i)))$ 成立，则称 $MCS_2(T_i)$ 为 T_i 的低优先级最大冲突集，记为：$LPMCS(T_i)$。

定义 8 定义 $\sum_{T_j \in HCMCS(T_i)} (|f(CL(T_i)) - (f(CL(T_j))|)$ 为 T_i 的安全违背因子，记为：$SVF(T_i)$。

$SVF(T_i)$ 反映了在冲突解决中让 T_i 处在阻塞状态导致的对安全性的违犯的严重程度。

定义 9 定义 $\sum_{T_j \in LPMCS(T_i)} (|P(T_i) - (P(T_j)|)$ 为 T_i 的实时影响因子，记为：$RIF(T_i)$。

$RIF(T_i)$ 反映了在冲突解决中夭折 T_i 对实时性能影响的严重程度。

3.3 安全实时乐观并发控制协议(OSRTCC)

OSRTCC 与 OPT-WAIT 的主要区别是 OSRTCC 在验证阶段引入了安全检查机制以实现在确保实时性能的情况下仍能满足应用所需的安全需求。在 OSRTCC 中，根据下面给出的准则决定验证事务是被夭折、被阻塞还是被允许提交。在下面准则描述中，我们用 T_i 表示验证事务；$CT(T_i)$ 表示 T_i 被允许提交；$BT(T_i)$ 表示 T_i 被阻塞；$AT(T_i)$ 表示 T_i 被夭折；ω 和 $(1-\omega)$ 分别表示安全性和实时性的权值，用来表示各自的重要性程度。

准则 1 若条件：$(\forall T_j \in MCS(T_i)(P(T_i) \geqslant P(T_j)))$ 满足，则 $CT(T_i) \wedge AT(T_j)$。

准则 2 若条件：$((\exists T_j \in MCS(T_i)(P(T_i) < P(T_j)) \wedge (\forall T_j \in MCS(T_i)(C(T_i) \geqslant C(T_j))))$ 满足，则 $BT(T_i)$。

准则 3 若条件：$((\exists T_j \in MCS(T_i)(P(T_i) < P(T_j)) \wedge (\exists T_j \in MCS(T_i)(C(T_i) < C(T_j))) \wedge (\omega \times SVF(T_i) \geqslant (1-\omega) \times RIF(T_i)))$ 成立，则 $AT(T_i)$。

准则 4 若条件：$((\exists T_j \in MCS(T_i)(P(T_i) < P(T_j)) \wedge (\exists T_j \in MCS(T_i)(C(T_i) < C(T_j))) \wedge (\omega \times SVF(T_i) < (1-\omega) RIF(T_i)))$ 成立，则 $BT(T_i)$。

因为仅当低许可级事务被高许可级事务阻塞时才可能导致并发控制隐通道，所以遵循准则 1 和准则 2 不会导致并发控制隐通道。当存在并发控制隐通道的可能时，准则 3 和准则 4 给出了在安全性和实时性两方面折中的原则。基于上述准则，OSRTCC 中事务 T_i 在验证阶段的处理过程可描述如下：

```
if (∀Tj∈MCS(Ti)(P(Ti)≥P(Tj))) then CT(Ti) and AT(Tj);
else
  if (∀Tj∈MCS(Ti)(C(Ti)≥C(Tj))) then BT(Ti);
  else
    if (ω×SVF(Ti)≥(1−ω)×RIF(Ti)) then AT(Ti);
    else BT(Ti);
end if
```

4 性能测试与结果

在性能仿真实验中我们主要将 OSRTCC 与 OPT-WAIT 进行了比较。主要性能指标为：事务错过截止期的比率

(下转第 128 页)

方法。高级分析方法包括:改善度分析、相关分析以及差异显著性检验。改善度分析是一种相对较为复杂的分析方法,又称为相互影响度分析。它用于说明类型之间的相互影响程度;相关性分析用于说明可计算类型 i 与可计算类型 j 之间关系的密切程度;差异显著性检验通过对各种类型的指标值的多重比较。找出具有显著大或者显著小或一般情况等特征的类别。

这些基本分析方法可以满足用户大多数情况下的分析要求。此外,用户也可以综合运用这五种分析方法进行复杂的统计分析。

3.6 用户接口

XOLDAS 框架有二个用户接口:用户分析需求和分析结果界面。

用户分析需求接口用来接受用户的分析请求,通过业务空间,用户可能方便地选择或输入期望得到的结果,业务空间层在组合其需求后调用方法库中的方法进行数据分析服务。

分析结果界面接口是向用户反馈数据分析结果,用户也可在该界面中灵活地变换分析需求得到新的结果以及图形曲线。

结束语 XOLDAS 框架是建立在业务空间和方法库基础上的,通过对业务空间的重新定义,可以快速建立一个新业务的 OLAP 系统,通过对方法库的扩充,可以方便地增加数据分析方法,因此系统具有良好的快速开发能力和可扩展性。

参 考 文 献

1 Thomsen E. OLAP Solution : Building Multidimensional Information System. 北京:电子工业出版社,2004. 4~5

2 Nguyen T B, Tjoa A M, Wagner R R. An Object Oriented Multidimensional Data Model for OLAP. In: Proc. WAIM. Shanghai, China, June 2000. 130~132

3 徐忠健,袁捷,陆菊康,陈毛狗. 超市决策支持系统的数据仓库的设计与实现. 计算机工程与应用,2003(18):56~57

4 肖昭媛. 统计学原理与应用. 上海:上海交通大学出版社,2002

(上接第 115 页)

(MDR)和每 5 秒钟内低许可级事务被高许可级事务阻塞的数目(LBN)。其中 MDR 的定义如下:MDR=(错过截止期事务数)/(系统接纳的事务数)。MDR 反映了系统实时性能,而 LBN 则用来衡量违犯安全性的严重程度。在我们的仿真实验中,优先级分派采用 EDF(Earliest Deadline First)策略。事务的截止期(DL)按如下公式计算:$DL=AT+SF\times ET$,这里,AT 表示事务的到达时间;SF 表示松弛因子,为满足均匀分布的随机变量;ET 表示事务的估计执行时间。仿真测试的主要参数设置如表 1 所示。

表 1 仿真测试参数

参数	值	含义
R	[5, 40]	事务到达率
Pu	0.4	更新操作的概率
SF	U[2.0, 6.0]	松弛因子
NCL	6	不同许可级的级别数目
RCL	1/6	不同许可级事务的比率
ω	0.5	确保数据安全的权值

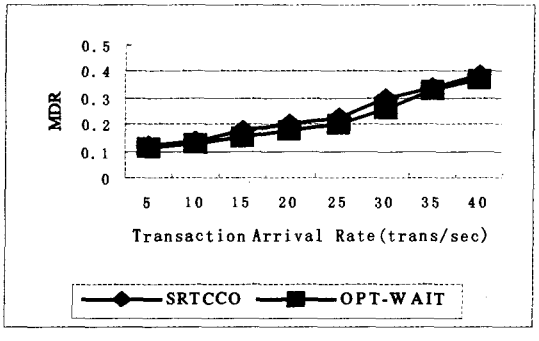

图 1 MDR 的比较

在表 1 中,$U[i, j]$表示在区间$[i, j]$上满足均匀分布的随机变量。性能测试结果如图 1 和图 2 所示。

图 1 显示了 OSRTCC 和 OPT-WAIT 在 MDR 方面的比较,随着事务到达率的增大,OSRTCC 和 OPT-WAIT 的 MDR 都相应增加,但 OPT-WAIT 稍稍优于 OSRTCC。图 2 给出了 OSRTCC 和 OPT-WAIT 在 LBN 方面的比较,显然,相对于 OPT-WAIT,OSRTCC 具有明显的优势。

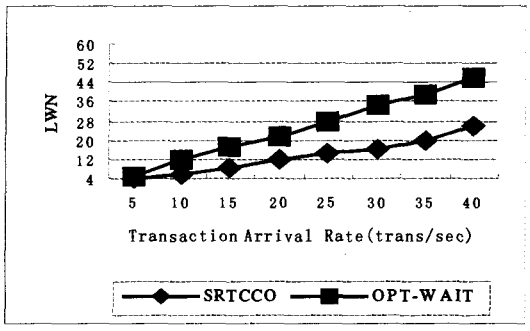

图 2 LWN 的比较

结束语 一个安全实时数据库系统必须同时满足两个目标:确保数据安全和尽可能减低实时事务错过截止期的比率。然而,这两方面需求通常是相互冲突的,满足一方面是以牺牲另一方面为代价。本文提出了一种基于乐观方法的安全实时并发控制协议(OSRTCC)。OSRTCC 在验证阶段引入了安全检查机制,并通过定义安全违背因子和实时影响因子来有效地实现在安全性和实时性两方面的折中。仿真实验显示,OSRTCC 在确保数据安全的同时仍能保持良好的实时性能。

参 考 文 献

1 刘云生. 现代数据库技术. 北京:国防工业出版社,2001

2 Sha L, Rajkumar R, Son S H, et al. A real-time locking protocol. IEEE Transactions on Computers, 1991. 793~800

3 Haritsa J, Carey M, Livay M. Data access scheduling in firm real-time database systems. Real-Time Systems Journal, 1992(4)

4 George B, Haritsa J. Secure transaction processing in firm real-time database systems. ACM SIGMOD Record, 1997, 26(2): 462~473

5 David R, Son S H, Mukkamala R. Supporting security requirements in multilevel real-time databases. IEEE Transactions on Knowledge and Data Engineering, 2000. 865~879

6 David R, Son S H. A secure concurrency control protocol for real-time database. In: Proc. of Annual IFIP WG 11. 3 Conference of Database Security, August 1995

7 朱虹,冯玉才. 避免隐通道的并发控制机制. 小型微型计算机系统,2000,21(8):844~846