

并行数据广播中的亚可串行化并发控制

党德鹏

(北京师范大学信息科学与技术学院 北京 100875)

摘 要 考虑到移动计算环境的诸多限制以及移动数据库应用的特征,传统冲突可串行化在不对称通信的数据广播环境中显得过于严格而不必要.为此,文中提出一种并行亚可串行化并发控制协议.首先,形式化定义亚可串行化正确性准则,证明亚可串行化虽然比冲突可串行化宽松但能始终保证一致性,给出保证亚可串行化的充分必要条件.进而,提出相应的并行亚可串行化并发控制协议 BCC-HSR.实验结果表明,BCC-HSR 协议能有效改进移动事务的平均响应时间,更好地满足并行数据广播环境中高性能事务处理的要求.

关键词 数据库系统;移动数据库;移动事务;并发控制;并行数据广播

中图法分类号 TP311

Hypo-Serializable Concurrency Control in Parallel Broadcast Environments

DANG De-Peng

(College of Information Science & Technology, Beijing Normal University, Beijing 100875)

Abstract Serializable concurrency control protocols are inapplicable in mobile broadcast environments due to a number of constraints of wireless communications and characteristics of mobile real time applications. In this paper, the use of hypo-serializability for mobile parallel broadcast environments is proposed. At first, hypo-serializability is formally defined. Next, it is proved that hypo-serializability is more relaxing than traditional serializability while consistency is maintained all the time. Then, the necessary and sufficient conditions for hypo-serializability are shown. Furthermore, the parallel hypo-serializable concurrency control protocol BCC-HSR is proposed. Finally, through a series of simulation studies, it is confirmed that the proposed protocol could improve average response time of mobile transactions significantly.

Keywords database systems; mobile database; mobile transaction; concurrency control; parallel data broadcast

1 引 言

随着无线通信和移动计算机技术的发展,近些年出现了许多需要以广播模式发布数据来支持大量并发用户的数据库应用^[1-6].比如移动拍卖、股票交易、下一代交通管理系统、电子商务应用等.目前,已经可以在 Internet 上提交股票买卖事务,过不了多

久,股民就可以在移动客户机上进行股票交易.诸如航空客票服务等电子商务应用中,携带便携机的用户可通过信用卡随时随地购买任意航班的飞机票.这些应用都涉及大量(包括来自移动客户的)数据更新,在广播数据的同时更新事务有可能装入数据项的新值,若对(包括来自移动客户的)更新事务与广播的交替执行不加以控制,将导致严重数据不一致^[1-6].为保证数据一致性并立即一致地发布数据更

新,需进行有效的并发控制^[1-6].

现有的分布式并发控制机制都需要各分布实体之间的双向通信来协调事务的分布式执行,这使得它们(即使是实时的)完全无法适应非对称通信的广播环境^[1,5-12].文献[5,7-10]针对广播环境提出的并发控制方法仅仅考虑了在移动端有效处理只读事务的问题;文献[6,11-12]针对广播环境分别提出了在移动端支持更新事务的 PVTO 协议、O-PreH 协议和 BCC-SR 协议,它们都基于非常严格的冲突可串行化正确性准则.考虑到无线通信的诸多限制以及移动数据库应用的特征,传统冲突可串行化在非对称通信的数据广播环境中显得过于严格而不必要^[1-2,4-5,7].现有的放松可串行化的并发控制技术 ϵ -可串行化^[13]、 δ -可串行化^[14] 和准一致性可串行化^[15] 都会导致数据库的受限不一致^[13-15].为此,本文提出一种并行亚可串行化并发控制协议.首先,形式化定义亚可串行化正确性准则,证明了亚可串行化虽然比冲突可串行化宽松但能始终保证一致性,给出了保证亚可串行化的充分必要条件.进而,提出相应的并行亚可串行化并发控制协议 BCC-HSR.实验结果表明,BCC-HSR 协议能有效改进移动事务的平均响应时间,更好地满足并行数据广播环境中高性能事务处理的要求.

2 亚可串行化

用 s 表示一个数据对象的状态, o 表示定义在该对象上的操作, $return(s, o)$ 表示定义在该对象上的操作 o 的返回值, $state(s, o)$ 表示操作 o 执行后的状态.设有事务集 T 且 $T = T_u \cup T_r$,其中 T_u 表示 T 中更新事务的集合而 T_r 表示 T 中只读事务的集合, T 的执行经历用 H 表示, H_d 表示涉及数据对象 d 的操作的投影. $H_d = o_1 <_T o_2 <_T \dots <_T o_n$ 既表明了操作的执行顺序(o_i 先于 o_{i+1}),又表明了操作的功能复合.因此,一系列操作所产生的状态 s 也就等于对对象的初始状态 s_0 应用这些操作相应的经历 H_d 所产生的状态,即 $s = state(s_0, H_d)$.为了简洁起见,始终隐含假定初始状态为 s_0 ,并可用 H_d 表示一个数据项由 H_d 产生的状态^[13].设 $t \in T$,用 $D(t)$ 表示事务 t 的操作数据集;用 $D_w(t)$ 表示事务 t 的写数据集;用 $D_r(t)$ 表示事务 t 的读数据集.

定义 1. 如果由 H_d 产生的状态中:

$$(state(H_d <_T o_1, o_2) \neq state(H_d <_T o_2, o_1)) \vee (return(H_d, o_2) \neq return(H_d <_T o_1, o_2)) \vee$$

$$(return(H_d, o_1) \neq return(H_d <_T o_2, o_1)),$$

称 o_1 与 o_2 冲突,记作 $C(H_d, o_1, o_2)$.

定义 2. 令 $t_i, t_j \in T, t_i \neq t_j, (o_{t_i}[d] \xrightarrow{H} o'_{t_j}[d])$ 表示在 H_d 中 $o_{t_i}[d]$ 出现在 $o'_{t_j}[d]$ 的前面.今定义 T 上的一个二元关系 C_H 如下:如果 $\exists d \exists o, o' (C(H_d, o_{t_i}[d], o'_{t_j}[d]) \wedge (o_{t_i}[d] \xrightarrow{H} o'_{t_j}[d]))$,称 t_i 对 t_j 有关系 C_H ,记作 $(t_i C_H t_j)$.令 C_H^* 为 C_H 的传递闭包,即若 $[(t_i C_H t_j) \vee \exists t_k (t_i C_H t_k \wedge t_k C_H^* t_j)]$ 则 $(t_i C_H^* t_j)$.

H 是(冲突)可串行化的当且仅当: $\forall t \in T \rightarrow (t C_H^* t)$.令 H 中关系 C_H 对应的关系图(即串行图)记作 S_H ,依据定义 2 有定理 1^[13].

定理 1. H (冲突)可串行化当且仅当 S_H 中无环,即 H (冲突)可串行化的充分必要条件是其中的提交事务不构成冲突关系的圈.

定义 3. 令 $t, t_i, t_j \in T, t_i \neq t_j$,又令 $V_t^r(x), V_t^w(x)$ 表示事务 t 读、写 d 的操作结果.定义 T 上关于 H 的二元关系 R_H 如下:如果 $\exists d \exists o_{t_i}^r(d), o_{t_j}^w(d) \in H((V_{t_i}^r(d) = V_{t_j}^w(d)) \wedge (o_{t_j}^w(d) \xrightarrow{H} o_{t_i}^r(d)))$,称 t_i 对 t_j 有关系 R_H ,记作 $t_i R_H t_j$.令 R_H^* 为 R_H 的传递闭包,即若 $[(t_i R_H t_j) \vee \exists t_k (t_i R_H t_k \wedge t_k R_H^* t_j)]$ 则 $(t_i R_H^* t_j)$.

定义 4. 令 $t, t_i, t_j \in T, t$ 的相关事务集记为 $A_H(t)$ 定义为按如下规则推出的闭包:

(a) t 属于 $A_H(t)$;

(b) 若 t_i 属于 $A_H(t)$,则满足 $t_i R_H^* t_j$ 的所有 t_j 也属于 $A_H(t)$.

定义 5. 如果执行经历 H 中 $[(\forall t_i \in T_u \rightarrow (t_i C_{H_{T_u}}^* t_i)) \wedge (\forall t_j \in T_r, (\forall t_k \in A_H(t_j) \rightarrow (t_k C_{H_{A_H(t_j)}}^* t_k)))]$,称 H 亚可串行化.

定义 6. 令 $t \in T_r, t$ 的相关读图 $G_H^t = (E, V)$ 是一个有向图:

$$V = A_H(t);$$

$$E = E_1 \cup E_2;$$

$$E_1 = \{(t, t_i) \mid t_i \in V \wedge \exists d (o_{t_i}^r[d] \xrightarrow{H_{A_H(t)}} o_t^w[d])\}$$

$$= \{(t, t_i) \mid t_i \in V \wedge \exists d (o_{t_i}^r[d] \xrightarrow{H} o_t^w[d])\};$$

$$E_2 = \{(t_i, t_j) \mid t_i, t_j \in V \wedge t_j R_{H_{A_H(t)}} t_i\}$$

$$= \{(t_i, t_j) \mid t_i, t_j \in V \wedge t_j R_H t_i\}.$$

引理 1. 设 $\forall t_i \in T_u \rightarrow (t_i C_{H_{T_u}}^* t_i), \forall t_j \in T_r$,当且仅当 $G_H^{t_j}$ 中无环时 $S_{H_{A_H(t_j)}}$ 中无环.

证明. 先证必要性.依据定义 5、定义 6, $\forall t_j \in$

$T_r G_H^{t_j}$ 是 $S_{H_{A_H(t_j)}}$ 的子图, 显然 $S_{H_{A_H(t_j)}}$ 中无环则 $G_H^{t_j}$ 中无环。

再证充分性, 用反证法. 设 $\exists t_j \in T_r, G_H^{t_j}$ 中无环而 $S_{H_{A_H(t_j)}}$ 中有一个环. 有且只有两种情况: t_j 在或不在这个环中. 依据定义 3, 4, 有 $\forall t_k \in A_H(t_j) t_j R_H^* t_k$, 这意味着 $\forall t_k \in A_H(t_j), G_H^{t_j}$ 中必有一条从 t_k 到 t_j 路径. 又因为 $G_H^{t_j}$ 中无环, 则 $\forall t_k \in A_H(t_j), G_H^{t_j}$ 中必没有从 t_j 到 t_k 的边. 加之 $t_j \in T_r$, 则 $S_{H_{A_H(t_j)}}$ 中 t_j 的出度必为 0. 这与前一种情况 (t_j 在这个环中) 相矛盾. 现考虑后一种情况即 t_j 不在这个环中. 因为 $A_H(t_j)$ 中除 t_j 之外别无其它只读事务, 则 $S_{H_{A_H(t_j)}}$ 中的环必是由更新事务组成, 从而该环也必然出现在 $S_{H_{T_u}}$ 中, 这又与 $\forall t_i \in T_u \rightarrow (t_i C_{H_{T_u}}^* t_i)$ 相矛盾.

故, 引理 1 成立.

证毕.

定理 2. 执行经历 H 是亚可串行化的, 当且仅当:

$$(\forall t_i \in T_u \rightarrow (t_i C_{H_{T_u}}^* t_i)) \wedge (\forall t_j \in T_r \rightarrow \Omega(G_H^{t_j})).$$

其中 $\Omega(G_H^{t_j})$ 表示 $G_H^{t_j}$ 中有理.

证明. 依据引理 1、定义 5, 定理 2 显然成立.

证毕.

定理 3. 若 $(\forall t'' \in T_u \rightarrow (t'' C_H^* t''))$, $t(t \in T_r)$ 始终读取一致性状态当且仅当 G_H^t 中无环.

证明. 假设 t 的相关读图中出现环, 如图 1 所示, 则有

$$S_{P_0}(DB - D_w(t_1)) = S_{P_1}(DB - D_w(t_1));$$

$$S_{P_0}(D_w(t_1)) \neq S_{P_1}(D_w(t_1));$$

$$S_{P_1}(DB - D_w(t_2)) = S_{P_2}(DB - D_w(t_2));$$

$$S_{P_1}(D_w(t_2)) \neq S_{P_2}(D_w(t_2));$$

...

$$S_{P_{n-1}}(DB - D_w(t_n)) = S_{P_n}(DB - D_w(t_n));$$

$$S_{P_{n-1}}(D_w(t_n)) \neq S_{P_n}(D_w(t_n)).$$

所以

$$S_{P_0}(DB - \bigcup_{i=1,n} D_w(t_i)) = S_{P_n}(DB - \bigcup_{i=1,n} D_w(t_i));$$

$$S_{P_0}(\bigcup_{i=1,n} D_w(t_i)) \neq S_{P_n}(\bigcup_{i=1,n} D_w(t_i)).$$

并有 $\exists d_1 (d_1 \in \bigcup_{i=1,n} D_w(t_i) \wedge V_t^r(d_1) = S_{P_0}(d_1))$ 且

$$\exists d_2 (d_2 \in \bigcup_{i=1,n} D_w(t_i) \wedge V_t^r(d_2) = S_{P_n}(d_2)).$$

显然, t 读取了不一致性状态.

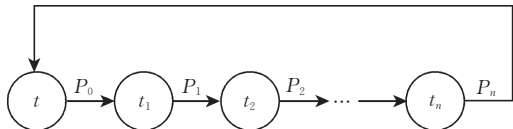


图 1 相关读图中的环

另一方面, 当 t 的相关读图中无环时, $\forall d \in D_r(t) (V_t^r(d) = S_{P_0}(d))$, 显然 t 读取了一致性状态.

证毕.

定理 4. 亚可串行化能使各并发事务读取、产生内部一致性状态, 保证数据库内部一致性.

证明. 依定义 5、定理 2、定理 3、定理 4 显然成立.

定理 5. 亚可串行化比冲突可串行化宽松.

证明. 依定理 1, H 是冲突可串行化的当且仅当 S_H 中无环. 当 S_H 中无环时, $\forall t \in T_r S_{H_{(t) \cup T_u}}$ 中无环, 依据定义 5 H 必是亚可串行化的; 然而, 当 S_H 中有环时 H 却也可能是亚可串行化的. 例如对于执行经历 H'' : $o_{t_1}^r[d_x] o_{t_2}^r[d_y] o_{t_2}^r[d_x] o_{t_2}^w[d_x] o_{t_2}^c[d_x] o_{t_3}^c[d_y] o_{t_3}^w[d_y] o_{t_3}^c[d_y] o_{t_1}^c[d_y] o_{t_1}^c[d_y] o_{t_1}^c[d_y]$ (图 2). 也就是说, 冲突可串行化经历都是亚可串行化的, 但是存在不是冲突可串行化的亚可串行化执行经历.

证毕.

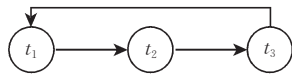


图 2 $S_{H''}$

3 并行亚可串行化并发控制协议 BCC-HSR

3.1 服务器端的处理

令 $p(t_u)$ 表示事务 t_u 提交的时刻, 服务器循环不断执行如下算法.

ServerProcessing()

{ Broadcast a data item d ;

Validate the mobile update transactions sent by mobile clients to the update transactions committed during the transmission;

While an update transaction t_u is committed

{ $Spaw(P_{11}, P_{12}, \dots, P_{nn})$

For all $P_{ij} (1 \leq i \leq n; 1 \leq j \leq n)$

{ $N(i, j) = M(i, j)$;

If $d_j \in D_r(t)$ then $S(i) = 1$

For $u = 0$ to $(\lceil \log_2 n \rceil - 1)$ do

{ If $(INT(\frac{i-1}{2^{u+1}}) \wedge (i+2^u) \leq n)$ then

{ If $(S(i) \neq 1 \wedge S(i+2^u) = 1)$ then

{ $S(i) = 1; N(i, j) = N(i+2^u, j)$ }

If $(S(i) = 1 \wedge S(i+2^u) = 1)$ then

{ If $(N(i, j) < N(i+2^u, j))$ then $(N(i, j) = N(i+2^u, j))$ }

}

```

    }
    If ( $d_i \in D_w(t) \wedge d_j \in D_w(t)$ )
    then  $M(i, j) = p(t_u)$ 
    else if ( $d_i \notin D_w(t) \wedge d_j \in D_w(t)$ )
    then  $M(i, j) = N(i, 1)$ 
    }
}

```

3.2 移动客户机上的处理

移动客户机上的活跃事务 t_a 执行如下算法直到完成它的所有操作:

```

MobileProcessing( $t_a$ )
{
  REPEAT
  {
    If ((a data item  $d_j$  is broadcasted)  $\wedge$  ( $d_j \in D(t_a)$ ))
    Then {If ( $d_i \in D^r(t_a)M(j, i) \geq M(t)$ ) then abort
           $t_a$ ;
        Else  $t_a$  operate  $d_j$ ;
      }
  }
  Until (EOT( $t_a$ ))
  If  $t_a$  is a read-only transaction then commit  $t_a$ ;
  If  $t_a$  is an update transaction then send  $t_a$  to the
  server for further validation;
  While EOT( $t_a$ ) denotes the end of  $t_a$ 
}

```

3.3 协议的正确性

定理 6. BCC-HSR 协议允许一个只读事务 t 提交的充分必要条件是該只读事务的相关读图中无环。

证明. 先证充分性, 用反证法. 假设只读事务 t 夭折, 则根据 BCC-HSR 协议: t 必定在某时刻 p_1 中读了某提交于时刻 p_2 的事务 t' ($t' \in A_H(t) - \{t\}$ 且 $p_2 > p_1$) 所写了的数据对象 d_j ; 并且 t 必定在时刻 p_3 ($p_3 > p_2$) 中又欲读数据对象 d' , 而最后一个写 d' 的事务 (t'') 在 p_3 开始前已经直接或间接地读取了 t , 此时相关读图中必有环 $t \rightarrow t' \rightarrow \dots \rightarrow t'' \rightarrow t$, 这与 t 的相关读图中无环相矛盾。

再证必要性, 用反证法. 设 t 的相关读图中有环. 出现这种情况的唯一可能性在于某提交于时刻 p_2 的事务 t' ($t' \in A_H(t) - \{t\}$) 在 t 读数据对象 d_i (在时刻 p_1 中) 后对 d_i 执行了写操作. 然而, 因为 $t' \in A_H(t) - \{t\}$, 必有某数据对象 d_j 以及事务 t'' ($t'' \in A_H(t) - \{t\}$, $t' \in A_H(t'')$), 且 t 读 (在时刻 p_3 中) 了 t'' 所写数据对象 d_j 的值. 因此, 在 p_3 时 $p_1 \leq p_2 \leq M(i, j) < p_3$, 根据 BCC-HSR 协议, 此时 t 由于 $M(i, j) \geq p_1$ 而夭折. 出现矛盾. 证毕。

4 试验模拟

现有的广播环境并发控制协议中 BCC-SR 协议

的性能最优. 我们通过模拟实验, 将新协议 BCC-HSR 与 BCC-SR 协议进行了比较, 主要以移动事务平均响应时间来考察. 实验中采纳目前数据广播研究中普遍使用的模型及参数设置 (见表 1, 其中时间单位为位时, 即传输一位所需时间)^[6, 11-12, 16]. 模型由一个服务器、一个客户机以及传递数据对象和控制信息的广播磁盘组成. 事务访问的数据对象均匀分布于数据库中. 数据库设置得较小以加强数据冲突来模拟热点效果. 为了比较两种方法的性能, 依次改变移动事务长度、服务器端事务长度、服务器端事务到达率、处理器数目等, 并分别统计平均响应时间. 为了提高实验数据的可靠性, 每次连续运行 5000s 以保证系统运行于稳定状态, 并且仅从第 1000s 开始统计实验结果。

图 3~6 分别给出了当移动事务长度、服务器端事务长度、服务器端事务到达率、处理器数量变化时 BCC-HSR 协议和 BCC-SR 协议的移动事务平均响应时间. 图 3 比较了当移动事务长度变化时两种方法的性能: 随着移动事务长度的增加, 每个事务中的操作数增加, 事务响应时间就会增加. 从图中可以看到, 两种方法的平均响应时间都随着移动事务长度的增加而增加, 当移动事务长度增加到 14 时两种方法开始表现出性能上的明显差异, 随着移动事务长度的增加, BCC-SR 方法的平均响应时间增加得非常快而 BCC-HSR 比 BCC-SR 增加慢得多. 图 4 给出了当服务器上的事务长度变化时的情况: 随着服务器上更新事务长度的增加, 每个时刻内服务器上的更新操作就增加, 发生冲突的概率增大, 从而响应时间也就增加, 然而可以从图中看到 BCC-HSR 比 BCC-SR 增加慢得多. 图 5 给出了当服务器上事务到达率变化时的情况: 服务器端事务到达率的增加对 BCC-HSR 方法的响应时间没有多大影响, 而 BCC-SR 则对事务到达率非常敏感. 图 6 给出的是当处理器数量变化时的情况. 当处理器数目增大时, 服务器处理端事务处理能力增强, BCC-SR 和 BCC-HSR 两种方法中响应时间都呈下降趋势, 由于 BCC-HSR 协议中服务器端的处理充分挖掘了算法的可并行性, 充分发挥服务器的并行处理能力, 其中事务的响应时间随处理器数目的增加而显著减少. 这些实验结果都表明, 新协议可无损数据一致性而显著提高系统性能。

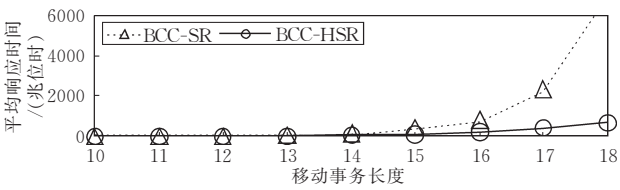


图 3 移动事务长度变化时的响应时间

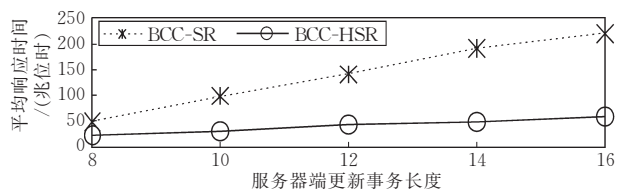


图 4 服务器端更新事务长度变化时的平均响应时间

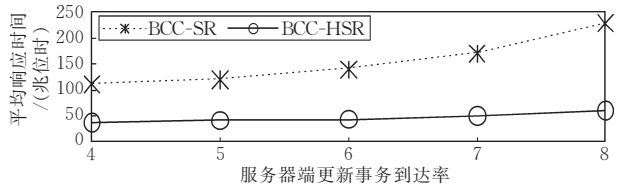


图 5 服务器端更新事务到达率变化时的响应时间

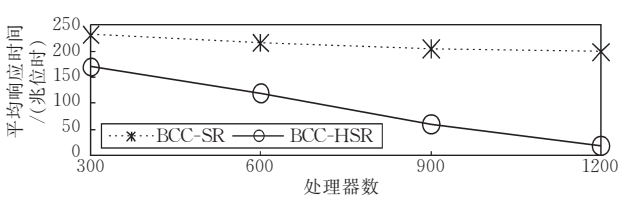


图 6 处理器数变化时的响应时间

表 1 模拟实验参数设置

参数项	移动端事务长度	服务器端事务长度	服务器端事务到达率/位时	数据库中数据对象数	处理器数	数据对象大小	服务器端读操作概率	移动端操作间延迟/千位时	移动端事务间延迟/千位时	移动端重启延迟/位时
默认设定值	6	10	$1/2.5 \times 10^5$	600	750	1KB	0.5	64	128	0

5 结 论

尽管移动计算和数据广播已有很多研究,数据广播环境中高性能并发控制机制的研究乃是一个新的重要问题.许多数据广播类数据库应用都需要维护数据一致性,本文为此提出了一种有效的并发控制方法,一系列的实验结果表明了新协议的有效性.

致 谢 感谢审稿人的中肯意见!

参 考 文 献

[1] Madria S K, Mohania M, Bhowmick S S, Bhargava B. Mobile data and transaction management. Information Sciences, 2002, 14(1): 279-309

[2] Barbara D. Mobile computing and databases—A survey. IEEE Transactions on Knowledge and Data Engineering, 1999, 11(1): 108-117

[3] Yucel S, Ozgur U. Exploiting data mining techniques for broadcasting data in mobile computing environments. IEEE Transactions on Knowledge and Data Engineering, 2002, 14(6): 1-13

[4] Barbara D. Mobile computing and databases—A survey. IEEE Transactions on Knowledge and Data Engineering, 1999, 11(1): 108-117

[5] Dang De-Peng, Liu Yun-Sheng. Concurrency control in real-time broadcast environments. The Journal of System and Software, 2003, 68(2): 137-144

[6] Lee V C S, Lam Kwok-Wa, Son S H, Chan E Y M. On transaction processing with partial validation and timestamp ordering in mobile broadcast environments. IEEE Transactions on Computers, 2002, 51(10): 1196-1211

[7] Lam K Y, Edward C, Au Mei-Wai. Concurrency control strategies for ordered data broadcast in mobile computing systems. Information Systems, 2004, 29(3): 207-234

[8] Lam K Y, Edward C, Au Mei-Wai. Broadcast of consistent data to read-only transactions from mobile clients//Proceedings of the 6th IEEE Workshop on Mobile Computing Systems and Applications. New Orleans, Louisiana, USA, 1999: 80-88

[9] Lee J S K, Hwang Chong-Sun, Yu Heong Chang. Supporting transactional cache consistency in mobile database systems//Proceedings of the ACM International Workshop on Data Engineering for Wireless and Mobile Systems. Washington, USA, 1999: 6-13

[10] Pitoura E. Supporting read-only transactions in wireless broadcasting//Proceedings of the 9th International Workshop on Database and Expert Systems Applications DEXA' 98, 1998: 111-118

[11] Kim S S, Lee S K, Jung S Y, Hwang C S. O-PreH: Optimistic transaction processing algorithm in hybrid broadcast environments//Proceedings of the 10th International Conference on Information and Knowledge Management. Atlanta, Georgia, USA, 2001: 553-555

[12] Dang De-Peng, Zhou Li-Zhu. A serializable concurrency control protocol in wireless broadcast environments. Journal of Computer Research and Development, 2006, 43(7): 1280-1284(in Chinese)

(党德鹏,周立柱. 移动广播环境中的可串行化并发控制协议. 计算机研究与发展, 2006, 43(7): 1280-1284)

[13] Ramamritham K, Calton P. A formal characterization of epsilon serializability. *IEEE Transactions on Knowledge and Data Engineering*, 1995, 7(6): 997-1007

[14] Lam Kam Yiu, Yau Wai Cheng. On using similarity for concurrency control in real-time database systems. *The Journal of Systems and Software*, 1998, 43(3): 223-232

[15] Liu Yun-Sheng, Li Guo-Hui. The quasi-consistent serializable concurrency control for real-time database. *Chinese Journal of Computers*, 1999, 22(4): 420-423(in Chinese)
(刘云生, 李国徽. 实时数据库的准一致性可串行化并发控制. *计算机学报*, 1999, 22(4): 420-423)

[16] Chen Guan-Chi, Lee Sun-Yin. An analytic model for performance analysis of concurrency control strategies in mobile environments. *The Computer Journal*, 1999, 42(6): 184-196



DANG De-Peng, born in 1970, Ph.D., associate professor. His current research interests include advanced database systems, distributed and mobile real-time systems, and information integration.

Background

Research on the problem of data consistency in mobile broadcast systems is very important. Most of recent studies that have focused on transaction processing in the asymmetric broadcast environments only support read only transactions at the mobile clients. Some methods that support update transactions have been provided. However, they are based on serializability that is too strict to support many mobile applications. Previous relaxing serializability such as epsilon and similarity serializability would sacrifice database consistency to some extent. This paper focuses on consistent and relaxing concurrency control for parallel broadcast systems,

and the proposed protocol BCC-HSR supports both read only transactions and update transactions, always maintains data consistency and can improve average response time of mobile transactions significantly.

This research is sponsored by National Science and Technology Support Plan Foundation of China under contract number 2006BAK01A07, Chinese Post-doctor Science Foundation under contract number 2003034146, National Natural Science Foundation of China under contract number 60573094.