МОДЕЛИРОВАНИЕ РАСПРЕДЕЛЕННОЙ ДВУХВЕРСИОННОЙ ДВУХФАЗНОЙ БЛОКИРОВКИ

С. Василева (Добрич, Болгария), Ю. М. Носков (Москва)

Управление паралельными транзакциями в системах баз данных (БД) исследуется в течение последнего времени. Опубликованы результаты, полученные с помощью аналитических и имитационных моделей, построенных для централизованных систем управления БД. Однако задача сравнения различных методов управления паралелизмом транзакций в распределенных базах данных (РБД) [7] исследована недостаточно. Почти отсутствуют результаты имитационного моделирования методов блокирования (двухфазное блокирование – *two-phase locking* (*2PL*)) в РБД: централизированное *2PL*, *2PL* первичных копий, распределенное *2PL* и *2PL* большинства копий. В работах [1] и [3] рассматриваются модели РБД с блокированием. Результаты моделирования многоверсионных алгоритмов двухфазного блокирования до сих пор, по-видимому, никем не публиковались. В [8] представлены результаты сравнительного анализа алгоритмов *Ordering Network* и распределенной *2PL* при условии полной репликации данных по узлам распределенной СУБД (РСУБД). А в [2] рассмотрены и проанализированы моноверсионный и многоверсионный алгоритм, но для протокола временных меток.

Моделирование распределенных транзакций в РСУБД GPSS транзактами детально описано в [9]. Однако там исследуется выполнение простых транзакций (обрабатывающих только один элемент данных), где невозможно возникновение взаимоблокировок. В данной работе рассматривается исполнение транзакций длиной 1 и 2 элемента данных. Вероятность возникновения коротких или длинных транзакций описывается функцией *BrEl*.

В современных РСУБД управление конкурентного выполнения транзакций основывается на *2PL* протоколах. Данные в таких информационных системах не полностью (частично) распределены по узлам системы. Поэтому актуальными являются разработка и моделирование многоверсионных (двухверсионных) алгоритмов управления транзакциями с блокированием.

Разработка аналитической модели для таких систем практически невозможна по следующим причинам:

- неполная репликация данных по узлам распределенной системы;
- наличие двухуровневой архитектуры данных.

Поэтому авторами выбран метод имитационного моделирования для исследования производительности РСУБД, работающих по методам *2PL*. Выбрана среда GPSS World для разработки имитационных алгоритмов и проведения экспериментов с моделью.

Рассмотрим алгоритм, моделирующий исполнение глобальных транзакций по двухверсионному протоколу двухфазной блокировки, который работает следующим образом.

Рассматриваются две копии каждого элемента данных. Протокол распределенной *2PL* детально рассмотрен в [1] и [3], а принципы двухверсионной блокировки – в [4] и [5]. На рис. 1 продемонстрировано выполнение глобальной транзакции, инициированной узлом S_{P2} и обновляющей элементы данных *El1* (имеющего копии в локальных базах данных LDB_{P6} и LDB_{P7}) и *El2* (с копиями в локальных базах данных LDB_{P8} и LDB_{P9}), соответственно на узлах S_{P6} и S_{P7}, и S_{P8} и S_{P9}.

В модели генерируются 6 потоков транзактов, каждый из которых отображает глобальную транзакцию в РСУБД. Каждый поток характеризуется интенсивностью λ (число транзакций в миллисекунду).

Рассмотрим последовательность операций в модели, схема которой представлена на рис. 1.

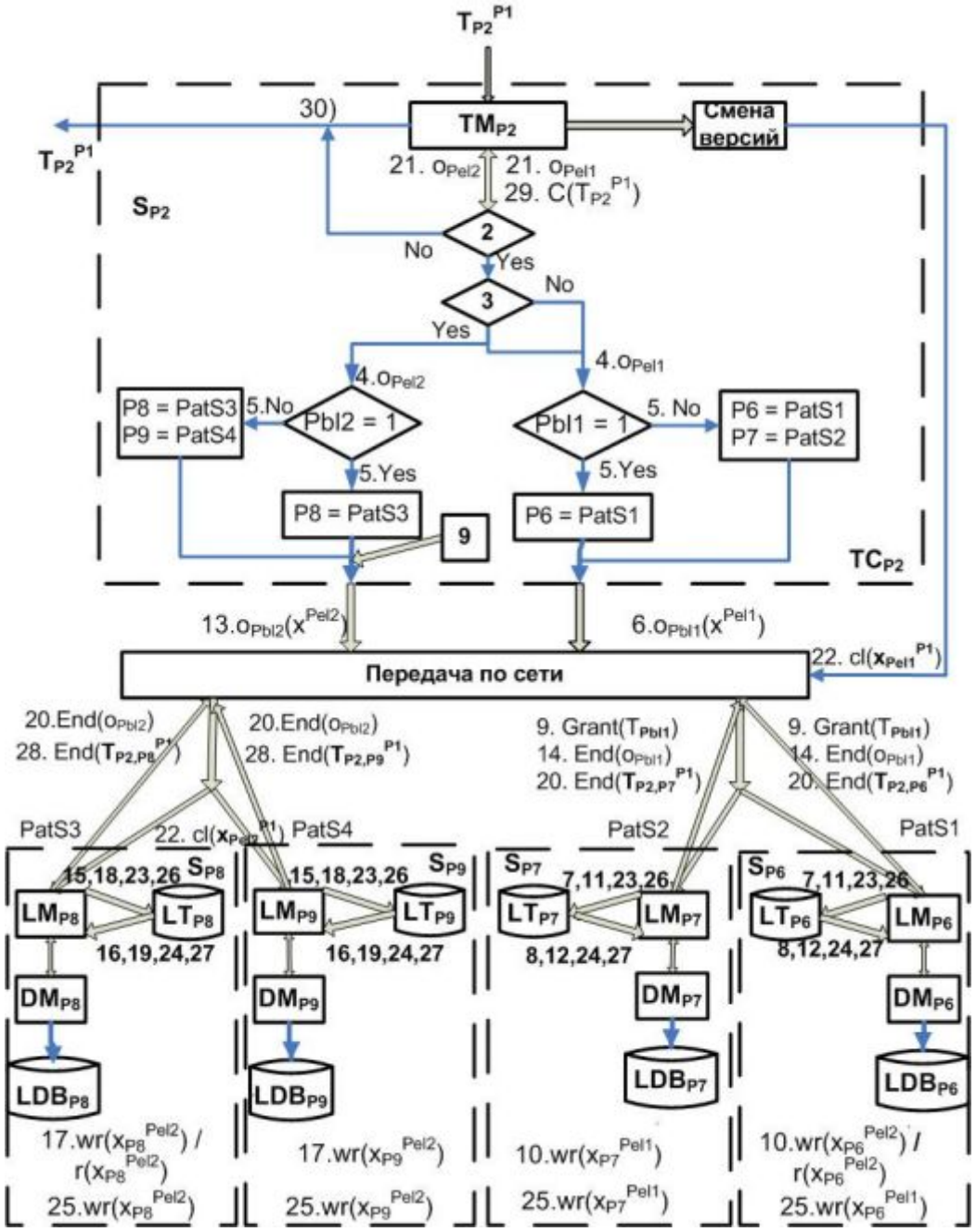

Рис. 1. Схема выполнения распределенной 2V2PL глобальной транзакции (при обновлении двух элементов данных и подтверждении версий)

При поступлении транзакции T_{P2}^{P1} в менеджере транзакции TM_{P2} проверяется (операция 2) возможность ее исполнения (уровень доступа и другие возможные условия), если невозможно выполнение T_{P2}^{P1}, она откатывается (выполняется операция 30 – $End(T_{P2}^{P1})$). Если T_{P2}^{P1} продолжает свое выполнение, проверяется ее длина (1 или 2 элемента данных будут обрабатываться) – операция 3 и транзакт подготавливается к расщеплению – операции 4. Операциями 5 присваиваются значения параметрам подтранзактов – номера диспечеров блокировок LM_{P6}, (LM_{P7}), (LM_{P8} и LM_{P9}), где подтран-

закты $T_{P2,P6}^{P1}$, ($T_{P2,P7}^{P1}$), ($T_{P2,P8}^{P1}$ и $T_{P2,P9}^{P1}$) должны выполнить операции чтения/записи копий элементов данных *El1* и *El2*. В общем случае выполняется передача заявок на блокирование копий элементов по сети к узлам исполнителям (операции 6 и 13).

В узлах-исполнителях S_{P6}, (S_{P7}) диспечеры блокировок LM_{P6}, (LM_{P7}) проверяют в таблицах блокировок LT_{P6}, (LT_{P7}) операциями 7 возможность предоставления блокировки копий элемента *El1* подтранзактам $T_{P2,P6}^{P1}$, ($T_{P2,P7}^{P1}$). Решение предоставления блокировки элемента принимается диспечерами *LM* в соответствии с таблицей совместимости «версионных» блокировок, приведенной в [4]. Если блокировка разрешена (операции-сообщения 8), подтранзакты расщепляются и их наследники (операциями 9) возвращаются в узел-инициатор для передачи подтверждения блокировки *El1* перед подтранзактам $T_{P2,P8}^{P1}$ и $T_{P2,P9}^{P1}$, чтобы глобальная транзакция продолжила свою первую «расширяющую» фазу, а родители $T_{P2,P6}^{P1}$, ($T_{P2,P7}^{P1}$) продолжают воспроизводить процесс выполнения операций чтение/запись (операции 10) копий элемента *El1* в локальных БД LBD_{P6}, (LBD_{P7}). В большинстве случаев операция записи копий элемента *El1* не требует создания новой версии элементов данных. Поэтому операция 11 с большей вероятностью является сообщением о снятии блокировки элемента, а не запросом о фиксирующей блокировке копии *El1* для корректной смены версий элемента. Соответственно операция 12 будет представлять подтверждение снятий блокировки (т.е. конец операции над элементом *El1*).

При получении подтверждения блокировки элемента *El1* (операция 9) менеджер транзакции $T_{P2}^{P1} - TM_{P2}$ передает подтранзакцию, обрабатывающую элемент *El2*, координатору транзакции TC_{P2}. С большой вероятностью (порядка 0,8) транзакция обновляет *El2*, поэтому транзакция расщепляется (операция 13) и подтранзакты $T_{P2,P8}^{P1}$ и $T_{P2,P9}^{P1}$ передаются через каналы сети к соответствующим диспетчерам блокировок LM_{P8} и LM_{P9} (операция 14 рис. 1). LM_{P8} и LM_{P9} через таблицы блокировок проверяют возможность занятия копий *El2* (операции 15). В случае блокировки (операция 16) подтранзакты продолжают двигаться к диспетчерам данных DM_{P8} и DM_{P9} для выполнения операций чтение/запись (операции 17). Если блокировка невозможна, подтранзакты встают в очередь перед копиями *El2* в таблицах блокировок LT_{P8} и LT_{P9}, выжидание моделируется списками пользователя с номером *P11*. Параметру *P11* каждого из подтранзактов $T_{P2,P6}^{P1}$, ($T_{P2,P7}^{P1}$), $T_{P2,P8}^{P1}$ и $T_{P2,P9}^{P1}$ присваивается значение перед входом в соответствующем диспетчере блокировок.

Если подтранзакция прочитала элемент или не создавала новую версию (в таблице блокировок элемент не был занят другим обновляющим транзактом блокировкой на запись, то операции 18 являются операциями освобождения блокировки, и соответственно операции 19 означают конец работы подтранзактов в узлах-исполнителях. Подтверждение конца чтения/записи элемента *El2* передается в менеджеру транзакции TM_{P2} (операция 20). Подтранзакты $T_{P2,P8}^{P1}$ и $T_{P2,P9}^{P1}$, а до этого $T_{P2,P6}^{P1}$ и $T_{P2,P7}^{P1}$ (если элемент *El1* был обновлен) объединяются в подтранзактах, обрабатывающих соответственно *El2* и *El1*.

Если необходима смена версий элемента *El1* и/или *El2*, высылаются запросы на фиксирующую блокировку копий соответствующего элемента к узлам-исполнителям (операция 22). Соответствующие диспетчеры *LM* ставят запись о фиксирующей блокировке в таблицах блокировок (операции 23). После получения блокировки (операции 24) текущая версия элемента становится завершенной (операция 25). Блокировка копии элемента освобождается (операции 26 – запрос и 27 – подтверждение снятия). Передается подтверждение конца работы соответствующих подтранзактов к менеджеру транзакции TM_{P2}. Подтранзакты работы над *El1* обьединяются, соответственно подтранзакты над *El2*, и транзакт-родитель, выжидающий их в TM_{P2} (операция 29). После сбора статистики об окончивших работу транзактах, они покидают систему (операция 30).

В моделирующем алгоритме используются следующие параметры транзактов моделирующих транзакций РСУБД:

P1 – номер генерируемого транзакта;

P2 – номер узла, генерирующего транзакт;

Pel1 – номер первого обрабатываемого транзактом элемента данных (*El1*);

Pbl1 – тип заявляемой блокировки элемента *El1*: 1 (*rl*) – *if* чтение(*El1*); 2 (*wl*) – *if* запись(*El1*); 4 – *certify lock*(*El1*), *if* вторая (*uncommitted*) версия *El1* создана транзактом;

Pel2 – номер второго обрабатываемого транзактом элемента данных (*El2*) ;

Pbl2 – тип заявляемой блокировки элемента *El2*: 1 (*rl*) – *if* чтение(*El2*); 2 (*wl*) – *if* запись(*El2*); 4 – *certify lock*(*El2*), *if* вторая версия *El2* создана транзактом;

P5 – фаза обработки транзакции: принимает значение 0 при поступлении транзакта в модель и после конца операций чтение/запись присваивается значение 1. Если какая-нибудь из подтранзакций прочитала или создала незавершенную версию элемента данных, *P5* = 2; после смены версий элемента данных принимает значение 3;

P6 – номер узла, где хранится первая копия первого элемента данных *El1*;

P7 – номер узла, где хранится вторая копия первого элемента данных *El1*;

P8 – номер узла, где хранится первая копия второго элемента данных *El2*;

P9 – номер узла, где хранится вторая копия второго элемента данных *El2*;

P11 – номер списка пользователя, где соответствующий подтранзакт ожидает освобождения копии элемента данных.

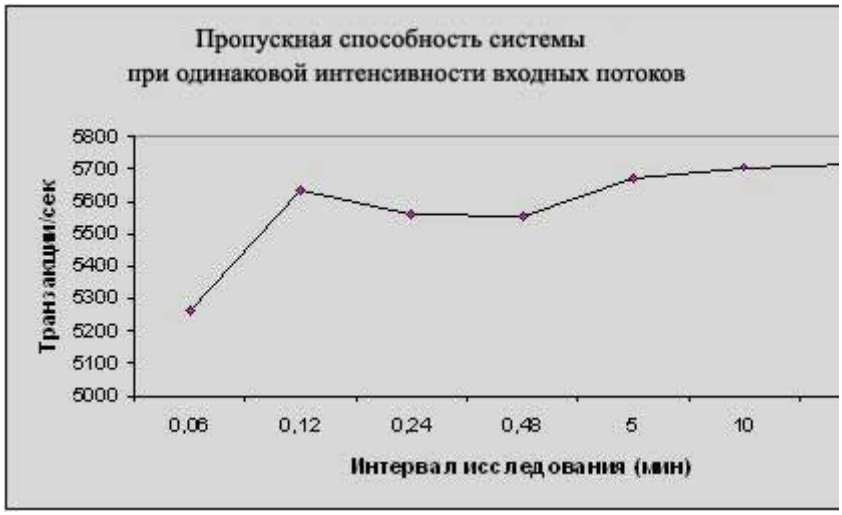

Рис. 2. Пропускная способность системы

Выводы

1. Разработка и моделирование многоверсионных (двухверсионных) алгоритмов управления транзакциями с блокированием является актуальной задачей.

2. Для исследования процесса обработки распределенных транзакций в системах РБД эффективно использование системы имитационного моделирования.

3. Предлагаемая в работе модель системы достаточно точно описывает реальные процессы и позволяет получить достоверную оценку изменения пропускной способности системы при заданных параметрах входных потоков транзакций. Это подтверждается результатами исследований для РБД [2] и результатами реальных исследований процессов в СУБД [8, с. 75] (рис. 2).

Литература

1. **Гарсиа-Молина Г., Ульман Д., Уидом Д**. Системы баз данных. М.: Вильямс, 2003.
2. **Гасанова Н.** Разработка алгоритмов управления транзакциями, основанных на методе временных меток в распределенных базах данных, 2004, http://science.az/autoreferats/ referat_hasanova_nazli.pdf.
3. **Конноли Т., Бегг К.** Базы данных. М.–СПб–Киев: Вильямс, 2003.
4. **Кузнецов С.** Базы данных. Вводный курс. http://www.citforum.ru/database/advanced_intro/43.shtml, 2008.
5. **Чардин П.** Многоверсионность данных и управление паралельными транзакциями. http://www.citforum.ru/database/articles/multiversion.
6. http://www.realcoding.net/article/view/1581.
7. **Srinivasa R., Williams C., Reynolds P.** Distributed Transaction Processing on an Ordering Network, Technical Report CS-2001-08, February 2001, http://citeseer.ist.psu.edu/cache/papers /cs/26119/srinivasa01distributed.pdf.
8. TPC Benchmark^{TM}C, 2006.
9. **Vasilev, S., Milev A.** Simulation Models of Two-Phase Locking of Distributed transactions// International Conference on Computer Systems and Technologies – CompSysTech 2008, Gabrovo, Bulgaria. P. V. 12-1–V. 12-6.