УДК 683.519

ГУСЕВ Е.И.

МОДЕЛИРОВАНИЕ СПОСОБОВ ОРГАНИЗАЦИИ ДОСТУПА К РАСПРЕДЕЛЁННЫМ СТРАНИЦАМ ПАМЯТИ В СИСТЕМАХ ОБЛАЧНЫХ ВЫЧИСЛЕНИЙ ОСНОВАННЫХ НА SHARED EVERYTHING АРХИТЕКТУРЕ

В работе предложена модель для исследования продолжительности конфликтов за распределённые ресурсы и страницы, применяемых в Oracle Real Application Clusters (Oracle RAC). Целью исследования является определение области модернизации существующих способов доступа и нахождение механизма оценки эффективности существующих и предлагаемых способов.

The subject of the article is research model of the of distributed resources and pages conflicts duration which used in the Oracle Real Application Clusters (Oracle RAC). The aim of the paper is to determine the area of modernization of the existing access methods and finding the way for effectiveness evaluation of existing and proposed assess methods.

Ключові слова: Oracle RAC, распределённая страница, организация доступа, общий ресурс;

1.Введение

Использование традиционных способов блокирования в системах облачных вычислений приводит к значительному увеличению времени обработки запроса и как следствие времени блокирования. Такая ситуация приводит к невозможности масштабирования облачной СУРБД при использовании классических способов блокирования общего ресурса без существенного увеличения скорости межузлового обмена [1]. Реализованная в Oracle RAC shared everything архитектура использует наиболее перспективный на сегодня способ обработки общих страниц памяти в облачных СУРДБ при обслуживании OLTP трафика. Для выявления проблем подхода shared everything предлагается использование модели, учитывающую специфику обработки страницы. Для моделирования shared everything архитектуры предлагается использовать 3 компонентную модель, состоящую из модели задержек [2], вероятностной модели и модели конфликтов. Структурно модель задержек реализует особенности аппаратной реализации, вероятностная модель (модель трафиков) – особенности функционирования, т.е. вероятностей прохождения тех или иных сценариев обработки в зависимости от трафика. А модель конфликтов описывает структуру конфликтов и позволяет оценить продолжительности конфликтов в зависимости от реализации и функционирования системы. Исходя из того, что система должна функционировать с различными трафиками и того, что возможности изменения аппаратной реализации системы ограничены – наиболее перспективным, с точки зрения модернизации, является изменение процедуры доступа к странице, которая полностью находится в фокусе модели конфликтов. Для решения этой задачи исследуются основные задержки, вносимые процедурой доступа исходя из модели конфликтов. Поскольку есть возможность снять временные задержки с реально функционирующего Oracle Real Application Clusters, целесообразно оценить адекватность предложенной модели, сравнив её с реально функционирующим Oracle RAC.

2.Модель конфликтов. Двухуровневая структура

Ключевой особенностью процедуры обработки общего ресурса в существующей реализации shared everything является наличие 2х конфликтов – это конфликт за ресурсы и конфликт за страницы содержащие ресурсы. Такой подход позволяет минимизировать количество пакетов пересылаемых при обмене ресурсами, но приводит к эскалации конфликта за ресурсы на уровень страниц. Во многом подход обусловлен наследием необлачных (традиционных) СУРДБ, где кэширование широко использовалось и выигрыш от него был очевиден. В контексте распределённой СУРДБ возникает конфликт за страницы, поскольку несколько узлов, а не один, как в традиционных системах, могут обрабатывать страницу. Кроме того актуальна задача обеспечения когерентности рас-

пределённых страниц. На рис.1 показана классическая процедура доступа к странице, используемая в Oracle RAC [3], и увеличение задержек (толстой сплошной линией) вследствие конфликта при обработке страницы на узле 4.

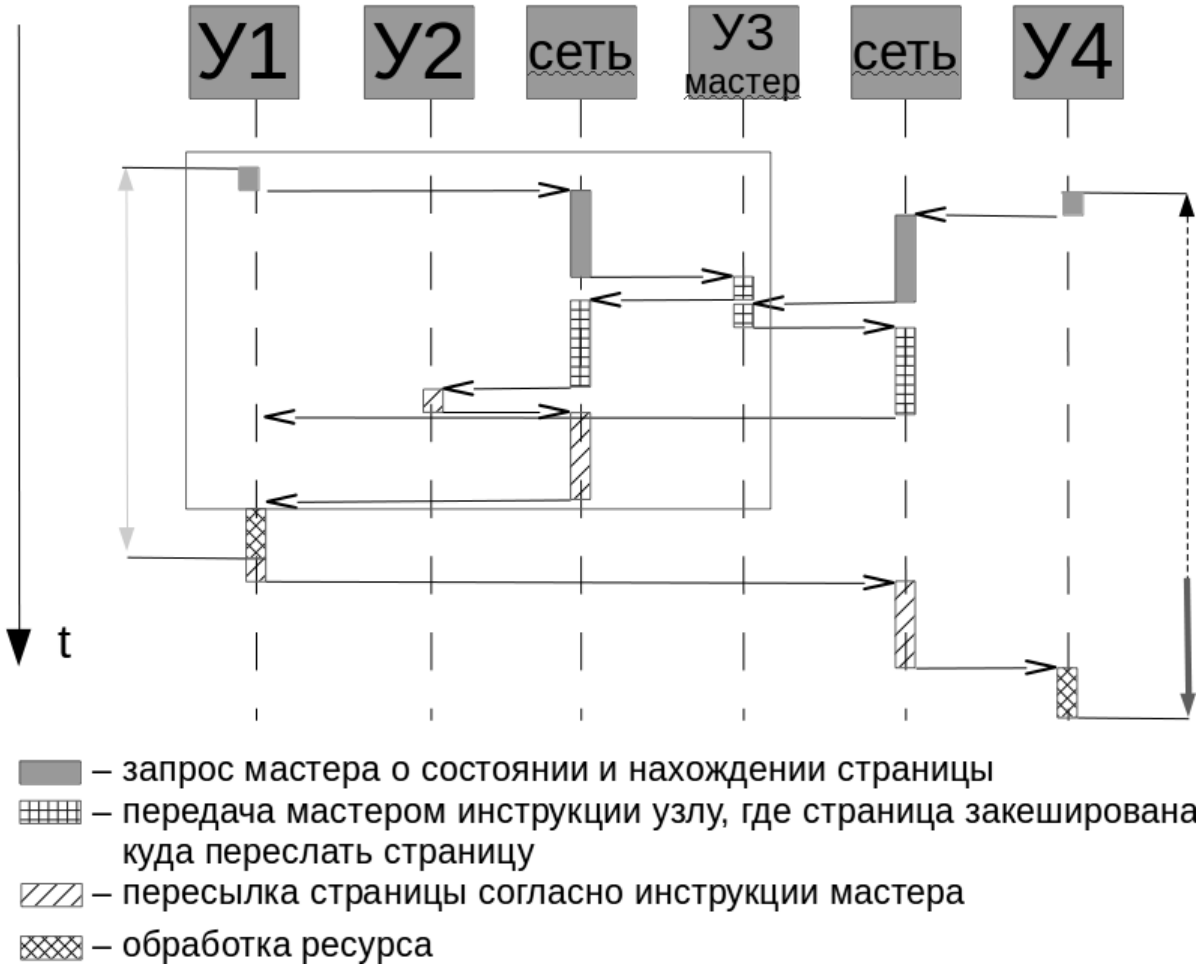

Рис.1 Процедура доступа к странице в Oracle RAC

Моделируя входящий поток доступа к страницам стационарным пуассоновским потоком, получаем отражённое в таблице 1 распределение ряда времён обработки (верхняя строка) и соответствующих им вероятностей (нижняя строка) в зависимости от количества конфликтов:

Таблица 1. Времена обработки и вероятности

$t_{send}/2$	$2t_{send}/2$	...	$nt_{send}/2$
$(\nu_{pl} t_{wpl}) e^{-\nu_{pl} t_{wpl}}$	$\frac{(\nu_{pl} t_{wpl})^2}{2!} e^{-\nu_{pl} t_{wpl}}$	...	$\frac{(\nu_{pl} t_{wpl})^n}{n!} e^{-\nu_{pl} t_{wpl}}$

Время t_{wpl} – это время в течение которого проводится наблюдение. t_{send} – время пересылки страницы между узлами. n – количество конфликтов за время наблюдения, ν_{pl} – интенсивность конфликтов за страницы для исследуемого ресурса (таблицы или индекса). Заметим, что для различных ресурсов интенсивности конфликта за страницы различные, что будет нами в дальнейшем учитываться через индексирование всех ресурсов входящих в транзакцию. Не сложно вычислить среднее время дополнительного ожидания как сумму ряда:

$$t_{p.queue} = \sum_{n=1}^{\infty} \frac{t_{send}}{2} \frac{(\nu_{pl} t_{wtl})^n}{n!} e^{-\nu_{pl} t_{wpl}} = \frac{\nu_{pl} t_{wtl} t_{send}}{2} \quad (1)$$

Нас в первую очередь интересует время в течение которого страницей ресурса может пользоваться одна заявка – т. е. время блокирования страницы. Это сумма t_{send} – времени пересылки и $t_{p.queue}$ – времени ожидания очереди к странице. Получаем уравнение и решаем его:

$$t_{wpl} = t_{send} + \frac{v_{pl}\, t_{wpl}\, t_{send}}{2} \quad => \quad t_{wpl} = \frac{2 t_{send}}{2 - v_{pl}\, t_{send}} \qquad (2)$$

Поскольку время доступа к странице (t_{acc}) состоит из двух времён пересылки служебного пакета (t_{net}), времени пересылки самой страницы (t_{send}) и времени ожидания очереди к странице ($t_{p.queue}$) [2], подставив в формулу доступа $t_{p.queue}$, получаем для традиционного shared everything способа доступа общее время доступа к странице:

$$t_{acc} = 2 t_{net} + t_{send} + \frac{v_{pl}\, t_{send}^2}{2 - v_{pl}\, t_{send}} = 2 t_{net} + \frac{2 t_{send}}{2 - v_{pl}\, t_{send}} \qquad (3)$$

Рассматривая модель конфликтов за ресурс необходимо отметить, что блокировка за ресурс удерживается до конца транзакции, в отличии от блокировки на страницу, удерживаемую до конца обработки страницы. Это значительно увеличивает время ожидания блокировки за ресурс и как следствие суммарные задержки вследствие конфликтов за ресурс. Расписав ожидаемую длительность транзакции получаем формулу:

$$t_{wtl} = n\, t_{acc} + \sum_{i=1}^{n} t_{(tl.i)} \frac{v_{l.i}}{v_i} + \sum_{i=1}^{n} v_{(l.i)} \left(t_{acc} + t_{tl.i} + t_{queue} \right) \qquad (4)$$

где t_{acc} – среднее время пересылок для получения страницы, $t_{tl.i}$ – среднее время ожидания очереди к i-тому ресурсу, t_{tl} - среднее время пересылок для получения ресурса, n – количество ресурсов в транзакции, v_{pl} – интенсивность обращений к i-му ресурсу, а t_{queue} - среднее время ожидания разрешения конфликта. Распишем время ожидания разрешения конфликта:

$$t_{queue} = t_{wtl} \sum_{i=1}^{n} v_i - \sum_{i=1}^{n} \left(\left(1 - e^{\left(-v_{(l.i)} t_{wtl} \right)} \right) \left(i \cdot t_{acc} + \sum_{j=1}^{i} t_{tl.j} \right) \right) \qquad (5)$$

3. Экспериментальная проверка модели сравнение с реально функционирующим Oracle RAC.

Для проверки адекватности предложенной модели реальному объекту была развёрнута система Oracle RAC состоящая из 4 узлов. Все узлы физически были развёрнуты на одном физическом сервере на платформе VMware, позволяющей эмулировать общий диск. Для обеспечения соотношения времени обработки ко времени пересылки (должно быть много меньше) будем использовать утилиту wondershaper, снизив скорость передачи между узлами до 600 KB/s командой "wondershaper vmnet1 600 600". Пятой виртуальной машиной, также развёрнутой на платформе VMware, будет генерироваться трафик с фиксированной интенсивностью в 32 потока. В процессе эксперимента мы изменяя интенсивность генератора измеряем общее время выполнения 1000000 транзакций, порциями по 10000, для уменьшения влияния времени пересылки между клиентом и сервером. Порции по 10000 сгруппированны в PL/SQL хранимых процедурах. В TPC-C трафике соотношение весов транзакций зафиксируем: New-Order – 41%, Payment – 44%, Order-Status – 5%, Delivery – 5%, Stock-Level – 5%.

На графике (Рис.2) показаны значения возвращаемые моделью и собранные в результате эксперимента: т.е. измеренное время обработки на Oracle RAC. Для удобства отображения приводятся нормированные величины: по оси абсцисс используется интенсивность умноженная на время одной пересылки, а по оси ординат время разделённое на время одной пересылки.

На основе формул 4 и 5, которые составляют систему уравнений, построен график (Рис. 2) зависимости длительности обработки транзакции от интенсивности. В эксперименте использовался элементарный трафик[4], трафик проводок[4] и синтетический тест TPC-C [5] используемый для оценки производительности СУРДБ. Соответствующие интенсивности конфликтов для каждого ресурса в каждом трафике вычисляются на основе модели трафиков (вероятностной модели) предложенной в [4].

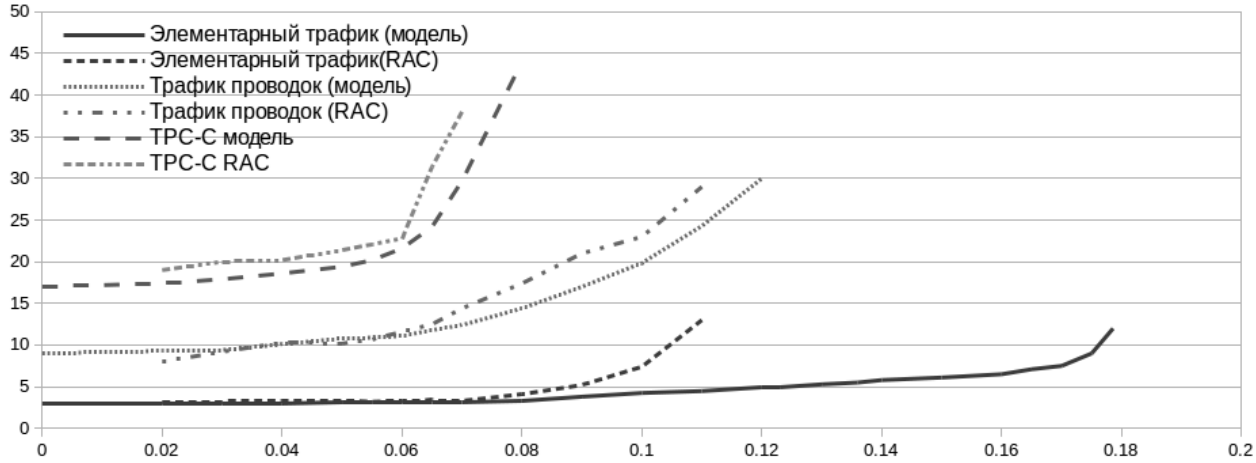

Рис.2 Моделируемое и фактическое время обработки транзакции

4. Оценка адекватности модели

Необходимо отметить, что вносимая погрешность носит не случайный, а систематический характер, поскольку при моделировании мы пренебрегали временем обработки данных по сравнению с временем пересылки, а также задержками при конфликтах за внутренние ресурсы. Под внутренними ресурсами в данной работе понимается очередь к центральному процессору, а также очереди к внутренним конструкциям Oracle, обеспечивающим целостность данных и корректность работы системы. Учитывая систематический, а не случайный характер вносимой погрешности, а также то, что прогнозирующая формула получена аналитическим путём, а не на основе обработки экспериментальной информации, критерии Фишера или Стьюдента, неприменимы. Для оценки адекватности предлагается оценить среднюю относительную погрешность. Однако, как видно из графика, переход к состоянию перегрузки (т.е. резкий рост времени обработки при увеличении интенсивности запросов) в реально-функционирующей системе происходит всегда раньше предсказанного значения. Это обуславливается в первую очередь увеличением времени обработки конфликтов за общие ресурсы (которое моделью игнорируется) при увеличении интенсивности.

В случае трафика проводок и TPC-C трафика реальная интенсивность перехода в перегрузку (коллапс) отличается от предсказанной моделью не более чем на 12%, а в случае элементарного трафика перегрузка (коллапс) возникает не от того, что время обработки транзакции превысило порог перехода в коллапс по ресурсам, а от того, что конфликт за горячие страницы (а в элементарном трафике на транзакцию приходится обработка 1 страницы, а с учётом данных UNDO – 2х страниц) возникает раньше. В первую очередь, такая ситуация является следствием эскалации конфликта за ресурсы на уровень страниц. Рассчитанная по формуле (3) при условии $t_{wpl}<2t_{send}$ интенсивность перехода в коллапс, с учётом нормирования как по оси у, так и по оси у составит 0.118, что также не превышает 12%. В таблице 2 приведены средние значения относительной погрешности в зависимости от выбранного диапазона значений нормированной интенсивности.

5. Выводы

Предложенная математическая модель позволяет с достаточной степенью достоверности определять временные задержки в реальной облачной системе использующей shared everything подход.

Таблица 2. Средняя погрешность в зависимости от трафика и диапазона

Трафик	Диапазон	Средняя относительная погрешность
элементарный	0-0.11	17.3%
	0-0.10	12.9%
	0-0.09	9.9%
Проводок	0-0.11	5.5%
	0-0.10	4.5%
	0-0.09	3.6%
TPC-C	0-0.07	11.9%
	0-0.065	10.4%
	0-0.06	8.3%

Модель позволила выявить такие недостатки организации доступа как эскалация конфликта за ресурсы на уровень страниц. На примере элементарного трафика продемонстрировано, что конфликт за страницы может существенно ограничивать возможности по пропускной способности cloud computing системы. Кроме того, на основе таких метрик как интенсивность перехода в коллапс (граничная интенсивность), среднее время доступа к странице и среднее время обработки транзакции, модель позволяет сравнивать различные способы доступа выявляя ограничения существующих и оценивая эффект от предлагаемых модернизаций. Модель может также применяться и в случае использования традиционных способов снижения вероятности конфликта за страницы (таких как партиционирование [6]), так и способов специфических для облачных систем [7]. Всё это позволяет сделать вывод о целесообразности применения модели при исследовании различных процедур доступа к распределённой странице.

Список литературы

1. Гусев Е.И. Исследование области применения неблокирующего алгоритма фиксации распределённых транзакций – 21 Вісник НТУУ "КПІ".Сер. Інформатика, управління та обчислювальна техніка. – 2012. Випуск 57. – С.76-80
2. Гусев Е.И. Математическое моделирование распределённой кластерной системы использующей shared everything подход (Oracle RAC). - Вісник НТУУ "КПІ".Сер. Інформатика, управління та обчислювальна техніка. - 2014. Випуск 60. - с.106 – 113
3. Markus Michalewicz, Burt Clouse, John McHugh Oracle Real Application Clusters (RAC). – Oracle White Paper . June 2013
4. Гусев Е.И. Моделирование трафиков и оценка скорости распределённого доступа в системах облачных вычислений с общим ресурсом на примере Oracle RAC -- Вісник НТУУ "КПІ".Сер. Інформатика, управління та обчислювальна техніка.- 2015. Випуск 62.
5. TPC BENCHMARK C Standard Specification Revision 5.11, February 2010, Transaction Processing Performance Council (TPC), www.tpc.org , © 2010 Transaction Processing Performance Council
6. Oracle Partitioning with Oracle Database 12c. Efficient Data Management and Performance Acceleration for every System -- ORACLE WHITE PAPER, SEPTEMBER 2014
7. Гусев Е.И. Оптимизация доступа к распределённым страницам памяти в cloud computing системах основанных на shared everything архитектуре используя метод разгрузки очередей. - Проблеми інформатизації та управління, Том 4, № 52 (2015)