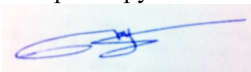
На правах рукописи



Вунна Джо Джо

«Оптимизация многопроцессорной обработки упорядоченных мультизапросов»

Специальность 05.13.11
Математическое и программное обеспечение вычислительных машин, комплексов и компьютерных сетей

АВТОРЕФЕРАТ

диссертации на соискание учёной степени
кандидата технических наук

Москва 2015

Работа выполнена на кафедре «Вычислительные машины, системы и сети» федерального государственного бюджетного образовательного учреждения высшего профессионального образования «Московский авиационный институт (национальный исследовательский университет)»

Научный руководитель: доктор технических наук, профессор
Брехов Олег Михайлович

Официальные оппоненты: доктор технических наук, профессор
Иванова Галина Сергеевна,
профессор МГТУ им. Баумана

кандидат технических наук, с.н.с.
Филиппов Владимир Александрович,
профессор МИЭМ НИУ ВШЭ

Ведущая организация: ОАО «Научно-исследовательский институт «Аргон»

Защита состоится « 27 » апреля 2015г. В 10:00 часов на заседании диссертационного совета Д212.125.01 при Московском авиационном институте (национальном исследовательском университете) по адресу: 125993, г. Москва, А-80, ГСП-3, Волоколамское шоссе, д.4.

С диссертацией можно ознакомиться в библиотеке Московского авиационного института (национального исследовательского университета).

Отзывы, заверенные печатью, просьба высылать по адресу: 125993, г. Москва, А-80, ГСП-3, Волоколамское шоссе, д.4, МАИ, Ученый совет МАИ

Автореферат разослан « »_...............2014г.

Ученый секретарь диссертационного совета Д212.125.01

кандидат технических наук, доцент

А.В.Корнеенкова

ОБШАЯ ХАРАКТЕРИСТИКА РАБОТЫ

Актуальность темы. Основным критерием при определении Плана реализации запроса является время выполнения мультизапроса, которое, вообще говоря, зависит от порядка выполнения элементарных запросов, его составляющих, и от времени проверки в строке и вероятности успеха в строке. Время выполнения элементарного мультизапроса зависит от метода доступа к столбцу таблицы. Известным методом увеличения производительности является использование многопроцессорных бортовых ВС.

В данной работе рассмотрена эффективность выполнения мультизапроса в базах данных многопроцессорной бортовой вычислительной системы, что является актуальным для авиационно-космических систем.

Цель работы. Оптимизация выполнения мультизапросов в базах данных с целью создания надстройки над СУБД, которая способна выполнять действия, необходимые для выбора наиболее эффективного плана выполнения мультизапроса, также осуществляя минимизацию накладных расходов.

Методы исследования. Аналитическое моделирование. Современная методология организации баз данных.

Научная новизна

- Предложен и обоснован метод оптимизации по времени выполнения мультизапроса при обращении к базе данных на основе упорядочивания элементарных запросов.
- Доказаны условия, при которых совместная обработка конъюнктивного мультизапроса обеспечивает не большее время выполнения по отношению к независимой обработке.
- Доказано, что параметр вероятность успеха при выполнении элементарного запроса является существенным параметром, влияющим как на выбор совместного и несовместного метода обработки мультизапроса, так и на определение числа процессоров.
- Разработан метод обеспечения оптимизации многопроцессорной обработки мультизапросов.
- Предложен оптимальный алгоритм распределения элементарных запросов на процессоры.

Достоверность полученных в диссертационной работе результатов подтверждается:

Применяемым математическим аппаратом. Соответствием полученных и известных результатов.

Практическая значимость. Разработан метод формирования и оценки времени выполнения плана выполнения мультизапроса с оптимизацией распределения элементарных запросов на процессоры.

Реализация результатов работы. Результаты диссертационной работы используются в учебном процессе кафедры «Вычислительные машины, системы и сети» Московского авиационного института (национального исследовательского университета) в форме информационного обеспечения блока дисциплин, а также в лекционном курсе «Моделирование».

На защиту выносятся следующие положения:

- Метод обеспечения оптимизации многопроцессорной обработки мультизапросов.
- Утверждение об условиях определения минимального времени обработки конъюнктивного мультизапроса для упорядоченных таблиц данных.
- Соотношения для минимального времени выполнения мультизапроса для упорядоченных или неупорядоченных данных таблиц при совместной обработке процессорами объединенного множества элементарных запросов.
- Оптимальный план распределения элементарных запросов на процессоры.

Апробация работы. Основные положения и результаты диссертационного исследования докладывались автором и обсуждались на научно-практической конференции студентов и молодых ученых МАИ «Инновации в авиации и космнавтике-2010», 13-ой Международной конференции МАИ «Авиация и космонавтика-2013»,12–15 ноября 2013 года.

Публикации. Основные материалы диссертационной работы опубликованы в 2 печатных работах из перечня ВАК.

Структура и объем диссертации. Диссертация состоит из введения, четырёх глав, заключения, библиографического списка из 50 наименований. Работа изложена на 145 страницах, содержит 27 таблиц и 28 рисунков.

СОДЕРЖАНИЕ РАБОТЫ

Во **введении** обоснована актуальность выполненного исследования сформулированы цель, задачи диссертационной работы, научная новизна, практическая ценность диссертационной работы по главам.

В главе 1 рассмотрены основные методы обработки мультизапрсов, сформулирована постановка решаемой в диссертации задачи обработки мультизапрсов на основе использования метода упорядоченной обработки элементарных запросов.

Основополагающей работой по оптимизации запросов является работа SELINGER. Идеи, изложенные в этой статье, легли в основу большинства исследований оптимизации.

Однако с растущей важностью оперативной обработки данных техника разработки и использования более сложных методов оптимизации запросов стала решающей. Для того чтобы быть эффективными, оптимизаторы должны адаптироваться к новым операторам, изменению в методах оценки стоимости и т.д. Эти требования привели к нынешнему поколению оптимизаторов запросов, из которых два оптимизатора Starburst и Volcano являются наиболее представительными. В то время как IBM DB2 оптимизатор основан на Startburst, Microsoft SQL-Server оптимизатор основан на системе Volcano. Основным различием между подходами является то, каким образом альтернативные планы создаются. Оптимизатор Starburst генерирует планы снизу вверх, тогда как оптимизатор Volcano создает планы сверху вниз.

План состоит из операторов (например, выбрать, присоединиться, сортировка). Стоимость оператора является функцией статистической сводки, которая включает в себя размер отношения, число различных значений атрибутов, распределение этих значений атрибутов в терминах гистограмм и т.д. Хотя точность этих статистических данных имеет решающее значение, оценка стоимости плана может быть чувствительна к этой статистике и поддержание этих статистических данных может требовать много времени. Проблеме эффективного поддержания достаточно точной статистики уделяется большое внимание в литературе, см. статью Poosala.

Кроме того, модель затрат должна учитывать влияние, например, буферизации отношений в кэше базы данных, модели доступа к памяти, доступность выполнения оператора, конвейерное выполнение и т. д.

Оптимизация мультизапроса

Предположим, что база данных D задана как ряд отношений $\{R_1, R_2$, …..,$R_m\}$, каждое отношение определено на ряде признаков. План доступа относительно запроса Q - последовательность основных операций, которые дают ответ на запрос Q. Например, имеется Служащий (ID, Имя, Род, Заплата, Город), с очевидными значениями для различных областей. Пусть задан следующий запрос SQL:

Select * From Employee1
Where Род=N'M' and Зарплата>=1000 and Город = N'Москва',
может быть обработан, посредством выполнения следующих задач

(T1) TEMP1 := Род=N'M'
(T2) TEMP2 := Зарплата>=1000
(T3) RESULT := Город = N'Москва'

Существует ряд возможных альтернатив плана обработки этого запроса.

У задач есть некоторая стоимость, связанная с ними, которая отражает стоимость центрального процессора, стоимость ввода / вывода, требуемых при их обработке. Стоимость плана доступа - стоимость обработки ее составляющих задач. Если имеется ряд запросов Q=$\{Q_1, Q_2, \ldots, Q_n\}$, то глобальный план доступа относительно Q соответствует плану, который обеспечивает способ вычислить результаты всех n запросов. Глобальный план доступа может быть создан, выбирая один план относительно каждого запроса и затем объединяя их вместе. Объединение в местном масштабе оптимальных планов не обязательно дает глобально оптимальный план. В связи с этим используются эвристические, динамические и другие алгоритмы.

В работах Selinger P.G., Astrahan M.M., Chamberlin D.D., Lorie R.A., Price T.G., Брехова О.М. показано, что минимальное время выполнения запроса может быть достигнуто при выполнении элементарных запросов в соответствующем порядке.

Имея это в виду, далее в данной работе исследуется возможность развития этого метода на случай выполнения мультизапроса в многопроцессорной базе данных.

Во **второй главе** разработан метод оптимизации однопроцессорной обработки мультизапросов.

Пусть множество запросов $З_i$, $i = \overline{1, n}$, образуют мультизапрос, при этом запросы $З_i$ сформированы конъюнкцией элементарных запросов, из которых ряд элементарных запросов повторно входят в запросы $З_i$. Назовем такой мультизапрос конъюнктивным мультизапросом.

Известным методом увеличения производительности баз данных ВС является одновременное выполнение нескольких запросов, образующих мультизапрос.

В данной работе рассмотрим эффективность выполнения мультизапроса в базах данных одно - и много - процессорной вычислительной системы.

Условие упорядоченнности. Минимальное время выполнения одного запроса в однопроцессорной ВС достигается для неупорядоченных столбцов таблицы при выполнении элементарных запросов в порядке

$$\frac{\tau_i}{1 - p_i} < \frac{\tau_{i+1}}{1 - p_{i+1}}, \qquad i = \overline{1, k-1},$$

и для упорядоченных столбцов таблицы в порядке

$$\frac{p_i\,\tau_i}{1 - p_i} < \frac{p_{i+1}\,\tau_{i+1}}{1 - p_{i+1}}, \qquad i = \overline{1, k-1},$$

где

n- число строк таблицы базы данных,

k - число столбцов таблицы,

τ_i- время обработки i -го элементарного запроса $ЭЗ_i$ для одной строки таблицы,

p_i - вероятность успеха при обработке i -го элементарного запроса $ЭЗ_i$ для одной строки таблицы (данные одной строки таблицы отвечают условию, заданным элементарным запросом $ЭЗ_i$).

Конъюнктивный мультизапрос может выполняться двумя способами, независимо друг от друга и совместно, когда выделяются подмножества совпадающих элементарных запросов, которые выполняются, в первую очередь, с целью уменьшения суммарного числа элементарных запросов.

Общий алгоритм формирования плана совместной обработки конъюнктивного мультизапроса

Алгоритм совместной обработки запросов конъюнктивного мультизапроса состоит в следующем: определяются подмножества совпадающих элементарных запросов, входящих в запросы конъюнктивного мультизапроса. Таких подмножеств может быть несколько, поэтому среди них выделяется одно по соответствующему критерию. Этому подмножеству соответствует элементарный запрос, который выполняется в первую очередь. Выделенный элементарный запрос удаляется из числа элементарных запросов мультизапроса.

Далее указанная процедура продолжается до тех пор, пока подмножество совпадающих элементарных запросов становится пустым.

В качестве критерия выделения подмножества (элементарного запроса) назовем следующие: номер элементарного запроса; число запросов, в которые входит элементарный запрос (глубина охвата); число элементарных запросов, образующих подмножество (ширина охвата).

Определение. Назовем запрос опорным запросом, если его первые (по условию упорядоченности) элементарные запросы образуют общее подмножество с другим запросом мультизапроса.

Теорема. Совместная обработка конъюнктивного мультизапроса обеспечивает не большее время выполнения по отношению к независимой обработке, при любых значениях τ_i и p_i, если алгоритм плана совместной обработки конъюнктивного мультизапроса отвечает условию, когда элементарные запросы опорного запроса выполняются первыми в порядке, отвечающем условиям упорядоченности.

Доказательство.

Пусть множество запросов ЭЗi, $i = \overline{1,k},$ образуют конъюнктивный мультизапрос, в котором выделим два запроса $З_1$ и $З_2$:

- $З_1 = ЭЗ_{i_1} \& ЭЗ_{i_2} \& \ldots \& ЭЗ_{i_r}$
- $З_2 = ЭЗ_{j_1} \& ЭЗ_{j_2} \& \ldots \& ЭЗ_{j_u} ЭЗ_{j_{u+1}} \& \ldots \& ЭЗ_{j_v}$

Пусть пересечение запросов $З_1$ и $З_2$ ($З_1 \cap З_2$) даёт подмножество с u элементарными запросами (в обозначениях запроса $З_2$ первыми u элементарными запросами: $ЭЗ_{j_1}, ЭЗ_{j_2}, \ldots, ЭЗ_{j_u}$), т.е. запрос $З_2$ является опорным запросом. По условию упорядоченности элементарные запросы опорного запроса $З_2$ выполняются первыми в порядке, отвечающем условию упорядоченности, т.е. время выполнения запроса $З_2$ остается минимальным. Так как запросы $З_1$ и $З_2$ выполняются совместно, то время выполнения запроса $З_1$ с учетом уже выполненнных элементарных запросов: $ЭЗ_{j_1}, ЭЗ_{j_2}, \ldots, ЭЗ_{j_u}$ ограничено выполнением подмножества элементарных запросов

$З_1 \backslash \left(ЭЗ_{j_1}, ЭЗ_{j_2}, \ldots, ЭЗ_{j_u}\right) = ЭЗ_{l_1}, ЭЗ_{l_2}, \ldots, ЭЗ_{l_{r-u}}$

и равно:

Для неупорядоченных данных

$T_{совм,1,н} = \mathrm{n}(p_{j_1}, p_{j_2}, \ldots, p_{j_u} (\tau_{l_1} + p_{l_1}\tau_{l_2} + p_{l_1}p_{l_2}\tau_{l_3} + \ldots + p_{l_1} \ldots p_{l_{r-u-1}}\tau_{l_{r-u}})),$

Для упорядоченных данных

$T_{совм,1,у} = \mathrm{n}(p_{j_1}, p_{j_2}, \ldots, p_{j_u} (p_{l_1}\tau_{l_1} + p_{l_1}p_{l_2}\tau_{l_2} + p_{l_1}p_{l_2}p_{l_3}\tau_{l_3} + \ldots + p_{l_1} \ldots p_{l_{r-u}}\tau_{l_{r-u}})),$

где параметры (τ_{l_1}, p_{l_1}), (τ_{l_2}, p_{l_2}),... $(\tau_{l_{r-u}}, p_{l_{r-u}})$ - параметры соответствующих элементарных запросов: $ЭЗ_{l_1}, ЭЗ_{l_2}, \ldots, ЭЗ_{l_{r-u}}$.

Очевидно, что время выполнения запроса $З_1$ и, следовательно, времени выполнения обоих запросов $З_1$ и $З_2$ при совместной обработке всегда не больше времени выполнения при независимой обработке.

Аналогичные рассуждения остаются и для любого числа запросов, образующих конъюнктивный мультизапрос, что и т. д.

Пример формирования плана совместной обработки конъюнктивного мультизапроса для мультизапроса.
Пусть мультизапрос состоит из четырех запросов

- $З_1 = ЭЗ_1 \,\&\, ЭЗ_2 \,\&\, ЭЗ_3 \,\&\, ЭЗ_4 \,\&\, ЭЗ_5 \,\&\, ЭЗ_6 \,\&\, ЭЗ_8 \,\&\, ЭЗ_9$
- $З_2 = ЭЗ_3 \,\&\, ЭЗ_4 \,\&\, ЭЗ_6 \,\&\, ЭЗ_7 \,\&\, ЭЗ_{10} \,\&\, ЭЗ_{11}$
- $З_3 = ЭЗ_4 \,\&\, ЭЗ_6 \,\&\, ЭЗ_7 \,\&\, ЭЗ_{12}$
- $З_4 = ЭЗ_3 \,\&\, ЭЗ_4 \,\&\, ЭЗ_6 \,\&\, ЭЗ_7 \,\&\, ЭЗ_{10}$

Сформируем план обработки этого мультизапроса на основе общего алгоритма формирования плана совместной обработки конъюнктивного мультизапроса.

Время выполнения конъюнктивного мультизапроса при независимой обработке равно сумме времен обработки запросов $З_1$, $З_2$, $З_3$ и $З_4$:

для неупорядоченных данных

$Т_{нез,н} = Т_{1,н} + Т_{2,н} + Т_{3,н} + Т_{4,н}$,

Возможный порядок обработки при совместной обработке элементарных запросов определяется алгоритмом совместной обработки запросов конъюнктивного мультизапроса, который для нашего примера дает два алгоритма, отвечающих определению опорного запроса и условию упорядоченности.

Алгоритм 1

В алгоритме 1 запрос 3 образует опорный запрос, элементарные запросы $ЭЗ_4, ЭЗ_6$ которого образуют общее подмножество и выполняются первыми.

Шаг 1: Выделяем элементарный запрос $ЭЗ_4$ на основе $\cap\, З_{i(i=\overline{1,4})} = ЭЗ_4$

Шаг 2: Выделяем элементарный запрос $ЭЗ_6$ на основе $\cap\, З_{i(i=\overline{1,4})} \backslash (ЭЗ_4\,) = ЭЗ_6$

Шаг 3: Выделяем элементарный запрос $ЭЗ_3$ на основе

$\cap\, З_{i(i=\overline{1,2,4})} \backslash (ЭЗ_4 \,\&\, ЭЗ_6) = ЭЗ_3$

Шаг 4: Выделяем элементарный запрос $ЭЗ_7$ на основе

$\cap З_{i(i=\overline{2,4})} \backslash (ЭЗ_3 \& ЭЗ_4 \ \& \ ЭЗ_6) = \ ЭЗ_7$

Шаг 5: Выделяем элементарный запрос $ЭЗ_{10}$ на основе

$\cap З_{i(i=\overline{2,4})} \backslash (ЭЗ_3 \ \& \ ЭЗ_4 \ \& \ ЭЗ_6 \& ЭЗ_7) = \quad ЭЗ_{10}$

Далее подмножество совпадающих запросов становится пустым.

Алгоритм 2

В алгоритме 2 запросы 2 и 4 образует опорный запрос, элементарные запросы $ЭЗ_3, ЭЗ_4, ЭЗ_6$ которого образуют общее подмножество и выполняются первыми.

Шаг 1: Выделяем элементарный запрос $ЭЗ_3$ на основе $\cap З_{i(i=\overline{1,2,4})} = ЭЗ_3$
Шаг 2: Выделяем элементарный запрос $ЭЗ_4$ на основе $\cap З_{i(i=\overline{1,2,4})} \backslash (ЭЗ_3\ ) = ЭЗ_4$
Шаг 3: Выделяем элементарный запрос $ЭЗ_6$ на основе
$\cap З_{i(i=\overline{2,4})} \backslash (ЭЗ_3 \ \& ЭЗ_4) = ЭЗ_6$
Шаг 4: Выделяем элементарный запрос $ЭЗ_7$ на основе
$\cap З_{i(i=\overline{2,4})} \backslash (ЭЗ_3 \ \& ЭЗ_4 \& ЭЗ_6) = ЭЗ_7$
Шаг 5: Выделяем элементарный запрос $ЭЗ_{10}$ на основе
$\cap З_{i(i=\overline{2,4})} \backslash (ЭЗ_3 \ \& ЭЗ_4 \& ЭЗ_6 \& ЭЗ_7) = ЭЗ_{10}$
Далее подмножество совпадающих запросов становится пустым.
Совместная обработка лучше независимой обработки по алгоритму 1 для неупорядоченных данных, если справедливо неравенство
$T_{нез,н} > T_{совм,з,н}$,
которое здесь имеет вид
$$\tau_1(1 - p_4p_6p_3) + (p_1 - p_4p_6p_3p_1)\tau_2 + (2 + p_1p_2 - p_4p_6\ )\tau_3 + (2p_3 + p_1p_2p_3)\tau_4 + (p_1p_2p_3p_4 - p_4p_6p_3p_1p_2)\tau_5 + (p_1p_2p_3p_4p_5 + 2p_3p_4)\tau_6 + \ p_3p_4p_6\tau_7 \ + p_3p_4p_6p_7\tau_{10} > \ 0,$$

Это неравенство выполняется при любых значениях параметров.

Оценка времени выполнения мультизапроса

Сформулируем ряд утверждений.

Пусть мультизапрос состоит из n запросов $З_i, i = \overline{1,n},$ элементарные запросы которых образуют упорядоченные множества $s_i, i = \overline{1,n},$ с последовательными (без пропусков) номерами элементарных запросов.

Пусть выполняются следующие условия:
Вложение множеств: $s_1 \supset s_2 \supset \cdots \supset s_i \supset \cdots \supset s_n$

Пересечение множеств: $s = s_n = \bigcap_{i=1}^{n} s_i$.

Утверждение 1. Выполнение элементарных запросов подмножеств $\mathrm{s_i}$ в порядке $\mathrm{i} = \overline{\mathrm{n}, 1}$, обеспечивает уменьшение времени совместного выполнения запросов мультизапроса.

Доказательство следует по индукции:

Обозначим параметры элементарного запроса: τ_i - время, требуемое для выполнения элементарного запроса ЭЗ$_i$, и p_i - вероятность успеха при его выполнении, $i = \overline{1, n}$.

Пусть запрос З$_{n\text{-}1}$ образован элементарными запросами с последовательными номерами:
$ЭЗ_{h+1}, ЭЗ_{h+2}, .., ЭЗ_{h+l}, ЭЗ_{h+l+1}, ЭЗ_{h+l+2}, .., ЭЗ_{h+l+m}, ЭЗ_{h+l+m+1}, ЭЗ_{h+l+m+2}, .., ЭЗ_{h+l+m+t}$
и запрос $З_n$ образован элементарными запросами, соответственно:
$ЭЗ_{h+l+1}, ЭЗ_{h+l+2}, \dots, ЭЗ_{h+l+m}$.

При несовместном выполнении этих двух запросов имеем следующее время:

для запроса $З_{n-1}$:

$$T_{З_{n-1}} = \tau_{h+1} + p_{h+1}\tau_{h+2} + p_{h+1}p_{h+2}\tau_{h+3} + \cdots + \left(\prod_{j=h+1}^{h+l-1} p_j\right)\tau_{h+l} + \left(\prod_{j=h+1}^{h+l} p_j\right)\left(\tau_{h+l+1} + p_{h+l+1}\tau_{h+l+2} + \cdots + \left(\prod_{j=h+l+1}^{h+l+m-1} p_j\right)\tau_{h+l+m}\right) + \left(\prod_{j=h+1}^{h+l+m} p_j\right)\left(\tau_{h+l+m+1} + p_{h+l+m+1}\tau_{h+l+m+2} + \cdots + \left(\prod_{j=h+l+m+1}^{h+l+m+f-1} p_j\right)\tau_{h+l+m+f}\right)$$

для запроса $З_n$:
$T_{З_n} = \tau_{h+l+1} + p_{h+l+1}\tau_{h+l+2} + \cdots + \left(\prod_{j=h+l+1}^{h+l+m-1} p_j\right)\tau_{h+l+m}$.

Суммарное несовместное время выполнения этих запросов:
$T_{\text{нс}} = T_{З_{n-1}} + T_{З_{n-1}}$.

При совместном выполнении этих запросов получаем:

$$T_{\text{совм}} = T_{З_n} + \left(\prod_{j=h+l+1}^{h+l+m} p_j\right)\left(\tau_{h+1} + p_{h+1}\tau_{h+2} + p_{h+1}p_{h+2}\tau_{h+3} + \cdots + \left(\prod_{j=h+1}^{h+l-1} p_j\right)\tau_{h+l}\right) + \left(\prod_{j=h+1}^{h+l+m} p_j\right)\left(\tau_{h+l+m+1} + p_{h+l+m+1}\tau_{h+l+m+2} + \cdots + \left(\prod_{j=h+l+m+1}^{h+l+m+f-1} p_j\right)\tau_{h+l+m+f}\right).$$

Очевидно, разность

$$T_{\text{несовм}} - T_{\text{совм}} = \left(1 - \left(\prod_{j=h+l+1}^{h+l+m} p_j\right)\right)\sum_{i=1}^{m}\left(\prod_{j=h+l+1}^{h+l+i-1} p_j \tau_{h+l+i}\right)$$

больше 0.

Последовательно проводя аналогичные рассуждения для пар запросов $(З_{n-i+1}, З_{n-i}), i = \overline{2, n-1}$, полностью завершаем доказательство.

Утверждение 2. Пусть пересечение множеств элементарных запросов $s_i, i = \overline{1, n}$, есть подмножество $s = s_n = \bigcap_{i=1}^{n} s_i$. Если подмножество s образовано элементарными запросами с последовательными номерами, начиная с первого элементарного запроса $ЭЗ_1$, то выполнение элементарных запросов подмножества s первыми обеспечивает уменьшение времени совместного выполнения относительно несовместного выполнения запросов мультизапроса на время (n-1) T_s, где T_s - время выполнения элементарных запросов подмножества s.

Доказательство следует из факта, что выполнение элементарных запросов подмножества s первыми не нарушает оптимальный по времени порядок обработки всех n запросов.

В **третьей главе** рассматривается время выполнения мультизапроса в многопроцессорной ВС.

Уменьшение суммарного времени выполнения запросов мультизапроса может быть достигнуто за счет совместной обработки подмножества элементарных запросов, являющегося пересечением запросов мультизапроса.

В случаях, когда запросы не удовлетворяют Утверждениям 1 и 2 эффективность совместной обработки по критерию времени выполнения мультизапроса зависит от параметров τ_i и p_i и не всегда лучше не совместной обработки запросов.

Далее мы рассмотрим именно этот случай для оценки времени выполнения мультизапроса в мультипроцессорной базе данных.

Пусть запросы мультизапроса определены следующими параметрами: k – число элементарных запросов, образующих запросы мультизапроса $З_1$, $З_2$, …, $З_v$, элементарные запросы образуют d групп по u элементарных запросов в каждой группе, каждый запрос $З_i$ $i = \overline{1, v}$, содержит две группы элементарных запросов с номерами:

1-ая группа: $ЭЗ_{(i-1)u+1}$,…, $ЭЗ_{iu}$, …, $ЭЗ_{j(v+1)u+(i-1)u+1}$,…, $ЭЗ_{j(v+1)u+iu}$,…, $ЭЗ_{(d-1)(v+1)u+(i-1)u+1}$,…, $ЭЗ_{(d-1)(v+1)u+iu}$, $j = \overline{0, d-1}$,

2-ая группа: $ЭЗ_{vu+1}$,…, $ЭЗ_{vu+u}$.

Рис. 3.1. иллюстрирует мультизапрос при значениях параметров k = 32, d =2, u=4, v = 3.

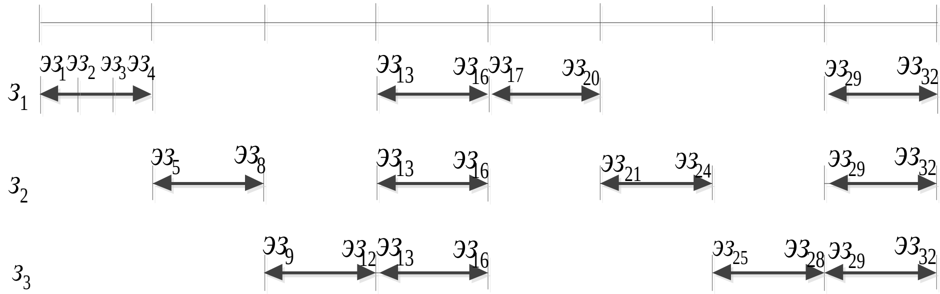


Рис. 3. 1. Мультизапрос при значениях параметров k = 32, d =2, u=4, v = 3

Определим время выполнения мультизапроса при совместном и несовместном выполнении запросов (далее для упорядоченных таблиц).

Степенная зависимость времени обработки элементарных запросов

Совместное выполнение запросов мультизапроса. СЗ

Пусть время τ_i обработки элементарных запросов в строке таблицы соответствует степенной зависимости (СЗ).

Время выполнения мультизапроса при совместном выполнении запросов:

при $r = 1$

$$\mathrm{T}_{\text{с,сов},r=1} = n\left(\frac{1-(ap)^u}{1-(ap)}\right)\left(\frac{1-\left(p^u a^{u(v+1)}\right)^d}{1-p^u a^{u(v+1)}}\right)\left(a^{uv}+p^{du}\left(\frac{1-a^{uv}}{1-a^u}\right)\right).$$

при $r > 1$

$$\mathrm{T}_{\text{с,сов},r>1,j} = n\left(a^{j-1}+pa^{2r-j}\right)\left(\frac{1-(pa^r)^{\frac{u}{r}}}{1-(pa^r)^2}\right)\left(\frac{1-\left(p^{\frac{u}{r}}a^{u(v+1)}\right)^d}{1-p^{\frac{u}{r}}a^{u(v+1)}}\right)\left(a^{uv}+p^{d\frac{u}{r}}\left(\frac{1-a^{uv}}{1-a^u}\right)\right), j=\overline{2,r}.$$

Несовместное выполнение запросов мультизапроса. СЗ

Время выполнения мультизапроса при несовместном выполнении запросов:

При $r = 1$

$$\mathrm{T}_{\text{с,несов},r=1} = n\left(\frac{1-(ap)^u}{1-(ap)}\right)\left(\frac{1-\left(p^{2u} a^{u(v+1)}\right)^d}{1-p^{2u} a^{u(v+1)}}\right)\left(vp^u a^{uv}+\left(\frac{1-a^{uv}}{1-a^u}\right)\right).$$

При $r > 1$

$$\mathrm{T}_{\text{с,несов},r>1,j} = n\left(a^{j-1}+pa^{2r-j}\right)\left(\frac{1-(pa^r)^{\frac{u}{r}}}{1-(pa^r)^2}\right)\left(\frac{1-\left(p^{2\frac{u}{r}}a^{u(v+1)}\right)^d}{1-p^{2\frac{u}{r}}a^{u(v+1)}}\right)\left(vp^{\frac{u}{r}}a^{uv}+\left(\frac{1-a^{uv}}{1-a^u}\right)\right), j=\overline{2,r}.$$

Сравнение совместной и несовместной обработки запросов. СЗ

Сравним выражения времени выполнения запроса доля несовместной и совместной обработки мультизапроса.

Совместное выполнение запросов мультизапроса имеет смысл, если выполняется неравенство $T_{с,несов} > T_{с,сов}$, т.е. если

$$\left(\frac{1-(ap)^u}{1-(ap)}\right)\left(\left(\frac{1-\left(p^{2u}a^{u(v+1)}\right)^d}{1-p^{2u}a^{u(v+1)}}\right)\left(vp^ua^{uv}+\left(\frac{1-a^{uv}}{1-a^u}\right)\right)-\left(\frac{1-\left(p^{u}a^{u(v+1)}\right)^d}{1-p^{u}a^{u(v+1)}}\right)\left(a^{uv}+p^{du}\left(\frac{1-a^{uv}}{1-a^u}\right)\right)\right)>0$$

Рассмотрим два частных случая:

1) При $d = 1$ находим

$$\left(vp^ua^{uv}+\left(\frac{1-a^{uv}}{1-a^u}\right)\right)-\left(a^{uv}+p^u\left(\frac{1-a^{uv}}{1-a^u}\right)\right)=a^{uv}(vp^u-1)+\left(\frac{1-a^{uv}}{1-a^u}\right)(1-p^u)$$

Это выражение, по крайней мере, больше 0,
если
$vp^u > 1,$
или
$p^u > \frac{1}{v}$

2) Пусть значение вероятности $p \to 0.$

Тогда, например, при $r > 1$ имеем:

$$T_{с,несов,r>1,j} = n(a^{j-1}+pa^{2r-j})\frac{a^{vu}-1}{a^u-1}+o(p)$$

$$T_{с,сов,r>1,j} = n(a^{j-1}+pa^{2r-j})a^{vu}+o(p)$$

Поэтому совместная обработка запросов требует меньше времени, чем несовместная, если

$$T_{с,несов,r>1,j}-T_{с,сов,r>1,j} = n(a^{j-1}+pa^{2r-j})\left(\frac{a^{vu}-1}{a^u-1}-a^{vu}\right)+o(p)>0\,, j=\overline{2,r},$$

или

$$\left(\frac{a^{vu}-1}{a^u-1}-a^{vu}\right) > 0\,,$$

или если

$$a^u+\frac{1}{a^{vu}}<2.$$

Например, для
$a = 1.1, v = 4, u = 6$
находим:

$1.1^6+\frac{1}{1.1^{24}}<2$ или $1.8731 < 2.$

Поэтому здесь совместная обработка запросов требует меньше времени, чем несовместная.

Для $a = 1.1, v = 4, u = 8$ получаем:

$$1.1^8 + \frac{1}{1.1^{32}} > 2\,,$$

т.е. здесь несовместная обработка запросов требует меньше времени, чем совместная.

В диссертации и на рис. 3.2 представлены сравнительные результаты расчетов времени совместной и несовместной обработки запросов мультизапроса.

Рис.3.2. Время выполнения мультизапроса при совместной и несовместной обработки запросов/n (a =1.2, $0.8 > p > 0.7$). СЗ

Очевидно, что совместная обработка запросов мультизапроса обеспечивает меньшее время по отношению к несовместной обработке при возрастании вероятности успеха, кроме того, увеличение числа процессоров может привести не к уменьшению, а к увеличению времени выполнения мультизапроса.

Линейная зависимость времени обработки элементарных запросов
Совместное выполнение запросов мультизапроса. ЛЗ

Пусть время τ_i обработки элементарных запросов в строке таблицы соответствует линейной зависимости (ЛЗ).

Время выполнения мультизапроса при совместном выполнении запросов:

При $r = 1$

$$T_{л,сов,r=1} = n\frac{1-p^u}{1-p}\left[(1+uv\Delta)\frac{1-p^{2du}}{1-p^{2u}} + p^{2u}u\Delta(v+1)\frac{1-dp^{(d-1)2u}+(d-1)p^{2ud}}{(1-p^{2u})^2} + p^{du}\left(v\frac{1-p^{2du}}{1-p^{2u}} + u\Delta\frac{(v-1)v}{2}\frac{1-p^{2du}}{1-p^{2u}} + v(v+1)u\Delta p^{2u}\frac{1-dp^{(d-1)2u}+(d-1)p^{2ud}}{(1-p^{2u})^2}\right)\right] + p\Delta\frac{1-up^{u-1}+(u-1)p^u}{(1-p)^2}\frac{1-p^{2du}}{1-p^{2u}}(v+p^u).$$

При $r > 1$

$$\text{Т}_{л,сов,r>1,j} = n.\frac{1-p^{\frac{u}{r}}}{1-p^2}\left[(1+(i-1)\Delta+(1+(2r-i)\Delta)p)\frac{1-p^{d\frac{u}{r}}}{1-p^{\frac{u}{r}}}\left(1+vp^{d\frac{u}{r}}\right)+\frac{1-p^{d\frac{u}{r}}}{1-p^{\frac{u}{r}}}uv\left(1+\frac{v-1}{2}p^{d\frac{u}{r}}\right)\right.$$
$$\left.+(v+1)u\left(1+vp^{d\frac{u}{r}}\right)p^{\frac{u}{r}}\left(\frac{1-dp^{\frac{u}{r}(d-1)}+(d-1)p^{d\frac{u}{r}}}{\left(1-p^{\frac{u}{r}}\right)^2}\right)\right]$$
$$+2r\Delta p^2(1+p)\left(\frac{1-\frac{u}{2r}p^{\frac{u}{r}-2}+\left(\frac{u}{2r}-1\right)p^{\frac{u}{r}}}{(1-p^2)^2}\right)\left(\frac{1-p^{d\frac{u}{r}(v+1)}}{p^{\frac{u}{r}}}\right)$$

Несовместное выполнение запросов мультизапроса. ЛЗ

Время выполнения мультизапроса при несовместном выполнении запросов:
При, $r = 1$

$$\text{Т}_{л,несов,r=1} = n\frac{1-p^u}{1-p}\left[v\frac{1-p^{2du}}{1-p^{2u}}+u\Delta\frac{(v-1)v}{2}\frac{1-p^{2du}}{1-p^{2u}}+v(v+1)u\Delta p^{2u}\frac{1-dp^{(d-1)2u}+(d-1)p^{2ud}}{(1-p^{2u})^2}\right.$$
$$\left.+vp^u\left((1+uv\Delta)\frac{1-p^{2du}}{1-p^{2u}}+p^{2u}u\Delta(v+1)\frac{1-dp^{(d-1)2u}+(d-1)p^{2ud}}{(1-p^{2u})^2}\right)\right]$$
$$+vp\Delta\frac{1-up^{u-1}+(u-1)p^u}{(1-p)^2}\frac{1-p^{2du}}{1-p^u}$$

При $r > 1$

$$\text{Т}_{л,несов,r>1,j} = n\frac{1-p^{\frac{u}{r}}}{1-p^2}\left[((1+(i-1)\Delta)+(1+(2r-i)\Delta)p)\left(\frac{1-p^{2\frac{u}{r}d}}{1-p^{2\frac{u}{r}}}\right)\left(1+p^{\frac{u}{r}}\right)v+p^{\frac{u}{r}}uv^2\Delta\left(\frac{1-p^{2\frac{u}{r}d}}{1-p^{2\frac{u}{r}}}\right)\right.$$
$$+p^{2\frac{u}{r}}(v+1)vu\Delta\left(1+p^{\frac{u}{r}}\right)\left(\frac{1-dp^{\frac{u}{r}(d-1)}+(d-1)p^{\frac{u}{r}d}}{(1-p^2)^2}\right)+u\Delta\left(\frac{v(v-1)}{2}\right)\left(\frac{1-p^{2\frac{u}{r}d}}{1-p^{2\frac{u}{r}}}\right)$$
$$\left.+2vr\Delta p^2(1+p)\left(\frac{1-\frac{u}{2r}p^{\frac{u}{r}-2}+\left(\frac{u}{2r}-1\right)p^{\frac{u}{r}}}{(1-p^2)^2}\right)\left(\frac{1-p^{2\frac{u}{r}d}}{p^{\frac{u}{r}}}\right)\right]$$

Сравнение совместной и несовместной обработки запросов. ЛЗ

Рассмотрим два случая. При $r = 1, d = 1$ имеем:

$$\text{Т}_{л,несов,r=1}(d=1) = \frac{1-p^k}{1-p}\left(v+u\Delta\frac{v(v-1)}{2}+p^u v(1+vu\Delta)\right)+(v+vp^u)\,p\Delta\left(\frac{1-p^k}{1-p}\right)'_p$$
$$\text{Т}_{л,сов,r=1}(d=1) = \frac{1-p^k}{1-p}\left((1+vu\Delta)+p^u\left(v+u\Delta\frac{v(v-1)}{2}\right)\right)+(1+vp^u)\,p\Delta\left(\frac{1-p^k}{1-p}\right)'_p$$

Тогда,

$$\text{Т}_{л,несов,r=1}(d=1)-\text{Т}_{а,сов,r=1}(d=1)$$
$$=\frac{1-p^k}{1-p}\left((v-1)+u\Delta\left(\frac{v(v-1)}{2}-v\right)+p^u(u\Delta)v\left(v-\frac{(v-1)}{2}\right)\right)$$
$$+\frac{1-kp^{k-1}+(k-1)p^k}{(1-p)^2}(v-1)$$

Легко показать, что если $v \geq 2$, то

$$Т_{л,несов,r=1}(d=1) - Т_{л,сов,r=1}(d=1)$$

всегда больше нуля, т.е. совместная обработка запросов требует меньше времени, чем несовместная.

1) Пусть значение вероятности $p \to 0$.

Тогда при $r > 1$ имеем:

$$Т_{л,несов,r>1,j} = v\left(1 + (j-1)\Delta + u\Delta\frac{(v-1)}{2}\right) + vp\left(1 + (2r-j)\Delta + u\Delta\frac{(v-1)}{2}\right) + o(p)$$

$$Т_{л,сов,r>1,j} = (1 + (j-1)\Delta + uv\Delta) + p(1 + (2r-j)\Delta + uv\Delta) + o(p)$$

При $v > 2$ всегда совместная обработка запросов требует меньше времени, чем несовместная.

При $v = 2$ имеем:

$$T_{несов_j} = 2\left(1 + (j-1)\Delta + u\frac{1}{2}\Delta\right) + 2p\left(1 + (2r-j)\Delta + u\Delta\frac{1}{2}\right) + o(p)$$

$$T_{сов_j} = 1 + (j-1)\Delta + 2u\Delta + p(1 + (2r-j)\Delta + 2u\Delta) + o(p), j = \overline{2,r},$$

$$Т_{л,несов,r>1,j} - Т_{л,сов,r>1,j} = 1 + (j-1)\Delta - u\Delta + p(1 + (2r-j)\Delta) - pu\Delta + o(p),$$

Следовательно, совместная обработка запросов требует меньше времени, чем несовместная, когда

$u < j - 1 + 1/\Delta$.

В диссертации представлены сравнительные результаты расчетов времени совместной и несовместной обработки запросов мультизапроса.

Совместная обработка запросов мультизапроса обеспечивает всегда меньшее время по отношению к несовместной обработке при $d = 1$, но при $d > 1$ это не всегда, кроме того, увеличение числа процессоров может привести не к уменьшению, а к увеличению времени выполнения мультизапроса.

В **четвертой главе** будем считать, что реализация плана выполнения мультизапроса осуществляется в многопроцессорной базе данных.

Не умоляя общих выводов, далее проведем исследование для неупорядоченных таблиц.

Реализация плана выполнения мультизапроса в многопроцессорной базе данных требует учета порядка выполнения элементарных запросов.

Оценка влияния числа процессоров на время выполнения мультизапроса.

Пусть мультизапрос состоит из пяти со сложной структурой запросов, образованных 24-мя элементарными запросами, см. Рис. 4.1:

- $З_1 = ЭЗ_1 \& ЭЗ_2 \& ЭЗ_3 \& ЭЗ_4 \& ЭЗ_5 \& ЭЗ_6 \& ЭЗ_7 \& ЭЗ_8 \& ЭЗ_9 \& ЭЗ_{10} \& ЭЗ_{11} \& ЭЗ_{12} \& ЭЗ_{13} \& ЭЗ_{14}$ $\& ЭЗ_{15} \& ЭЗ_{16} \& ЭЗ_{17} \& ЭЗ_{18}\ ЭЗ_{19} \& ЭЗ_{20} \& ЭЗ_{21} \& ЭЗ_{22} \& ЭЗ_{23} \& ЭЗ_{24}$
- $З_2 = ЭЗ_1 \& ЭЗ_2 \& ЭЗ_3 \& ЭЗ_4 \& ЭЗ_9 \& ЭЗ_{10} \& ЭЗ_{11} \& ЭЗ_{12} \& ЭЗ_{17} \& ЭЗ_{18} \& ЭЗ_{19} \& ЭЗ_{20}$
- $З_3 = ЭЗ_3 \& ЭЗ_4 \& ЭЗ_5 \& ЭЗ_6 \& ЭЗ_{11} \& ЭЗ_{12} \& ЭЗ_{13} \& ЭЗ_{14} \& ЭЗ_{19} \& ЭЗ_{20} \& ЭЗ_{21} \& ЭЗ_{22}$
- $З_4 = ЭЗ_5 \& ЭЗ_6 \& ЭЗ_7 \& ЭЗ_8 \& ЭЗ_{13} \& ЭЗ_{14} \& ЭЗ_{15} \& ЭЗ_{16} \& ЭЗ_{21} \& ЭЗ_{22} \& ЭЗ_{23} \& ЭЗ_{24}$
- $З_5 = ЭЗ_1 \& ЭЗ_2 \& ЭЗ_3 \& ЭЗ_7 \& ЭЗ_8 \& ЭЗ_9 \& ЭЗ_{13} \& ЭЗ_{14} \& ЭЗ_{15} \& ЭЗ_{19} \& ЭЗ_{20} \& ЭЗ_{21}$

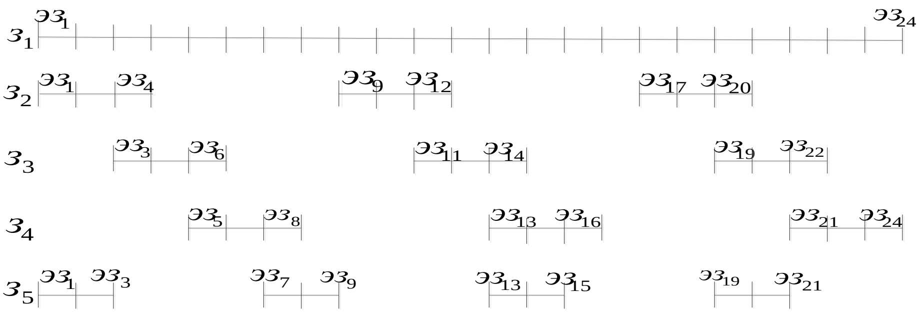


Рис. 4.1. Мультизапрос, состоящий из пяти запросов.

Время выполнения конъюнктивного мультизапроса при независимой обработке равно сумме времен обработки запросов $З_1$, $З_2$, $З_3$, $З_4$и $З_5$.

Пусть мультизапрос образует конъюнкции элементарных запросов:

- со степенным изменением параметра времени выполнения элементарных запросов по правилу $\tau_i = a^{i-1}$ и со значением параметра вероятности $p_i = p, i \in 1, \ldots$ и
- с линейным изменением параметра времени по правилу $\tau_i = 1+(i-1)\Delta$ и со значением параметра вероятности $p_i = p, i \in 1, \ldots$.

Совместная обработка запросов мультизапроса

Обработка элементарных запросов мультизапроса может быть выполнена, используя 3 возможных алгоритма.

Алгоритм 1

В алгоритме запрос 1,2 и 4 образуют опорные запросы, элементарные запросы которого образуют общее подмножество и выполняются первыми. Запрос 2 и 3 образуют опорные запросы, элементарные запросы $ЭЗ_3, ЭЗ_{13}, ЭЗ_{14}, ЭЗ_{19}, ЭЗ_{20}, ЭЗ_{21}$ которого образуют общее подмножество и выполняются первыми.

Шаг 1: Выделяем элементарный запрос $ЭЗ_1$ на основе $\cap\, З_{i(i=\overline{1,2})} = ЭЗ_1$

Шаг 2: Выделяем элементарный запрос $ЭЗ_2$ на основе $\cap З_{i(i=\overline{1,2})} \setminus (ЭЗ_1\ ) = ЭЗ_2$
Шаг 3: Выделяем элементарный запрос $ЭЗ_3$ на основе $\cap З_{i(i=\overline{1,2})} \setminus (ЭЗ_1\ \&\ ЭЗ_2) = ЭЗ_3$
Шаг 4: Выделяем элементарный запрос $ЭЗ_4$ на основе
$\cap З_{i(i=\overline{1,2})} \setminus (ЭЗ_1 \& ЭЗ_2 \&\ ЭЗ_3) =\ ЭЗ_4$
Шаг 5: Выделяем элементарный запрос $ЭЗ_9$ на основе
$\cap З_{i(i=\overline{1,2})} \setminus (ЭЗ_1 \& ЭЗ_2 \&\ ЭЗ_3 \& ЭЗ_4) =\ \ ЭЗ_9$
Шаг 6: Выделяем элементарный запрос $ЭЗ_{10}$ на основе
$\cap З_{i(i=\overline{1,2})} \setminus (ЭЗ_1 \& ЭЗ_2 \&\ ЭЗ_3 \& ЭЗ_4 \&\ ЭЗ_9) =\ ЭЗ_{10}$
Шаг 7: Выделяем элементарный запрос $ЭЗ_{11}$ на основе
$\cap З_{i(i=\overline{1,2})} \setminus (ЭЗ_1 \& ЭЗ_2 \&\ ЭЗ_3 \& ЭЗ_4 \&\ ЭЗ_9 \& ЭЗ_{10}) =\ \ ЭЗ_{11}$
Шаг 8: Выделяем элементарный запрос $ЭЗ_{12}$ на основе
$\cap З_{i(i=\overline{1,2})} \setminus (ЭЗ_1 \& ЭЗ_2 \&\ ЭЗ_3 \& ЭЗ_4 \&\ ЭЗ_9 \& ЭЗ_{10} \& ЭЗ_{11}) =\ \ ЭЗ_{12}$
Шаг 9: Выделяем элементарный запрос $ЭЗ_{17}$ на основе
$\cap З_{i(i=\overline{1,2})} \setminus (ЭЗ_1 \& ЭЗ_2 \&\ ЭЗ_3 \& ЭЗ_4 \&\ ЭЗ_9 \& ЭЗ_{10} \& ЭЗ_{11} \&\ ЭЗ_{12}) =\ ЭЗ_{17}$
Шаг 10: Выделяем элементарный запрос $ЭЗ_{18}$ на основе
$\cap З_{i(i=\overline{1,2})} \setminus (ЭЗ_1 \& ЭЗ_2 \&\ ЭЗ_3 \& ЭЗ_4 \&\ ЭЗ_9 \& ЭЗ_{10} \& ЭЗ_{11} \&\ ЭЗ_{12} \& ЭЗ_{17}) =\ \ ЭЗ_{18}$
Шаг 11: Выделяем элементарный запрос $ЭЗ_{19}$ на основе
$\cap З_{i(i=\overline{1,2})} \setminus (ЭЗ_1 \& ЭЗ_2 \&\ ЭЗ_3 \& ЭЗ_4 \&\ ЭЗ_9 \& ЭЗ_{10} \& ЭЗ_{11} \&\ ЭЗ_{12} \& ЭЗ_{17} \&\ ЭЗ_{18}) =\ ЭЗ_{19}$
Шаг 12: Выделяем элементарный запрос $ЭЗ_{20}$ на основе
$\cap З_{i(i=\overline{1,2})} \setminus (ЭЗ_1 \& ЭЗ_2 \&\ ЭЗ_3 \& ЭЗ_4 \&\ ЭЗ_9 \& ЭЗ_{10} \& ЭЗ_{11} \&\ ЭЗ_{12} \& ЭЗ_{17} \& ЭЗ_{18}) =\ \ ЭЗ_{20}$
Шаг 13: Выделяем элементарный запрос $ЭЗ_5$ на основе $\cap З_{i(i=\overline{1,4})} = ЭЗ_5$
Шаг 14: Выделяем элементарный запрос $ЭЗ_6$ на основе $\cap З_{i(i=\overline{1,4})} \setminus (ЭЗ_5\ ) = ЭЗ_6$
Шаг 15: Выделяем элементарный запрос $ЭЗ_7$ на основе
$\cap З_{i(i=\overline{1,4})} \setminus (ЭЗ_5\ \&\ ЭЗ_6) =\ ЭЗ_7$
Шаг 16: Выделяем элементарный запрос $ЭЗ_8$ на основе
$\cap З_{i(i=\overline{1,4})} \setminus (ЭЗ_5\ \&\ ЭЗ_6 \&\ ЭЗ_7) =\ ЭЗ_8$
Шаг 17: Выделяем элементарный запрос $ЭЗ_{13}$ на основе
$\cap З_{i(i=\overline{1,4})} \setminus (ЭЗ_5\ \&\ ЭЗ_6 \&\ ЭЗ_7 \& ЭЗ_8) =\ \ ЭЗ_{13}$
Шаг 18: Выделяем элементарный запрос $ЭЗ_{14}$ на основе
$\cap З_{i(i=\overline{1,4})} \setminus (ЭЗ_5\ \&\ ЭЗ_6 \&\ ЭЗ_7 \& ЭЗ_8 \&\ ЭЗ_{13}) =\ ЭЗ_{14}$
Шаг 18: Выделяем элементарный запрос $ЭЗ_{15}$ на основе
$\cap З_{i(i=\overline{1,4})} \setminus (ЭЗ_5\ \&\ ЭЗ_6 \&\ ЭЗ_7 \& ЭЗ_8 \&\ ЭЗ_{13} \& ЭЗ_{14}) =\ \ ЭЗ_{15}$
Шаг 20: Выделяем элементарный запрос $ЭЗ_{16}$ на основе
$\cap З_{i(i=\overline{1,4})} \setminus (ЭЗ_5\ \&\ ЭЗ_6 \&\ ЭЗ_7 \& ЭЗ_8 \&\ ЭЗ_{13} \& ЭЗ_{14} \& ЭЗ_{15}) =\ \ ЭЗ_{16}$
Шаг 21: Выделяем элементарный запрос $ЭЗ_{21}$ на основе
$\cap З_{i(i=\overline{1,4})} \setminus (ЭЗ_5\ \&\ ЭЗ_6 \&\ ЭЗ_7 \& ЭЗ_8 \&\ ЭЗ_{13} \& ЭЗ_{14} \& ЭЗ_{15} \&\ ЭЗ_{16}) =\ ЭЗ_{21}$
Шаг 22: Выделяем элементарный запрос $ЭЗ_{22}$ на основе
$\cap З_{i(i=\overline{1,4})} \setminus (ЭЗ_5\ \&\ ЭЗ_6 \&\ ЭЗ_7 \& ЭЗ_8 \&\ ЭЗ_{13} \& ЭЗ_{14} \& ЭЗ_{15} \&\ ЭЗ_{16} \& ЭЗ_{21}) =\ \ ЭЗ_{22}$

Шаг 23: Выделяем элементарный запрос $ЭЗ_{23}$ на основе
$\cap З_{i(i=\overline{1,4})} \backslash (ЭЗ_5 \,\&\, ЭЗ_6 \& \, ЭЗ_7 \& ЭЗ_8 \& \, ЭЗ_{13} \& ЭЗ_{14} \& ЭЗ_{15} \& \, ЭЗ_{16} \& ЭЗ_{21} \& \, ЭЗ_{22}) = \; ЭЗ_{23}$
Шаг 24: Выделяем элементарный запрос $ЭЗ_{24}$ на основе
$\cap З_{i(i=\overline{1,4})} \backslash (ЭЗ_5 \,\&\, ЭЗ_6 \& \, ЭЗ_7 \& ЭЗ_8 \& \, ЭЗ_{13} \& ЭЗ_{14} \& ЭЗ_{15} \& \, ЭЗ_{16} \& ЭЗ_{21} \& \, ЭЗ_{22} \& ЭЗ_{23}) = \; ЭЗ_{24}$
Шаг 25: Выделяем элементарный запрос $ЭЗ_3$ на основе $\cap З_{i(i=\overline{3,5})} = ЭЗ_3$
Шаг 26: Выделяем элементарный запрос $ЭЗ_{13}$ на основе $\cap З_{i(i=\overline{3,5})} \backslash (ЭЗ_3) = ЭЗ_{13}$
Шаг 27: Выделяем элементарный запрос $ЭЗ_{14}$ на основе
$\cap З_{i(i=\overline{3,5})} \backslash (ЭЗ_3 \,\&\, ЭЗ_{13}) = \; ЭЗ_{14}$
Шаг 28: Выделяем элементарный запрос $ЭЗ_{19}$ на основе
$\cap З_{i(i=\overline{3,5})} \backslash (ЭЗ_3 \,\&\, ЭЗ_{13} \& ЭЗ_{14}) = \; ЭЗ_{19}$
Шаг 29: Выделяем элементарный запрос $ЭЗ_{20}$ на основе
$\cap З_{i(i=\overline{3,5})} \backslash (ЭЗ_3 \, \& ЭЗ_{13} \& ЭЗ_{14} \& ЭЗ_{19}) = \; ЭЗ_{20}$
Шаг 30: Выделяем элементарный запрос $ЭЗ_{21}$ на основе
$\cap З_{i(i=\overline{3,5})} \backslash (ЭЗ_3 \, \& ЭЗ_{13} \& ЭЗ_{14} \& ЭЗ_{19} \& ЭЗ_{20}) = \; ЭЗ_{21}$
Далее подмножество совпадающих запросов становится пустым.

Граф обработки запросов по алгоритму 1 приведен на рис. 4.2.

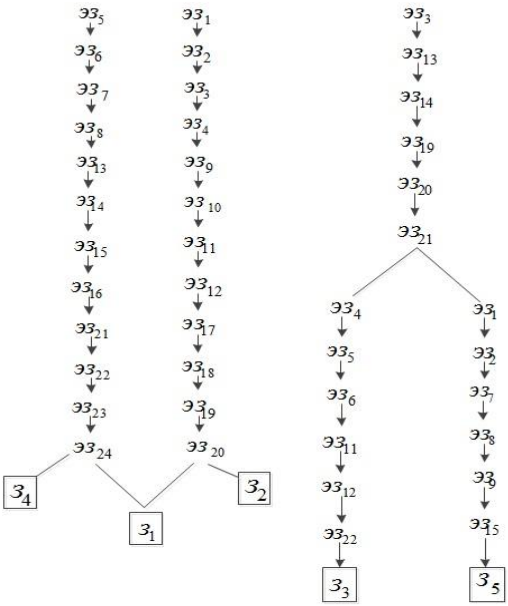


Рис. 4.2. Граф обработки запросов по алгоритму 1.

Минимальное время выполнения мультизапроса при степенном изменении параметра времени

Оптимальное время выполнения мультизапроса при степенном изменении параметра времени для r процессоров определяются в диссертации, в частности, для трех процессоров при распределении запросов по процессорам в соответствии с Таблицей

Число процессоров	Порядок обработки ЭЗ
1	1,2,3,4,9,10,11,12,17,18,19,20
2	5,6,7,8,13,14,15,16,21,22,23,24
3	3,13,14,19,20,21,4,5,6,11,12,22,1,2,7,8,9,15

$$T^{a_1}_{\text{ст,совм},1,3} = n\left(\left(1 + p^4a^8 + (p^4a^8)^2\right)\left(\frac{1-(pa)^3}{1-(pa)}\right)\right)$$

$$T^{a_1}_{\text{ст,совм},2,3} = n\left(a^4\left(\left(1 + p^4a^8 + (p^4a^8)^2\right)\left(\frac{1-(pa)^3}{1-(pa)}\right)\right)\right)$$

$$T^{a_1}_{\text{ст,совм},3,3} = n(a^2 + pa^{12} + p^2a^{13} + p^3a^{18} + p^4a^{19} + p^5a^{20} + p^6a^3 + p^7a^4 + p^8a^5 + p^9a^{10} + p^{10}a^{11} + p^{11}a^{21} + p^6 + p^7a + p^8a^6 + p^9a^7 + p^{10}a^8 + p^{11}a^{14})$$

С учетом завершения работы трех процессоров время выполнения мультизапроса равно максимальному времени для этих процессоров:
$T^{a_1}_{\text{ст,совм},3} = n\,(\max(T^{a_1}_{\text{ст,совм},1,3}, T^{a_1}_{\text{ст,совм},2,3}, T^{a_1}_{\text{ст,совм},3,3}))$.

Минимальное время выполнения мультизапроса при линейном изменении параметра времени

Оптимальное время выполнения мультизапроса при линейном изменении параметра времени для r определяются в диссертации, в частности, для трех процессоров при распределении запросов по процессорам в соответствии с Таблицей

Число процессоров	Порядок обработки ЭЗ
1	1,2,3,4,9,10,11,12,17,18,19,20
2	5,6,7,8,13,14,15,16,21,22,23,24
3	3,13,14,19,20,21,4,5,6,11,12,22,1,2,7,8,9,15

$$T^{a_1}_{\text{л,совм},1,3} = n\big(1 + p(1+\Delta) + p^2(1+\Delta 2) + p^3(1+\Delta 3) + p^4(1+\Delta 8) + p^5(1+\Delta 9) + p^6(1+\Delta 10) + p^7(1+\Delta 11) + p^8(1+\Delta 16) + p^9(1+\Delta 17) + p^{10}(1+\Delta 18) + p^{11}(1+\Delta 19)\big)$$

$$T^{a_1}_{\text{л,совм},2,3} = n\big((1+\Delta 4) + p(1+\Delta 5) + p^2(1+\Delta 6) + p^3(1+\Delta 7) + p^4(1+\Delta 12) + p^5(1+\Delta 13) + p^6(1+\Delta 14) + p^7(1+\Delta 15) + p^8(1+\Delta 20) + p^9(1+\Delta 21) + p^{10}(1+\Delta 22) + p^{11}(1+\Delta 23)\big)$$

$$\mathrm{T}^{a_1}_{\text{л,совм,3,3}} = n\big((1+\Delta 2) + p(1+\Delta 12) + p^2(1+\Delta 13) + p^3(1+\Delta 18) + p^4(1+\Delta 19) + p^5(1+\Delta 20) + p^6(1+\Delta 3) + p^7(1+\Delta 4) + p^8(1+\Delta 5) + p^9(1+\Delta 10) + p^{10}(1+\Delta 11) + p^{11}(1+\Delta 21) + p^6 + p^7(1+\Delta) + p^8(1+\Delta 6) + p^9(1+\Delta 7) + p^{10}(1+\Delta 8) + p^{11}(1+\Delta 14)\big)$$

С учетом завершения работы трех процессоров время выполнения мультизапроса равно максимальному времени для этих процессоров:

$\mathrm{T}^{a_1}_{\text{л,совм,3}} = n\ (\max(\mathrm{T}^{a_1}_{\text{л,совм,1,3}}, \mathrm{T}^{a_1}_{\text{л,совм,2,3}}, \mathrm{T}^{a_1}_{\text{л,совм,3,3}}))$

Алгоритм 2

В алгоритме 2, запросы 1, 2 и 3 образуют опорные запросы, элементарные запросы $ЭЗ_3, ЭЗ_4, ЭЗ_{11}, ЭЗ_{12}, ЭЗ_{19}, ЭЗ_{20}$ которых образуют общее подмножество и выполняются первыми. Запрос 4 и 5 образуют опорные запросы, элементарные запросы $ЭЗ_7, ЭЗ_8, ЭЗ_{13}, ЭЗ_{14}, ЭЗ_{15}, ЭЗ_{21}$ которыхобразуют общее подмножество и выполняются первыми.

Алгоритм 3

В алгоритме 3, запросы 2,3 и 5 образуют опорные запросы, элементарные запросы $ЭЗ_3, ЭЗ_{19}, ЭЗ_{20}$ которых образуют общее подмножество и выполняются первыми. Запрос 1 и 4 образуют опорные запросы, элементарные запросы $ЭЗ_5, ЭЗ_6, ЭЗ_7, ЭЗ_8, ЭЗ_{13}, ЭЗ_{14}, ЭЗ_{15}, ЭЗ_{16}, ЭЗ_{21}, ЭЗ_{22}, ЭЗ_{23}, ЭЗ_{24}$ которых образуют общее подмножество и выполняются первыми.

Формирование оптимального плана выполнения мультизапроса при степенном изменении параметра времени

В диссертации и таблицах 4.3 и 4.4 приведены минимальные времена выполнения мультизапроса при совместном и несовместном выполнении образующих его запросов при степенном и линейном изменениях параметра времени для алгоритмов 1, 2 и 3 для r процессоров.

Таблица 4.3

p	*r*	K=24				
		a=1.1				
		$\mathrm{T}^{a_1}_{\text{ст,совм}}$	$\mathrm{T}^{a_2}_{\text{ст,совм}}$	$\mathrm{T}^{a_3}_{\text{ст,совм}}$	$\mathrm{T}_{\text{ст,несовм}}$	
0.4	1	**5.34538**	6.0291	7.5722	9.7590	$\mathrm{T}^{a_1}_{\text{ст,совм}}$
	2	4.25837	**3.4629**	4.9101	5.4431	$\mathrm{T}^{a_2}_{\text{ст,совм}}$
	3	3.61718	**3.45479**	5.7207	4.1541	$\mathrm{T}^{a_2}_{\text{ст,совм}}$
	4	3.6070	3.5504	5.7207	**3.3520**	$\mathrm{T}_{\text{ст,несовм}}$
0.5	1	**7.15334**	8.0586	10.1379	12.0486	$\mathrm{T}^{a_1}_{\text{ст,совм}}$
	2	5.25824	**4.5297**	6.7324	6.7021	$\mathrm{T}^{a_2}_{\text{ст,совм}}$

	3	5.01863	**4.48819**	6.8634	5.1104	$Т^{a_2}_{ст,совм}$
	4	4.9686	5.0273	6.8634	**4.1200**	$Т_{ст,несовм}$
0.6	1	**10.0551**	11.5634	14.2044	16.0049	$Т^{a_1}_{ст,совм}$
	2	7.20652	**6.3466**	9.4697	8.9066	$Т^{a_2}_{ст,совм}$
	3	7.20652	**6.17144**	8.3802	6.7886	$Т^{a_2}_{ст,совм}$
	4	7.0062	7.2839	8.3802	**5.4708**	$Т_{ст,несовм}$
0.7	1	**15.1211**	18.2678	21.4017	23.6827	$Т^{a_1}_{ст,совм}$
	2	10.8309	**9.7939**	13.9863	13.1891	$Т^{a_2}_{ст,совм}$
	3	10.8309	**9.15141**	10.5715	10.0526	$Т^{a_2}_{ст,совм}$
	4	10.1293	10.6360	10.5715	**8.1011**	$Т_{ст,несовм}$
0.8	1	**25.0494**	32.7634	35.7316	40.2614	$Т^{a_1}_{ст,совм}$
	2	17.3258	**17.0313**	22.1869	22.0741	$Т^{a_2}_{ст,совм}$
	3	17.3107	15.732	**14.1557**	16.8294	$Т^{a_3}_{ст,совм}$
	4	15.0944	15.4833	14.1557	**13.5662**	$Т_{ст,несовм}$
0.9	1	**48.2132**	69.1557	68.3955	80.6487	$Т^{a_1}_{ст,совм}$
	2	**32.0877**	35.9788	38.2220	40.9444	$Т^{a_1}_{ст,совм}$
	3	**29.8064**	35.9788	30.1735	31.2313	$Т^{a_1}_{ст,совм}$
	4	25.2942	**24.8870**	25.2942	25.1877	$Т^{a_2}_{ст,совм}$

Таблица 4.4

p	*r*	K=24				
		Δ=0.4				
		$Т^{a_1}_{л,совм}$	$Т^{a_2}_{л,совм}$	$Т^{a_3}_{л,совм}$	$Т_{л,несовм}$	
0.4	1	**8.15808**	10.6973	12.106	16.1353	$Т^{a_1}_{л,совм}$
	2	**5.97696**	6.53988	7.25843	8.3540	$Т^{a_1}_{л,совм}$
	3	**5.96663**	6.52999	9.88252	7.9178	$Т^{a_1}_{л,совм}$
	4	5.9595	6.51581	9.88252	**5.6703**	$Т_{л,несовм}$
0.5	1	**11.2456**	14.18	16.2116	22.1801	$Т^{a_1}_{л,совм}$
	2	**8.23438**	8.39707	10.0037	11.5596	$Т^{a_1}_{л,совм}$
	3	**8.17979**	8.34307	11.5969	10.9580	$Т^{a_1}_{л,совм}$
	4	8.1483	8.27275	11.5969	**7.8206**	$Т_{л,несовм}$
0.6	1	**16.238**	19.9814	22.7508	33.2437	$Т^{a_1}_{л,совм}$

	2	11.6711	**11.4154**	14.2163	17.4571	$T^{a_2}_{л,совм}$
	3	11.4406	**11.1763**	13.8943	16.5505	$T^{a_2}_{л,совм}$
	4	11.3282	**11.0136**	13.8943	11.7867	$T^{a_2}_{л,совм}$
0.7	1	**25.1876**	30.6837	34.306	55.3091	$T^{a_1}_{л,совм}$
	2	17.3056	**16.8481**	21.3675	29.1793	$T^{a_2}_{л,совм}$
	3	16.4742	**15.9402**	17.3171	27.6644	$T^{a_2}_{л,совм}$
	4	16.1195	**15.4215**	17.3171	19.6927	$T^{a_2}_{л,совм}$
0.8	1	**44.0209**	53.1009	57.1404	102.8575	$T^{a_1}_{л,совм}$
	2	**27.4158**	27.6947	34.7528	53.8058	$T^{a_1}_{л,совм}$
	3	24.7617	25.4062	**23.1764**	51.0104	$T^{a_3}_{л,совм}$
	4	23.7322	**22.0195**	23.1764	36.3398	$T^{a_2}_{л,совм}$
0.9	1	**95.7552**	107.874	108.735	213.8461	$T^{a_1}_{л,совм}$
	2	**47.123**	56.9411	61.6042	106.9504	$T^{a_1}_{л,совм}$
	3	**39.4506**	56.9411	47.1313	101.3358	$T^{a_1}_{л,совм}$
	4	**39.2801**	39.3514	**39.2801**	72.2773	$T^{a_1}_{л,совм}$

В приведенных таблицах жирным шрифтом выделены значения, для которых достигается минимальное время для конкретных значений числа процессоров *r* и вероятности успеха *p*.

Непосредственно из таблиц следует, что при степенном возрастании параметра времени (*a* =1.1, 1.15) для достижения минимального времени выполнения мультизапроса наблюдается неустойчивость в смысле использования конкретного алгоритма и порядка выполнения (совместного и несовместного) образующих запросов.

При линейном возрастании параметра времени (Δ =0.4, 0,5) предпочтительной по достижению минимального времени выполнения мультизапроса является совместное выполнение мультизапроса по алгоритмам 1 и 2, однако для конкретных значений числа процессоров *r* и вероятности успеха *p* требуется выбор алгоритма 1 или алгоритма 2.

ОСНОВНЫЕ РЕЗУЛЬТАТЫ РАБОТЫ

- Предложен и обоснован метод оптимизации по времени выполнения мультизапроса при обращении к базе данных на основе упорядочивания элементарных запросов.
- Доказаны условия, при которых совместная обработка конъюнктивного мультизапроса обеспечивает не большее время выполнения по отношению к независимой обработке.
- Доказано, что параметр вероятность успеха при выполнении элементарного запроса является существенным параметром, влияющим как на выбор совместного и несовместного метода обработки мультизапроса, так и на определение числа процессоров.
- Разработан метод обеспечения оптимизации многопроцессорной обработки мультизапросов.
- Предложен оптимальный алгоритм распределения элементарных запросов на процессоры.

Основные публикации по теме диссертации

1. Брехов О.М., Вунна Джо Джо, Тан Хлаинг Мьинт. Оптимизация плана выполнения мульти и вложенных запросов // Журнал «Наукоёмкие технологии» 2014г. №1, с. 101-106.
2. Брехов О.М., Вунна Джо Джо. Оценка времени выполнения мультизапроса // Электронный журнал «Труды МАИ», 2014г. №(76).

Тираж 100 экз
Отпечатно в Московском авиационном институте (национальном исследовательском университете)
г. Москва, А-80, ГСП-3, Волоколамское шоссе, д.4.
http://www.mai.ru//