

*кандидат технічних наук, доцент  
доцент кафедри інформаційних систем і технологій в діяльності ОВС  
Харківського національного університету внутрішніх справ*

**Д. О. РУДЕНКО,**

*кандидат технічних наук, доцент  
доцент кафедри інформатики  
Харківського національного університету радіоелектроніки*

**І. О. ЯКОВЛЕВА,**

*кандидат технічних наук, доцент  
професор кафедри інформаційних технологій і систем управління  
Університету цивільного захисту України*

## **МІНІМІЗАЦІЯ РЕЗУЛЬТАТУ З'ЄДНАННЯ ВІДНОШЕНЬ У ЗАПИТАХ РЕЛЯЦІЙНОЇ БАЗИ ДАНИХ**

Метою виконання процедури **обробки запитів** є перетворення цього запиту, написаного на мові високого рівня, наприклад SQL, в коректну й ефективну послідовність дій, представлену на мові низького рівня, що реалізовує операції реляційної алгебри [1]. При цьому найважливішим аспектом обробки запиту є його оптимізація. В результаті перетворення вихідного запиту може бути отримано безліч його еквівалентних варіантів.

Основна проблема пов'язана зі складністю виконання обчислювальних операцій та великою кількістю взаємозв'язків. Як правило, на практиці найчастіше обирається стратегія, найближча з можливих оптимальних рішень. Більшість таких стратегій передбачають еквівалентні перетворення виразів алгебри, на яких будується запит.

Мови маніпулювання даними високого рівня дозволяють формулювати складні запити для великої кількості відношень, виконання яких вимагає багато часу. Швидкість обробки таких запитів може бути значно збільшена, якщо перед їх виконанням модифікувати план проведення операцій алгебри. Мета таких модифікацій – отримати еквівалентний вираз, котрий потребує менше часу і пам'яті для його виконання.

Однією з важливих властивостей, що впливають на ефективність запиту, є по-

слідовність обчислення з'єднань в ланцюжку операцій. З'єднання – це вирішальна операція, оскільки необхідний час пропорційний добутку розмірів відношень, що сполучаються. Як критерії поліпшення структури запиту можна вибрати розмір проміжних результатів операцій з'єднання й порядок виконання комутуючих та асоціативних операцій.

При докладному розгляді операції з'єднання можна відзначити важливість способу її обчислення. Вибраний спосіб може визначатися або тим, як зберігаються відношення, або наявними додатковими структурами. При цьому саме обчислення може проводитися різними методами, наприклад, послідовно або індексним переглядом, з урахуванням або без урахування дублікатів.

Для вирішення питання ефективної реалізації запиту необхідно розглядати техніку управління файлами, взаємозв'язок між зверненнями до зовнішньої пам'яті й операцій реляційної алгебри, а також деякі технічні характеристики обчислювальної техніки.

Розглянуті в статті способи реалізації запиту використовують властивості операцій алгебри і теорії множин.

Подальші положення ґрунтуватимуться на властивостях операції з'єднання і правилах, що дозволяють здійснювати деякі перетворення алгебри [2].

Методи перебору даних. Алгоритм перебору повинен вибрати недорогий план виконання запиту на основі дослідження простору пошуку. Алгоритми перебору багатьох існуючих систем розраховані на вибір лише оптимального порядку лінійних з'єднань.

На практиці бажано мати такий алгоритм перебору, який міг би легко пристосовуватися до змін простору пошуку внаслідок додавання нових перетворень, нових фізичних операцій (наприклад, нових реалізацій з'єднань) і до змін методів оцінювання вартості. Сучасна архітектура оптимізації побудована на основі цієї парадигми і називається розширеними оптимізаторами [3].

Побудова розширеного оптимізатора – це складне завдання, оскільки необхідна не лише наявність алгоритму перебору, але й інфраструктура для розвитку техніки оптимізації [4]. Проте спільність архітектури має бути збалансована з потребою ефективного перебору. Таким чином, одним із важливих компонентів оптимізатора є наявність ефективного методу перебору, який визначає послідовність виконання операцій запиту.

Мінімізація суми проміжних з'єднань бази даних. Одним із критеріїв підвищення ефективності виконання запиту є зменшення числа кортежів в стосунках при багатократному з'єднанні. За умови, що операція з'єднання передбачає звернення до кожного кортежу відношень, що сполучаються, виникає завдання знайти таку послідовність, яка гарантує найменше сумарне число звернень до кортежів при послідовному з'єднанні.

Наприклад, база даних містить шість відношень і кожне відношення містить  $r(R) = \{100, 50, 80, 5, 20, 10\}$  кортежів відповідно. Тоді, при послідовному з'єднанні  $R = R_1 \triangleright \triangleleft R_2 \triangleright \triangleleft R_3 \triangleright \triangleleft R_4 \triangleright \triangleleft R_5 \triangleright \triangleleft R_6$  сумарне число кортежів буде відповідати виразу  $r = r_1 + r_2 + r_3 + r_4 + r_5$ ,

де  $r_1 = 100 * 50$ ,  $r_2 = r_1 * 80$ ,  $r_3 = r_2 * 5$ ,  $r_4 = r_3 * 20$ ,  $r_5 = r_4 * 10$ , а  $r(R) = 442405000$ .

Використовуючи властивості асоціати-

вності і комутативної операції з'єднання, змінюючи спільну послідовність з'єднання, можна зменшити (збільшити) значення  $r(R)$ .

Змінимо послідовність операцій. Поміняємо місцями  $R_1$  і  $R_6$ , тоді відношення  $R'$  буде отримано виразом  $R = R_6 \triangleright \triangleleft R_2 \triangleright \triangleleft R_3 \triangleright \triangleleft R_4 \triangleright \triangleleft R_5 \triangleright \triangleleft R_1$  і відповідно  $r(R) = 404240500$ . Таким чином, для обчислення відношення при такій послідовності буде потрібно на 38164500 дискових операцій менше.

Враховуючи, що кількість перестановок кінцева і відповідає  $n!$  (для даного прикладу – 6!), завжди можна отримати найменшу суму послідовного добутку.

Припустимо,  $X = \{x_1, x_2, \dots, x_n\} \in \mathbb{N}$ , де  $\mathbb{N}$  – безліч натуральних чисел, і результати операцій добутку утворюють безліч  $P = \{p_1, p_2, \dots, p_m\}$ , де  $p_1 = x_1 * x_2$ ,  $p_2 = p_1 * x_3$ , ...,  $p_m = p_{m-1} * x_n$ ,  $p_m = p_{m-1} * x_n$ . Необхідно знайти таку послідовність, щоб  $\sum_{i=1}^m p_i \rightarrow \min$ .

*Твердження 1.* Значення суми проміжних результатів ( $i = \overline{1, m}$ ) послідовного добутку елементів безлічі  $X = \{x_1, x_2, \dots, x_n\}$  буде найменшим, якщо значення  $x_1, x_2, \dots, x_n$  впорядковані за збільшенням, тобто  $x_1 < x_2 < \dots < x_n$ .

*Доведення.* Припустимо,  $x_i, x_j \in X$ ,  $i \neq j$ ,  $i, j = \overline{1, n}$  та  $p = x_i * x_j$ ; звідси добуток з черговим елементом безлічі  $X$  має вигляд  $p_i = p_{i-1} * x_{i+2}$  і буде найменшим, якщо  $k = \overline{1, n - (i+1)}$ . З іншого боку, значення  $p$  буде найменшим, якщо початкові значення будуть найменшими, тобто  $x_i, x_j = \min_k X$ ,  $k = \overline{1, n}$ .

Таким чином, при послідовному - перемножуванні деяких значень  $x_1, x_2, \dots, x_n \in X$  сума проміжних результатів буде найменшою, якщо дані значення впорядковані за збільшенням

$$x_1 < x_2 < \dots < x_n.$$

Із твердження про найменшу суму проміжних результатів послідовного добутку виходить, що якщо змінити порядок перемножування, то сума також зміниться.

Оскільки операція з'єднання є комутативною, послідовність її виконання не обов'язково має бути лінійною. Зокрема, запит для БД R може бути алгебрично представлений як

$$R = (R_1 \triangleright \triangleleft R_2) \triangleright \triangleleft (R_3 \triangleright \triangleleft R_4) \triangleright \triangleleft (R_5 \triangleright \triangleleft R_6).$$

При такій послідовності значення  $r(R)$  дорівнюватиме 400000000, що значно менше за попередні значення отриманих послідовним виконанням операції з'єднання. Надалі таку послідовність називатиме попарною.

Таким чином, за відповідне число кроків можна знайти таку послідовність пар добуток, при якій сума проміжних результатів буде мінімальною.

Відмінність попарного з'єднання від послідовного полягає в тому, що загальну суму проміжних результатів отримують за кілька незалежних кроків. Спочатку підсумовуються добутки довільних пар, після чого результати так само перемножуються парами в довільному порядку, поки не будуть перемножені всі можливі значення.

Теоретично не виключено варіант змішаного добутку, коли частина операцій виконується лінійно, а частина – попарно [1]. Формулу такого добутку можна відобразити, наприклад, як послідовність.

Використовуючи числові значення наведеного прикладу, порівняємо результати з'єднання різними способами. Як було відзначено вище, для лінійної впорядкованої за значеннями послідовності  $R = R_4 \triangleright \triangleleft R_6 \triangleright \triangleleft R_5 \triangleright \triangleleft R_2 \triangleright \triangleleft R_3 \triangleright \triangleleft R_1$  результат відповідатиме значенню, для довільного попарного з'єднання, наприклад,  $R = (R_1 \triangleright \triangleleft R_4) \triangleright \triangleleft (R_2 \triangleright \triangleleft R_6) \triangleright \triangleleft (R_3 \triangleright \triangleleft R_5)$ , результат буде таким же при довільному змішаному способі з'єднання, наприклад,  $R = (R_1 \triangleright \triangleleft R_2) \triangleright \triangleleft (R_3 \triangleright \triangleleft R_4) \triangleright \triangleleft (R_5 \triangleright \triangleleft R_6)$  результат відповідатиме  $r(R) = 2000000000$ .

Аналізуючи результати, отримані різні

способи в контексті даного завдання необхідно сформулювати і вирішити задачу пошуку мінімальної суми проміжних результатів на кожному кроці вибору пари з'єднання.

У загальному випадку задачу можна представити як підбір таких значень з деякої безлічі цілих чисел, добуток яких дає мінімальну суму проміжних результатів.

Стверджуватимемо, що мінімальна сума довільних пар добуток досягається при перемножуванні найменших і найбільших значень заданої множини. Для доведення цього факту розглянемо твердження.

*Твердження 2.* Хай задана множина  $X = \{x_1, x_2, \dots, x_n\} \in \mathbb{N}$ . Мінімум суми

попарних добуток  $\sum_{i,j=1}^n (x_i x_j) \rightarrow \min$  дося-

гається при  $x_i = \min\{x_1, x_2, \dots, x_n\}$  і  $x_j = \max\{x_1, x_2, \dots, x_n\}$ .

*Доведення.* Грунтуючись на тому, що кількість можливих пар множини X скінченна і залежить від перестановок елементів, яке відповідає  $n!$ , покажемо, що завжди можна знайти такі пари елементів, які визначають найменшу суму при перемножуванні.

Припустімо,  $x_i, x_j, x_k, x_l \in X$  та  $x_i < x_j < x_k < x_l$ . Розглянемо добуток пар  $p_1 = x_i * x_k$ ,  $p_2 = x_j * x_l$ ,  $p_3 = x_i * x_l$  та  $p_4 = x_j * x_k$ . Покажемо, що  $(p_1 + p_2) < (p_3 + p_4)$ .

Представимо елементи  $x_i, x_j, x_k, x_l$  як добуток різниці  $(x_j - x_i) * (x_l - x_k)$ . Очевидно, що  $x_j - x_i > 0$  і  $x_l - x_k > 0$ , таким чином  $(x_j - x_i) * (x_l - x_k) > 0$ . Розкривши дужки, отримаємо  $x_j x_l - x_j x_k - x_i x_l + x_i x_k > 0$ , згрупувавши від'ємні і додатні пари добуток, отримаємо  $x_j x_l + x_i x_k > x_j x_k + x_i x_l$ , тобто  $(p_2 + p_1) > (p_4 + p_3)$ .

Застосовуючи цю властивість для всіх

елементів множини  $X$ , завжди можна отримати найменшу суму проміжних результатів у середньому за  $2n!$  можливих перестановок. Таким чином, починаючи з будь-якої пари за скінченну кількість кроків можна знайти послідовність, відповідну мінімальній сумі добутоків елементів множини  $X$ .

З твердження про мінімальність суми попарних добутоків випливає, що якщо елементи впорядковані, то алгоритм складання пар добутоків повинен вибрати крайні елементи (тобто мінімальний і максимальний) і зсувати лічильник до середини множини, поки індекси зсуву справа і зліва не будуть рівними.

**Висновки.** У багатьох системах послідовність операцій з'єднання синтаксично обмежується для зменшення розмірів простору пошуку.

Як було показано вище, попарна послідовність з'єднань вимагає матеріалізації проміжних відношень. І хоча така послідовність приводить до дешевшого плану виконання запиту, вона значно збільшує витрати на перебір простору пошуку. З іншого боку, найбільшою є не вартість генерації синтаксичних послідовностей з'єднань, а вибір фізичних операцій та оцінювання вартості кожного можливого плану.

Для кожної таблиці необхідна статистична інформація включає число кортежів у потоці даних, оскільки цей параметр визначає вартість сканування даних,

з'єднань і відповідні вимоги до пам'яті. Слід розрізняти статистичні властивості виконання запиту і вартість плану. Статистична інформація – це логічна властивість, а вартість плану – це властивість фізична [4].

Отримані в статті результати визначають статистичні властивості, строго задаючи послідовність виконання операцій, при цьому мінімізуючи число кортежів у проміжних результатах, і загальне завдання побудови плану виконання запиту зводиться до оцінювання вартості за заданими критеріями.

З іншого боку, оптимізація означає набагато більше, ніж перетворення й еквівалентність запитів. Розробка ефективних перетворень запитів є складним завданням. Незважаючи дивлячись на багаторічні дослідження, суттєві проблеми залишаються відкритими.

Одним із важливих напрямків є той, у якому допускається генерація повних планів за умови доступності інформації часу виконання. Крім того, відкритою залишається проблема обліку інших важливих ресурсів (завантаженість центрального процесора, час доступу до сторінок пам'яті, методи доступу до даних тощо) при визначенні планів виконання запиту. Крім того, при використанні БД у контекстах мультимедіа і Web виникають цікаві задачі у зв'язку з нечіткими (неточними) запитамі [3].

### Література

1. Коннолли Т. Базы данных: проектирование, реализация и сопровождение. Теория и практика : учеб. пособ. / Коннолли Т., Бегг К., Страчан А. – 2-е изд. – [пер. с англ.]. – М. : Изд. дом «Вильямс», 2000. – 1120 с., ил.
2. Ульман Дж. Основы систем баз данных / [пер. с англ. М. Р. Когаловского и В. В. Когутовского ; под ред. М. Р. Когаловского]. – М. : Финансы и статистика, 1983. – 334 с., ил.
3. Чаудхари С. Методы оптимизации запросов в реляционных системах / С. Чаудхари // Системы управления базами данных – 1998. – № 3. – С. 22.
4. Кузнецов С. Методы оптимизации выполнения запросов в реляционных СУБД / Центр информационных технологий [Электронный ресурс]. – Режим доступа : [http://www.citforum.ru/database/articles/art\\_26.shtml.htm](http://www.citforum.ru/database/articles/art_26.shtml.htm)

*Надійшла до редколегії 18.06.2009*

### Анотації

У статті розглядаються принципи обробки реляційних запитів і способи їх реалізації. Як основна операція, що впливає на ефективність виконання запиту, виділена операція з'єднання. Запропоновано варіанти послідовності з'єднання відношень, які мінімізують кількість кортежів у проміжних результатах. Доведено оптимальність таких послідовностей.

В статье рассматриваются принципы обработки реляционных запросов и способы их реализации. В качестве основной операции, влияющей на эффективность выполнения запроса, выделена операция соединения. Предложены варианты последовательности соединения отношений минимизирующих количество кортежей в промежуточных результатах. Доказана оптимальность таких последовательностей.

Principles of processing of relational inquiries and methods of their realization are considered in the article. As a basic operation, influencing on efficiency of implementation of query is select the operation of compound. The variants of sequence of compound of relations of minimizing an amount corteges are offered in intermediate results. The optimumness of such sequences is proved

---

УДК 65.012.8.

**В. В. НОСОВ,**

*кандидат технічних наук, доцент  
професор кафедри інформаційної безпеки*

*Харківського національного університету внутрішніх справ*

---

### МІЖНАРОДНА СТАНДАРТИЗАЦІЯ З ІНФОРМАЦІЙНОЇ БЕЗПЕКИ У ФІНАНСОВИХ ІНФОРМАЦІЙНИХ СИСТЕМАХ

---

На сьогодні як у національному, так і в міжнародному нормативному полі стандартизації процесів з інформаційної безпеки в інформаційних системах, і зокрема в фінансових інформаційних системах, відсутня струнка класифікація наявних та потрібних перспективних стандартів. Тому **актуальним** є завдання системного аналізу опублікованих міжнародних стандартів з інформаційної безпеки у фінансових інформаційних системах та складання їх класифікації за визначеними критерієм, що надасть змогу в майбутньому виявити теоретичні потреби у нових стандартах.

У найбільш загальному вигляді будь-яка економіка складається з процесу виробництва товарів (продуктів, послуг) і комерційної діяльності, основною метою якої є отримання прибутку за допомогою реалізації виробленого товару.

Розвиток натурального товарного обміну (бартеру) привів до виділення з маси

товарів одного, який став відігравати роль загального еквіваленту, тобто грошей. Спочатку цим еквівалентом були дорогоцінні метали (в основному золото і срібло). Потім з'явилися карбовані з металу монети. На зміну металевим грошам прийшли «знаки» грошей – паперові гроші. Далі з'явилися інші платіжні засоби.

На цей час як платіжні інструменти можуть використовуватися: готівка (cash) (у формі металевих монет або паперових банкнот); платіжні доручення; платіжні вимоги; вимоги-доручення; векселі; чеки; банківські платіжні картки; інші дебетові й кредитові платіжні інструменти.

Основний грошовий обіг здійснюється в електронних платіжних системах. Структуру типової електронної платіжної системи показано на рис. 1 [1], де основними учасниками системи є:

– **банк-емітент**, який випускає платіжні засоби (наприклад, пластикові картки) і