

Moshel Mykola – Doctor of Technical Sciences, Professor, Professor of Information and Computer Systems, Chernihiv National University of Technology (95 Shevchenka Str., 14027 Chernihiv, Ukraine).

E-mail: mikorajj@mail.ru

Гриценко Микола Іванович – доктор фізико-математичних наук, професор, професор кафедри фізики та астрономії, Чернігівський національний педагогічний університет імені Т.Г. Шевченка (вул. Гетьмана Полуботка, 53, м. Чернігів, 14013, Україна).

Гриценко Николай Иванович – доктор физико-математических наук, профессор, профессор кафедры физики и астрономии, Черниговский национальный педагогический университет имени Т.Г. Шевченко (ул. Гетьмана Полуботка, 53, г. Чернигов, 14013, Украина).

Hrytsenko Mykola – Doctor of Physical and Mathematics Sciences, Professor, Professor of Physics and Astronomy, Chernihiv National Pedagogical University named after T. Shevchenko (53 Hetman Polubotka Str., 14013 Chernihiv, Ukraine).

E-mail: H.grit@yandex.ua

Рогоза Олександр Володимирович – кандидат фізико-математичних наук, професор, професор кафедри інформаційно-вимірювальних технологій, метрології та фізики, Чернігівський національний технологічний університет (вул. Шевченка, 95, м. Чернігів, 14027, Україна).

Рогоза Александр Владимирович – кандидат физико-математических наук, профессор, профессор кафедры информационно-измерительных технологий, метрологии и физики, Черниговский национальный технологический университет (ул. Шевченко, 95, г. Чернигов, 14027, Украина).

Rohoza Oleksandr – PhD in Physical and Mathematics Sciences, Professor, Professor of the Department of Information and Measuring Technologies, Metrology and Physics, Chernihiv National University of Technology (95 Shevchenka Str., 14027 Chernihiv, Ukraine).

E-mail: AVRogozal1010@gmail.com

Ковтун Анатолій Олексійович – кандидат фізико-математичних наук, доцент, доцент кафедри інформаційно-вимірювальних технологій, метрології та фізики, Чернігівський національний технологічний університет (вул. Шевченка, 95, м. Чернігів, 14027, Україна).

Ковтун Анатолий Алексеевич – кандидат физико-математических наук, доцент, доцент кафедры информационно-измерительных технологий, метрологии и физики, Черниговский национальный технологический университет (ул. Шевченко, 95, г. Чернигов, 14027, Украина).

Kovtun Anatolii – PhD in Physical and Mathematics Sciences, Associate Professor, Associate Professor of the Department of Information and Measuring Technologies, Metrology and Physics, Chernihiv National University of Technology (95 Shevchenka Str., 14027 Chernihiv, Ukraine).

E-mail: Kovtun38@mail.ru

Тепла Тетяна Мирославівна – асистент, асистент кафедри інформаційно-вимірювальних технологій, метрології та фізики, Чернігівський національний технологічний університет (вул. Шевченка, 95, м. Чернігів, 14027, Україна).

Теплая Татьяна Мирославовна – ассистент, ассистент кафедры информационно-измерительных технологий, метрологии и физики, Черниговский национальный технологический университет (ул. Шевченко, 95, г. Чернигов, 14027, Украина).

Tepla Tetiana – assistant, assistant of the Department of Information and Measuring Technologies, Metrology and Physics, Chernihiv National University of Technology (95 Shevchenka Str., 14027 Chernihiv, Ukraine).

E-mail: tanikoch@gmail.com

УДК 004.658

Вадим Мухин, Ярослав Корнага

АНАЛІЗ ЕФЕКТИВНОСТІ ОБРОБЛЕННЯ ЗАПИТІВ СЕРВЕРАМИ ГЕТЕРОГЕННИХ РОЗПОДІЛЕНИХ БАЗ ДАНИХ

Вадим Мухин, Ярослав Корнага

АНАЛИЗ ЭФФЕКТИВНОСТИ ОБРАБОТКИ ЗАПРОСОВ СЕРВЕРАМИ ГЕТЕРОГЕННЫХ РАСПРЕДЕЛЕННЫХ БАЗ ДАННЫХ

Vadym Mukhin, Yaroslav Kornaha

EFFICIENCY ANALYSIS OF REQUESTS' PROCESSING BY THE SERVER IN THE HETEROGENEOUS DISTRIBUTED DATABASES

Розподілені бази даних використовуються в різних сферах діяльності: освітній, промисловій, транспортній, бізнесовій, управлінській. Зокрема, такі бази даних широко застосовуються в закладах освіти для автоматизації навчального процесу та прискорення оброблення даних. Важливим елементом визначення швидкості роботи є середній час оброблення запитів у базах даних. На час оброблення впливає багато факторів, які пов'язані з обробленням даних на сервері та передачею по комп'ютерній мережі. На оброблення запитів на серверах баз даних витрачається значно більше часу ніж на передачу пакетів. Потрібно розробити формальну модель бази даних та порахувати середній час оброблення запитів. У роботі розглянуто оцінювання часу оброблення запитів серверами гетерогенних розподілених баз даних. Визначено основні простори та час, за який обробляються дані. Показано, на скільки відсотків змінюється час оброблення під час застосування моніторингу оброблення запитів та методів прискорення оброблення запитів.

© Мухин В. С., Корнага Я. І., 2016

Ключові слова: розподілені бази даних, індекси, оброблення запитів, модель даних.

Рис.: 2. Бібл.: 5.

Распределенные базы данных используются в различных сферах деятельности: образовательной, промышленной, транспортной, бизнесной, управленческой. В частности, такие базы данных широко применяют в учебных заведениях для автоматизации учебного процесса и ускорения обработки данных. Важным элементом определения скорости работы является оценка среднего времени обработки запросов в базах данных. На время обработки оказывает влияние много факторов, которые связаны с обработкой данных на сервере и передачей их по компьютерной сети. На обработку запросов на серверах баз данных тратится гораздо больше времени, чем на передачу пакетов. Необходимо разработать формальную модель базы данных и рассчитать среднее время обработки запросов. В работе рассмотрена оценка времени обработки запросов серверами гетерогенных распределенных баз данных. Определены основные пространства и временные интервалы, за которые обрабатываются данные. Показано, на сколько процентов изменяется время обработки при применении мониторинга обработки запросов и методов ускорения обработки запросов.

Ключевые слова: распределенные базы данных, индексы, обработка запросов, модель данных.

Рис.: 2. Библ.: 5.

The distributed databases are used in the various applications: education, industry, transport, business, management. In particular, such databases are widely used in universities to support the automation of the learning process and to accelerate the data processing. An important element for the data processing rate determining is the average estimation of the queries processing in databases. The time data processing is depended on the many factors that are associated with the data processing on the server and data transfer via a computer network. The queries' processing in the database servers is needed more time than on the packet. It should be developed a formal model of the database and calculated the average time of queries processing. In this work is described the evaluation of the query processing time on servers in heterogeneous distributed databases. There is defined the main areas and time needed for data processing. There is shown the changing in the time for the application processing when the queries processing monitoring and the methods for speed up the queries processing are used.

Key words: distributed database, indices, query processing, data model.

Fig.: 2. Bibl.: 5.

Постановка проблеми. Збільшення швидкості оброблення запитів є актуальним завданням для багатьох дослідників баз даних. Вона вирішується багатьма шляхами: оновленням серверного обладнання, оновленням комутаційного обладнання, оновленням програмного забезпечення та операційних систем. Системи управління базами даних постійно модернізуються і для цього використовують нові наукові досягнення.

Аналіз останніх досліджень. Одним з напрямів прискорення оброблення запитів є покращення механізмів використання класичних дерев пошуку, які модернізувалися постійно з бінарних дерев до В+-дерев. Постійні модифікації дерев пошуку дозволяють змінювати структуру методів оброблення запитів. Іншим засобом оброблення є оперативна пам'ять СУБД, тому що вона працює на кілька порядків швидше.

Мета статті. Метою статті є розроблення формальної моделі реляційної бази даних для оцінювання середнього часу оброблення запитів розподілених баз даних.

Формальна модель оброблення запитів

Для оцінювання часу оброблення запитів потрібно побудувати модель логічного рівня бази даних, яка у свою чергу будується на основі формальної моделі. Побудуємо формальну модель для реляційної клієнт-серверної системи управління базою даних. Вона складається (рис. 1) з таких елементів: простору користувачів, простору запитів, простору перевірки запитів, простору внутрішнього оброблення, простору даних та простору відповідей.

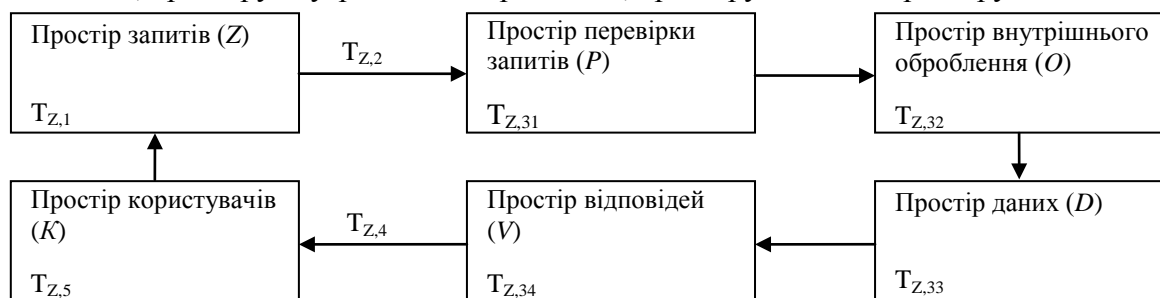


Рис. 1. Модель взаємодії на логічному рівні реляційної бази даних

Для внесення або отримання даних з сервера СУБД користувачі користуються спеціально написаними клієнтськими модулями, які призначені для підключення до бази даних

TECHNICAL SCIENCES AND TECHNOLOGIES

та формування запитів мовою SQL, мають необхідний інтерфейс для реалізації будь-яких задач та можливість простого написання запитів. Простір K складається з k_1, k_2, \dots, k_n користувачів, де n – кількість користувачів, кожен з яких має свої права доступу до сервера та можливість написання запитів і отримання відповідей. Для простоти будемо вважати, що всі користувачі мають однакові права та доступ на зміну всіх елементів бази даних.

Усі запити з клієнтського модуля користувача входять у простір запитів Z . Користувач k_i формує вектор запитів (z_1, z_2, \dots, z_m) :

z_1 – це перший запит користувача;

z_2 – це другий запит користувача;

...

z_m – це m -тий запит користувача,

де m – кількість запитів, які формуються користувачем.

Після формування вектора простору Z для перевірки коректності запитів з погляду мови SQL він передається у простір перевірки запитів P . Простір перевірки складається з небінарного вектора (p_1, p_2, \dots, p_m) , де p_i – відповідь перевірки запиту, яка набуває значення “true” у разі, якщо все коректно написано, та “false” – якщо є помилки.

Після проходження перевірки у просторі P запит, який побудований правильно, передається до простору внутрішнього оброблення O , що виконує функцію перетворення запиту у внутрішню мову, зрозумілу БД, тобто в цьому просторі проходять усі етапи роботи компілятора SQL. У ньому створюється вектор (o_1, o_2, \dots, o_m) , який складається з внутрішніх команд, що фактично є функціями від вектора запитів (z_1, z_2, \dots, z_m) , а саме $O(Z) = (o_1(z_1), o_2(z_2), \dots, o_m(z_m))$.

На першому етапі компілятор SQL здійснює синтаксичний аналіз операторів та перетворює їх у внутрішнє подання. На другому етапі відбувається перевірка відформатованих операторів на предмет цілісності посилань та на предмет виконання умов тригерів. Далі (третій етап) компілятор SQL ще раз переписує оператори, замінюючи посилання реальними іменами стовбців, та перетворює їх для подальшого оброблення оптимізатором. У процесі оптимізації виключаються надлишкові зв'язки, додаються неявні предикати та конвертуються деякі запити. Потім оптимізатор, використовуючи вартісні алгоритми, визначає метод, який найбільш ефективний для оброблення SQL – запиту (четвертий етап). Наприклад, оптимізатор знаходить найкращий порядок зв'язків і приймає рішення, чи виконання запиту є обчислювальною задачею, чи задачею, швидкість виконання якої обмежена швидкістю роботи пристроїв вводу-виводу. На п'ятому етапі SQL – запит зберігається в пам'яті, щоб надалі його можна було використовувати під час оптимізації подальших запитів, а відкомпільований і оптимізований SQL-код o_i передається механізму БД, який знаходиться у просторі даних D .

Простір даних D , до якого і належать усі дані (d_1, d_2, \dots, d_j) , де j – кількість даних, що обробляються та зберігаються в базі даних. Після виконання SQL-коду роботи з даними o_i ($i=1 \dots m$) з простору даних D у простір відповідей V передаються дані $d = (d_1, d_2, \dots, d_k)$, де k – кількість даних, які було знайдено під час проведення пошуку.

Відповідно до того, чи проводиться запис (модифікація), чи пошук даних, відповідь простору відповідей V формується по-різному. У першому випадку передається вектор $v_i = (v_1, v_2, \dots, v_g)$, де g – кількість запитів запису (модифікації), про успішний з кількістю полів, які були модифіковані, чи неуспішний запис (модифікацію) даних з відповідною причиною, чому не відбувся запис (модифікація). У другому випадку – вектора $v_i = (v_1(d), v_2(d), \dots, v_j(d))$, де j – кількість запитів на пошук даних, а d – вектор даних з простору даних D . У разі, якщо дані не знайдені, то значення $v_i(d)$ буде пустим.

$$V_i = Z_i \cap (P_j \cup O_j \cup D_j),$$

$$V = \sum_{i=1}^n Z_i \cap \left(\sum_{j=1}^m P_j \cup \sum_{j=1}^m O_j \cup \sum_{j=1}^m D_j \right).$$

З простору відповідей V сформована відповідь v_i потрапляє у простір користувачів K , в якому вони передивляються відповіді та за необхідності готують нові запити.

Оцінювання середнього часу оброблення запитів

Під час проведення дослідження роботи розподілених баз даних була розроблена методика, яка дозволяє оцінювати ефективність оброблення запитів серверами розподіленої системи згідно з формальною моделлю, сформованої вище. Вона полягає в оцінюванні часу, за який проводяться операції з базою даних.

Час виконання запиту потрібно розбити на кілька етапів, які відповідають різним просторам оброблення даних:

$$T_{Z_i} = T_{Z_i,1} + T_{Z_i,2} + T_{Z_i,3} + T_{Z_i,4} + T_{Z_i,5},$$

де $T_{Z_i,1}$ – час підготовки запиту для передачі на сервер показує час створення запитів простору Z ;

$T_{Z_i,2}$ – час, за який запит потрапляє на сервер по комп'ютерній мережі;

$T_{Z_i,3}$ – час, за який сервер обробляє запит та показує час оброблення запитів у просторах: P, O, V, D ;

$T_{Z_i,4}$ – час, за який відповідь на запит повертається до користувача по комп'ютерній мережі;

$T_{Z_i,5}$ – час, за який клієнт з простору K обробляє відповідь на запит.

Отже, час на оброблення запиту сервером поділяється на:

$$T_{Z_i,3} = T_{Z_i,31} + T_{Z_i,32} + T_{Z_i,33} + T_{Z_i,34},$$

$T_{Z_i,31}$ – час підготовки запиту на сервері та оцінювання сенсором подій простору перевірки запитів P ;

$T_{Z_i,32}$ – час, за який проводиться операції з даними простору даних D ;

$T_{Z_i,33}$ – час, за який проводиться операції з індексами простору внутрішнього оброблення O ;

$T_{Z_i,34}$ – час підготовки відповіді на запит на сервері простору підготовки відповідей V .

Індексний простір (час $T_{Z_i,33}$) у базах даних використовує для пошуку даних В+ дерева. Для класичних дерев пошуку $T_{Z_i,33} = m_B(T_D + T_B)$, де T_D – час зчитування вузла з диску, T_B – час оброблення вузла дерева в оперативній пам'яті, а m_B – висота дерева пошуку. Під час модифікації дерев, в напрямку збільшення наповненості вузла, до КМФ-дерев змінюється тільки висота дерева, тобто $T_{Z_i,33} = m_{КМФ}(T_D + T_B)$. Крім модифікації наповненості вузлів дерева, доцільно проводити модифікацію листових сторінок (додавання додаткового параметра для використання індексу по кількох таблицях), тобто використовувати метод об'єднання індексів, для якого час пошуку по дереву рівний $T_{Z_i,33} = \sum_{o=1}^{m_j} (m_O(T_D + T_B) + d(T_D + T_B)) + \sum_{s=1}^{q_i} m_s(T_D + T_B)$, де d – кількість вузлів, які потрібно додатково зчитати з жорсткого диску.

Для оброблення даних у базах даних $T_{Z_i,32}$ буде різною: для класичного методу $T_{Z_i,32} = n(T_{3T} + T_{3P})$, тобто під час кожного запису витрачається час на сам запис інформації T_{3T} та час на перебудову індексів T_{3P} і n – кількість записів, а під час використання методу тимчасових таблиць для запису $T_{Z_i,32} = n T_{3T} + T_{3P} + T_{3D}$, тобто замість вико-

нання n перебудов індексів ми перебудовуємо їх тільки один раз, а час $T_{ПД}$ – час копіювання даних з тимчасової таблиці в основну.

Відповідно до вищесказаного середній час оброблення запитів у базах даних буде різним у разі застосування даних модифікацій.

Для класичного варіанта середній час оброблення запитів рівний:

$$T_K = \sum_{i=1}^n (T_{Z_{i,1}} + T_{Z_{i,2}} + T_{Z_{i,31}} + \sum_{z=1}^{k_i} q_z (T_{3T} + T_{ПП}) + \sum_{s=1}^{q_i} m_{s(B)} (T_D + T_B) + T_{Z_{i,34}} + T_{Z_{i,4}} + T_{Z_{i,5}}).$$

Під час застосування модифікованих дерев пошуку КМФ-дерев:

$$T_{КМФ} = \sum_{i=1}^n (T_{Z_{i,1}} + T_{Z_{i,2}} + T_{Z_{i,31}} + \sum_{z=1}^{k_i} q_z (T_{3T} + T_{ПП}) + \sum_{s=1}^{q_i} m_{s(КМФ)} (T_D + T_B) + T_{Z_{i,34}} + T_{Z_{i,4}} + T_{Z_{i,5}}).$$

Відповідно під час модифікації листових вузлів дерев:

$$T_{МОІ} = \sum_{i=1}^n (T_{Z_{i,1}} + T_{Z_{i,2}} + T_{Z_{i,31}} + \sum_{z=1}^{k_i} q_z (T_{3T} + T_{ПП}) + (\sum_{o=1}^{m_j} (m_o (T_D + T_B) + d(T_D + T_B))) + \sum_{s=1}^{q_i} m_s (T_D + T_B)) + T_{Z_{i,34}} + T_{Z_{i,4}} + T_{Z_{i,5}}).$$

Під час застосування тимчасових таблиць середній час оброблення запитів рівний:

$$T_{МТТ} = \sum_{i=1}^n (T_{Z_{i,1}} + T_{Z_{i,2}} + T_{Z_{i,31}} + \sum_{z=1}^{k_i} (q_z T_{3T} + T_{ПП} + T_{ПД}) + \sum_{s=1}^{q_i} m_{s(B)} (T_D + T_B) + T_{Z_{i,34}} + T_{Z_{i,4}} + T_{Z_{i,5}}).$$

І під час застосування всіх методів прискорення оброблення запитів:

$$T_{c_all} = \sum_{i=1}^n (T_{Z_{i,1}} + T_{Z_{i,2}} + T_{Z_{i,31}} + \sum_{z=1}^{k_i} (q_z T_{3T} + T_{ПП} + T_{ПД}) + (\sum_{o=1}^{m_j} (m_o (T_D + T_B) + d(T_D + T_B))) + \sum_{s=1}^{q_i} m_s (T_D + T_B)) + T_{Z_{i,34}} + T_{Z_{i,4}} + T_{Z_{i,5}}) / n.$$

За допомогою розробленої методики оцінювання часу оброблення запитів було проведено експеримент визначення середньої швидкості під час застосування різних методів прискорення оброблення запитів. Для експерименту було взято $n=1000$, що дозволило порахувати середній час оброблення з похибкою (середньоквадратичне відхилення) 0,91 % (рис. 2).

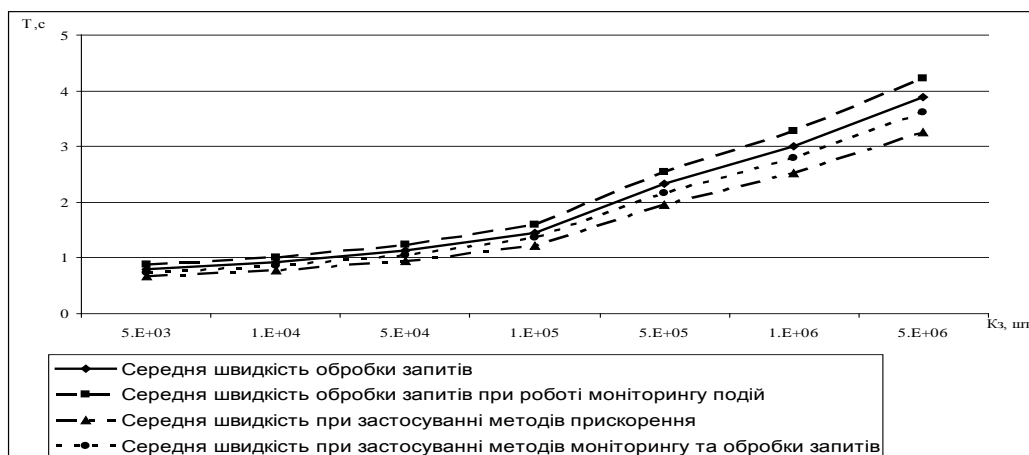


Рис. 2. Оцінки сумарного часу обробки даних подій моніторингу з застосуванням методів прискорення

Згідно з вищенаведеними дослідженнями представлена необхідність підтримки реалізації прискореного оброблення даних під час застосування моніторингу. Швидкість оброблення запитів у розподілених базах даних збільшується на 9 %.

Висновки і пропозиції.

Для проведення обрахунку середнього часу оброблення запитів гетерогенними розподіленими базами даних було побудовано формальну модель даних та виведено формули для розрахунку. Достовірність отриманих теоретичних положень підтверджена проведенням експериментальних досліджень, які показали ефективність застосування методів прискорення оброблення запитів під час виконання моніторингу в динамічному режимі. Вивільнені ресурси бази даних можна використовувати для підключення більшої кількості користувачів та створення додаткового навантаження на СУБД.

Список використаних джерел

1. Кирилов В. В. Введение в реляционные базы данных / В. В. Кирилов, Г. Ю. Громов. – СПб. : БХВ-Петербург, 2009. – 578 с.
2. Корнага Я. І. Методи підвищення швидкості запису та пошуку даних у базах даних / Я. І. Корнага // Вісник Чернігівського державного технологічного університету. Серія «Технічні науки». – Чернігів, 2012. – № 4 (61). – С. 166–172.
3. Корнага Я. І. Порівняльні оцінки застосування методів підвищення швидкості пошуку та запису даних в базах даних / Я. І. Корнага // Адаптивні системи автоматичного управління. – 2013. – № 1 (22). – С. 37–44.
4. Кузин А. В. Базы данных / А. В. Кузин, С. В. Ливонисова. – М. : Академия, 2008. – 489 с.
5. Мухін В. Є. Підвищення ефективності механізмів пошуку в базах даних на основі К-дерев / В. Є. Мухін, Я. І. Корнага, Л. Снегірев // Вісник Національного університету «Львівська політехніка». Серія «Комп'ютерні науки та інформаційні технології». – Львів, 2012. – № 744. – С. 53–57.

References

1. Kirilov, V. V., Gromov, G. Yu. (2009). Vvedenie v reliatsionnye bazy danykh [Introduction to relational database]. Saint-Petersburg: BKhV-Peterburg, p. 578 (in Russian).
2. Kornaha, Ya. I. (2012). Metody pidvyshchennia shvydkosti zapysu ta poshuku danykh u bazakh danykh. *Visnyk Chernihiv State Technological University. Series "Technical sciences"*, no. 4 (61), pp. 166–172 (in Ukrainian).
3. Kornaha, Ya. I. (2012). Porivnialni otsinky zastosuvannia metodiv pidvyshchennia shvydkosti poshuku ta zapysu danykh v bazakh danykh. *Adaptyvni systemy avtomatychnoho upravlinnia – Adaptive systems of automatic control*, no. 1, pp. 37–44 (in Ukrainian).
4. Kuzin, A. V., Livonisova, S. V. (2008). *Bazy danykh [Databases]*. Moscow: Akademiia, p. 489 (in Russian).
5. Mukhin, V. Ye., Kornaha, Ya. I., Sniehiriev, L. (2012). Pidvyshchennia efektyvnosti mekhanizmv poshuku v bazakh danykh na osnovi K-derev [Improving the efficiency of search engines in the databases on the basis of K-trees]. *Visnyk Natsionalnoho universytetu «Lvivska politehnika» – Visnyk of Lviv Polytechnic National University. Series "Computer science and information technology"*. Lviv, no. 744, pp. 53–57 (in Ukrainian).

Мухін Вадим Євгенійович – доктор технічних наук, доцент, професор кафедри обчислювальної техніки, Національний технічний університет України «Київський політехнічний інститут» (просп. Перемоги, 37, м. Київ, 03056, Україна).

Мухін Вадим Євгенійович – доктор технических наук, доцент, профессор кафедры вычислительной техники, Национальный технический университет Украины «Киевский политехнический институт» (просп. Победы, 37, г. Киев, 03056, Украина).

Mukhin Vadym – Doctor of Technical Sciences, Professor of computer systems department, National Technical University of Ukraine “Kyiv Polytechnic Institute” (37 Peremohy Av., 03056 Kyiv, Ukraine).

E-mail: v_mukhin@mail.ru

Корнага Ярослав Ігорович – кандидат технічних наук, доцент кафедри технічної кібернетики, Національний технічний університет України «Київський політехнічний інститут» (просп. Перемоги, 37, м. Київ, 03056, Україна).

Корнага Ярослав Ігорович – кандидат технических наук, доцент кафедры технической кибернетики, Национальный технический университет Украины «Киевский политехнический институт» (просп. Победы, 37, г. Киев, 03056, Украина).

Kornaha Yaroslav – PhD in Technical Sciences, Associate Professor, Department of Technical Cybernetics, National Technical University of Ukraine “Kyiv Polytechnic Institute” (37 Peremohy Av., 03056 Kyiv, Ukraine).

E-mail: slovyan_k@ukr.net