

ДОСЛІДЖЕННЯ ЕФЕКТИВНОСТІ КЕШУВАННЯ ПРИ ВИКОРИСТАННІ MS SQL SERVER, REDIS ТА MEMCACHED

Семко Д.

Науковий керівник – к.т.н., доцент Мазурова О. О.

Харківський національний університет радіоелектроніки, каф. ПІ

м.Харків, Україна

e-mail: denys.semko@nure.ua

Each modern system wants to provide fascinating experience for its users. There are plenty of factors that can make the system become one of the best, such as user-friendly UI, modern features, customer support, etc. However, one of the most important things you want to bring to your customers is a speed response. Therefore, the product must be able to respond as soon as possible, be scalable and flexible. All of that can be done with the help of caching on different levels. This paper describes the efficiency of caching mechanisms that are built-in in the modern databases. The research of caching mechanisms was carried out on the basis of databases in the field of e-commerce based on the use of relational DBMS MSSQL and NoSQL systems – Redis and Memcached.

Щохвилини користувачі глобальної мережі Інтернет здійснюють понад 9 мільйонів запитів у пошуковій системі Google. При цьому час отримання відповідей сягає декількох секунд чи навіть менше. Більш унікальні запити вимагають більше часу для пошуку, проте кожен запит містить інформацію, яка вже збережена в пам'яті пошукової системи, а отже може бути виконаний на рівні кешування.

Кешування – це процес тимчасового зберігання даних у кеш-пам'яті з метою збільшення продуктивності програм та систем у цілому [1]. Кешування може здійснюватись на різних рівнях, це і на рівні веб-браузера, чи веб-серверу, CDN та на рівні вихідного серверу системи. Кожен з рівнів так чи інакше покращують спілкування користувача з системою задля пришвидшення відповіді на кожний з запитів.

Багато існуючих СКБД реалізують свій механізм кешування та передбачають власну вбудовану логіку, яка є доволі унікальною та прискорює швидкість взаємодії з базою даних. Отже постає питання, яка зі СКБД найкраще забезпечує роботу з великим об'ємом даних за допомогою власних механізмів, в тому числі у рамках кешування.

Доцільність проведення дослідження методів кешування в різних СКБД зумовлена існуванням ряду проблеми, що можуть бути вирішені в результаті дослідження, а саме:

– наявність запитів, що потребують значних обчислювальних або мережеских ресурсів для виконання; для таких випадків кешування може значно підвищити продуктивність, особливо при запитах з великим обсягом даних [2];

– висока навантаженість на базу даних, яка є доволі популярною під час користування системи; велика кількість користувачів, одночасні запити – усе це є проблемою та кешування може забезпечити кращу та миттєву відповідь користувачам;

– наявність системи з обмеженими ресурсами; якщо система має обмежені обчислювальні ресурси на сервері бази даних, то кешування може допомогти ефективно використовувати ці ресурси та запобігти перевантаженню сервера.

Як бачимо, існує чимало проблем з якими розробник програмного забезпечення може зіштовхнутись, особливо якщо система передбачає масштабованість та популярність серед користувачів [3]. Отже для отримання рішення цих проблем була поставлена задача виконати дослідження механізмів кешування у реляційних та нереляційних базах даних, а саме:

– провести аналіз та обрати СКБД для подальшого дослідження ефективності кешування;

– провести аналіз реалізації механізму кешування в обраних СКБД;

– спланувати експериментальне дослідження, що включає в себе вибір, аналіз та моделювання предметної галузі для розробки БД для дослідження, розробку критеріїв оцінки ефективності, можливості конфігурації даних в певному форматі одночасно для усіх СКБД та розробку стресових ситуацій для перевірки;

– провести експериментальне дослідження ефективності кешування на операціях читання даних;

– оцінити якість результатів, сформулювати рекомендації стосовно використання СКБД.

Було проведено аналіз та обрано наступні СКБД для проведення дослідження: серед реляційних – це MSSQL, а серед NoSQL систем – це Redis та Memcached [4].

Був проведений аналіз реалізації механізмів кешування у кожній з СКБД та визначено можливості отримання даних про вбудовані процеси.

Проведено планування експериментальних досліджень, а саме:

– в якості предметної області для побудови бази даних, на якій буде проводитися дослідження, обрано тематику електронної комерції, яка підходить для проведення дослідження через наявність великого об'єму даних, високого навантаження та складної структури;

– розроблена база даних, яка включає в себе такі сутності, як «User», «Product», «Category», «Basket», «Order», «Product_Basket» та «Product_Order»;

– в якості метрик для проведення замірів було обрано коефіцієнти попадання та промахів у кеш-пам'ять, час відповіді на запит, використання ресурсів та розмір кешу;

– заплановано також умови проведення експериментів, а саме забезпечити окреме тестове середовище, яке не матиме зовнішніх чинників впливу на швидкість роботи БД.

Були розроблені програмні рішення для проведення експериментів із механізмами кешування обраних СКБД, а саме програмна система, яка дозволяє:

– завантажити дані для обраної предметної області для усіх СКБД одночасно;

– провести відповідні експерименти та отримати результати;

– отримати інфографіку та рекомендації щодо роботи з кожною СКБД.

Під час проведення експериментів очікується отримати результати, які дозволять зробити висновки, щодо найбільш ефективною СКБД для використання в рамках кешування.

Отримані в результаті рекомендації будуть корисні для будь-якого розробника або команди, які планують створити масштабовану та ефективну програмну систему, яка у кінцевому результаті оброблятиме великий об'єм даних. Отримані рекомендації будуть важливими також в процесі проектування гетерогенних баз даних, оптимізації роботи бази даних та покращенні її продуктивності [5].

Результати дослідження дозволять спростити та прискорити такі процеси в системах баз даних, як доступ до інформації, виконання запитів та обробка даних в різних сценаріях використання СКБД.

Список використаних джерел:

1. Blokdyk G. Caching A Complete Guide. The Art of Service – Caching Publishing, 2020 – 312 p.

2. Barker T. Intelligent Caching. O'Reilly Media, Inc., 2017 – 215 p.

3. Kleppmann M. Designing Data-Intensive Applications: The Big Ideas Behind Reliable, Scalable, and Maintainable Systems. O'Reilly Media, Inc., 2017 – 611 p.

4. Kuzochkina, A., Shirokopetleva, M., Dudar, Z. Analyzing and Comparison of NoSQL DBMS, International Scientific-Practical Conference on Problems of Infocommunications Science and Technology, PIC S and T 2018. Proceedings. 2019. P. 560–564. DOI 10.1109/INFOCOMMST.2018.8632133.

5. Fowler M. NoSQL Distilled: A Brief Guide to the Emerging World of Polyglot Persistence. Addison-Wesley Professional, 2012 – 192 p.