УДК 004.042

М. О. Шевчук

Рекомендації по покращенню роботи з даними з використанням MySQL і MongoDB

Вінницький національний технічний університет

Анотація

За результатами проведеного аналізу була описана необхідність підвищення продуктивності систем для покращення якості життя користувачів, а також наведені методи як організувати дані в залежності від специфіки роботи бази даних.

Ключові слова: реляційна база даних, документо-орієнтована база даних, SQL, NoSQL, MySQL, MongoDB, мікросервісна архітектура, оптимізація.

Abstract

According to the results of the analysis, the need to increase the productivity of systems to improve the quality of life of users was described, as well as methods for organizing data depending on the specifics of the database.

Keywords: relational database, document-oriented database, SQL, NoSQL, MySQL, MongoDB, microservice architecture, optimization.

У сучасному світі, де час є одним із найважливіших ресурсів, люди потребують високого рівню комфорту в користуванні різноманітними цифровими системами. Зазвичай ніхто не хоче довго чекати, тому для забезпечення цього комфорту потрібно створювати швидкі, оптимізовані програми. Для цього було створено велику кількість різноманітних мов програмування, які ефективні кожна в своїх сферах інформаційного простору.

Однак не тільки вірний вибір мови програмування сприяє покращенню швидкодії системи, також важливі технології, бібліотеки, фреймворки, бази даних та архітектура. Кожна з цих речей в значній мірі впливає на кінцевий результат, тому до їх вибору потрібно віднестися з максимальною відповідальністю.

Також важливим фактором в комфорті користування системою є відмовостійкість та безпека. В багатьох проектах де є клієнт та сервер використовується база даних. Як правило використовують реляційні бази даних але з появою NoSQL з'явились нові способи організації інформації. У кожної з цих методик є свої плюси та мінуси. Тому з метою покращення продуктивності обробки даних було б доцільно об'єднати і виділити переваги кожної з баз даних.

Для коректної та продуктивної роботи баз доцільно було б використати підходи мікросервісної архітектури і винести їх в окремі сервіси, що підвищить відмовостійкість і безпеку системи. Дані між мікросервісами передаються по протоколу HTTP в форматі JSON. Це є невеликим мінусом в продуктивності, однак це дозволяє зробити реалізацію з двома базами даних, а також ORM для MongoDB швидко серіалізує та десеріалізує JSON в BJSON, оскільки ці формати досить схожі.[1]

Також важливо розуміти, що всі запити до бази даних проводяться да допомогою ORM і в реаліях їх реалізації, тому деякі з них можуть виконувати запити повільніше чим чисті запити до бази даних. Також є проблема серіалізації та десеріалізації даних.

Основною ідеєю є розподіл даних та операцій над ними по різним базам даних, я приведу список рекомендацій по яким це потрібно робити.

Використовувати MySQL потрібно коли:

1) Об'єкти добре зв'язуються за відношеннями, причому звертання до цих об'єктів відбувається відносно нечасто. Тоді можна використати lazy ініціалізацію і не робити постійно операцію JOIN. Це дозволить скоротити час мапінгу даних в об'єкт, а також кількість часу яку витрачає база даних для запиту.
2) При необхідності часто додавати елементи відношення багато-до-багатьох та багато-до-одного. В цьому випадку потрібно зробити INSERT в таблицю, без необхідності шукати та вивантажувати весь об'єкт з бази даних так як це потрібно робити в MongoDB.

3) При необхідності редагувати елемент вкладеної таблиці, зв’язаної відношенням.
4) При необхідності робити різноманітні складні запити по кільком таблицям. MySQL це робить краще.
5) При запитах на один або кілька невеликих об’єктів без відношень і складних структур.[2]

Використовувати MongoDB потрібно коли:

1) Потрібно зберігати складні незмінні структури даних. Тобто при реалізації архіву.
2) Коли потрібно часто звертатися до складних структур даних в яких завжди потрібно отримувати всі дані. Аналог eager JOIN.
3) При необхідності кардинально редагувати весь документ з всіма його вкладеними об’єктами.
4) При необхідності зберігати списки та мапи.[3]

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Nadareishvili I., Mitra R., McLarty, M., Amundsen M., Microservice Architecture: Aligning Principles, Practices, and Culture. — O'Reilly Media 2016. — 146 с.
2. Garcia-Molina H., Ullman J. D., Widom J., Database Systems The Complete Book Second Edition. — Department Of Computer Science Stanford University, 2009. — 1240 с.
3. Banker K., MongoDB in Action. — Manning Publications 2011.— 312 с.

Шевчук Максим Олегович — студент групи 1КІ-20м, кафедри комп'ютерної інженерії ВНТУ, Вінницький національний технічний університет, м. Вінниця, e-mail: m.shevchuk.o@gmail.com

Науковий керівник: ***Ткаченко Олександр Миколайович*** — к.т.н., доцент кафедри обчислювальної техніки, Вінницький національний технічний університет, м. Вінниця, e-mail: alextk1960@gmail.com

Shevchuk Maksym Olehovych — student of the Computer Techniques Chair, Vinnytsia National Technical University, Vinnytsia, e-mail: m.shevchuk.o@gmail.com

Research advisor: ***Tkachenko Oleksandr Mykolaiovych*** — Ph.D., Associate Professor of Computer Engineering, Vinnytsia National Technical University, Vinnytsia, e-mail: alextk1960@gmail.com