

**СТРАТЕГІЇ ПОКРАЩЕННЯ ПРОДУКТИВНОСТІ РЕЛЯЦІЙНИХ СИСТЕМ  
КЕРУВАННЯ БАЗАМИ ДАНИХ****STRATEGIES FOR IMPROVING PERFORMANCE OF RELATIONAL DATABASE  
MANAGEMENT SYSTEMS**

У сучасних інформаційних системах продуктивність баз даних є критичною, адже затримки у виконанні запитів, надмірне споживання ресурсів та ускладнення масштабування можуть впливати на швидкість обробки інформації, зручність роботи користувачів та загальну продуктивність бізнесу. З огляду на це, виникає потреба у впровадженні рішень, які дозволять не лише покращити швидкість і надійність доступу до даних, але й забезпечити гнучкість та стійкість баз даних в умовах підвищених навантажень.

Індексація є основною технікою для прискорення пошуку та вибірки даних. Вона включає одиночні індекси для прискорення виконання запитів, що здійснюють фільтрацію чи сортування на основі одного стовпця, складені індекси, що забезпечують оптимізацію складніших запитів, а також спеціалізовані структури, такі як GIN (Generalized Inverted Index) та GiST (Generalized Search Tree) індекси, які забезпечують ефективну роботу з нетиповими даними або специфічними типами пошуку [1].

Грамотна побудова SQL-запитів, а також використання відповідних інструментів для аналізу та оптимізації коду також дозволяють скоротити час виконання складних запитів і зменшити навантаження на базу даних. Правильне налаштування кешу дозволяє суттєво знизити затрати на обчислення і скоротити час виконання запитів, зменшуючи навантаження на сервер бази даних. Кешування є ефективним способом зменшення кількості звернень до бази даних при однотипних запитах, забезпечуючи швидкий доступ до часто використовуваних даних.

Нормалізація бази даних допомагає уникнути надлишковості даних та оптимізувати структуру. Проте, в деяких випадках, денормалізація може бути також доцільною, особливо коли потрібно мінімізувати кількість JOIN-операцій у запитах для прискорення їх виконання [2].

Реплікація та фрагментування (шардінг) допомагають розподіляти дані між кількома вузлами або серверами, забезпечуючи балансування навантаження та зменшення часу доступу до даних [3]. Апаратні ресурси, зокрема об'єм пам'яті, потужність процесора, тип дискової підсистеми, також грають важливу роль в оптимізації, а для систем з інтенсивним використанням дискової підсистеми, використання різних рівнів RAID може істотно підвищити продуктивність баз даних.

Отже, впровадження вищенаведених стратегій покращення продуктивності реляційних баз даних дозволяє суттєво підвищити ефективність роботи систем.

**Література**

1. Recommendations for optimizing data. Microsoft Learn. URL: <https://learn.microsoft.com/en-us/azure/well-architected/performance-efficiency/optimize-data-performance>.
2. Udasi A. Denormalized Data Explained. Zenduty Blog. URL: <https://www.zenduty.com/blog/data-denormalization>.
3. Learn When to Use Database Replica and Database Sharding for Systems Design. Java Challengers. URL: <https://javachallengers.com/database-replica-and-sharding>.