

*Михайляк М. О., здобувачка вищої освіти,  
Веселовська Н. Р., д-р техн. наук, професор,  
професор кафедри інформаційних технологій,  
Донецький національний університет імені Василя Стуса*

## **МЕТОДИ ПІДВИЩЕННЯ ПРОДУКТИВНОСТІ ТА ОПТИМІЗАЦІЇ ЗБЕРІГАННЯ У СУЧАСНИХ СИСТЕМАХ БАЗ ДАНИХ**

*Анотація. У роботі розглянуто основні методи підвищення продуктивності та оптимізації зберігання даних у сучасних СУБД. Проаналізовано індексацію, нормалізацію, оптимізацію запитів, кешування, шардінг, реплікацію та використання in-тету технологій. Показано вплив вибору архітектурних рішень на ефективність роботи системи.*

*Ключові слова: СУБД, оптимізація, індексація, кешування, реплікація, продуктивність.*

Сучасні системи управління базами даних відіграють важливу роль у забезпеченні зберігання та доступу до інформації для різних програмних систем. Зі зростанням обсягів даних та збільшенням кількості користувачів виникає потреба у забезпеченні високої продуктивності та оптимального використання ресурсів. Для цього в СУБД застосовуються різні методи підвищення швидкодії та оптимізації зберігання [1]. До основних методів належать:

1. Індексація даних. Індекси дають змогу прискорити пошук і вибірку даних, зменшуючи кількість операцій читання з диска. Найпоширенішими типами індексів є B-Tree, Hash-індекси та бітові індекси. Проте надмірна кількість індексів може уповільнювати операції вставки та оновлення.

2. Нормалізація та денормалізація. Нормалізація усуває дублювання даних і зменшує ймовірність їх неузгодженості. Денормалізація, навпаки, передбачає дублювання даних для прискорення запитів читання. Вибір залежить від типу навантаження: транзакційного чи аналітичного.

3. Оптимізація запитів. Ефективність виконання SQL-запитів значною мірою впливає на загальну продуктивність. Оптимізація включає аналіз планів виконання запитів, правильне використання індексів, уникнення зайвих JOIN та повних сканувань таблиць.

4. Кешування. Часто використовувані дані можуть зберігатися у кеші СУБД або в зовнішніх сервісах (Redis, Memcached). Це дає змогу зменшити кількість звернень до основної бази даних і пришвидшити роботу системи [2].

5. Шардінг та реплікація. Шардінг передбачає розподіл даних між декількома серверами з метою горизонтального масштабування. Реплікація створює копії баз даних для підвищення відмовостійкості та збільшення швидкості читання (master-slave, multi-master підходи).

6. Колоночні сховища. Для аналітичних систем застосовують бази даних, де дані зберігаються по стовпцях. Це підвищує швидкість виконання складних аналітичних запитів та дає змогу ефективно стискати інформацію.

7. Стиснення та архівація даних. Стиснення зменшує обсяг, який займають дані, і може прискорювати вибірку завдяки зменшенню обсягу читання. Архіва-

ція старих або рідко використовуваних даних у менш затратні сховища знижує навантаження на основну базу.

8. In-memory технології. Бази даних, що зберігають інформацію в оперативній пам'яті, наприклад, Redis чи SAP HANA, забезпечують найвищу швидкість доступу. Вони використовуються у системах з високими вимогами до затримки (реального часу).

У висновку можна сказати, що підвищення продуктивності та оптимізації зберігання в СУБД є комплексним процесом, який включає правильне структурне проєктування бази даних, оптимізацію SQL-запитів, використання індексів, кешування, реплікацію та інші методи масштабування. Вибір конкретного підходу залежить від типу навантаження, обсягу даних та вимог системи до швидкодії і надійності. Правильне поєднання цих методів дає змогу забезпечити ефективну роботу бази даних та стабільне функціонування всієї інформаційної системи.

#### Список використаних джерел

1. Оптимізація та продуктивність бази даних. *Hostragons*. 06.04.2025. URL: <https://surl.li/kamjtr> (дата звернення: 29.10.2025).
2. Рекомендації щодо оптимізації продуктивності даних. *Learn. Microsoft*. URL: <https://surl.li/dkvdzr> (дата звернення: 29.10.2025).