

УДК 681.5.004.65

**Швець М.Ю.**

Національний технічний університет України  
«Київський політехнічний інститут імені Ігоря Сікорського»

**Заруба Д.С.**

Національний технічний університет України  
«Київський політехнічний інститут імені Ігоря Сікорського»

**Хохлов Ю.В.**

Національний технічний університет України  
«Київський політехнічний інститут імені Ігоря Сікорського»

## ПОРІВНЯННЯ SQL ТА NOSQL БАЗ ДАНИХ

*За останні десятиліття значно збільшилось використання додатків, що генерують нові потоки даних, які раніше були немислимі як за різноманітністю, так і за кількістю. Але ці нові дані будуть корисними лише якщо база даних не буде відставати і підтримуватиме необхідну швидкість обробки. Метою даної роботи є порівняння двох типів баз даних та визначення того, який тип краще використовувати залежно від поставленого завдання. Було проведено аналітичний огляд реляційних та нереляційних баз даних. Проаналізовані переваги та недоліки SQL та NoSQL баз даних. Проведено тестування систем керування SQL та NoSQL базами даних. За допомогою програми Sysbench на основі операційної системи Linux було виконано тестування СУБД MySQL та MongoDB.*

**Ключові слова:** база даних, система керування базами даних, SQL, NoSQL, MySQL, MongoDB, Sysbench.

За останні десятиліття значно збільшилось використання датчиків. Вони можуть бути застосовані при вимірюванні фізичної структури для виявлення аномалій, моніторингу енергетичної мережі, щоб зрівноважити виробництво та споживання енергії, при роботі з інтернетом речей та розумними середовищами, такими як розумні будинки [1, с. 431]. Датчики стають меншими, простішими у використанні та дешевшими. Ці зміни призводять до зростання попиту на зберігання та обробку даних.

Ці дані зазвичай незмінні, великі за обсягом, впорядковані за часом і в основному агреговані для доступу. Для обробки всієї інформації потрібно створення величезної кількості сховищ даних різного призначення. Сучасні інформаційні системи характеризуються величезними обсягами збережених даних, складною організацією, необхідністю задовольняти різноманітні вимоги численних користувачів. Основою інформаційної системи є бази даних.

Метою будь-якої інформаційної системи є обробка даних про об'єкти реального світу. У широкому сенсі слова, база даних – це сукупність відомостей про конкретні об'єкти реального світу у будь-якій предметній області.

Актуальність роботи надає пришвидшення темпів автоматизації керівничо-виробничої, офісної та інших видів робіт, де необхідні збір, збе-

рігання, переробка, отримання та передача інформації. Вибір необхідної бази даних відіграє дуже важливу роль у забезпеченні швидкості обробки даних, пошуку необхідної інформації серед безлічі даних та коректному доступі до них.

**Постановка проблеми.** У роботі з великою кількістю даних основна проблема полягає у тому, що чим більший обсяг накопичених даних, тим більш вимоглива система зберігання і управління цими даними. Необхідно купувати або дороге обладнання, або миритися з недоліками зберігання даних у хмарних середовищах.

На тепер немає єдиного підходу для всіх технологій баз даних. Саме тому у залежності від завдання використовують як реляційні, так і нереляційні бази даних.

**Аналіз останніх досліджень і публікацій.** Галузеві аналітики очікують, що до 2020 року у повсякденні об'єкти будуть вбудовані 40 мільярдів датчиків. У сучасному поколінні автомобілів генерується 25 ГБ даних на годину. Наступне покоління буде генерувати 250 ГБ на годину. Традиційні технології управління даними не були призначені для обробки такої кількості даних або темпів зміни.

Згідно з рейтингом DB-Engines Ranking [10], реляційні бази даних є більш використовуваними та популярними за кількістю згадок про них.

Аналізуючи реляційні і нереляційні бази даних, Ян Сіпке ван дер Вейн говорить, що NoSQL бази даних намагаються вирішити деякі проблеми традиційних баз даних SQL [1, с. 431]. Послаблюючи вимоги до узгодженості або доступності, база даних NoSQL може працювати досить добре в умовах, коли ці обмеження не є необхідними [2, с. 2].

У своїй статті Ішань Лі і Сатімаорті Манохара вказують на те, що не завжди є беззаперечна перевага у швидкодії між реляційними і нереляційними базами даних [3, с. 18].

Постановка завдання. Метою даної роботи є порівняння двох типів баз даних та визначення який тип краще використовувати у залежності від поставленого завдання, проведення аналітичного огляду реляційних та нереляційних баз даних. Аналіз переваг та недоліків SQL та NoSQL баз даних. Тестування швидкодії систем керування SQL та NoSQL базами даних.

Виклад основного матеріалу дослідження. Реляційні (SQL) сховища забезпечують найкращу суміш простоти, стійкості, гнучкості, продуктивності, масштабованості і сумісності. Але основною проблемою є те, що різноманітність додатків зростає, а отже, виникає необхідність у збільшенні місця. З ростом кількості баз даних масштабованість починає переважати над іншими особливостями. Оскільки все більше додатків працюють в умовах високого навантаження, наприклад, таких як веб-сервіси, їх вимоги до масштабованості дуже швидко змінюються і сильно зростають. Саме тому почали з'являтися бази даних, що забезпечують інший механізм зберігання та видобування даних, ніж звичайний підхід таблиць-відношень у реляційних базах даних. Сплеск популярності був спричинений потребами Web 2.0 компаніями, такими як Facebook, Google та Amazon.com.

Нереляційні (NoSQL) бази даних є документно-орієнтовані, їх співвідносять із папкою, наповненою файлами. Однією з основних особливостей технології NoSQL є відмова від реляційної моделі. Кожне рішення у рамках технології NoSQL використовує свою власну модель. NoSQL бази даних все більше і більше використовуються у задачах зі застосуванням Big Data та веб-додатках.

NoSQL бази даних, у порівнянні з SQL, набагато краще масштабуються, не накладають обмежень на типи даних, які можна зберігати разом, і дозволяють додавати нові типи при зміні потреб. З NoSQL базами даних можна зберігати різні типи даних в одному місці, не вказуючи завчасно, які типи даних необхідно використовувати. Нереляційні бази даних дешевші і мають швидшу

обробку даних. Але при цьому NoSQL бази даних жертвують відповідністю вимогам ACID для забезпечення гнучкості та швидкості обробки, що є не бажаним для фінансових додатків. Натомість нереляційні бази даних засновуються на принципі «кінцевої послідовності». Це забезпечує деякі переваги продуктивності, але створює ризик того, що дані на одному вузлі бази даних можуть не синхронізуватися з даними іншого вузла. Однак правильно підібрані критерії, згідно з теоремою CAD, допоможуть досягти необхідної швидкості обробки інформації, доступності та узгодженості даних.

Переваги SQL бази даних:

1. Відповідність ACID [5] (атомарність, узгодженість, ізолюваність, довговічність), що захищає цілісність бази даних, визначаючи точно, як транзакції взаємодіють з базою даних.

2. Кожна таблиця містить одну або декілька категорій даних у стовпцях [6].

3. Кожен рядок містить унікальний екземпляр даних для категорій, визначених стовпцями [7].

4. Користувач може отримати доступ до даних з бази даних, не знаючи структури таблиці бази даних [6].

Недоліки SQL бази даних:

1. Масштабованість: користувачі повинні масштабувати реляційні бази даних на потужних серверах, які є дорогими та складними для обробки. Щоб масштабувати реляційну базу даних, вона повинна бути розподілена на декілька серверів [6].

2. Складність: дані SQL-сервера повинні бути вписані у таблиці. Якщо дані не вкладаються у таблиці, необхідно проектувати структуру бази даних, яка буде складною і важкою для обробки [6].

3. Вартість: реляційні системи керування базами даних (СКБД) вимагають дорогих систем зберігання та запатентованих серверів [7].

Переваги NoSQL бази даних:

1. Зберігання великих обсягів даних, які погано структуровані. База даних NoSQL не обмежує типи даних, які можна зберігати разом, і дозволяє додавати нові типи. За допомогою баз даних на базі документів можна зберігати дані в одному місці, не визначаючи заздалегідь, які «типи» цих даних [8].

2. Швидша обробка даних, ніж у реляційних базах даних [6].

3. Дешевше: не реляційні бази даних використовують дешеві кластери товарних серверів для управління операціями та даними [7].

4. Краща масштабованість у порівнянні з реляційними базами даних. Проте NoSQL бази даних не повністю масштабовані у всіх ситуаціях [9].

Недоліки NoSQL бази даних:

1. Узгодженість даних: більшість баз даних NoSQL не виконують транзакції ACID. Натомість NoSQL покладається на принцип «кінцевої послідовності». Це забезпечує деякі переваги продуктивності, але це створює ризик того, що дані на одному вузлі бази даних можуть не синхронізуватися з даними іншого вузла [9];

2. Відсутність стандартизації: NoSQL не є специфічним типом інтерфейсу бази даних або програмування. Мова дизайну та запитів баз даних NoSQL різко відрізняється між різними продуктами NoSQL – набагато ширше, ніж серед традиційних баз даних SQL [9].

3. Деякі не реляційні бази даних погано розповсюджуються на декілька вузлів. Якщо база даних не може відокремитись автоматично, вона не може автоматично збільшуватись або зменшуватись у відповідь на коливальний попит.

Для того, щоб визначити який тип баз даних, реляційні чи нереляційні, найбільше підходить для роботи з великими даними, було обрано найбільш популярні системи керування базами даних (згідно з рейтингом DB-Engines Ranking [10]). Для реляційних баз даних для тестування було обрано СКБД MySQL (2 місце у рейтингу серед реляційних баз даних), для нереляційних – MongoDB (1 місце серед нереляційних баз даних).

Гнучкість СКБД MySQL забезпечується підтримкою великої кількості типів таблиць: користувачі можуть вибрати як таблиці типу MyISAM, що підтримують повнотекстовий пошук, так і таблиці InnoDB, що підтримують транзакції на рівні окремих записів. Більш того, СКБД MySQL поставляється зі спеціальним типом таблиць EXAMPLE, що демонструє принципи створення нових типів таблиць. Завдяки відкритій архітектурі і GPL-ліцензуванню, у СКБД MySQL постійно з'являються нові типи таблиць.

MongoDB – це документно-орієнтована база даних. Основною причиною відходу від реляційної моделі є полегшення масштабування. Документно-орієнтована база даних замінює поняття «рядок» більш гнучкою моделлю – «документом». За допомогою вкладеного документа та масиву, документно-орієнтований підхід дозволяє представляти складні ієрархічні зв'язки з єдиним записом [11, с. 3].

Горизонтальне масштабування, яке наявне у MongoDB, значно полегшує роботу з додатками.

З кількома варіантами масштабування, у тому числі на основі діапазонів, з використанням хеш і обміну зонами – MongoDB може підтримувати тисячі вузлів, петабайт даних і сотні тисяч операційних систем у секунду, не вимагаючи створення користувацьких розділів і кешування.

Аналіз, візуалізація і реагування на вихідні дані з датчиків вимагають потужних інструментів, які можуть запускати складні запити з низькою затримкою у наборах даних, що швидко змінюються. Завдяки потужній індексації і підтримці запитів, у тому числі підтримці вторинних запитів, наявності геопросторових і текстових пошукових індексів, структури агрегації і вбудованій підтримці Apache Spark – MongoDB може запускати складну спеціальну або звітну аналітику даних з датчиків прямо на місці.

За безпеку даних у MongoDB відповідають надійні засоби перевірки автентичності, авторизації, аудиту та шифрування. Вони захищають цінні дані з датчиків і отримані на основі них дані аналітики.

Обрані СКБД було протестовано на швидкодію за допомогою програми для тестування ефективності системи з (потенційно) високим навантаженням СКБД Sysbench.

Для тестування СКБД MySQL було встановлено на налаштовано додаток, створено базу даних. При тестуванні було встановлено 10 одночасно запущених запитів, тип тесту oltp. У таблицю було записано 10 000 рядків.

Для тестування MongoDB було встановлено розширення для Sysbench, яке дозволяє проводити тестування цієї бази даних. Далі, за умови того, що встановлено MongoDB, необхідно створити досліджувану БД, та користувача. Для тесту створено 16 колекцій, кожна з яких складається з 10 мільйонів документів. Можливо переглянути розмір бази даних відносно розміру пам'яті, щоб забезпечити тестування лише робочого навантаження на основі пам'яті та робочого навантаження, що перевищує обсяг пам'яті та використання диска.

З отриманих результатів тестування було вираховано середню кількість записів. Система керування нереляційною базою даних MongoDB робить у 15 разів більшу кількість вставок за секунду, ніж система керування реляційною базою даних MySQL. Різницю у швидкодії можна побачити на рис. 1.

Результати, отримані у ході тестування, слід порівнювати лише між собою, оскільки вони були виконані за однакових умов. При викорис-

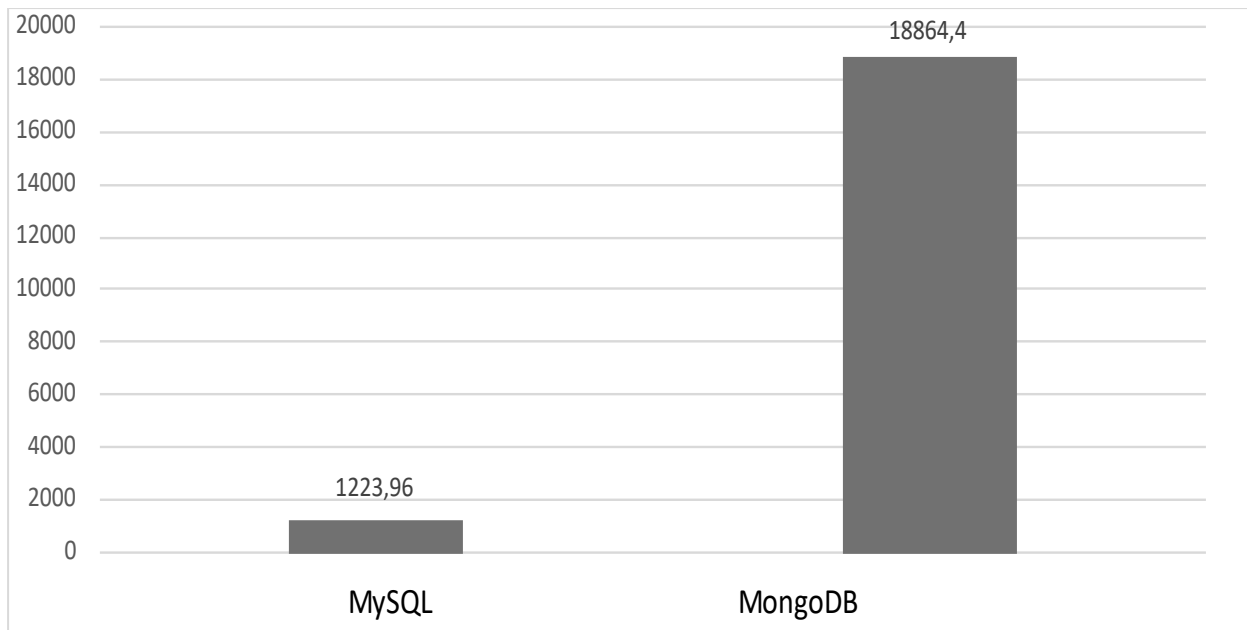


Рис. 1. Кількість вставок для СКБД MySQL і MongoDB

танні більш потужної техніки результати можуть змінитись.

Але не можна стверджувати, що нереляційні бази даних повністю витіснять реляційні у майбутньому. Навіть оскільки бази даних NoSQL посилюють популярність за їх швидкістю та масштабованістю, все ще є ситуації, коли доцільною є високо структурована SQL база даних.

На тепер не існує єдиного підходу для всіх технологій баз даних. Саме тому у залежності від завдання використовують як реляційні, так і нереляційні бази даних. NoSQL бази даних стають все більш популярними внаслідок їх швидкодії та здатності до масштабування, але все одно є ситуації, коли перевага віддається більш структурованим SQL базам даних.

Як правило, база даних NoSQL жертвує ACID для забезпечення гнучкості та швидкості обробки, але для електронної комерції та фінансових додатків переважним варіантом залишається база даних, сумісна з ACID.

**Висновки.** Аналізуючи все, що було викладено у даній статті, можна зробити висновок, що SQL бази даних є сенс використовувати, коли необхідно забезпечити транзакції та надійність системи у сенсі зв'язків між таблицями. Якщо дані вдається структурувати, то далі працювати з ними набагато легше.

Всі реляційні бази даних підтримують SQL, саме тому при зміні реляційних баз даних у користувача і розробника не виникне суттєвих проблем.

NoSQL бази даних краще використовувати, коли інформацію, яку необхідно зберегти, важко структурувати і її тип відмінний для різних об'єктів; при необхідності зберігати великі кількості інформації з датчиків температури, вологості, місцезнаходження, датчиків медичної апаратури для аналізу і подальшої їх обробки; коли навантаження на систему може зростати з часом, і система потребує масштабування на декілька машин; при розробці додатків, де можна пожертвувати зв'язками між таблицями для отримання високої швидкодії.

Виходячи з отриманих результатів, вважаємо, що нереляційні бази даних будуть все більше застосовуватись при розробці додатків. Адже кількість інформації, яка передається з датчиків, використовується у веб-сервісах, необхідна для аналізу – стрімко зростає. Те, що NoSQL бази даних здатні до масштабування, робить їх суттєво більш вигідними для розробників, а перевага у швидкодії дозволяє підтримувати необхідну швидкість обробки потоків даних та вчасно реагувати на збої у системі.

Отримані результати можуть бути використані для досліджень у сферах інформатики та електроніки. При виборі необхідного типу бази даних для зберігання і подальшої обробки результатів вимірювань з датчиків, опрацювання великих даних, при розробці додатків для інтернету речей, алгоритмів машинного навчання, розробці веб-ресурсів та в інших сферах життя людини.

**Список літератури:**

1. J. S. van der Veen, B. van der Waaij, and R. J. Meijer. Sensor data storage performance: SQL or NoSQL, physical or virtual, in Proceedings of the IEEE 5th International Conference on Cloud Computing (CLOUD '12), pp. 431–438, IEEE, June 2012.
2. Agrawal R., Ailamaki A., Bernstein P. A., Brewer E. A. and G. Weikum, The claremont report on database research. 2008, 11 pages.
3. Li Y., Manohara S. A performance comparison of SQL and NoSQL databases. Conference: Communications, Computers and Signal Processing (PACRIM), 2013 IEEE Pacific Rim Conference on, pp. 15-19, IEEE, August 2013.
4. SQL – Transactions. URL: <https://www.tutorialspoint.com/sql/sql-transactions.htm> (дата звернення: 07.11.2018).
5. Richards J. Advantages of NoSQL Databases. URL: <https://bigdata-madesimple.com/advantages-of-nosql-databases-what-you-need-to-know/> (дата звернення: 11.05.2018).
6. The Limitations of NoSQL Database Storage. URL: <http://www.channelfutures.com/cloud-services/limitations-nosql-database-storage-why-nosqls-not-perfect> (дата звернення: 11.05.2018).
7. Wodehouse C. SQL vs. NoSQL Databases: What's the Difference? / Carey Wodehouse. URL: [www.upwork.com/hiring/data/sql-vs-nosql-databases-whats-the-difference/](http://www.upwork.com/hiring/data/sql-vs-nosql-databases-whats-the-difference/) (дата звернення: 07.06.2018).
8. 10gen – ещё один стартап в области Cloud Computing. URL: [special.habrahabr.ru/kyocera/p/3660/](http://special.habrahabr.ru/kyocera/p/3660/) (дата звернення: 15.05.2018).
9. Knowledge Base of Relational and NoSQL Database Management Systems. URL: <https://db-engines.com/en/ranking> (дата звернення: 07.11.2018).
10. DB-Engines Ranking. URL: <https://db-engines.com/en/ranking> (дата звернення: 21.10.2018).
11. K. Chodorow and M. Dirolf, MongoDB: The Definitive Guide. O'Reilly Media, 2010, 409 pages.
12. Babarika E. Как создать пользователя MySQL и настроить права доступа. URL: [www.hostinger.ru/rukovodstva/kak-sozdat-polzovatelya-mysql-i-nastroit-prava-dostupa/](http://www.hostinger.ru/rukovodstva/kak-sozdat-polzovatelya-mysql-i-nastroit-prava-dostupa/) (дата звернення: 02.03.2018).
13. MongoDB CRUD Operations. URL: <https://docs.mongodb.com/manual/crud/> (дата звернення: 07.06.2018).
14. MySQL 5.6 Reference Manual. URL: <https://dev.mysql.com/doc/refman/5.6/en/> (дата звернення: 02.03.2018).

**СРАВНЕНИЕ SQL И NOSQL БАЗ ДАННЫХ**

*За последние десятилетия значительно увеличилось использование приложений, генерирующих новые потоки данных, которые ранее были немислимы как по разнообразию, так и по количеству. Но эти новые данные будут полезными только если база данных не будет отставать и будет поддерживать необходимую скорость обработки. Цель данной работы – сравнить два типа баз данных и определить какой тип лучше использовать в зависимости от поставленного задания. Был проведен аналитический осмотр реляционных и нереляционных баз данных. Проанализировано преимущества и недостатки SQL та NoSQL баз данных. Проведено тестирование системы управления SQL и NoSQL базами данных. С помощью программы Sysbench на основе операционной системы Linux было выполнено тестирование СУБД MySQL и MongoDB.*

**Ключевые слова:** база данных, система управления базами данных, SQL, NoSQL, MySQL, MongoDB, Sysbench.

**COMPARISON OF SQL AND NOSQL DATABASES**

*Over the past decades, the number of applications that generate big amount of data has increased significantly. But this data will be useful only if the database keeps up and maintains the necessary processing speed. The purpose of this article is to compare two types of databases. An analytical review of relational and non-relational databases was conducted. The advantages and disadvantages of SQL and NoSQL databases was analyzed.*

*The purpose of this work is to compare two types of databases. Determine which type suits best depending on the task. Performance of SQL and NoSQL database-management system was measured. With using of the program Sysbench based on the Linux operating system, MySQL and MongoDB were tested.*

**Key words:** database, database-management system, SQL, NoSQL, MySQL, MongoDB, Sysbench.