ІНФОКОМУНІКАЦІЙНІ ТЕХНОЛОГІЇ ТА ЕЛЕКТРОННА ІНЖЕНЕРІЯ
INFORMATION AND COMMUNICATION TECHNOLOGIES, ELECTRONIC ENGINEERING

№ 5 (2), 2025

https://doi.org/10.23939/ictee2025.02.083

МЕТОД КОМБІНОВАНОГО ПАРТИЦІЮВАННЯ ВЕЛИКИХ ДАНИХ В ІНФОРМАЦІЙНИХ СИСТЕМАХ

В. Сологуб [ORCID: 0000-0003-1553-530X], **М. Бешлей** [ORCID: 0000–0002–7122–2319]

Національний університет "Львівська політехніка", вул. С. Бандери, 12, Львів, 79013, Україна

Відповідальний за рукопис: М. Бешлей (e–mail: mykola.i.beshlei@lpnu.ua)

(Подано 6 серпня 2025)

У сучасних умовах швидкого зростання обсягів даних інформаційні системи мають забезпечувати не лише зберігання та доступ до великих масивів інформації, а й стабільну продуктивність під час виконання різнотипних запитів. Важливим завданням є досягнення балансу між ефективністю аналітичних (OLAP) операцій та швидкодією транзакційних (OLTP) процесів. Традиційні методи організації даних у реляційних СУБД часто втрачають ефективність у масштабованих середовищах, що призводить до збільшення тривалості оброблення, зниження гнучкості та ускладнення управління базами даних. Це зумовлює актуальність пошуку нових підходів до оптимізації партиціювання даних, здатних забезпечити високу швидкодію та масштабованість у гібридних інформаційних системах. У статті досліджено відомі методи партиціювання даних в інформаційних системах, орієнтованих на роботу з великими обсягами структурованої інформації та здатних обслуговувати одночасно OLAP і OLTP навантаження. Проаналізовано механізми розділення таблиць у сучасних СУБД. Визначено переваги й недоліки кожного підходу з урахуванням вимог до швидкодії, масштабованості та зручності управління даними. Запропоновано метод комбінованого партиціювання даних (range + list), адаптований до гібридних інформаційних систем, що одночасно обслуговують OLAP та OLTP навантаження. Відмінністю від традиційних підходів є не лише застосування комбінованого партиціювання для аналітичних завдань, а й комплексний аналіз його впливу на ефективність виконання аналітичних запитів та швидкодію транзакційних операцій. Отримані результати підтверджують, що розроблений метод забезпечує баланс між продуктивністю обох типів навантажень, сприяє підвищенню масштабованості та гнучкості інформаційних систем і може розглядатися як універсальний підхід для роботи з великими обсягами даних. Для дослідження побудовано уніфіковану імітаційну модель оброблення даних в інформаційних системах за схемою "зірки" з фактологічною таблицею продажів, що дає змогу виконувати як операції бізнес-аналітики, так і транзакційні CRUD-операції. Експериментальні результати доводять, що комбіноване партиціювання зменшує тривалість виконання аналітичних запитів на 30–40 % без істотних втрат швидкодії CRUD-операцій, що робить його ефективним інструментом підвищення продуктивності масштабованих інформаційних систем із великими обсягами даних.

Ключові слова: *партиціювання даних, OLAP, OLTP, інформаційна система, бази даних, оптимізація запитів, великі дані.*
УДК: 621.391

Вступ

Реляційні системи управління базами даних (СУБД) тривалий час залишаються основою для зберігання й оброблення інформації у багатьох прикладних системах. Водночас сучасний етап розвитку цифрових технологій характеризується появою нових типів програмних застосунків, інтенсивним зростанням обсягів даних і поширенням хмарних обчислень [1]. За цих умов до систем зберігання даних ставлять нові вимоги, що стосуються не лише забезпечення цілісності інформації, але й високої швидкодії, масштабованості та адаптивності до різнотипних навантажень.

Обробка великих масивів інформації потребує використання високопродуктивних обчислювальних платформ і спеціалізованих механізмів управління даними, здатних забезпечити оперативне реагування на запити користувачів із високою точністю та продуктивністю [2]. Проте сучасні прикладні завдання не завжди можна ефективно реалізувати в межах класичної реляційної моделі, яка орієнтована переважно на роботу зі структурованими даними. Натомість сучасні застосунки (мобільні, веб- та IoT-сервіси) генерують різнорідні потоки інформації, що містять структуровані, напівструктуровані та неструктуровані дані [3]. Це ставить під сумнів ефективність традиційних підходів до зберігання та опрацювання даних, оскільки вони виявляються недостатньо гнучкими для таких умов. У таких умовах особливого значення набувають гібридні інформаційні системи, які повинні одночасно підтримувати транзакційні навантаження (OLTP), орієнтовані на швидке виконання великої кількості коротких операцій у режимі реального часу, та аналітичні навантаження (OLAP), спрямовані на оброблення великих масивів даних із використанням агрегацій, багатовимірного аналізу та формування звітів.

Із огляду на це, актуальним завданням є пошук нових методів, здатних підвищити ефективність управління великими обсягами даних у гібридних інформаційних системах, де одночасно поєднуються і OLAP і OLTP навантаження. Один із перспективних напрямів розв’язання цього завдання – застосування комбінованих методів партиціювання даних, які дають змогу оптимізувати швидкодію, забезпечити масштабованість та гнучкість інформаційних систем у сучасних цифрових середовищах.

2. Розроблення імітаційної моделі даних інформаційної системи для дослідження методів партиціювання

Зі збільшенням розміру таблиць бази даних до сотень гігабайт і більше істотно ускладнюється виконання базових операцій, таких як завантаження нових записів, видалення застарілих даних та підтримка індексів. Великі обсяги інформації призводять до істотного зростання тривалості оброблення, унаслідок чого навіть стандартні CRUD-операції стають малоефективними. Для оптимізації цих процесів сучасні системи управління базами даних передбачають використання методів партиціювання, які полягають у розподілі таблиць на менші частини (партиції), що підвищує швидкодію, масштабованість і керованість системи [4].

У цьому дослідженні основну увагу зосереджено на розробленнii та оцінюванні методу партиціювання, здатного одночасно задовольняти вимоги аналітичних сховищ даних (OLAP) та транзакційних баз даних (OLTP). Ключова складність полягає у різній природі навантажень: у транзакційних системах критичною є швидкість вставлення, оновлення та видалення записів, тоді як в аналітичних – ефективність виконання запитів і формування агрегованих результатів.

Мета роботи – удосконалення методу партиціювання даних із комбінованим використанням відомих підходів, який забезпечує високу продуктивність для обох типів навантажень і дає змогу перевикористовувати єдиний об’єкт даних у різних сценаріях. Важливою перевагою такого підходу є можливість уникнути створення окремих структур або об’єктів, які дублюються, для різних типів навантажень, що часто спостерігається у традиційних підходах, коли аналітична й транзакційна логіка реалізуються окремо. Комбіноване партиціювання дає змогу створювати єдину логічну

модель даних, яка ефективно працює для обох типів запитів, зменшуючи потребу в дублюванні даних або створенні додаткових об'єктів лише під одну конкретну систему. Це, своєю чергою, відкриває додаткові можливості для малих і середніх підприємств, оскільки забезпечує повторне використання ресурсів та інфраструктури без потреби у створенні окремих інформаційних систем для кожного виду навантаження

Для дослідження різних методів порівняно із удосконаленим комбінованим у роботі побудовано модель даних, що імітує систему зберігання інформації про продажі. Вибрана модель придатна як для оброблення транзакційних даних, так і для виконання аналітичних операцій, що повністю відповідає поставленим завданням дослідження. Її структуру відображено на ER-діаграмі (рис. 1).

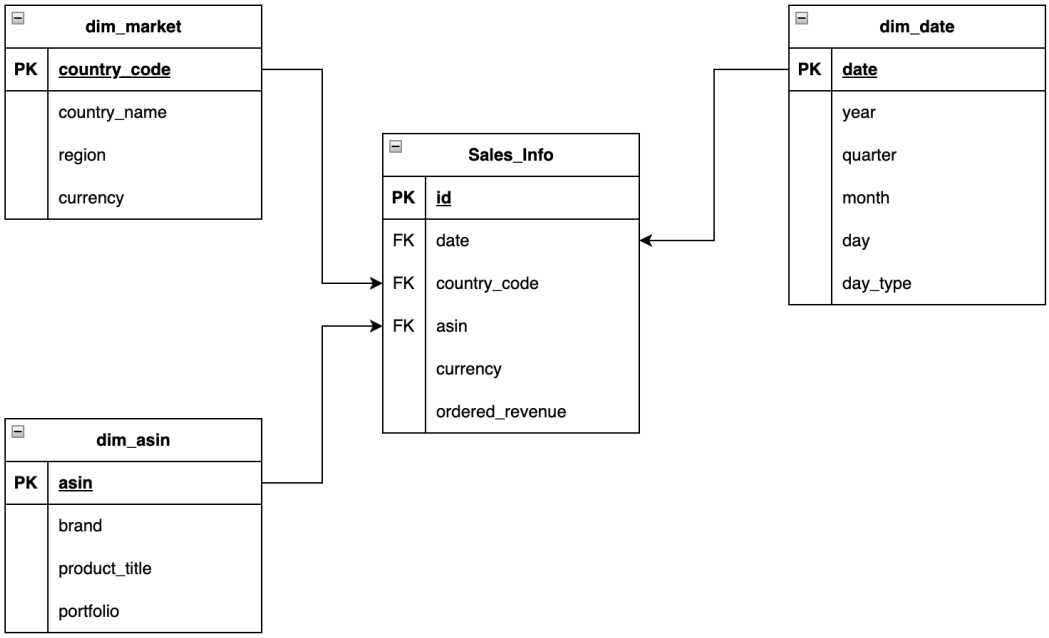

Рис. 1. ER-діаграма моделі даних

У поданій діаграмі відображено схему зіркоподібної моделі даних, призначеної для аналізу продажів. Центральною є фактова таблиця Sales_Info, яка містить інформацію про обсяги замовленого доходу за визначеними параметрами, деталізованими у відповідних вимірних таблицях (dim_market, dim_asin, dim_date).

Завдяки такій структурі модель дає змогу ефективно виконувати багатовимірний аналіз продажів за різними вимірами, а саме географічним, продуктовим та часовим. Її легко інтегрувати з OLAP-системами та інструментами бізнес-аналітики. Вона також відзначається гнучкістю, що дає змогу перевикористовувати її для запису транзакційних даних. Це забезпечує можливість масштабування під нові джерела інформації чи бізнес-процеси без потреби в істотній модифікації основної схеми. Фокус дослідження зосереджено на таблиці Sales_Info, структура якої найпридатніша для партиціювання. Цей метод передбачає розподіл таблиці та її індексів на менші частини (розділи), що дає змогу застосовувати операції окремо до кожного сегмента, а не до всієї таблиці. Завдяки цьому оптимізатор бази даних може спрямовувати відфільтровані запити безпосередньо до потрібних партицій, що істотно підвищує продуктивність.

Налаштування стовпця розділення, функції та схеми партиціювання дає змогу швидко додавати або видаляти великі обсяги даних, вносячи зміни лише у метадані, без фізичного переміщення рядків. Функція партиціювання, схема та розділена таблиця (або індекс) утворюють ієрархію залежностей. На верхньому рівні визначається функція партиціювання, яка задає принцип

розподілу даних між партиціями. На наступному рівні формується схема, що вказує на фізичне розміщення розділів у файлових групах (рис. 2). Лише після цього створюється таблиця або індекс, які використовують визначені партиції для зберігання інформації. Граничні значення функції визначають діапазони, що розмежовують розділи; кількість партицій завжди дорівнює кількості граничних значень плюс один. Файлова група може містити один або кілька файлів даних, розміщених на одному чи кількох дисках. Її можна налаштувати як доступну лише для читання, а також резервно копіювати та відновлювати окремо. Серед ключових переваг розміщення кожної партиції в окремій файловій групі варто виділити:

- можливість зберігання рідко використовуваних даних на повільніших носіях, тоді як найактивніші дані розташовують на швидших дисках;
- переведення історичних або незмінних даних у режим “тільки для читання” із вилученням їх із регулярного резервного копіювання, що зменшує навантаження на систему;
- вибіркове відновлення лише тих розділів, які містять найважливішу інформацію.

Отже, застосування партиціювання не лише оптимізує процеси зберігання та оброблення даних, а й підвищує ефективність резервного копіювання та відновлення у масштабованих інформаційних системах.

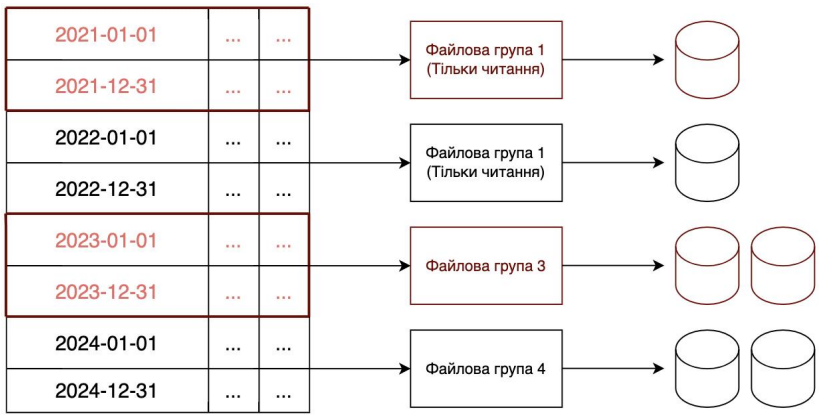

Рис. 2. Схема зберігання даних

У роботі розглянуто чотири використовувані методи партиціювання даних: циклічний, діапазонний, списковий та хеш-розподіл. Найпростішим серед них є циклічний метод (round-robin), який передбачає рівномірний розподіл рядків таблиці між партиціями у певній послідовності (рис. 3). Кожен новий запис додається до наступної партиції незалежно від його значень чи властивостей. Такий підхід простий у реалізації та забезпечує базовий рівень балансування навантаження між розділами.

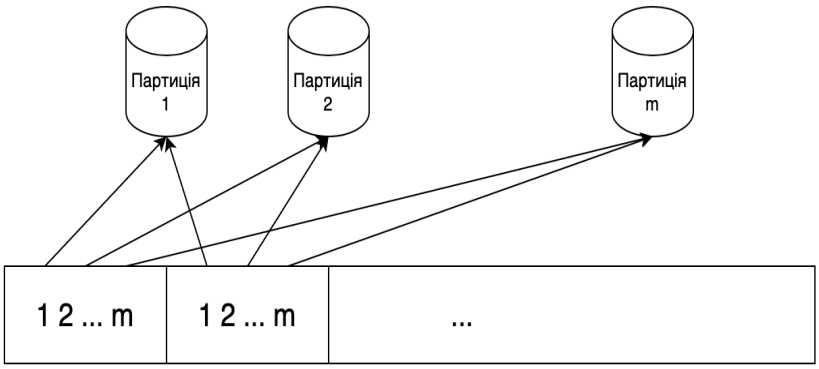

Рис. 3. Схема роботи циклічного методу партиціювання даних

Для методів партиціювання за діапазоном, списком або хешем необхідно попередньо визначити стовпець таблиці, який буде ключем розділення. Розподіл рядків у таблиці відбувається відповідно до значень цього ключа.

У разі застосування діапазонного партиціювання кожній партиції призначають певний інтервал значень. Дані розподіляються так, що кожна партиція містить записи, значення ключа яких належить до відповідного діапазону. Наприклад, за наявності атрибута мітки часу дані можуть бути розділені за часовими проміжками. Такий підхід особливо ефективний, коли впорядкування даних природне (наприклад, за датою), що дає змогу рівномірно розподіляти записи та забезпечувати швидке опрацювання запитів із фільтрацією за часовими межами.

У моделі даних, використаній у дослідженні, партиціювання за діапазоном для таблиці Sales_Info реалізовано на основі стовпця date. У цьому випадку кожна партиція охоплює певний проміжок часу, що підвищує ефективність виконання запитів та спрощує масштабування даних у часовому вимірі. Принцип роботи цього методу візуалізовано на рис. 4.

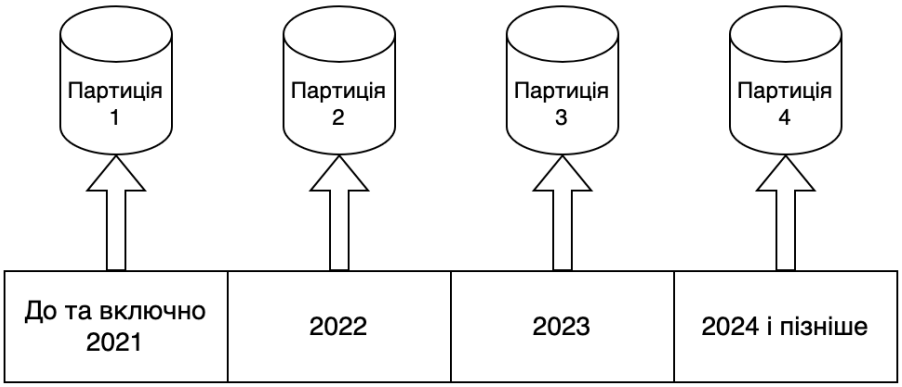

Рис. 4. Схема роботи методу партиціювання даних за діапазоном

Метод партиціювання за списком подібний до діапазонного підходу, проте замість інтервалів для кожної партиції визначається конкретний набір значень. Рядок потрапляє у відповідну партицію, якщо значення ключа розділення збігається з одним із елементів цього списку. У межах кожної партиції зберігаються всі записи, що належать до певного значення ключа. Такий метод часто застосовують у розподілених базах даних або системах, у яких логічно виправдане групування даних за визначеними категоріями. Він забезпечує ефективний розподіл навантаження та швидкий доступ до інформації за ключовими атрибутами. У дослідженні цей метод реалізовано для таблиці Sales_Info на основі атрибута country_code, що дало змогу розділити дані на дев’ять партицій відповідно до значень цього поля. Принцип роботи методу наведено на рис. 5.

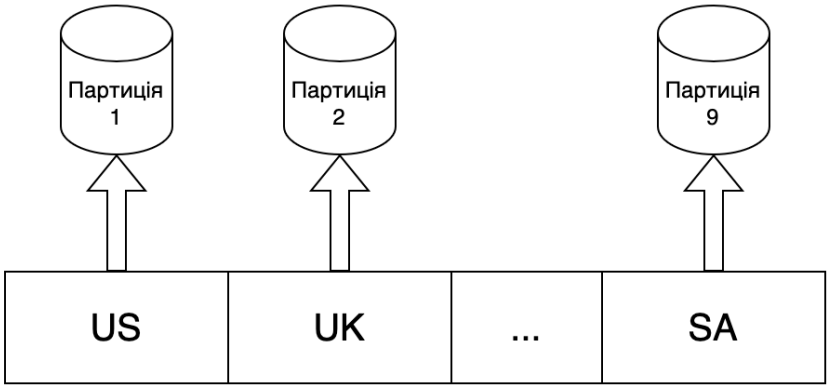

Рис. 5. Схема роботи методу партиціювання даних за списком

Метод хеш-партиціювання ґрунтується на використанні хеш-функції для визначення відповідності між значенням ключа розділення та конкретною партицією (рис. 6).

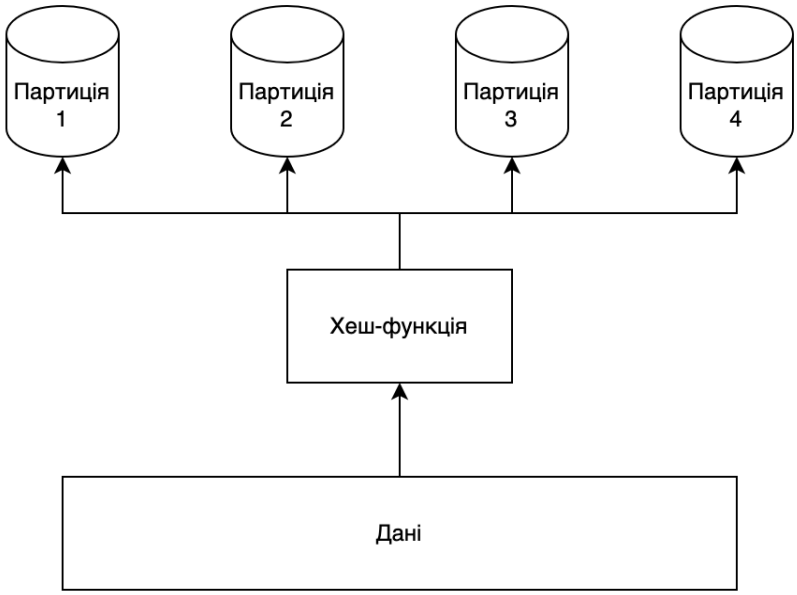

Рис. 6. Схема роботи методу партиціювання із хеш-функцією

Значення ключа передається у хеш-функцію, результат якої визначає, до якого розділу потрапить рядок. Після опрацювання система створює хеш-значення, що використовується для однозначного розподілу даних між партиціями. Такий підхід забезпечує випадковий, але рівномірний розподіл записів, що сприяє балансуванню навантаження між партиціями та підвищує ефективність доступу до даних у масштабованих системах.

3. Дослідження та порівняння ефективності використання наявних методів партиціювання із удосконаленим

На основі розробленої моделі даних інформаційної системи реалізовано методи партиціювання, спрямовані на підвищення ефективності зберігання та обробки інформації. Наступний етап дослідження – оцінювання впливу кожного методу на ключові показники ефективності. Доцільність застосування для OLAP-систем визначають за швидкодією формування аналітичних результатів, для OLTP-систем за продуктивністю операцій вставлення, оновлення та видалення даних.

Експерименти виконано на наборах даних різного масштабу: 100 тис., 1 млн і 10 млн рядків. Датасети мали однакову структуру та зберігали первинний розподіл записів, що гарантувало сталість статистичних характеристик незалежно від обсягу інформації (табл. 1).

Для перевірки оптимальності методів у OLAP-середовищі використано тестовий аналітичний запит, який формує звіт про продажі, виконуючи об’єднання фактологічної таблиці Sales_Info з вимірними таблицями. Структура запиту відповідає типовим вимогам бізнес-аналітики і забезпечує репрезентативність результатів. У запиті застосовано:

- складні агрегації;
- віконні функції;
- конструкції CTE (Common Table Expressions);
- вирази CASE;
- операції JOIN із усіма вимірними таблицями;
- аналіз доходів за категоріями, брендами, регіонами та святковими днями;
- обчислення динаміки продажів у часі.

Таблиця 1

Час оброблення аналітичного запиту OLAP за різних методів партиціювання, с

Стратегія партиціювання	100 тис. (розмір ~50 MB)	1 млн (розмір ~500 MB)	10 млн (розмір ~5 GB)
Без партиціювання	1,1	12,4	136,0
Циклічне (Round-Robin)	1,0	11,1	122,5
За діапазоном (Range by date)	0,4	3,2	31,4
За списком (List by country)	0,6	4,7	41,2
Хеш (Hash by ASIN)	0,8	6,5	56,0

Оскільки результати дослідження демонструють лінійну залежність тривалості виконання від обсягу даних, розраховано часовий коефіцієнт $K(A)$, який відображає середній час обробки одного рядка.

Обчислення здійснюється за формулою:

$$K(A) = \frac{t}{N}, \tag{1}$$

де t – час виконання аналітичного запиту; N – кількість рядків, на яких здійснюється експеримент.

Залежність часу виконання аналітичного запиту від обсягу даних із використанням різних методів партиціювання подано на рис. 7.

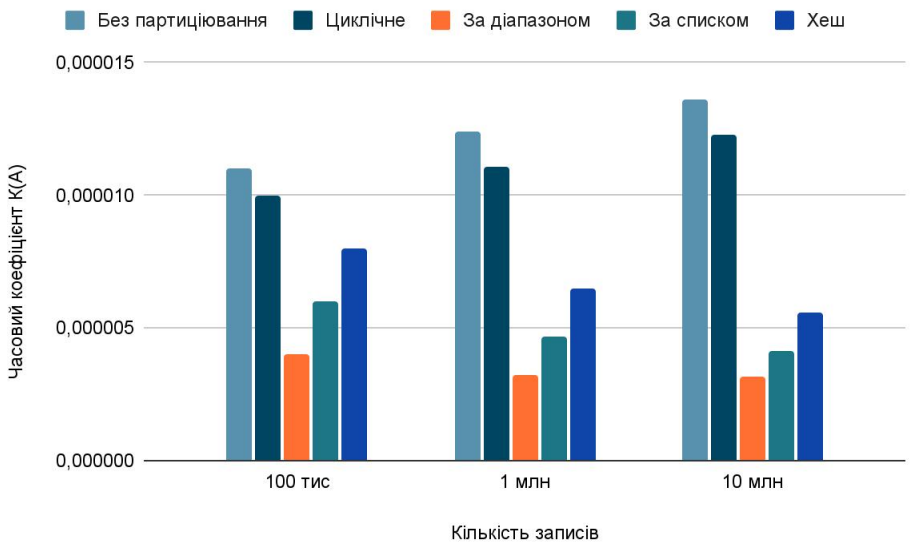

Рис. 7. Залежність часу виконання аналітичного запиту від обсягу даних із використанням різних методів партиціювання

Найгірші результати спостерігалися без партиціювання: тривалість виконання запитів зростала непропорційно до обсягу даних. Найвищу ефективність продемонстрував метод партиціювання за діапазоном, тоді як підходи за списком і хешем забезпечили помірне підвищення продуктивності. Циклічне партиціювання зумовило лише часткове зменшення часу виконання й виявилося малоефективним для великих масивів даних. Для оцінювання впливу методів у транзакційних системах було змодельовано типове OLTP-навантаження з операціями вставлення, оновлення та видалення:

- масове вставлення нових записів у фактологічну таблицю;
- оновлення значень у вибірках рядків, відібраних за ключовими атрибутами (наприклад, дата, країна, ідентифікатор товару);
- видалення записів за часовими та регіональними критеріями.

Для виконання порівняльного аналізу впливу різних методів партиціювання на оброблення OLTP навантаження у роботі введено часовий коефіцієнт ***K(T)***, який відображає середній час опрацювання одного рядка. На відміну від розрахунків, виконаних для OLAP орієнтованих інформаційних систем, у цьому випадку коефіцієнт визначали із урахуванням середнього часу виконання трьох основних транзакційних операцій типу UPSERT (вставлення, оновлення та видалення). Наведемо формулу розрахунку:

$$K(T) = \frac{t(i) + t(u) + t(d)}{N}, \tag{2}$$

де $t(i)$ – час вставлення із табл. 2, с; $t(u)$ – час оновлення із табл. 3, с; $t(d)$ – час видалення із табл. 4, с; N – кількість операції, що порівнюються.

Таблиця 2

Час вставлення даних за різних методів партиціювання, с

Стратегія партиціювання	100 тис. (розмір ~50 MB)	1 млн (розмір ~500 MB)	10 млн (розмір ~5 GB)
Без партиціювання	1,1	1,2	1,4
Циклічне (Round-Robin)	1,2	1,4	1,6
За діапазоном (Range by date)	1,3	1,7	2,0
За списком (List by country)	1,3	1,6	1,9
Хеш (Hash by ASIN)	1,2	1,5	1,7

Таблиця 3

Час оновлення даних за різних методів партиціювання, с

Стратегія партиціювання	100 тис. (розмір ~50 MB)	1 млн (розмір ~500 MB)	10 млн (розмір ~5 GB)
Без партиціювання	2,0	2,3	3,0
Циклічне (Round-Robin)	2,1	2,5	3,1
За діапазоном (Range by date)	2,7	3,6	4,5
За списком (List by country)	2,5	3,1	3,8
Хеш (Hash by ASIN)	2,4	2,9	3,5

Таблиця 4

Час видалення даних за різних методів партиціювання, с

Стратегія партиціювання	100 тис. (розмір ~50 MB)	1 млн (розмір ~500 MB)	10 млн (розмір ~5 GB)
Без партиціювання	2,2	2,5	3,2
Циклічне (Round-Robin)	2,2	2,6	3,3
За діапазоном (Range by date)	2,9	3,8	5,0
За списком (List by country)	2,6	3,2	4,2
Хеш (Hash by ASIN)	2,5	3,0	3,9

Результати вимірювань впливу різних методів партиціювання на продуктивність OLTP-операцій (вставлення, оновлення та видалення) підтверджують, що вибір підходу має вирішальне значення для досягнення оптимальної швидкодії в умовах масштабування. Залежність часу виконання транзакційного запиту від обсягу даних із використанням різних методів партиціювання наведено на рис. 8.

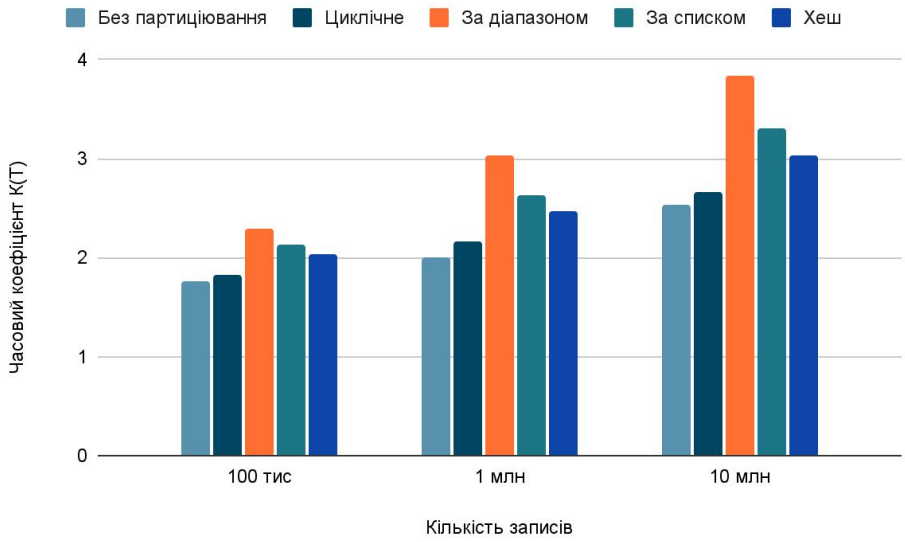

Рис. 8. Залежність часу виконання транзакційного запиту від обсягу даних із використанням різних методів партиціювання

Найменші затримки спостерігаються за відсутності партиціювання, що пояснюється мінімальними накладними витратами. Водночас зі зростанням обсягів даних ускладнюється управління таблицями та істотно знижується ефективність. Циклічний метод забезпечує стабільну продуктивність у середовищах із високим рівнем паралельності, тоді як хеш-партиціювання демонструє переваги у разі оновлення даних за ключем. Метод за діапазоном характеризується зростанням тривалості виконання операцій із розширенням кількості ключів у партиції, що особливо помітно у великих масивах даних. Партиціювання за списком виявило стабільність у разі поступового розширення довідників (наприклад, додаванні нових країн).

Отже, жоден із розглянутих методів окремо не може повною мірою задовольнити одночасні вимоги OLAP і OLTP систем. Тому запропоновано комбінований підхід, що поєднує переваги діапазонного та спискового методів (рис. 9). Для опрацювання транзакційних навантажень використано партиціювання за часовим атрибутом (date), тоді як для оптимізації аналітичних запитів додатково застосовано розділення за категоріальною ознакою (country_code). У результаті кожна діапазонна партиція поділяється на підмножини за категоріями, що забезпечує підвищену ефективність і прозору логіку доступу до даних. Для реалізації комбінованого методу доцільно створити службовий стовпець, у якому зберігатимуться значення, що поєднують часовий діапазон і категоріальну ознаку, що спрощує роботу СУБД із партиціями. У перспективі передбачений розвиток цього підходу із інтеграцією методів партиціювання з індексуванням, що дасть змогу сформувати вдосконалений комбінований метод із розширеними можливостями.

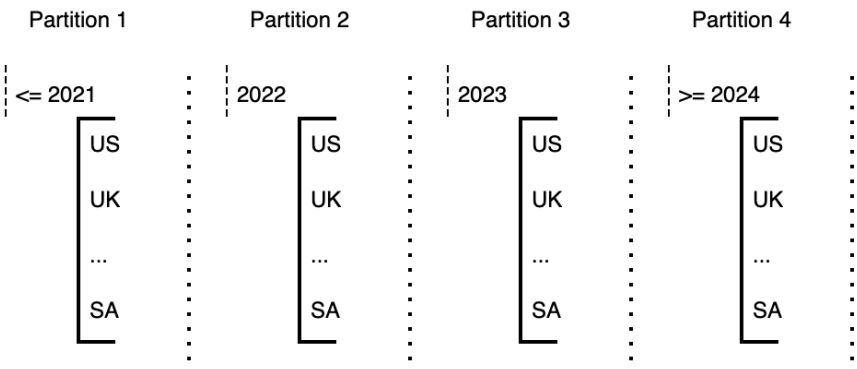

Рис. 9. Запропонований комбінований метод партиціювання даних

Для оцінювання продуктивності запропонованого методу виконано експериментальні дослідження, які відповідали умовам виконання як аналітичних запитів, так і транзакційних операцій. Для об'єктивного порівняння використано показники найефективнішого методу в кожній із відповідних категорій. За результатами порівняння в аналітичному середовищі встановлено, що найефективнішим підходом є партиціювання за проміжком (range partitioning), яке забезпечує найвищу продуктивність під час виконання аналітичних запитів (рис. 10).

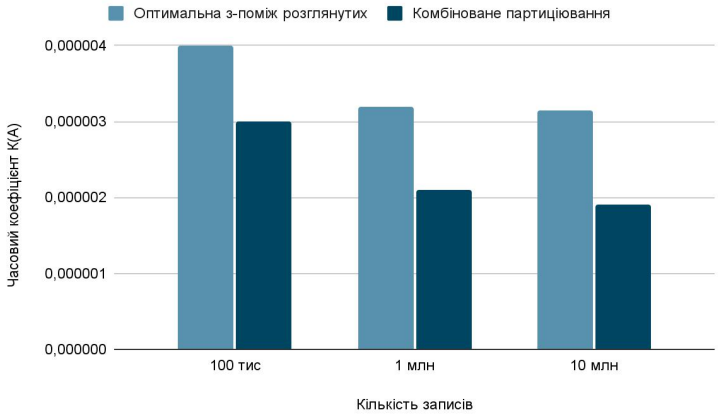

Рис. 10. Порівняння ефективності запропонованого методу для OLAP

Експериментальні результати показали, що зі збільшенням обсягу даних значення коефіцієнта $K(A)$ зменшується, що особливо помітно у разі використання комбінованого методу партиціювання (range + list). Таке зниження пояснюється підвищенням масштабованості та ефективності розподілу навантаження на систему. Комбіноване партиціювання забезпечує підвищену селективність доступу до даних: діапазонне розбиття дає змогу швидко локалізувати потрібний інтервал, тоді як спискове впорядковує записи всередині цього інтервалу за категоріальними ознаками. У результаті зменшується кількість рядків, що підлягають зчитуванню й опрацюванню, що безпосередньо зменшує тривалість виконання запитів. Додатково, у разі зростання обсягів даних оптимізатор запитів ефективніше розподіляє ресурси між операціями фільтрації, агрегації та сортування, що ще більше зменшує загальний час обробки. Для повноти аналізу розглянуто також результати у транзакційних системах, де у таблицях наведено значення продуктивності у форматі: вставлення / оновлення / видалення. На рис. 11 наведено результати розрахунку та порівняння коефіцієнта ефективності $K(T)$ в умовах використання запропонованого комбінованого методу партиціювання для OLTP навантаження в інформаційних системах

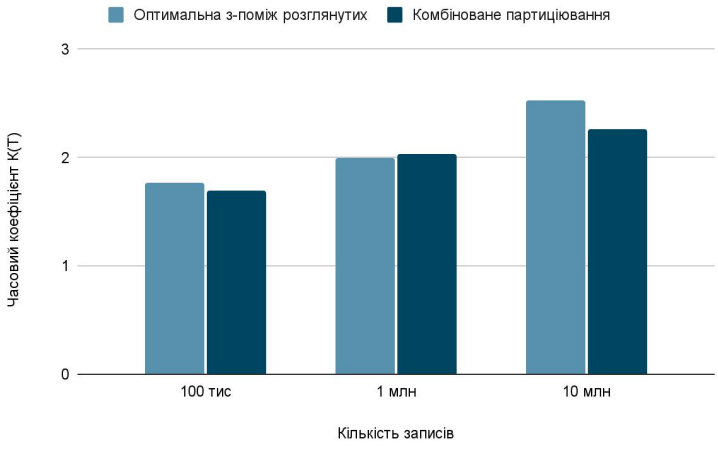

Рис. 11. Порівняння ефективності запропонованого методу для OLAP

На основі отриманих результатів доведено, що комбіноване партиціювання забезпечує стабільну продуктивність за будь-яких обсягів даних і за більшістю показників не поступається іншим відомим методам, разом із циклічним партиціюванням. Останній у деяких випадках демонструє дещо нижчі значення коефіцієнта $K(T)$ завдяки рівномірному розподілу навантаження між партиціями. Водночас комбінований підхід має істотні переваги в організації структурованого доступу до даних: поєднання діапазонного та спискового розбиття зменшує кількість фізичних звернень до диска під час виконання складних транзакцій, підвищує локальність даних і сприяє кращій оптимізації індексів. Такі властивості критично важливі в умовах зростання обсягів даних та роботи систем у режимі змішаних навантажень. Отже, хоча у транзакційних сценаріях комбіноване партиціювання не завжди забезпечує абсолютне домінування за швидкодією, його ефективність залишається стабільно високою та порівнюваною із найрезультативнішими підходами. На відміну від циклічного методу, доведено перевагу комбінованого підходу також в аналітичних завданнях, оскільки він продемонстрував значно кращі показники. Сукупність цих факторів дає підстави стверджувати, що комбіноване партиціювання – найуніверсальніший метод, придатний як для транзакційних, так і для аналітичних систем, особливо в умовах гібридних навантажень.

Висновок

У роботі змодельовано та проаналізовано ефективність різних методів партиціювання даних в інформаційних системах, орієнтованих на одночасне обслуговування OLTP- та OLAP-навантажень. Для експериментів побудовано уніфіковану модель таблиці Sales_Info, яку протестовано із застосуванням шести підходів партиціювання, разом із базовим варіантом без розбиття. Отримані результати підтвердили, що відсутність партиціювання призводить до різкого зниження продуктивності зі зростанням обсягів даних, особливо у випадку OLAP-запитів, що потребують опрацювання великих масивів інформації. Партиціювання за діапазоном продемонструвало найвищу ефективність у виконанні аналітичних завдань, однак виявило обмежену продуктивність у транзакційних сценаріях. Натомість циклічний підхід забезпечив стабільність більшості OLTP-операцій завдяки рівномірному розподілу даних між партиціями, проте поступався за швидкодією під час виконання аналітичних запитів. Метод партиціювання за списком продемонстрував покращення у випадках із чітко визначеними категоріями, а хеш-партиціювання забезпечило стабільну продуктивність за довільного розподілу ключів. Найзбалансованішим рішенням виявився запропонований комбінований метод (range + list), який забезпечив зменшення тривалості виконання аналітичних запитів на 30–40 % без істотних втрат ефективності під час транзакційних операцій. Підтверджено доцільність використання комбінованого партиціювання як універсального підходу, що гарантує високу ефективність опрацювання великих даних в інформаційних системах. Запропонований метод перспективний в умовах зростання обсягів даних і різнотипних сценаріїв використання, забезпечує адаптивність до сучасних вимог цифрових середовищ.

Список використаних літературних джерел

[1] S. Ponnusamy and P. Gupta, "Scalable Data Partitioning Techniques for Distributed Data Processing in Cloud Environments: A Review", in IEEE Access, Vol. 12, pp. 26735–26746, 2024. DOI: 10.1109/ACCESS. 2024.3365810.

[2] H. Song, W. Zhou, H. Cui, X. Peng, and F. Li, "A survey on hybrid transactional and analytical processing", VLDB J., Vol. 33, No. 5, pp. 1485–1515, 2024. DOI: 10.1007/s00778-024-00858-9

[3] D. Corral-Plaza, I. Medina-Bulo, G. Ortiz, and J. Boubeta-Puig, "A stream processing architecture for heterogeneous data sources in the Internet of Things", Comput. Stand. Interfaces, Vol. 70, No. 103426, p. 103426, 2020. DOI:10.1016/j.csi.2020.103426

[4] P.-J. Liu, C.-P. Li, and H. Chen, "Enhancing storage efficiency and performance: A survey of data partitioning techniques", J. Comput. Sci. Technol., Vol. 39, No. 2, pp. 346–368, 2024. DOI:10.1007/s11390-024-3538-1

COMBINED DATA PARTITIONING METHOD FOR BIG DATA IN INFORMATION SYSTEMS

Volodymyr Solohub, Mykola Beshley

Lviv Polytechnic National University, 12, S. Bandery str., Lviv, 79013, Ukraine

In today's environment of rapidly growing data volumes, information systems must not only provide storage and access to massive datasets but also maintain stable performance when handling diverse query types. A critical challenge lies in balancing the efficiency of analytical (OLAP) operations with the responsiveness of transactional (OLTP) processes. Traditional approaches to data organization in relational DBMS often lose effectiveness in large-scale environments, resulting in longer processing times, reduced flexibility, and greater complexity in database management. This underscores the importance of developing new partitioning optimization methods capable of ensuring both high performance and scalability in hybrid information systems. This article investigates existing data partitioning methods in information systems designed to handle large volumes of structured information while simultaneously serving OLAP and OLTP workloads. The mechanisms of table partitioning in modern DBMSs are analyzed, and the strengths and limitations of each approach are identified with respect to performance, scalability, and ease of data management. A combined partitioning method (range + list) is proposed, tailored for hybrid information systems that concurrently process analytical and transactional workloads. Unlike traditional approaches, the proposed method not only applies combined partitioning for analytical tasks but also provides a comprehensive evaluation of its impact on query performance and transaction processing speed. The results demonstrate that the developed method achieves a balance between OLAP and OLTP performance, enhances scalability and flexibility of information systems, and can be considered a universal approach to managing large-scale data. To conduct the study, a unified simulation model of data processing was built using a star schema with a fact sales table, supporting both business analytics queries and transactional CRUD operations. Experimental findings confirm that the combined partitioning approach reduces analytical query execution time by 30–40 % without significant degradation of CRUD performance, making it an effective tool for improving the performance of large-scale information systems.

Keywords: *Data partitioning, OLAP, OLTP, information system, databases, scalability, big data.*