



## ДОСЛІДЖЕННЯ АЛГОРИТМІВ ПАРАЛЕЛЬНОГО ОПРАЦЮВАННЯ ІНФОРМАЦІЇ В БАЗАХ ДАНИХ

**М. Климаш, О. Гордійчук-Бублівська, І. Чайковський, Т. Данильченко**

*Національний університет “Львівська політехніка”, вул. С. Бандери, 12, Львів, 79013, Україна*

Відповідальний за рукопис: О. Гордійчук-Бублівська (e-mail: olena.v.hordiichuk-bublivska@lpnu.ua).

*(Подано 28 травня 2021)*

У роботі досліджено питання зменшення часу оброблення інформації в базах даних. Для швидкого пошуку та аналізу запитів запропоновано використовувати розподілені бази даних, в яких інформація розподіляється і зберігається на декількох пристроях. Для взаємозв'язку всіх даних та швидкого пошуку застосовується метод колонкових індексів, у якому враховано подібність даних та передбачено можливість знаходження інформації за ключем, навіть якщо вона міститься розподілено на різних пристроях. Такий підхід спрощує проблеми пошуку великих обсягів інформації в базах даних і дає можливість ефективніше опрацювати користувацькі запити.

**Ключові слова:** бази даних; розподілені системи; паралельні алгоритми.

**УДК:** 621.126

### 1. Вступ

Для сучасного світу проблема збирання та опрацювання інформації стає дедалі актуальнішою. Безліч програм та додатків, якими користуються мільйони людей, надсилають та отримують дані з різних баз даних, зберігають важливу інформацію, аналізують тощо. Щоб забезпечити необхідну якість обслуговування та підтримувати діяльність складних інформаційних систем, постійно застосовують нові алгоритми обробки даних, методи оптимального зберігання великих обсягів інформації. Важливим елементом у структурі інформаційних систем є бази даних. В них зберігається вся інформація, якою потім користуються інші пристрої та програми. Для оброблення великих обсягів даних база даних, що складається з одного, навіть дуже потужного обчислювального пристрою, неефективна. Необхідно застосовувати розподілені бази даних, що складаються із декількох пристроїв зберігання інформації. Можливе використання як декількох фізичних пристроїв, розташованих розподілено на певній території, так і хмарних ресурсів, призначених опрацювати запити користувачів. Для розподілу даних між різними серверами необхідно застосовувати алгоритми, що дають змогу розділяти великі таблиці даних, проте зберігати взаємозв'язок між ними, щоб була можливість швидко знаходити необхідну інформацію. Паралельні методи й алгоритми роботи з базами даних визначають основні правила розподілу інформації та можливості одночасного доступу до неї для декількох користувачів. Такий підхід дає змогу збільшити можливості наявних баз даних для оброблення великої кількості запитів, гнучко організувати керування системою та за необхідності збільшувати ресурси, додаючи нові пристрої. Таблиці для

зберігання даних у разі паралельного оброблення запитів мають спільні ключі для зв'язаних елементів, що дає змогу пошуковим алгоритмам ідентифікувати необхідну інформацію. Перевагою паралельних алгоритмів є надійність системи.

## 2. Аналіз та постановка задачі

*Визначення поняття великих даних в інформаційних системах.* Big Data або великими даними називають великий обсяг даних, отриманих із різних джерел, що повинен бути опрацьований та наданий користувачам для використання у зручній для них формі. Великими даними є зібрані відомості про користувачів інформаційних корпоративних систем, про їхні запити, історію покупок, уподобань тощо.

Також можна подати Big Data як сукупність технологій, що вирішують такі завдання:

- опрацьовують порівняно великі обсяги даних;
- можуть забезпечувати необхідну якість обробки за умови, що кількість даних постійно стрімко збільшується;
- опрацьовують та аналізують із однаковою ефективністю різні за структурою та форматом типи даних.

Для ефективної роботи інформаційних систем необхідно опрацьовувати запити користувачів з максимальною ефективністю. Наприклад, для системи аналізу даних про клієнтів мережі торгових центрів потрібно забезпечити можливість надійно зберігати відомості про покупців, особливості купівлі товарів, замовлення послуг. Також слід організувати швидкий доступ до необхідної інформації у разі нового запиту від клієнта. Раніше зібрану інформацію аналізують спеціальні алгоритми, визначаючи найімовірніші цікаві товари. У випадку, коли інформація опрацьовується повільно, інформаційна система не встигне швидко надати персоналізовані рекомендації користувачам, які, не знайшовши товарів чи послуг, через декілька секунд залишають ресурс та продовжують пошуки в іншому місці.

Також великі дані використовують для аналізу втраченого прибутку в маркетингу. Збирають інформацію про ненадані послуги через недоступність сервісів, недостатню оперативність під час надання рекомендацій конкретним клієнтам. Тому в маркетинговій сфері від ефективності аналізу Big Data прямо залежить прибуток.

Для систем безпеки швидкість і надійність опрацювання інформації великих обсягів визначає ефективність забезпечення необхідних показників якості роботи. Тому вимоги до алгоритмів оброблення даних дуже високі та потребують постійного вдосконалення.

*Принципи роботи із великими даними.* Для ефективності роботи система оброблення великих даних повинна дотримуватися таких принципів:

- можливість збільшення масштабу системи. Кількість інформації збільшується з кожним днем, тому необхідно постійно додавати обчислювальні ресурси;
- локалізація даних. Для ефективної роботи доцільно опрацьовувати дані на тих пристроях, де вони зберігаються, не витрачаючи часу на їх передавання по розподіленій системі іншим пристроям;
- надійність – в системі повинна бути передбачена можливість, у разі виходу із ладу одного чи декількох пристроїв, забезпечити подальшу ефективну роботу.

Для дотримання цих принципів необхідно використовувати розподілену інформаційну систему з організованим управлінням. В розподілених системах всі дані розподіляються для опрацювання між декількома вузлами. Це дає змогу істотно підвищити швидкість роботи та надійність інформаційних систем. Керує розподілом та збиранням даних вузол управління.

*Оброблення та зберігання великих даних. Підходи і технології опрацювання великих даних:*

Модель розподілених обчислень у кластерах комп'ютерів MapReduce. Згідно із цією моделлю кожне завдання розподіляється порівну між всіма вузлами в кластері, а після опрацювання на керуючому вузлі об'єднуються всі результати.

Hadoop – технологія, що дає змогу реалізувати модель MapReduce. Містить набори бібліотек та фреймворків, що забезпечують паралельне виконання програм на безлічі розподілених пристроїв;

Spark – технологія, що працює подібно до Hadoop, проте на обчислювальних пристроях всі дані записуються не в постійну, а в оперативну пам'ять, що істотно пришвидшує процес зчитування і запису та підвищує ефективність роботи.

NoSQL (Not Only SQL) – не стосується певної технології, визначає загалом поняття нереляційних баз даних. В реляційних базах, придатних для швидкого опрацювання невеликих і простих запитів, доволі складно і неефективно організувати доступ складними розподіленими алгоритмами. Тому пропонують інші структури для роботи з великими обсягами інформації.

*Основні поняття паралельної обробки даних.* Для паралельного опрацювання інформації її розподіляють на менші частини, які можна незалежно обчислювати на різних пристроях. Це істотно пришвидшує процес оброблення даних. У базах даних розподілення інформації означає поділ їх на менші частини, що можуть зберігатися у різних місцях.

Такий підхід дає змогу не тільки пришвидшити оброблення запитів, а й забезпечити можливість одночасного доступу до інформації. У випадку, коли дані зосереджені на одному пристрої в повному обсязі, необхідно очікувати завершення попереднього запиту, що мати змогу надати доступ до них наступному користувачеві. За розподілу ресурсів інформація розміщена частинами на різних серверах. Це дає змогу організувати роботу із запитом гнучко та ефективно.

Також розподілені сховища даних часто дублюють інформацію на декількох пристроях. Завдяки цьому в системі можна отримати доступ до даних одночасно за декількома запитом та підвищити надійність їх зберігання.

Для опрацювання даних в інформаційних системах важливим елементом є бази даних. Звичайні нерозподілені сховища не можуть ефективно обробляти великі обсяги запитів, тому доцільно їх організувати розподілено.

Розподіленими базами даних називають декілька баз даних, що є фізично розподіленими в інформаційній мережі та взаємозв'язаними. Для управління розподіленою базою даних необхідно використовувати розподілену систему управління базами даних. Це програмна система, що дає змогу забезпечити зрозуміле та ефективне функціонування усіх елементів бази даних.

В архітектурі системи баз даних, що призначені для опрацювання великої кількості запитів, необхідно виділити певні особливості. По-перше, в системі є два види вузлів. Це вузли приймання користувачьких запитів та вузли зберігання даних. На вузлах, що зберігають інформацію, передбачені функції зберігання даних. На вузлах оброблення запитів є тільки функції ефективного пошуку та доступу до необхідних даних.

По-друге, кожен вузол є незалежним обчислювальним пристроєм, тому має операційну систему, постійну та оперативну пам'ять, різне спеціальне програмне забезпечення. На різних вузлах можуть бути встановлені як однакові програми й операційні системи, так і різні.

Для організації роботи із великими даними їх розподіляють на менші частини та зберігають на різних пристроях.

Структурну схему системи бази даних, що опрацьовує великі обсяги інформації, зображено на рис. 1.

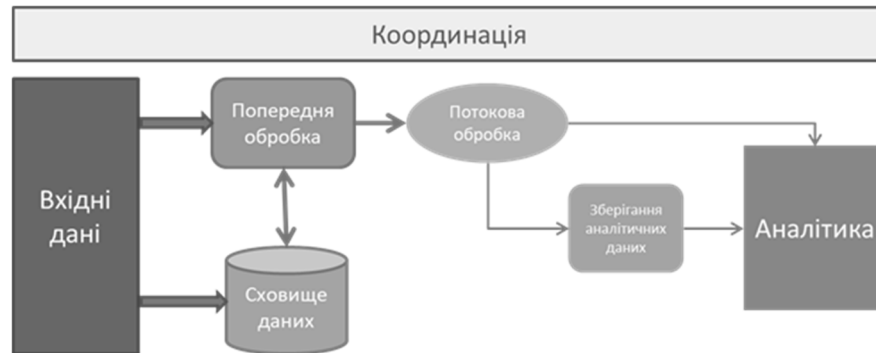


Рис. 1. Схема бази даних зі зберіганням Big Data

*Розподілені та паралельні бази даних.* Ці поняття взаємопов'язані й важко визначити відмінності між ними. Архітектура бази даних, в якій немає ресурсів, що спільно використовуються, досить подібна на структуру слабкозв'язаних розподілених систем.

Для того, щоб визначити особливості розподілених та паралельних систем управління базами даних, визначають такі характеристики:

- Розподілені системи баз даних – не те саме, що просто файли, які зберігаються на фізично віддалених пристроях. Бази даних характеризуються логічним взаємозв'язком інформації в системі, централізованим управлінням серверами, організованим розподілом ресурсів, структуризованим опрацюванням запитів від користувачів.
- Системи управління розподілених або паралельних баз даних виконують весь набір функцій оброблення даних. Це опрацювання запитів, транзакцій, робота алгоритмів пошуку, аналізу, структуризації, зберігання інформації.
- Розподіл даних відбувається непомітно для користувачів розподілених або паралельних систем управління базами даних. Ця властивість називається прозорістю системи та є однією із найважливіших для забезпечення ефективної обробки великих даних.

Прозорість складніше забезпечити в розподілених базах даних, ніж у паралельних, оскільки:

- Паралельні багатопроцесорні системи працюють під керівництвом однієї операційної системи. Ця операційна система може мати деякі функції управління системою бази даних та забезпечувати, відповідно, певну прозорість у розподілі інформації.
- Для організації паралельних обчислень користуються паралельними мовами програмування. Це також дає змогу використовувати вбудовані механізми забезпечення прозорості.

Найважливішим завданням розподілених та паралельних систем управління базами даних є підвищення ефективності їхньої роботи. Досягають цього використанням різних методів, алгоритмів, технологій організування роботи бази даних.

Зокрема, використовуючи операційні системи, призначені для роботи з розподіленими базами даних, значно простіше реалізувати роботу з розподілом і зберіганням інформації, визначити інші особливості фізичного рівня системи.

Для збільшення ефективності комунікації між вузлами в системі використовують спеціальні протоколи обміну повідомленнями. Вони дають змогу швидко надсилати й отримувати необхідну інформацію у розподіленій системі. Також використовується спеціальна оптимізація структури баз даних, за якої на вузлі, де розміщена інформація, розташовані й методи її опрацювання. Це дає змогу уникнути непотрібного передавання даних між різними вузлами та прискорити їх опрацювання.

Розподілені бази даних також істотно підвищують надійність роботи із великими даними. Оскільки використовуються механізми реплікації, інформація з одного пристрою зберігається також на декількох інших. Тому в разі виходу з ладу одного вузла системи дані не втрачаються.

### 3. Алгоритми паралельного опрацювання інформації в розподілених базах даних

Для розподілених баз даних необхідно забезпечити розташування інформації на різних вузлах та зручний доступ до неї у будь-який час.

Відомі два методи розподілу таблиць у базах даних:

- Розташування окремих таблиць на декількох вузлах.
- Розподіл однієї таблиці між декількома вузлами.

Розташування окремих таблиць на різних пристроях є зрозумілим і простим методом роботи системи баз даних. Далі розглянемо детальніше розподіл однієї таблиці між різними пристроями оброблення даних [1].

Розподілена/паралельна база даних – це саме база даних, а не “колекція” файлів, індивідуально збережених на різних вузлах мережі. У цьому полягає відмінність між DDB і розподіленою файловою системою. Система характеризується повною функціональністю СУБД, вона не зводиться за можливостями ні до розподілених файлових систем, ні до систем оброблення транзакцій. Обробка транзакцій – тільки одна з функцій, що надають такі системи. Поряд з цим вони повинні також забезпечувати функції запитів і структурної організації даних, які не обов’язково підтримують системи обробки транзакцій. Розподіл (ураховуючи фрагментацію і реплікацію) даних по безлічі вузлів невидимий для користувачів. Ця властивість називається прозорістю [2].

Для того, щоб досягти максимальної ефективності в організації розподілених чи паралельних обчислень, застосовують гібридну архітектуру систем обробки даних. У ній не тільки центральний процесор займається розподілом задач на підзадачі. Керуючий вузол здійснює тільки загальне розбиття, а вже обчислювальні пристрої виконують подальше розпаралелювання задач із урахуванням особливостей конкретних задач і алгоритмів.

Для організації гібридних паралельних обчислень визначають такі рівні паралелізму:

- Паралелізм на рівні кластера – вся обчислювальна система розділена на кластери, що складаються із обчислювальних пристроїв. Головна задача розділяється між кластерами, де здійснюється аналіз отриманих даних і подальший розподіл для обробки між декількома процесорами.
- Паралелізм на рівні вузла – спочатку аналізують, чи окремі вузли здатні виконувати обробку, потім задачі розподіляють між вибраними пристроями відповідно до їхніх обчислювальних ресурсів.

Для підвищення швидкості виконання обчислювальних алгоритмів у розподілених системах також застосовують графічні процесори. Вони дають змогу опрацювати та аналізувати різноманітні графічні дані. Для цього можна використати особливість спільного доступу пам’яті для всіх ядер графічного процесора. Такий підхід дає змогу значно швидше і ефективніше здійснити розпаралелювання обчислень та зменшити час на обмін службовими повідомленнями між обчислювальними процесами [3].

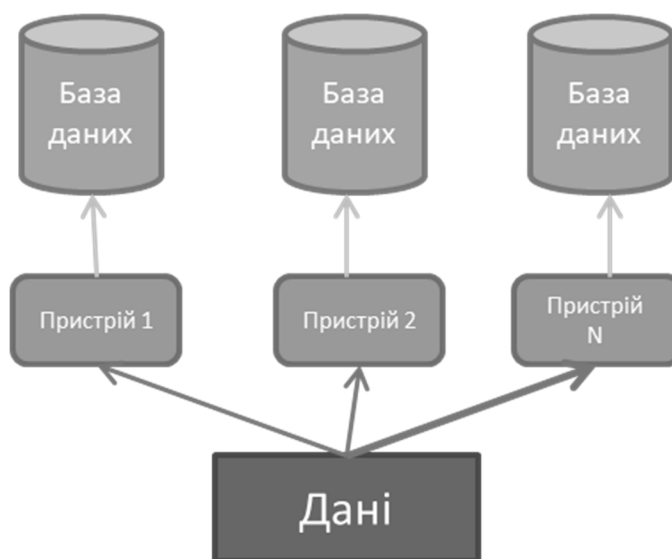


Рис. 2. Паралельна обробка в базах даних

*Федеративне навчання для оброблення інформації в базах даних.* Для ефективної роботи з базами даних доцільно використовувати різноманітні методи машинного навчання, що дають змогу виконувати інтелектуальне опрацювання інформації, визначати і відкидати несуттєву, встановлювати статистичні закономірності.

Проте класичні методи машинного навчання не завжди ефективні в розподілених базах даних, оскільки в такій архітектурі є декілька пристроїв, що займаються опрацюванням інформації. Тому запропоновано застосовувати методи федеративного навчання [4].

Федеративне навчання передбачає створення локальних навчальних моделей на кожному обчислювальному пристрої. Ці моделі працюють із локальними даними. Після отримання результатів навчання локальної моделі їх надсилають на центральний вузол, де утворюється глобальна навчальна модель. Під час роботи системи вузли постійно надсилають нові дані на центральний вузол та оновлюють глобальну модель. Такий підхід уможливило глобальне машинне навчання з великою кількістю навчальних даних, що зосереджені на безлічі віддалених вузлів.

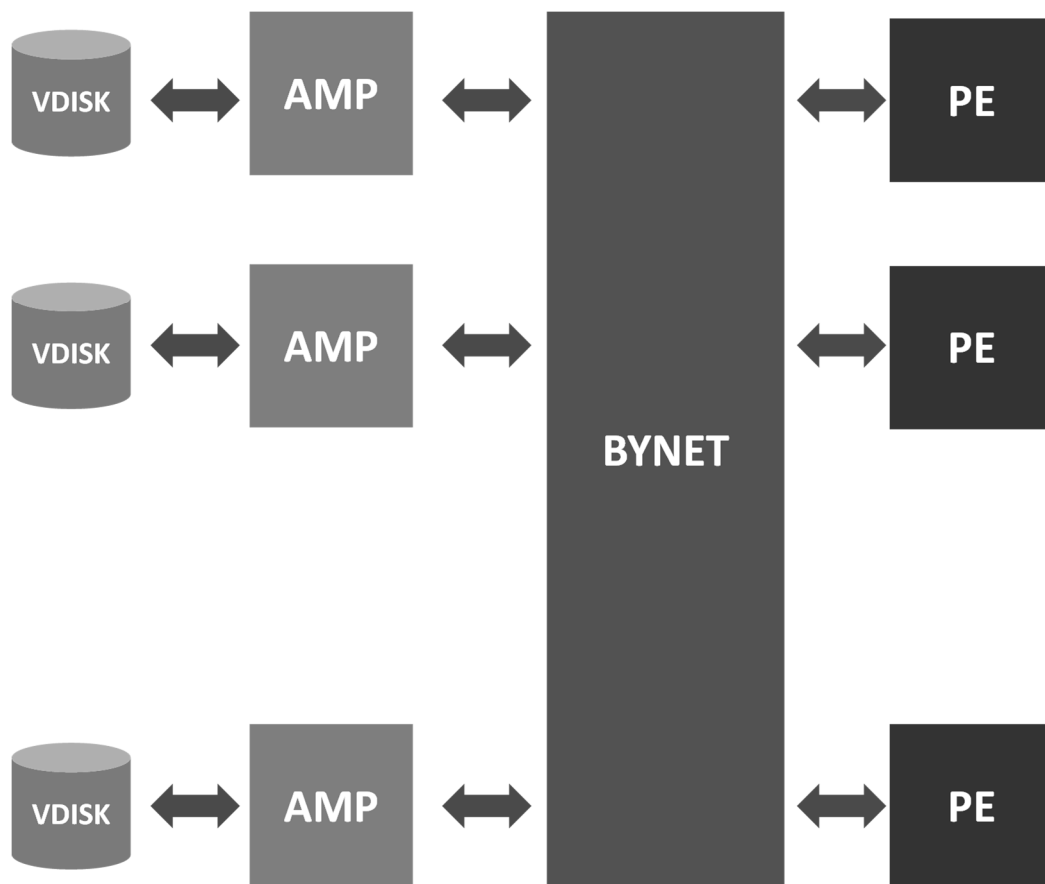


Рис. 3. Архітектура бази даних Teradata SQL Engine Shared-Nothing

На рис. 3 подано архітектуру баз даних Teradata SQL Engine Shared-Nothing. Це архітектура, яку пропонують використовувати для розподіленої бази даних із високим трафіком [5–8]. Вона складається з:

- Віртуального багатопотокового сервера PE (Parsing Engine), що займається встановленням та контролем сесій, забезпеченням безпеки та попереднім аналізом даних.
- AMP-процесор оброблення даних, займається опрацюванням задач аналізу та впорядкування інформації (Access Module Processor).
- VDISK – сховище даних для процесора оброблення інформації (Virtual Disk).
- BYNET – комунікаційне середовище для різних елементів архітектури.

Для застосування федеративного навчання у розподіленій базі даних запропоновано вдосконалити архітектуру Teradata SQL Engine, як показано на рис. 4. Отже, кожен процесор тренуватиме локальний набір даних згідно із власними потребами, а потім збиратимуть глобальний результат.

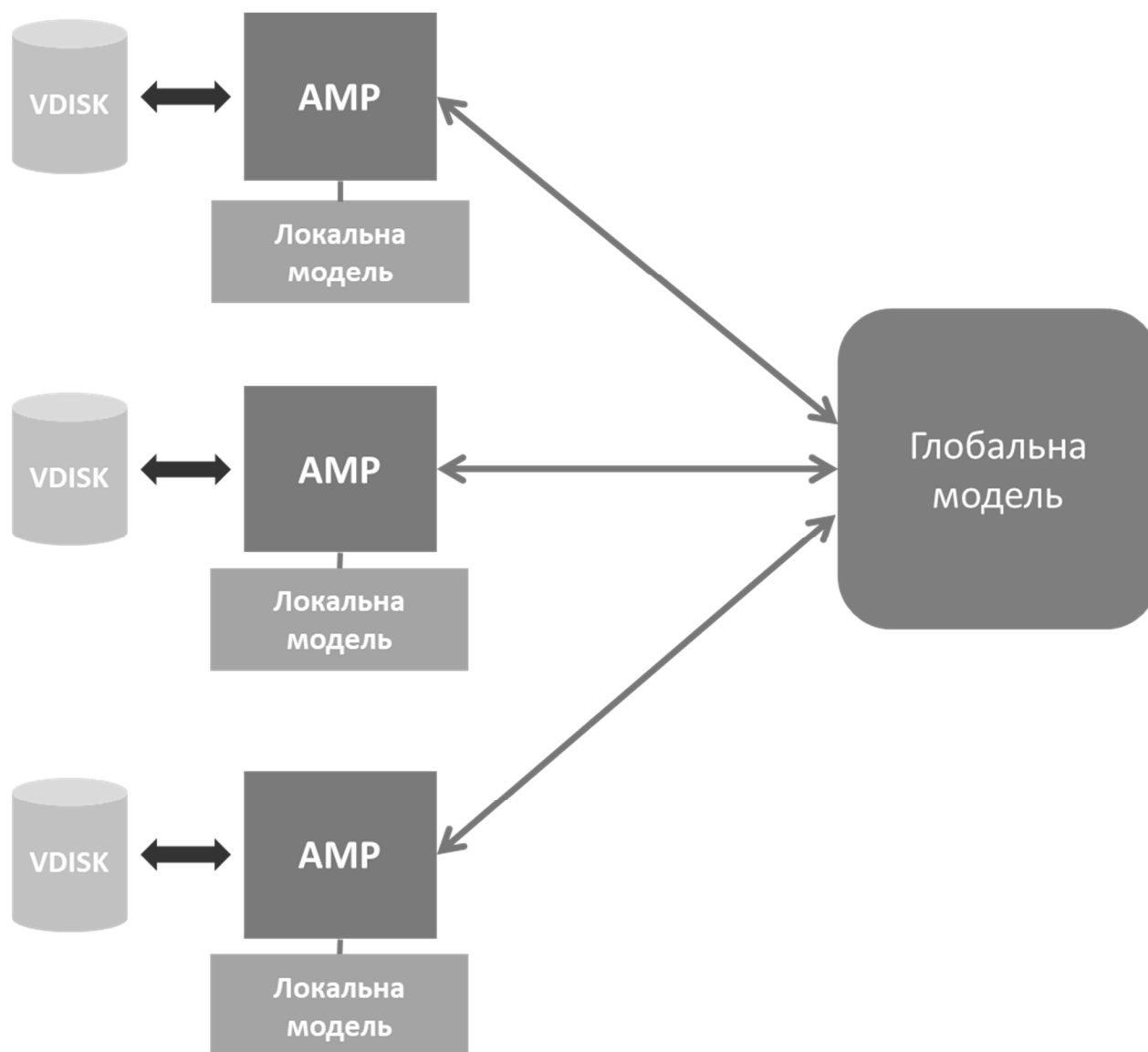


Рис. 4. Організація федеративного навчання на основі архітектури бази даних Teradata SQL Engine Shared-Nothing

#### 4. Дослідження та моделювання опрацювання інформації в розподілених базах даних

Найпопулярніші алгоритми управління одночасним доступом ґрунтовані на механізмі блокувань. У таких схемах щоразу, коли транзакція намагається отримати доступ до будь-якої одиниці пам'яті (як правило, сторінки), на цю одиницю накладається блокування в одному з режимів – спільному (shared) або монопольному (exclusive). Блокування накладаються відповідно до правил сумісності блокувань, що виключають конфлікти читання–запис, запис–читання і запис–запис.

Технології розподілених і паралельних СУБД досягли того рівня розвитку, коли на ринку вже є доволі розвинені й надійні комерційні системи. Водночас залишається низка питань, які ще очікують на вирішення [9–12].

Приклад архітектури розподіленої бази даних зображено на рис. 5.

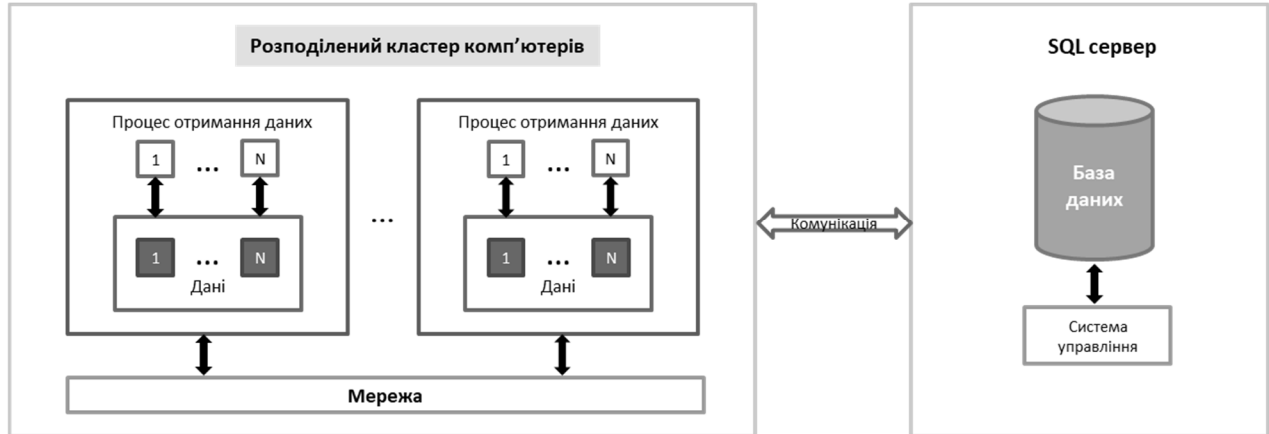


Рис. 5. Приклад архітектури розподіленої бази даних

Як видно з рис. 5, дані в таблицях розміщені на різних вузлах, що дає змогу одночасно доступитися до них із різних точок.

Для формування розподілених баз даних необхідно здійснити модифікацію таблиць записів. Запропоновано застосовувати метод колонкових індексів для розміщення даних (рис. 6).



Рис. 6. Колонкові індекси

На рис. 6 показано організацію розподілених таблиць у базах даних. За допомогою такого методу можна розподілити дані по різних таблицях, але зберігати зв'язок між ними за допомогою ключів та знаходити в потрібний момент.

На рис. 7 зображено фрагментацію і сегментацію записів у базі даних. Таблиці великих розмірів розподіляють на менші для зручнішої роботи.



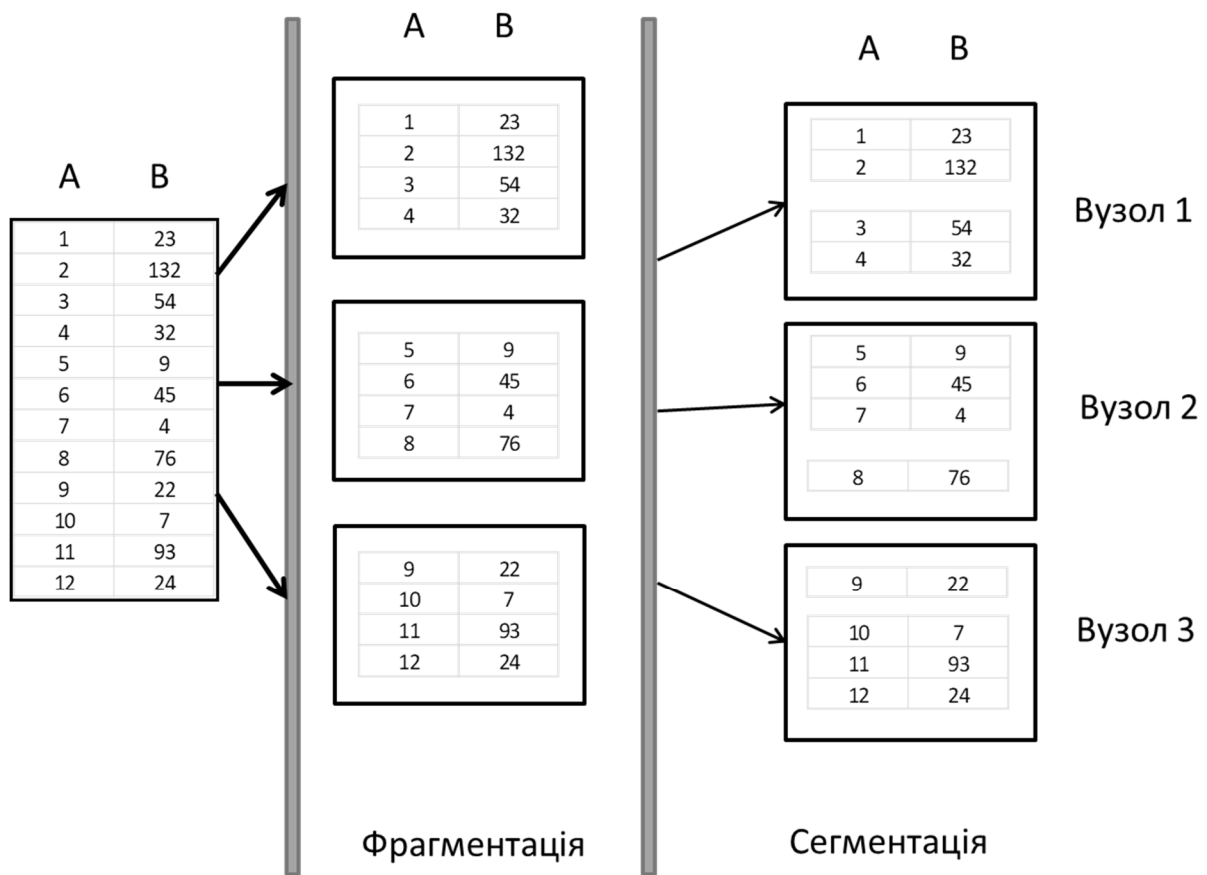


Рис. 7. Фрагментація записів у базі даних

На рис. 8 зображено приклад доступу до розподіленої бази даних.

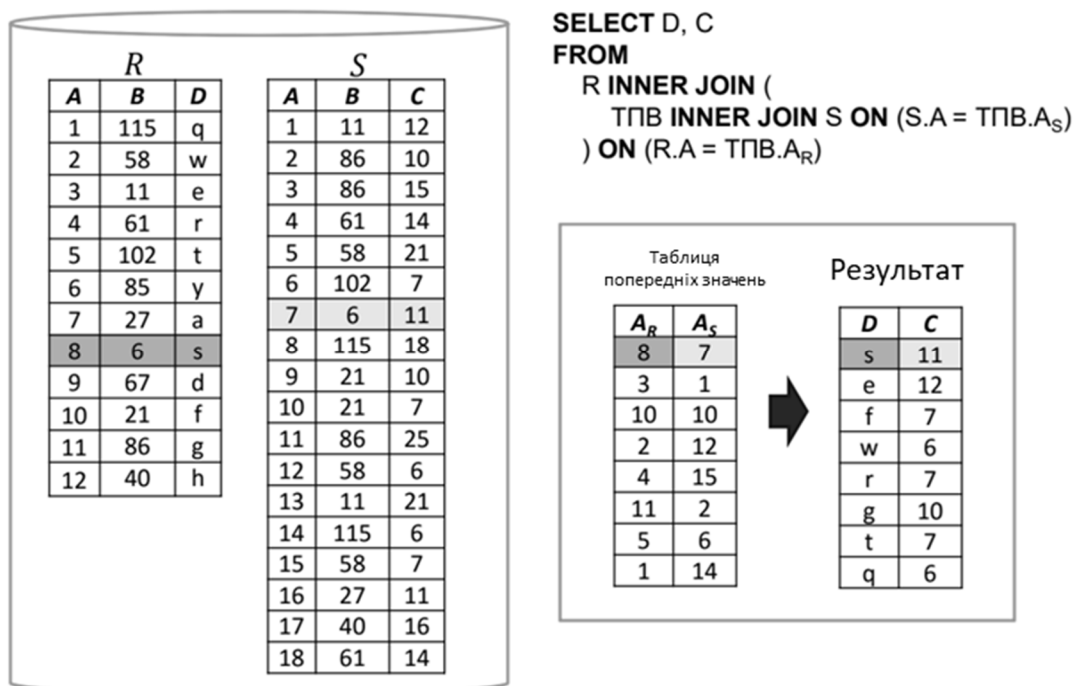


Рис. 8. Приклад пошуку в розподіленій базі даних

Необхідно вибрати дані із ключем  $B = 6$  з двох таблиць ( $R$  і  $S$ ). В подібних таблицях дані розташовуються на різних вузлах та групуються за певними ознаками, проте їх зручно отримати за допомогою ключів.

Такий підхід до організації баз даних дозволяє підвищити надійність роботи, зменшити розмір баз даних, пришвидшити доступ до інформації, особливо за значного її обсягу.

Для перевірки ефективності використання паралельної обробки запитів у SQL-серверах створено програмну модель на мові Java, що забезпечує вигреш у часі для модифікованих алгоритмів роботи з базами даних (рис. 9).

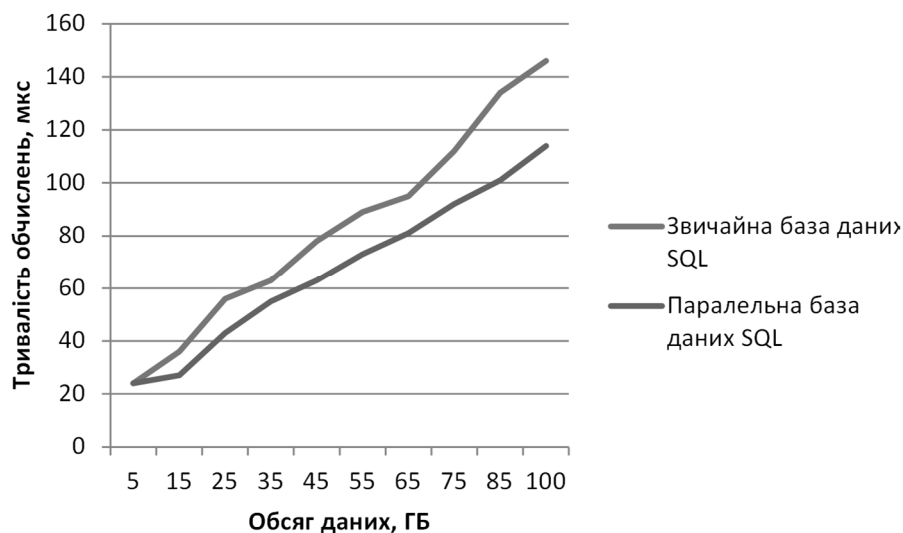


Рис. 9. Порівняння швидкості доступу до даних у звичайному та паралельному SQL серверах

Отже, паралельна обробка інформації в базах даних дає можливість підвищити ефективність роботи та поліпшити якість обслуговування.

Для підвищення ефективності обчислень можна збільшувати кількість вузлів у розподіленій базі даних (рис. 10).

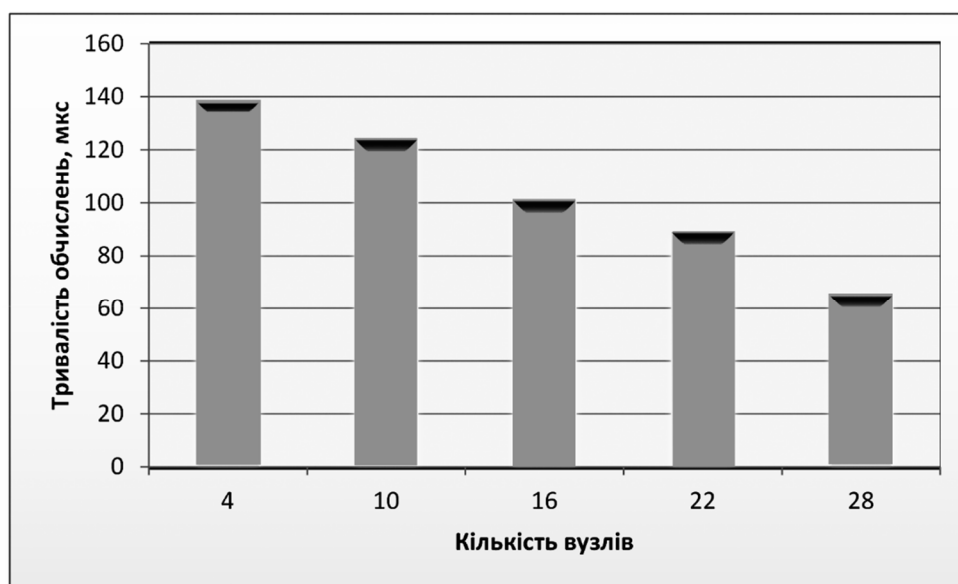


Рис. 10. Порівняння швидкості доступу до даних у паралельному SQL сервері залежно від кількості вузлів

На рис. 10 показано закономірне зменшення часу обчислень зі збільшенням кількості вузлів, на яких розташовані дані.

Вибір архітектури паралельної бази даних залежить від вимог до складності системи та швидкості обчислень. Детальніше залежність ефективності роботи паралельних баз даних від методів розподілу обчислювальних даних розглянемо у наших подальших дослідженнях.

## Висновки

Через постійне збільшення обсягу даних, які необхідно обробляти та зберігати, традиційні бази даних, де інформація зберігається на одному пристрої, працюють неефективно. Тому використовують розподілені бази даних, у яких інформація розподіляється між декількома базами даних та пов'язана спеціальними ключами доступу. Такий підхід дає змогу підвищити ефективність обробки даних великих обсягів. Для опрацювання великого обсягу інформації істотно значення мають бази даних. В роботі виконано порівняння роботи звичайних SQL баз даних та паралельних. Запропоновано використовувати модифіковані паралельні сервери в базах даних, оскільки вони дають змогу істотно зменшити тривалість опрацювання інформації. Досліджено методи фрагментації та сегментації у базах даних для розподілу інформації на різних вузлах. Це дає змогу отримувати доступ до даних одночасно декільком користувачам та підвищити надійність їх зберігання. Порівняно швидкість доступу до даних у звичайному та паралельному SQL серверах. Отримані результати свідчать про більшу ефективність паралельних баз даних. Також визначено, що зі збільшенням кількості вузлів, у яких зберігаються дані, в паралельній базі даних тривалість обчислень зменшується. Це визначає доцільність подальших досліджень у цій галузі.

## Список використаних джерел

- [1] F. Ortega, and A. González-Prieto, "Recommender systems and collaborative filtering", *Appl. Sci.*, vol. 10, 7050, 2020.
- [2] Z. Wang, H. Wu, Z. Jiang, P. Ju, J. Yang, Z. Zhou, and X. Chen, "Singular value decomposition-based load indexes for load profiles clustering", *Transmission Distribution IET Generation*, vol. 14, issue 19, pp. 4164–4172, 2020.
- [3] M. Khan, Y. Jin, M. Li, Y. Xiang, and C. Jiang, "Hadoop performance modeling for job estimation and resource provisioning", *IEEE Transactions on Parallel and Distributed Systems*, no. 27, issue 2, pp. 441–454, 2016.
- [4] V. Yeromenko, and O. Kochan, "The conditional least squares method for thermocouples error modeling", in *Proc. IEEE Conference IDAACS 2013. Berlin, Germany, 2013*, pp. 157–163.
- [5] K. Sridharan, G. Komarasamy, and S. D. M. Raja, "Hadoop framework for efficient sentiment classification using trees", *IET Networks*, vol. 9, issue 5, pp. 223–228, 2020.
- [6] Z. Hu, D. Li, and D. Guo, "Balance resource allocation for spark jobs based on prediction of the optimal resource", *Tsinghua Science and Technology*, vol. 25, issue 4, pp. 487–497, 2020.
- [7] V. Iannino, C. Mocci, M. Vannocci, V. Colla, A. Caputo, and F. Ferraris, "An event-driven agent-based simulation model for industrial processes", *Appl. Sci.*, vol. 10, pp. 4343, 2020.
- [8] T. Zhao, and Z. Ding, "Distributed agent consensus-based optimal resource management for microgrids", *IEEE Transactions on Sustainable Energy*, no. 9, issue 1, pp. 443–452, 2018.
- [9] M. Beshley, N. Kryvinska, M. Seliuchenko, H. Beshley, E. M. Shakshuki, and A.-U.-H. Yasar, "End-to-End QoS "smart queue" management algorithms and traffic prioritization mechanisms for narrow-band internet of things services in 4g/5g networks", *Sensors*, vol. 20, pp. 2324, 2020.
- [10] M. Klymash, M. Beshley, and B. Stryhaluk, "System for increasing quality of service of multimedia data in convergent networks", in *Proc. Problems of Infocommunications Science and Technology, Kharkiv, Ukraine, 2014*, pp. 63–66.
- [11] V. Romanchuk, M. Beshley, A. Polishuk, and M. Seliuchenko, "Method for processing multiservice traffic in network node based on adaptive management of buffer resource", in *Proc. TCSET-2018, Slavske, Ukraine, 2018*; pp. 1118–1122.

- [12] S. Jun, K. Przystupa, M. Beshley, O. Kochan, H. Beshley, M. Klymash, J. Wang, and D. Pieniak, A cost-efficient software based router and traffic generator for simulation and testing of IP network. *Electronics*, vol. 9, pp. 40, 2020.

## INVESTIGATION OF PARALLEL ALGORITHMS FOR INFORMATION PROCESSING IN DATABASES

**M. Klymash, O. Hordiichuk-Bublivska, I. Tchaikovskiy, T. Danylchenko**  
*Lviv Polytechnic National University, 12, S. Bandery Str., Lviv, 79013, Ukraine*

The paper has been devoted to the problem of reducing the time of information processing in databases. It is suggested to use distributed databases for quick search and analysis of queries. In them the information is distributed and stored on several devices. For the interconnection of all data and quick search, it is proposed to use the method of column indexes, which takes into account the similarity of data and provides the ability to find information by key, even if it is distributed on different devices. This approach simplifies the problem of finding large amounts of information in databases

**Key words:** *databases; distributed systems; parallel algorithms.*