

Д. Зубарева

СУЧАСНІ МЕТОДИ АДМІНІСТРУВАННЯ БАЗ ДАНИХ

D. Zubarieva

MODERN METHODS OF DATABASE ADMINISTRATION

Сучасні технології БД є одним із визначних факторів вдосконалення процесів зберігання та доступу до великих масивів даних. Потужність сучасних БД базується на результатах досліджень та технологічних розробках, які отримані протягом кількох останніх десятиріч. Управління БД виконується спеціалізованими програмами (СУБД), які є ефективним інструментом підтримки БД. Особливим видом БД є розподілені БД, важливими характеристиками яких є ступінь паралелізму операцій та швидкого пошуку даних. Слід зауважити, що технології розподіленої обробки інформації ускладнюють всі компоненти і спонукають переглядати їх структури аж до базових елементів. Водночас у багатьох розподілених БД витрати на підтримку телекомунікацій можуть переважити вартість обчислювальних процесів та програмно-технічного забезпечення.

Тому при дослідженні та проектуванні розподілених БД найважливішу роль відіграють задачі компактного кодування даних та швидкості обміну даними, при заданій пропускній здатності каналного зв'язку. Отже, задача вдосконалення архітектури та методів кодування інформації в розподілених БД є актуальною (В. Шаряк).

Перші спроби створити системи управління базами даних були здійснені компанією IBM у 1960-х рр. Вона була фактичним монополістом ринку з впровадження новітніх технологій. У той період часу компанія орієнтувалася на ринок мейнфреймів та програмного забезпечення для них. Математики компанії розробили мережеву модель бази даних, а згодом ієрархічну модель. До сьогодні найпоширенішою організацією баз даних є реляційна структура бази даних. СУБД на основі реляційної теорії витіснили ієрархічні та мережеві СУБД, оскільки надавали інтуїтивно зрозумілий механізм збереження даних на основі таблиць. Відразу велика кількість компаній обрала саме цей вид організації баз даних. Нині більш популярним методом масштабування є горизонтальний. Суть цього методу полягає в тому, що кілька серверів паралельно працюють і розділяють між собою запити користувачів. У цьому випадку сервери можна додавати відповідно до збільшення кількості користувачів, але є одна велика проблема — відсутність синхронізації даних між серверами баз даних. На початку 2000-х рр. у Каліфорнійському університеті в Берклі Ерік Брюер запропонував принцип, який до сьогодні відомий під назвою CAP-теорема. Сьогодні до CAP-теореми входять наявні СУБД: MS SQL Server, MySQL, Oracle, Postgres, Aster Data, Greenplum, Vertica, BigTable, Hypertable, Hbase, MongoDB, Terrastore, Scalaris, Berkeley BD, MemcacheDB, Redis, Dynamo, Vodemort, Tokyo Cabinet, KAI, Cassandra, SimpleDB, CouchDB, Riak.

Абревіатура CAP складається з трьох характеристик: узгодженість (consistency), доступність (availability), стійкість до розділення (partition tolerance). Абревіатура CAP отримала свою назву саме від перших літер цих трьох характеристик. Ця теорема стверджує, що жодна СУБД не може одночасно виконувати всі три вище перераховані характеристики. Розробники повинні йти на компроміс при виборі СУБД. У 2000-х рр. розпочався новий етап розвитку баз даних. Кілька компаній почали розробляти СУБД не на основі реляційної моделі. У 2008 р. почали набувати

популярності так звані NoSQL бази даних. NoSQL означає “not only SQL” — «не тільки SQL». Саме 2008 р. вважається стартом розвитку моделей баз даних, відмінних від традиційної реляційної моделі, яка домінувала довгі роки на ринку.

Безперечно, найпопулярнішою NoSQL системою керування базами даних є MongoDB. Вона стала популярною завдяки зручності використання в Інтернет-проектах. Наприклад, газета “New York Times” використовує саме MongoDB для свого сайту. Слід зазначити, що MongoDB є документоорієнтованою, що дає певні переваги перед реляційною базою даних.

Окремо треба виокремити продукт PostgreSQL. Ця СУБД не пропонує настільки багато можливостей, як Microsoft SQL Server. Тим не менш, завдяки можливості безкоштовно використовувати, ця СУБД останніми роками є особливо популярною. Крім того, PostgreSQL дає змогу розробникам створювати нові типи даних і розробляти компоненти. Для прикладу, існує розроблена компонента для PostgreSQL, яка дає можливість використовувати відеокарти компанії Nvidia для обробки даних. Обчислення проводяться за допомогою технології CUDA. З одного боку, можливість написання власних компонентів дає багато можливостей для програмістів, а з іншого боку — підтримувати ці компоненти іноді дуже складно. Нема гарантій, що в наступній версії продукту буде підтримка компонентів із попередніх версій.

Компанія Oracle також не відставала від Microsoft, купила продукт MySQL, що дало їй додаткові можливості для розвитку. Багато технологій компанії Microsoft мають аналоги в продуктах компанії Oracle. Окремим напрямом розвитку стали хмарні технології. Вони дозволяють зручно передати всю роботу з адміністрування і лише зробити заявку на масштабування обчислювальних потужностей. Саме в таких умовах почали розвиватися CosmosDB від Microsoft та DynamoDB від компанії Amazon. Фактично Azure Cosmos DB має у своєму складі різні моделі бази даних, що дозволяє розробникам більш гнучко обирати схему баз даних. Amazon DynamoDB представляє собою NoSQL базу даних типу «ключ-значення».

Після проведення аналізу розвитку сучасних СУБД можна виснувати, що на сьогодні всі вендори намагаються зробити не просто СУБД, а системи роботи з великими масивами даних. Сучасні СУБД дають змогу проводити аналіз даних, будувати сховища даних, використовувати отримані дані для прогнозування та навчання нейронних мереж. Усе це, безперечно, призведе до ще більшого наукового прогресу. Огляд наявних СУБД засвідчив, що вони можуть по-різному працювати з різними об'єктами й різними функціями та засобами. Показано, що більшість СУБД ґрунтується на єдиному сталому комплексі основних понять. Це дає можливість виділити одну (дві) системи й узагальнити її поняття, прийоми й методи на весь клас СУБД.

Отже, вивчення середовища баз даних, на нашу думку, краще поділити на дві частини: спочатку слід навчити студентів працювати з популярною реляційною СУБД для розуміння основних понять реляційної теорії. Далі можна вивчати можливості NoSQL баз даних. До таких систем, як показав аналіз, належать MongoDB, Neo4J, Redis, HBase.