УДК 004.89:[004.65:004.032.2]

СХЕМА ВИБОРУ ВАРІАНТА СХОВИЩА ДАНИХ ДЛЯ ВИСОКОНАВАНТАЖЕНИХ СИСТЕМ

Курач А. І.
Науковий керівник – к.т.н., доц. Вишняк М.Ю.
Харківський національний університет радіоелектроніки, каф. СТ
м. Харків, Україна
тел.: +38(099) 426-74-16, email: anna.kurach@nure.ua

This work is devoted to defining modern data storage options for highly loaded systems. Relational (SQL) and non-relational (NoSQL) databases were considered. CAP theorem is used as a major indication for choosing storage type. It states that distributed system can deliver only two of three desired characteristics: consistency, availability, and partition tolerance. SQL databases are suited for storing structured data, not in partitioned fashion. NoSQL databases show better performance considering partitioning, replication, fault-tolerance, on-demand scalability and storing semi-structured and non-structured data.

На сьогоднішній день дані генеруються та споживаються в безпрецедентних масштабах. Однак різноманітність існуючих систем перешкоджає обґрунтованому вибору технології зберігання даних. Вибір сховища відіграє вирішальну роль у забезпеченні якісної роботи системи, а отже потребує системного підходу та детального розбору існуючих рішень. Обираючи сховище даних, слід звернути увагу на такі властивості системи, як кількість можливих одночасних користувачів, суворість вимог до безпеки даних, можливість пожертвувати продуктивністю системи взамін на її доступність, а також необхідність масштабування сховища в майбутньому, аналізу збережених даних та рівень його складності, інтеграція сховища даних з іншими рішеннями тощо.

Реляційна база даних (БД) – тип сховища даних, що організує дані в строгі таблиці, пов’язані одна з одною. Реляційна БД зазвичай масштабується вертикально, тобто дані зберігаються на одному сервері, а масштабування здійснюється шляхом додавання додаткової потужності до цього сервера. Зазвичай цей тип БД втілює в собі властивості ACID: атомарність, узгодженість, ізоляцію та довговічність [1].

Нереляційна база даних – це база даних, що не базується на концепції таблиць, та використовує різні моделі даних для зберігання, керування та доступу до даних. NoSQL бази даних дозволяють зберігати неструктуровані дані, що можна змінювати без прив’язки до чіткої схеми. NoSQLбази можна запускати на кількох серверах, тому їх масштабування дешевше та легше, ніж масштабування реляційних БД. А оскільки бази даних NoSQL не покладаються на один сервер, вони більш відмовостійкі. Тож якщо один компонент виходить з ладу, база даних може

продовжувати роботу. NoSQL системи можна згрупувати у категорії – стовпчик, граф, документ, та сховища типу ключ-значення. Вони мають різний спосіб зберігання та надання доступу до даних. Прикладами нереляційних БД можуть слугувати MongoDB, Redis, Apache HBase, Apache Cassandra тощо. Іншою визначальною властивістю бази даних є забезпечений рівень узгодженості. Деякі БД створено для гарантування надійної узгодженості та серіалізації, що притаманно саме реляційним БД, тоді як інші БД віддають перевагу доступності. Згідно з теоремою CAP системи баз даних можна класифікувати за їхніми властивостями, яких не може бути більше двох одночасно – за узгодженістю (C), доступністю (A) та відмовостійкості при розділенні в мережі (P) [2].

З точки зору CAP, звичайні SQL системи дотримуються тільки CA властивостей, бо працюють в режимі одного сервера: уся система стає недоступною у разі відмови. І навпаки, системи NoSQL, такі як Dynamo, BigTable або Cassandra, розроблені для обсягів даних і запитів, які неможливо обробити на одній машині, і тому вони працюють на кластерах, що складаються з тисяч серверів. Тож класифікація систем NoSQL як AP, CP або CA широко прийнята як засіб для високорівневих порівнянь.

На базі всебічного аналізу сучасних варіантів сховищ даних та наявних потреб користувачів розроблена і обговорюється логіко-лексична схема вибору найбільш прийнятного варіанту. Схема представлена набором правил продукцій, що еволюціонує.

Вибір системи сховища даних для високонавантажених систем означає вибір одного набору бажаних властивостей над іншим. Якщо основна пам’ять одного серверу може вмістити всі дані, система з одним вузлом, як-от Redis (CP), є найкращим вибором. Якщо обсяг даних перевищує ємність оперативної пам’яті, система з кількома вузлами, яка масштабується горизонтально, може бути більш доцільною. Потрібно вирішити, чи віддати перевагу доступності (AP) чи узгодженості (CP). Cassandra може забезпечити постійну роботу, тоді як HBase, MongoDB і DocumentDB, забезпечують надійну узгодженість. Якщо сховище даних повинно підтримувати набагато складніші запити, ніж простий пошук, потрібно визначити, чи буде необхідно подальше масштабування. Для обробки транзакційних даних підходять реляційні бази, або графові бази даних, такі як Neo4J, оскільки вони підтримують семантику ACID. Якщо доступність має суттєве значення, краще використовувати розподілені NoSQL системи, такі як MongoDB та DocumentDB.

Список використаних джерел:

1. Kleppmann, M. (2017). Designing Data-Intensive Applications: The Big Ideas Behind Reliable, Scalable, and Maintainable Systems. O'Reilly Media.

2. What is the CAP theorem? | IBM. (б. д.). IBM – Deutschland | IBM. https://www.ibm.com/topics/cap-theorem