УДК 004.75.05

О.М. Тарасюк, К.П. Таранова, А.В. Горбенко

Національний аерокосмічний університет імені М.Є. Жуковського «ХАІ», Харків

АНАЛІЗ ОСОБЛИВОСТЕЙ ТА КЛАСИФІКАЦІЯ СИСТЕМ УПРАВЛІННЯ НЕРЕЛЯЦІЙНИМИ БАЗАМИ ДАНИХ

У статті дана загальна характеристика найбільш поширених типів нереляційних баз даних, а також виконана їхня класифікація за типами моделі даних, що використовуються, та нефункціональними характеристиками, базовими з яких є узгодженість даних, доступність та стійкість до розподілу.

***Ключові слова**: NoSQL, нереляційні бази даних, моделі даних, класифікація, теорема CAP.*

Вступ

Нереляційні, так званні NoSQL [1], бази даних дістали значного розвитку та поширення впродовж кількох останніх років при створенні розподілених інформаційних систем, які повинні зберігати та швидко обробляти великі обсяги інформації. Найбільш активно зараз вони використовуються розробниками Інтернет-додатків, зокрема соціальних мереж, в системах Internet-of-Things, пошукових сервісах, таких як Google або Yandex, тощо. У сучасному розумінні термін NoSQL був запропонований Джоеном Оскарссоном та Еріком Евансом на початку 2009 року [2] для позначення все більшого числа нереляційних, розподілених сховищ даних, які досить часто не намагаються забезпечити атомарність, узгодженість, ізоляцію і довговічність.

Системи NoSQL також називають «Не Тільки (Not Only) SQL» [1], щоб підкреслити, що вони є альтернативою SQL. NoSQL бази даних використовують більш гнучкі моделі даних у порівнянні з традиційними реляційними базами. Перевагами такого підходу є простота архітектури, горизонтальне масштабування і більш точний контроль над доступністю даних. Нереляційні бази даних переважно зберігають інформацію у вигляді пар ключ-значення, що надає швидку обробку операцій пошуку і додавання нових записів, та в результаті призводить до значного виграшу у продуктивності з точки зору часу очікування та пропускної здатності. Однак, платою за легкість горизонтального масштабування, гнучкість моделі даних та велику швидкість виконання операцій читання/запису/оновлення є неможливість гарантії так званих ACID-властивостей (**A**tomicity, **C**onsistency, **I**solation, **D**urability), притаманних традиційним реляційним базам даних [3].

Таким чином, **метою статті** є аналіз особливостей сучасних NoSQL баз даних, їхня класифікація та вироблення рекомендації щодо використання тої або іншої нереляційної бази даних в залежності від особливостей та задач, що постають перед розробниками сучасних розподілених інформаційних систем.

Класифікація нереляційних баз даних на основі моделі даних

Існують різні підходи до класифікації нереляційних баз даних NoSQL, наприклад запропоновані у [2]. Через різні підходи щодо визначення нефункціональних вимог і набору виконуваних функцій, досить важко зробити єдиний вірний огляд ринку нереляційних база даних. Проте, більшість погоджується, що найголовніша класифікація базується на основі використаної моделі даних.

Основними моделями даних, що використовуються нереляційними базами даних є такі [4]:

– *документо-орієнтовані (Document):* Couchbase, MongoDB;

– *графові (Graph):* Neo4J, Allegro, Virtuoso;

– *типу ключ-значення (Key-value):* Dynamo, Riak, Redis, Cache, Project Voldemort;

– *об'єктно-орієнтовані (Object-oriented):* db4o, GemStone/S, InterSystems Caché, JADE, NeoDatis ODB, ObjectDB, Objectivity/DB, ObjectStore, ODABA, OpenLink Virtuoso, Versant Object Database, WakandaDB, ZODB;

– *стовпчикові (Column-oriented):* Hbase, Accumulo, Cassandra.

1. Документо-орієнтована модель даних. Центральне поняття нереляційних баз даних цього типу – «документ». Хоча кожна реалізація документо-орієнтованої бази даних відрізняється деталями цього визначення, в загальному, всі вони погоджуються, що документи інкапсулюють і кодують дані в деяких стандартних форматах або кодуваннях [5].

Кодування відбувається з використанням однієї з наступних технологій: XML, YAML або JSON, а також бінарні форми, такі як BSON, PDF і документи Microsoft Office (MS Word, Excel, і так далі).

Різні реалізації пропонують різні способи організації та/або групування документів:

– збірники (Collections);

– ключові слова (Tags);

– невидимі метадані (Non-visible Metadata);

– ієрархія каталогів (Directory hierarchies).

У порівнянні з реляційними базами даних, наприклад, збірники можна розглядати як таблиці, а документи можуть розглядатися в якості записів. Однак є значна відмінність: кожен запис в реляційній таблиці має ту ж послідовність полів, в той час як документи в нереляційній колекції можуть мати поля, які повністю відрізняються.

Документи розрізняються в базі даних за допомогою унікального ключа, який представляє цей документ. Однак крім простого пошуку по ключу документів, бази даних такого типу мають спеціальні API або мову запитів, яка дозволяє отримувати документи на основі їх змісту.

Однак деякі документо-орієнтовані NoSQL [6] бази реалізують альтернативний спосіб отримання інформації з використанням методів MapReduce.

2. Графова модель даних. Графові бази даних зберігають дані, які можна представити у вигляді графу. Прикладами таких даних є опис суспільних відносин, громадського транспорту, дорожніх карт або мережевих топологій [5].

Бази даних цього типу використовують структури даних, наприклад RDF (Resource Description Framework), оптимізовані для представлення вузлів графу та зв'язків між ними, а також виконання операцій та алгоритмів на графах (наприклад, пошук найкоротшим маршрутів, тощо).

3. Модель даних типу ключ-значення. Модель даних типу ключ-значення (key/value) дозволяє зберігати дані у найбільш простому вигляді. Увесь кортеж даних (вся строчка у термінах реляційних баз даних, або ж увесь документ чи об'єкт) зберігаються цілком у вигляді послідовності байт у полі «value». Доступ до кожного такого полю виконується за допомогою унікального ключу «key». Нереляційні бази даних цього типу не підтримують API, що дозволяють виконувати доступ до окремих елементів/атрибутів, що зберігаються у полі «value». Усі данні повинні бути прочитанні/записані цілком, але ж підтримується можливість серіалізації/де серіалізації будь-якої структури даних або об'єкту мови програмування у послідовність бітів та навпаки.

Однією з найбільш поширених баз даних типу «ключ-значення» є Couchbase. Ця БД з'явилася за рахунок об'єднанням проектів CouchDB і Membase, та є послідовником розподіленої системи кешування memcached, від якої успадкувала сумісність на рівні протоколу доступу і пріоритет зберігання даних в ОЗУ.

4. Стовпчикова модель даних. Дані в базах цього типу зберігаються у вигляді сімейств стовпчиків, об'єднаних в сімейства ключів.

На відміну від реляційної моделі даних, кількість колонок від рядка до рядка може бути змінна, а загальне число колонок може вимірюватися мільярдами. Також кожен рядок такої структури має унікальний ключ. Можна розглядати таку модель даних як хеш-таблицю хеш-таблиці, першим ключем якої є ключ рядку, а другим – ім'я колонки.

Стовпчикова модель даних часто розглядається в якості більш складного підкласу моделі даних типу «ключ-значення». Найбільш поширеною операцією, яка підтримується стовпчиковими базами даних є додавання нових колонок з відповідними значеннями до існуючих строк.

Однією з найбільш відомих нереляційних стовпчикових баз даних є СУБД Cassandra, яка була розроблена Facebook і передана для подальшого розвитку до співтовариства Apache в 2008 році. Є ідейним продовженням пропрієтарної бази даних Google BigTable.

Cassandra – єдина БД, операції запису в якій працюють швидше, ніж операції читання [7]. Це пояснюється тим, що запис успішно завершується (в найшвидшому варіанті) відразу ж після фіксації результату в таблиці Memtable, яка зберігається в оперативній пам'яті комп'ютера, та реєстрації операції запису у журналі транзакцій Commit-log на диску (для забезпечення довговічності). Оскільки журнал транзакцій дозволяє тільки дописувати данні у кінець, швидкість запису Cassandra фактично обмежена швидкістю послідовного запису на жорсткий диск. У той же час операція читання вимагає перевірок, кількох читань з жорсткого диску, вибору найсвіжішого запису, що є досить повільним. На сьогодні Cassandra – це надійний і досить швидкий масштабований архів даних.

Крім Apache Cassandra до баз даних стовпчикового типу відносяться: Google BigTable, Hadoop HBase, Druid, Hypertable, KAI, KDI, OpenNeptune, Qbase.

5. Об'єктно-орієнтована модель даних. Об'єктно-орієнтована СУБД – це система, що дозволяє створювати, зберігати і використовувати інформацію в формі об'єктів. Зазвичай об'єктно-орієнтована СУБД забезпечує також об'єктно-орієнтований інтерфейс взаємодії з користувачем.

Найбільш широке застосування об'єктно-орієнтовані бази даних знайшли в таких областях, як системи автоматизованого конструювання/виробництва (CAD/CAM), системи автоматизованої розробки програмного забезпечення (CASE), системи управління складовими документами – в областях не цілком традиційних для баз даних.

Низка американських компаній – Autotrol Technology, STEP Tools, DEC та інші – використовують об'єктно-орієнтовані СУБД (наприклад, Object Store виробництва компанії Object Design) для роботи зі складно організованими даними, відповідними стандарту STEP (Standart of Exchange of Product Model Data – Стандарт обміну даними моделей продуктів).

До найбільш популярних об'єктно-орієнтованих баз даних відносяться: db4o, GemStone/S, Inter Systems Caché, JADE, ObjectDatabase++, ObjectDB, Objectivity/DB, ObjectStore, ODABA, Perst, OpenLink Virtuoso, Versant Object Database, ZODB.

Порівняльна характеристика різних типів нереляційних та реляційних баз даних наведена у табл. 1.

Таблиця 1

Характеристика нефункціональних властивостей баз даних

Характеристика / Тип	Продуктивність (Performance)	Масштабованість (Scalability)	Гнучкість (Flexibility)	Складність (Complexity)	Функціональність (Functionality)
Key-Value Stores	висока	висока	висока	дуже низька	мінімальна
Column Store	висока	висока	помірна	низька	помірна
Document Store	висока	помірна	висока	низька	помірна
Graph Database	помірна	помірна	висока	висока	теорія графів
Relational Database	помірна	помірна	низька	помірна	реляційна алгебра

Висновки

У статті розглянуто особливості сучасних нереляційних баз даних NoSQL. Їхня особливість полягає у підтримки гнучких моделей даних, легкості горизонтального масштабування, що дозволяє створювати розподілені сховища даних для зберігання дуже великих обсягів інформації. Однак, зворотною стороною є відсутність підтримки гнучкої мови структурних запитів SQL, а також неможливість одночасно гарантувати високу готовність та цілісність даних.

Таким чином, можна зробити висновок, що NoSQL бази даних не є повноцінною заміною традиційних реляційних СУБД, таких як Oracle, MySQL чи MS SQL. Вони повинні застосовуватися для вирішення специфічних задач зберігання та обробки інформації, з урахуванням найбільш пріоритетних характеристик, таких як висока швидкість доступу к даним та можливість горизонтального масштабування. Одночасно з тим, для вирішення задач, які потребують суворого дотриманні цілісності даних, а також можливості виконання гнучких аналітичних запитів SQL без необхідності змінювати структуру бази даних повинні використовуватися реляційні СУБД.

Список літератури

1. S. George. NoSQL – NOT ONLY SQL [Текст] / S. George // International Journal of Enterprise Computing and Business Systems ISSN, 2013. – 11p.

2. A Yes For a NoSQL Taxonomy [Електронний ресурс] // High Scalability. – Режим доступу: http://highscalability.com/blog/2009/11/5/a-yes-for-a-nosql-taxonomy.html.

3. Haerder, T. Principles of transaction-oriented database recovery [Текст] / T. Haerder, A.Reuter // ACM Computing Surveys. – 1983. – Vol 15(4). – P. 287 –317.

4. Обзор и сравнительный анализ систем управления нереляционными базами данных [Текст] / С.И. Лисин // Электронный журнал. Молодежный научно–технический вестник. – М.: ФГБОУ ВПО "МГТУ им. Н.Э.Баумана", 2013. – 12 с.

5. Tood Lipcon. Design Patterns for Distributed Non-Relational Databases [Текст] / Tood Lipcon. – Cloudera, 2009. – 48 p.

6. The NoSQL Alternative. [Електронний ресурс] // DATABASE. – Режим доступу: http://www.drdobbs.com/database/224900500.

7. Дмитрий Сиващенко. Распределённые хэш–таблицы на примере NoSQL СУБД Cassandra [Текст] / Дмитрий Сиващенко – М.: ДМК Пресс, 2010. – 45 с.

Надійшла до редколегії 24.04.2017

Рецензент: д-р техн. наук, проф. К.С. Козелкова, Державний університет телекомунікацій, Київ.

АНАЛИЗ ОСОБЕННОСТЕЙ И КЛАССИФИКАЦИЯ СИСТЕМ УПРАВЛЕНИЯ НЕРЕЛЯЦИОННЫМИ БАЗАМИ ДАННЫХ

О.М. Тарасюк, К.П. Таранова, А.В. Горбенко

В статье представлена общая характеристика наиболее распространённых типов нереляционных баз данных, а также выполнена их классификация по типу используемой модели данных и нефункциональным характеристикам, базовыми из которых являются согласованность данных, доступность и устойчивость к разделению.

Ключевые слова: *NoSQL, нереляционные базы данных, модели данных, классификация, теорема CAP.*

FEATURES ANALYSIS AND CLASSIFICATION ON NON-RELATIONAL DATA BASE CONTROL MANAGEMENT SYSTEMS

O.M. Tarasyuk, K.P. Taranova, A.V. Gorbenko

The general characteristic of the most common types of non-relational data bases has been given in the article. These data bases have also been classified according to their used data models as well as non-functional characteristics such as consistency, availability and partition tolerance.

Keywords: *NoSQL, non-relational data bases, data models, classification, CAP theorem.*