Basic Comparison of Relational vs. NoSQL Databases

Apo, Bella

Bellevue University, CSD310

Assignment 1.3

May 26, 2024

**In the context of relational databases, what are relationships? Provide an example.**

In relational databases, relationships are how the tables connect to one another and how their data correlates. There are several types of relationships: one-to-one, one-to-many/many-to-one, and many-to-many. Relationships are important to relational databases, being a major contributor to the organization of data, ensuring it is done in a structured, consistent, and efficient way (Adron, 2023). An online store database is an example of a one-to-many relationship between ‘Customers’ and ‘Orders’. One customer can place many different orders, but each order is only made by one customer. This structure helps to maintain the data’s integrity and simplifies the data retrieval.

**What are the advantages of relational databases? What are the advantages of NoSQL databases?**

A relational database is one that stores and organizes data into tables. These tables are structured in rows and columns, each representing a record or a piece of data. The advantages of these relational databases are that they are flexible and can be used in a multitude of applications. They are scalable, making them available for smaller or widescale projects. They are useful for ensuring data integrity and consistency as relational databases enforce atomicity, consistency, isolation, and durability. Relational databases do a great job at managing complex relationships between data and foreign keys, ensuring that there is accuracy across any related tables.

As for NoSQL databases, they are revered for their scalability. NoSQL is designed to scale horizontally, making it easy to distribute the data across multiple servers, even ones that handle high amounts of traffic. NoSQL databases are also flexible, allowing for changing data requirements. These databases have high performance rates, making them suitable for projects with big data processing and real-time web applications. In addition, it also allows for a variety of data models, making it applicable for different application needs.

**What are the disadvantages of relational databases? What are the disadvantages of NoSQL databases?**

While relational databases have many advantages, they also come with a few downsides. These disadvantages include complex schema, which can complicate handling evolving or unstructured data, and scalability limitations because of its vertical scaling, which necessitates complex horizontal scaling solutions. Complex queries might lead to performance issues, requiring optimization efforts to fix.

Meanwhile, NoSQL databases shine in scalability and flexibility; they struggle with consistency guarantees because of challenges in complex querying and require a learning curve because of different data models. In addition, consistency and data duplication may also be of concern. Organizations should carefully consider the pros and cons of these two database types to ensure they are picking the best option for their projects.

**Identify at least two features of MySQL and two features of MongoDB, and describe what they are and how they are used.**

MySQL is a database that supports data query, storage, and security. MySQL is compliant with ACID (atomicity, consistency, isolation, and durability) meaning it ensures reliable transactions and data integrity. This guarantees that the database transactions are processed reliably despite system failures. This is important for applications that require data consistency and reliability. Another feature of MySQL is replication, which allows the data to be replicated across several servers in real time. This enhances fault tolerance, scalability, and data availability by balancing redundancy and load.

MongoDB is is database for e-commerce and content serving sites that allows for scalability and flexibility. One feature of MongoDB is database triggers, which allow you to execute code when specific events trigger it within the database. These triggers can be scheduled to happen at specific times, or can occur when triggered by a script when you insert, update, or delete a document. These database triggers are important for data consistency and integrity. Another feature of MongoDB is replication. By replicating the database across multiple servers decreases the chance of failure or interruptions. Replication tries to solve these risks by using multiple servers for disaster recovery and backup, through horizontal scaling. MongoDB utilizes replication so if the primary server fails, there is a backup server that can be promoted to primary ensuring that data access is continued.

References

Adron, V. a. P. B. (2023, September 20). *Composite Thrashing Code Adron’s success in misadventures of thrashing coding and calamity! The Keys & Relationships of Relational Databases*. Composite Thrashing Code. <https://compositecode.blog/2023/09/18/the-keys-relationships-of-relational-databases/>

*Data Relationships in MongoDB | MongoDB Tutorial | StudyTonight*. (n.d.). <https://www.studytonight.com/mongodb/relationships-in-mongodb>

MongoDB. (n.d.). *MongoDB Features & Key Characteristics | MongoDB*. <https://www.mongodb.com/resources/products/capabilities/features>

*MySQL :: MySQL 8.0 Reference Manual :: 1.2.2 The main features of MySQL*. (n.d.). <https://dev.mysql.com/doc/refman/8.0/en/features.html>