James Buzzell

10/23/2025

Assignment 1.3

**Basic Comparison of Relational vs. NoSQL Databases**

In the context of relational databases, relationships are how the data contained in one table is connected to the data in another table. They create logical links between these tables allowing for their data to be organized efficiently and retrieved with purpose.

One type of relationship is a one-to-one. A one-to-one relationship indicates that a row contained in Table 1 is related to at least one row in Table 2, and vice versa. An example of this would be websites that require a login. Each user has one account, and each account belongs to one user.

Another type of relationship is a one-to-many. This type of relationship means that a row in Table 1 can relate to multiple rows in Table 2, but each row within Table 2 relates to only one row in Table 1. A great example of this is a customer being able to place multiple orders on Amazon. Each order belongs to that one customer.

**Advantages of relational databases:**

* They contain data that is structured and consistent.
* Applications that utilize these such as banking or inventory systems are guaranteed to have reliability.
* They are coded via SQL, a programming language that provides a standardized and declarative method to query and manipulate data.

**Advantages of NoSQL databases:**

* Exhibit high levels of scalability and performance. They are designed to scale horizontally across a decent number of servers
* They are great for unstructured data, and can handle JSON, XML, etc. easily.
* They read and write at fast rates and are optimized for quick access and updates

**Disadvantages of relational databases:**

* Scalability is limited. They scale vertically, thus add more CPU and RAM to a single server rather than horizontally where more servers are added.
* There is the potential for an excess of normalization. Normalization decreases the level of redundancy but increases the number of entities needed, which can cause queries to get complicated.
* It is not ideal for unstructured data. Handling documents, JSON files, etc. is cumbersome.

**Disadvantages of NoSQL databases:**

* They lack standardization. They aren’t created using a universal query language like SQL.
* Data can be redundant. An action like embedding a document, for example, can lead to duplication of data.
* There is a lack of support resources and tools for reporting, backup, and analytics.

One feature of MySQL is something called ACID compliance. ACID allows for reliable and consistent transactions. For example, if you are paying your credit card bill on a banking app, MySQL guarantees that the payment is successfully made without any data loss or corruption.

Another feature of MySQL is the ability to perform JOIN operations between related tables. You can actually combine data from multiple tables in a single query. For example, if you have a customers table and an orders one, you can use the JOIN query to get all orders and the names of the customers who ordered them.

One feature of MongoDB is that it stores data as documents in a flexible BSON (Binary JSON) format in opposed to rows and columns. Each document is able to have its own structure, making it perfect for unstructured data. This allows you to store all necessary data together in one document. There isn’t a need to join.

Another feature of MongoDB is horizontal scalability, also known as sharding. Sharding is when large collections are split across multiple servers (shards). This allows for better performance and storage distribution.