Database System

April 21, 2023

What is Database ?

- A structured collection of data organized, stored, and managed in a computer system designed to efficiently store, retrieve, and manipulate large amounts of data.
- It consists of one or more tables, which are made up of rows and columns.
  - Rows represent data records, and columns represent attributes or fields.
  - Each row is identified by a unique primary key.
- Used in various industries and applications for storing and managing data.
- Popular **database management systems** include MySQL, Microsoft SQL Server and Oracle Database, among others.

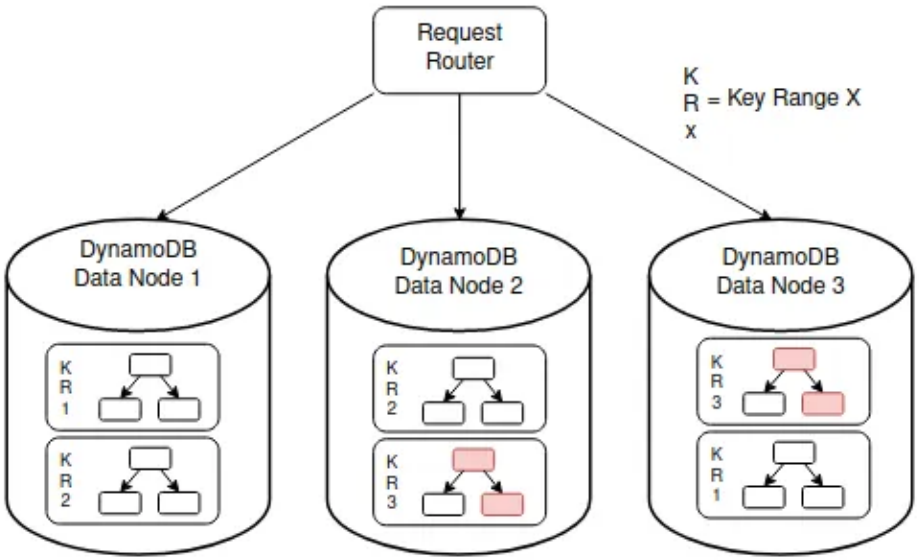
What is NoSQL ?

NoSQL (not only SQL) databases are designed for handling large-scale data processing and are suitable for unstructured, semi-structured, and rapidly changing data.

- They have a flexible and dynamic data model, allowing for storage of data without a fixed schema.
- NoSQL databases are scalable and high-performing, often supporting distributed and parallel processing.
- NoSQL databases are used in agile development environments, allowing for faster development cycles and schema evolution.
- NoSQL databases find applications in social media, e-commerce, gaming, IoT, analytics, and more.
- Popular examples of **NoSQL databases** include MongoDB, Cassandra, Couchbase, Amazon DynamoDB, and Redis.

How Merkle tree works in Amazon DynamoDB I

Let's consider a simple example using three with 3 (virtual) **DynamoDB nodes**, each holding 2 key ranges, each key range duplicated across two nodes.



file:data.png

How Merkle tree works in Amazon DynamoDB II

- In this type of a distributed system, there will always come a time when some of these virtual nodes are out of sync with other virtual nodes holding the same key-range. Dynamo DB uses **Merkle trees** to perform effective comparison and synchronization of the key-ranges in these nodes.
- **Merkle Trees at Each Replica:** Within each replica, Merkle trees are constructed for the data items where the leaf nodes of the tree are the key-range data values. The Merkle root summarizes the data in each node.

How Merkle tree works in Amazon DynamoDB III

- **Merkle Roots Comparison:** The Merkle roots of the Merkle trees in each replica are periodically compared. The Merkle root is a hash value computed from the hashes of all the data items in the replica.

  In the above figure, the KR3 key-range tree is shown partially in red to illustrate how the nodes would have diverged and need resolving to find correct values.

How Merkle tree works in Amazon DynamoDB IV

- **Consistency Checks:** By comparing the Merkle roots of each virtual node hosting the same key-range, divergence in nodes is immediately visible. If there is a difference, one can keep comparing the branches of the tree to efficiently find the exact divergent spots, and the data values to synchronize.If the Merkle roots do not match, it indicates that there is a discrepancy in the data items between the replicas.
- **Data Reconciliation:** When a discrepancy is detected, Amazon DynamoDB uses data reconciliation techniques to bring the data items in the different replicas back to consistency.

How Merkle tree works in Amazon DynamoDB V

- By using Merkle trees and periodic Merkle root comparisons, Amazon DynamoDB ensures that data is replicated and consistently managed across multiple replicas, helping to maintain data integrity and consistency in a distributed database environment.

Amazon DynamoDB

Amazon DynamoDB is a fully managed, key-value NoSQL database designed to run high-performance applications at any scale.

- DynamoDB offers built-in security, continuous backups, automated multi-Region replication, in-memory caching, and data import and export tools.
- The architecture of Dynamo DB is described in the AWS Dynamo DB paper from 2007, including its use of Merkle trees for efficient synchronization