ONE PAGE SUMMARY

IT & Software › Database › Sharding

Introduction to Sharding

Sharding: Scaling Databases for Performance and Efficiency

DEFINITION

> Sharding is a database architecture pattern that splits data across multiple servers. This improves performance and allows for better resource management.

KEY CONCEPTS

Horizontal Partitioning

Sharding involves dividing a database into smaller, more manageable pieces called shards.

Load Balancing

Distributing data across shards helps balance the load and improves response times.

Data Distribution

Data is distributed based on a sharding key, which determines where records are stored.

Scalability

Sharding allows databases to scale horizontally, adding more servers as needed.

Complexity

Implementing sharding can introduce complexity in data management and queries.

EXAMPLES

→ User accounts are split across multiple servers based on geographic location.

→ E-commerce products are divided into shards based on product categories.

→ Social media posts are distributed by user ID to different shards.

MEMORY TIPS

★ Think of sharding as slicing a pizza into equal pieces for easier sharing.

★ Remember 'SHARD' as 'Split, Handle, And Redistribute Data'.

★ Visualize shards as different shelves in a library, each holding specific books.

COMMON MISTAKES

✗ Over-sharding, leading to too many small shards and management issues.

✗ Neglecting to choose an effective sharding key, causing uneven data distribution.

✗ Failing to account for cross-shard queries, which can complicate data retrieval.

QUICK RECAP

Sharding is a technique that enhances database performance by distributing data across multiple servers. It improves scalability and load balancing but can add complexity to data management.