Design and Modeling with MySQL

ALKIN TEZUYYSAL
SCALE 21X Pasadena, CA Mar 2024
Let’s get connected with Alkin first

Alkin Tezuysal - EVP - Global Services @chistadata

- Linkedin: https://www.linkedin.com/in/askdba/

Open Source Database Evangelist

- Previously PlanetScale, Percona and Pythian as Technical Manager, SRE, DBA
- Previously Enterprise DBA, Informix, Oracle, DB2, SQL Server

@ask_dba
Recognitions

- Most Influential in Database Community 2022 - The Redgate 100
- MySQL Rockstar 2023 - Oracle (MySQL Community)
- Database Design and Modeling with PostgreSQL and MySQL 2024 - <Packt>

Born to Sail, Forced to Work!

@ChistaDATA Inc. 2024
About ChistaDATA Inc.

Founded in 2021 by Shiv Iyer - CEO and Principal

Has received 3M USD seed investment (2021)

Focusing on ClickHouse infrastructure engineering and performance operations

What’s ClickHouse anyway?

Services and Products around dedicated DBaaS, Managed Services, Support and Consulting

www.chistadata.io  www.chistadata.com

@ChistaDATA Inc. 2024
Maritime Trivia

WHAT'S THE NAME OF THIS DEVICE?
Sextant

Trivia Question: Sextants

The sextant is an astronomical instrument used to measure the angle between two visible objects, usually the horizon and a star. It was invented independently in the 18th century by John Hadley in England and Thomas Godfrey in America.
How to contribute to MySQL to community?

Born to Sail, Forced to Work!

@ChistaDATA Inc. 2024
Agenda

1. **Database design**
   - Design a schema, create ERDs, and apply normalization techniques

2. **Scaling databases**
   - Scale databases with sharding, replication, load balancing, and implement backup and recovery strategies

3. **Integrating databases**
   - Integrate databases with web apps, utilize SQL effectively and implement best practices

4. **Emerging trends**
   - Explore emerging trends, including NoSQL databases, cloud databases, and the impact of AI and machine learning
History of Data Models

1970
The network model was first formulated by the Database Task Group of the Conference on Data Systems Languages (CODASYL).

1980s
Relational databases became dominant in commercial data management applications.

1990s
Object databases emerged as an alternative to relational databases.

2000s
Graph databases gained popularity for modeling networked data.

2010s
Wide column stores, document databases, and other NoSQL systems offered alternatives to relational databases.

@ChistaDATA Inc. 2024
Database Design Models

Popularity percentages

- **Relational Model**: 60%
- **Hierarchical Model**: 20%
- **Network Model**: 10%
- **Object Oriented Model**: 5%
- **Graph Model**: 5%
Relational Model

DATA ORGANIZED INTO TABLES
Tables consist of rows and columns representing records and attributes

ROWS REPRESENT RECORDS
Each row represents an individual data entry

COLUMNS REPRESENT ATTRIBUTES
Each column represents a property or attribute of the data

KEYS LINK TABLES
Keys establish relationships between tables and ensure data integrity

THE RELATIONAL MODEL STRUCTURES DATA INTO INTERCONNECTED TABLES VIA KEYS
Normalization

**Normalization reduces data redundancy**
Normalization organizes data into multiple tables so each piece of information is stored only once.

**Normalization improves data integrity**
Normalization eliminates inconsistent data through techniques like removing duplicate data.

**There are different normal forms**
First Normal Form, Second Normal Form, and Third Normal Form have specific criteria for organizing data.

Normalization is key for reducing redundancy and improving data integrity in relational databases before applying SQL. 

@ChistaDATA Inc. 2024
ACID Properties

**TRANSACTIONS ARE ATOMIC**
Transactions are treated as a single unit that either completes fully or fails entirely.

**TRANSACTIONS MAINTAIN CONSISTENCY**
Transactions transform the database from one valid state to another, following predefined rules.

**TRANSACTIONS ARE ISOLATED**
Transactions are executed independently to prevent visibility of intermediate states.

**TRANSACTIONS ARE DURABLE**
Once committed, a transaction's effects are permanently preserved despite failures.

TOGETHER, THESE ACID PROPERTIES MAKE SQL DATABASES ROBUST AND RELIABLE FOR DEMANDING DATA INTEGRITY NEEDS.
CAP Theorem

**CAP Theorem** is about the trade-off between consistency, availability, and partition tolerance.

**Availability means every request gets a response.**
High availability systems aim to always be operational but can’t guarantee consistency.

**Partition tolerance means the system works despite network failures.**
A partitioned system can remain available but consistency is difficult to maintain.

**Consistency means all nodes see the same data at the same time.**
Strong consistency provides linearizability but reduces availability.

CAP Theorem states it’s impossible for a distributed system to provide strong consistency and high availability in presence of network partitions. Systems need to trade-off between these guarantees. 

@ChistaDATA Inc. 2024
CAP Theorem Visualized

AVAILABILITY

PARTITION TOLERANCE

CONSISTENCY
Data Models in NoSQL

**KEY-VALUE MODEL**
Stores data as key-value pairs, ideal for caching and simple data retrieval

**COLUMN-FAMILY MODEL**
Organizes data in column families, suitable for wide-column stores

**GRAPH MODEL**
Represents data as nodes and edges, making it suitable for complex relationships and graph-based operations

NOSQL DATABASES ADOPT DIFFERENT DATA MODELS LIKE KEY-VALUE, COLUMN-FAMILY, AND GRAPH MODELS TO STORE AND ORGANIZE DATA BASED ON THE USE CASE REQUIREMENTS.
CAP Theorem for NOSQL

CAP THEOREM PROPOSED BY ERIC BREWER
States consistency, availability and partition tolerance cannot be achieved simultaneously in distributed systems

MUST MAKE TRADEOFFS BETWEEN CAP PROPERTIES
Designers must choose which properties to optimize based on requirements and constraints

CAP THEOREM IS FUNDAMENTAL FOR NOSQL DATABASE DESIGN CHOICES INVOLVING DISTRIBUTED SYSTEMS
@ChistaDATA Inc. 2024
Key Terms and Data Models

**ENTITY**
An entity is a distinct object or concept represented in a database, like a person, place, thing or event.

**ATTRIBUTE**
Attributes are characteristics or properties of an entity that define specific pieces of information stored about it.

**RELATIONSHIP**
A relationship is a connection between entities defining how they are associated with each other.

**KEY TERMS LIKE ENTITY, ATTRIBUTE AND RELATIONSHIP ARE THE BUILDING BLOCKS OF DATA MODELS IN DATABASE DESIGN.**

@ChistaDATA Inc. 2024
Hierarchical Data Model

Hierarchical Model organizes data in parent-child relationships
Data is structured in a tree with each parent node having multiple children nodes and each child having only one parent

Relationships are unidirectional from parent to child
The relationship flows only from parent to child, a child cannot be parent of its own parent

Each parent can have multiple children
A parent node in the hierarchy can have multiple child nodes under it representing different branches

The hierarchical model allows organizing data in a top-down structure with clear relationships between parents and children, supporting multiple children under each parent.
Hierarchical Model
Network Model
Object Oriented Model
Object Relational Model

Activity Report
- Date
- Activity Code
- Duration

Object 1 instance
- 10-23-2023
- 621
- 30

Activity
- Customer
- Product
- Sales Unit
- Total

@ChistaDATA Inc. 2024
Relational Model
Let's focus on ER Model

**ERD - ENTITY RELATIONSHIP DIAGRAM**
ERD shows entities, attributes, relationships between entities

**NORMALIZATION**
Breaks data into multiple tables to avoid redundancy and anomalies

**KEYS**
Primary key uniquely identifies rows, foreign key links tables

DATABASE DESIGN INVOLVES USING MODELS LIKE ERD TO STRUCTURE DATA INTO NORMALIZED TABLES WITH PROPER KEYS.

@ChistaDATA Inc. 2024
Steps for ER Database Design

1. **Requirements Gathering**
   - Meet with stakeholders to understand the scope and requirements of the system.

2. **Entity Identification**
   - Look for nouns in the requirements that represent key entities in the system.

3. **Attribute Identification**
   - For each identified entity, determine the attributes or data points needed to describe it.

4. **Relationship Identification**
   - Determine how the identified entities relate to one another.

5. **Preliminary ERD Construction**
   - Construct an initial ERD diagram showing entities, attributes and relationships.

6. **Review and Refinement**
   - Verify the ERD against requirements and make refinements as needed.

7. **Normalization**
   - Break down tables to minimize data redundancy and anomalies.

@ChistaDATA Inc. 2024
MySQL Workbench

@ChistaDATA Inc. 2024
MySQL Workbench Features

- **MODELING**: Allows designing database models visually.
- **VALIDATION**: Validates models for issues before creating databases.
- **QUERY BUILDING**: Provides a visual query builder to construct complex queries.
- **ADMINISTRATION**: Allows administering MySQL servers and databases.
First Normal Form

**FIRST NORMAL FORM REQUIRES NO DUPLICATE ROWS**
Each row in a table should be unique with a primary key to identify it

**ONLY ATOMIC VALUES IN COLUMNS**
Columns should not contain sets, arrays or lists

FOLLOWING 1NF ENSURES DATA INTEGRITY AND AVOIDS UPDATE ANOMALIES.
Second Normal Form

A Table Must Be in 1NF to Qualify for 2NF
2NF requires the table to already satisfy all conditions for 1NF before applying additional criteria

No Partial Dependencies Allowed
All non-key attributes must depend fully on the entire primary key, not just part of it

Composite Keys Complicate 2NF
With a composite primary key, ensuring no partial dependencies on just part of the key is trickier

2NF Builds on 1NF by Further Restricting Allowable Dependencies to Help Normalize the Table Structure
3rd Normal Form

<table>
<thead>
<tr>
<th>ENTITY NORMALIZATION</th>
<th>ATOMICITY</th>
<th>REMOVING PARTIAL DEPENDENCIES</th>
</tr>
</thead>
<tbody>
<tr>
<td>Breaking down data into atomic units to remove redundancy</td>
<td>Each attribute contains only a single value</td>
<td>No non-key attribute depends on just part of a key</td>
</tr>
</tbody>
</table>

3RD NORMAL FORM ELIMINATES DATA REDUNDANCY AND ANOMALIES BY ENSURING ATTRIBUTES ARE DEPENDENT ON THE PRIMARY KEY

@ChistaDATA Inc. 2024
Isolation Levels - Ensuring Data Integrity

- **ISOLATION LEVELS**
  Isolation levels define the degree to which concurrent transactions can interact with each other, balancing data integrity and efficiency.

- **ENSURING DATA INTEGRITY**
  Isolation levels play a key role in ensuring data consistency in databases beyond just ACID properties.

- **LEVELS OF ISOLATION**
  The choice of isolation level impacts data consistency, concurrency, and system performance.
Isolation Levels in MySQL

MySQL uses Repeatable Read isolation level by default. Repeatable Read prevents dirty reads and non-repeatable reads but allows phantom reads.

Isolation levels balance consistency and concurrency. Options like Read Uncommitted offer less isolation but more concurrency.

Choose isolation level carefully for app needs. Consider data consistency needs and performance goals.

Understanding isolation levels in MySQL helps optimize for an app’s specific database needs.

@ChistaDATA Inc. 2024
Scaling Databases

**Scaling Databases is Crucial**
Database scaling is important for maintaining performance, reliability, and availability as data volumes grow.

**Exponential Data Growth**
Data from social media, IoT, and enterprise apps is growing exponentially, driving the need to scale databases.

**Strategies for Scaling**
Common database scaling techniques include sharding, replication, caching, and moving to the cloud.

Database scaling through sharding, replication, caching, and the cloud is key for handling exponential data growth from modern applications.
Primary Methods of Database Scaling

HORIZONTAL SCALING
Adding more machines or nodes to existing database system

VERTICAL SCALING
Upgrading existing hardware of database server

HORIZONTAL SCALING EXPANDS CAPACITY BY ADDING RESOURCES, VERTICAL SCALING UPGRADES EXISTING RESOURCES FOR BETTER PERFORMANCE

@ChistaDATA Inc. 2024
Scaling MySQL Vertically vs Horizontally

VERTICAL SCALING COMES NATURALLY
Vertical scaling is easier and faster to implement initially

VERTICAL SCALING ADDS RESOURCES
Additional CPU, RAM can be added to handle more load

VERTICAL SCALING HAS LIMITS
There is a physical limit to how much you can vertically scale

VERTICAL SCALING IS A GOOD SHORT-TERM TACTIC, BUT HORIZONTAL SCALING IS BETTER FOR THE LONG-TERM

@ChistaDATA Inc. 2024
Proxy Solution

Diagram showing a database setup with application servers, master, slaves, and replication flows.
Clustering Solutions
Sharding Solutions

Vitess Architecture Summary

APP VITESS

Load Balancer

APP server

vtgate

vttablet

mysql

Primary

Shard 1

Big Data

Topo server

Shard 2

Shard 3

Replica

@vitessio
Serverless MySQL Overview

Serverless MySQL Solutions
Several cloud providers offer serverless MySQL options like AWS Aurora Serverless, Google Cloud SQL, Azure Database for MySQL, and PlanetScale.

Instant Scaling
These solutions provide instant and automatic scaling based on application needs, aligning with serverless principles.

High Availability
They offer high availability and resilience by dynamically adjusting capacity and resources.

Reduced Operational Burden
Serverless MySQL offloads database management, freeing developers to focus on apps.

Serverless MySQL solutions from major cloud providers offer automatic scaling, high availability, and reduced ops burden.

@ask_dba
The Future of Database Management

INTEGRATION OF ADVANCED TECHNOLOGIES
The future of database management will see the integration of advanced technologies like AI, ML, automation etc. to enable more intelligent and efficient data handling.

SHIFT TOWARDS FLEXIBLE AND SECURE SOLUTIONS
There will be a move towards more flexible, scalable and secure database solutions that can meet changing business needs and safeguard data integrity.

LEVERAGING DATA IN NEW WAYS
New opportunities will emerge to analyze, visualize and extract value from data across industries in ways that drive innovation and digital transformation.

THE FUTURE OF DATABASE MANAGEMENT LOOKS PROMISING, WITH TECHNOLOGY INTEGRATION, FLEXIBLE ARCHITECTURES AND INNOVATIVE DATA USE CASES SET TO ADDRESS EXISTING CHALLENGES AND UNLOCK NEW POTENTIAL.
Database Design and Modeling with PostgreSQL and MySQL

Build efficient and scalable databases for modern applications using open-source databases

ALKIN TEZUYSAL | IBRAR AHMED

@ChistaDATA Inc. 2024
MySQL Cookbook
Solutions for Database Developers and Administrators

Sveta Smirnova & Alkin Tezuysal

@ChistaDATA Inc. 2024
THANK YOU

Q&A
References & Credits

- **Vitess**
- **MySQL 8.0 Reference Manual :: 23 InnoDB Cluster**
- **ProxySQL**
- **MySQL Workbench**
- **Percona XtraDB Cluster - MySQL Software**
- **PlanetScale**
- **TiDB: The Advanced Distributed SQL Database**
- **YugabyteDB: High-Performing Distributed SQL Database**