What are ACID properties in RDBMS?

Doug Ortiz
May 2023

What are ACID properties in RDBMS?

Welcome

Housekeeping Items

Slides and recording will be available within 24 hours

?

Questions will be answered at the end

EDB™

Doug Ortiz

Senior Postgres DevOps Engineer

Technologies

- Cloud
- Big Data, Data Analytics, Databases
- DevOps and Platform Engineering

Original Co-Author of Open Source Projects:

- edb-deployment (aka postgres-deployment)
- edb-ansible

Experience in:

- Multi-Cloud
- Software Architecture and Development
- DevOps
- Microservices, Containerization, and K8s
- Automation
- Database Technologies

in https://www.linkedin.com/in/doug-ortiz-illustris/

https://www.youtube.com/@techbits-dougortiz

https://dougortiz.blogspot.com/

https://github.com/DougOr/

EDB™

Agenda

1. Introduction
2. ACID Properties
   a. Atomicity
   b. Consistency
   c. Isolation
   d. Durability
3. Demo
4. Takeaways
5. Maximizing ACID properties in Postgres
6. Q&A

Let’s review…

What ACID properties are

What does ACID stand for?

1 - Atomicity

2 - Consistency

3 - Isolation

4 - Durability

Atomicity

Definition

Guarantees that a transaction is treated as a single, indivisible unit of work.

Atomicity guarantees that transactions are all-or-nothing operations.

Atomicity

Characteristics

1. Indivisible
2. Consistent
3. Isolated

Atomicity

Role of Transaction Logs

Transaction logs a.k.a.

- Redo Log
- Write Ahead Log (WAL)

Role

1. Durability and recovery
2. Undo and rollback
3. Redo and commit

Atomicity

Real-world examples

1. Fund transfers
2. E-commerce transactions
3. Reservation systems

Consistency

Definition

Ensures that a database remains in a valid state before and after a transaction.

Consistency guarantees that the data is accurate, valid, and consistently available across the entire database by enforcing data integrity rules and constraints to prevent inconsistencies and anomalies.

Consistency

How it is accomplished

1. Constraints for data integrity

2. Triggers for consistency enforcement

3. Foreign keys and referential integrity for maintaining data integrity

Consistency

Implementation

Methods

1. Primary keys
2. Foreign keys
3. Constraints
4. Triggers

Isolation

Definition

Ensures that concurrent transactions do not interfere with each other.

Provides a level of separation between each transaction, allowing them to execute as if they were the only transaction running.

EDB™

Isolation

Levels

1. Read committed
2. Repeatable read
3. Serializable
4. *Read uncommitted - Not supported in Postgres*

Isolation

Impact

1. Data visibility
2. Data modification conflicts
3. Performance trade-offs

Isolation

Techniques

- Locks
- Optimistic Concurrency Control - OCC
- Multiversion Concurrency Control - MVCC
- Snapshot isolation

Isolation

Trade offs

- Higher isolation levels
  - Provide stronger consistency guarantees
  - May lead to more blocking and decreased concurrency
- Lower isolation levels
  - Provider higher concurrency
  - May result in phenomena such as:
    - Dirty reads
    - Non-repeatable reads

Durability

Definition

Guarantees that once a transaction is committed, changes are permanent and will survive system failures.

This is ensured by utilizing Transaction Logs or Write-Ahead Logs (WAL) to record changes made during a transaction.

Durability

Encompasses

1. Write durability
2. Crash recovery

Durability

Mechanisms

1. Transaction logs
2. Key aspects
   a. Log records
   b. Write-ahead Logging (WAL)

Durability

Maintaining durability

- Transaction logs
  - Record all changes made to the database
    - Committed
    - Un-committed
- Checkpoints
  - Points in the transaction log that ensure that all the data pages have been written to disk
  - Provide a consistent state for recovery

Durability

Strategies

- Full backups
- Incremental backups
- Point-in-time recovery (PITR)
- Recovery Point Object (RPO)
- Replication
- High availability
- **Implement and test Disaster recovery procedures**

Demo

Takeaways

Maximizing ACID in Postgres

- Use transactions
- Define constraints
- Optimize database design
- Choose the right isolation level
- Handle concurrent access
- Implement error handling and rollback mechanisms
- Regularly
  - Backup and perform PITR
  - Monitor and tune performance
- Stay updated with Postgres releases
- Disaster
  - Planning and documenting
  - Recovery drills

References

- Wikipedia - ACID Definition

  https://en.wikipedia.org/wiki/ACID
- IBM - ACID properties

  https://www.ibm.com/docs/en/cics-ts/5.4?topic=processing-acid-properties-transactions
- Geeks for Geeks - ACID properties in DBMS

  https://www.geeksforgeeks.org/acid-properties-in-dbms/

Q&A

THANK YOU