

Assignment 1

Database Engineer (MongoDB)

Q1. If we want to transfer some money from one bank account to another bank account, how does MongoDB achieve ACID property?

Answer. ACID properties include:

1. **Atomicity:** Multiple operations can be grouped into a single logical entity.
2. **Consistency:** Requirement that any given database transaction must change affected data only in allowed ways.
3. **Isolation:** All transactions occur independently.
4. **Durability:** Data should be persistent even if one system fails.

MongoDB achieves ACID during transaction by using Multi document transaction. This ensures that a set of operations is being executed as a single /atomic transaction. MDTs are supported in MongoDB version 4.0 or later.

Following operations can be applied along with the basic CRUD operations:

1. **startSession()** : After using the basic command as `db.collection.find()` to find which collection you want to use, this can be used to create the session where we will perform the transaction.
2. **startTransaction()** : Within each session we need to start some transaction (sending or receiving).

Following commands can be used in transaction:

`db.collection.insertOne()` or `insertMany()` : can be used to insert amount.

`db.collection.distinct()` : to find distinct objects/values from the account

`db.collection.deleteMany()` or `deleteOne()` or `remove()` : can be used to delete any element.

3. **update()**: the account is updated with the changes performed during the transaction.

`db.collection.updateOne` or `updateMany()` or `replace()` can be used

4. **commitTransaction():** The changes are fixed using commit. Whenever a transaction related to addition of money or deduction of money occurs, it is logged in using the commit command, so that later if we want to check transaction at any particular time, we can do so.
5. **abortTransaction():** This can be used if you want to cancel the ongoing transaction if something goes wrong , i.e. one of the transaction fails. Hence, none of the documents will be affected.

Q2. Describe three real life application where MongoDB is used. What are the drawbacks if we use RDBMS in these application

Answer.

1. **Content management:** It can be used to store a large amount of unstructured data including posts, images, videos and other multimedia related content. MongoDB provides compatibility and access to such content on a single database because of its ability to store structured and unstructured both type of data.
2. **Gaming Applications:** Large amount of data such as scores, activity, achievements, history etc are stored in games now a days. This requires large amount of space to store all this huge amount of data and also requires real time updates whenever the player hits new game and gets new score. MongoDB can be used to do this with easy handling.
3. **Social Networking Applications:** Many social media platforms such as Twitter, LinkedIn, Instagram, YouTube etc creates a large amount of data such as tweets, videos, audios, text, posts, likes, comments etc in a very few amounts of time. It needs to be managed efficiently and MongoDB is useful in storing all kind of data generated by these applications. It can easily store large data volume and can separately manage them.

Some of the real life examples are: MetLife, Weather Channel, India's Aadhaar.

Drawbacks of using RDBMS in these applications:

1. **Scalability:** RDBMS cannot handle a large amount of data at a time. So scaling up would be challenging and expensive.
2. **Flexibility:** RDBMS can not handle the unstructured data, hence not suitable.
3. **Performance limitation:** RDBMS uses specified schema mostly which can lead to slow performance while handling complex problems.

References:

<https://hevodata.com/learn/mongodb-use-case/>

<https://digitallearn.xebiaacademyglobal.com/course/view.php?id=375>

Submitted by : Alisha Chauhan

Roll no. : 102017062

3CS3