International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395 -0056
Volume: 04 Issue: 06 | June -2017 www.irjet.net p-ISSN: 2395-0072

Real Time Databases for Applications

Sonam Khedkar[1], Swapnil Thube[2]

MCA (IMCOST), Mumbai University
C-4, Wagle Industrial Estate, Near Mulund (W)
Check Naka, Thane (w) – 400604

Abstract - The purpose of this study is to introduce everyone with real time databases like Google firebase API, Mongo DB, Re think DB and their features. Firebase is a Google provided API for database storage and syncing into your android, IOS or web application. Rethink DB is the first open-source, scalable JSON database built from the ground up for the real time web. Mongo DB is a product between relational database and nosql database which uses distributed technology. A real-time database is one which stores data to database and fetches data from it very quickly but Firebase is not just a real-time database, it is much more than that. Mongo DB is non-relational database just stores data without explicit and structured mechanisms to link data from different buckets to one another.

Key Words: Google Firebase API, MongoDB, RethinkDB, Features, Drawbacks,Database Model,Sceduling,Deadlock.

1. INTRODUCTION

Real-time Database is a cloud-hosted database. Data is stored as JSON format and synchronized continuously to each associated client. When you build cross-platform applications with IOS, Android, and JavaScript SDKs, the greater part of your customers' demand is based on one Real-time Database instance and consequently getting updates with the most current data. A Database is an organized collection of data. Databases can be stored locally on your computer or can be stored in cloud storages. Every application whether android, IOS or web application, it has its own database. In the android application, we can create databases using SQLite, shared preferences, websites or some cloud-based storage sites. The basic idea behind creating database is to store data systematically and fetch data when required. Firebase is also a database backend for android, iOS and web applications. Firebase is Google provided API to create a database and fetch from it in real time with only a few lines of code. Data is stored as JSON and is accessible from all the platforms. MongoDB is an open-source document database that provides high performance, high availability, and automatic scaling. RethinkDB has a query language that supports really useful queries like table joins and group by, and is easy to setup and learn. Real-Time Data Base System can be defined as those computing systems that are designed to operate in a timely manner.

2. FEATURES

2.1 FIREBASE FEATURES

1. Authentication: Authentication feature in firebase let you let only authorised users access your application. Firebase provides login through Gmail, Github, twitter, facebook and also let the developer create custom authentication.

2. Hosting: Hosting is production-grade web content that facilities the developers. With Hosting, you can rapidly and effectively send web applications and static content to a Content Delivery Network (CDN) with a single command. It contains custom domain support, Global CDN and Auto Provisioned SSL Certificate.

3. Messaging: It is a cross-platform messaging solution that lets you dependably convey messages at zero expense. You can inform a customer that new email or other information is accessible to sync. You can send notification messages to drive user reengagement and maintenance.

4. Analytics: This feature enables the application developer to understand how users are using his application. The SDK capture events and properties on its own and also allows you to get custom data. The dashboard provides details like your most active user or what feature of your application is used most. It also provides you with summarised data.

5. Storage: Firebase also provides storage facility. It can store and retrieve content like images, videos and audio directly from client SDK. Uploading and downloading is done in the background. Data stores are safe and the only authorised user can access it.

6. Real-time Database: Database in firebase is a cloud-based database and does not need SQL-based queries to store and fetch data. Database is highly reliable thus even if connection is lost data is maintained.

7.Crash reporting: crash reporting feature on firebase creates reports of error in your app after its release. Errors are grouped into different groups according to how severe error is. You can also create custom events to catch steps leading to the crashing of the application.

8.App Indexing: By using this component you easily get to index your application in Google Search. For an instance, if

your application is already installed in user's device when he searches for related content, it will live your app directly from the search results. If users have not installed your application yet, an install card shows up in search results.

9. AdMob: It is a simple approach to adapt mobile apps with focused in-application promoting. It is mobile advertising platform which you can use to create revenue with the help of your application. It gives you extra application usage information and analytics capabilities.

2.2 Mongo DB FEATURES

1. Mongo DB makes it easy: Mongo DB can consolidate any sort of information – any structure, any arrangement, and any source – regardless of how frequently it changes, analytical engines can be real-time and continuous.

2. Scalability: MongoDB is built to scale out on commodity hardware, in your data center or in the cloud. And without complex hardware or extra software

3. Real Time: It can analyze data of any structure from the database, giving you results in real time, without any data warehouse burdens.

4. High Performance: Mongo DB provides high performance data persistence. Indexes support faster queries so CRUD transaction perform smoothly. In support for embedded data models reduces I/O action on database system.

5. Rich Query Language: Mongo DB supports easy syntax for read and write operations (CRUD) as well as: Data Aggregation Text Search and Geospatial Queries.

6. High Availability: Mongo DB's replication facility, called replica set, provides-automatic failover and data redundancy. It keeps similar informational index, giving excess and expanding information accessibility.
7.Pricing: Depends on which RDBMS of course, but Mongo DB is free and can run on Linux, ideal for running on cheaper commodity kit.

2.3 Rethink DB FEATURES

1. NoSQL – schema-less approach**:** Rethink DB is a JSON document database. A JSON document represents a structured object consisting of key/value pairs. The value can be either a primitive data type (integer, string, floating point number) or a nested JSON object (represented in document form). This means, of course, that JSON can describe arbitrarily complex objects. Rethink DB stores documents in tables. While this might lead one to think that Rethink DB has relational database ingredients, the fact is a table is simply a logical container; the Rethink DB engineers chose to call that container "table" so that developers coming from a relational background would feel comfortable. A table in Rethink DB places no real restrictions on the structure of its contained documents. In the relational world, all rows within a table necessarily have the same structure. But Rethink DB is schema-less: No two documents, not even documents within the same table, need to have the same structure. Of course, it's generally beneficial for all documents in a table to have the same structure, as it simplifies organization and management. But the flexibility is there, if needed.

2. Real time push architecture: Typically, clients are in touch with alterations in database contests by querying the database itself. Thus, if we want to figure what user A has updated in the database, we will need to query it (in different approaches, depends which language is used). Nevertheless, the application is aware of changes only when polling the database. In Rethink DB, this whole idea has changed. It provides a real-time push architecture. Clients can listen do DB changes, that they're notified once there's any change, any time, without a need to poll the database repeatedly.

3. Failover in Rethink DB: Rethink DB supports failover, which requires that the cluster have at least three nodes and tables be configured to have more than two shards. If a node becomes unavailable and happens to host the primary replica for a table, then one of the secondary nodes is selected by Rethink DB to become the new primary. No data is lost. Should the lost node come back online, it will resume its position as primary.

4. JOINs and GMR: In Rethink DB, the equivalent of a relational JOIN operation can actually be performed.

3. DRAWBACKS

3.1 FIREBASE

1. Pricing: Firebase's paid infrastructure offers the Real time Database, Storage, Cloud Functions, Hosting, Test Lab, Phone Authentication and Google Cloud Platform use Big Query & other IAAS.

2. Downloads all sub trees on load: If you are building a Slack-like app, you will have to download all channels data on app load.

3. Inconsistencies: It supports offline operation but if client goes offline and then comes online it may lead to inconsistencies.

4. Data migration: You can't deal easily with data-migration like you can do with a simple SQL database, an ORM or ODM.

5. Queues: It introduce queues to share operations between servers and prevent concurrency. Queues scale bottleneck as

queue items can be stacked quickly, but unstacked very slowly.

6.Complex Queries: Still impossible to query database to find fields with some properties. Operations cannot be performed to get active users, or batch operations to update documents with some fields.

7. Data Export: Fact that your data is hosted on servers that you don't own, it's not possible to export your user data.

3.2 MONGO DB

1. Higher data size: Data size in Mongo DB is typically higher due to e.g. each document has field names stored it

2. Flexibility: Mongo DB is less flexible with querying as it doesn't support joins

3**. Transportation:** It doesn't support for transactions but certain atomic operations are supported at a single level.
4.At the moment Map/Reduce (e.g. to do aggregations/data analysis) is OK, but not blisteringly fast. So if that's required, something like Hadoop may need to be added into the mix

3.2 Rethink DB

1. Though there is no limit on the number of databases that can be created but there is limit of 64 shards.
2. There is a recommended limit of 16MB for memory due to performance reasons.
3. Primary keys are limited to 127 characters. Secondary indexes do not store objects or null values.
4. Numbers are double precision IEEE 754 floating point. Rethink DB does not allow NaN or infinite values.
5. Rethink DB is not a decent decision if you require full ACID support or solid mapping implementation—for this situation RDBMS database is best option. For example, MySQL or PostgreSQL.

4. COMPARISON

1. FIREBASE VS. MONGODB

	Firebase Real time Database	Mongo DB
Description	Cloud-hosted realtime document store. Ios, Android, and JavaScript clients share one Realtime Database instance and automatically receive updates with the newest data.	One of the most popular document stores
DB Engine Ranking	Score 3.81 Rank 60(overall)	Score 335.0 Rank 5(overall)
Documentation	https://firebase.google.com/docs/database/	https://docs.mongodb.com/manual/
Website	https://firebase.google.com/	https://www.mongodb.com/
License	commercial	Open Source
Cloud based	Yes	No
Server Operating system	hosted	Linux OS X Solaris Windows

2. RETHINK VS MONGODB

	Rethink	MongoDB
Platforms	Linux,OSX,Windows	Linux,OSX,Windows,Solaris
Data Model	JSON documents	BJSON documents
Javascript Integration	V8 engine	Spidermonkey/V8 engine
Data Access	Unified chainable dynamic query language	Dynamic rich query language
Index Type	Primary key Compound Secondary Geospatial Arbitrarily computed	Unique (unsharded only) Compound Secondary Geospatial Sparse
Cloud Deployment	AWS, dotCloud, Compose.io	Many cloud platforms
Access Language	JSON protocol 3 official libraries Many community supported libraries	BSON protocol 13 official libraries Many community supported libraries

5. REAL TIME DATABASE MODEL

- Any new transaction must go through an Admission Control component, which monitors and manages the aggregate number of simultaneously active transactions within the system in order to avoid transaction failure.
- Each new or resubmitted transaction is assigned a Priority Level, which arranges its planning inclination in respect to the next simultaneous transactions inside the system.
- Before a transaction performs an operation on a data object, it must experience the Concurrency Control part so as to accomplish the required synchronization. On the off chance that the exchange's demand for a granule is denied, the transaction will be put into a Wait Queue.
- The waiting transaction will be reactivated when the requested granule becomes available, after which the transaction performs its operation.

- Similarly, if a transaction a thing that is right now not in fundamental memory, an I/O ask for is started and the transaction will be put into a hold up line.
- The holding up exchange will be reactivated when the requested granule becomes available in fundamental memory, and there is no dynamic higher-need exchange.
- When a transaction completes all of its operations, it commits its result(s) and releases all of the data items in its possession.
- A transaction may abort/restart a number of times before it commits. There are various types of aborts :
- Terminating abort:
    - An abort due to missing a deadline, or
    - Self-abort – a transaction may abort itself due to an exceptional condition.
- Non-terminating abort: An abort due to a deadlock or a data conflict. In this case, the transaction maybe restarted if its deadline remains feasible.

6. REAL TIME DATABASE TRANSACTION SCEDULING

Real Time database system provides feature with respect to standard physical resource is the data objects stored in the database. Transactions accessing this data have to be scheduled in accordance with real-time performance objectives. The scheduling process of transactions in a real time database system consists of:

- Concurrency Control
- Conflict Resolution

Concurrency Control Protocols: Locking, Time-stamping, Multiversion, Validation-all of which have the same goal; i.e., enforcing serializability. These Protocols need to be modified and their trade-off(s) must be reevaluated under RTDB systems.

1. Concurrency Control Protocol: Locks are used to synchronize concurrent actions-Two-Phase Locking (2PL).All locking operations precedes the first unlock operation in the transaction, expanding phase (locks are acquired), shrinking phase (locks are released), suffers from deadlock, priority inversion.

2. Conflict Resolution: Priority-based Wound-Wait Conflict Resolution-the original scheme was designed to use timestamps. It was modified so that the scheme uses priorities instead of timestamps. Modified scheme consist of High-Priority and Priority-Abort.

7. DEADLOCK

Deadlocks: Whenever a set of transactions gets involved in a circular wait in what is known as a wait-for graph. Five deadlock resolution policies that take into account: the timing properties of the transactions, the cost of abort operations.

1. **Policy 1**:
    a. Always aborts the transaction invoking deadlock detection.
2. **Policy 2**:
    a. Trace the deadlock cycle
    b. Abort the first tardy transaction encountered in a deadlock cycle.
    c. If no tardy transaction is found, abort the transaction with the furthest deadline.
3. **Policy 3:**
    a. Trace the deadlock cycle.
    b. Abort the first tardy transaction encountered in a deadlock cycle.
    c. If no tardy transaction is found, abort the transaction with the earliest deadline.
4. **Policy 4**:
    a. Trace the deadlock cycle, and abort the first tardy transaction encountered in a deadlock cycle.
    b. If no tardy transaction is found, abort the transaction with the least criticalness.
5. **Policy 5**:
    a. Abort the infeasible transaction with the least criticalness.
    b. If all transactions are feasible, then abort a feasible transaction with the least criticalness.
    c. This policy is sensitive to the accuracy of the computation time because it requires information about remaining execution time
    d. So; Total execution time requirements at the start of each transaction must be known.

8. CONCLUSION

In this paper we undergo comparison between firebase, Mongo DB &Rethink. We came to the conclusion that -Real-Time Data Base System can also be defined as Traditional Databases that uses an extension to give additional power to yield reliable response. Since data is stored on cloud it is readily available any where. If your app does run of a centralized DB, and is updated by a lot of users – then it's more than capable of handling the Real-Time data updates between devices. We have studied about Google provided firebase API, Mongo DB & Rethink and their unique features along with drawbacks. By utilizing this feature, there is no necessity to make your own database or own API, real time database handles all the components that usually come along with creating a backend for applications This study provides an extensive survey of different real-time systems research.

ACKNOWLEDGMENT

We thank our colleagues from IMCOST who provided insight and expertise that greatly assisted the research, we thank

Mrs.Reeta Singh for assistance us by providing proper IEEE formats and all faculties for guiding us. We would also like to show our gratitude to the librarian staff Mr Milind Dubal for sharing references with us during the course of this research.

REFERENCES

[1] Dipinadamodaran B1 , Shirin Salim2 And Surekhamariumvargese 3, PERFORMANCE EVALUATION OF MYSQL AND MONGODB DATABASES ,International Journal On Cybernetics & Informatics (IJCI) Vol. 5, No. 2, April 2016
[2]Ebrahimabduljalil1,Dr. S.B. Thorat2, Security & Concurrency Control In Real-Time Database Systems: A Survey, IJARCCE Vol. 5, Issue 2, February 2016
[3]International Journal Of Innovative Research In Computer And Communication Engineering (An ISO 3297: 2007 Certified Organization), September 2016
[4]Navdeep Singh, Study Of Google Firebase API For Android, IJIRCCE (An ISO 3297: 2007 Certified Organization) Vol. 4, Issue 9, September 2016
[5]Fadia A. Elbagir, Ahmed Khalid, Khalid Khanfar, A Survey Of Commit Protocols In Distributed Real Time Database Systemsinternational Journal Of Computer Trends And Technology (IJCTT),Vol.31,Issue 2, 2016