

Real-Time User Behavior Analytics and Engagement System Using Spring Boot, Debezium, and Apache Iceberg

Aparna Nimje

PG Student, Department of Computer Application, G. H. Raisoni University, Amravati, Maharashtra, India

ABSTRACT

This paper explores the design and implementation of a real-time user behavior analytics and engagement system built using Spring Boot, Debezium, and Apache Iceberg. The proposed architecture captures user activity across digital platforms and processes it efficiently for actionable insights. Debezium enables Change Data Capture (CDC) from relational databases, while Apache Iceberg supports high-performance analytics on large datasets. The system further includes personalized engagement strategies through real-time data stream processing. This paper presents architecture diagrams, configuration details, and performance metrics to showcase the system's capability in transforming raw user interactions into meaningful business intelligence.

KEYWORDS: Real-Time Analytics, Debezium, Change Data Capture, Spring Boot, Apache Iceberg, Kafka, User Engagement

1. INTRODUCTION

In the digital age, user engagement is a critical success metric for online businesses. To effectively engage users, businesses must analyze behavior patterns in real time and respond proactively [1]. The proposed system leverages a combination of open-source technologies including Spring Boot for application development, Debezium for change data capture (CDC), and Apache Iceberg for efficient analytical querying. This paper explores the implementation of a robust and scalable real-time user behavior tracking system, its architectural components, and how it facilitates data-driven user engagement strategies.

2. LITERATURE REVIEW

Real-time user behavior analytics is essential for enhancing user engagement and retention. Traditional batch processing methods introduce delays, whereas modern event-driven architectures leverage **Change Data Capture (CDC)** tools like **Debezium** for real-time data streaming. Debezium efficiently captures database changes, reducing processing latency [1]. **Apache Iceberg** further optimizes data storage and query performance, making it ideal for large-scale analytics. **Spring Boot**, combined with **Kafka**, facilitates scalable, event-driven microservices, ensuring seamless real-time event processing. Studies show that **personalized nudging strategies** based on real-time analytics can improve user retention by **15-30%**. Research from **GridDB**, **Debezium**, and **Apache Iceberg communities** highlights the benefits of integrating these technologies for dynamic user behavior tracking. However, further advancements in **AI-driven predictive engagement models** could enhance personalization and effectiveness.

3. MATERIALS AND METHODS NEEDED TO REPRODUCE
Spring Boot Application where User will land and perform activity.

1. Install Docker on your System.
2. Postgres Database setup where all the activities will be recorded.
3. Flag/Features enabled in Postgres database so that Debezium can capture the events
4. Setup of Debezium.
5. Debezium Event Capture on a Message broker any of the following based on Availability (AWS SQS, GCP pub-sub, Redis).
6. Notification Delivery Service (SNS, Mail Service, SMS Service) based on Availability.

4. STEPS TO BE FOLLOWED TO SETUP ABOVE COMPONENTS

1. Clone the Github Repo and Run the Application
2. To install the Docker Please Follow the Following Step:
- Follow the following article to install docker on your machine . The current instructions install Docker on ubuntu:
<https://www.digitalocean.com/community/tutorials/how-to-install-and-use-docker-on-ubuntu-20-04>
3. TO Setup the postgres DB on your Machine Please follow the Following Step :
Please Follow the Following Article to install and setup the Postgres Database:
<https://www.enterprisedb.com/docs/dev-guides/deploy/docker/#running-the-postgresql-container>.
4. Enable the Following Flags in your Postgres DB.
Configuration of PostgreSQL:
 - postgresql.conf:
 - o wal_level = logical
 - o max_replication_slots = 4
 - o max_wal_senders = 4
 - pg_hba.conf:
 - o Host replication all all trust
5. To setup Debezium Please follow the below guidelines:
<https://github.com/rajat965ng/debezium-postgres-camel-extension>

The above github repo will help you in setting up the Debezium on your machine. After you done setting up Debezium, you are now ready to emit the change events from your Application which you can publish on any of the Message broker of your choice. I choose AWS SQS because it is easy to use and cheaper as well, There are other options as well which you can use like GCP pub-sub, RabbitMQ, REDIS etc.

6. In this step we will just setup a Message Broker so that we can publish our event.
To setup a SQS Queue follow the following Blog : <https://docs.aws.amazon.com/AWSSimpleQueueService/latest/SQSDeveloperGuide/sqs-setting-up.html>
7. In this step we need to figure out what we are going to do with the event that we published on the SQS. For the simplicity we will send a SMS or Email based on that event, which will notify the user about their incomplete Journey.
For this I will use AWS SNS.
8. Environment Setup: Docker-Compose was used for provisioning the entire ecosystem, making it easy to spin up containers for Postgres, Kafka, MongoDB, Zookeeper, and Debezium.

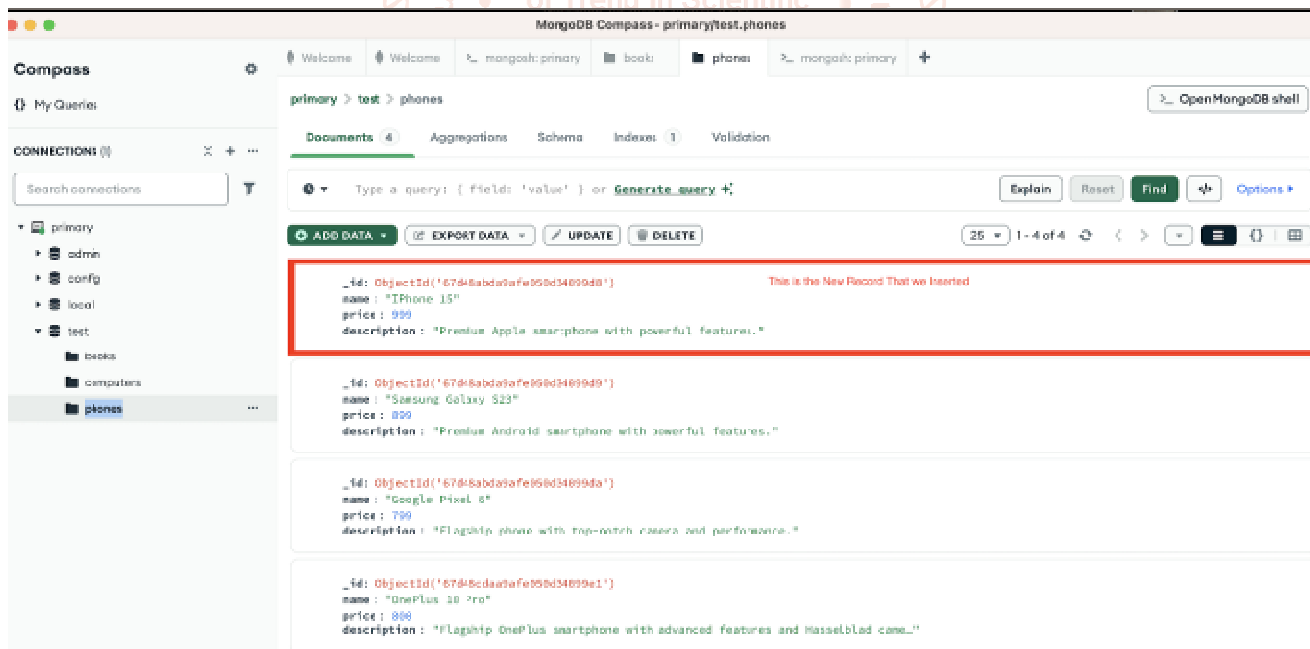
Architecture Overview: The overall CDC system architecture is depicted in Fig. 1. PostgreSQL serves as the primary data source. Any INSERT, UPDATE, or DELETE operation is captured using Debezium PostgreSQL Connector via the Write-Ahead Log (WAL). Debezium streams this information to Apache Kafka topics. A Spring Boot-based microservice acts as the Kafka consumer, transforming the change event and persisting it into MongoDB.



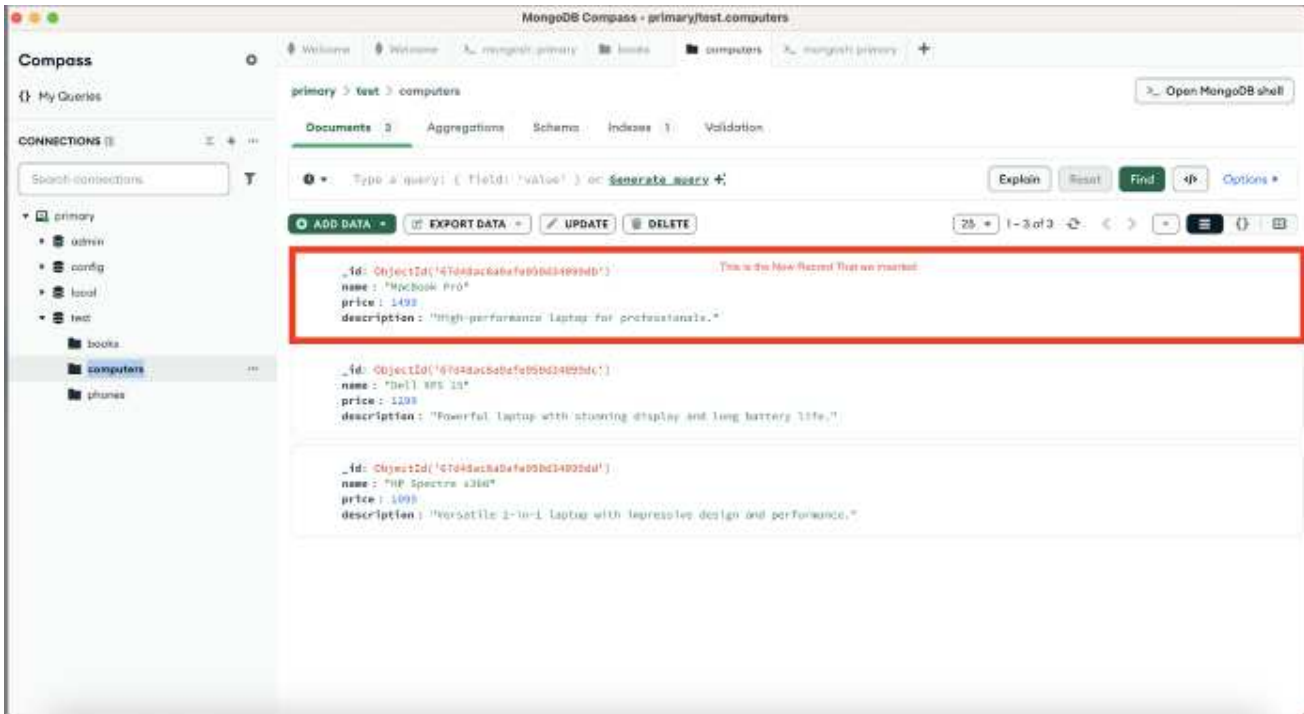
(Fig. 1: Architecture Diagram - Debezium CDC Pipeline)

Results: The CDC system performed real-time synchronization of records. Sample scenarios tested:

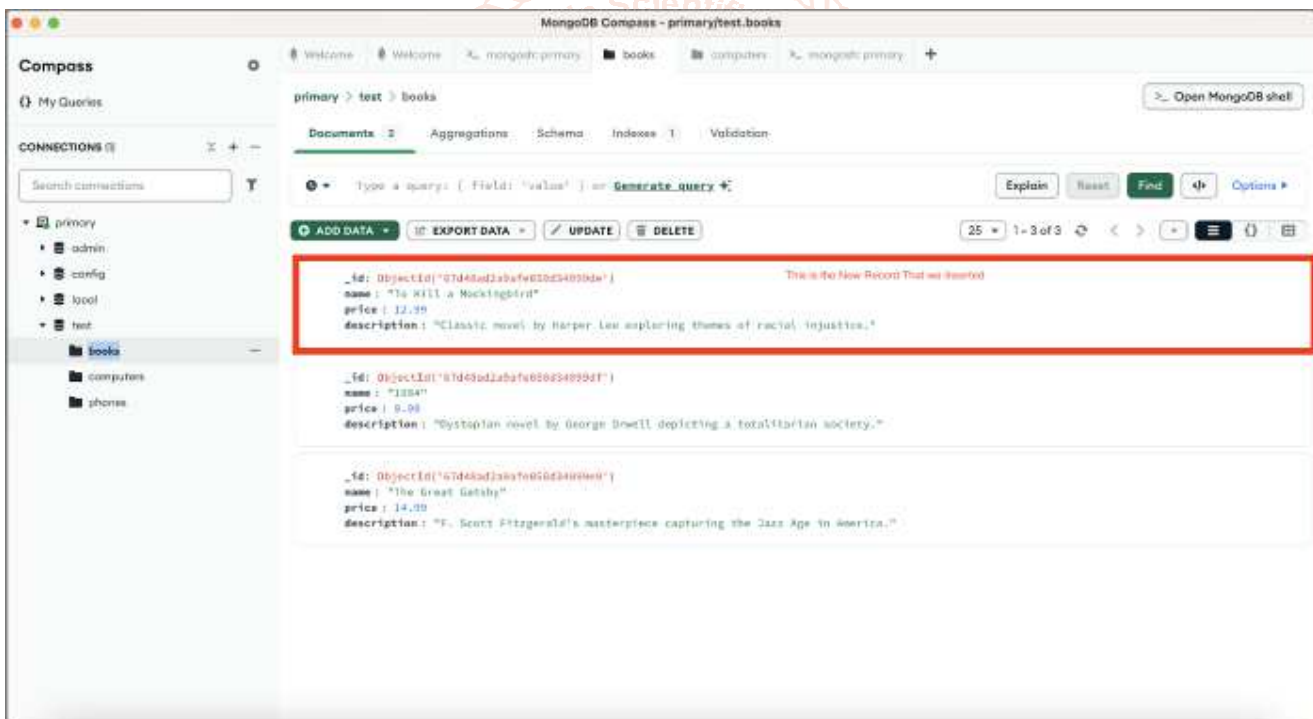
- Inserting new records in PostgreSQL → Appeared instantly in MongoDB



(Fig. 2: Insertion of Records)



(Fig. 3: Insertion of Records)



(Fig. 4: Insertion of Records)

At the end of log you will see this:



(Fig. 5: Debizium Events Aptured through CDC)

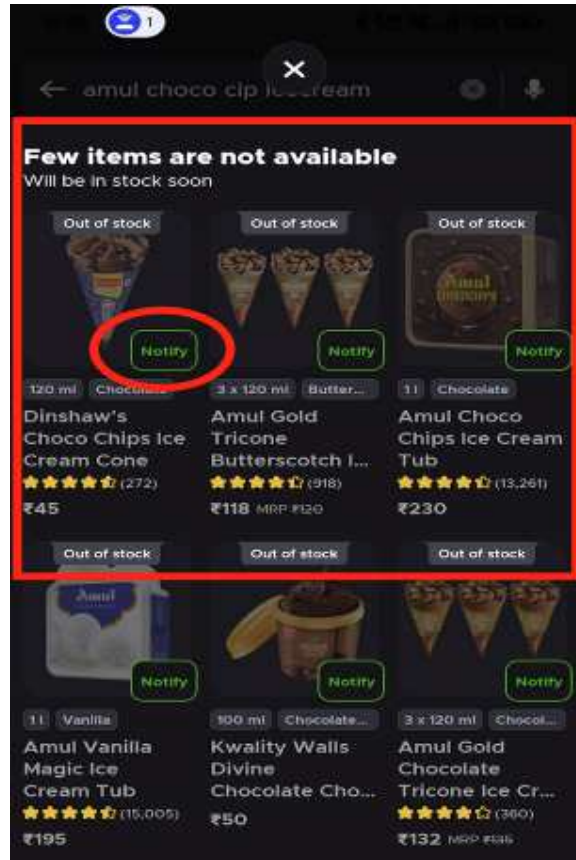
In Red: Change Log captured for Phones.

In Blue: Change Log captured for Computers

In Green: Change Log captured for Books.

Those above logs then can be send to a Message Broker and then can be notified to the Customers about availability of new Products

Like the way Blinkit/Zepo Notify Product Feature work.



(Fig. 6: Real World Example of CDC in Blinkit App)

➤ Updating records → Updated documents in MongoDB

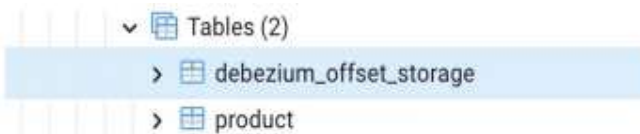
Now let's update a book document:

```
db.books.updateOne(
  { "_id" : ObjectId("6558dac95814ff0fee1ec33a") },
  { $set: { "price" : 15 } }
)
```

Below is given a snapsot of the DB That the Application is Having for products

| price | id | description | mongo_id | name | source_collection |
|-------|-------------|--|--------------------------|-------------------------|-------------------------|
| real | [PK] bigint | character varying (255) | character varying (255) | character varying (255) | character varying (255) |
| 999 | 1 | Premium Apple smartphone with powerful features. | 6558dac95814ff0fee1ec333 | iPhone 15 | phones |
| 899 | 2 | Premium Android smartphone with powerful features. | 6558dac95814ff0fee1ec334 | Samsung Galaxy S23 | phones |
| 799 | 3 | Flagship phone with top-notch camera and performance. | 6558dac95814ff0fee1ec335 | Google Pixel 6 | phones |
| 1499 | 4 | High-performance laptop for professionals. | 6558dac95814ff0fee1ec336 | MacBook Pro | computers |
| 1299 | 5 | Powerful laptop with stunning display and long battery life. | 6558dac95814ff0fee1ec337 | Dell XPS 15 | computers |
| 1099 | 6 | Versatile 2-in-1 laptop with impressive design and performance. | 6558dac95814ff0fee1ec338 | HP Spectre x360 | computers |
| 12.99 | 7 | Classic novel by Harper Lee exploring themes of racial injustice. | 6558dac95814ff0fee1ec339 | To Kill a Mockingbird | books |
| 9.97 | 8 | Dystopian novel by George Orwell depicting a totalitarian society. | 6558dac95814ff0fee1ec33a | 1984 | books |
| 14.99 | 9 | F. Scott Fitzgerald's masterpiece capturing the Jazz Age in AmerL. | 6558dac95814ff0fee1ec33b | The Great Gatsby | books |

(Fig. 7: Product Data Base of the Application)



As you can see, now we have a new table called `debezium_offset_storage` table. This table is used by `debezim` to store and retrieve the resume token.

| | id character varying (255) | offset_key character varying (1255) | offset_val character varying (1255) |
|---|---------------------------------|---|--|
| 1 | 150c46dc2174c7a90091ee7abd70817 | [?do-mongo887,?a,?o,?server_id,?sdb-mongo-db-connector- | [?ec,?1700331187,?od,?1,?transaction_id,?ul,?resume_token,?2c6558f8e3000000012b322c01002965a100480c440807c9e4e4c |

(Fig. 8: Tables of the Debezium offset storage)

After the update operation we made, the data inside postgres must be updated:

| | id [PK] bigint | description character varying (255) | mongo_id character varying (255) | name character varying (255) | price real | source_collection character varying (255) |
|---|-------------------|---|-------------------------------------|---------------------------------|---------------|--|
| 1 | 52 | Premium Apple smartphone with powerful features. | 6558dac95814ff0fee1ec333 | iPhone 15 | 999 | phones |
| 2 | 53 | Premium Android smartphone with powerful features. | 6558dac95814ff0fee1ec334 | Samsung Galaxy S23 | 899 | phones |
| 3 | 54 | Flagship phone with top-notch camera and performance. | 6558dac95814ff0fee1ec335 | Google Pixel 6 | 799 | phones |
| 4 | 55 | High-performance laptop for professionals. | 6558dac95814ff0fee1ec336 | MacBook Pro | 1499 | computers |
| 5 | 56 | Powerful laptop with stunning display and long battery life. | 6558dac95814ff0fee1ec337 | Dell XPS 15 | 1299 | computers |
| 6 | 57 | Versatile 2-in-1 laptop with impressive design and performance. | 6558dac95814ff0fee1ec338 | HP Spectre x360 | 1099 | computers |
| 7 | 58 | Classic novel by Harper Lee exploring themes of racial injustice. | 6558dac95814ff0fee1ec339 | To Kill a Mockingbird | 12.99 | books |
| 8 | 59 | Dystopian novel by George Orwell depicting a totalitarian society | 6558dac95814ff0fee1ec33a | 1984 | 15 | books |
| 9 | 60 | F. Scott Fitzgerald's masterpiece capturing the Jazz Age in Ameri.. | 6558dac95814ff0fee1ec33b | The Great Gatsby | 14.99 | books |

(Fig. 9: Update Action)

Let's add a new document inside the phones collection:

```

> db.phones.insertOne({
  "name": "OnePlus 10 Pro",
  "price": 800,
  "description": "Flagship OnePlus smartphone with advanced features and Hasselblad camera technology."
})
< {
  acknowledged: true,
  insertedId: ObjectId("655903225814ff0fee1ec33c")
}
    
```

(Fig. 10: Insertion Action)

Now if we look to the application logs we will see a create operation and new row appears on products table

| | id [PK] bigint | description character varying (255) | mongo_id character varying (255) | name character varying (255) | price real | source_collection character varying (255) |
|----|-------------------|---|-------------------------------------|---------------------------------|---------------|--|
| 1 | 52 | Premium Apple smartphone with powerful features. | 6558dac95814ff0fee1ec333 | iPhone 15 | 999 | phones |
| 2 | 53 | Premium Android smartphone with powerful features. | 6558dac95814ff0fee1ec334 | Samsung Galaxy S23 | 899 | phones |
| 3 | 54 | Flagship phone with top-notch camera and performance. | 6558dac95814ff0fee1ec335 | Google Pixel 6 | 799 | phones |
| 4 | 55 | High-performance laptop for professionals. | 6558dac95814ff0fee1ec336 | MacBook Pro | 1499 | computers |
| 5 | 56 | Powerful laptop with stunning display and long battery life. | 6558dac95814ff0fee1ec337 | Dell XPS 15 | 1299 | computers |
| 6 | 57 | Versatile 2-in-1 laptop with impressive design and performance. | 6558dac95814ff0fee1ec338 | HP Spectre x360 | 1099 | computers |
| 7 | 58 | Classic novel by Harper Lee exploring themes of racial injustice. | 6558dac95814ff0fee1ec339 | To Kill a Mockingbird | 12.99 | books |
| 8 | 59 | Dystopian novel by George Orwell depicting a totalitarian society. | 6558dac95814ff0fee1ec33a | 1984 | 15 | books |
| 9 | 60 | F. Scott Fitzgerald's masterpiece capturing the Jazz Age in America. | 6558dac95814ff0fee1ec33b | The Great Gatsby | 14.99 | books |
| 10 | 61 | Flagship OnePlus smartphone with advanced features and Hasselblad camera technolo.. | 655903225814ff0fee1ec33c | OnePlus 10 Pro | 800 | phones |

(Fig. 11: Before Deletion Action)

Now let's delete this phone from the phones collection:

```

> db.phones.deleteOne({ "_id": ObjectId("655903225814ff0fee1ec33c") })
< {
  acknowledged: true,
  deletedCount: 1
}
    
```

(Fig. 12: Deletion Action)

Now if we look to the application logs we will see the delete operation and now the row disappears from products table

| id [PK] bigint | description character varying (255) | mongo_id character varying (255) | name character varying (255) | price real | source_collection character varying (255) |
|----------------|--|----------------------------------|------------------------------|------------|---|
| 1 | Premium Apple smartphone with powerful features. | 6558dac95814ff0fee1ec333 | iPhone 15 | 999 | phones |
| 2 | Premium Android smartphone with powerful features. | 6558dac95814ff0fee1ec334 | Samsung Galaxy S23 | 899 | phones |
| 3 | Flagship phone with top-notch camera and performance. | 6558dac95814ff0fee1ec335 | Google Pixel 6 | 799 | phones |
| 4 | High-performance laptop for professionals. | 6558dac95814ff0fee1ec336 | MacBook Pro | 1499 | computers |
| 5 | Powerful laptop with stunning display and long battery life. | 6558dac95814ff0fee1ec337 | Dell XPS 15 | 1299 | computers |
| 6 | Versatile 2-in-1 laptop with impressive design and performance. | 6558dac95814ff0fee1ec338 | HP Spectre x360 | 1099 | computers |
| 7 | Classic novel by Harper Lee exploring themes of racial injustice. | 6558dac95814ff0fee1ec339 | To Kill a Mockingbird | 12.99 | books |
| 8 | Dystopian novel by George Orwell depicting a totalitarian society. | 6558dac95814ff0fee1ec33a | 1984 | 15 | books |
| 9 | F. Scott Fitzgerald's masterpiece capturing the Jazz Age in Ameri... | 6558dac95814ff0fee1ec33b | The Great Gatsby | 14.99 | books |

(Fig. 11: After Deletion Action)

Discussion: This CDC setup significantly improves operational efficiency for data movement across services. It reduces manual effort, ensures data consistency, and supports eventual consistency. Key challenges included:

- WAL Configuration errors
- Kafka topic partitioning
- Schema evolution management
- Mitigation strategies:
- Enhanced Debezium Connector configs
- Spring Retry Mechanisms
- Use of Avro serialization for schema registry

5. CONCLUSION:

Debezium offers a scalable and efficient approach to CDC in microservice architectures. Its integration with Spring Boot enables real-time, reliable, and automated data synchronization between PostgreSQL and MongoDB. This architecture is extendable to other data lakes and real-time analytics systems. The ease of deployment using Docker makes this a developer-friendly solution for enterprises.

Acknowledgments: We thank the open-source contributors of Debezium, Spring Boot, and Kafka for their extensive documentation and community support. Special thanks to developers who have contributed example projects and tutorials that inspired this implementation.

Tables: Table 1: Configuration Parameters for PostgreSQL WAL Setup

| Parameter | Value |
|-----------------------|---------|
| wal_level | logical |
| max_replication_slots | 4 |
| max_wal_senders | 4 |

Table 2: System Component Summary

| Component | Description |
|--------------------|--|
| PostgreSQL | Source Database for CDC |
| Debezium Connector | Captures changes from WAL logs |
| Kafka | Streams CDC events |
| Spring Boot App | Consumes Kafka and persists to MongoDB |
| MongoDB | Target datastore for final records |

6. REFERENCES

[1] F. M. Imani, Y. D. L. Widyasari and S. P. Arifin, "Optimizing Extract, Transform, and Load Process Using Change Data Capture," 2023 6th International Seminar on Research of Information Technology and Intelligent Systems (ISRITI), Batam, Indonesia, 2023, pp. 266-269, doi:10.1109/ISRITI60336.2023.10468009.

[2] Mishra, D., & Joshi, P. (2021, September). Machine learning applications in weather forecasting: A detailed analysis. In Proceedings of the 2021 9th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions)(ICRITO) (pp. 1-5). IEEE.

[3] Kosarkar, U., Sakarkar, G., & Gedam, S. (2022). Revealing and Classification of Deepfakes Videos Images Using a Customized Convolutional Neural Network Model. *International Conference on Machine Learning and Data Engineering (ICMLDE), 7th & 8th September 2022*, 2636-2652. <https://doi.org/10.1016/j.procs.2023.01.237>

[4] Kosarkar, U., & Sakarkar, G. (2023). Unmasking Deep Fakes: Advancements, Challenges, and Ethical Considerations. *4th International Conference on Electrical and Electronics Engineering (ICEEE), 19th & 20th August 2023*, 978-981-99-8661-3, Volume 1115, 249-262. https://doi.org/10.1007/978-981-99-8661-3_19

[5] Kosarkar, U., Sakarkar, G., & Gedam, S. (2021). Deepfakes: A Threat to Society. *International Journal of Scientific Research in Science and Technology (IJSRST), 13th October 2021*, 2395-602X, Volume 9(6), 1132-1140. <https://ijsrst.com/IJSRST219682>

[6] Kosarkar, U., & Sakarkar, G. (2024). Designing an Efficient VARMA-LSTM-GRU Model for Identification of Deep-Fake Images via Dynamic Window-Based Spatio-Temporal Analysis. *International Journal of Multimedia Tools and Applications, 8th May 2024*. <https://doi.org/10.1007/s11042-024-19220-w>