

Debezium

Streaming de cambios de tu base de datos

Agenda

- ¿Qué es Change Data Capture (CDC)?
- ¿Qué es Debezium?
- Arquitectura
- Demo
- Transformaciones
- Demo

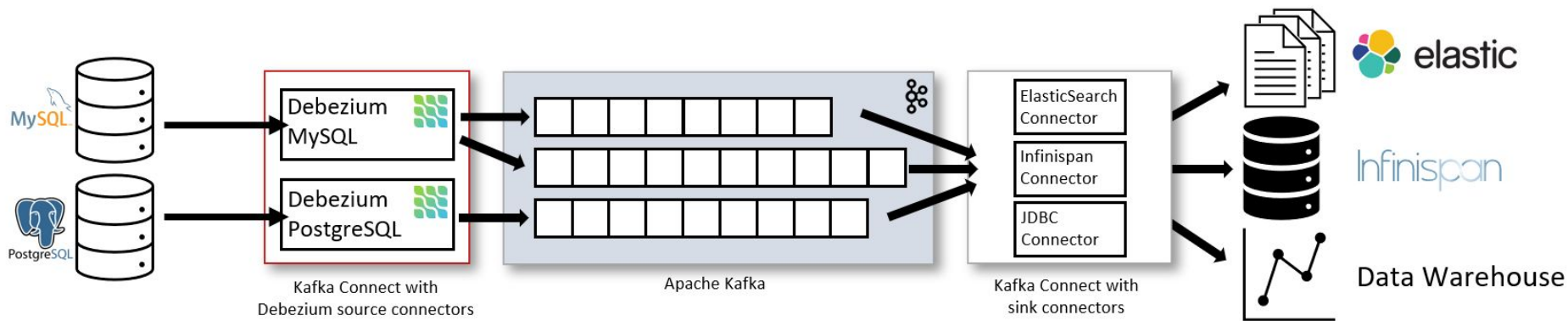
¿Qué es Change Data Capture (CDC)?

- Técnica que permite detectar cambios que ocurren en las bases de datos para transmitirlos de manera inmediata y así poder hacer cosas con esa información.
- Los sistemas CDC suelen ser no intrusivos, es decir, que no tienen ningún tipo de impacto en los sistemas que ya existen en producción.
- Se basan en la lectura de los logs de la base de datos.
- Se pueden integrar distintas fuentes de datos.
- Nos permiten evolucionar sistemas obsoletos o monolíticos.
- Reactividad, la información del cambio la tenemos en el mismo momento en que ocurre (tiempo real)
- Eliminar procesos batch (schedules, cron, etc...)

¿Qué es Debezium?

- Es un CDC de código abierto, desarrollado por Red Hat.
- Captura los cambios a nivel de fila de cada tabla de la base de datos.
- Los cambios en la base de datos son informados en forma de stream de eventos.
- Debezium lee los logs de la base de datos.
- Podemos leer datos de: MySQL, MongoDB, PostgreSQL, SQL Server, Oracle y Db2

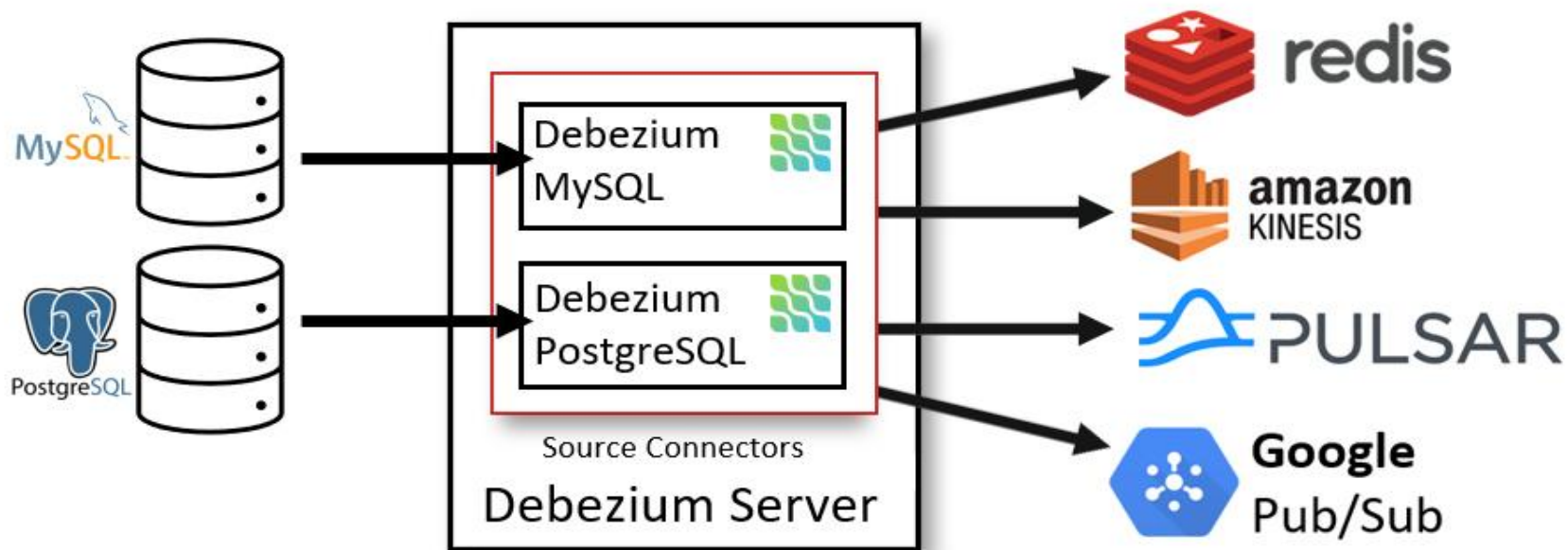
Arquitectura. Conectores de Kafka



Arquitectura. Conectores de Kafka

- Es la configuración más usada y la que se recomienda usar.
- Por defecto, los cambios de una tabla de la base de datos son escritos en topics de Kafka cuyo nombre se corresponde con el nombre de la tabla.
- Se puede cambiar el nombre del topic de destino mediante transformaciones que ofrece el propio Debezium.
- También se puede enrutar cambios de múltiples tablas en un mismo topic.
- Una vez que los cambios los tenemos en Kafka, podemos tener múltiples aplicaciones consumidoras, como Elasticsearch, datawarehouse, cachés...

Arquitectura. Debezium Server



Arquitectura. Debezium Server

- Es una aplicación lista para leer cambios en la base de datos y enviarlos a una variedad de infraestructuras de mensajería: Amazon Kinesis, Google Cloud Pub/Sub, Apache Pulsar,...

Arquitectura. Motor embebido

- Se puede embeber el core de Debezium dentro de nuestras aplicaciones Java.
- Este caso no usa ni Kafka ni Kafka Connect.

```
{
  "schema": {...},
  "payload": {
    "before": { ①
      "id": 1004,
      "first_name": "Anne",
      "last_name": "Kretchmar",
      "email": "annek@noanswer.org"
    },
    "after": { ②
      "id": 1004,
      "first_name": "Anne Marie",
      "last_name": "Kretchmar",
      "email": "annek@noanswer.org"
    },
    "source": { ③
      "name": "1.8.1.Final",
      "name": "dbserver1",
      "server_id": 223344,
      "ts_sec": 1486501486,
      "gtid": null,
      "file": "mysql-bin.000003",
      "pos": 364,
      "row": 0,
      "snapshot": null,
      "thread": 3,
      "db": "inventory",
      "table": "customers"
    },
    "op": "u", ④
    "ts_ms": 1486501486308 ⑤
  }
}
```

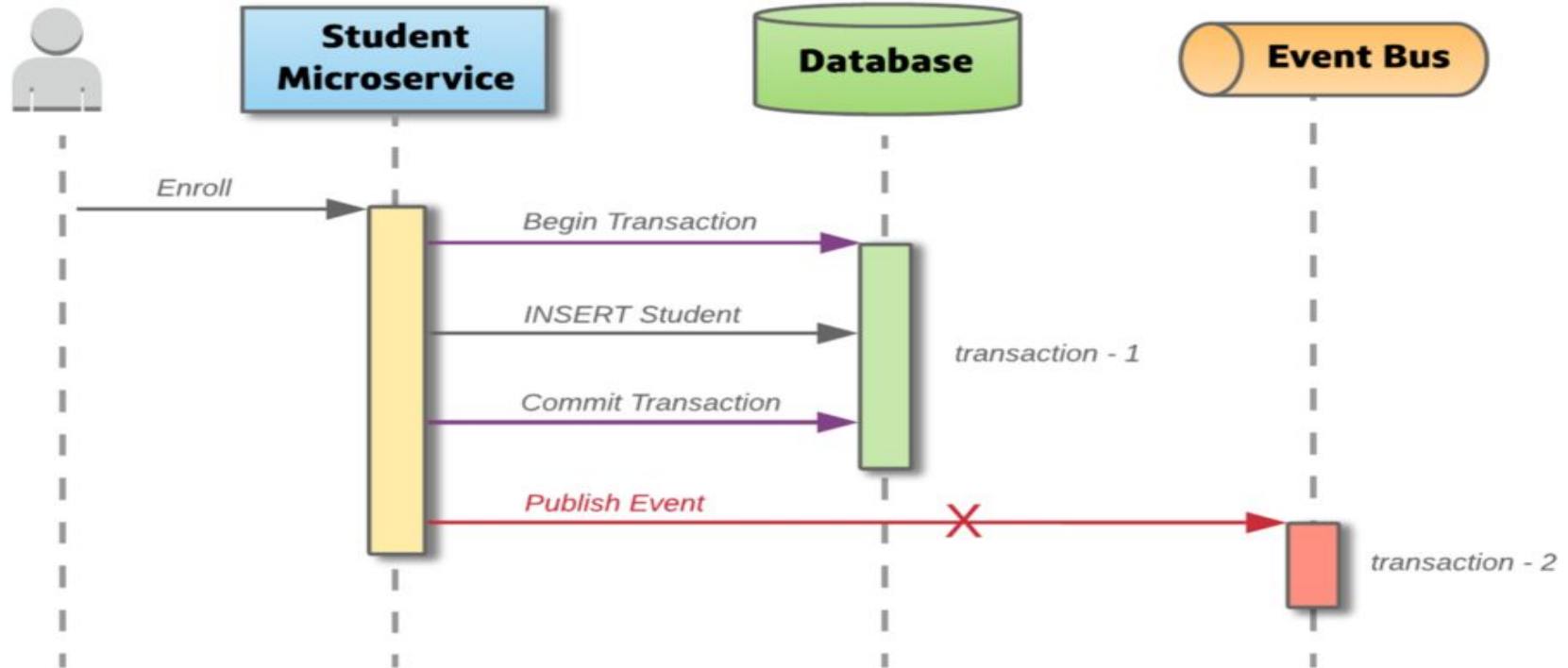
```
{  
  "name": "inventory-connector", ①  
  "config": { ②  
    "connector.class": "io.debezium.connector.mysql.MySqlConnector",  
    "tasks.max": "1", ③  
    "database.hostname": "mysql", ④  
    "database.port": "3306",  
    "database.user": "debezium",  
    "database.password": "dbz",  
    "database.server.id": "184054", ⑤  
    "database.server.name": "dbserver1", ⑤  
    "database.include.list": "inventory", ⑥  
    "database.history.kafka.bootstrap.servers": "kafka:9092", ⑦  
    "database.history.kafka.topic": "schema-changes.inventory" ⑦  
  }  
}
```

Demo

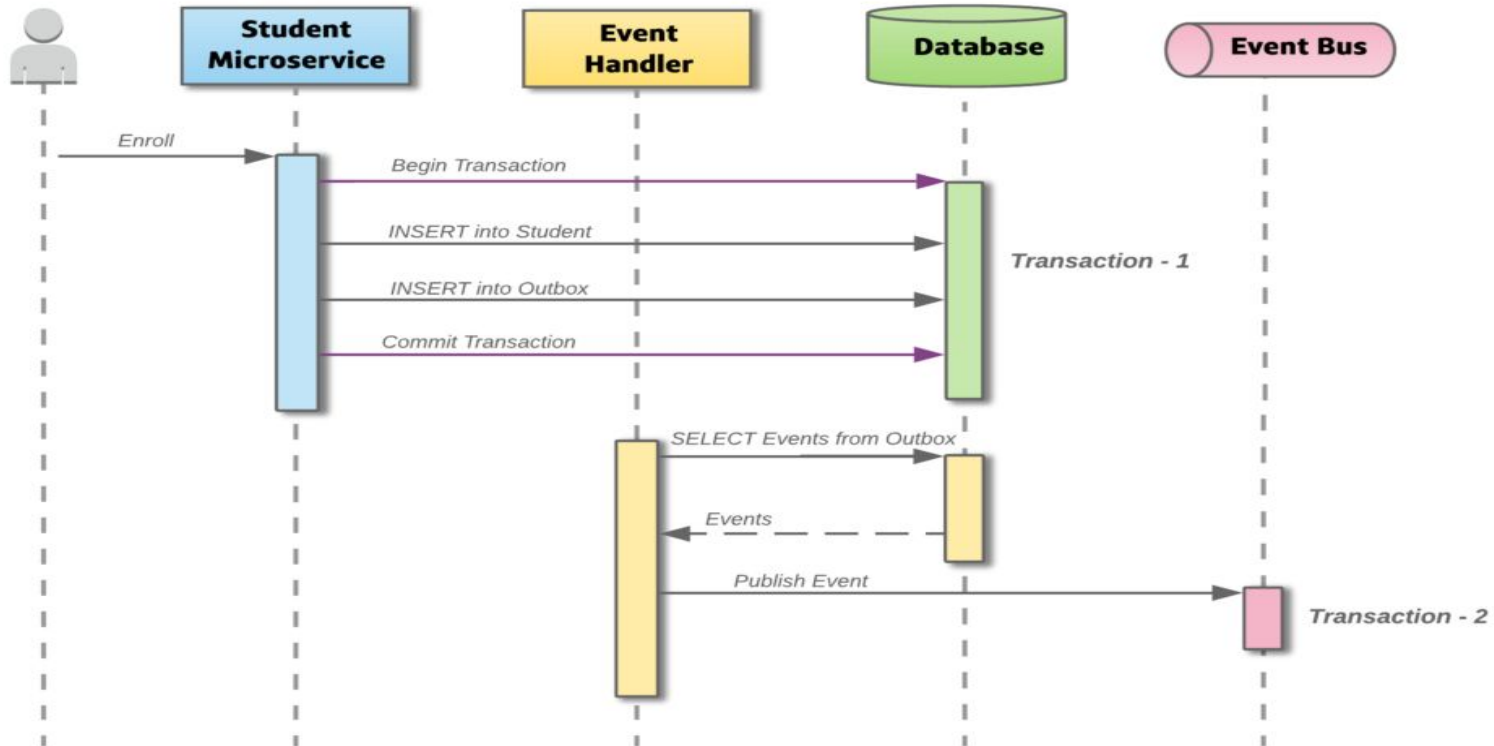
Transformaciones

- Se pueden configurar transformaciones en los mensajes que contienen el cambio de la base de datos.
- Se pueden usar antes de enviar el mensaje a Kafka o cuando son leídos por un conector de tipo sink.
- Debezium Server también permite el uso de transformaciones.
- Son:
 - Topic Routing
 - Content-Based Routing
 - New Record State Extraction
 - MongoDB New Document State Extraction
 - Outbox Event Router
 - MongoDB Outbox Event Router
 - Message Filtering

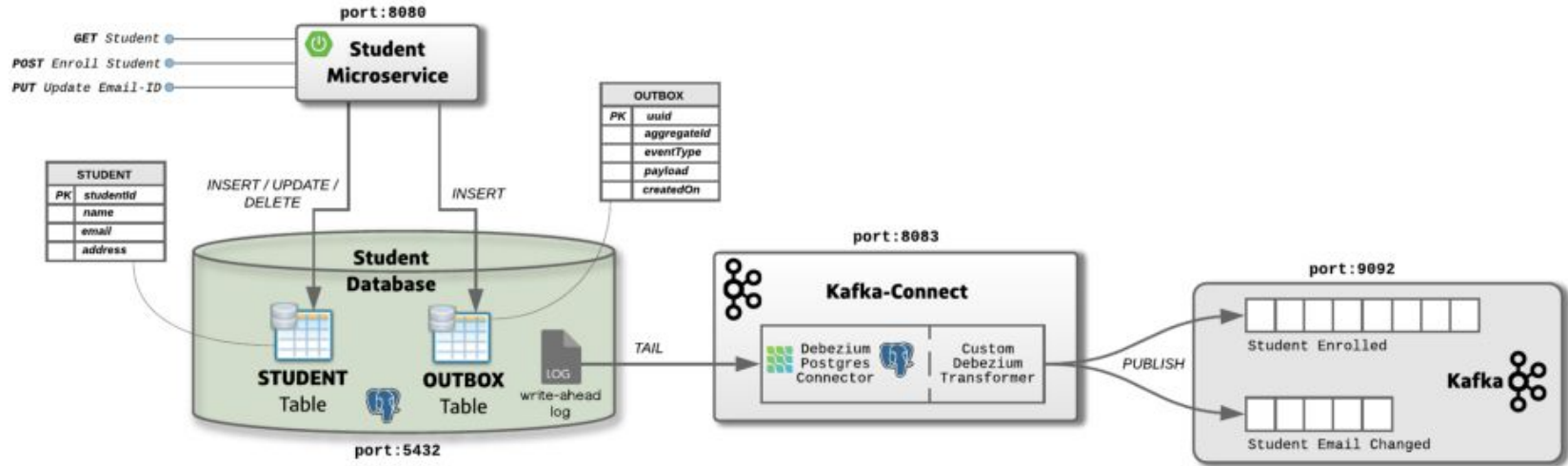
Transformaciones. Outbox Event Router



Transformaciones. Outbox Event Router



Transformaciones. Outbox Event Router



Demo

Referencias

- <https://debezium.io/documentation/reference/1.8/index.html>
- https://medium.com/@sohan_ganapathy/resilient-eventing-in-microservices-using-the-outbox-pattern-ed0b10ea3ef8