ApacheCon @ Home 2020

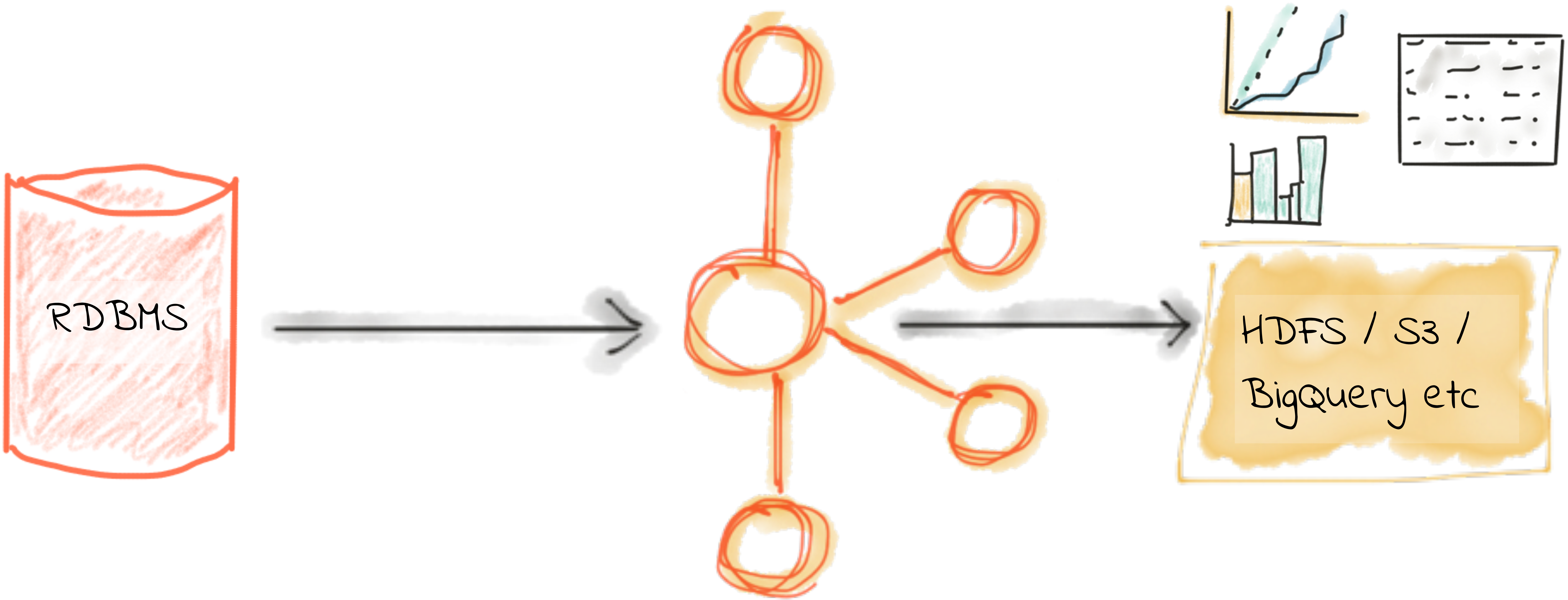
Integrating Databases and Apache Kafka®

@rmoff

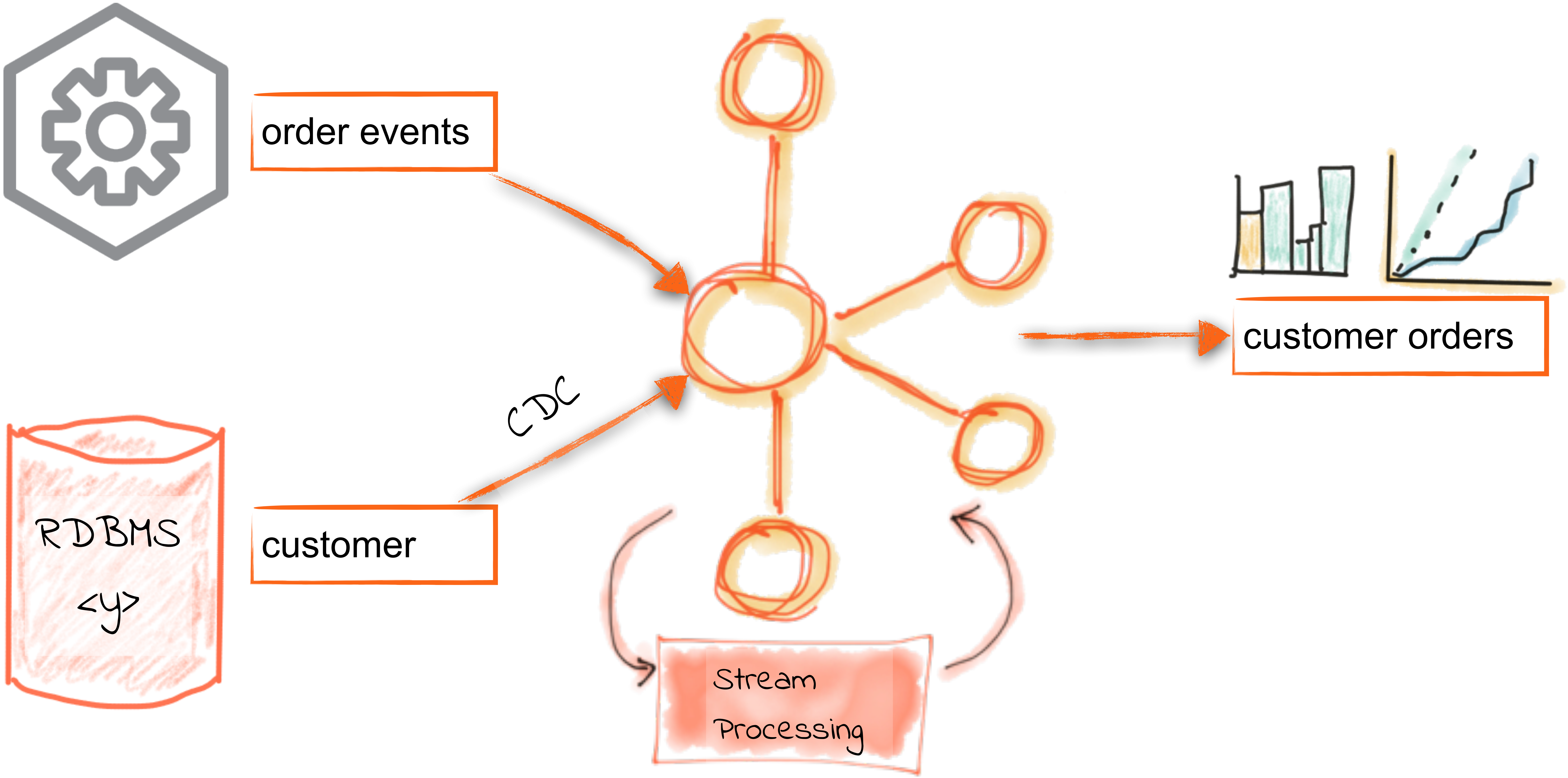
#acah2020

Photo by Emily Morter on Unsplash

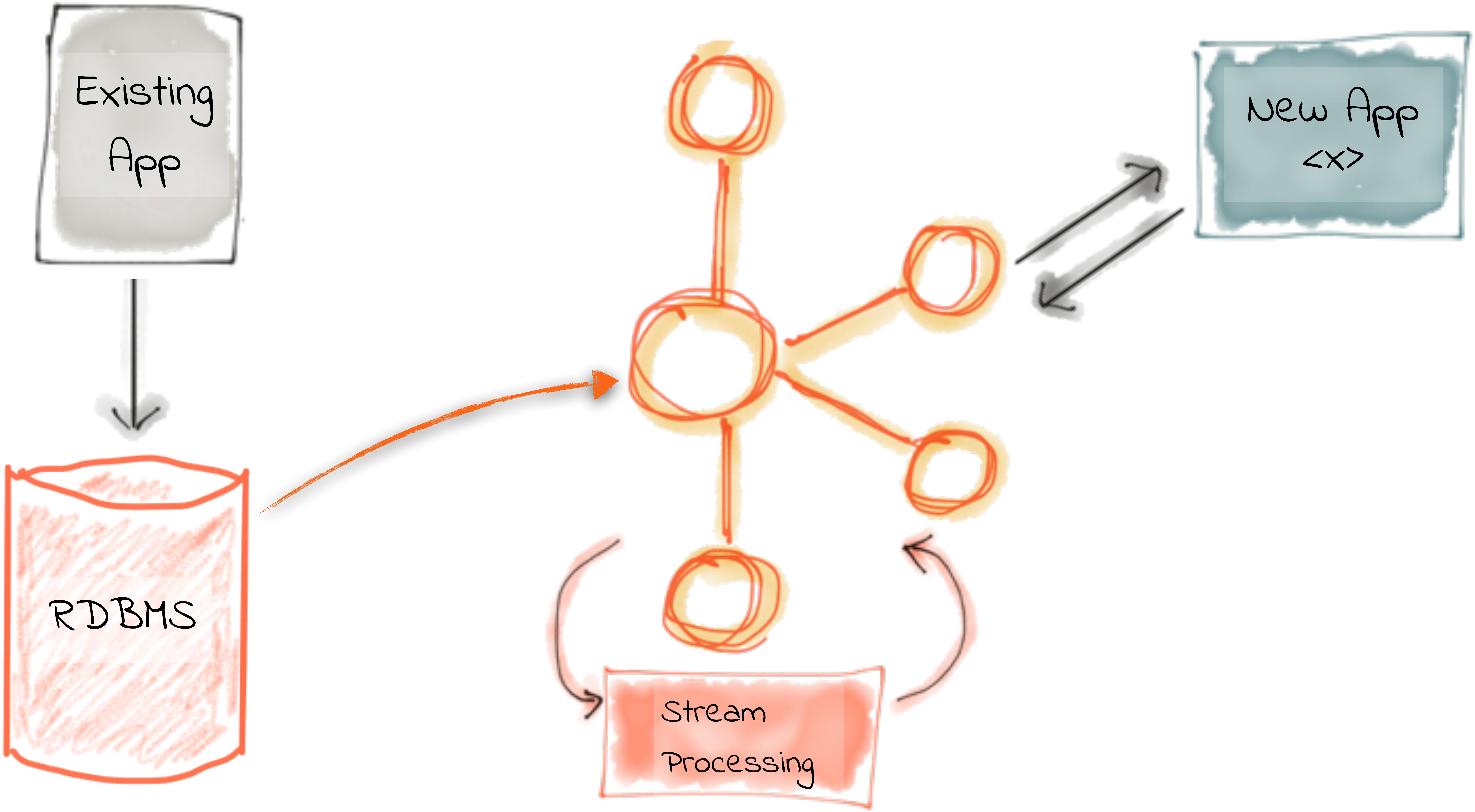
Analytics - Database Offload



Real-time Event Stream Enrichment

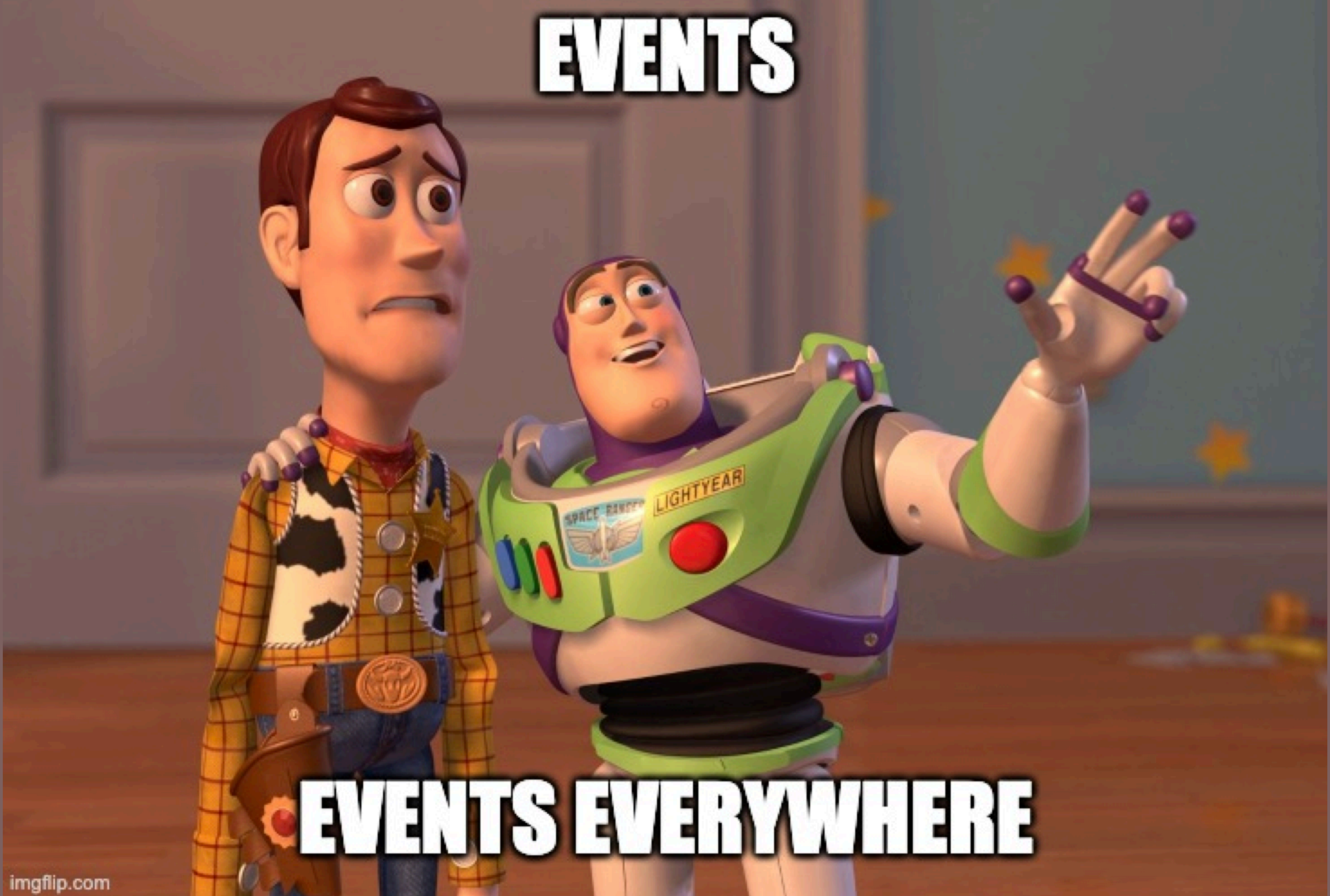


Evolve processing from old systems to new



"

But streaming…I've just got data in a database…right?


EVENTS
LIGHTYEAR
EVENTS EVERYWHERE
imgflip.com

"Bold claim: all your data is event streams

Human generated events

A Sale

A Stock movement

Machine generated events

Networking

IoT

Applications

Databases


Do you think that's a table
you are querying?

The Stream Table Duality

Account ID	Balance
12345	€50

The Stream Table Duality

Account ID	Balance
12345	€50

Account ID	Amount
12345	+ €50

Time

The Stream Table Duality

Time

Account ID	Amount
12345	+ €50
12345	+ €25

Account ID	Balance
12345	€75

The Stream Table Duality

Time

Account ID	Amount
12345	+ €50
12345	+ €25
12345	-€60

Account ID	Balance
12345	€15

The Stream Table Duality

Stream

Time

Account ID	Amount
12345	+ €50
12345	+ €25
12345	-€60

Table

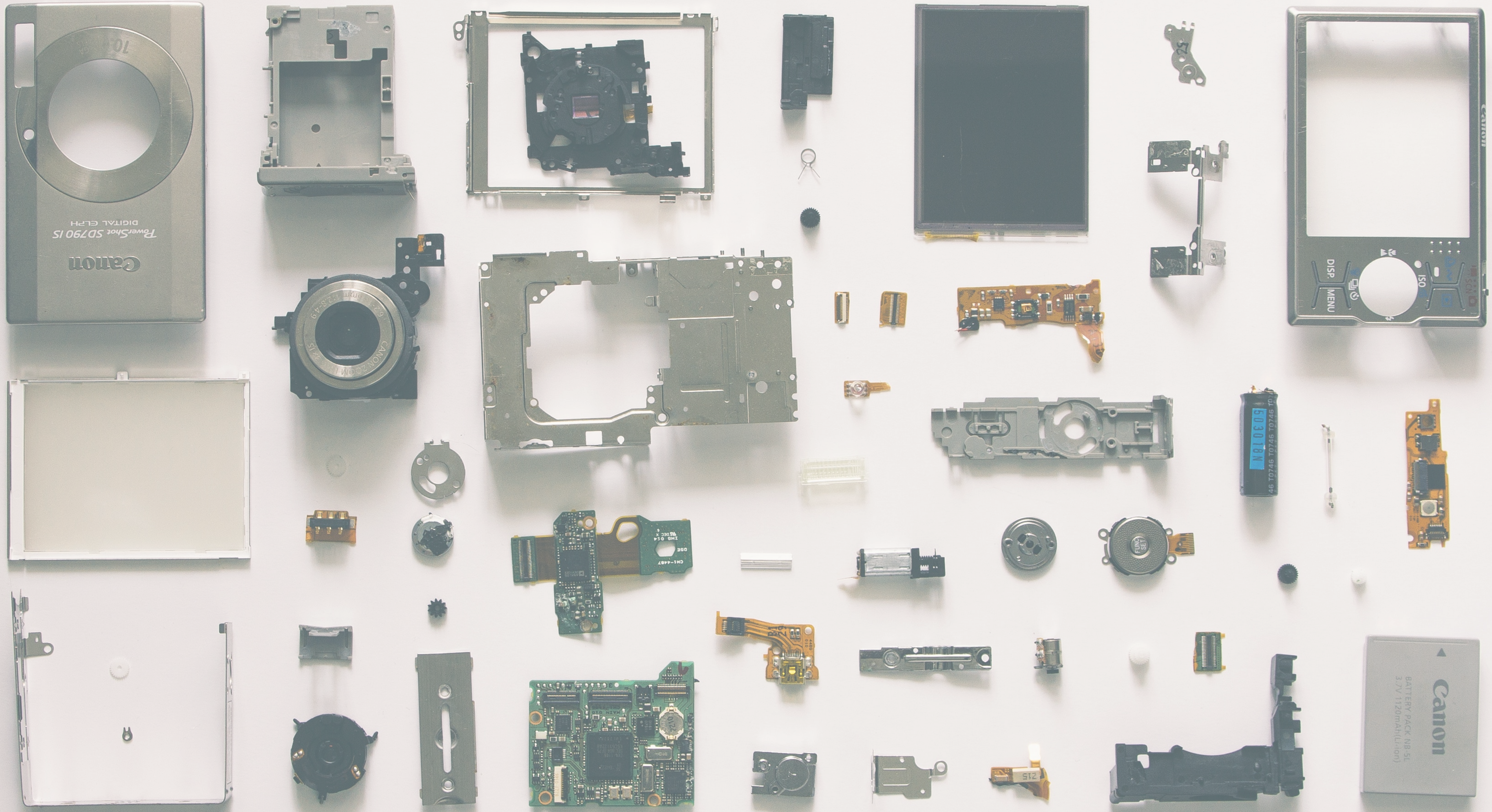
Account ID	Balance
12345	€15

The truth is the log.

The database is a cache of a subset of the log.

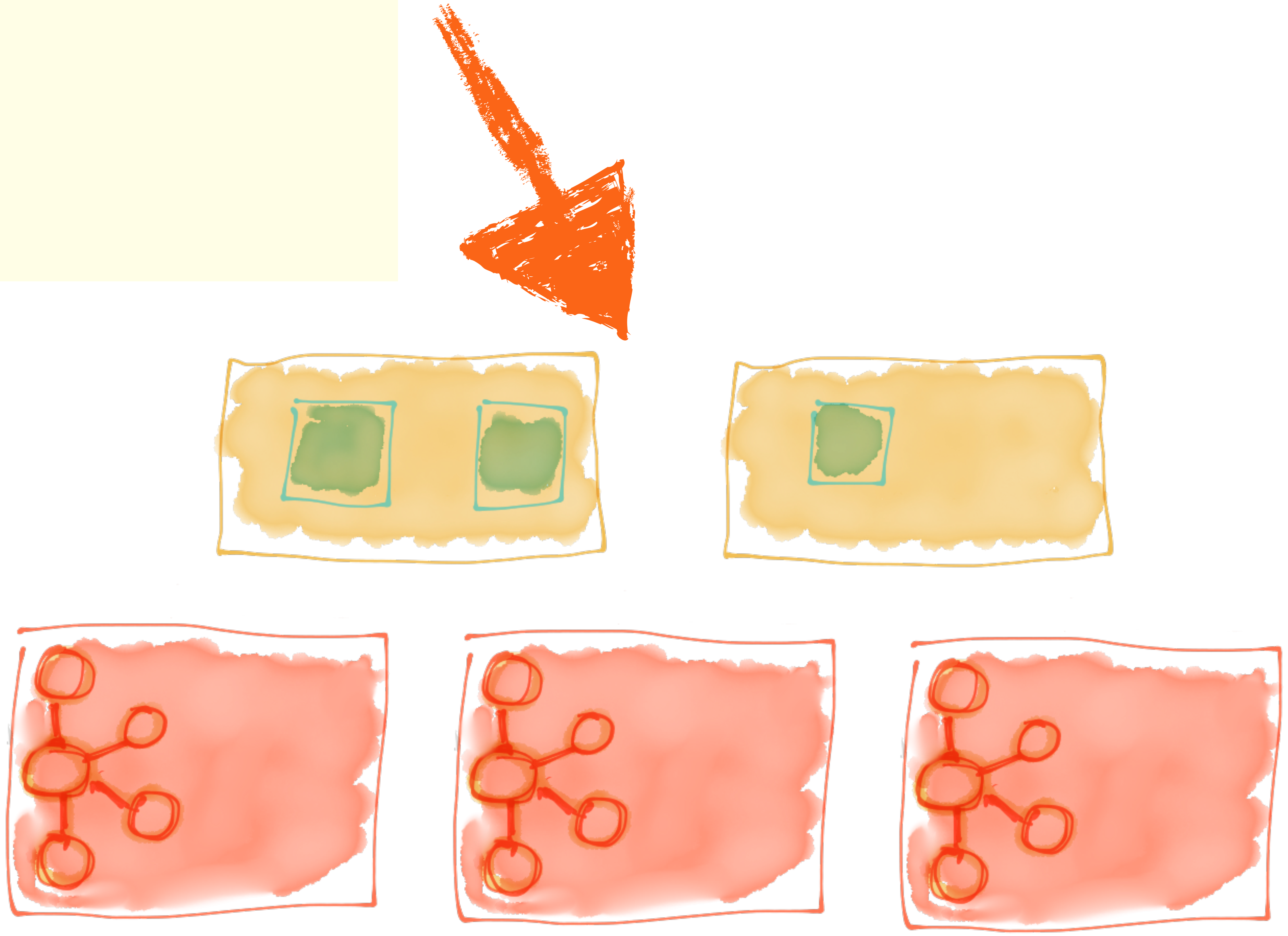
—Pat Helland
Immutability Changes Everything
http://cidrdb.org/cidr2015/Papers/CIDR15_Paper16.pdf

Photo by Bobby Burch on Unsplash


Canon
PowerShot SD790 IS
DIGITAL ELPH
DISP.
MENU
ISO
FUNC SET
Canon
BATTERY PACK NB-5L

Streaming Integration with Kafka Connect

Sources

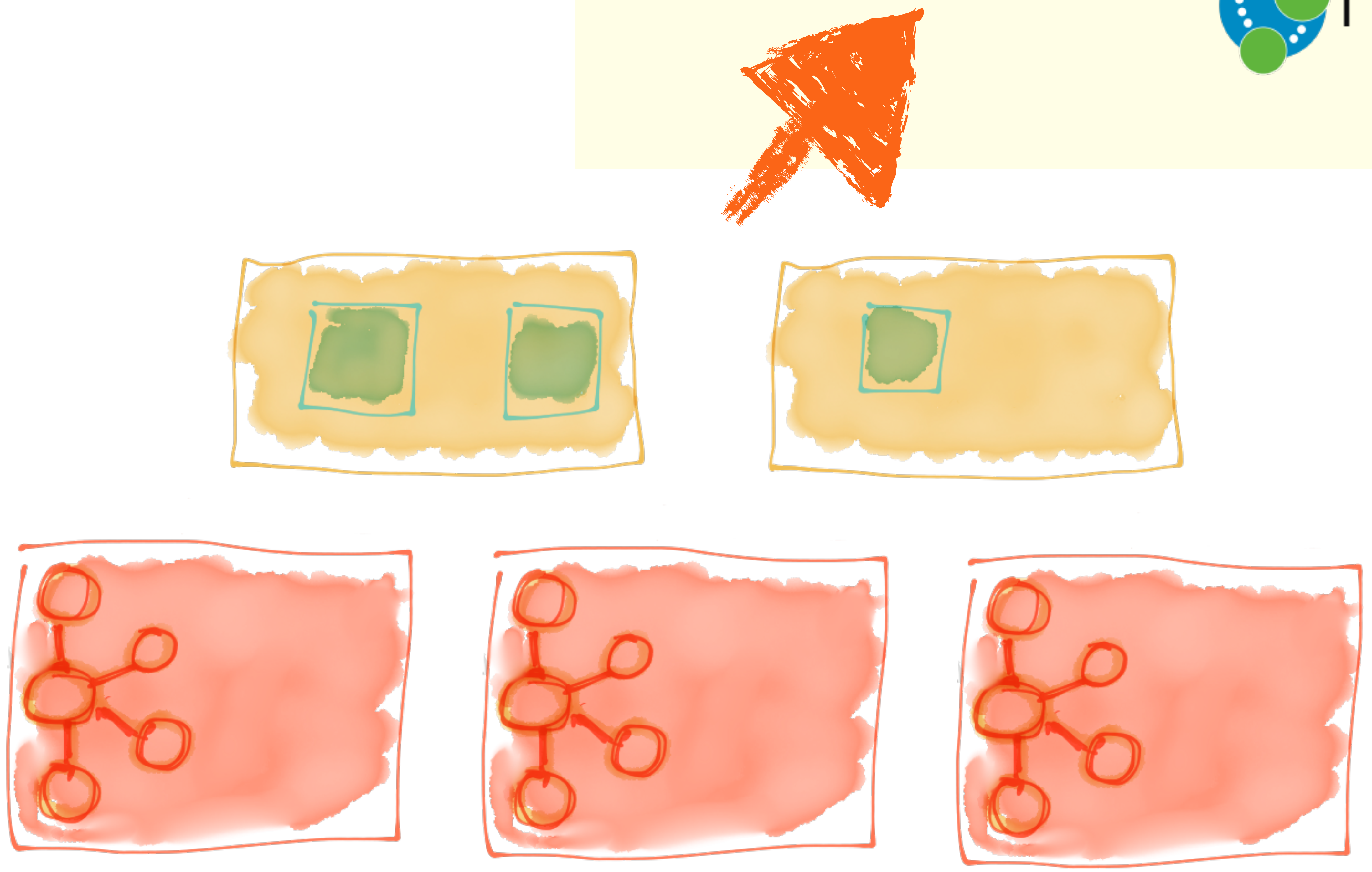


Kafka Connect

Kafka Brokers

Streaming Integration with Kafka Connect

Sinks



Kafka Connect

Kafka Brokers

Streaming Integration with Kafka Connect

Look Ma, No Code!

```
{
  "connector.class":
    "io.confluent.connect.jdbc.JdbcSourceConnector",
  "connection.url":
    "jdbc:mysql://asgard:3306/demo",
  "table.whitelist":
    "sales,orders,customers"
}
```

Kafka Connect basics

Connectors

Connectors


Connector
Native data
Connect Record
Source
Kafka Connect
Kafka

Converters


Connector
Converter
Native data
Connect
Record
bytes[]
Source
Kafka Connect
Kafka

Serialisation & Schemas

Avro 👍

Protobuf 👍

JSON Schema 👍

JSON 😬

CSV

Gwen (Chen) Shapira
@gwenshap

If your dev process doesn't validate schema compatibility somewhere between your IDE and production - you are screwed and don't know it.

5:50 AM - 5 Apr 2017

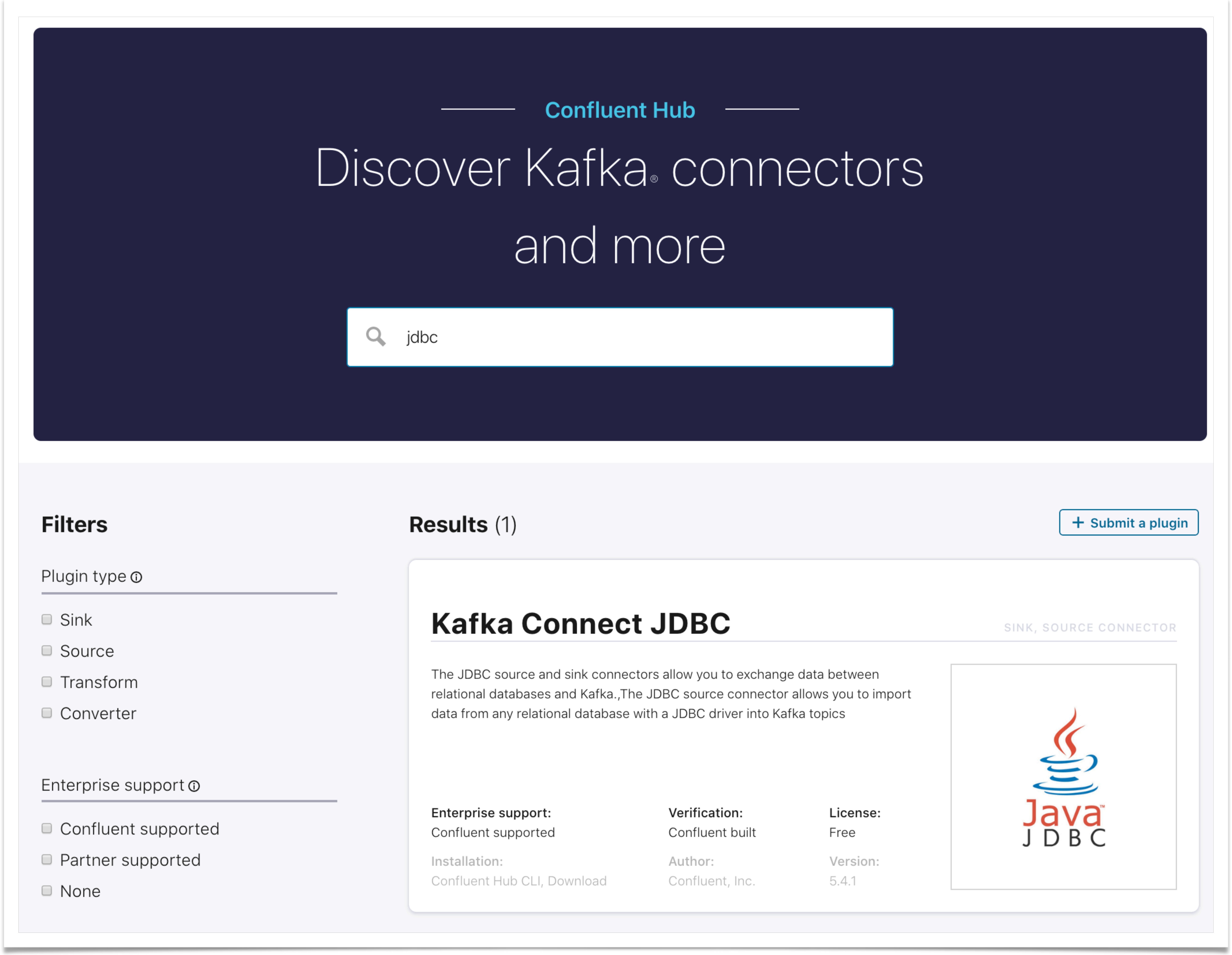
https://rmoff.dev/qcon-schemas

The Confluent Schema Registry

Single Message Transforms

Extensible

hub.confluent.io

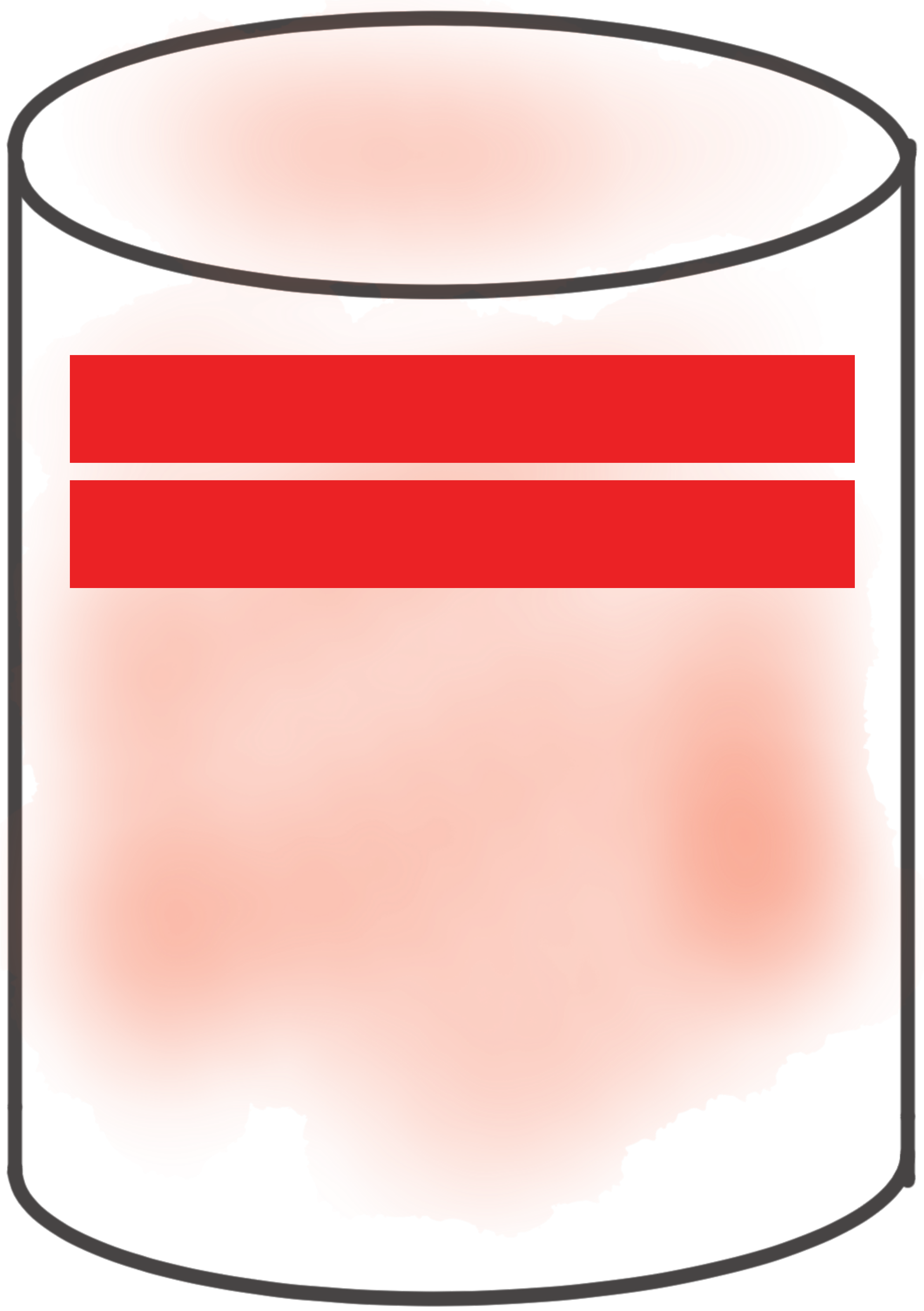



Change-Data-Capture (CDC)
Query-based
Log-based

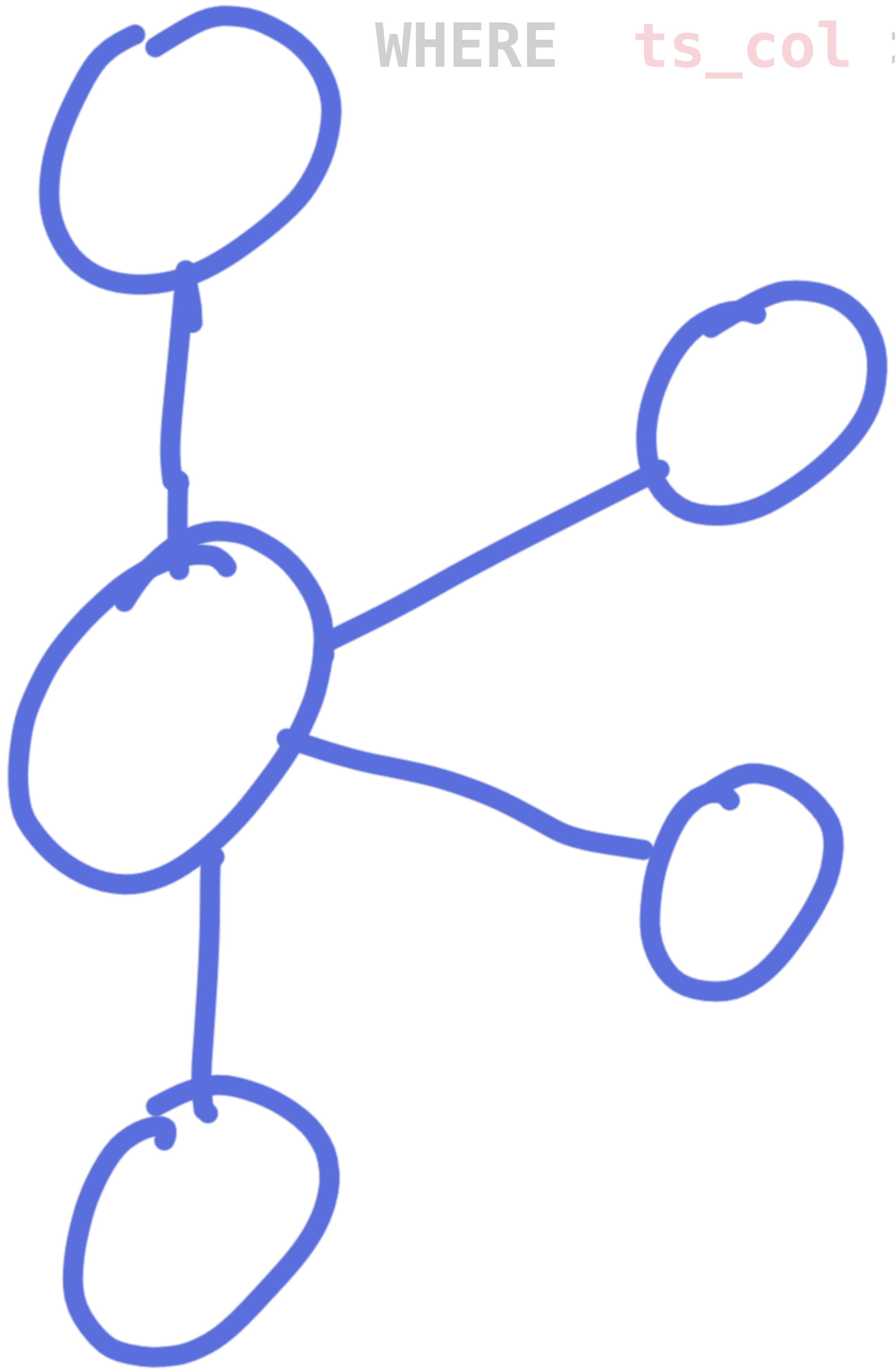
Query-based CDC

```
SELECT *
FROM   my_table
WHERE  ts_col > previous ts
```

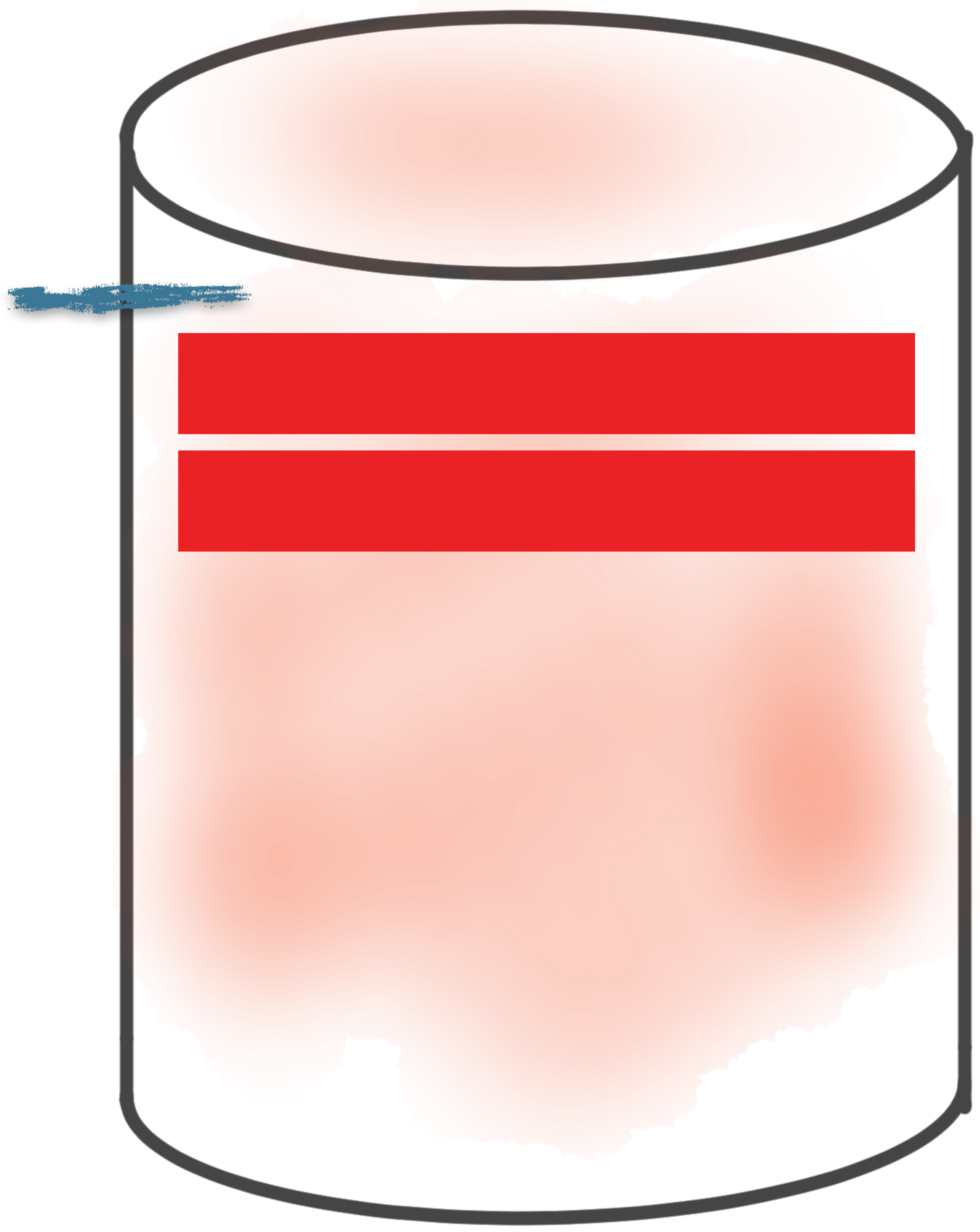
Query-based CDC



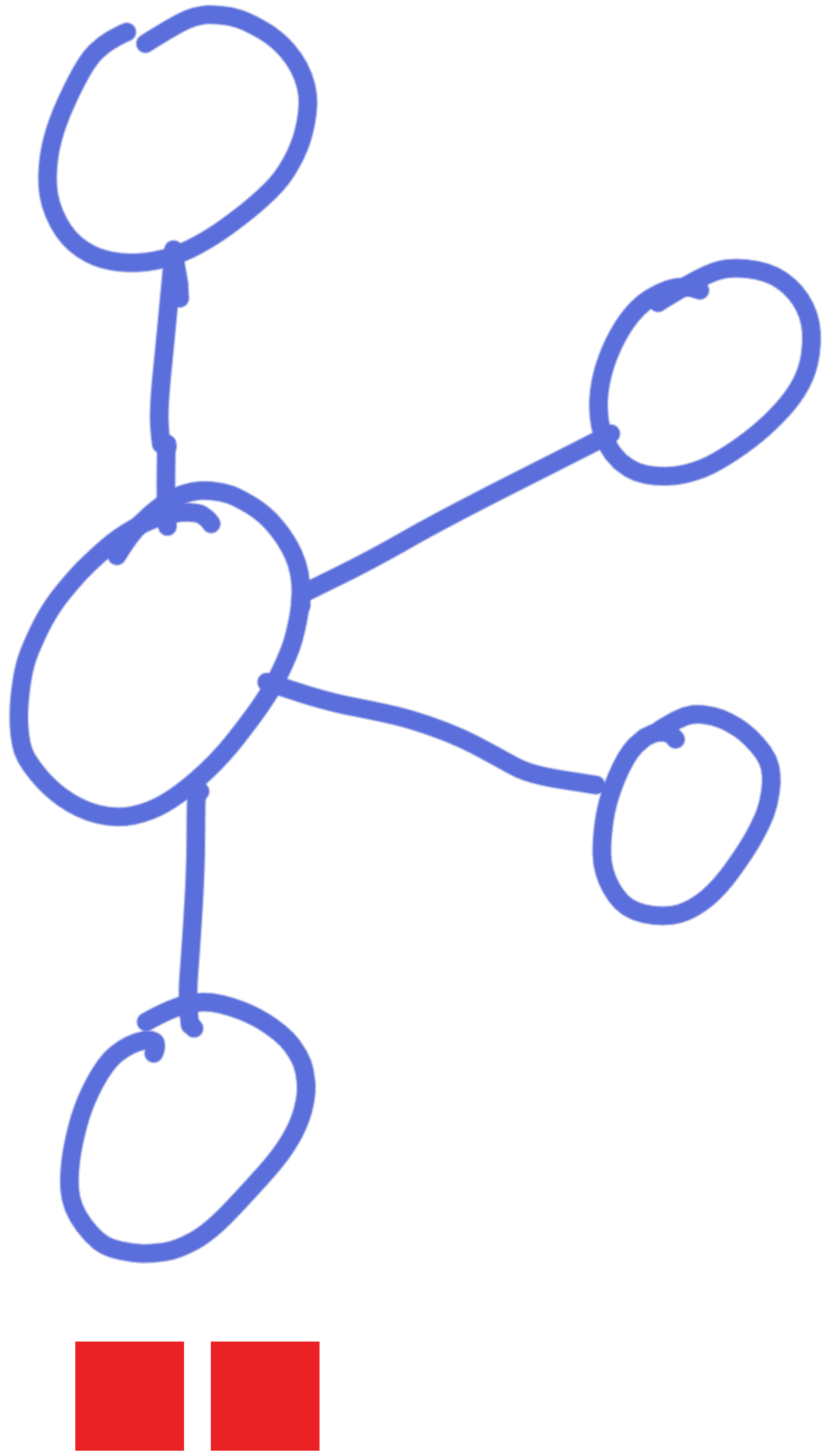
```
SELECT *
FROM   my_table
WHERE  ts_col > previous ts
```



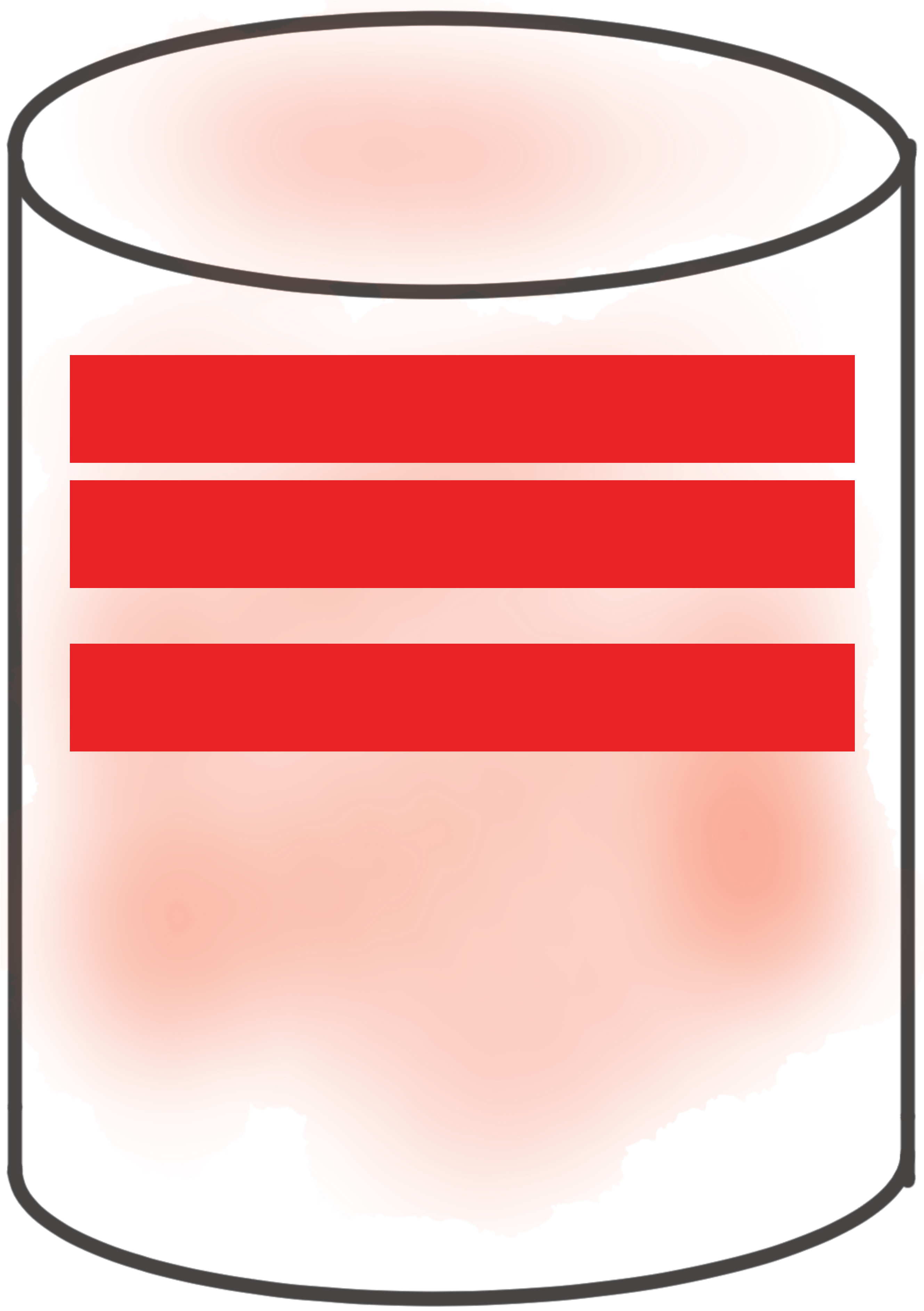
Query-based CDC



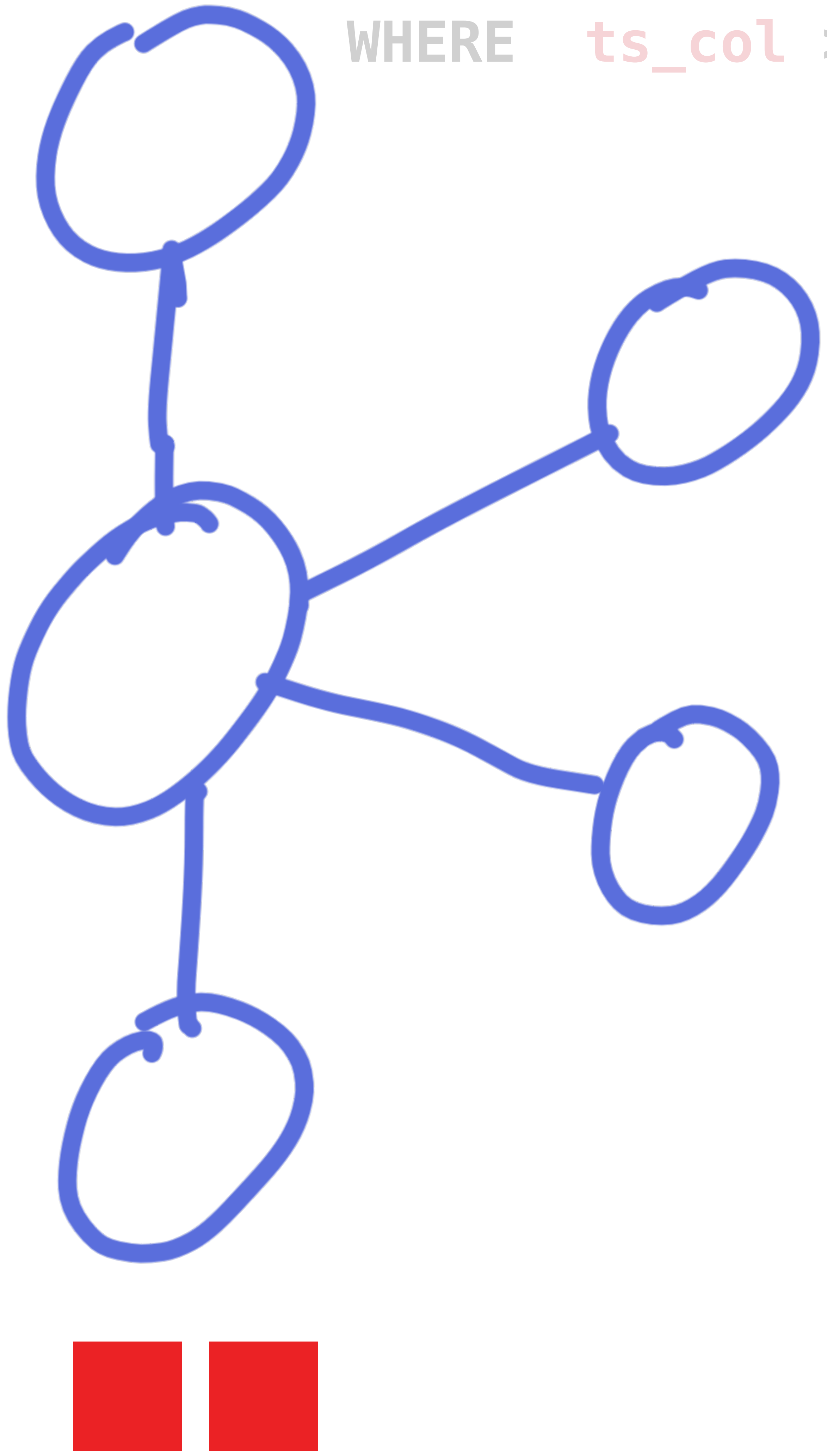
```
SELECT *
FROM   my_table
WHERE  ts_col >
        previous ts
```



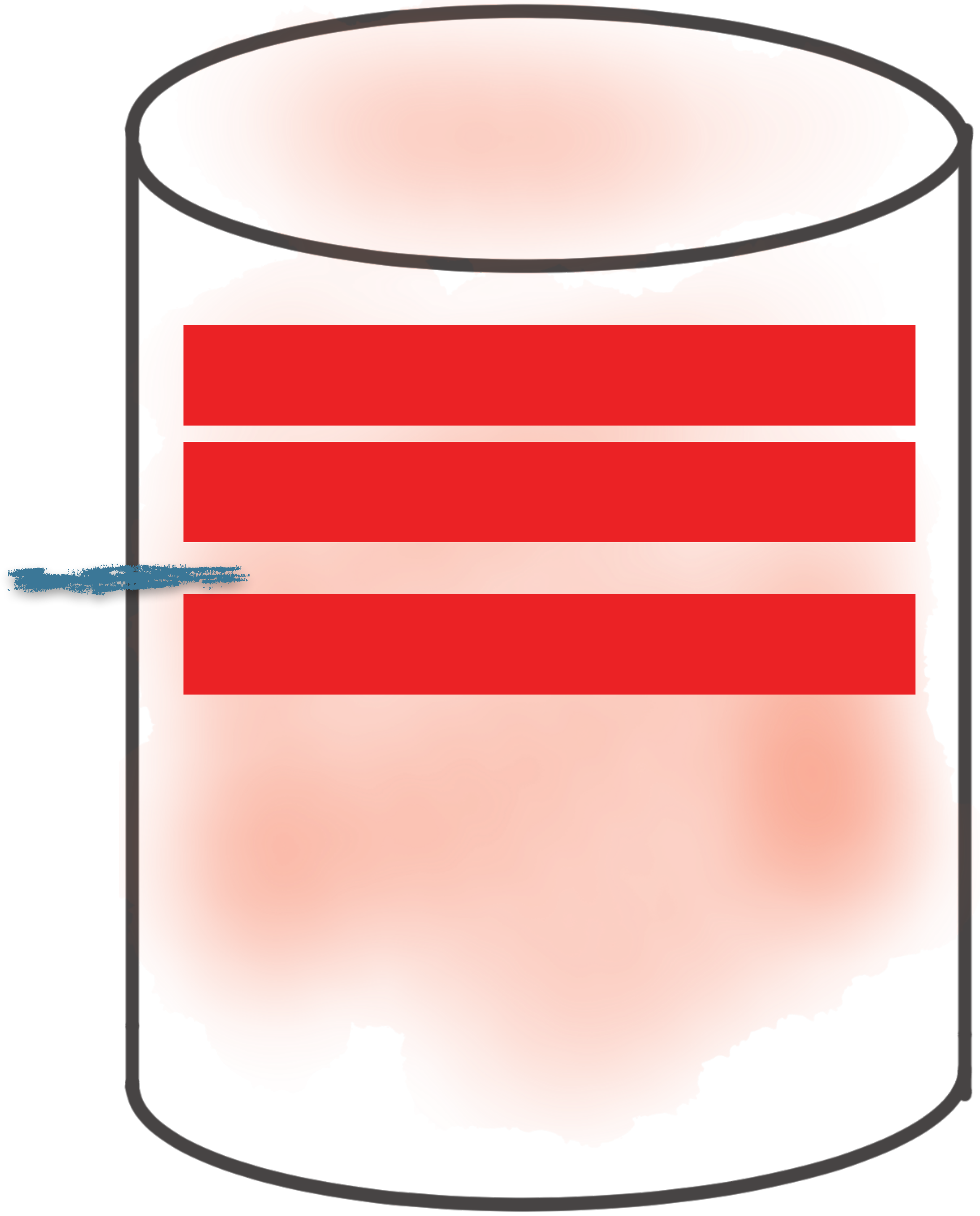
Query-based CDC



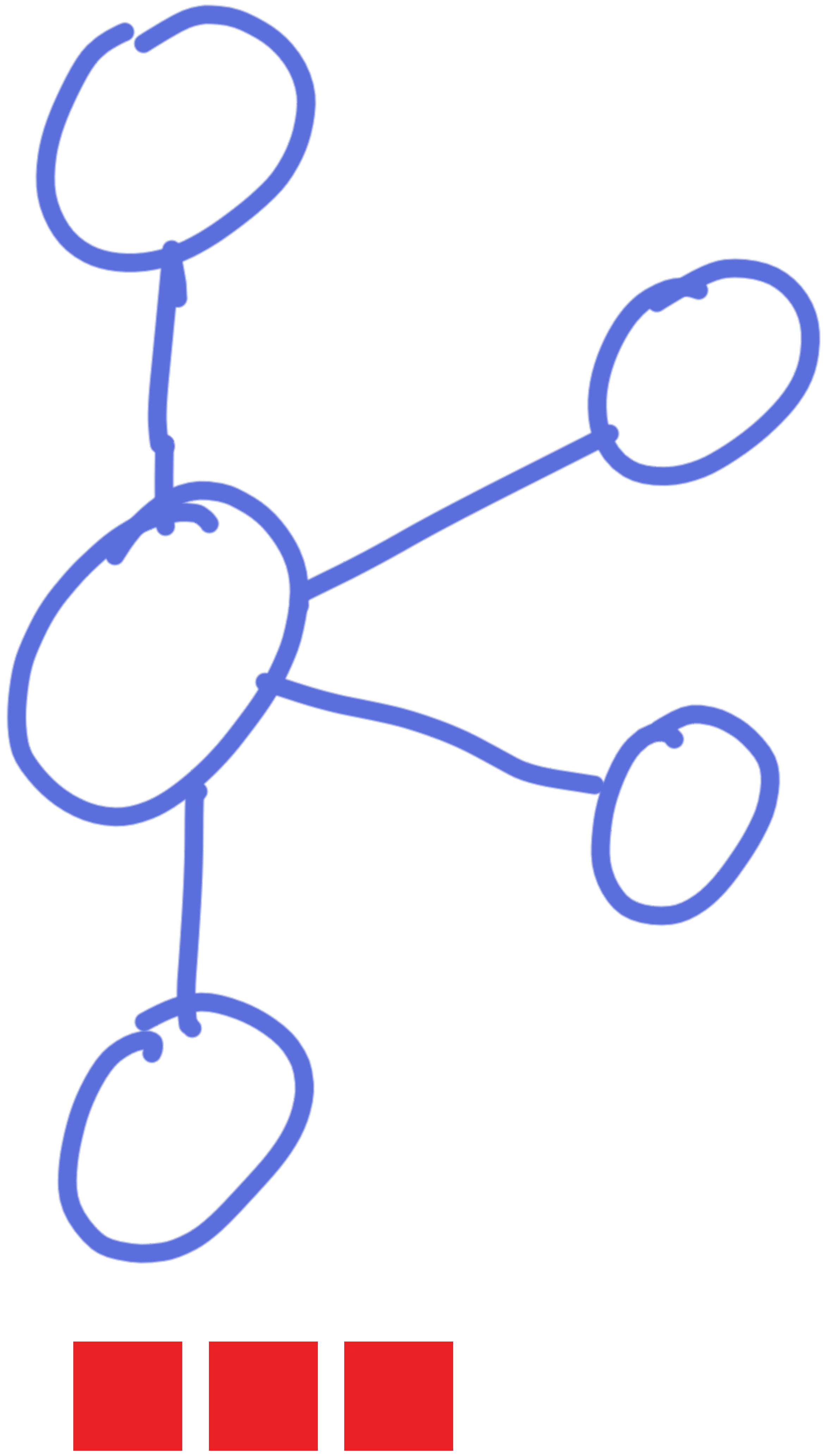
```
SELECT *
FROM   my_table
WHERE  ts_col > previous ts
```



Query-based CDC



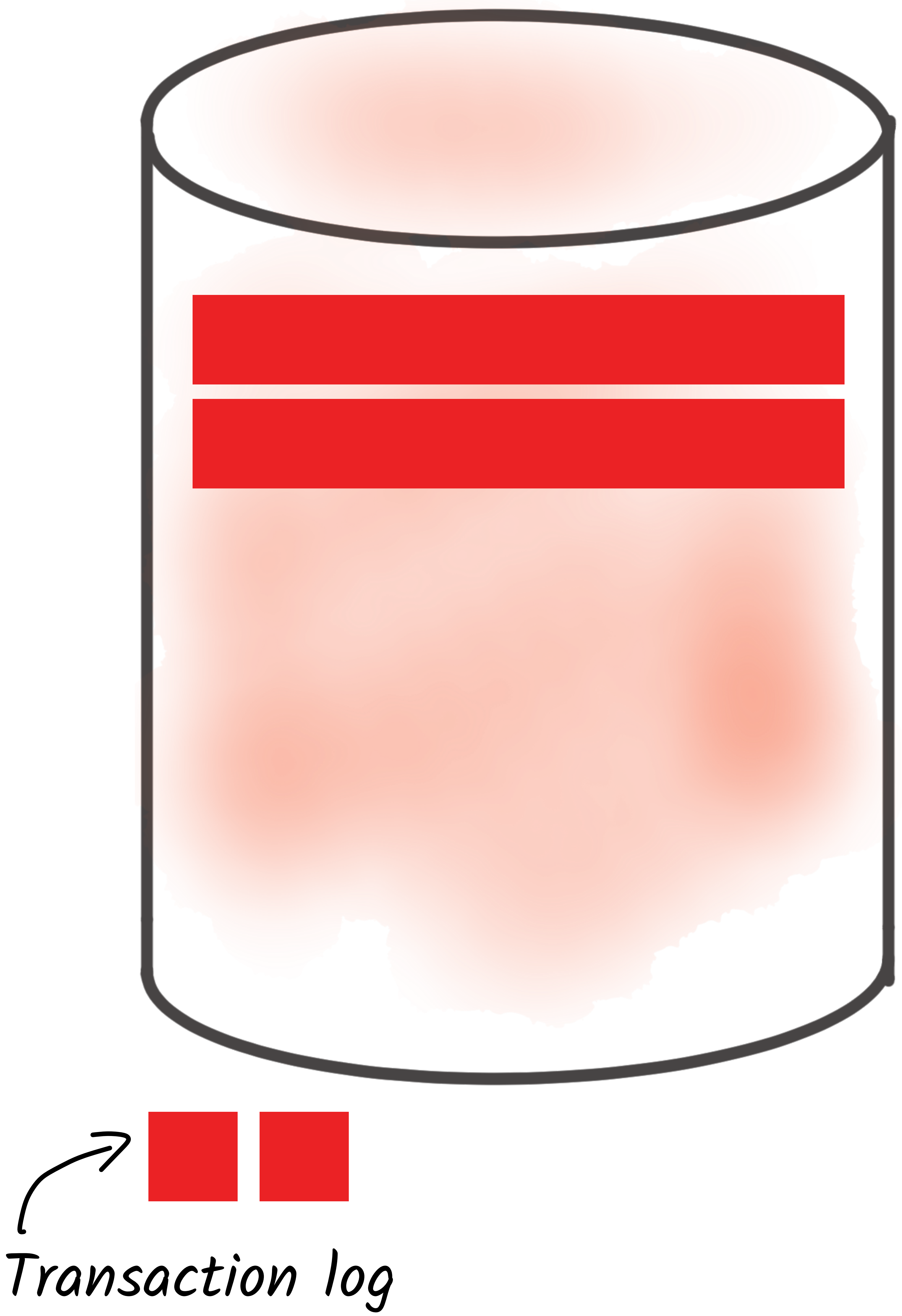
```sql
SELECT *
FROM   my_table
WHERE  ts_col >
        previous ts
```

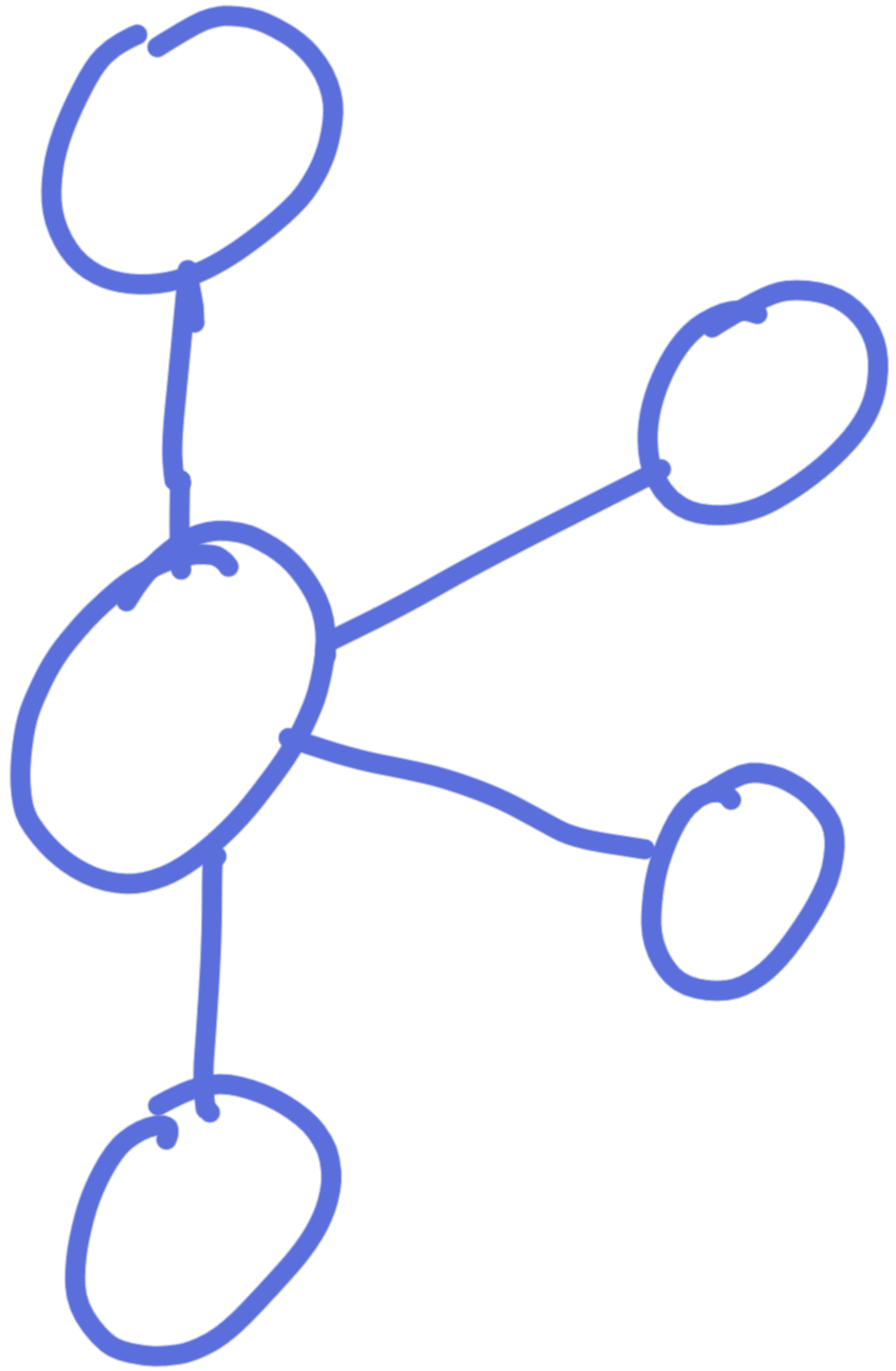


Log-based CDC

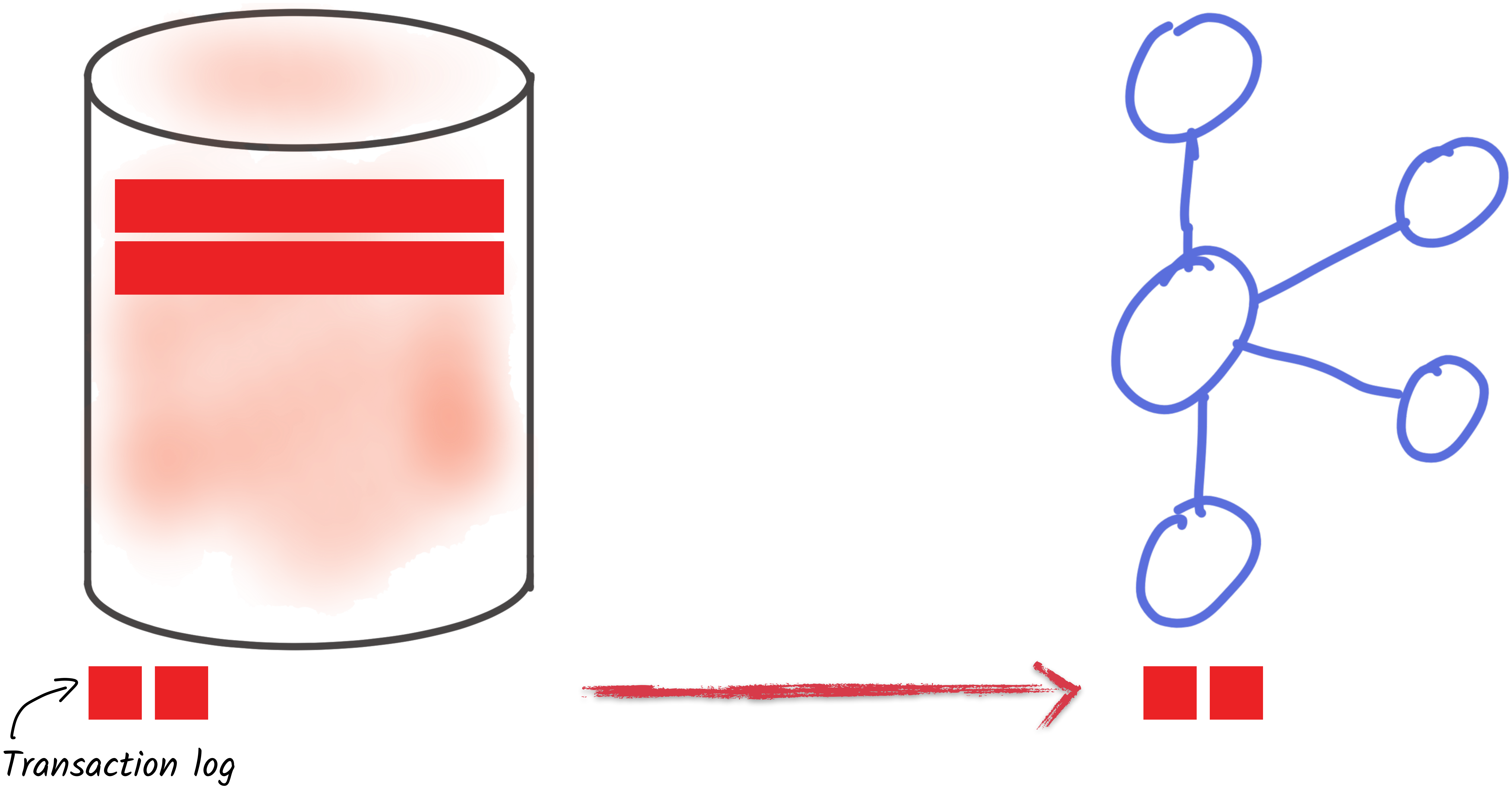
```
#051024 17:24:13 server id 1  end_log_pos 98
# Position  Timestamp   Type   Master ID        Size      Master Pos    Flags
# 00000004 9d fc 5c 43   0f   01 00 00 00   5e 00 00 00   62 00 00 00   00 00
# 00000017 04 00 35 2e 30 2e 31 35  2d 64 65 62 75 67 2d 6c |..5.0.15.debug.l|
# 00000027 6f 67 00 00 00 00 00 00  00 00 00 00 00 00 00 00 |og..............|
# 00000037 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 |................|
# 00000047 00 00 00 00 9d fc 5c 43  13 38 0d 00 08 00 12 00 |.......C.8......|
# 00000057 04 04 04 04 12 00 00 4b  00 04 1a                |.......K...|
```

Log-based CDC

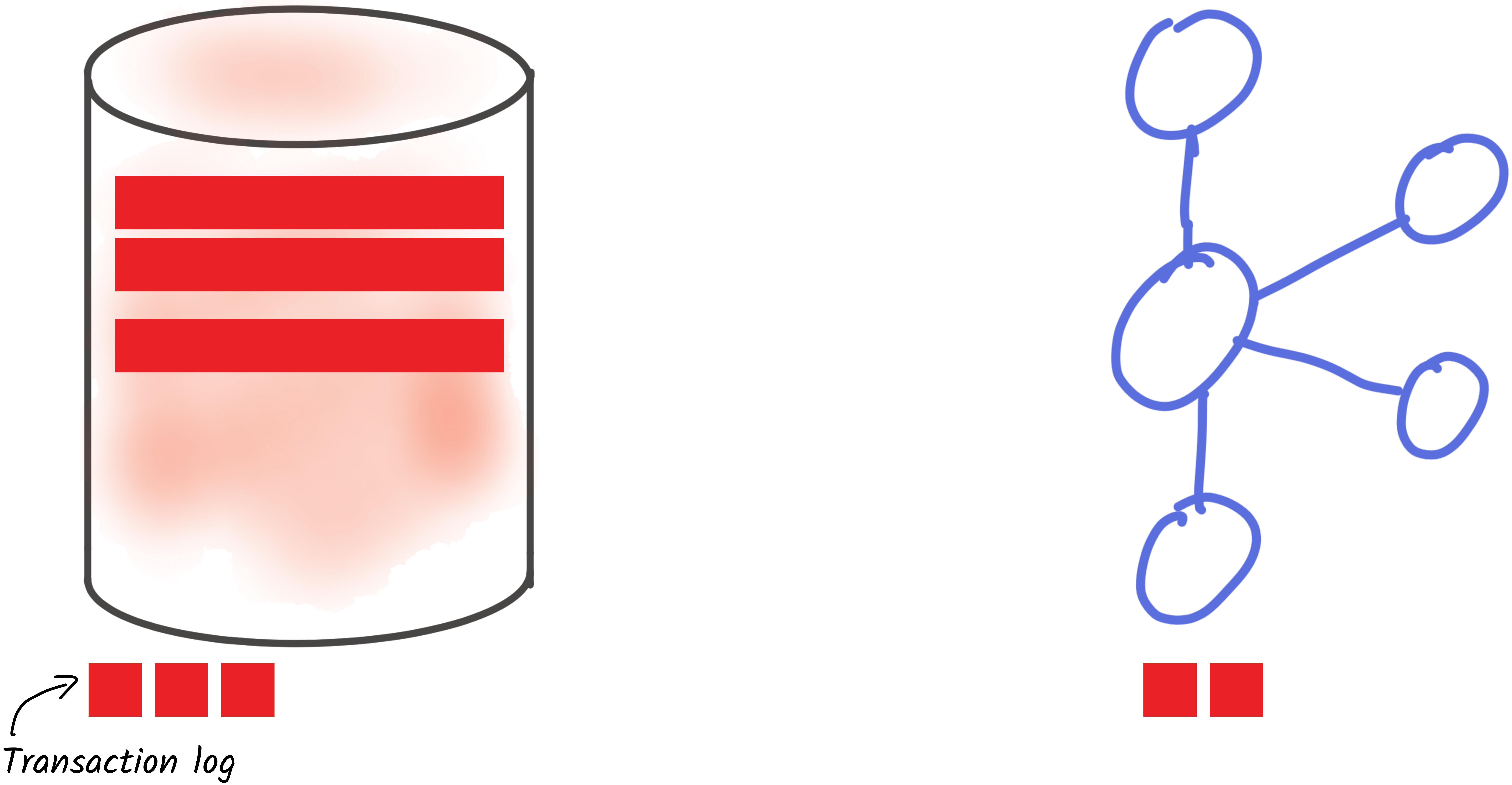




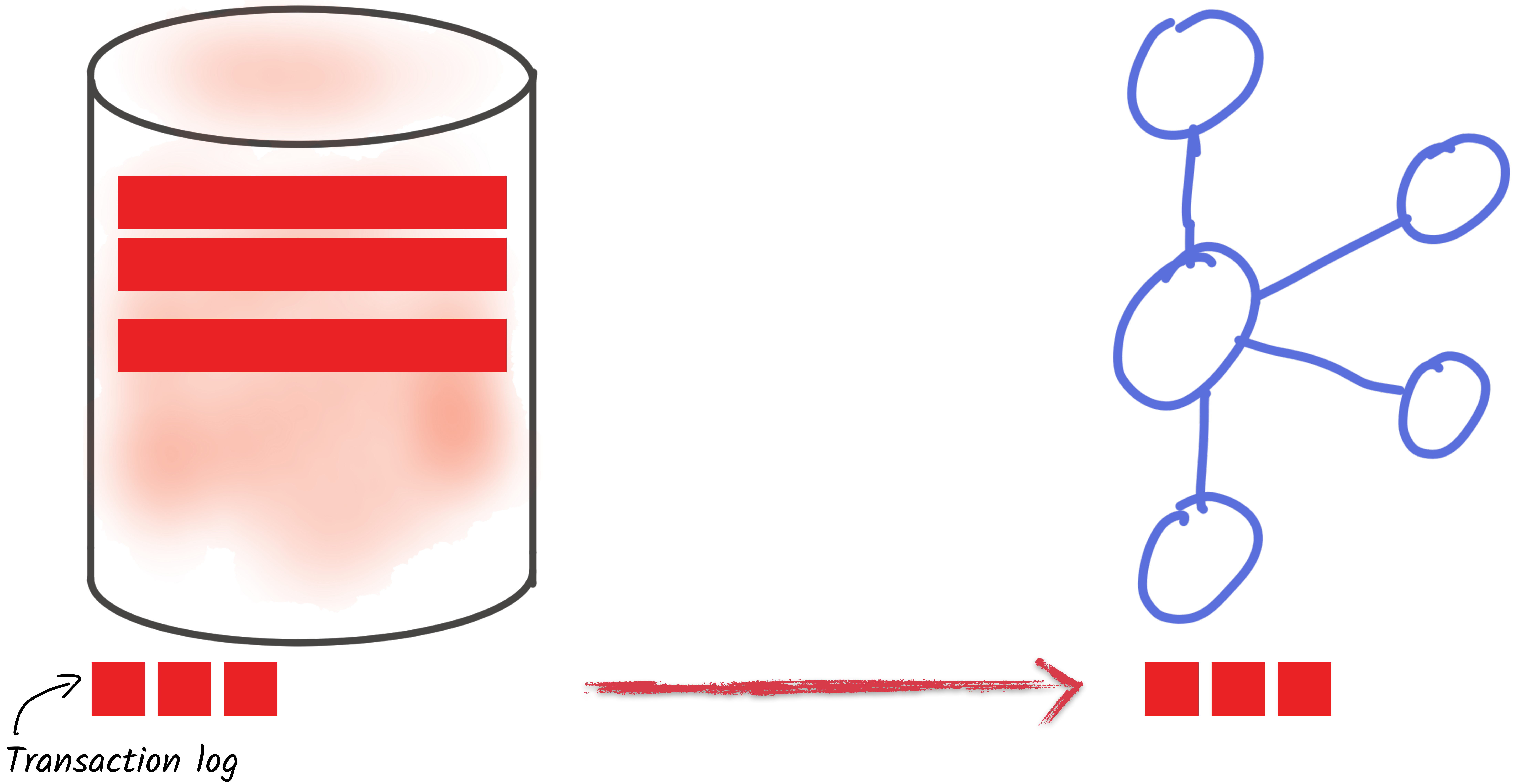
Log-based CDC



Log-based CDC



Log-based CDC



Demo Time!

https://rmoff.dev/cdc-demo


Query-based CDC
Log-based CDC
JAKE-CLARK.TUMBLR
imgflip.com

Query-based CDC

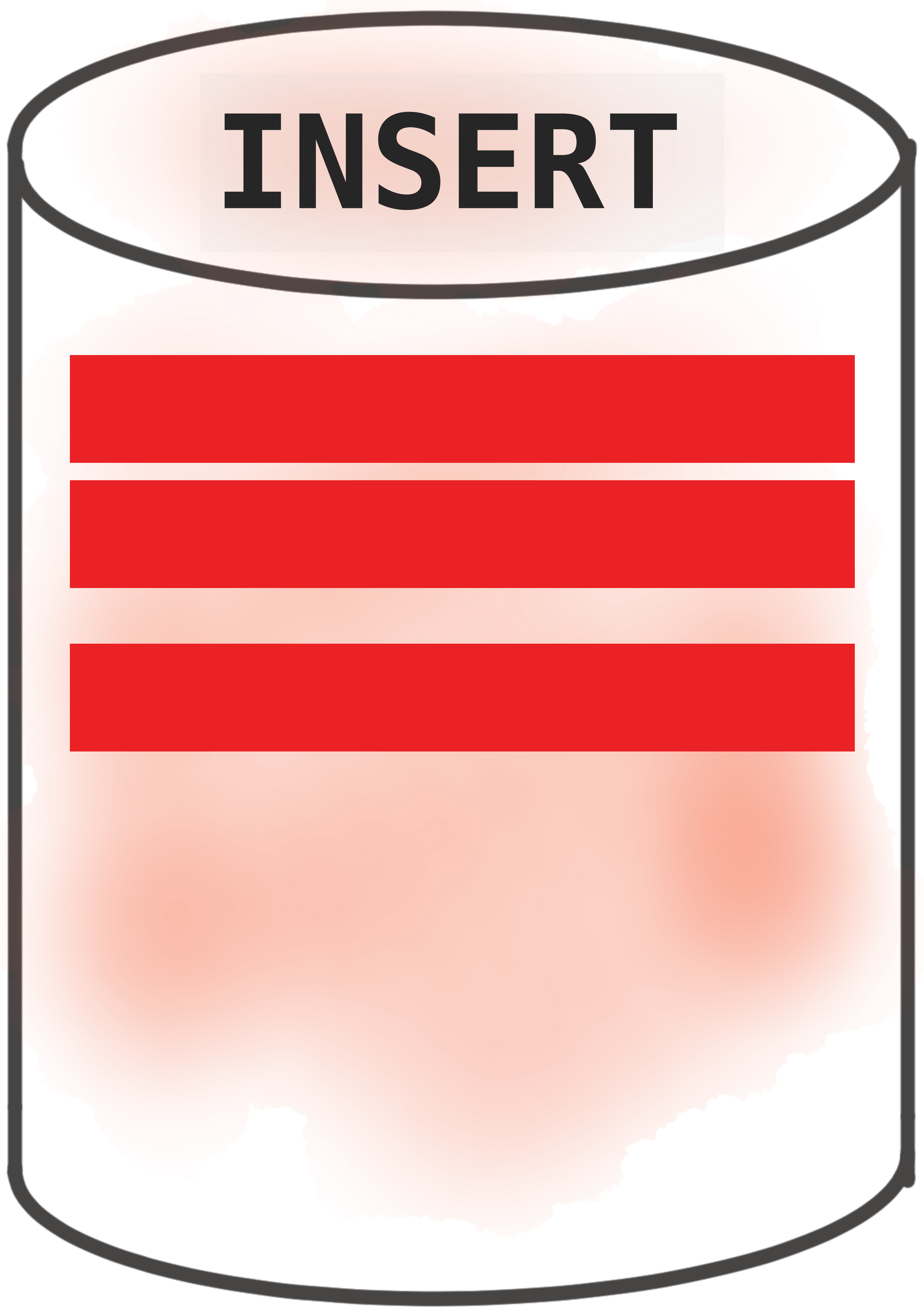
```
SELECT *
FROM   my_table
WHERE  ts_col > previous ts
```

Query-based CDC

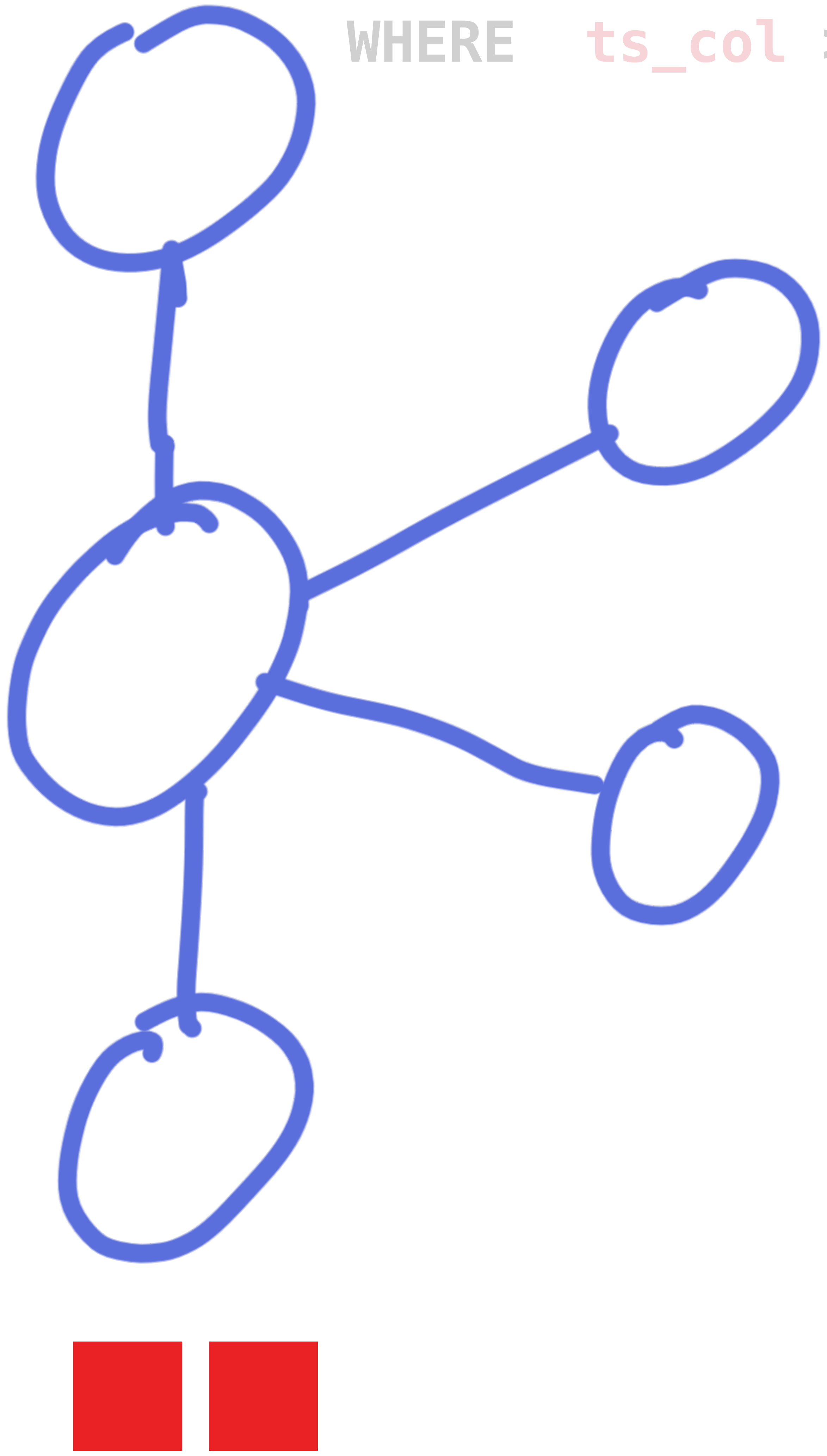
```
SELECT *
FROM   my_table
WHERE  ts_col > previous ts
```

```
CREATE TABLE my_table (
  ID INT,
  FOO VARCHAR,
  BAR VARCHAR,
  WIBBLE VARCHAR,
  TS_COL TIMESTAMP
    DEFAULT CURRENT_TIMESTAMP
    ON UPDATE CURRENT_TIMESTAMP
)
```

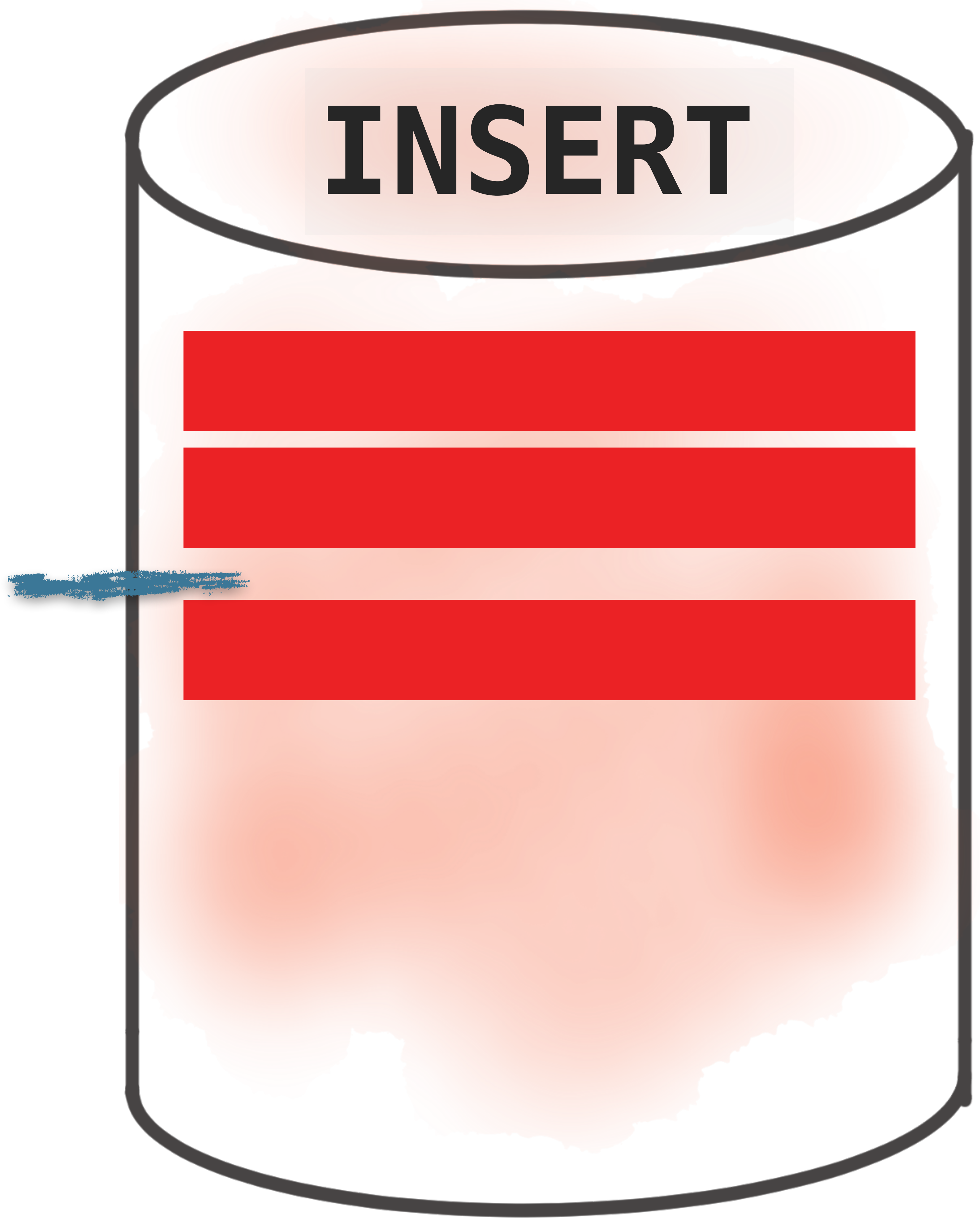
Query-based CDC



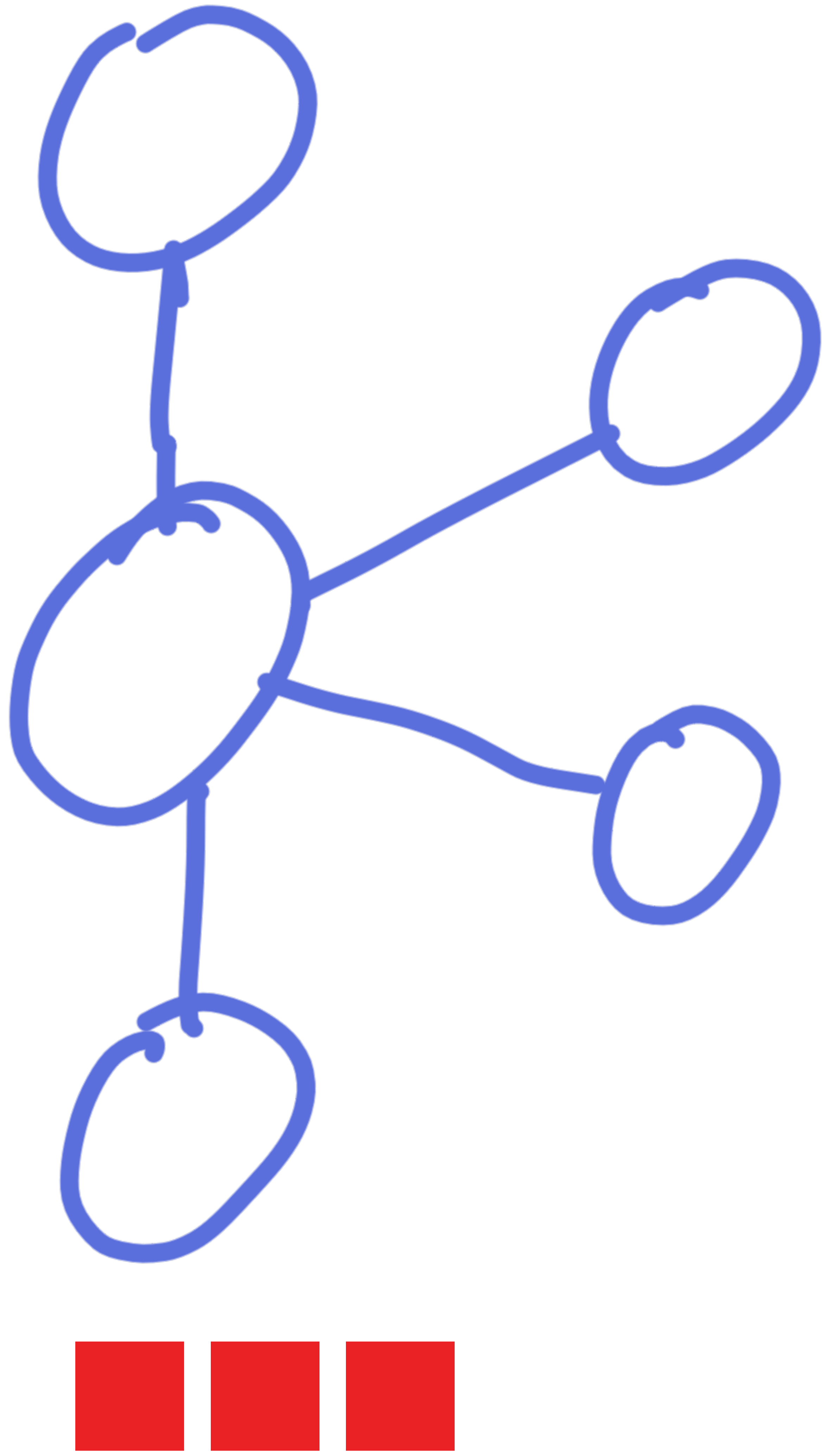

```
SELECT *
FROM   my_table
WHERE  ts_col > previous ts
```



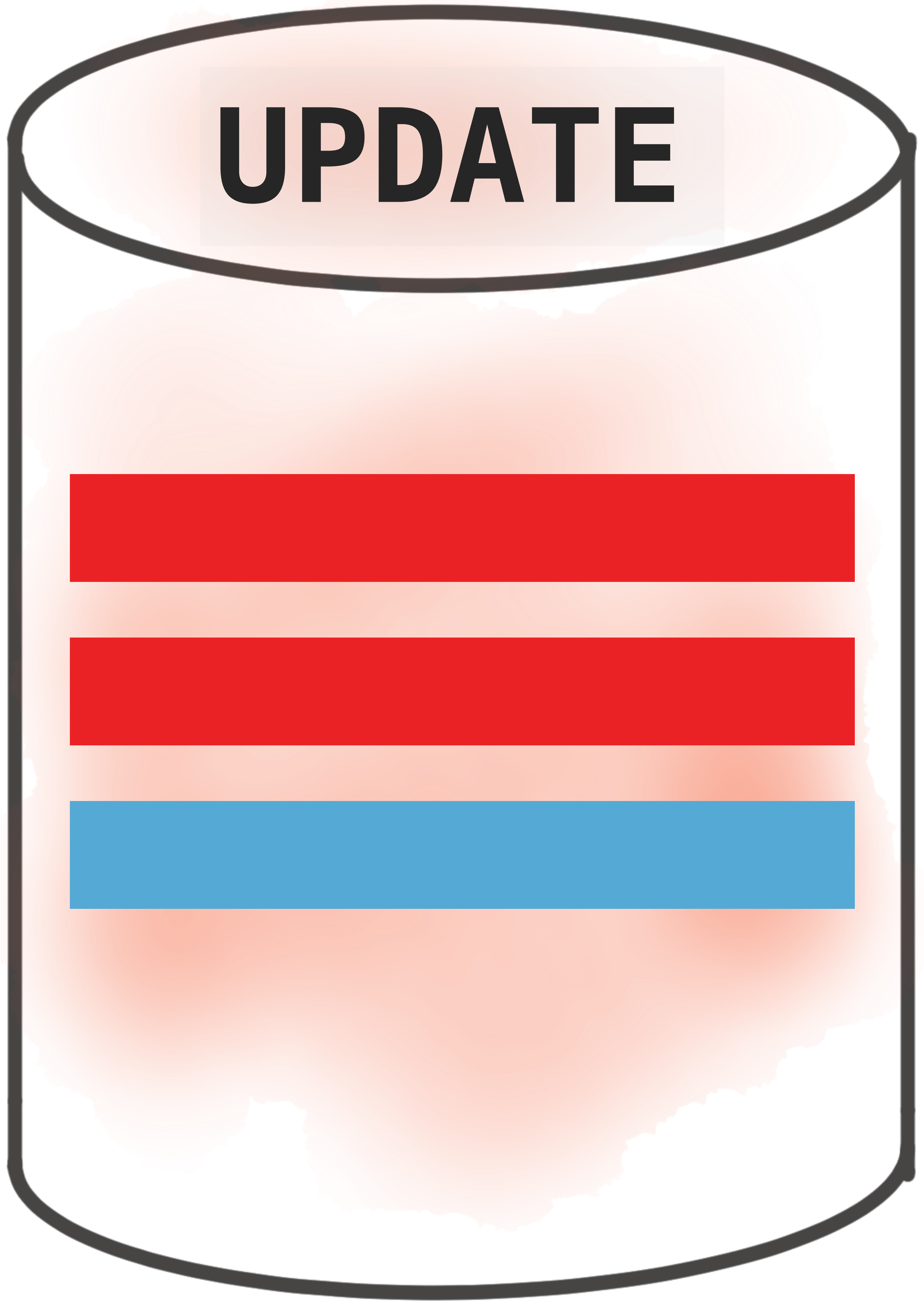
Query-based CDC

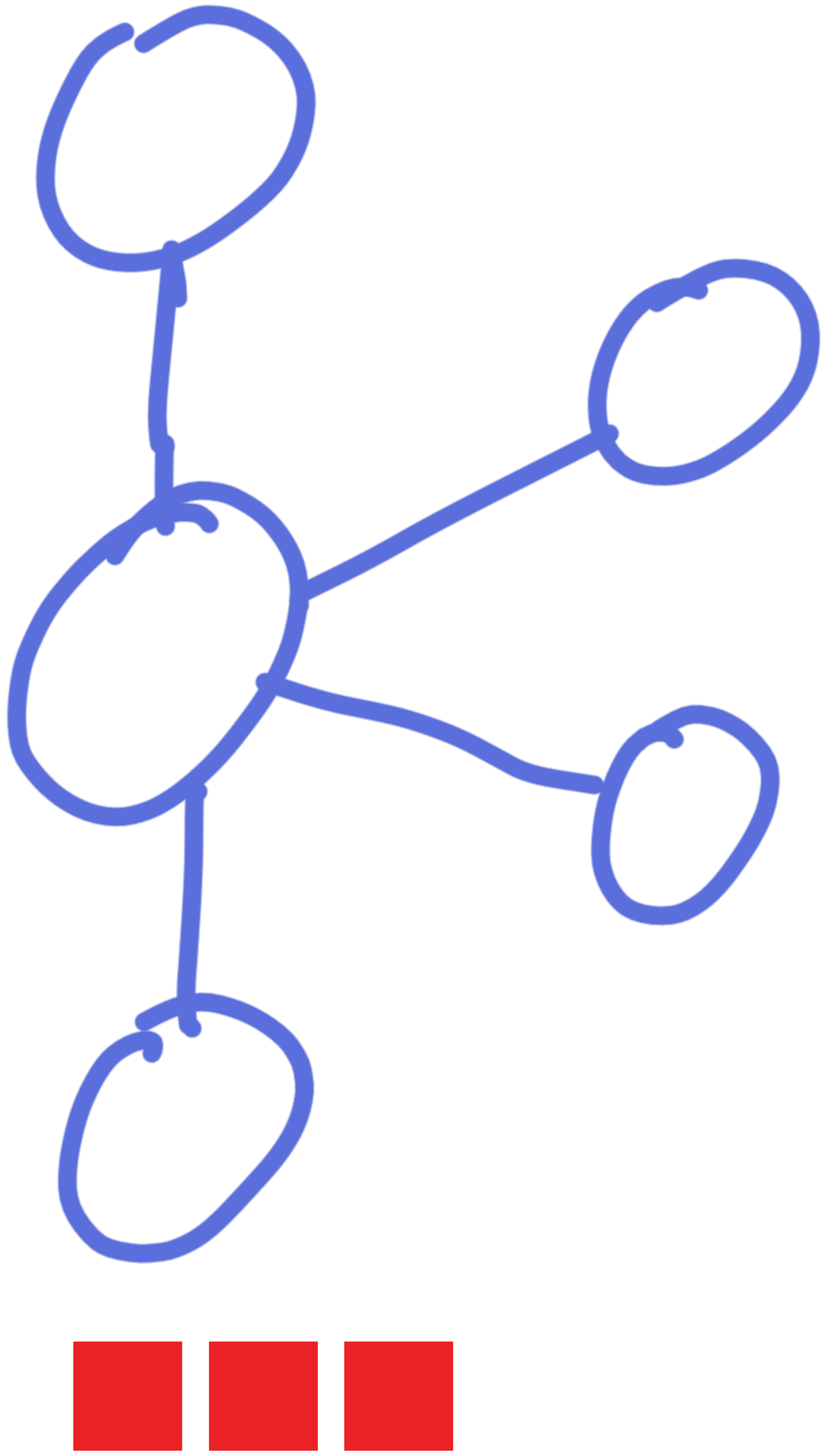



```
SELECT *
FROM   my_table
WHERE  ts_col >
        previous ts
```

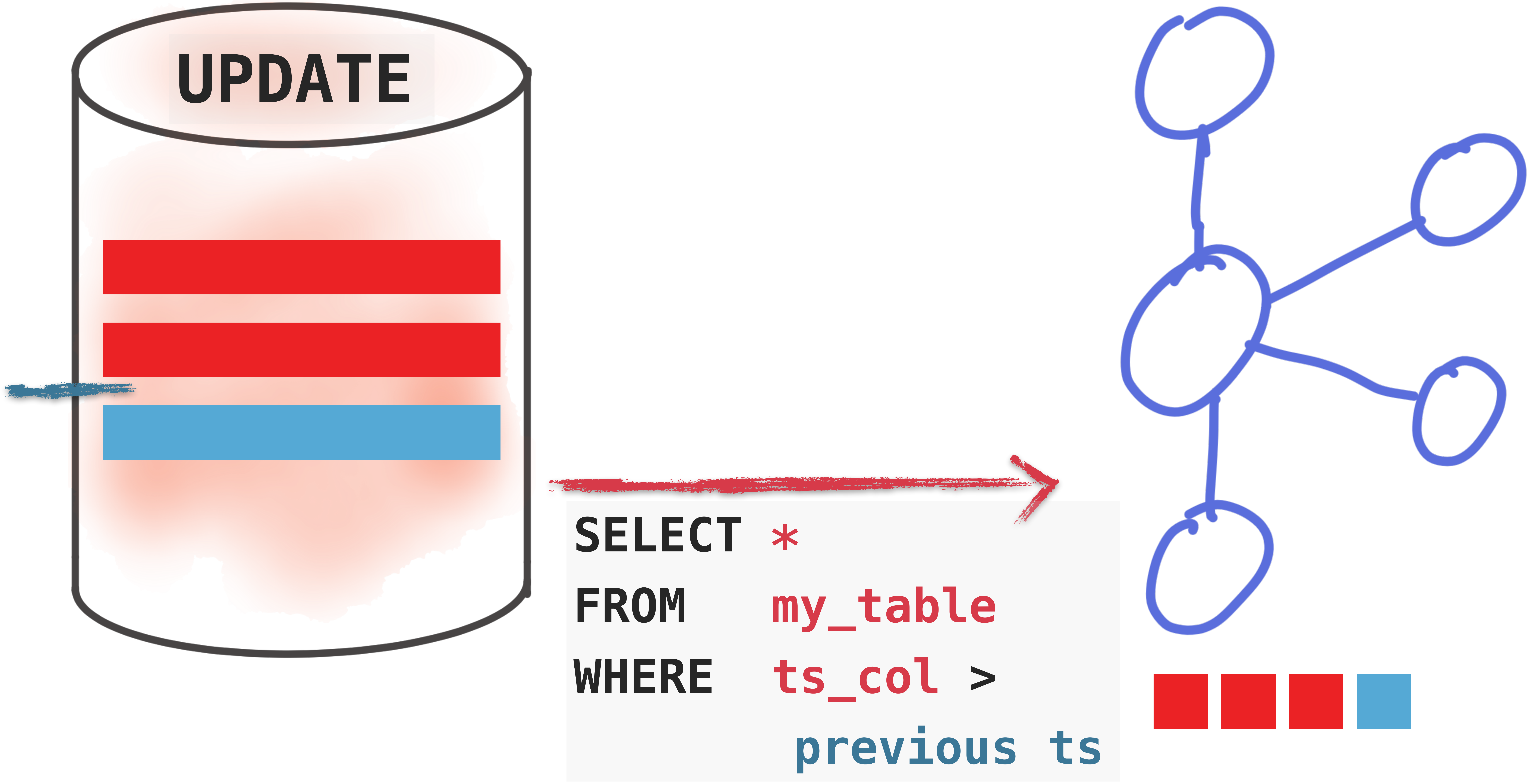


Query-based CDC

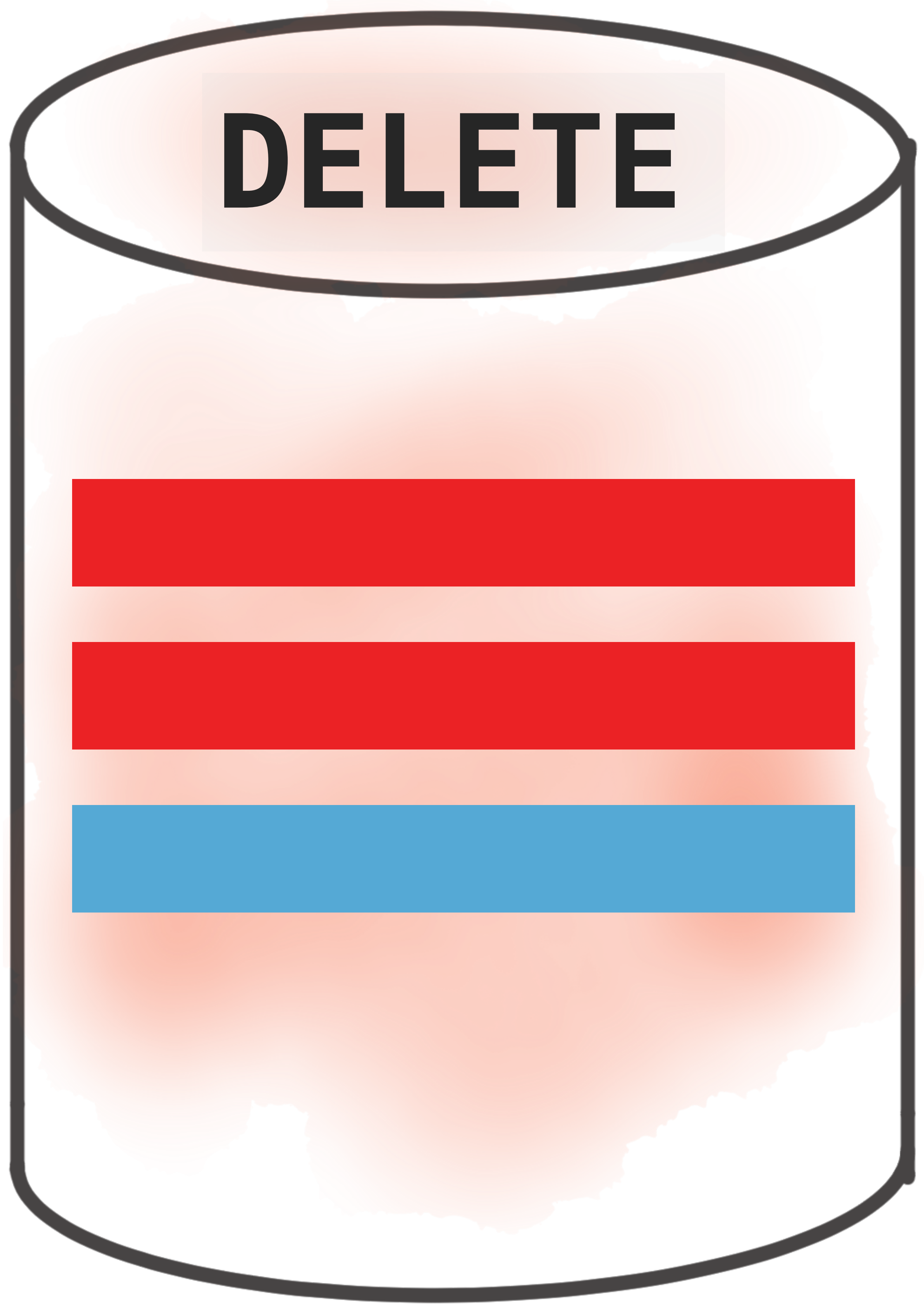


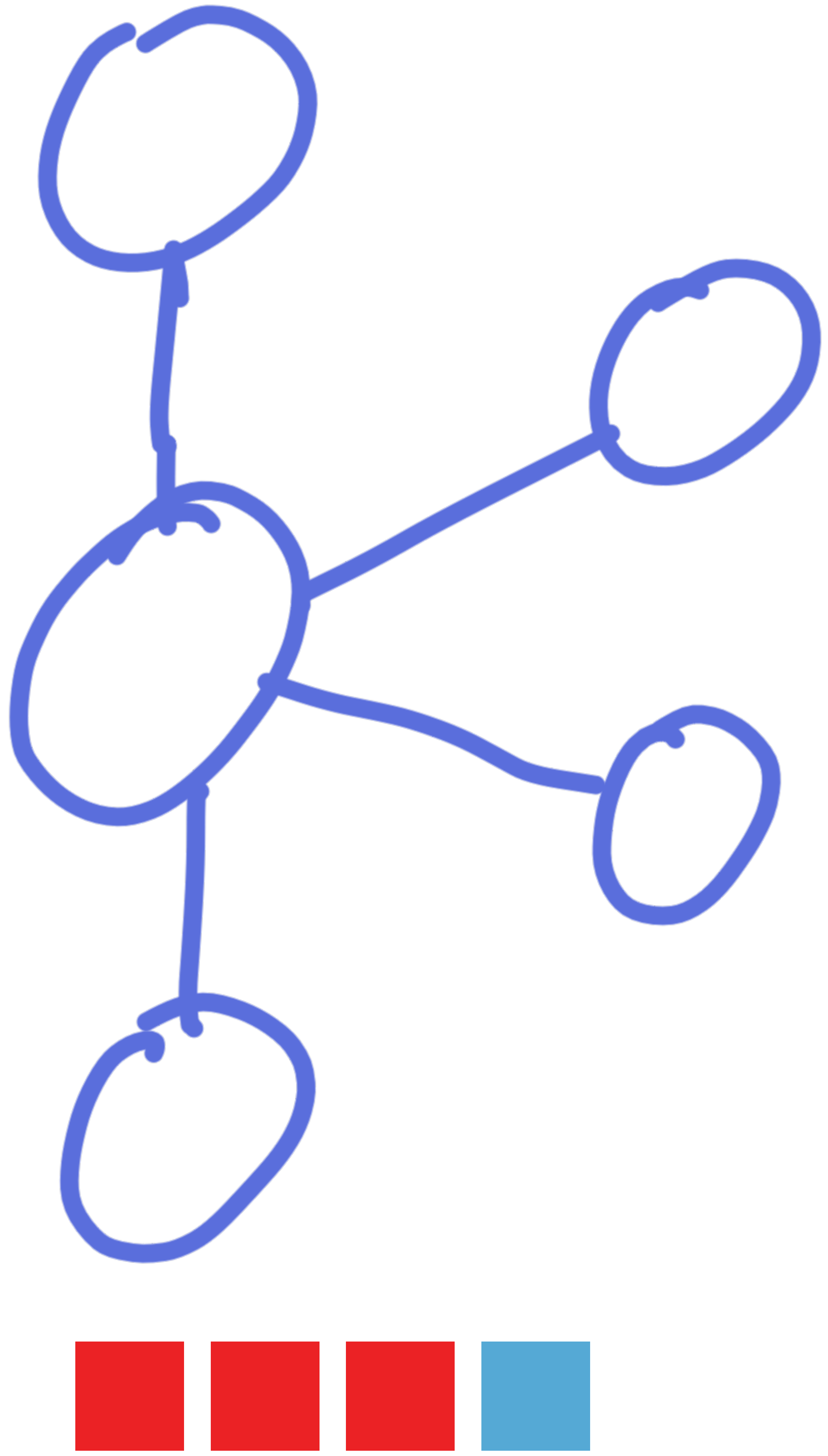


Query-based CDC



Query-based CDC





Query-based CDC


```
SELECT *
FROM   my_table
WHERE  ts_col >
        previous ts
```

Query-based CDC

orderID	status	address	updateTS

```
SELECT *
FROM   orders
WHERE  updateTS >
        previous ts
```

Query-based CDC

orderID	status	address	updateTS
42	SHIPPED	29 Acacia Road	10:54:29

```sql
SELECT *
FROM   orders
WHERE  updateTS >
        previous ts
```

```json
{
  "orderID": 42,
  "status": "SHIPPED",
  "address": "29 Acacia Road",
  "updateTS": "10:54:29"
}
```

Query-based CDC

orderID	status	address	updateTS
42	PENDING	--	10:54:01

```
INSERT INTO orders (orderID, status)
VALUES (42, 'PENDING');
```

Query-based CDC

orderID	status	address	updateTS
42	PENDING	1640 Riverside Drive	10:54:20

```sql
UPDATE orders
SET address = '1640 Riverside Drive'
WHERE orderID = 42;
```

Query-based CDC

orderID	status	address	updateTS
42	PENDING	29 Acacia Road	10:54:25

```sql
UPDATE orders
SET address = '29 Acacia Road'
WHERE orderID = 42;
```

Query-based CDC

orderID	status	address	updateTS
42	SHIPPED	29 Acacia Road	10:54:29

```sql
UPDATE orders
SET status = 'SHIPPED'
WHERE orderID = 42;
```

Query-based CDC

orderID	status	address	updateTS
42	PENDING	--	10:54:01
42	PENDING	1640 Riverside Drive	10:54:20
42	PENDING	29 Acacia Road	10:54:25
42	SHIPPED	29 Acacia Road	10:54:29

Query-based CDC

orderID	status	address	updateTS
42	SHIPPED	29 Acacia Road	10:54:29

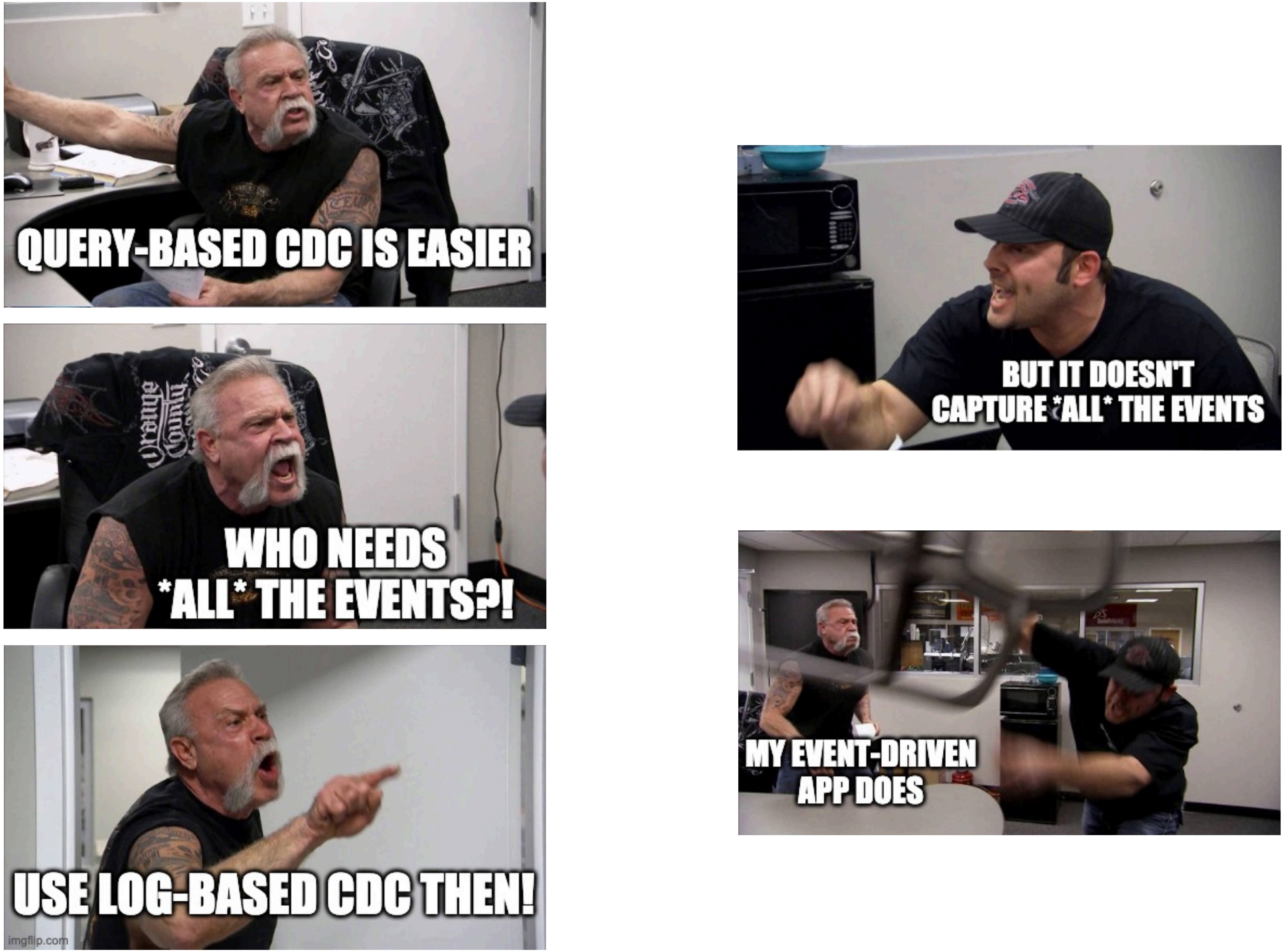
```
SELECT *
FROM   orders
WHERE  updateTS >
        previous ts
```

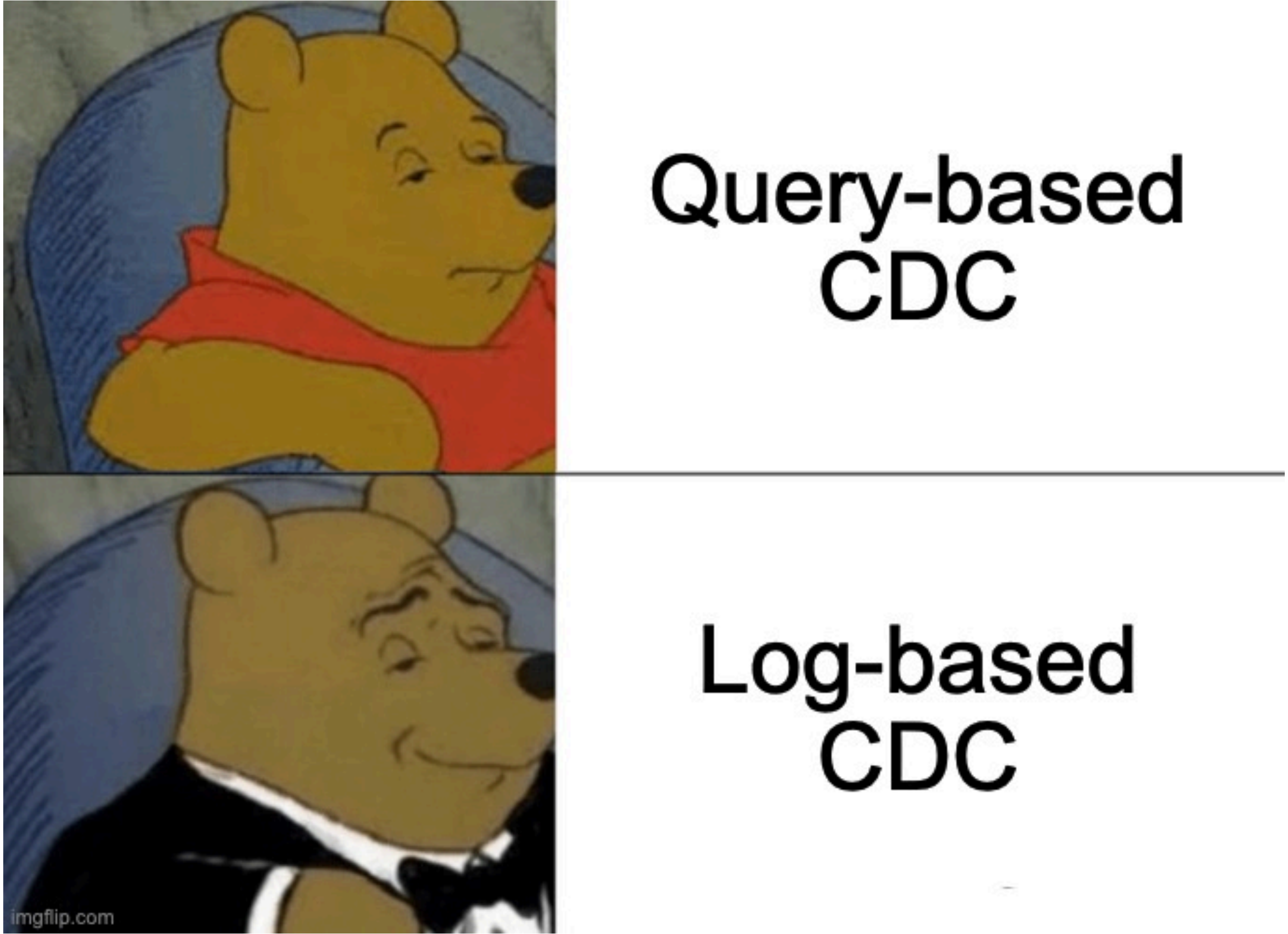

```
{
  "orderID": 42,
  "status": "SHIPPED",
  "address": "29 Acacia Road",
  "updateTS": "10:54:29"
}
```

Query-based CDC

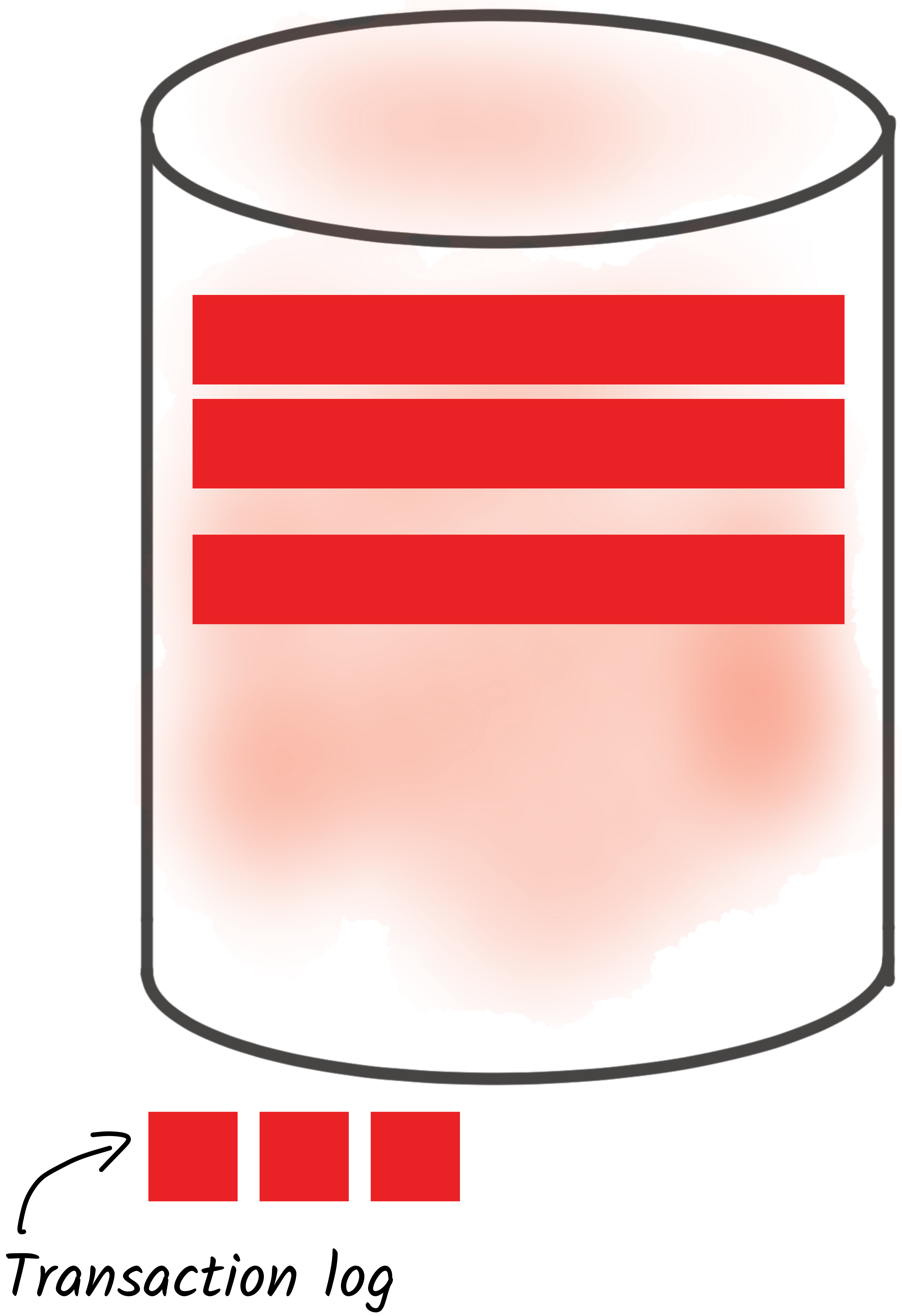
- ✅ Usually easier to setup
- ✅ Requires fewer permissions
- 🛑 Needs specific columns in source schema to track changes
- 🛑 Impact of polling the DB (or higher latencies tradeoff)
- 🛑 Can't track deletes
- 🛑 Can't track multiple events between polling interval

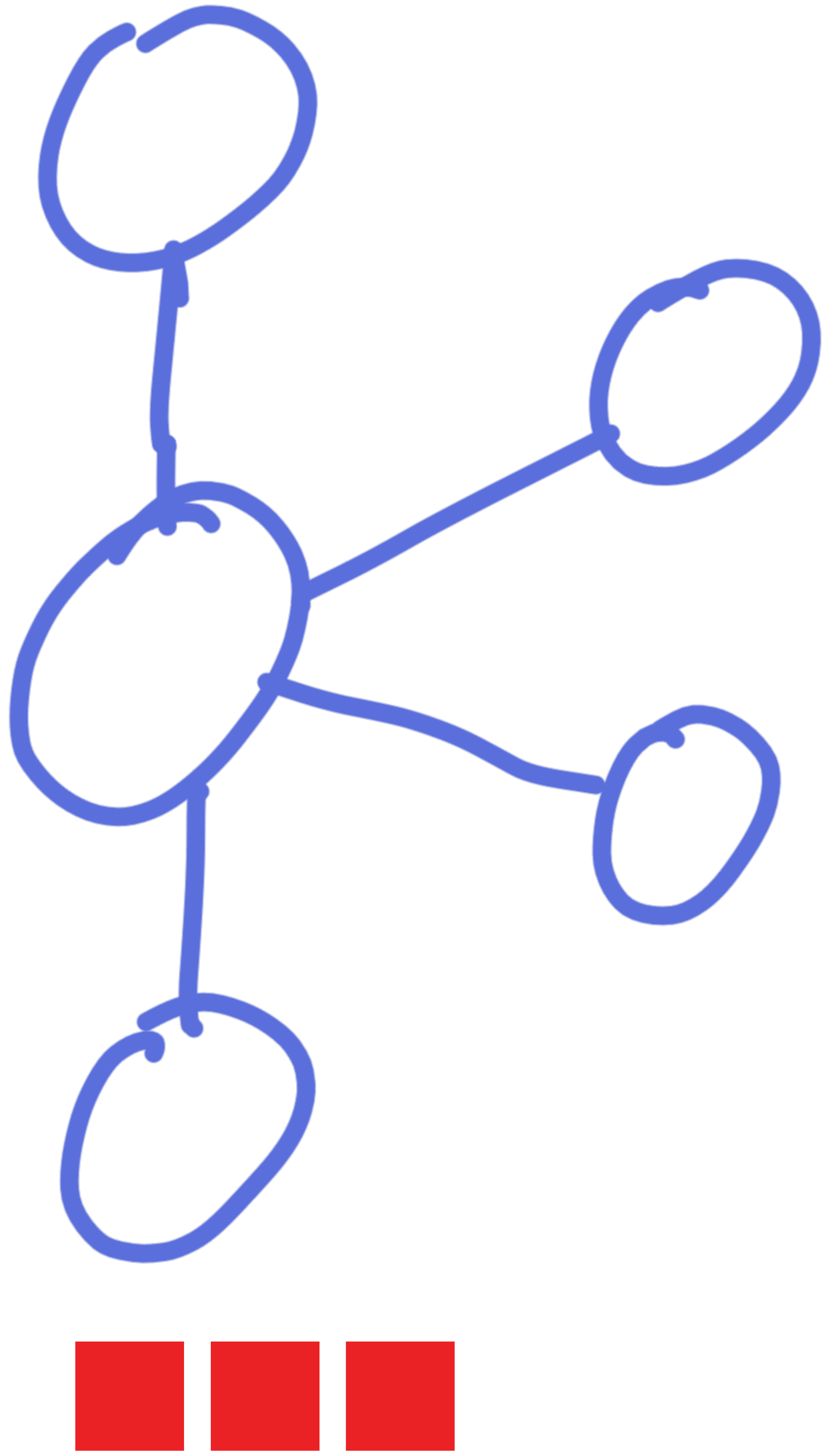
Photo by Matese Fields on Unsplash


QUERY-BASED CDC IS EASIER
BUT IT DOESN'T CAPTURE *ALL* THE EVENTS
WHO NEEDS *ALL* THE EVENTS?!
MY EVENT-DRIVEN APP DOES
USE LOG-BASED CDC THEN!
imgflip.com

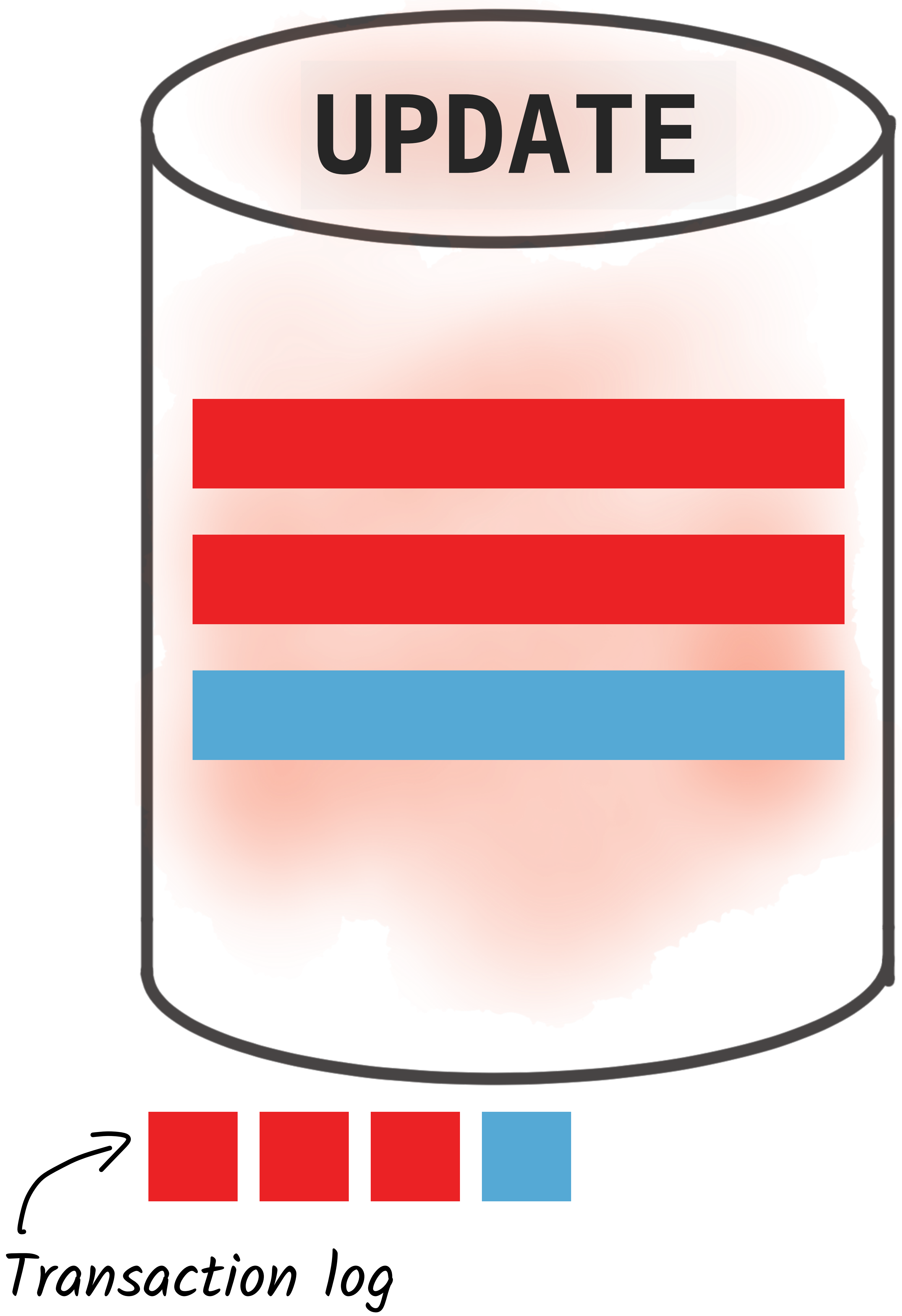

Query-based
CDC
Log-based
CDC
imgflip.com

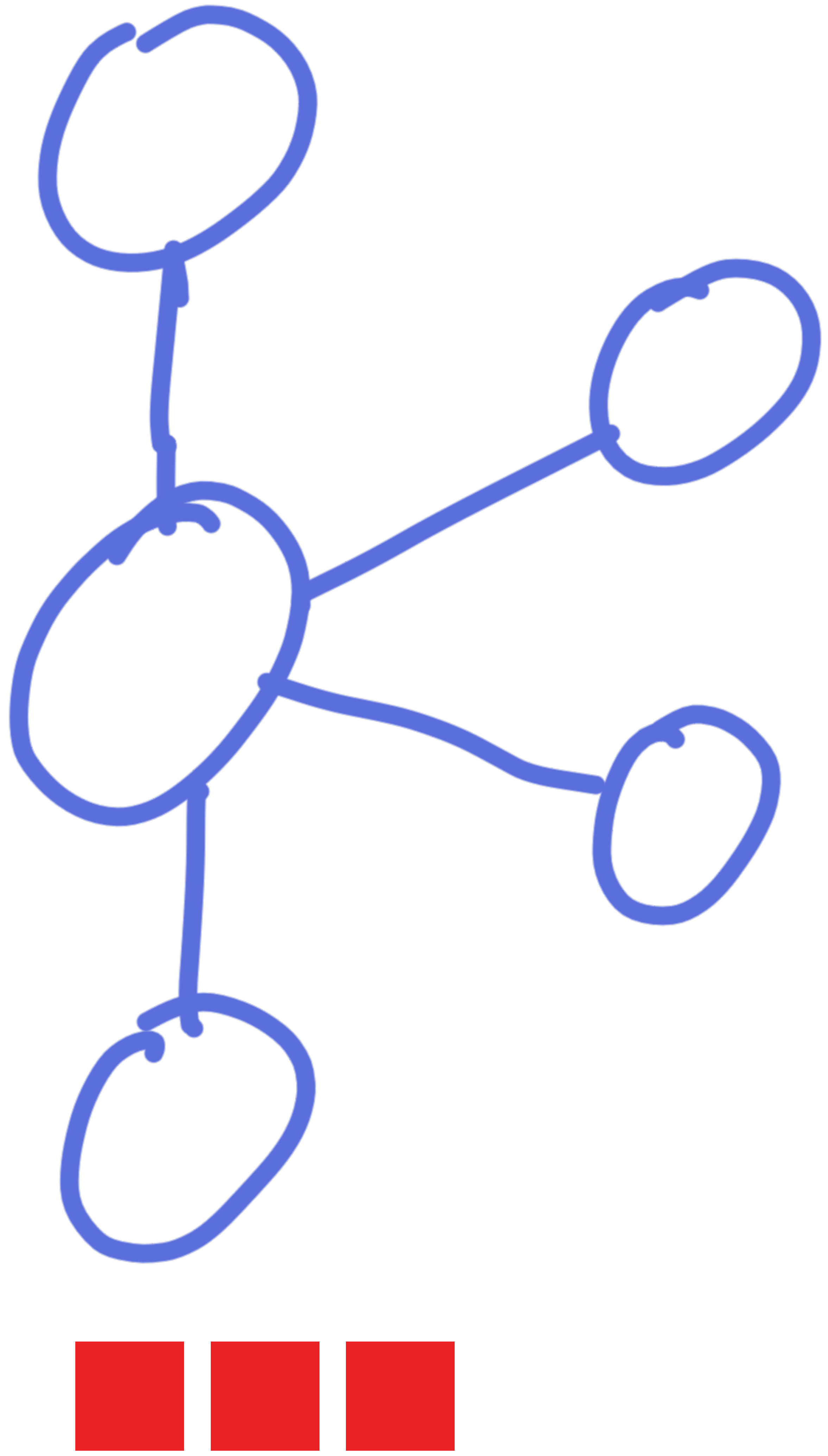
Log-based CDC



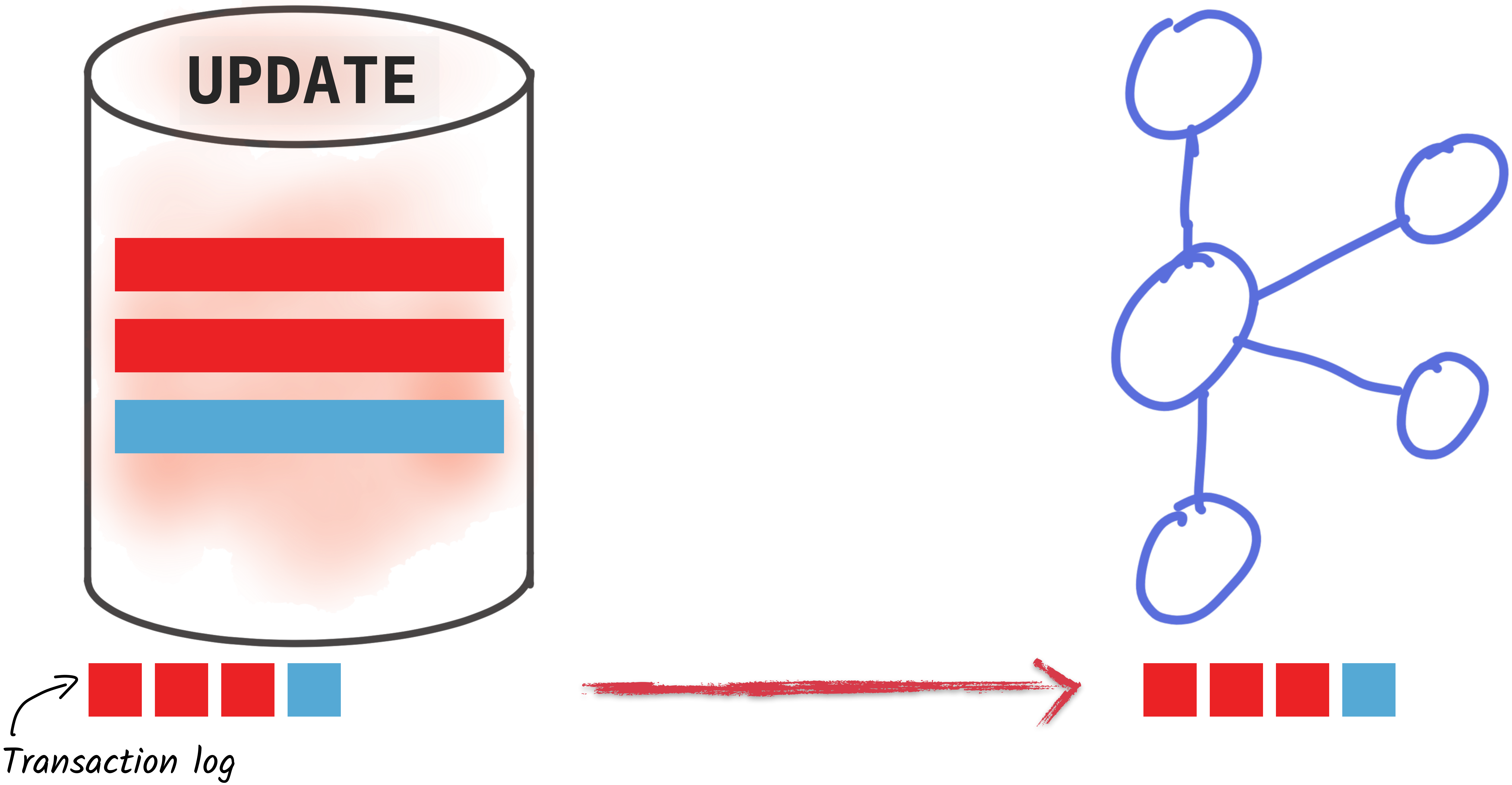


Log-based CDC

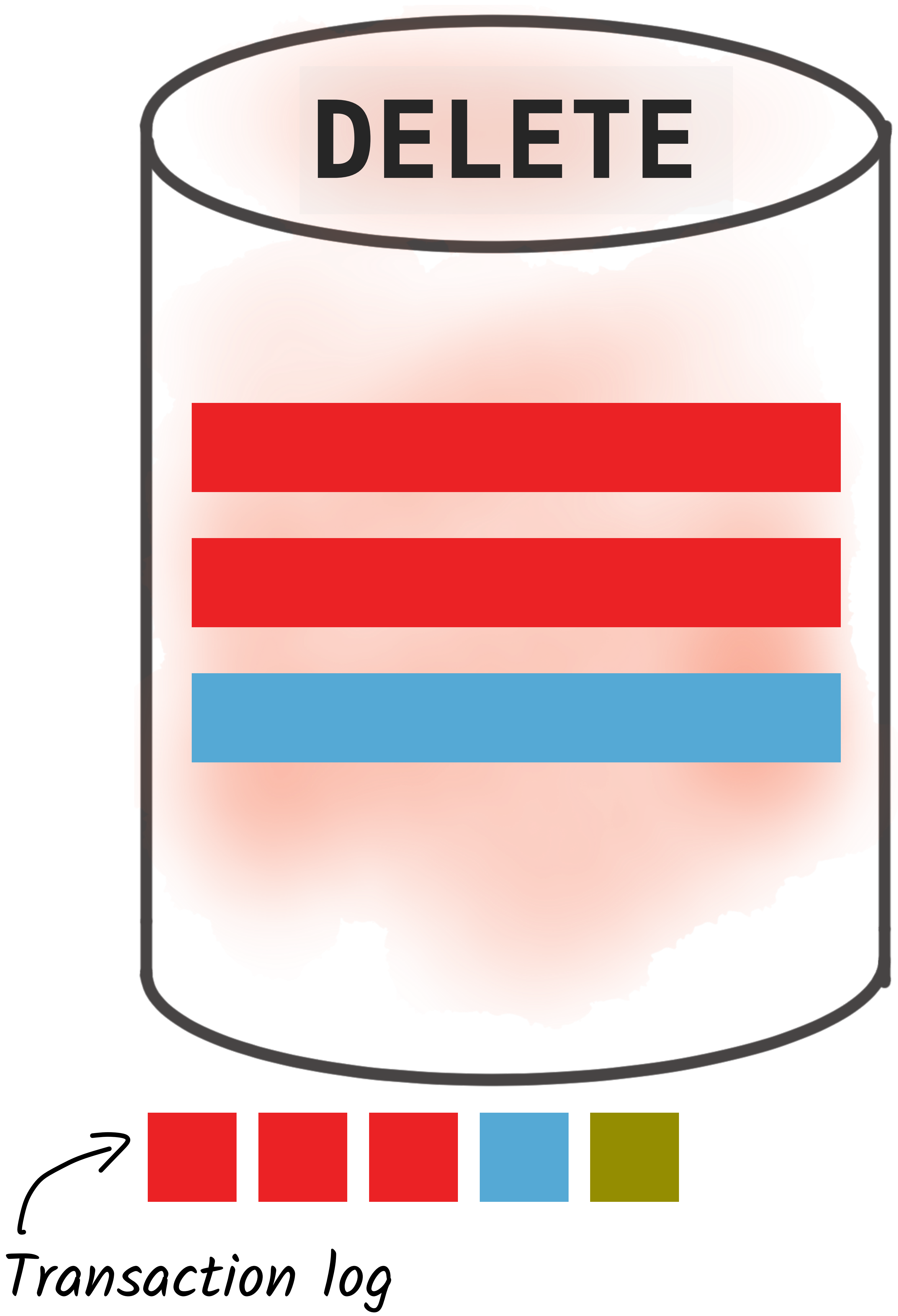


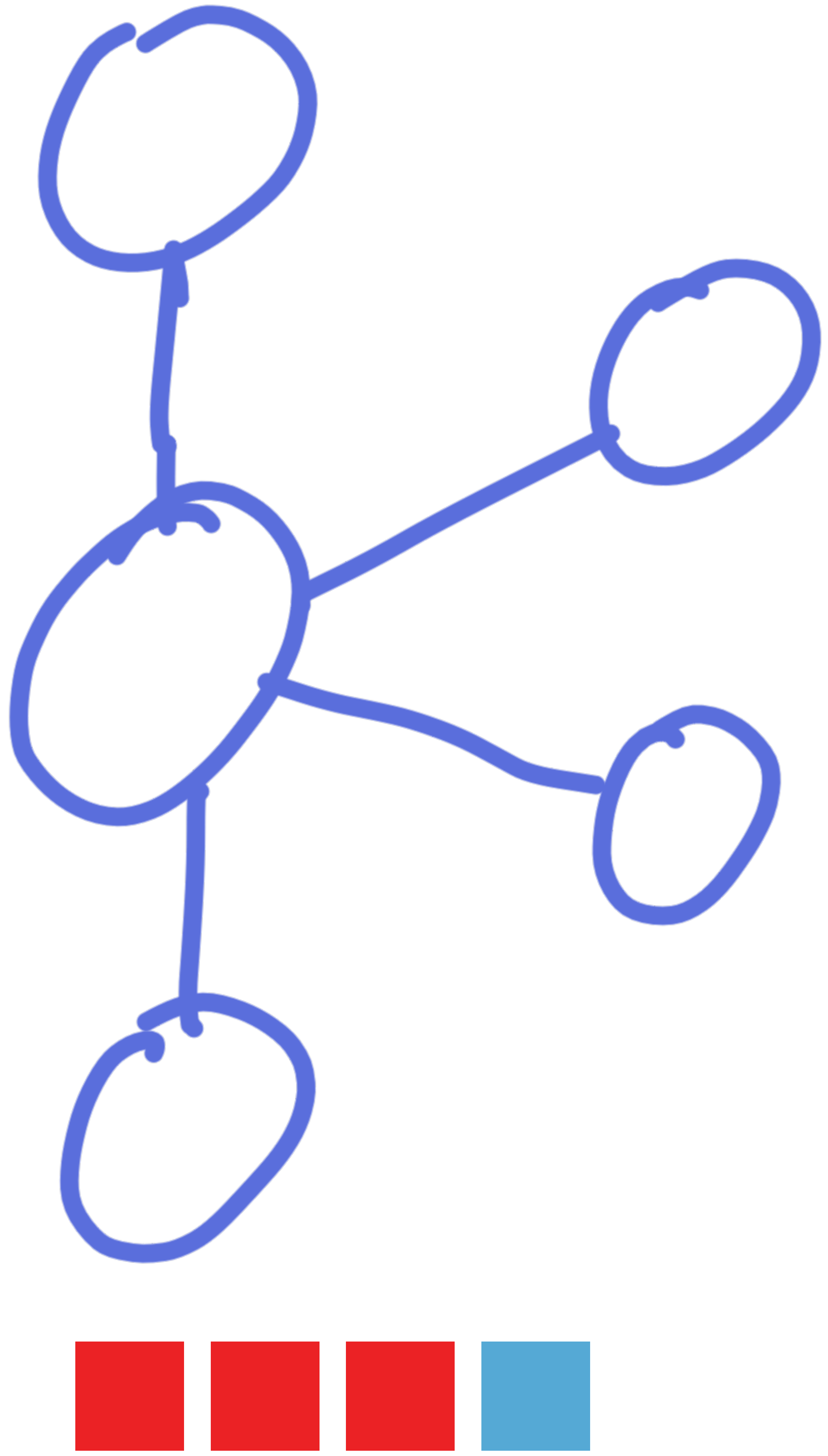


Log-based CDC

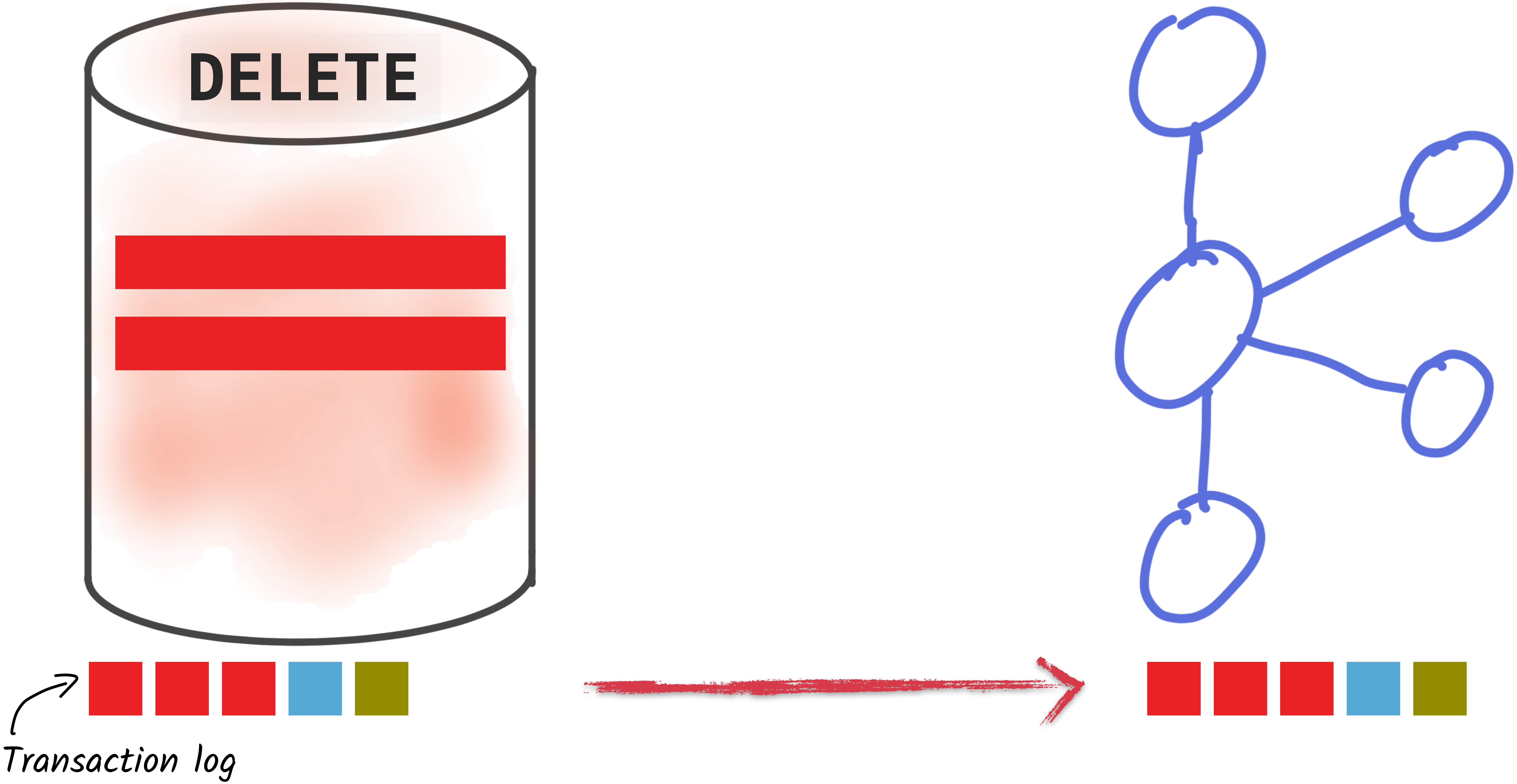


Log-based CDC

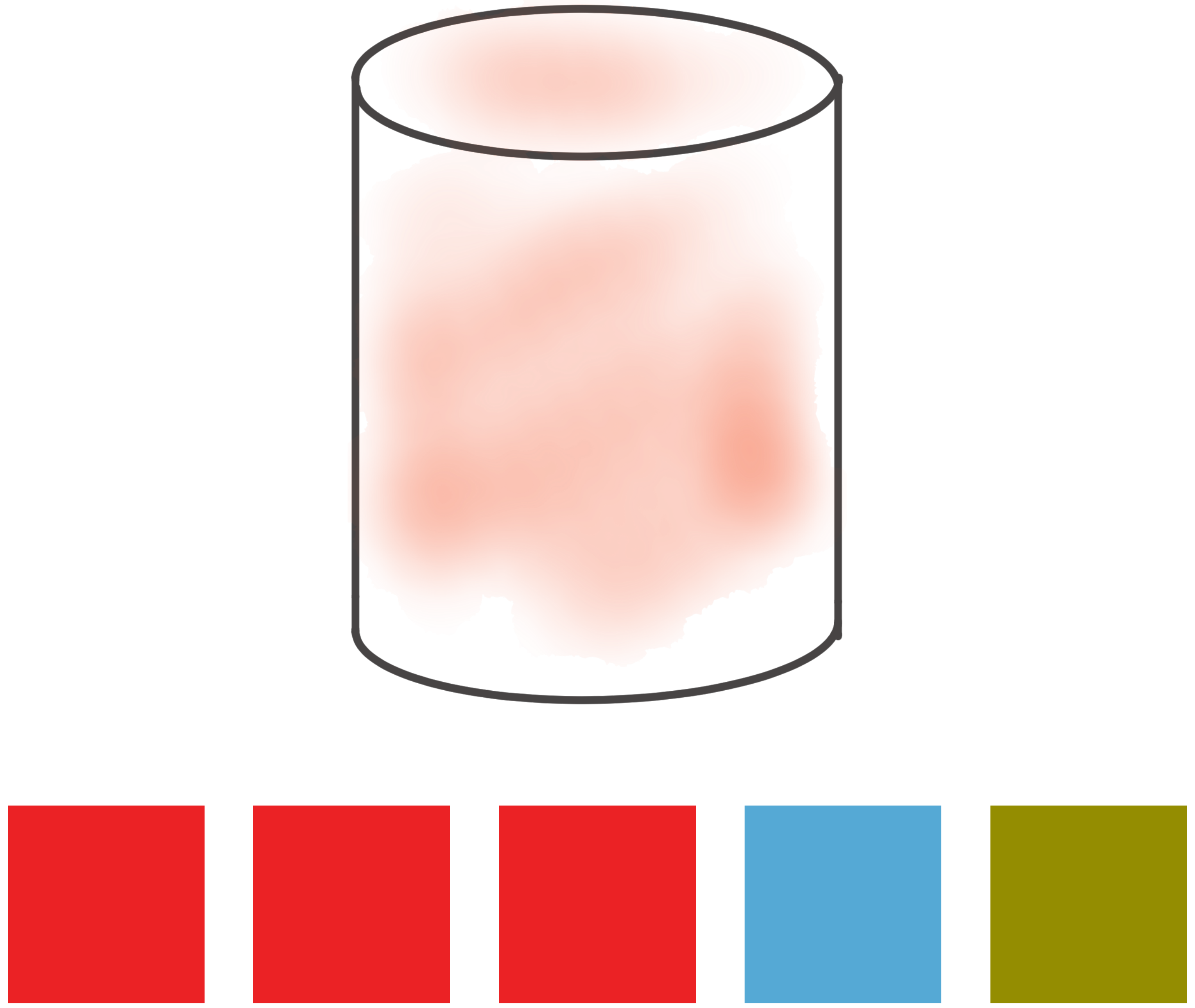


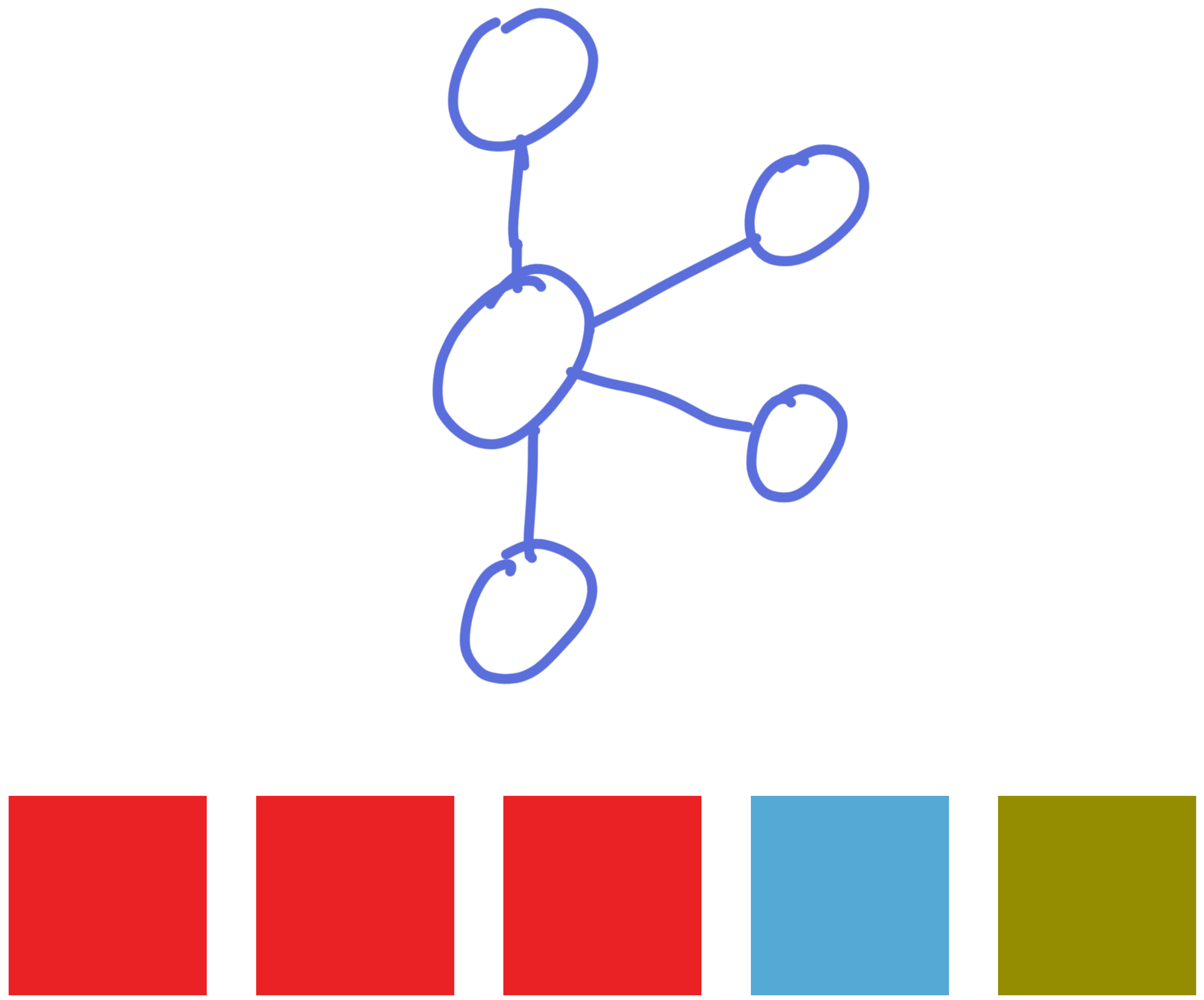


Log-based CDC



Log-based CDC





Commit log

Photo by Sebastian Pociecha on Unsplash

Log-based CDC

✅ Greater data fidelity

✅ Lower latency

✅ Lower impact on source

🛑 More setup steps

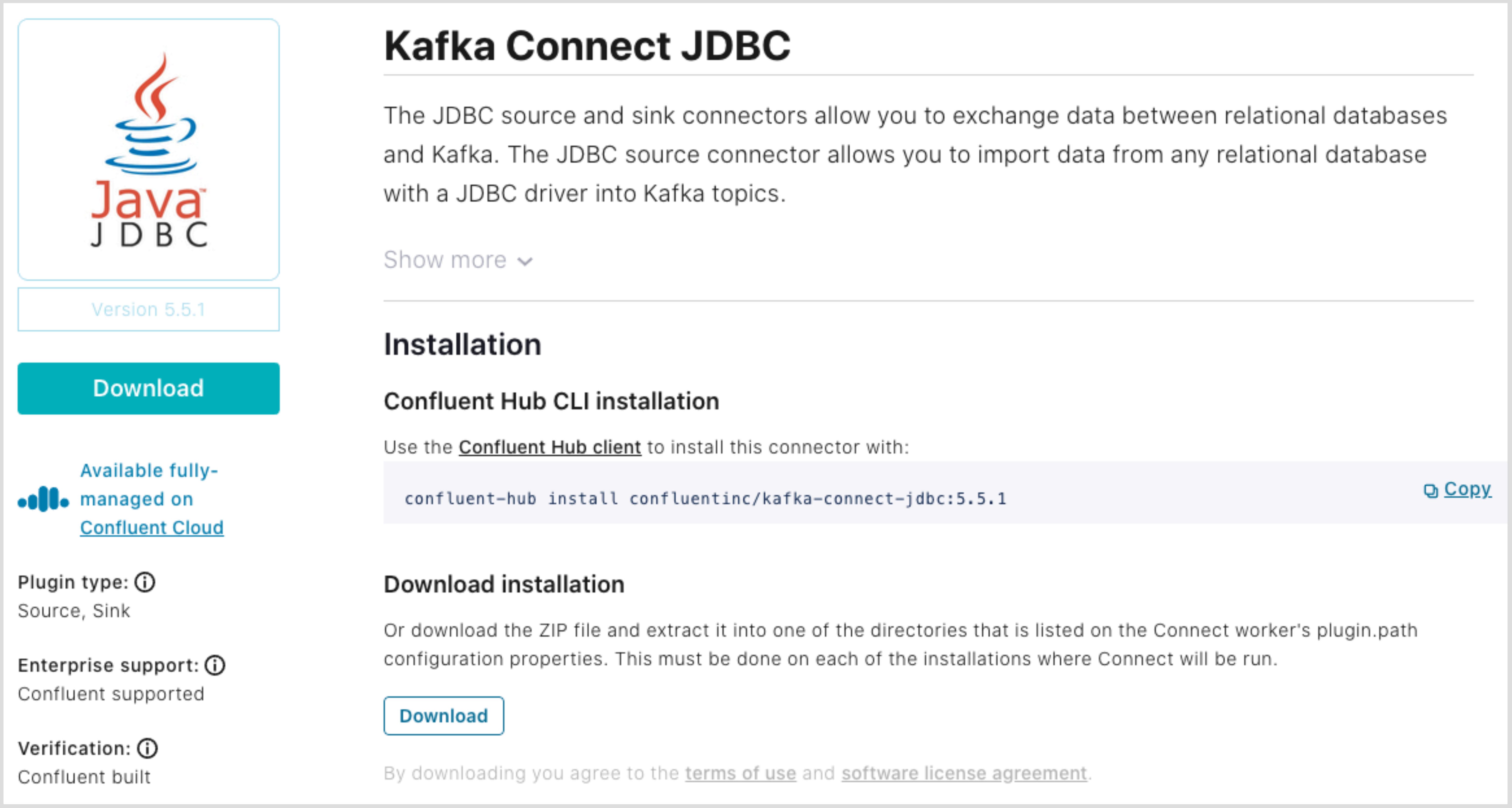
🛑 Higher system privileges required

🛑 For propriatory databases, usually $$$

Read more: http://cnfl.io/kafka-cdc

tl;dr : which tool do I use?

- If you're wanting to do query-based CDC, the choice is simple:



`confluent.io/hub`

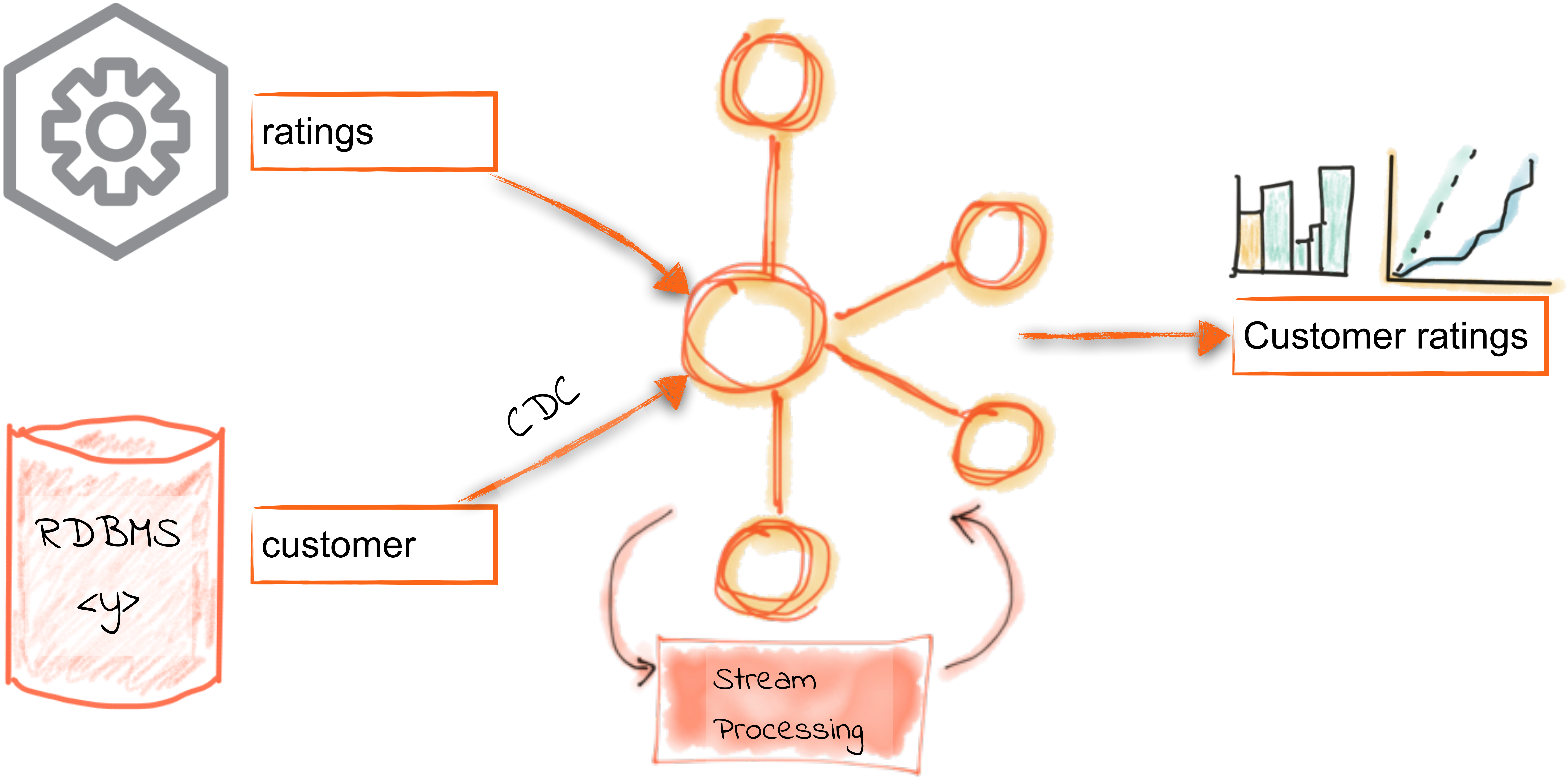
Which Log-Based CDC Tool?

- Open Source RDBMS,
  e.g. **MySQL**, **PostgreSQL**
  - Debezium
  - (+ paid options)

- Mainframe
  e.g. **VSAM**, **IMS**
  - Attunity
  - SQData
  - tcVISION

- Proprietory RDBMS,
  e.g. **Oracle**, **MS SQL, DB2**
  - Oracle GoldenGate
    - Debezium + XStream
  - Qlik (Attunity)
  - IBM InfoSphere Data Replication
  - SQData
  - HVR
  - tcVISION
  - Etc

Real-time Event Stream Enrichment



Demo Time!

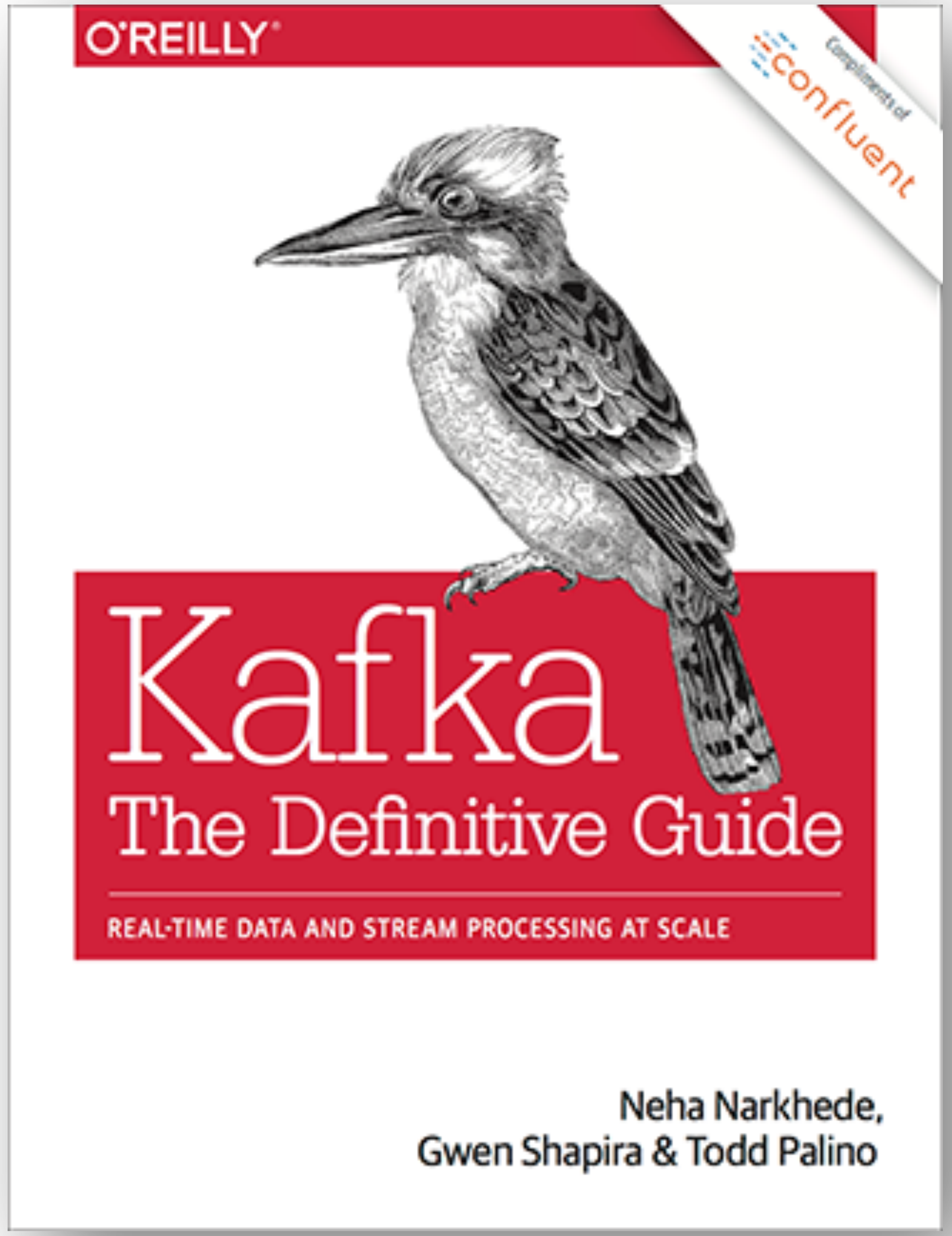
https://rmoff.dev/cdc-demo

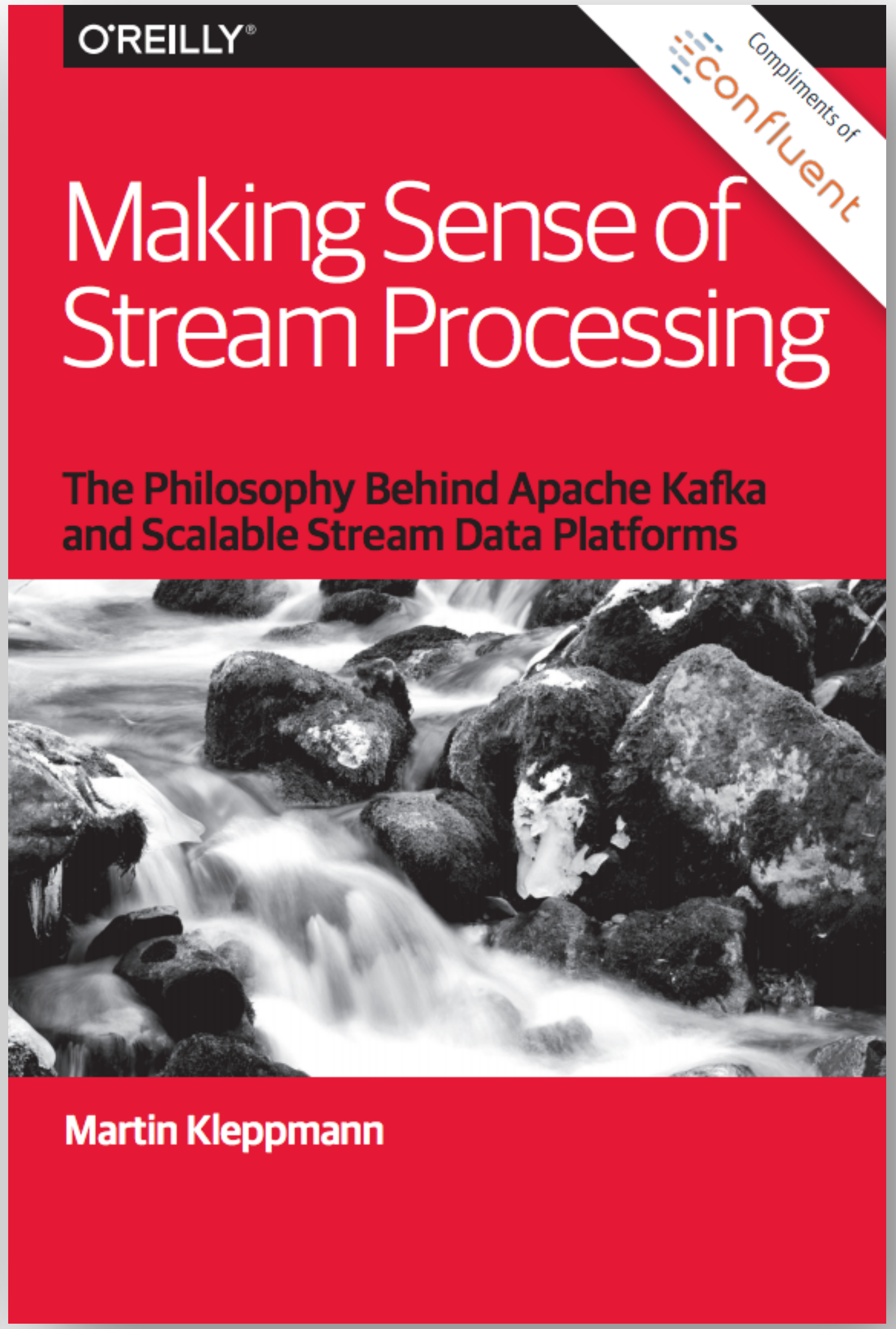
Want to learn more?

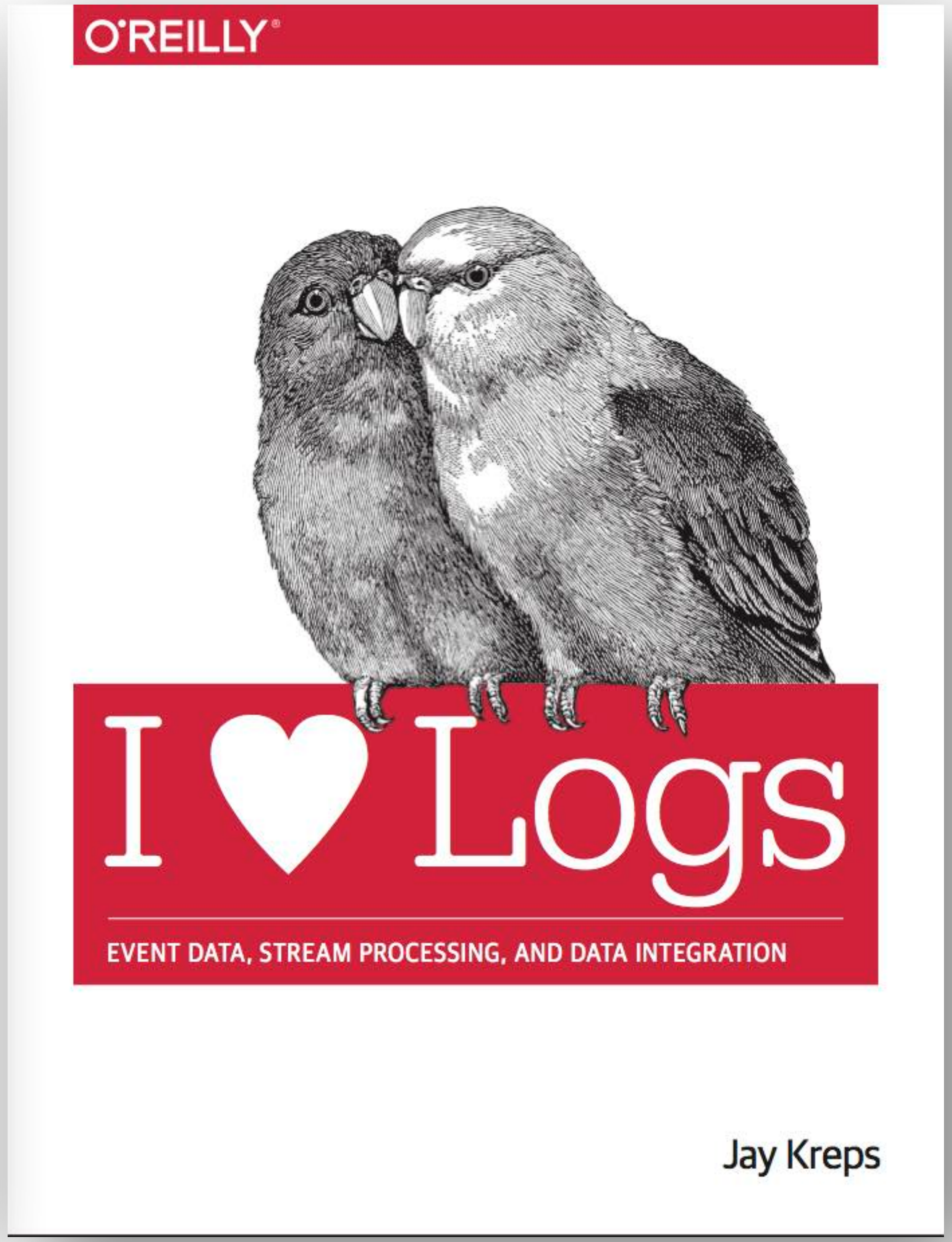
CTAs, not CATs
(sorry, not sorry)

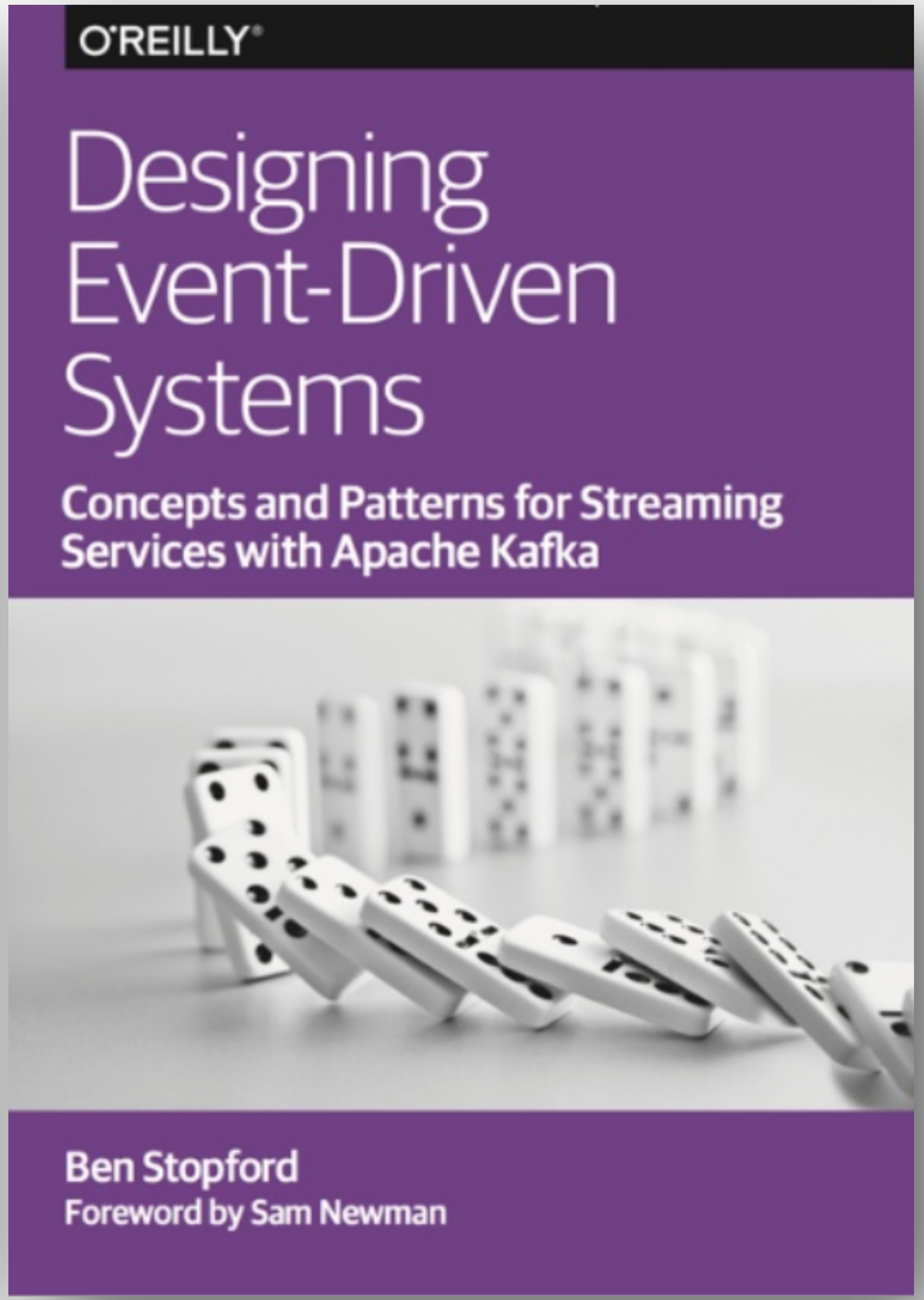
Free Books!

https://rmoff.dev/acah20









60DEVADV

Free money!

(additional $60 towards your bill 😄)

$200 USD off your bill each calendar month for the first three months when you sign up

`https://rmoff.dev/ccloud`

confluent cloud

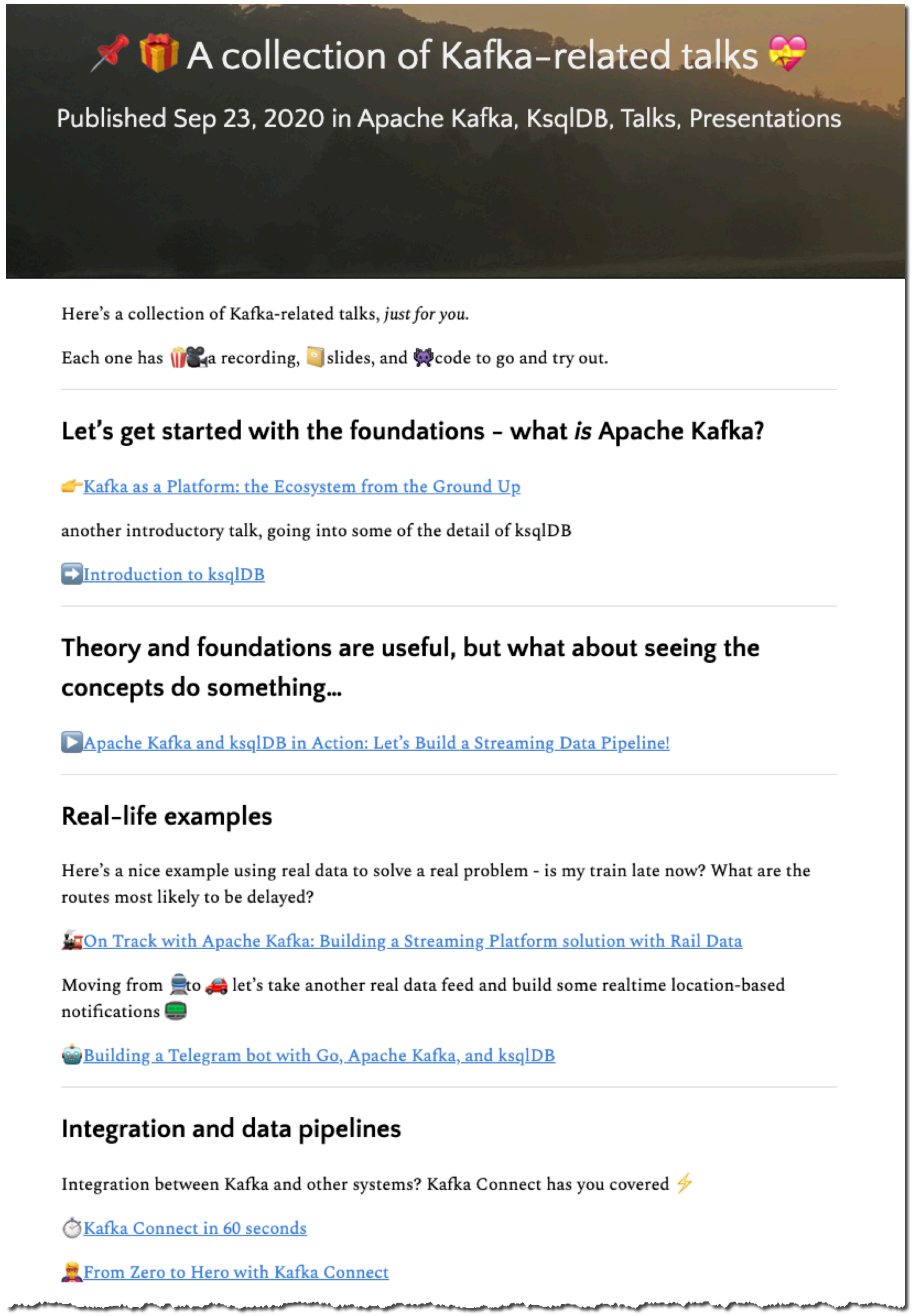
Fully Managed Apache Kafka as a Service


Learn Kafka.
Start building with Apache Kafka at Confluent Developer.
developer.confluent.io

Confluent Community Slack group

cnfl.io/slack

Further reading / watching


A collection of Kafka-related talks

Published Sep 23, 2020 in Apache Kafka, KsqlDB, Talks, Presentations

Here's a collection of Kafka-related talks, *just for you.*

Each one has a recording, slides, and code to go and try out.

Let's get started with the foundations – what *is* Apache Kafka?

Kafka as a Platform: the Ecosystem from the Ground Up

another introductory talk, going into some of the detail of ksqlDB

Introduction to ksqlDB

Theory and foundations are useful, but what about seeing the concepts do something…

Apache Kafka and ksqlDB in Action: Let's Build a Streaming Data Pipeline!

Real-life examples

Here's a nice example using real data to solve a real problem - is my train late now? What are the routes most likely to be delayed?

On Track with Apache Kafka: Building a Streaming Platform solution with Rail Data

Moving from to let's take another real data feed and build some realtime location-based notifications

Building a Telegram bot with Go, Apache Kafka, and ksqlDB

Integration and data pipelines

Integration between Kafka and other systems? Kafka Connect has you covered

Kafka Connect in 60 seconds

From Zero to Hero with Kafka Connect

https://rmoff.dev/kafka-talks

#EOF

@rmoff rmoff.dev/talks youtube.com/rmoff