



# *Building a Telegram bot with Apache Kafka and ksqlDB*



@rmoff

#NDCSydney

Robin Moffatt





NCP Welcome to Hall Ings Car Park

BRADFORD CAPITAL OF CYCLING



*Where's my nearest carpark  
with available spaces?*





*How many spaces are available  
in this car park?*







💡 *Tell me when a car park with spaces is available*

N.C.P. Welcome to Hall Ings Car Park P

BRADFORD CAPITAL OF CYCLING





*How does occupancy vary over time?*





# \$ whoami

- > Robin Moffatt (@rmoff)
- > Senior Developer Advocate at Confluent  
*(Apache Kafka, not Wikis 😊)*
- > Working in data & analytics since 2001
- > ♠️ Oracle ACE Director (Alumnus)



<http://rmoff.dev/talks> • <http://rmoff.dev/blog> • <http://rmoff.dev/youtube>





Datasets



Products



District Dashboard (visual)



Latest CQC Results (visual)



Other open data sites



Contact

## Bradford car parks

City of Bradford Metropolitan District Council



Resources from the City of Bradford Metropolitan District Council (CBMDC) Parking service.

### 1 - Car park locations

a simple csv containing name and location including latitude / longitude

### 2 - Car park current status.

API that returns a csv dataset of the current status of 8 Bradford city centre car parks. The dataset returns capacity, empty places, status together with location details.

The dataset is updated every 3 minutes for a live view of spaces in these car parks.

### 3 - Car park historic status

API that returns a csv dataset building up the historic status of the 8 city centre car parks.

The dataset is updated every 30 minutes.

## 4 Resources

[csv](#) [api](#) [geojson](#)

### More Information

 [Map of car parks across Bradford district](#)

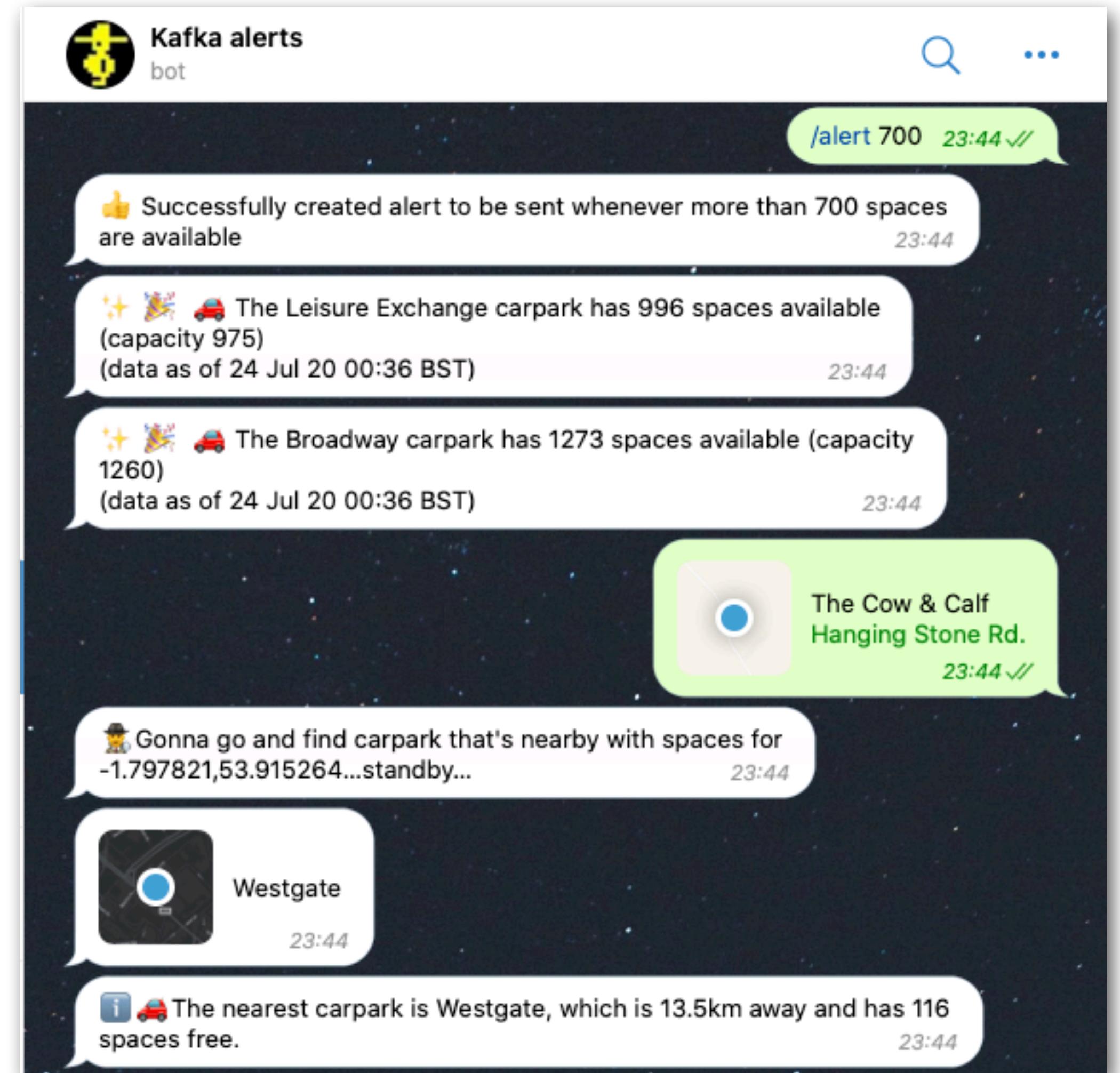
[html](#) [Live map of Bradford car parks](#)

**License** UK Open Government Licence (OGL v3)

**Frequency** daily



# Telegram





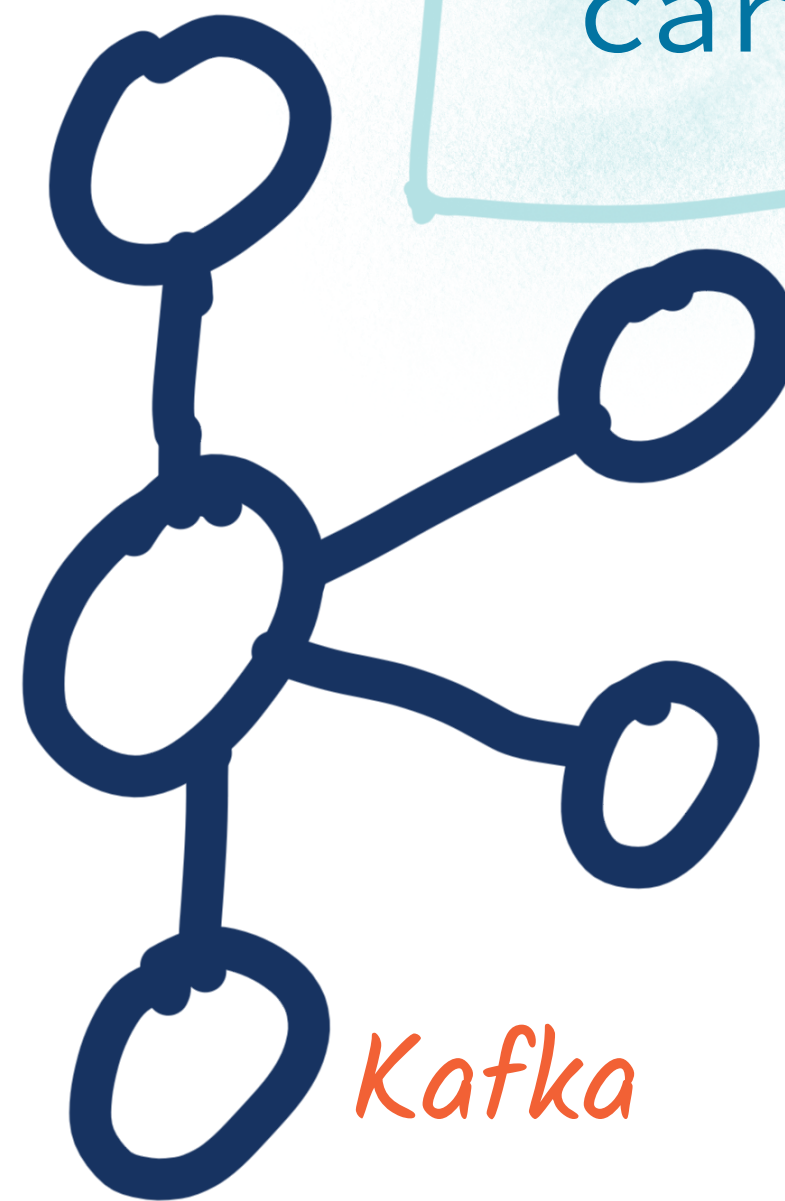
*Don't just tell me...*

*show me!*



Demo code: <https://rmoff.dev/carparks>



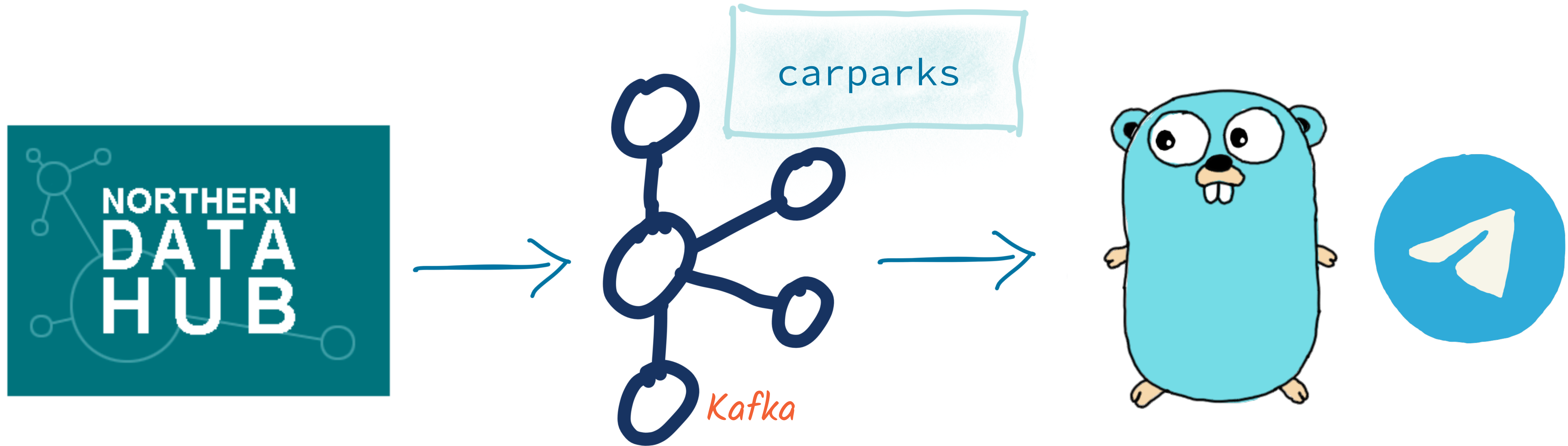




*What are the  
key pieces of  
the design?*

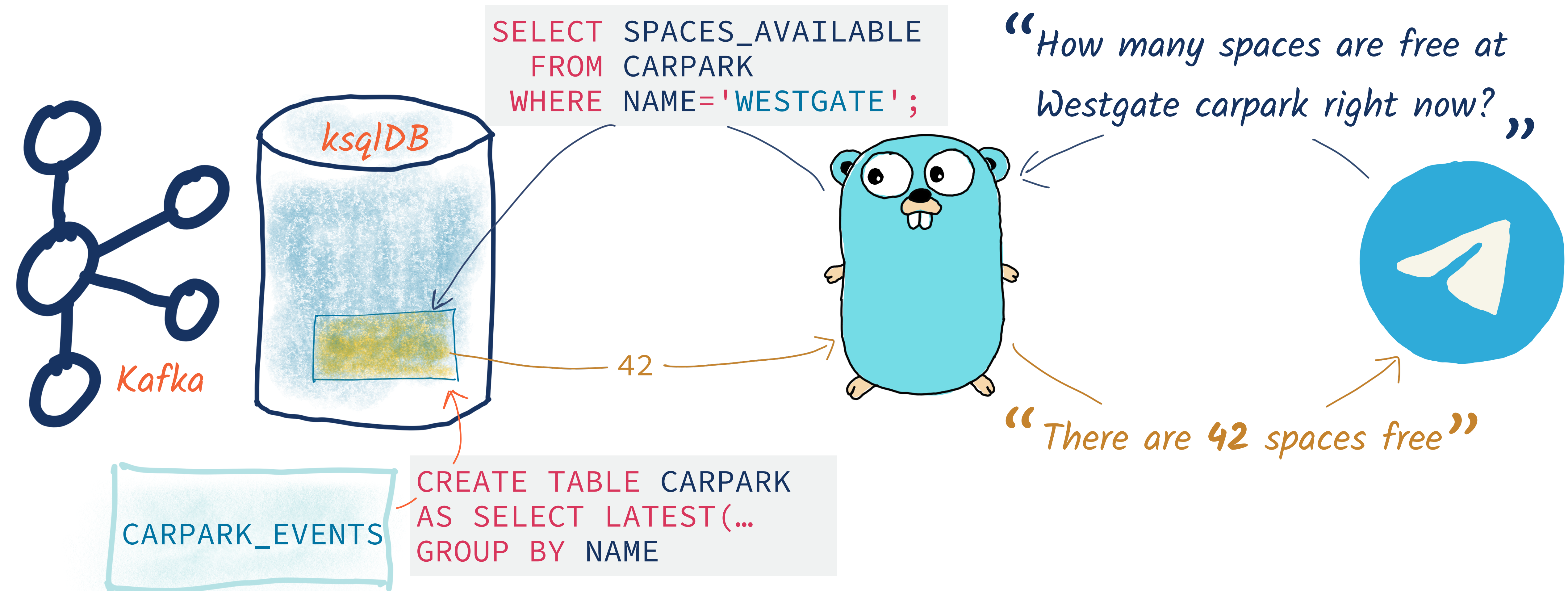


# Event Driven Alerts



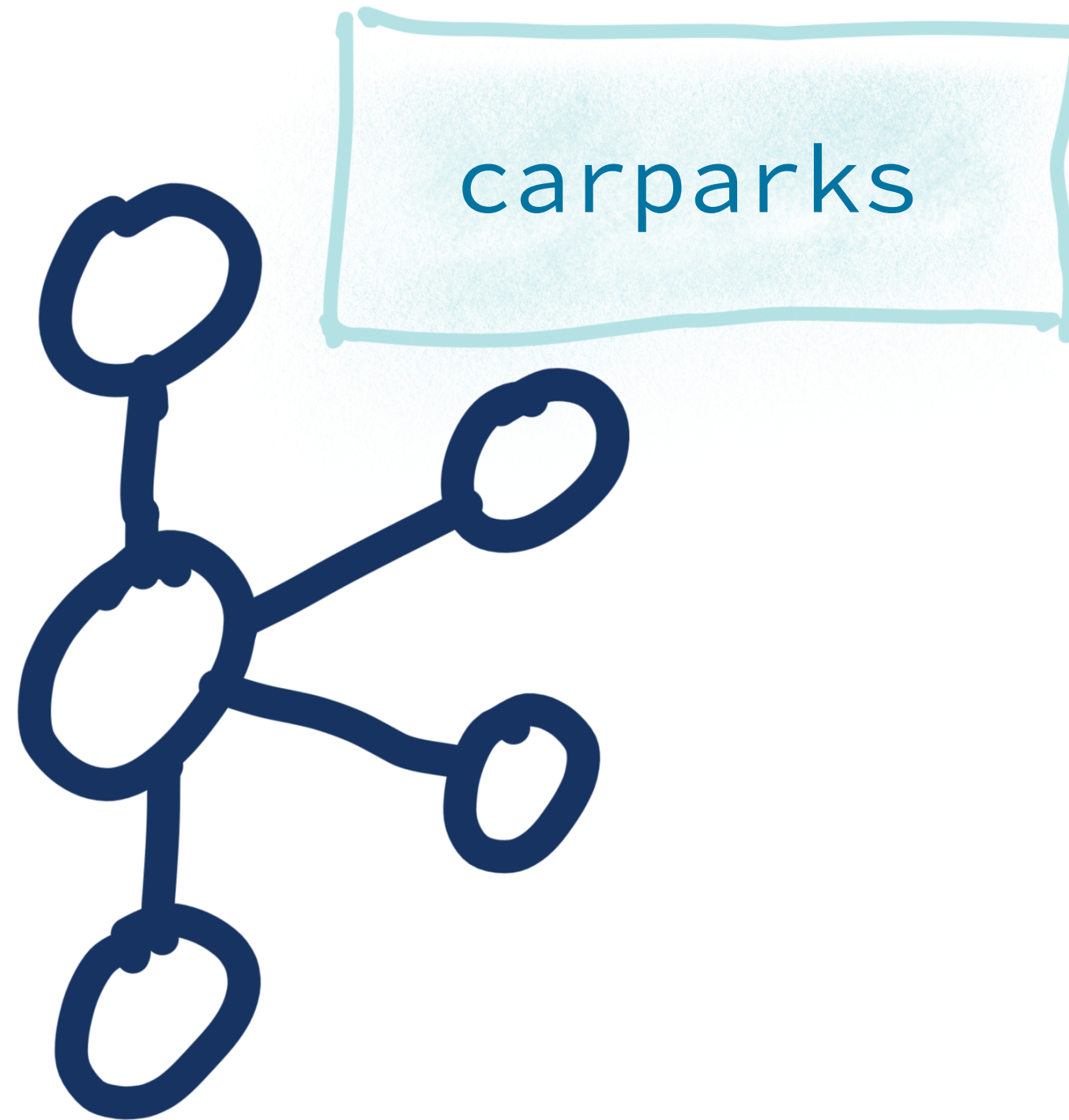


# KN Lookups (materialised views)



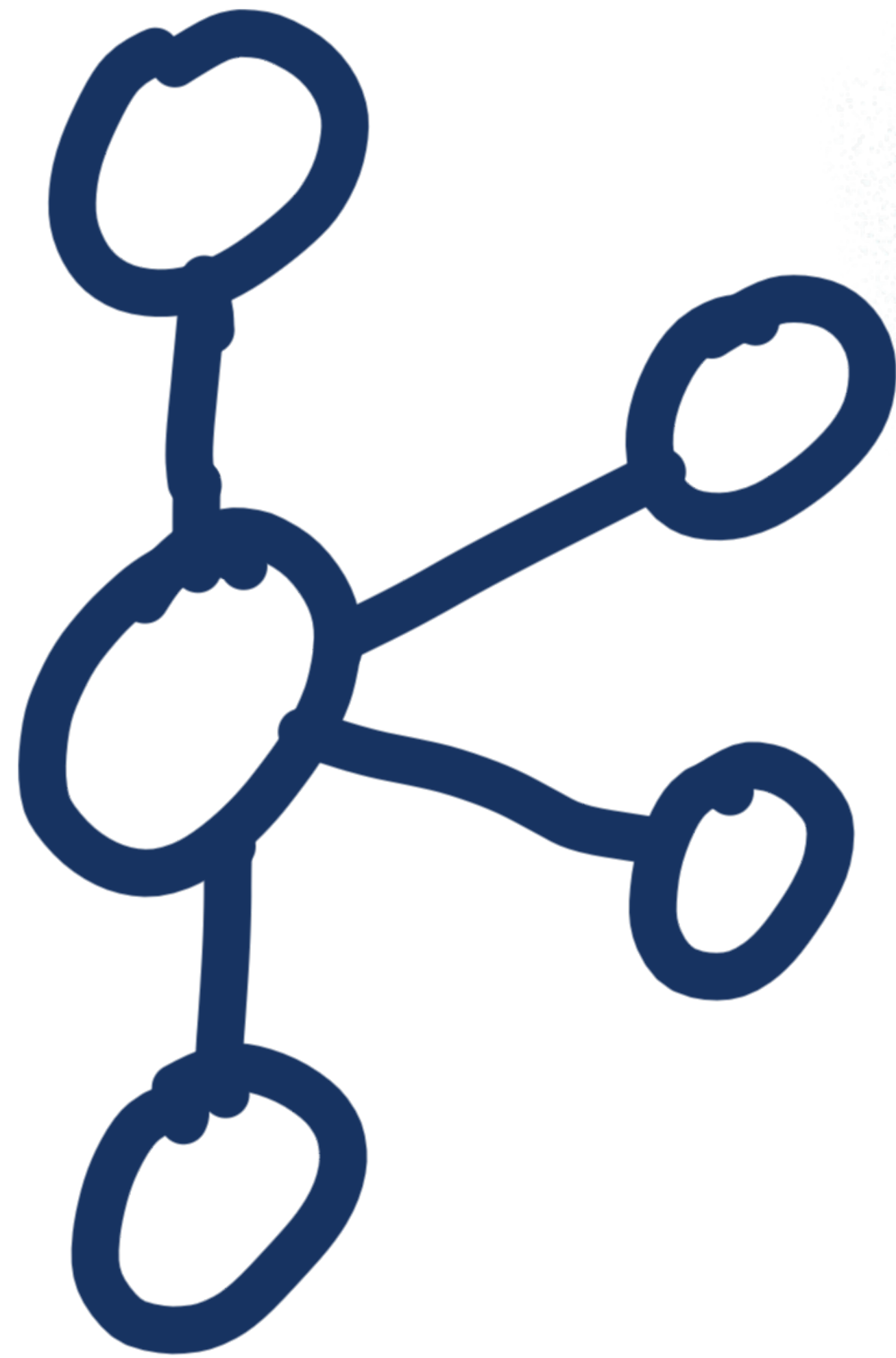


# A schema...

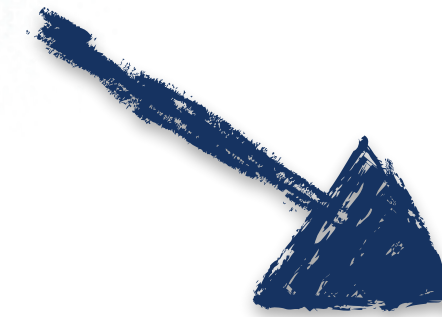




# A schema...

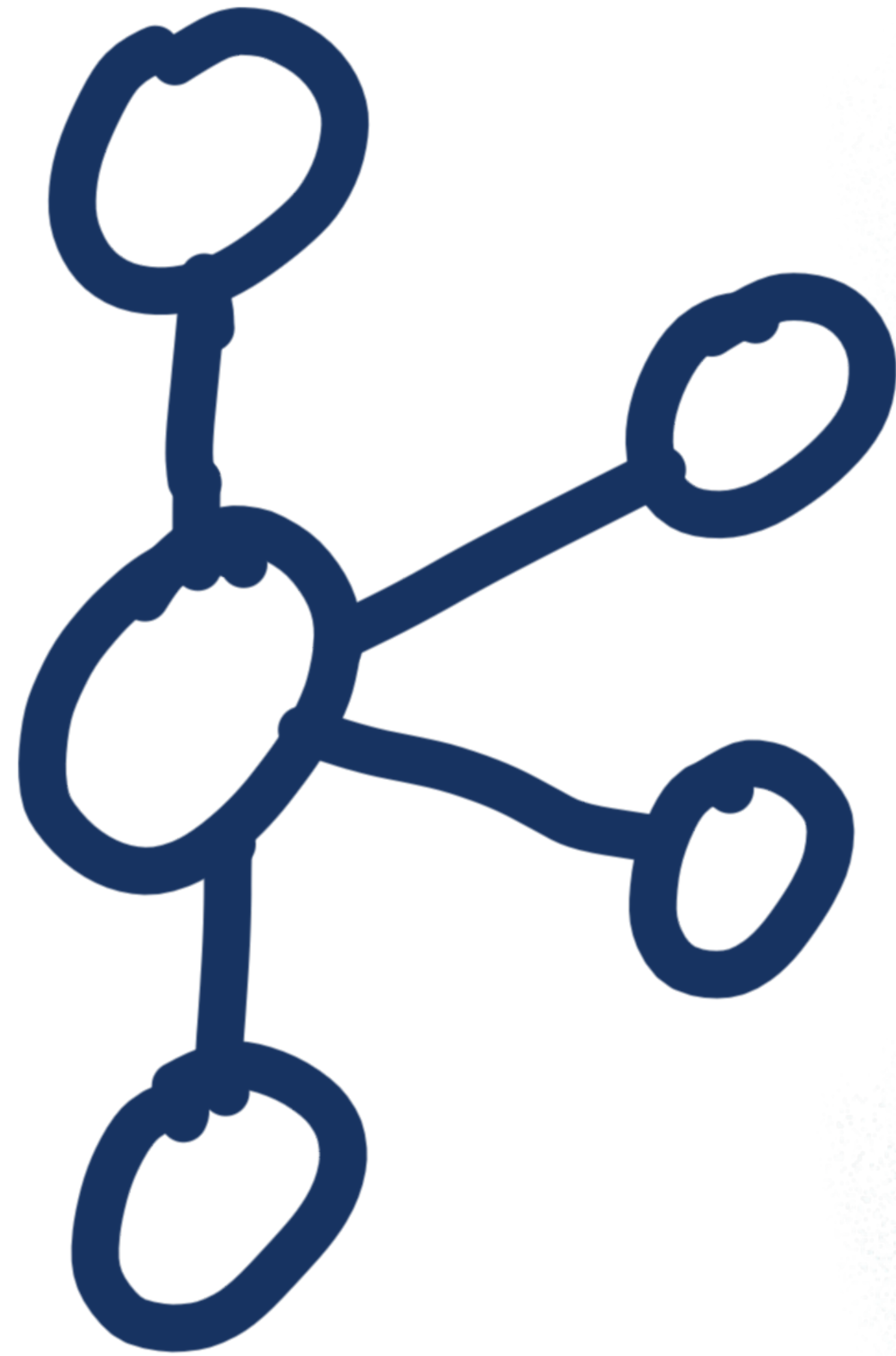


2020-10-14,12:28,Broadway,1132,921  
2020-10-14,12:28,Kirkgate Centre,611,474  
2020-10-14,12:28,Sharpe Street,98,63



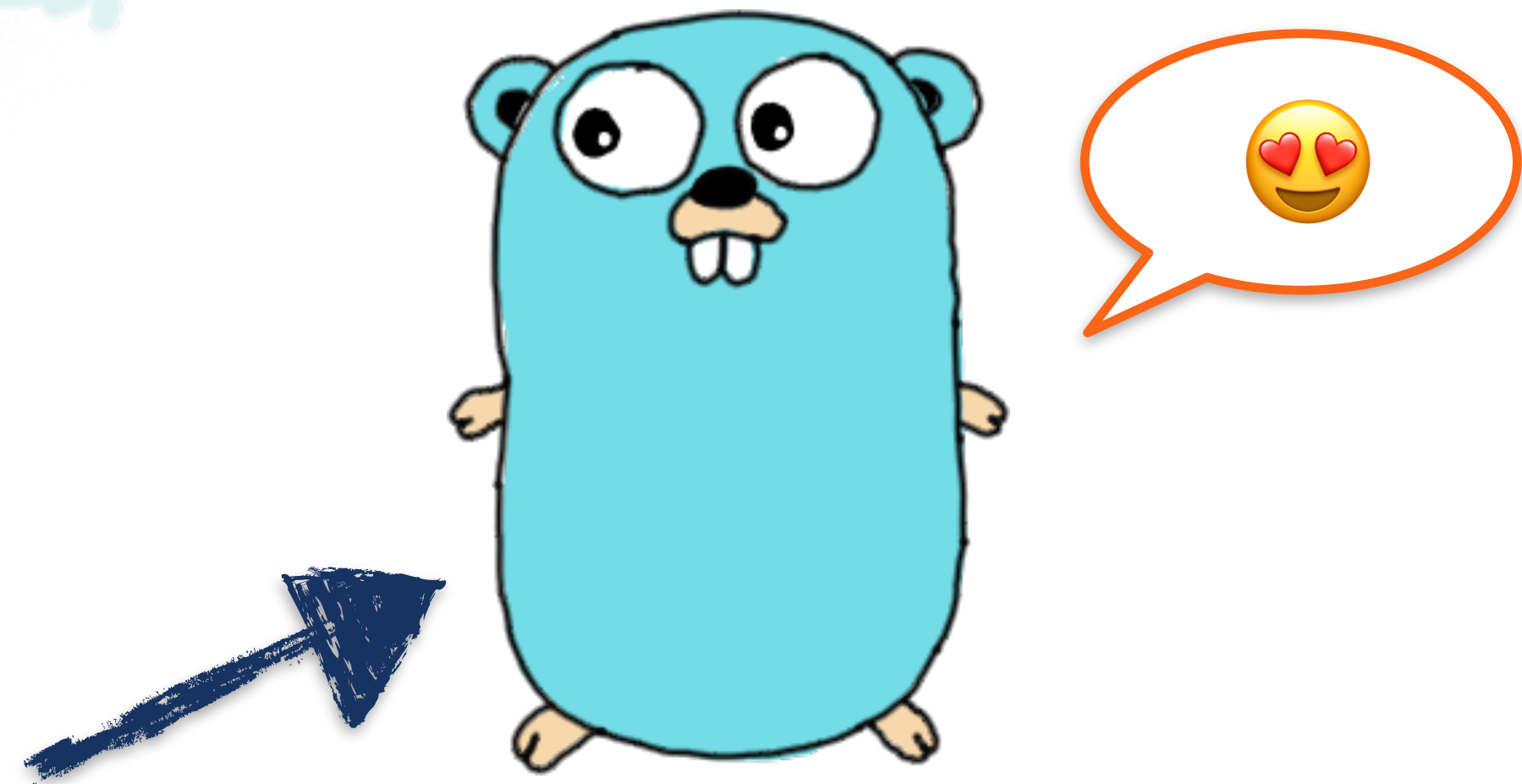


# My kingdom for a schema!



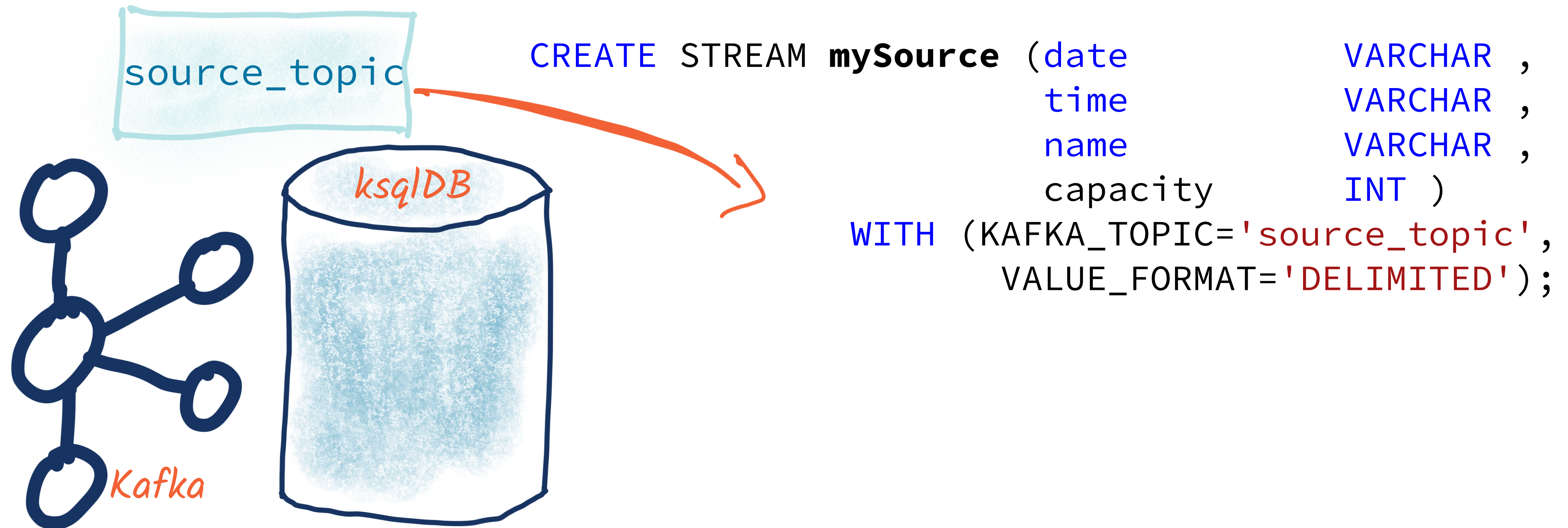
```
2020-10-14,12:28,Broadway,1132,921  
2020-10-14,12:28,Kirkgate Centre,611,474  
2020-10-14,12:28,Sharpe Street,98,63
```

```
{  
  "ts": "2020-10-14T12:28 UTC+1",  
  "name": "Broadway",  
  "capacity": 1132,  
  "empty": 921  
}  
...
```



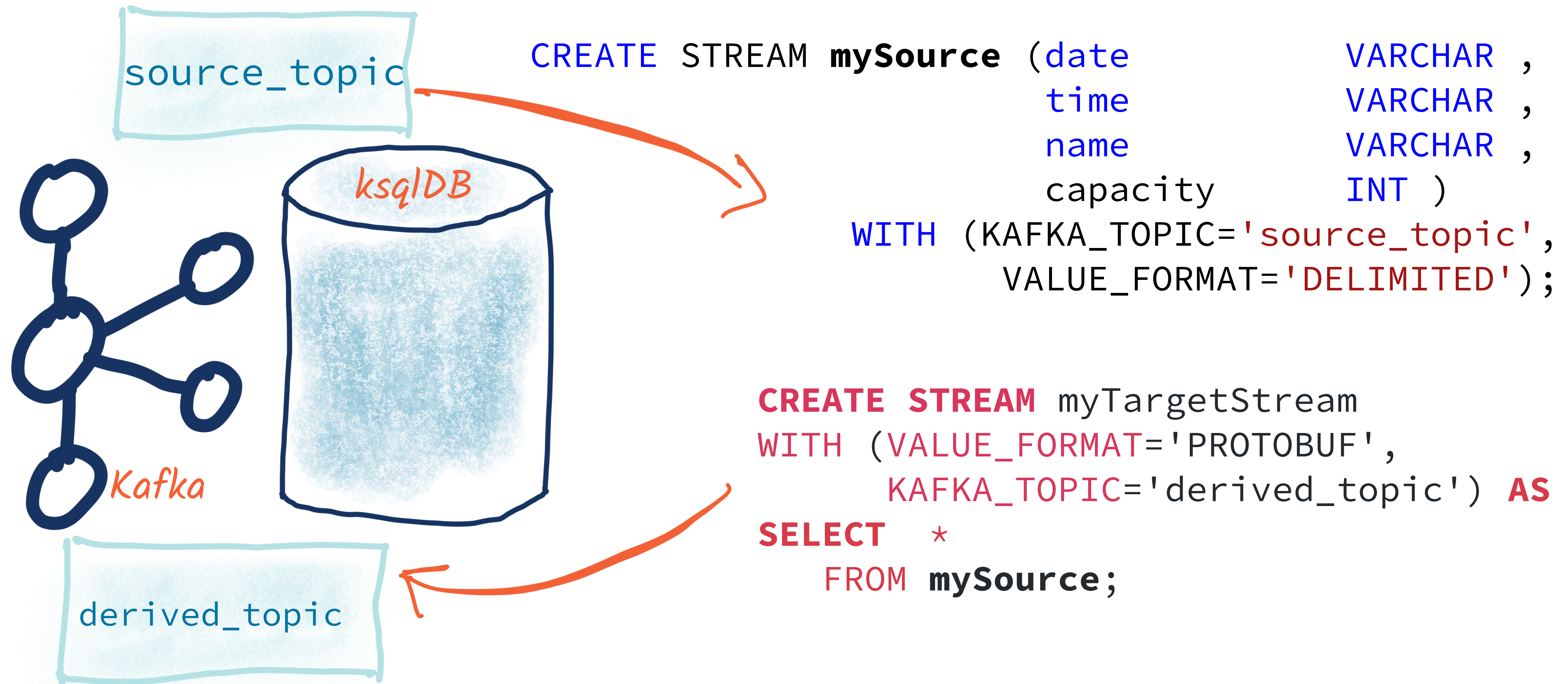


# Applying a schema to streams of data



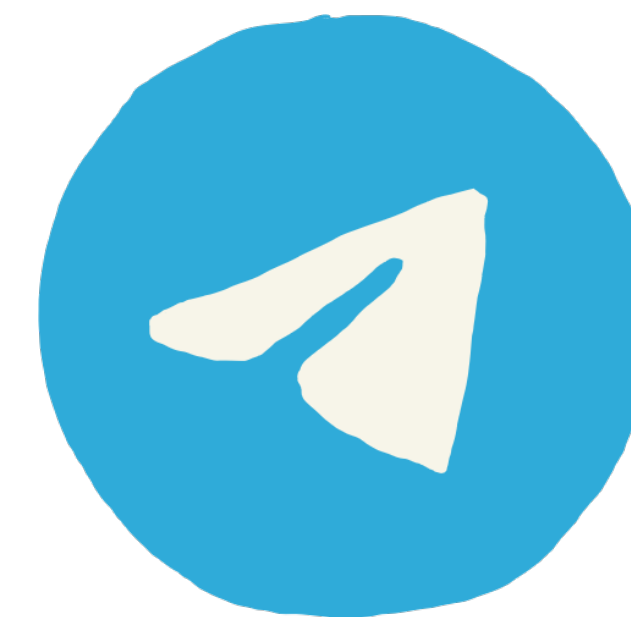
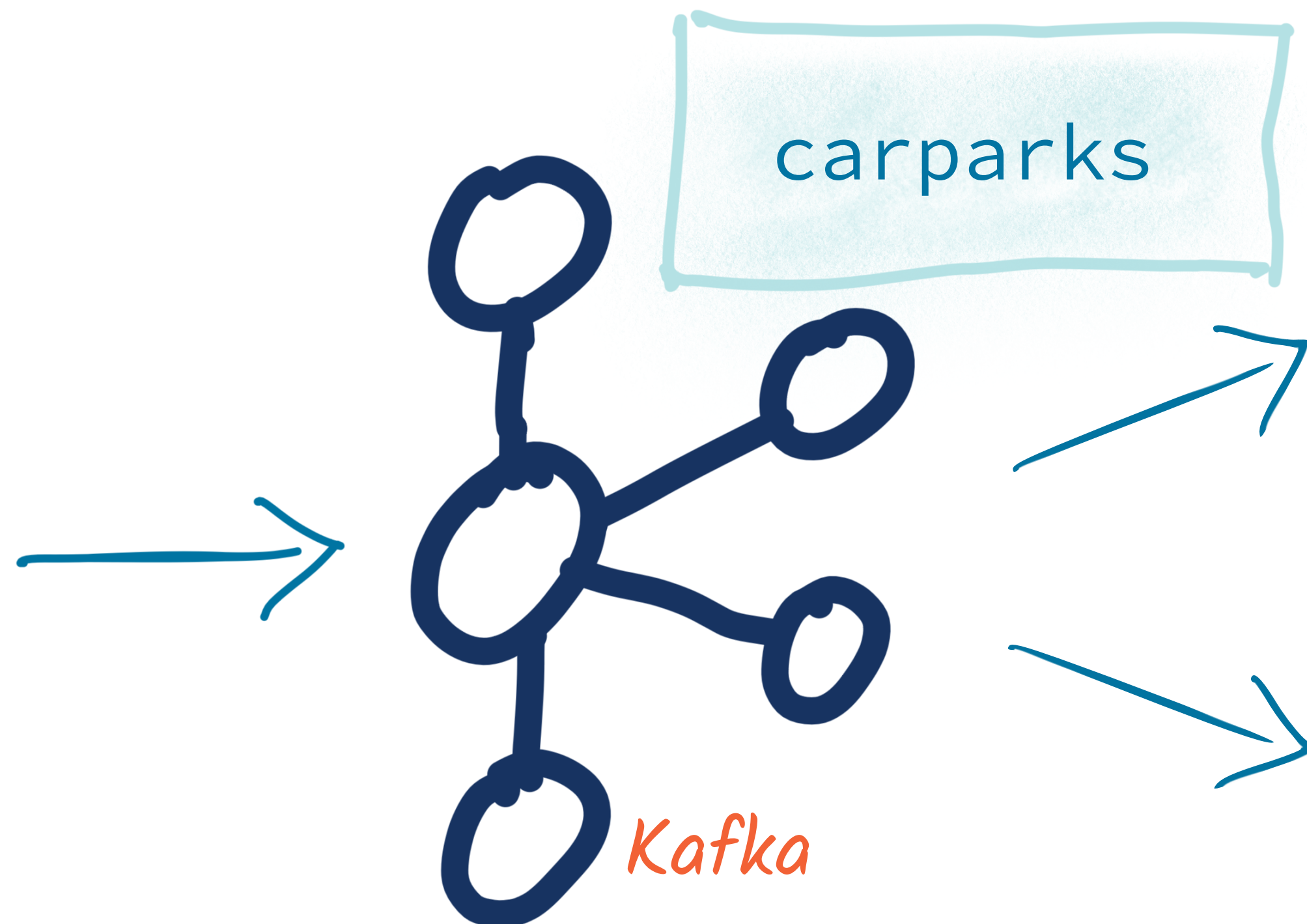


# Applying a schema to streams of data



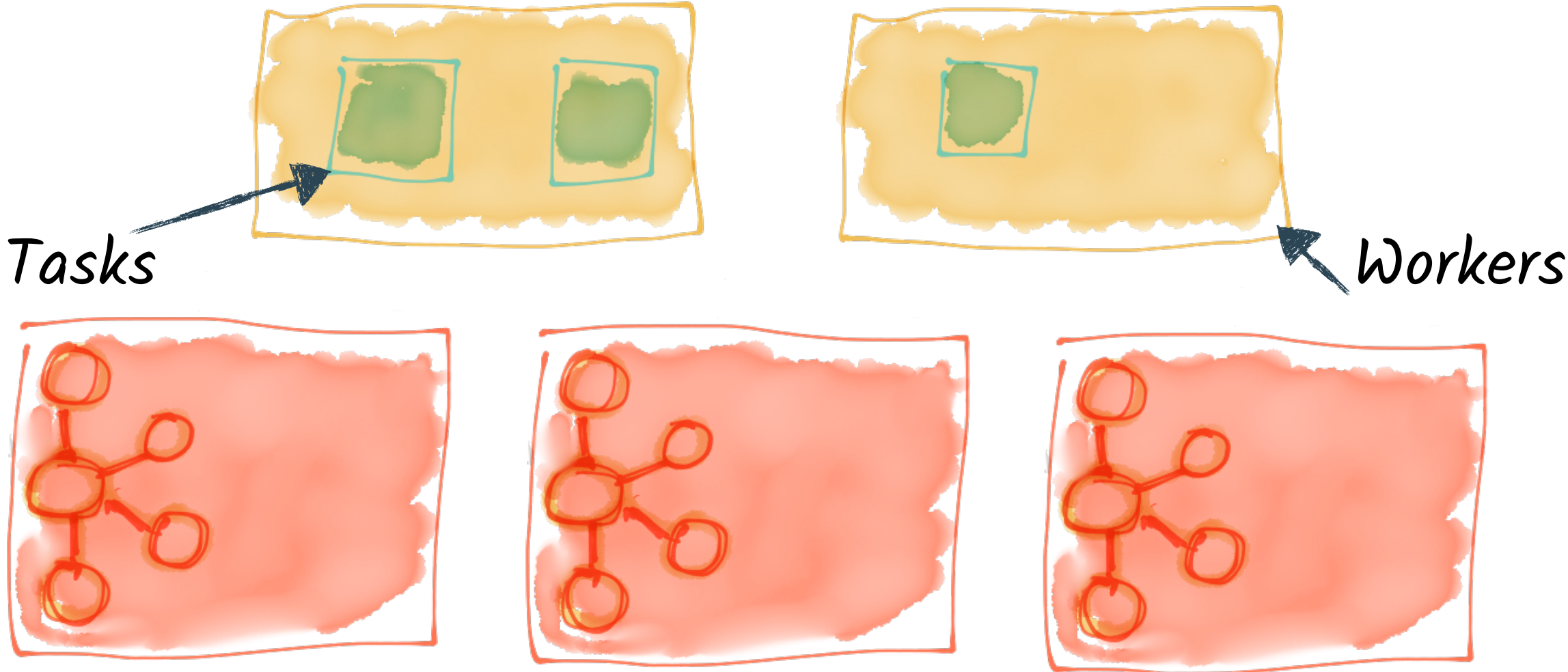


# Integration





# Streaming Integration with Kafka Connect

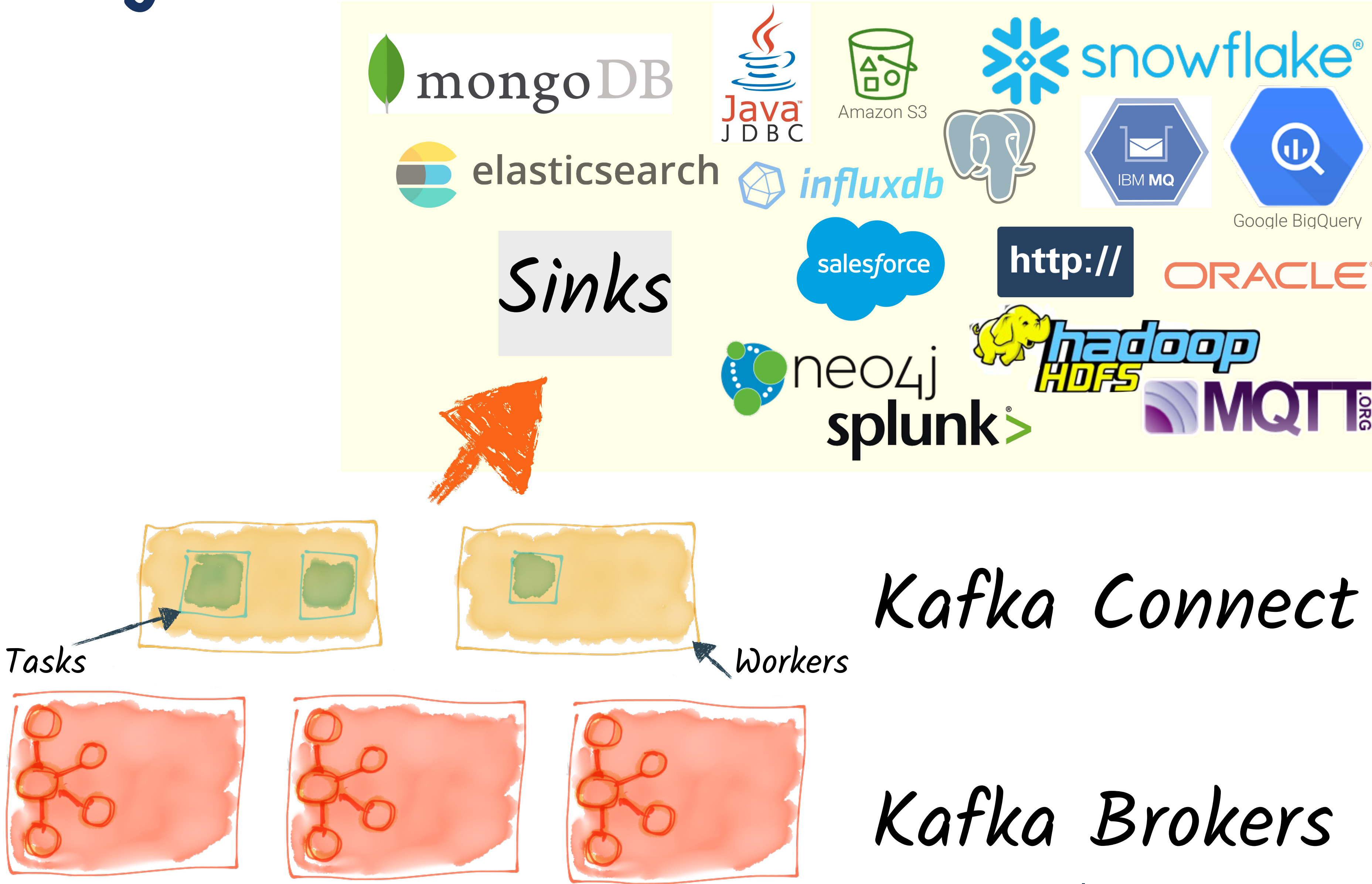


*Kafka Connect*

*Kafka Brokers*

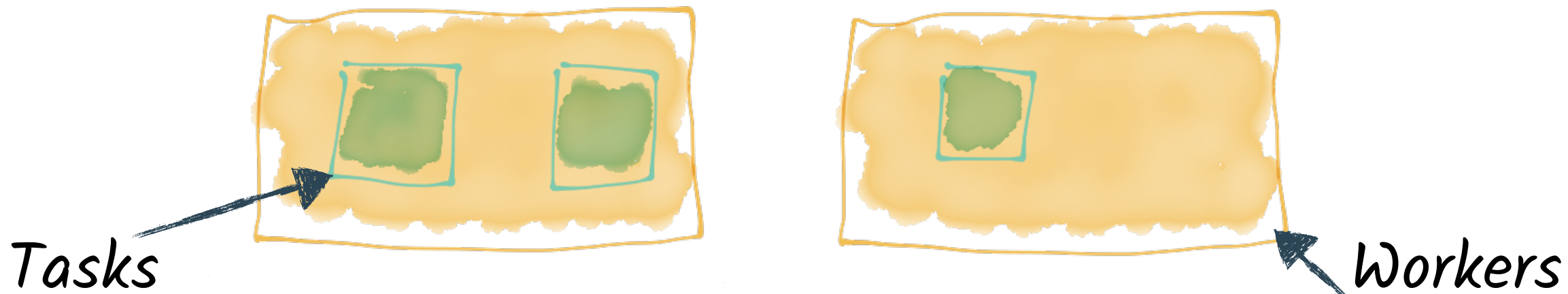


# Streaming Integration with Kafka Connect

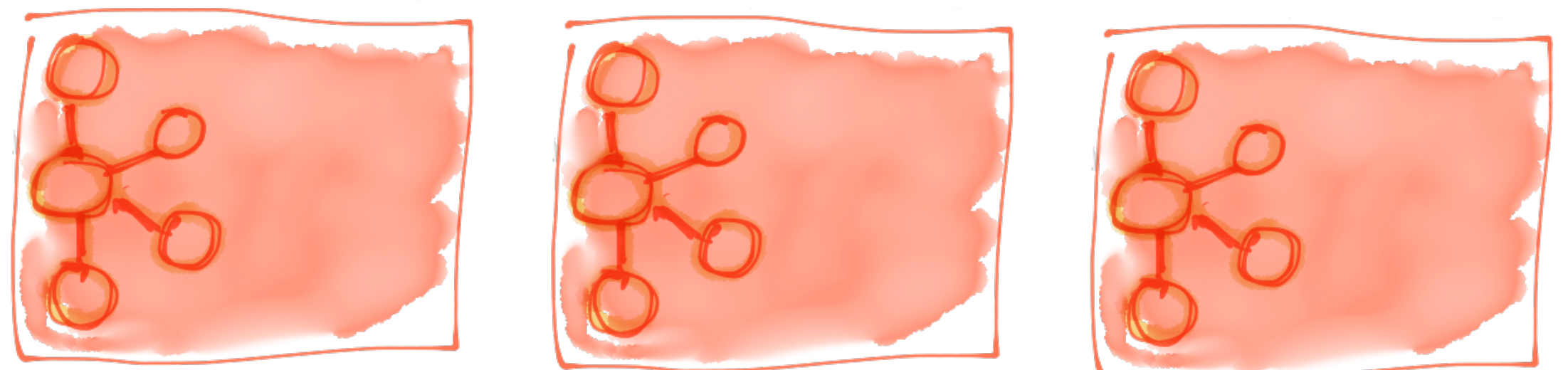




# Streaming Integration with Kafka Connect



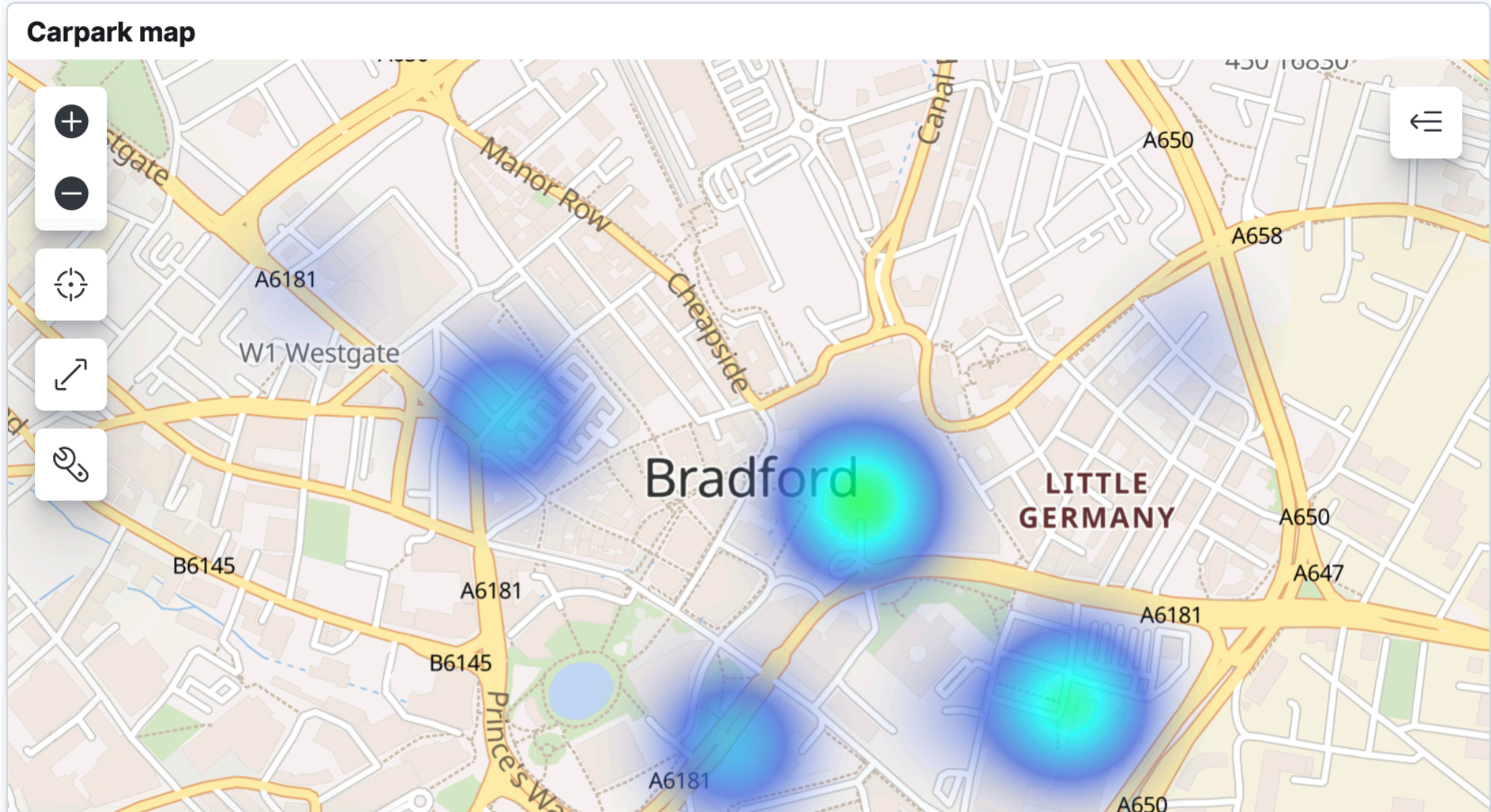
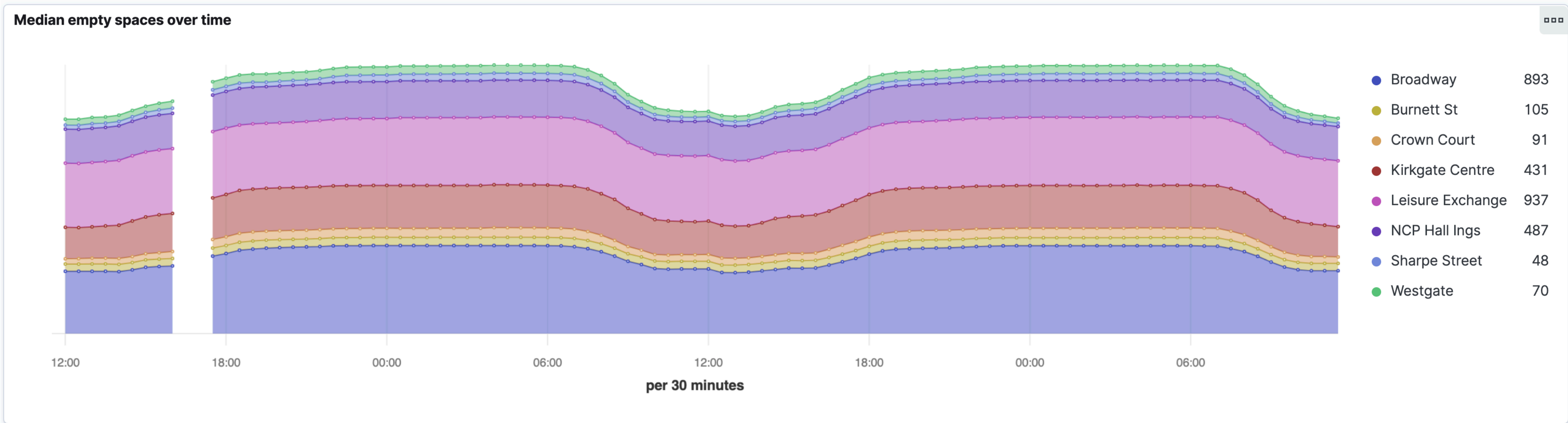
Kafka Connect



Kafka Brokers



# Streaming Analytics





Why build  
it this way?



# Events



# Streams of Events



We want to *react* to  
them as they happen



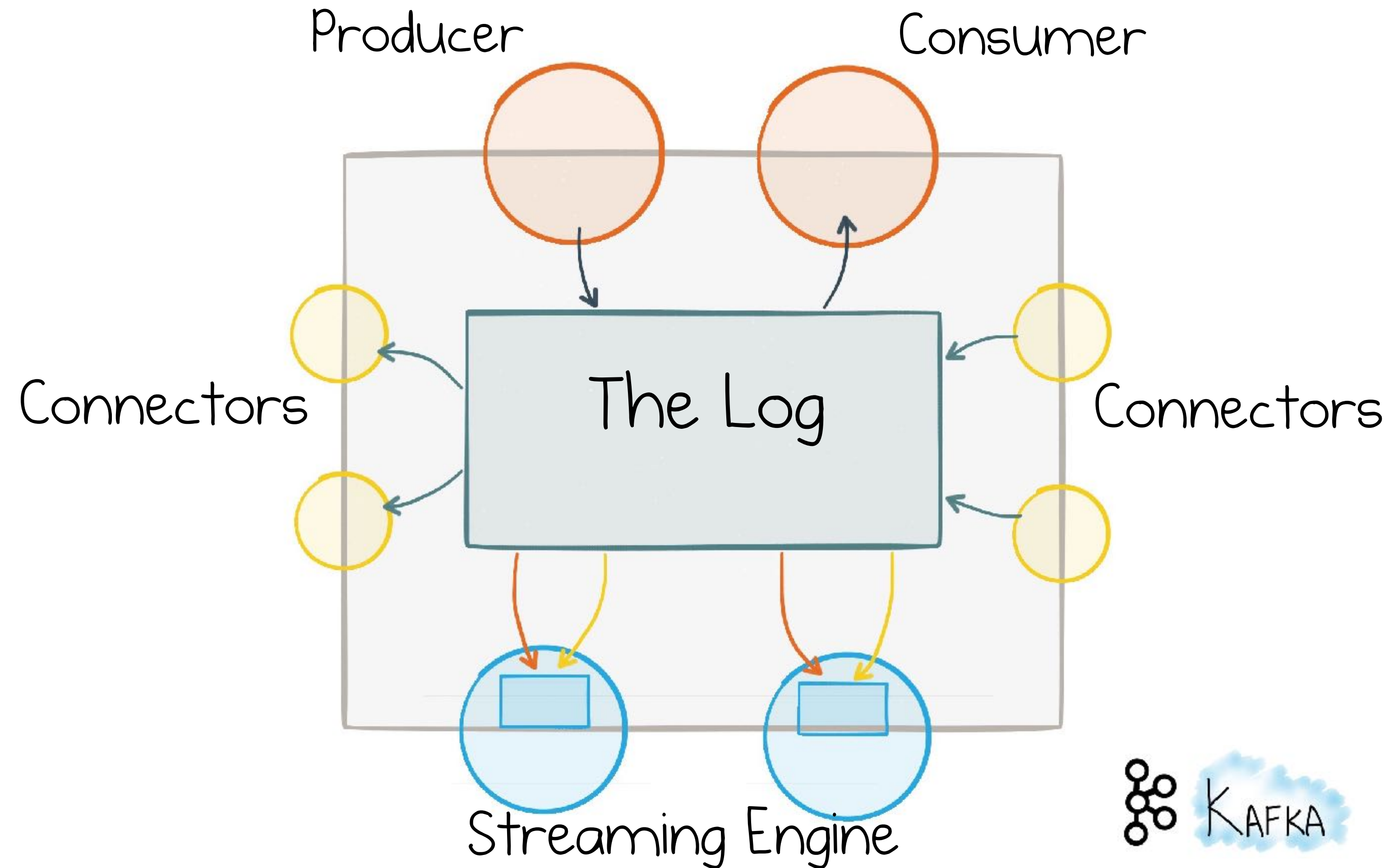
We want to build  
*state* from a stream  
of events



We want to provide  
the *latest* data in our  
analytics



# Apache Kafka - an Event Streaming Platform

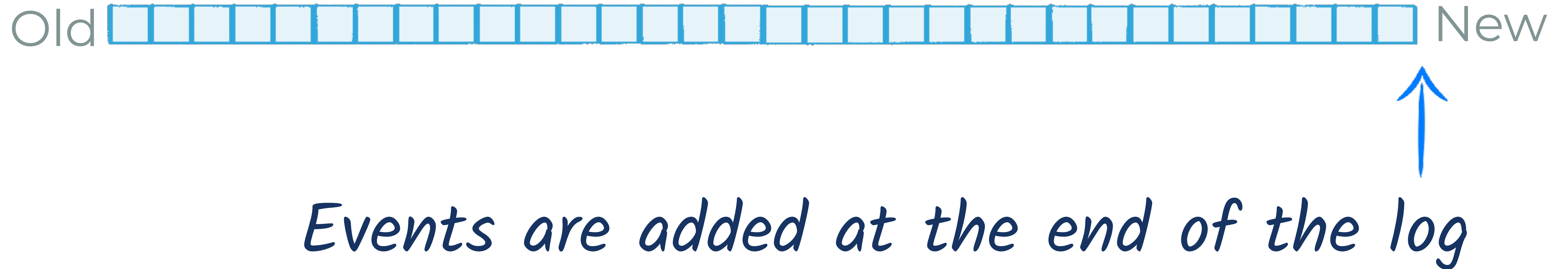




Why  
Kafka?



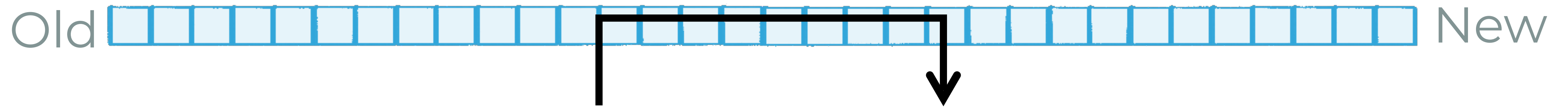
# *Distributed, Immutable, Event Log*





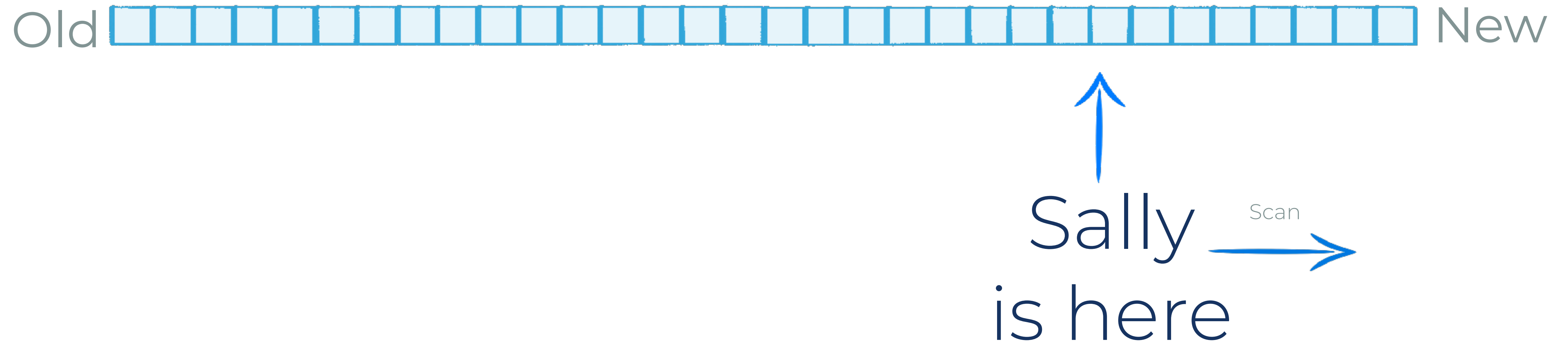
# Consumers can seek to any point

*Read to offset & scan*



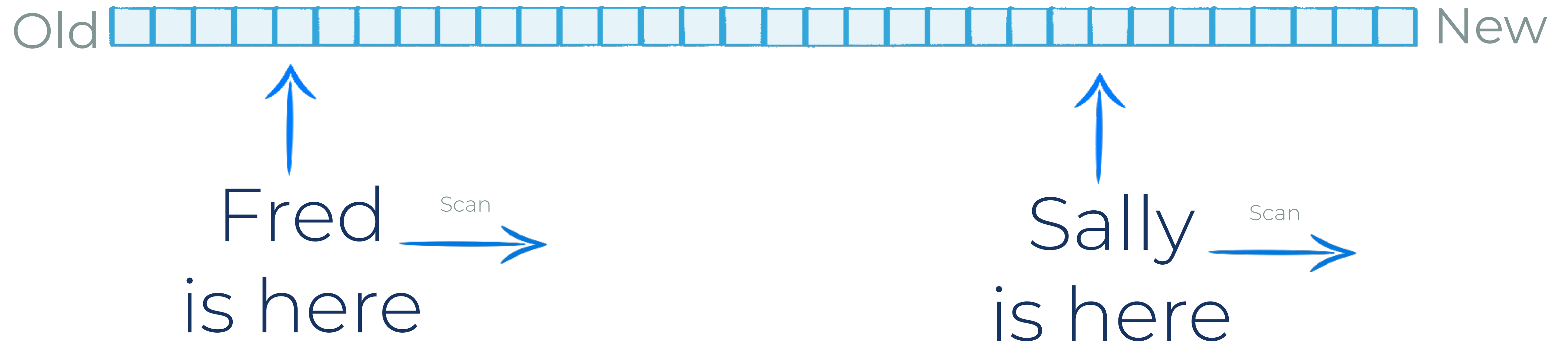


# *Data is not deleted once read*



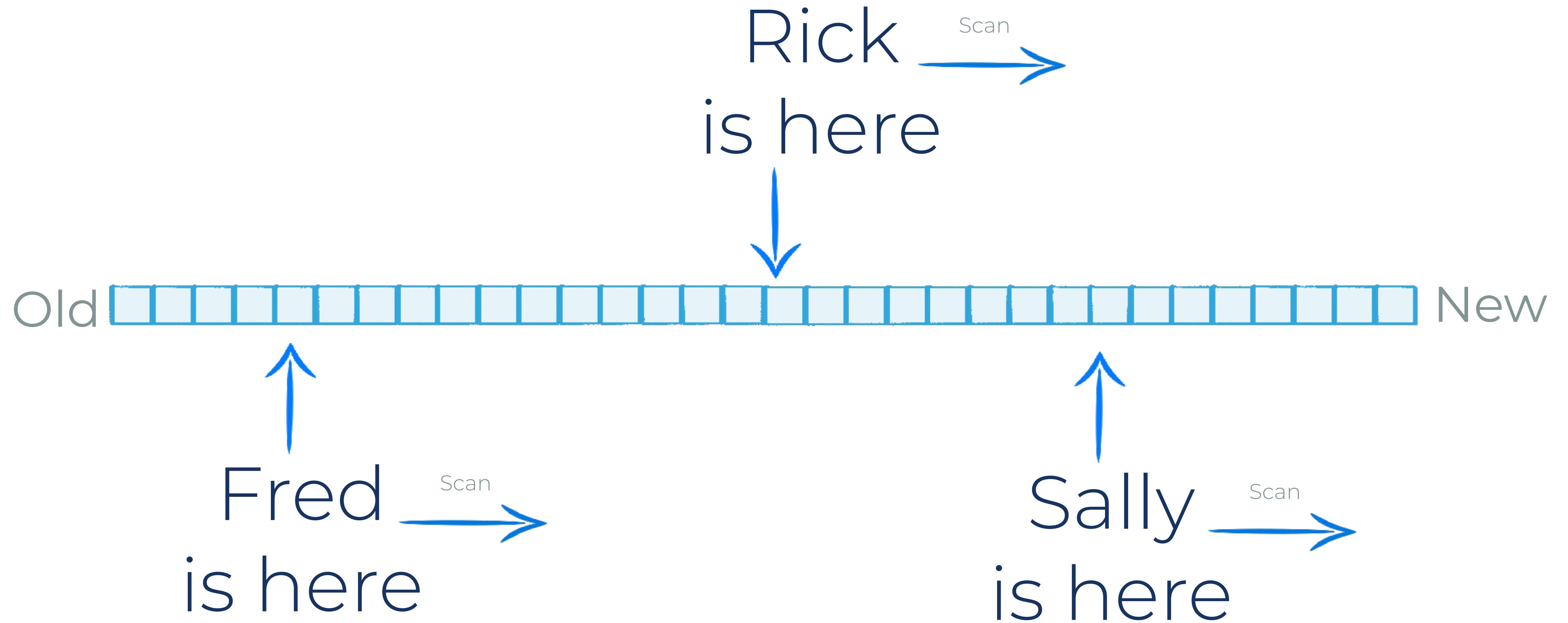


# Consumers are independent of each other





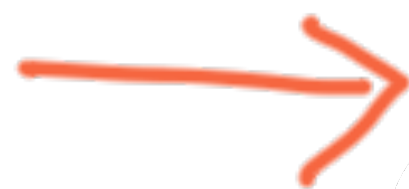
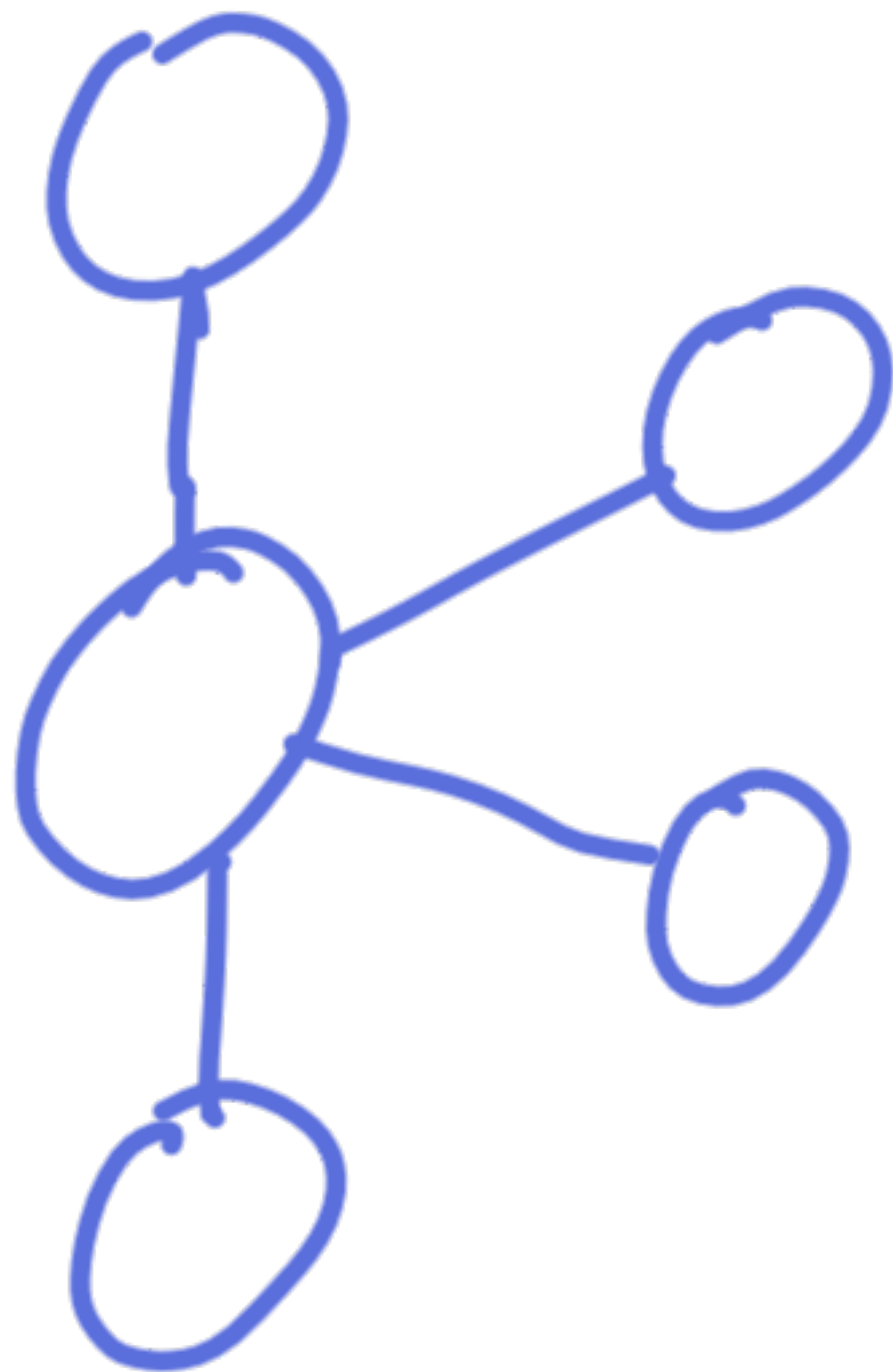
# Consumers can be added later





# Stream Processing with ksqlDB

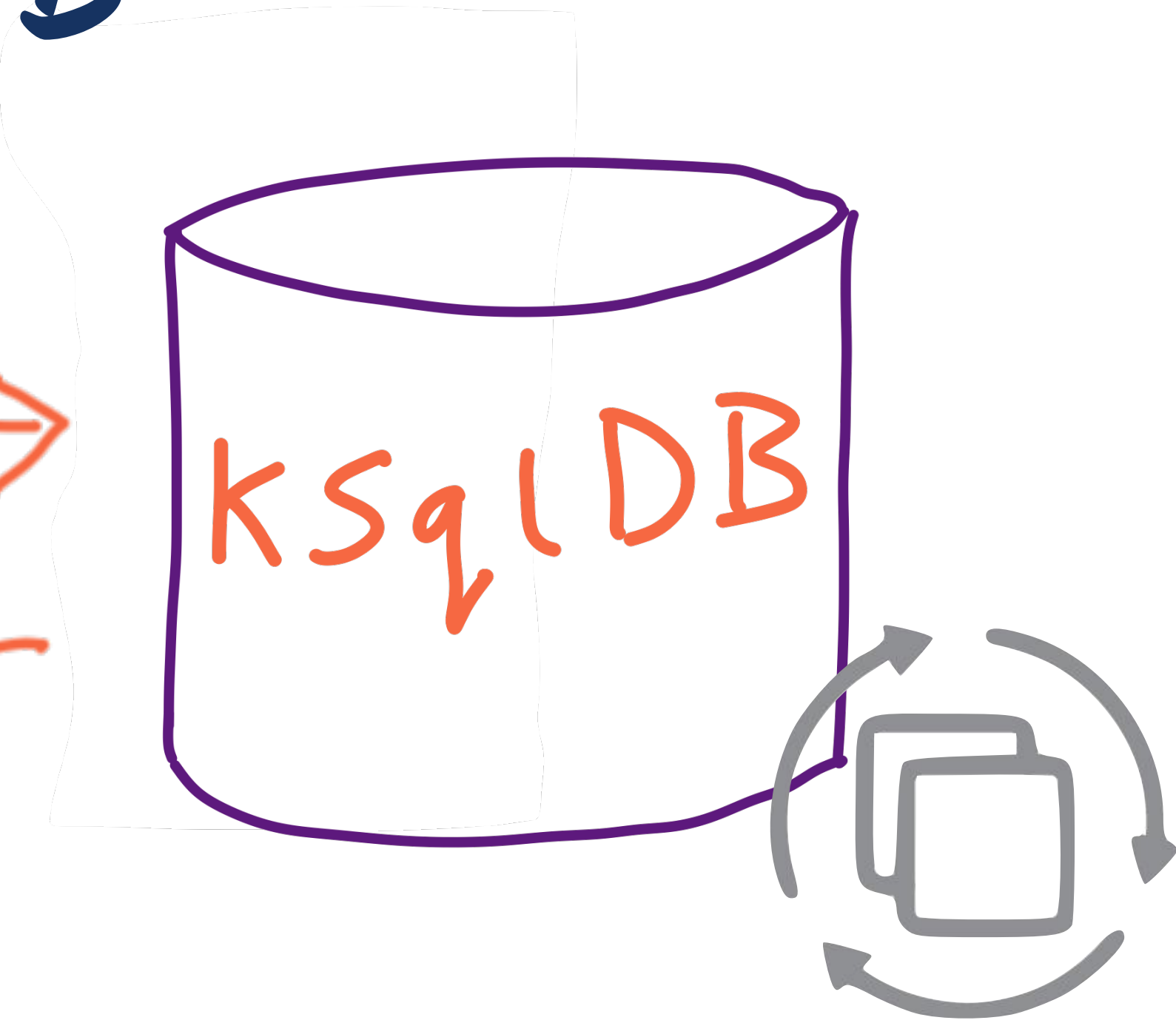
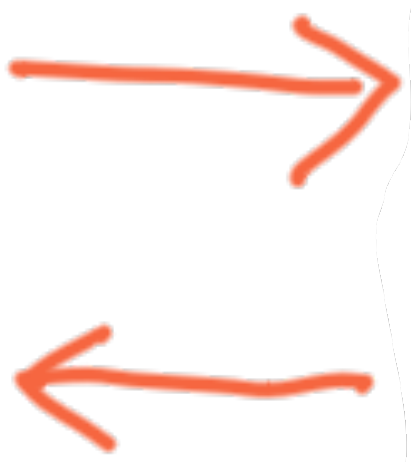
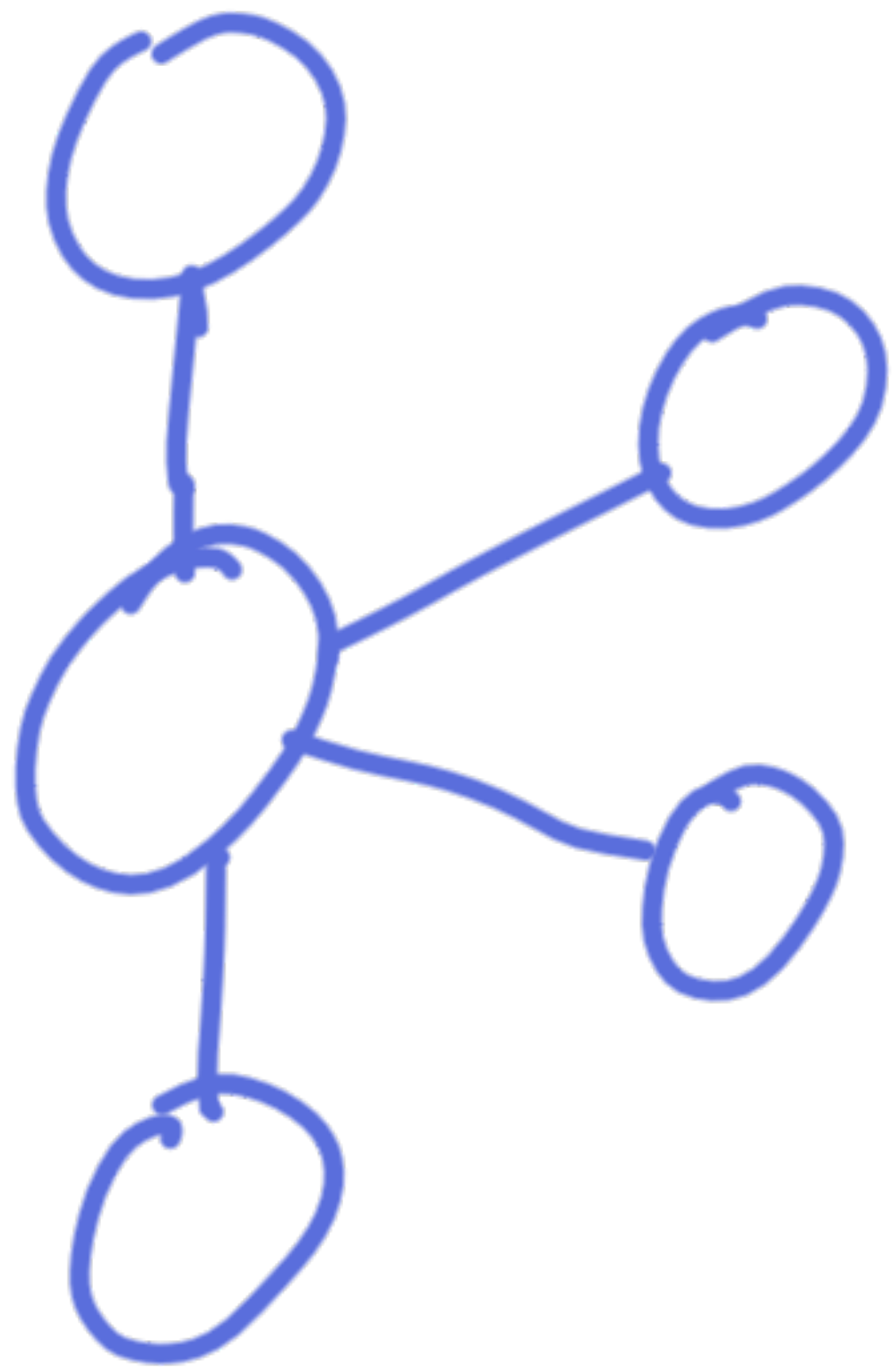
Source stream





# Stream Processing with ksqlDB

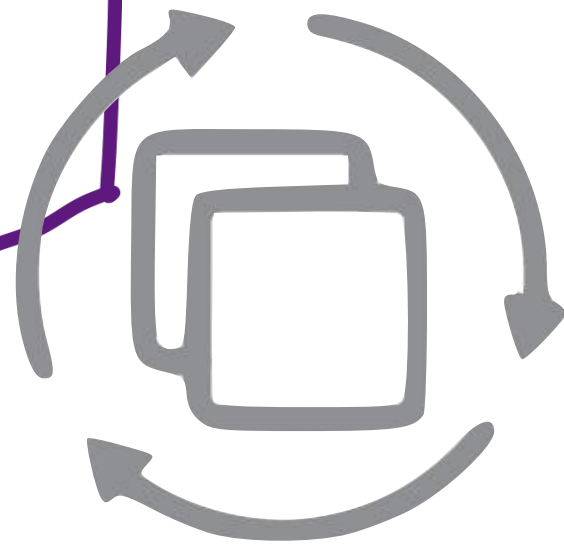
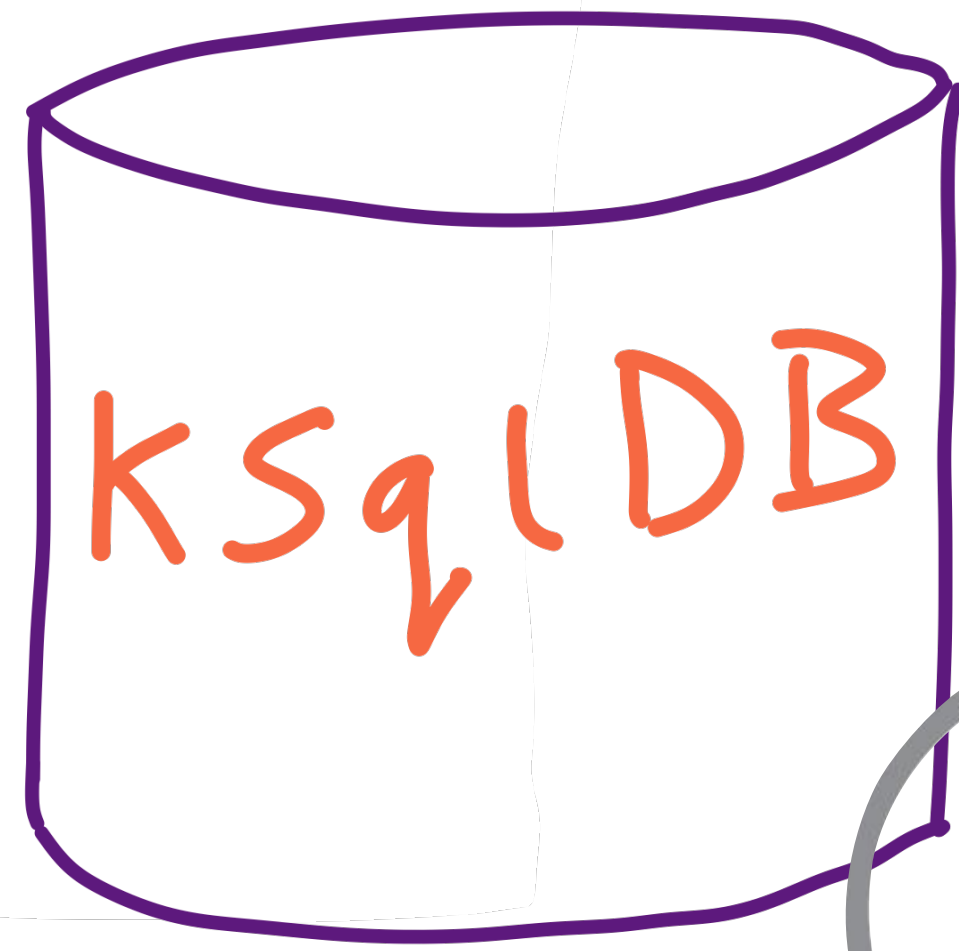
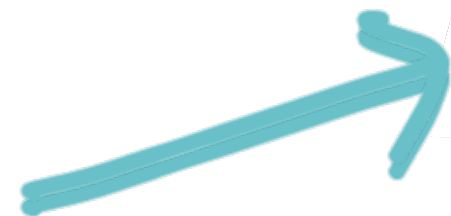
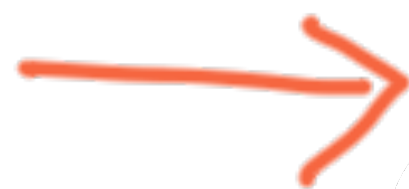
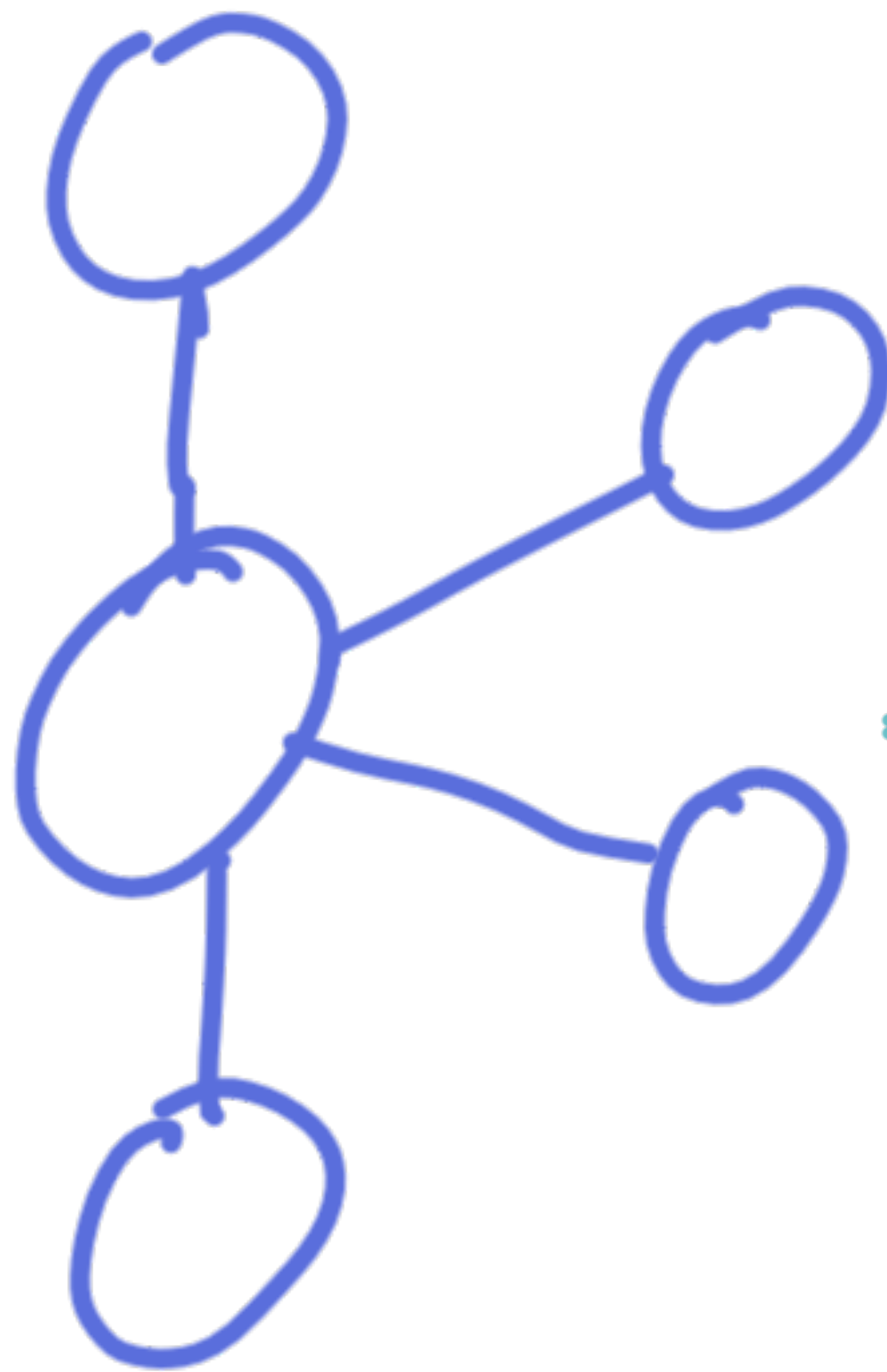
Source stream





# Stream Processing with ksqlDB

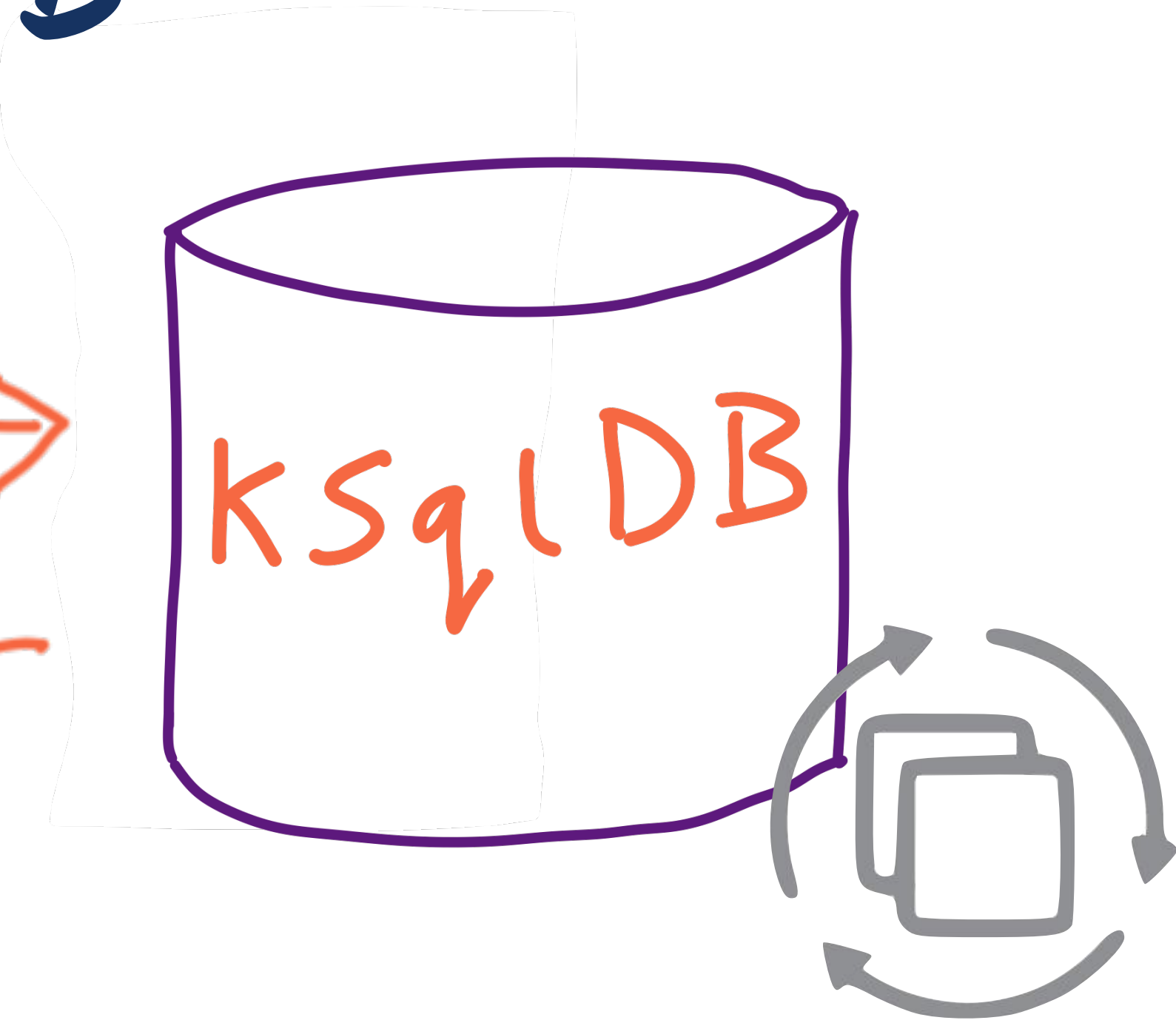
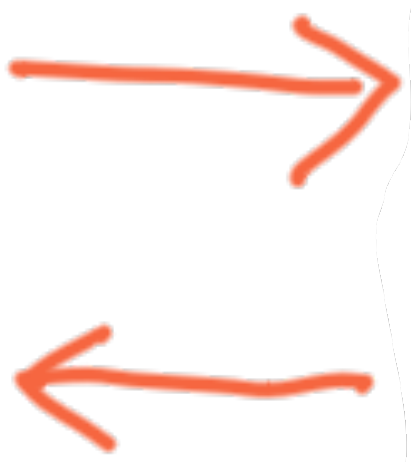
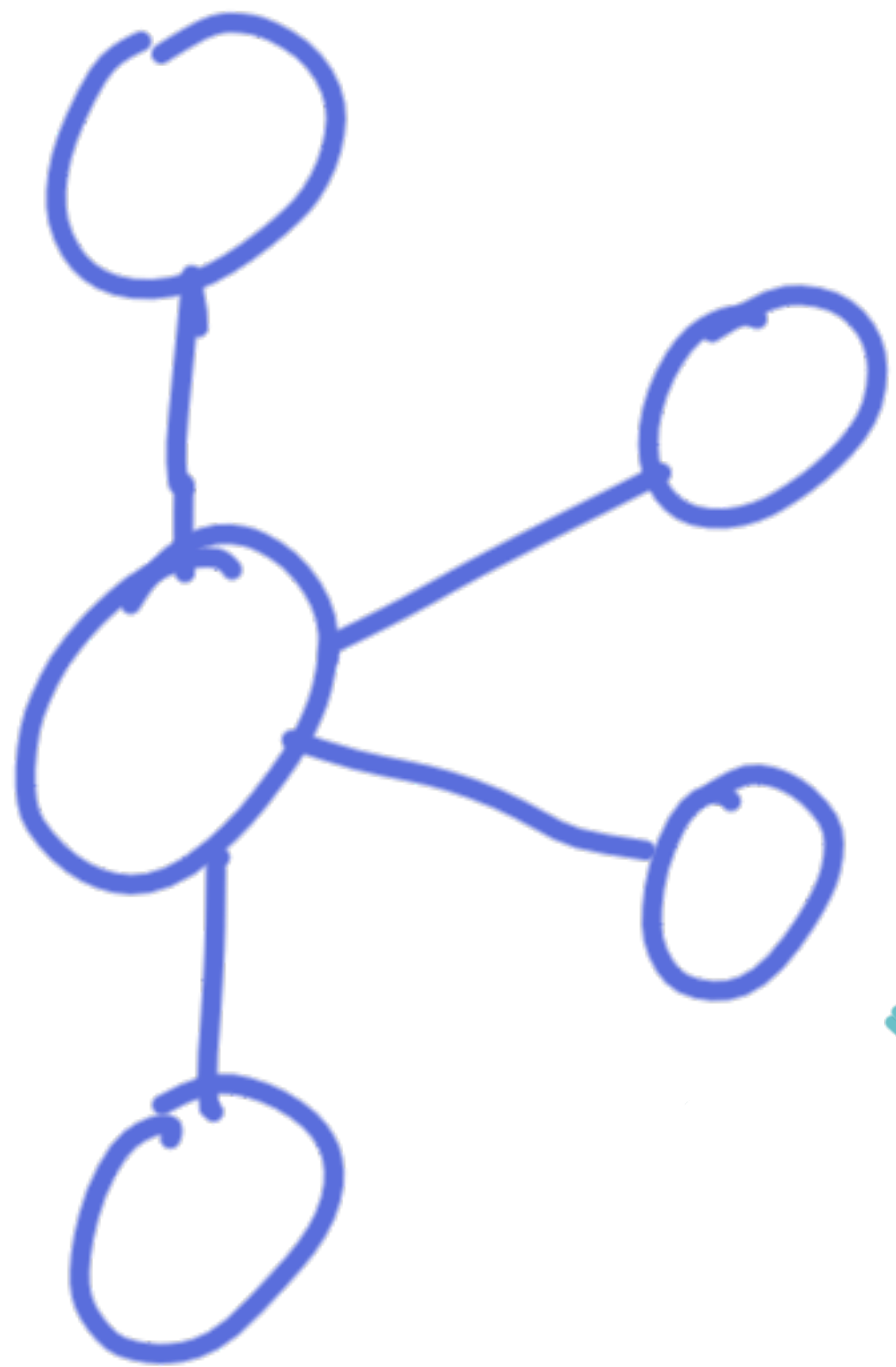
Source stream





# Stream Processing with ksqlDB

Source stream



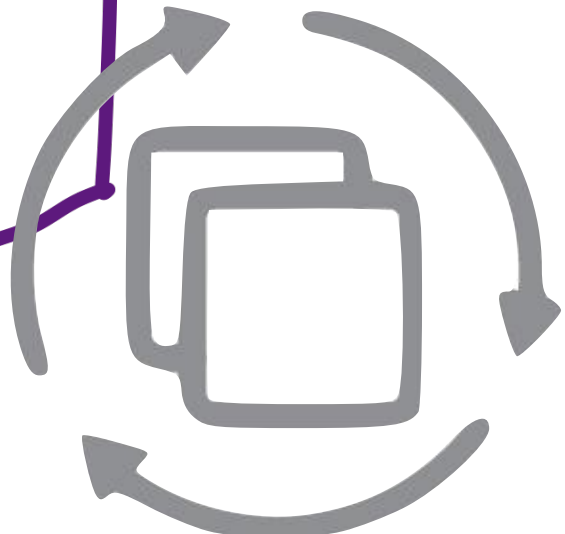
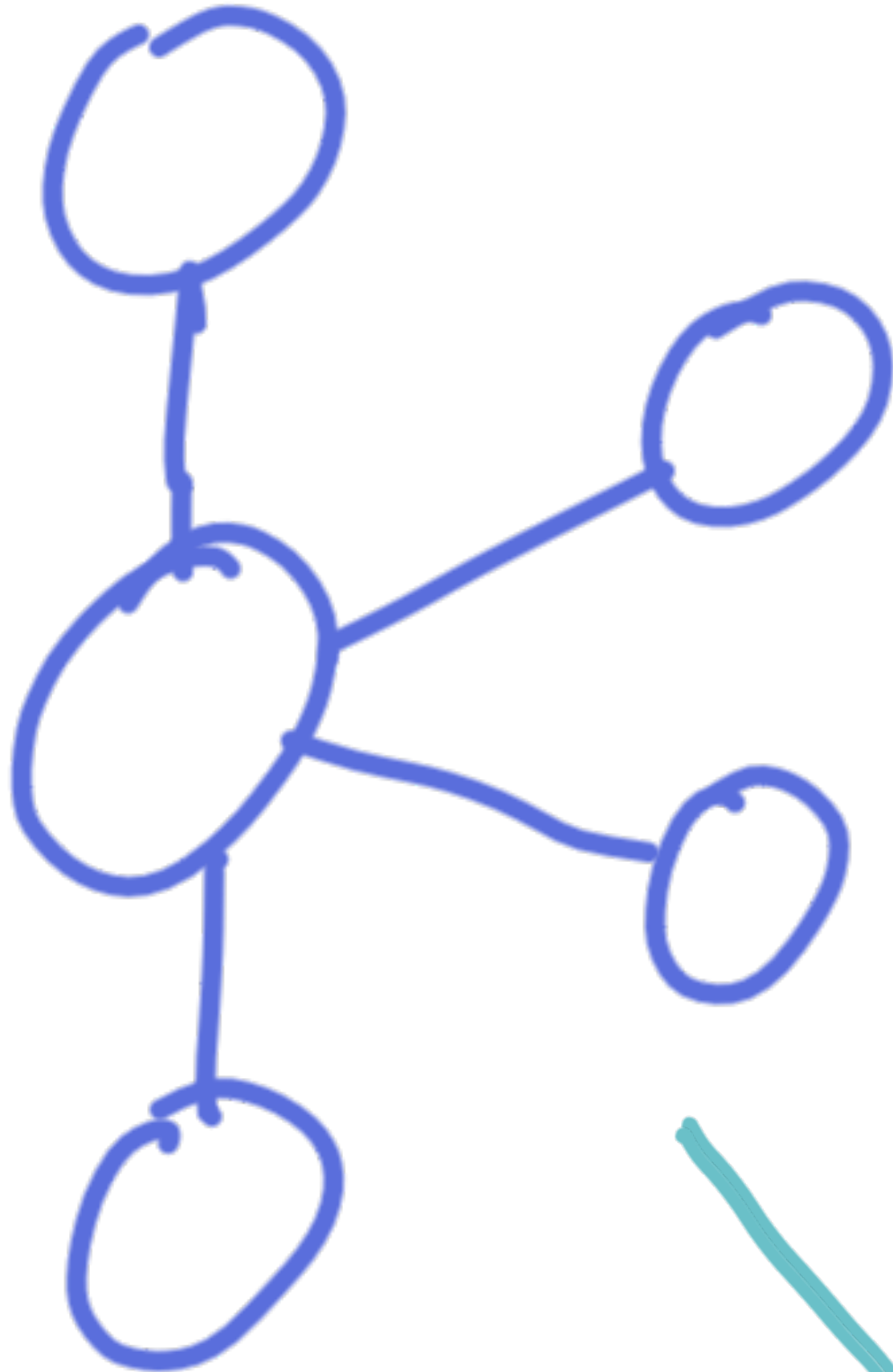
Analytics





# Stream Processing with ksqlDB

Source stream

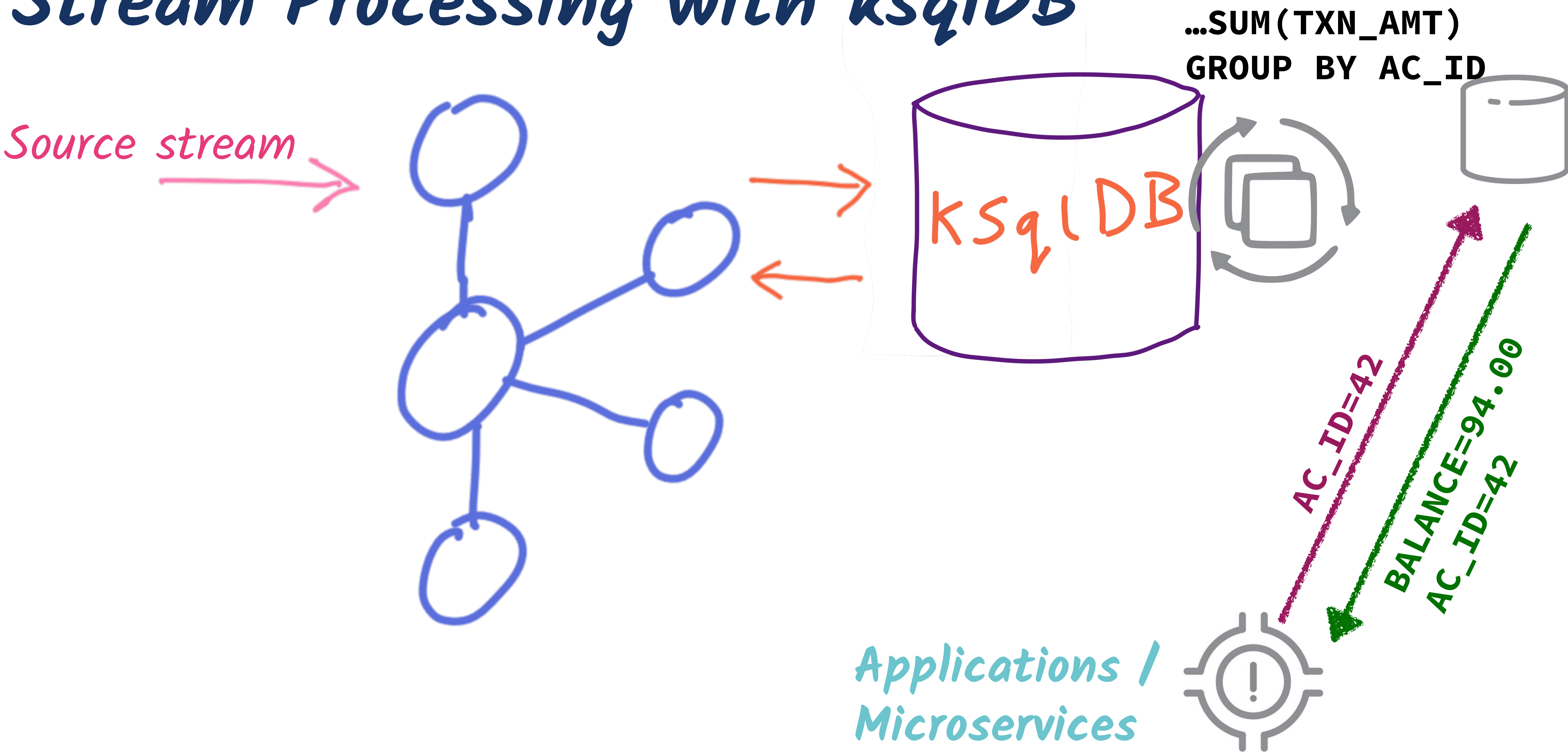


Applications /  
Microservices





# Stream Processing with ksqlDB



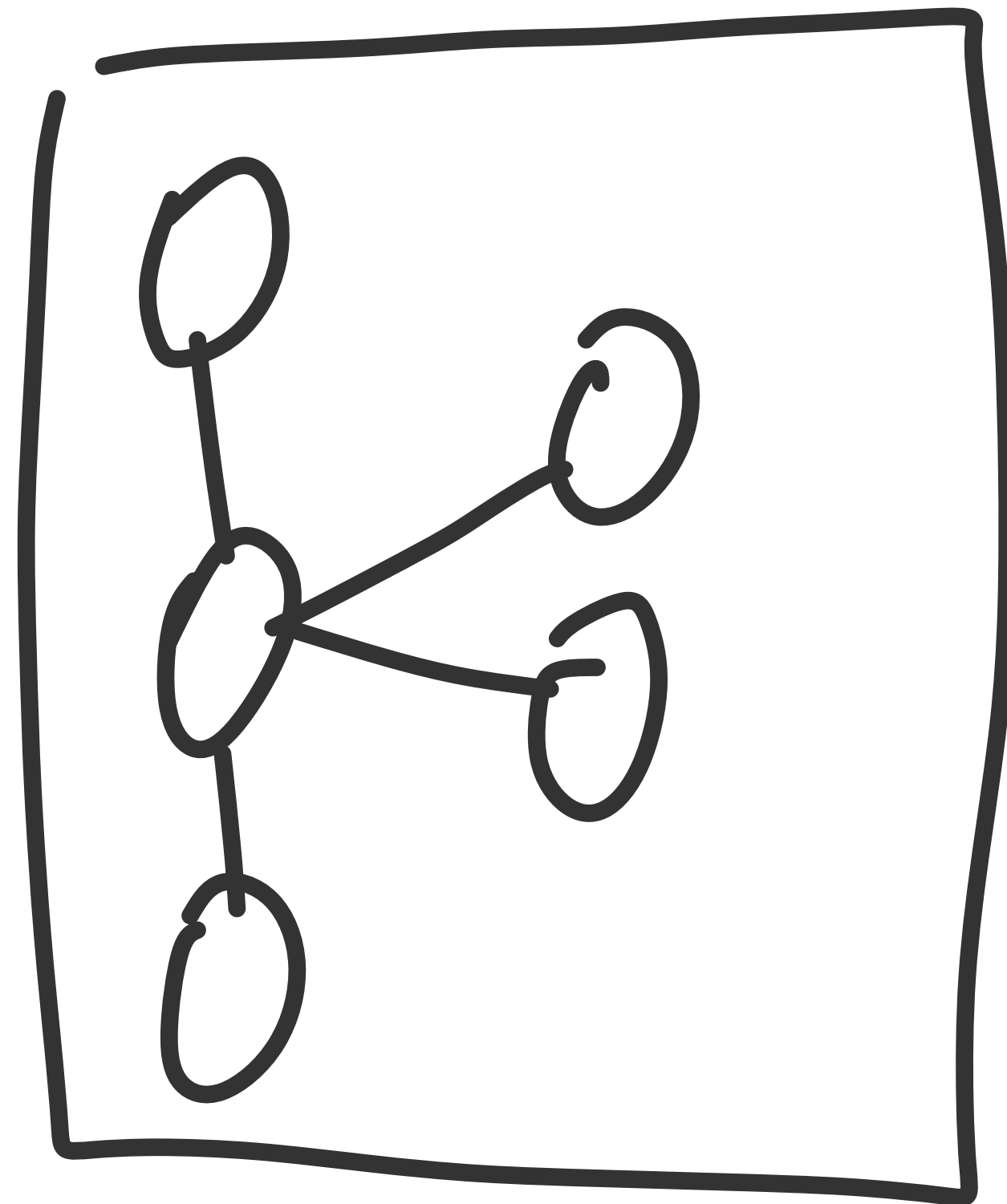


# *Under the covers of ksqlDB*



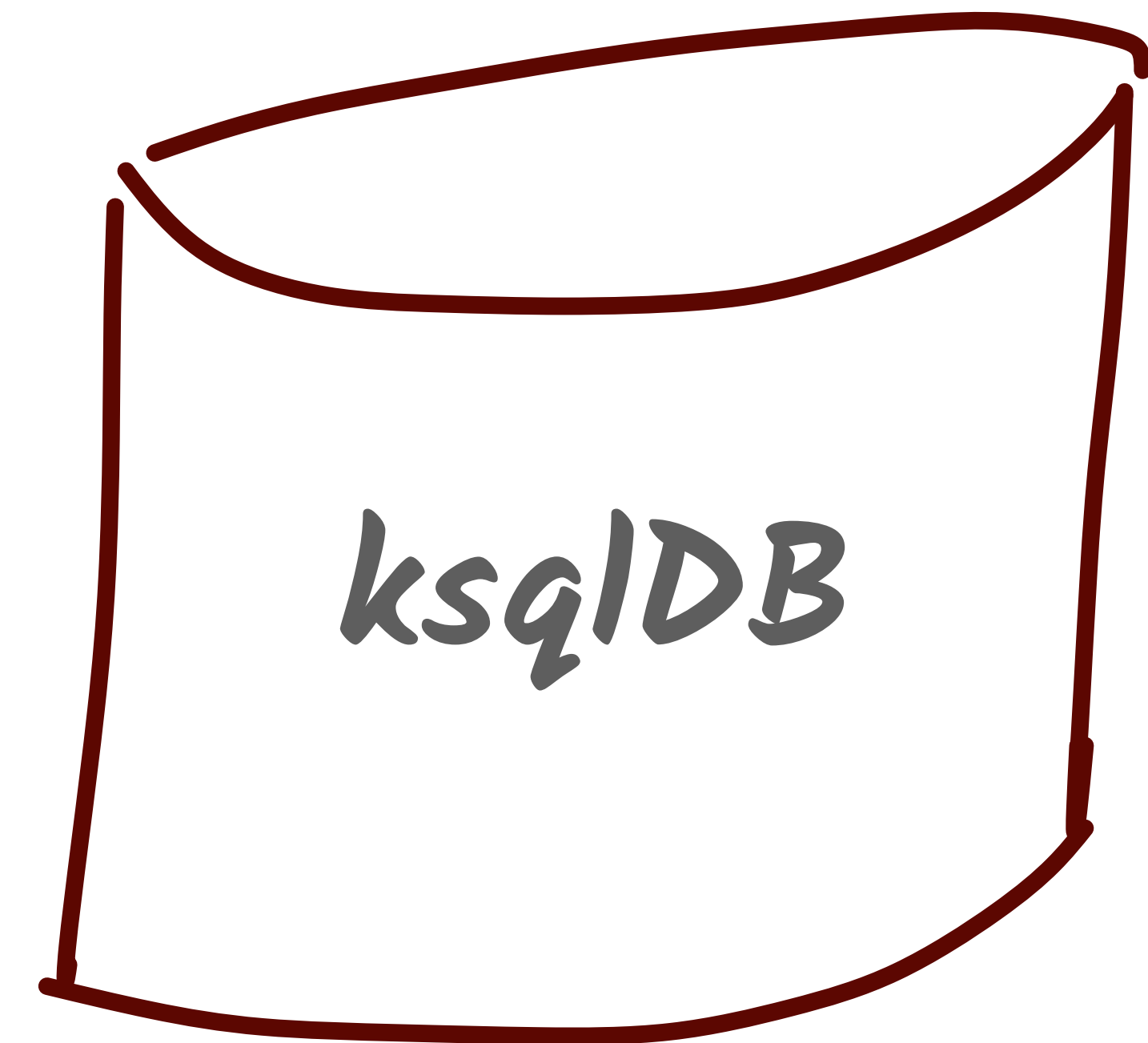


Kafka cluster



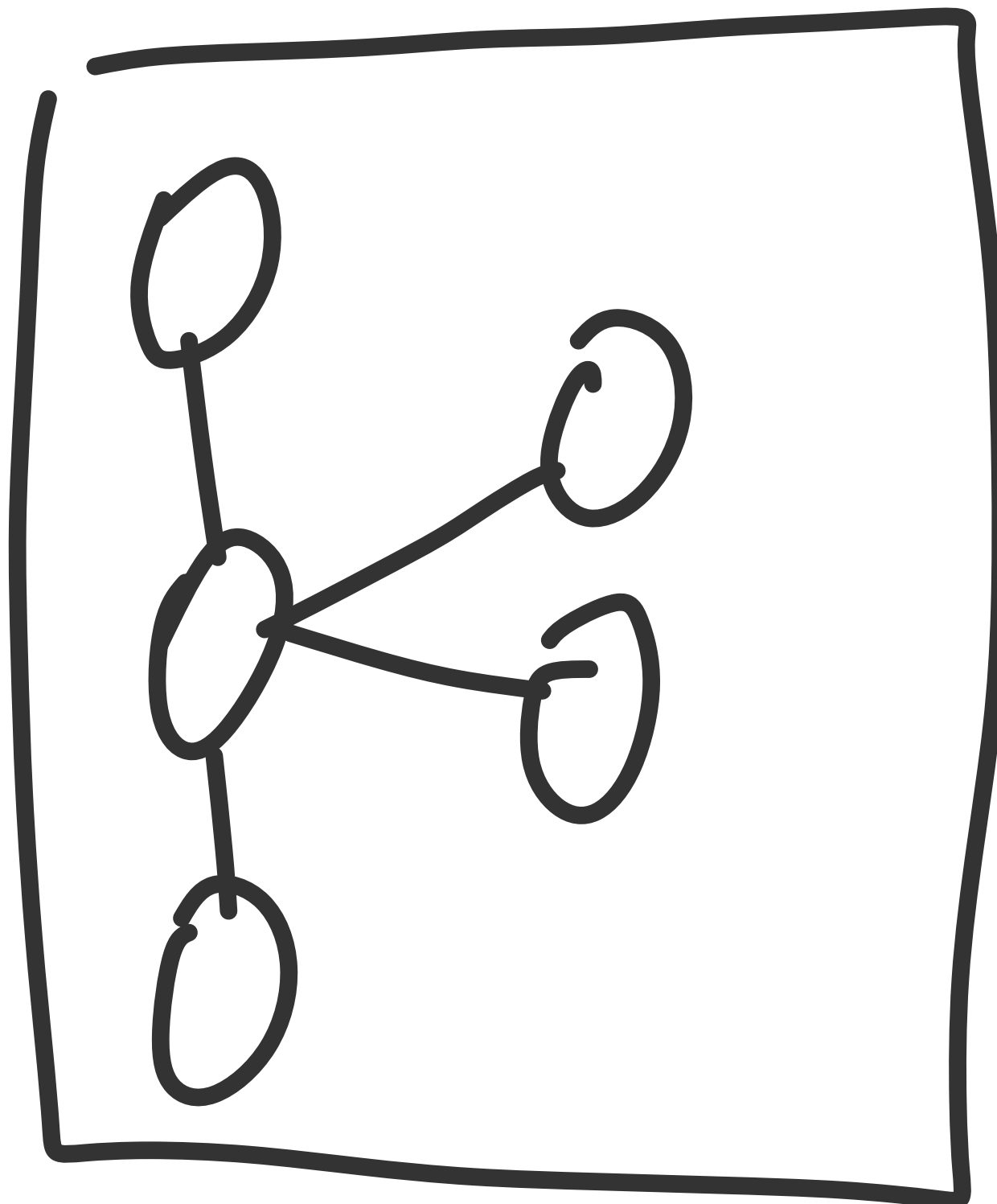
consume

produce





Kafka cluster



consume

produce

JVM

ksqlDB



RocksDB



Kafka Streams





**WHERE CAN YOU RUN KSQLDB?**



**ANYWHERE!**



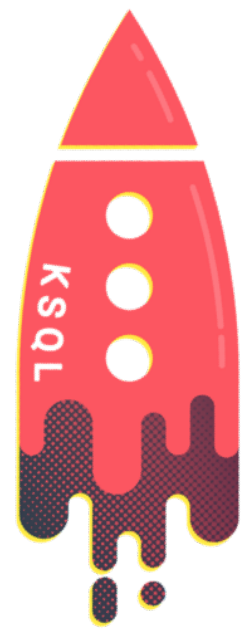


confluent cloud

# Fully Managed Kafka<sup>& KSQLDB</sup> as a Service



# Running ksqlDB - self-managed



ksqlDB Server  
(JVM process)

DEB, RPM, ZIP, TAR downloads

<http://confluent.io/download>

Docker images

[confluentinc/ksqldb-server](#)



Physical



docker



kubernetes



openstack®

vmware®



Microsoft  
Azure



Google Cloud Platform



amazon  
web services

...and many more...



Why  
Kafka?



*Stream Store*

*Process Integrate*



*Stream*

*Store*

*Process Integrate*



*Stream Store*

*Process*

*Integrate*



*Stream*

*Store*

*Process Integrate*

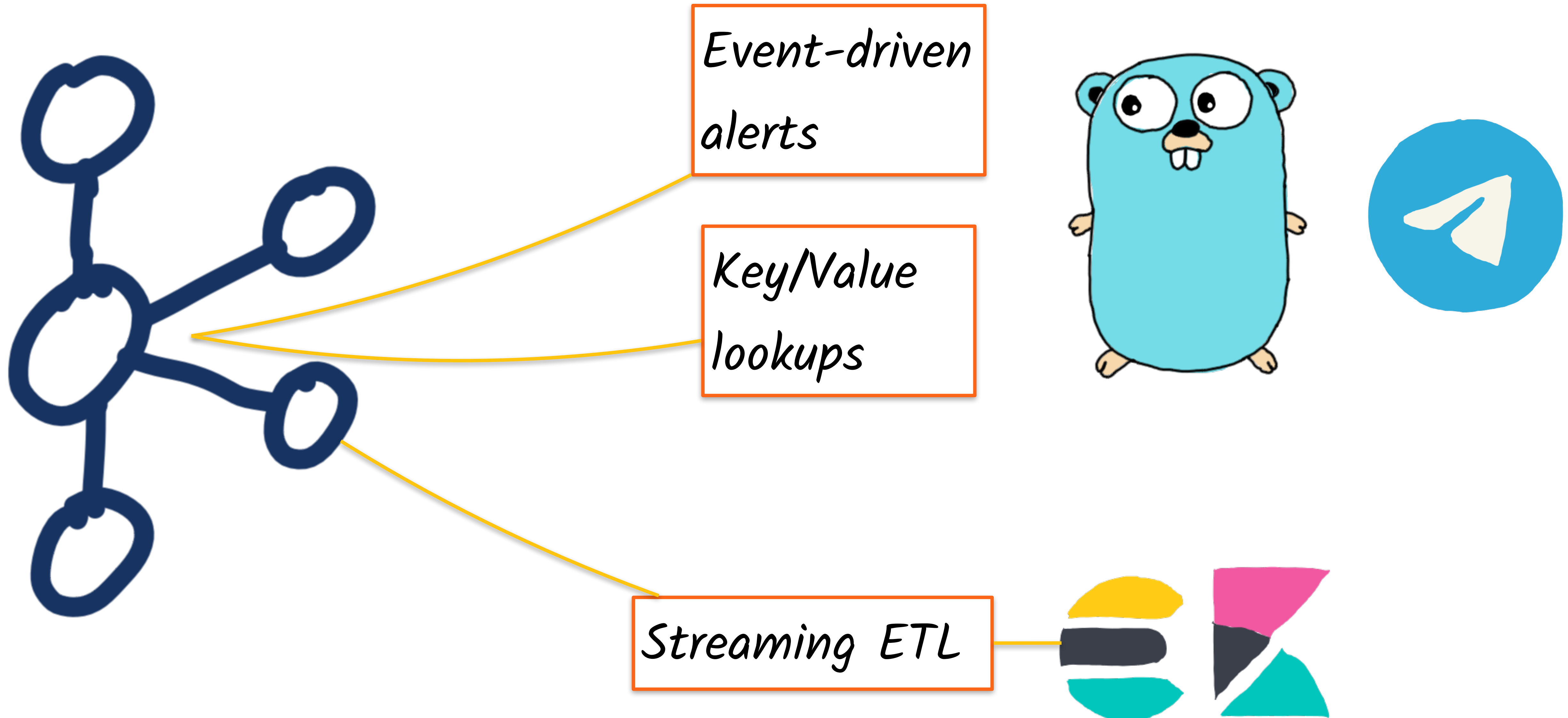


*Stream Store*

*Process Integrate*



# Flexible, event-driven applications





*Want to  
learn  
more?*





*Try it out for yourself*

<https://rmoff.dev/carparks>



60DEVADV

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

<https://rmoff.dev/ccloud>

Free money!  
(additional \$60 towards  
your bill 😊)



confluent cloud

# Fully Managed Kafka as a Service





# Learn Kafka.

Start building with  
Apache Kafka at  
Confluent Developer.



[developer.confluent.io](https://developer.confluent.io)



# *Confluent Community Slack group*



**[cnfl.io/slack](https://cnfl.io/slack)**






# Further reading / watching

<https://rmoff.dev/kafka-talks>

## 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](#)



**#EOF**

*<https://talks.rmoff.net>*

*@rmoff*