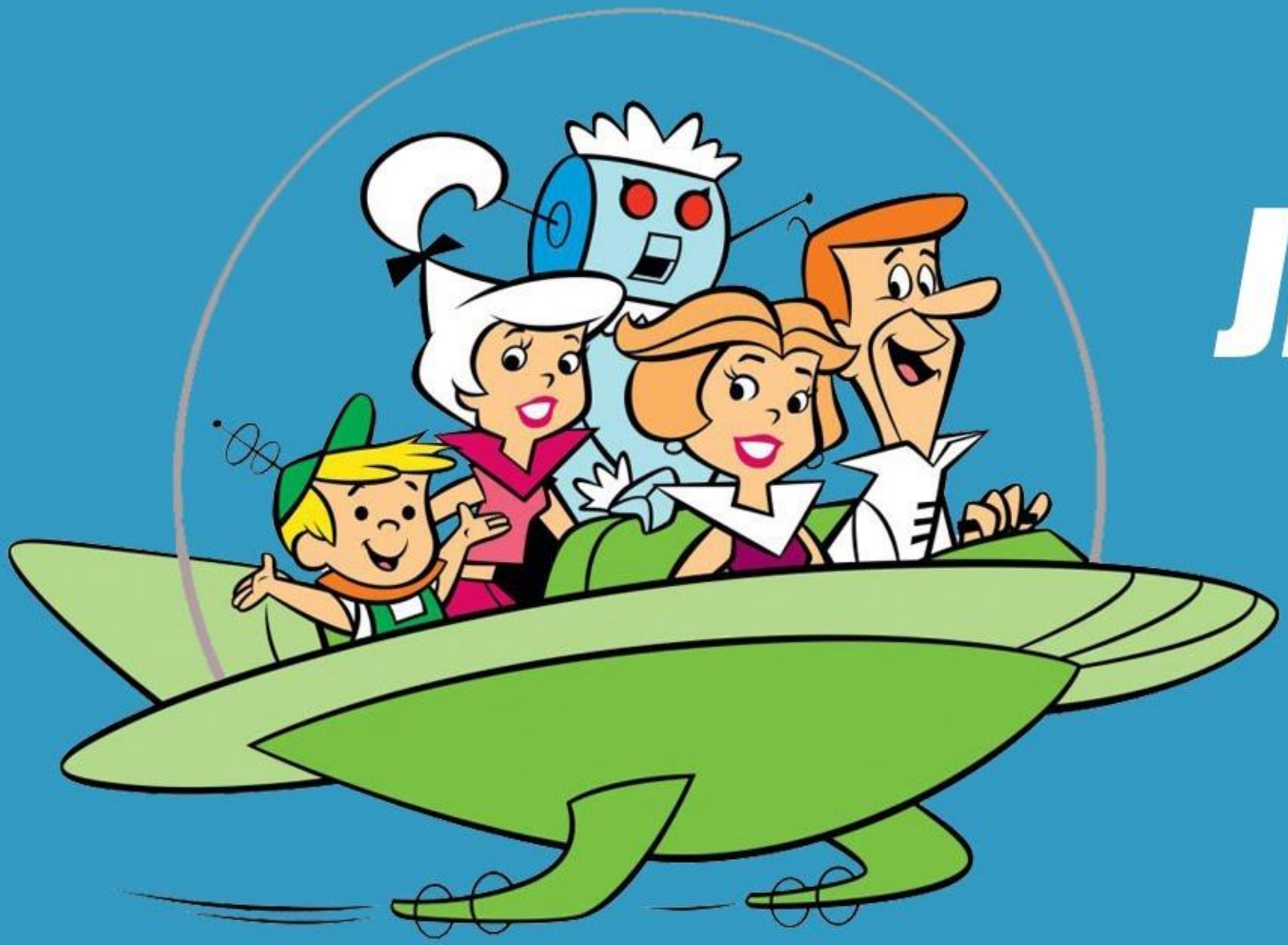
The Future of ETL

Gwen Shapira, Principal Data Architect
@gwenshap

https://www.confluent.io/blog/the-future-of-etl-isnt-what-it-used-to-be/



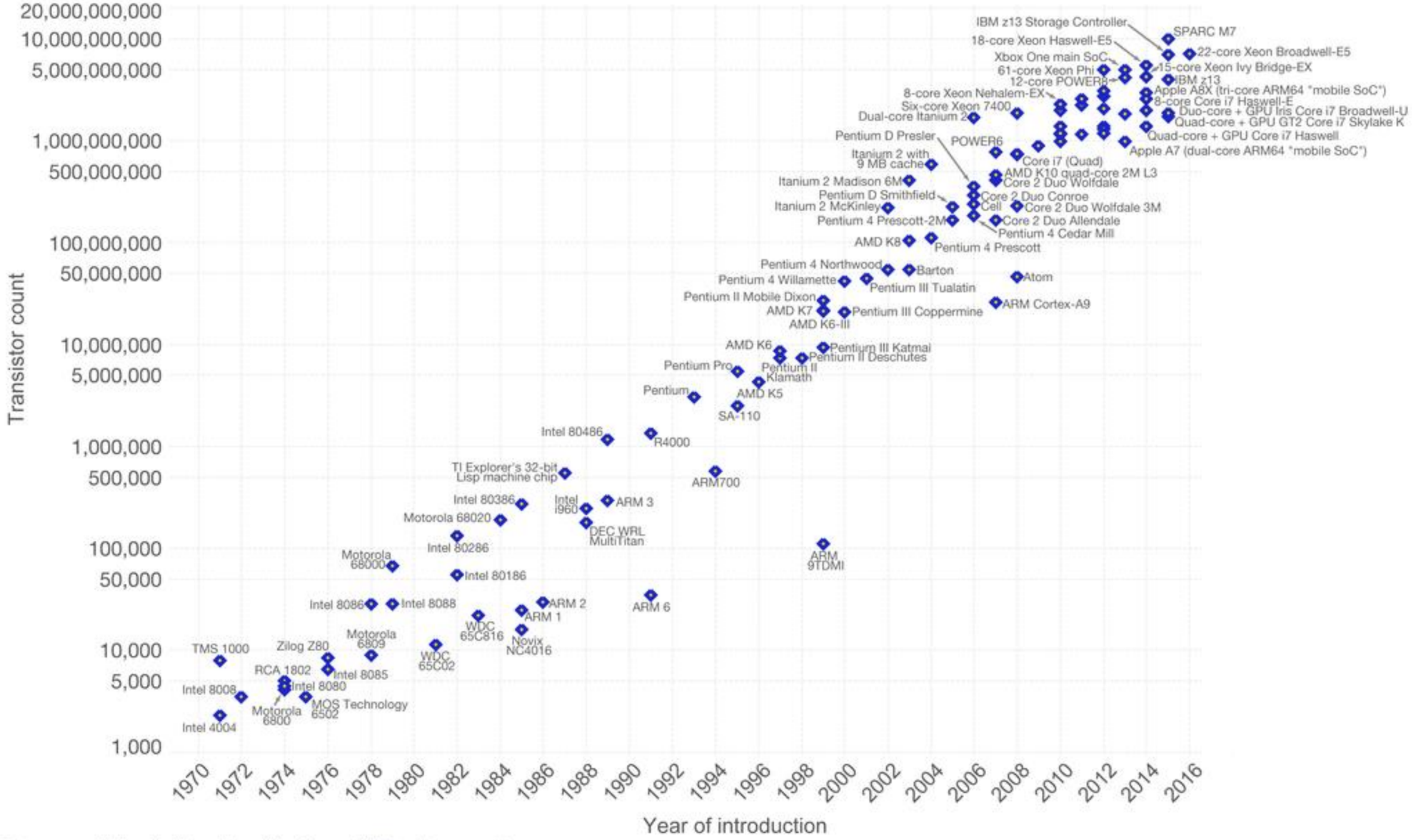
THE JETSONS

You extrapolate the trends that you see

Moore's Law – The number of transistors on integrated circuit chips (1971-2016)

Our World in Data

Moore's law describes the empirical regularity that the number of transistors on integrated circuits doubles approximately every two years. This advancement is important as other aspects of technological progress – such as processing speed or the price of electronic products – are strongly linked to Moore's law.



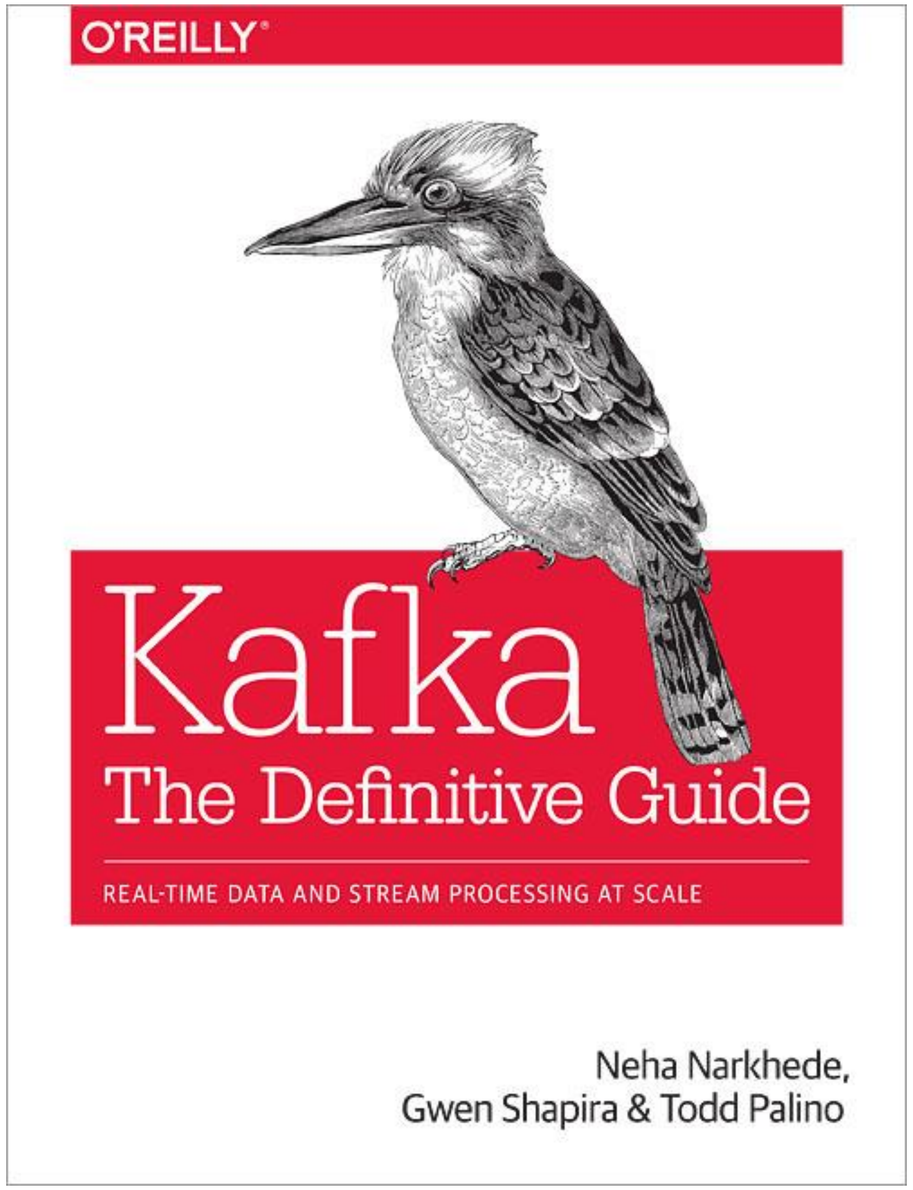
Data source: Wikipedia (https://en.wikipedia.org/wiki/Transistor_count)

The data visualization is available at OurWorldinData.org. There you find more visualizations and research on this topic.

So, to think about future of ETL, we want to look at…

- The future we imagined in the past
- The trends we are seeing now
- The future these trends indicate
- Practical suggestions
- Final thoughts

Me



- Moving data around for 20 years
- Works at Confluent.
- Apache Kafka Committer
- Wrote a book or two
- Tweets a lot

What was it like 15 years ago?

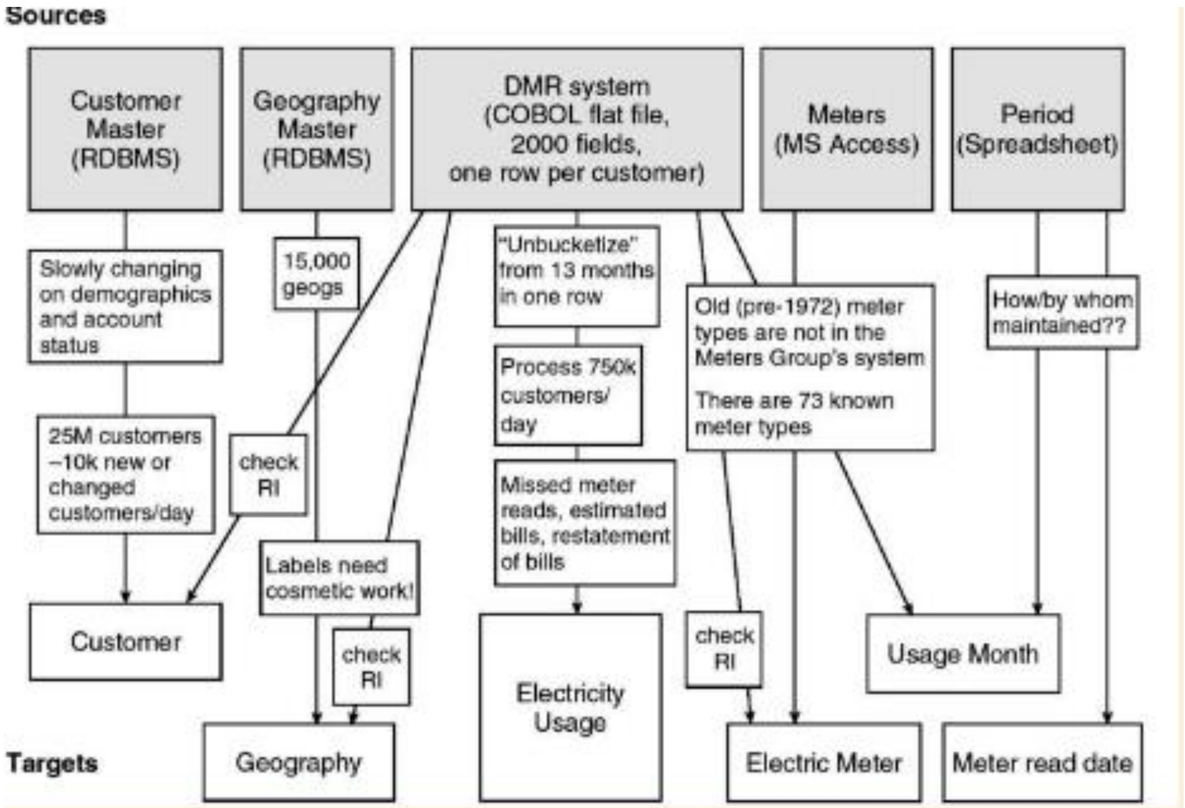


Figure 10-1 Example high level data staging plan schematic.

Very few sources

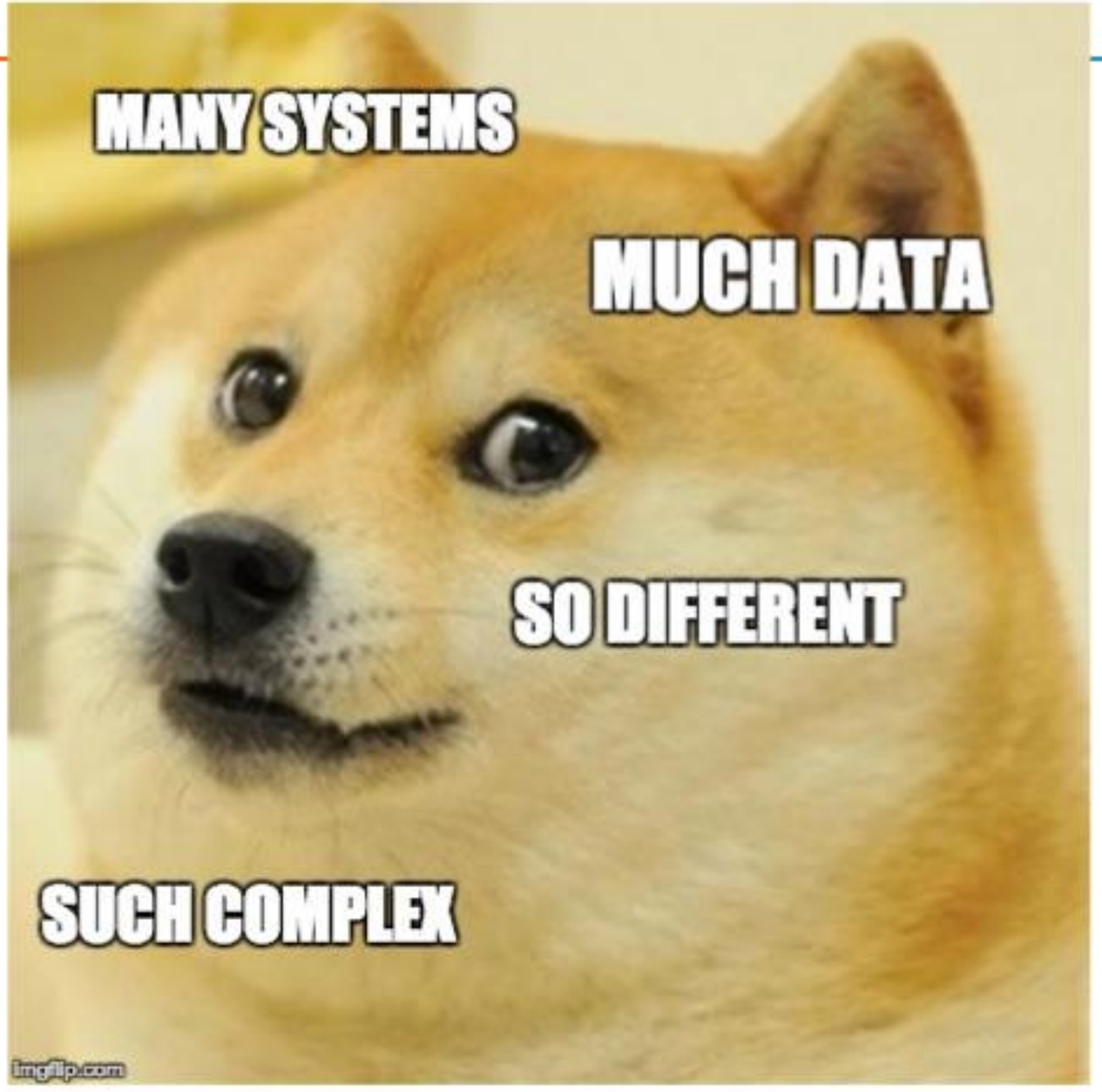
One Data Warehouse to rule them all

One big “friendly” ETL tool

Significant effort modeling, cleaning and converging.

“Although it is a huge leap to move from a monthly or weekly process to a nightly one…”

Main pain-points of data integration?



What did we dream about then?

- Data modeling is painful.

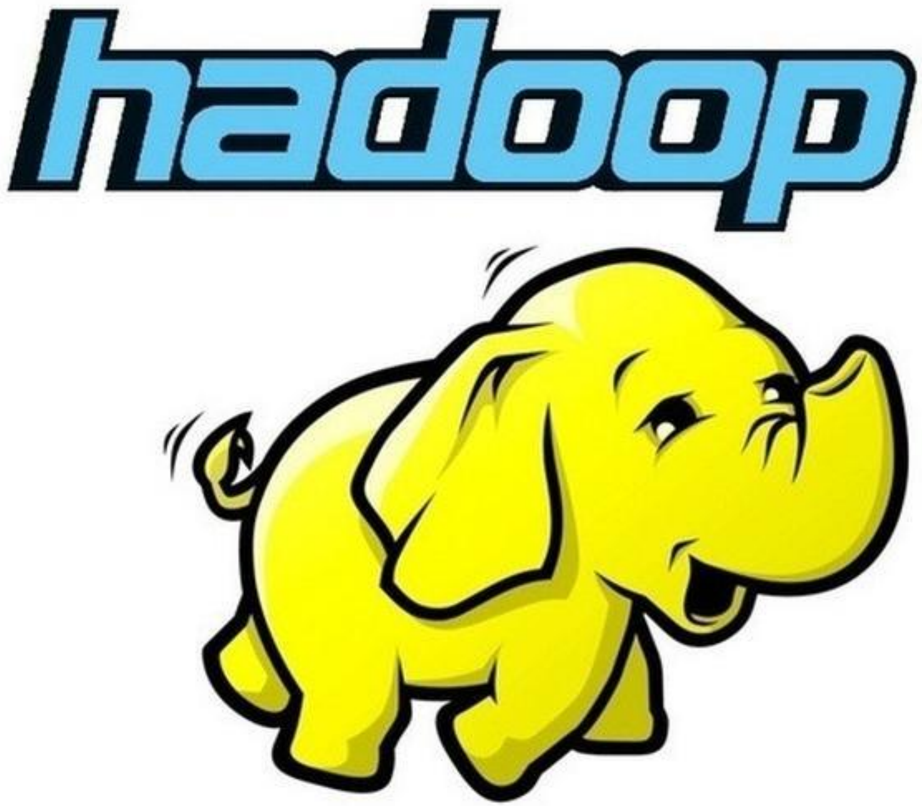
  Is there a way around that?
- We are always missing data.

  Can we handle more diverse data sources?
- Scale is painful. Our daily run takes 25 hours.

  Can we do it in 5?
- Data Warehouses are expensive.

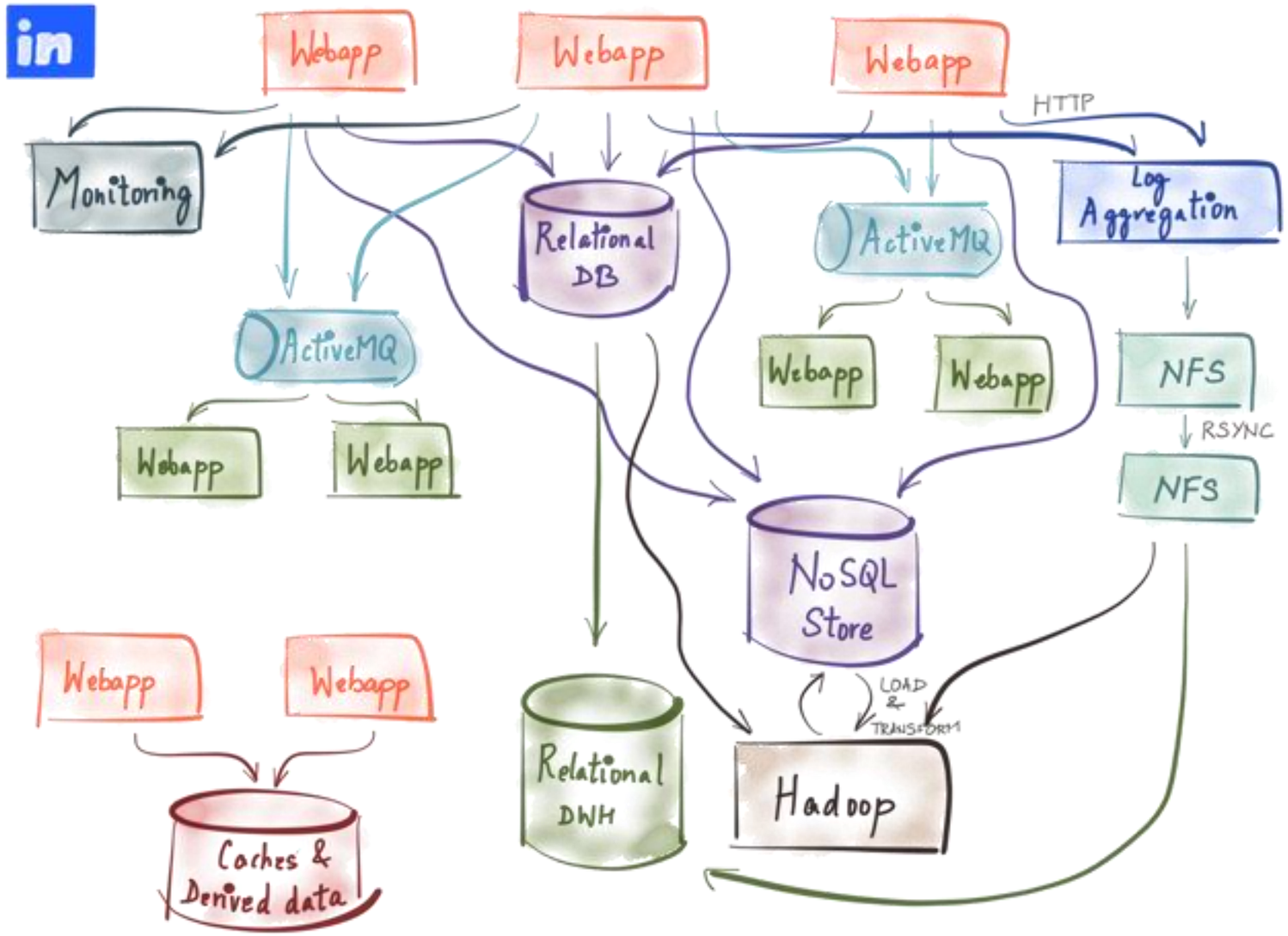
  If only there was a free option.

And 5 years ago?



- Many diverse sources.
- One big target
- Complete confusion of ETL tools
- “Schema on Read”
- … and still loading data around once a day.

And since change is difficult, we also got this:



What did we dream about then?

- **Real-time.**

  We don’t want to wait even 5 minutes for insights.

- **Governance.**

  Save us from the mess we created.

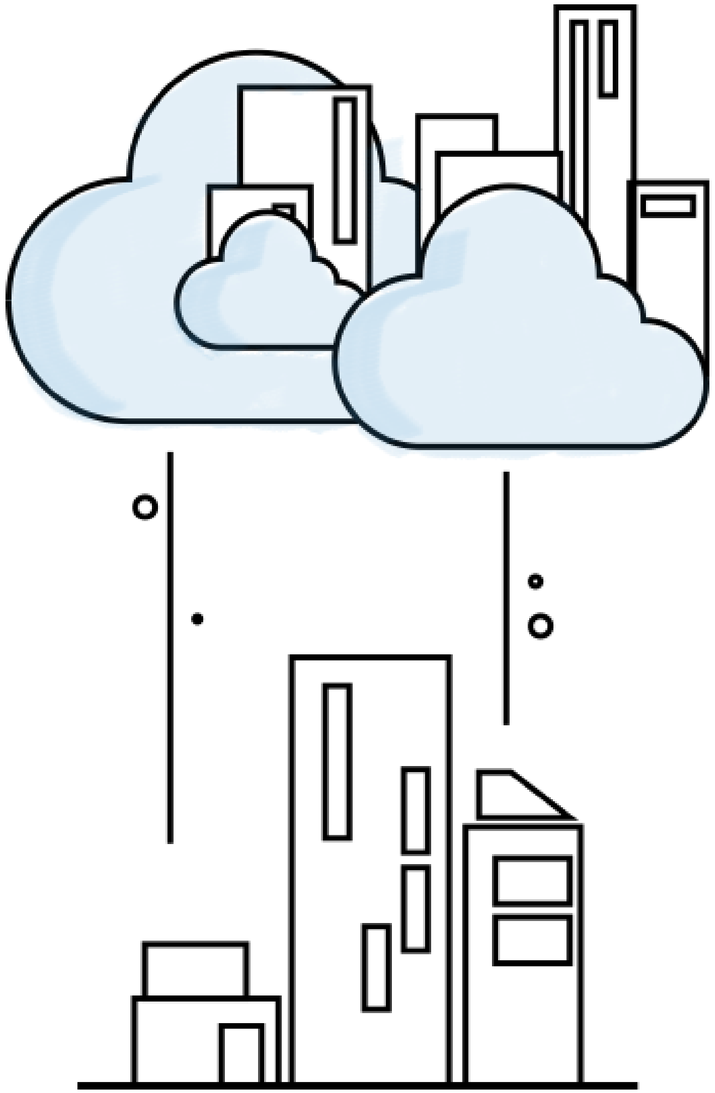
- **Better tooling.**

  We imagined really fancy drag-and-drop.

But we missed some key ways in which the world is changing

#1 Cloud

We want ETL that:

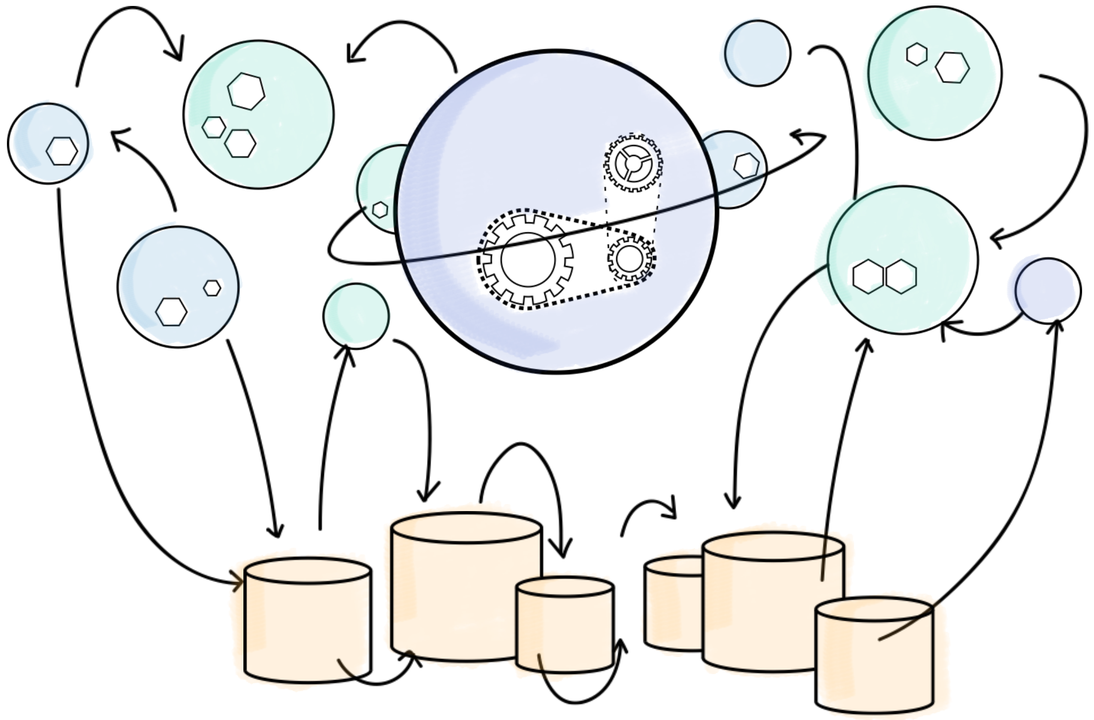


- Integrates with Managed Services
- Seamlessly integrate across many DCs
- Cloud-native.

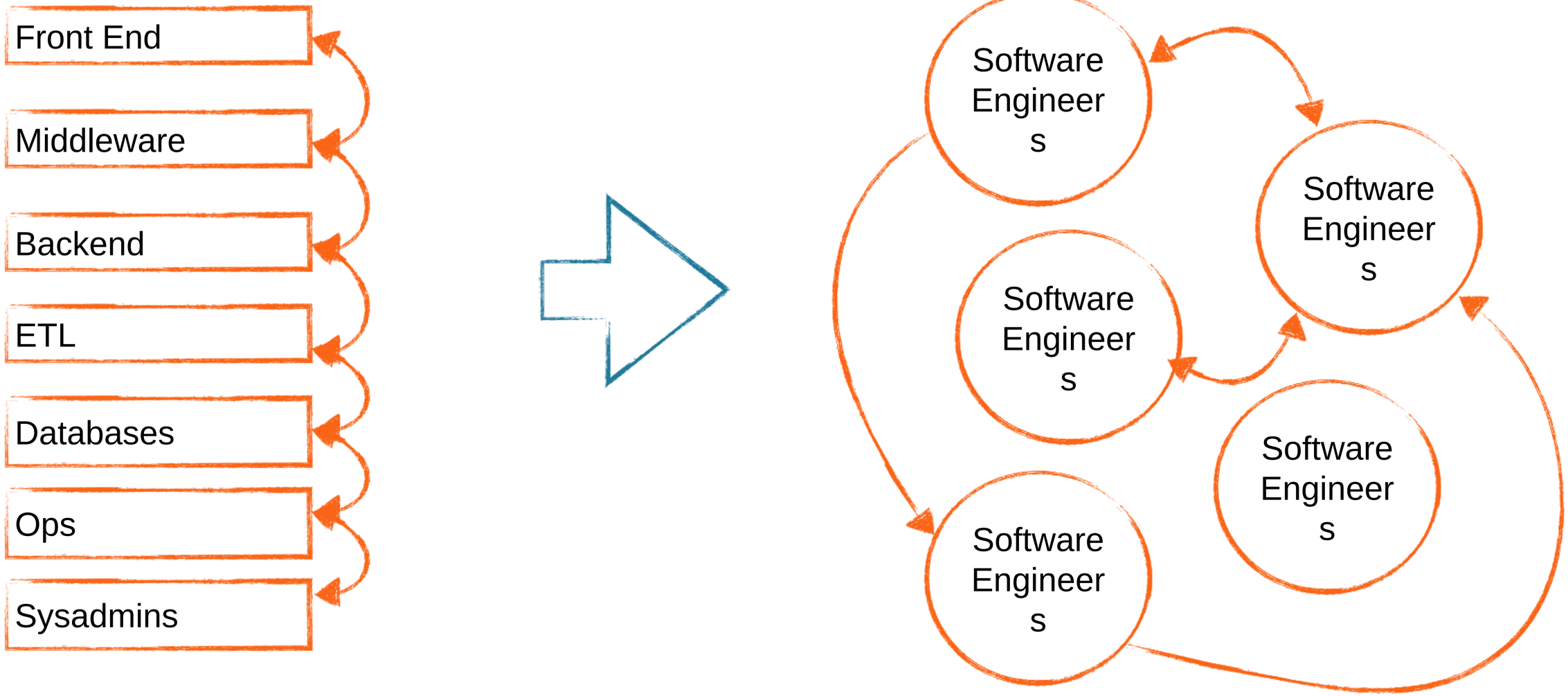
  Containerized, dynamic scale, automatic recovery

#2 Microservices

We want ETL that:



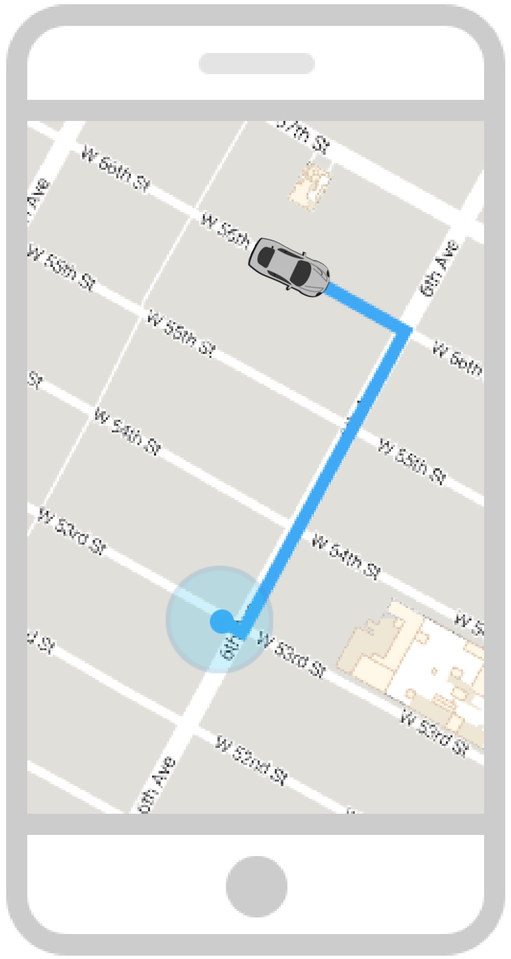
- Integrates applications and data stores
- Platform for building data-intensive apps
- Agile and Composable. DRY.



Cloud and Microservices drove more changes

#3 Software Engineers do Everything

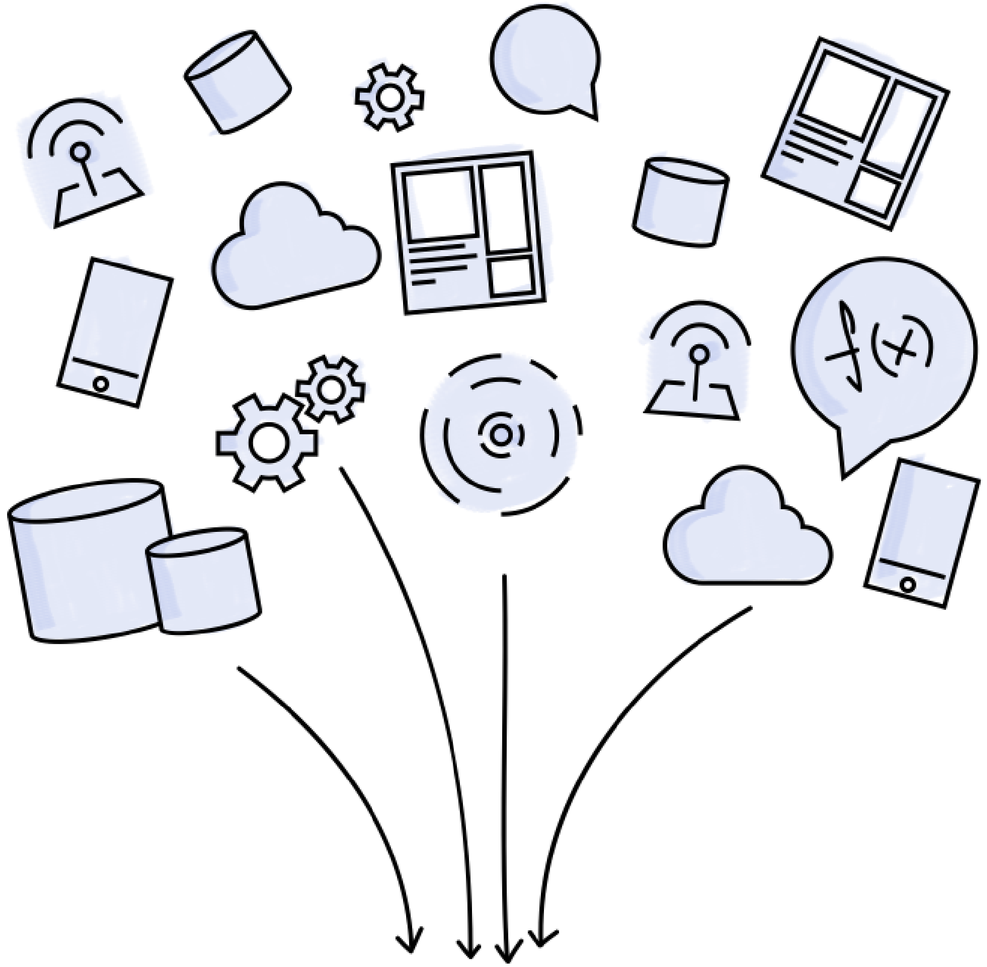
We want ETL that:



- Open Source
- Integrates with engineering tools
- Low / No inter-team dependencies
- APIs
- Source Control, CI/CD, Test frameworks
- Part of the application

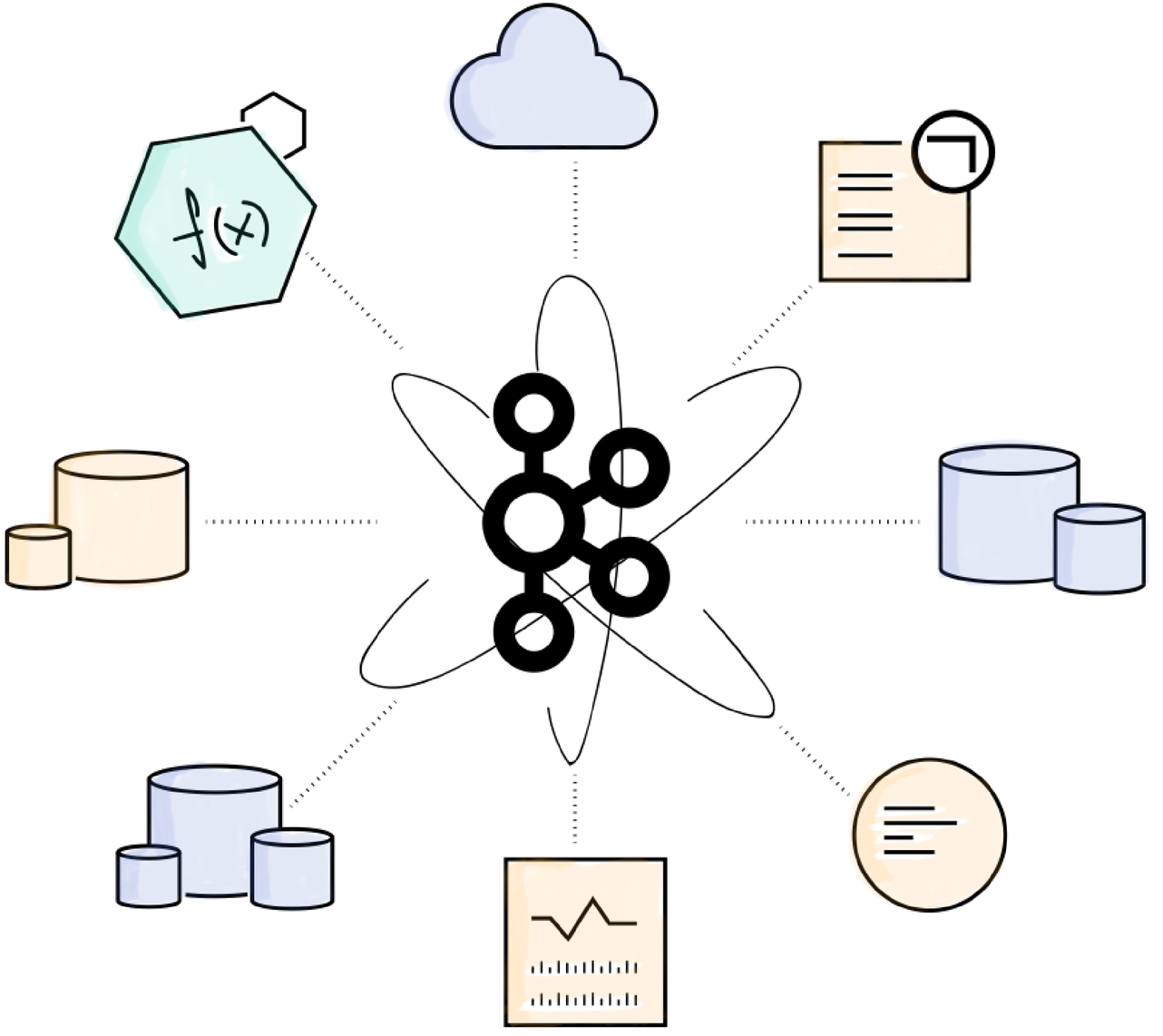
What shall we build now?

#1 Model Everything as a Stream

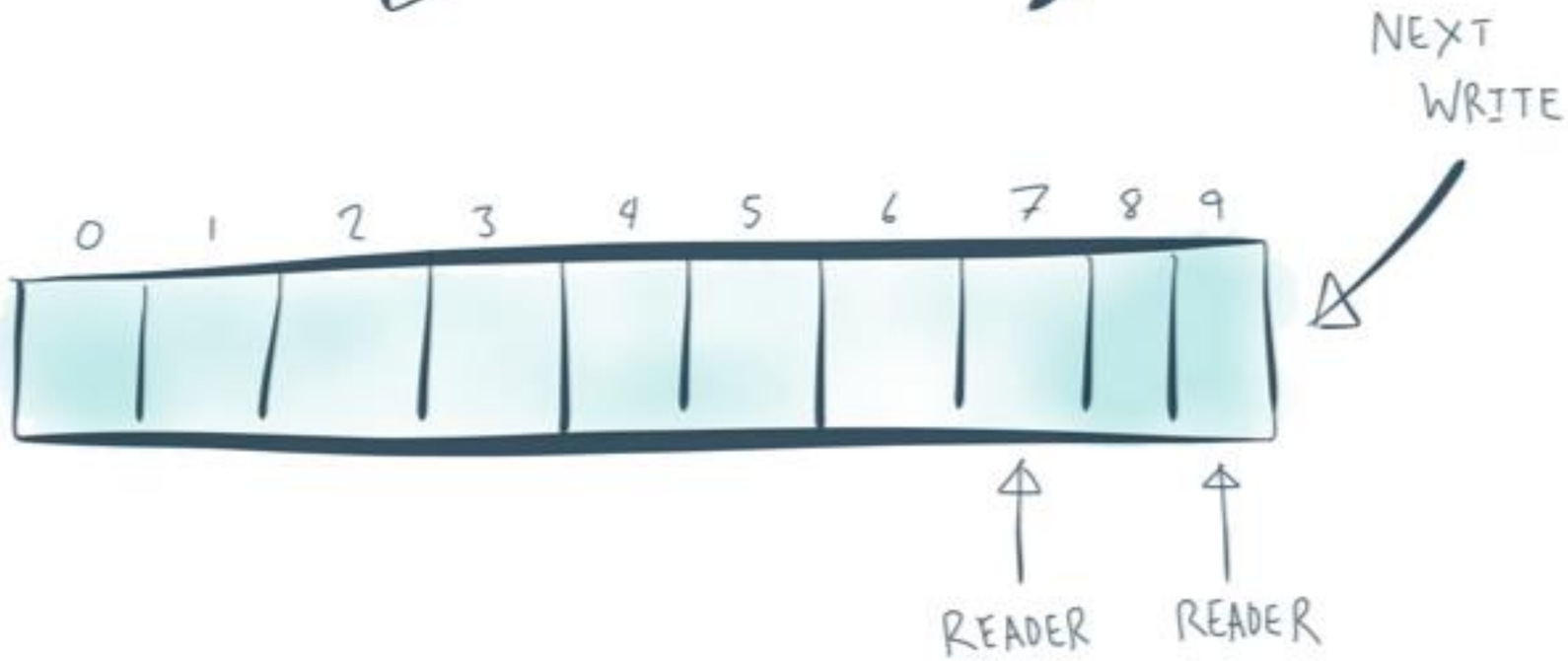


Imagine…

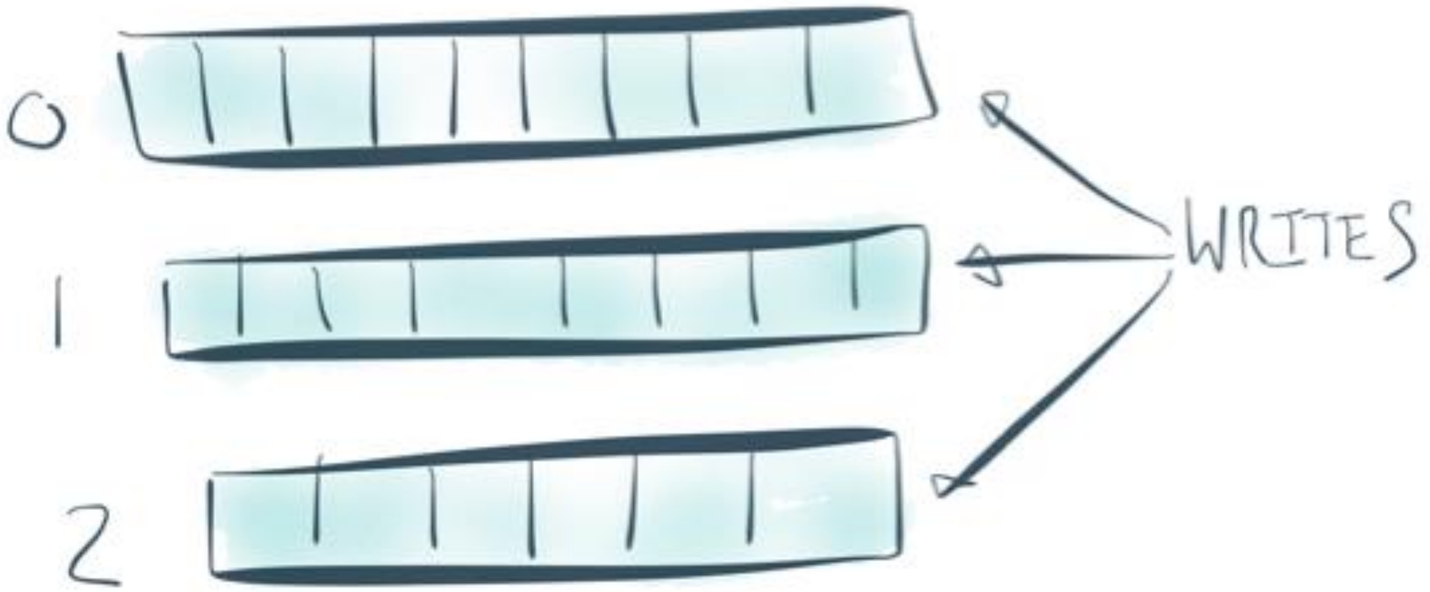
all your data as a stream


f(x)

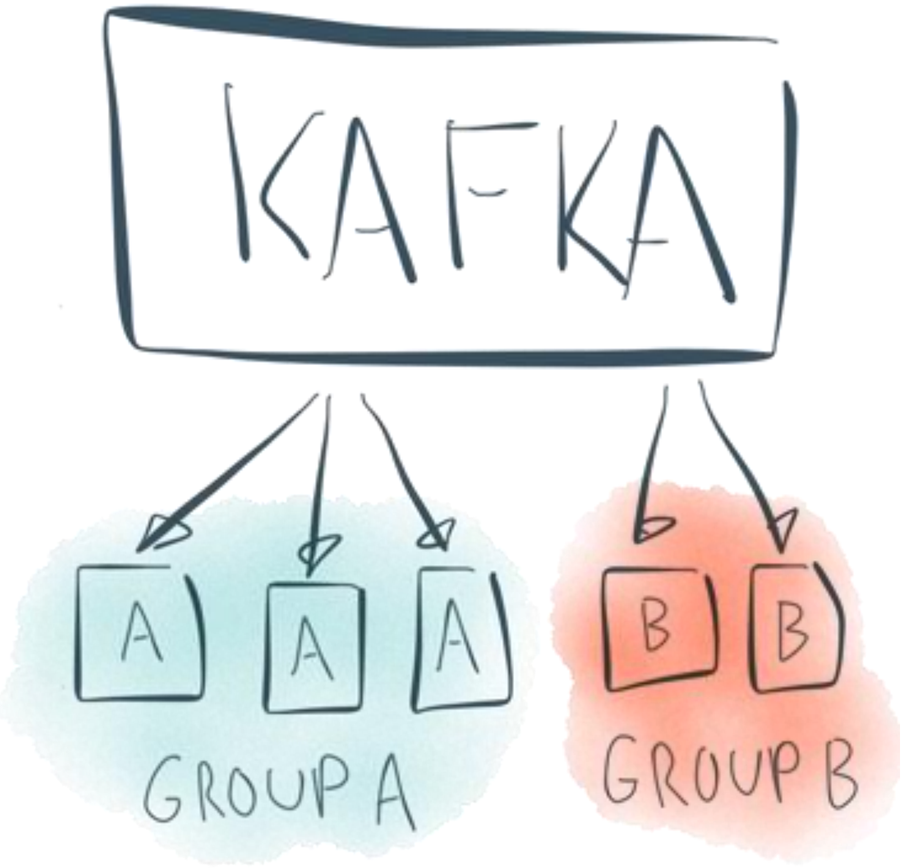
LOGS

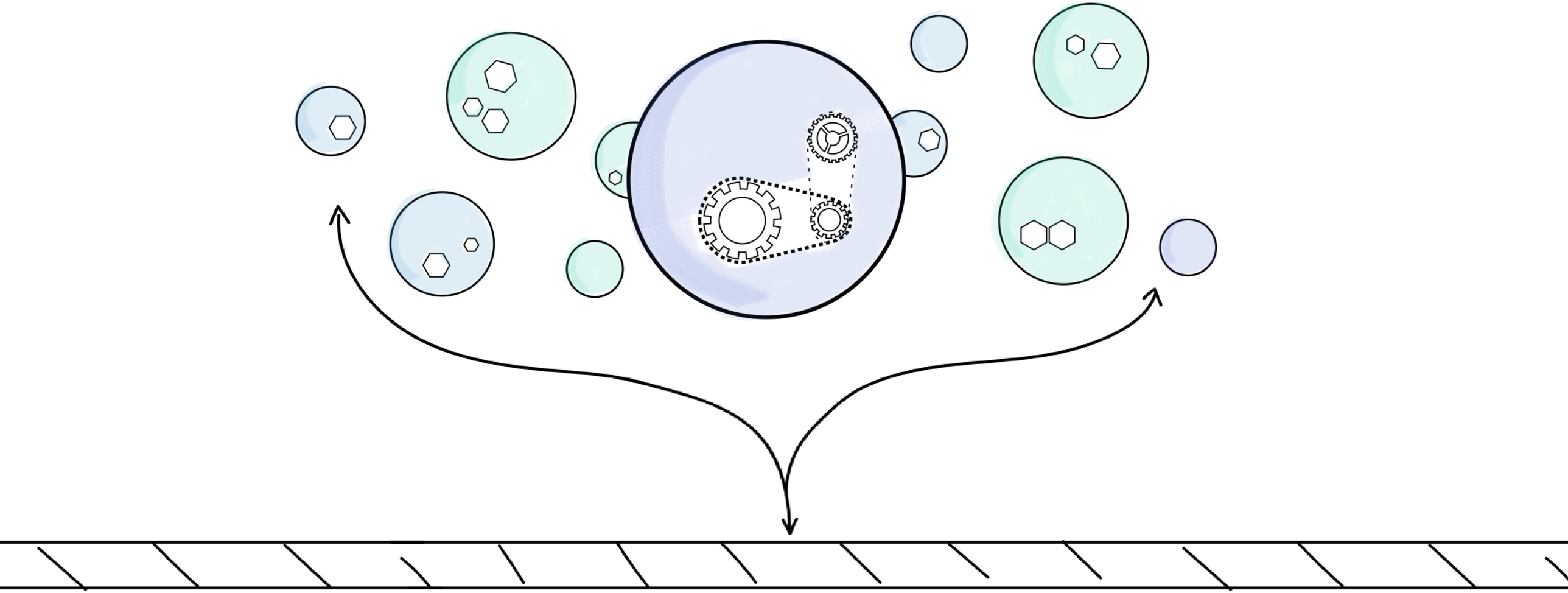


KAFKA TOPIC = PARTITIONED LOG



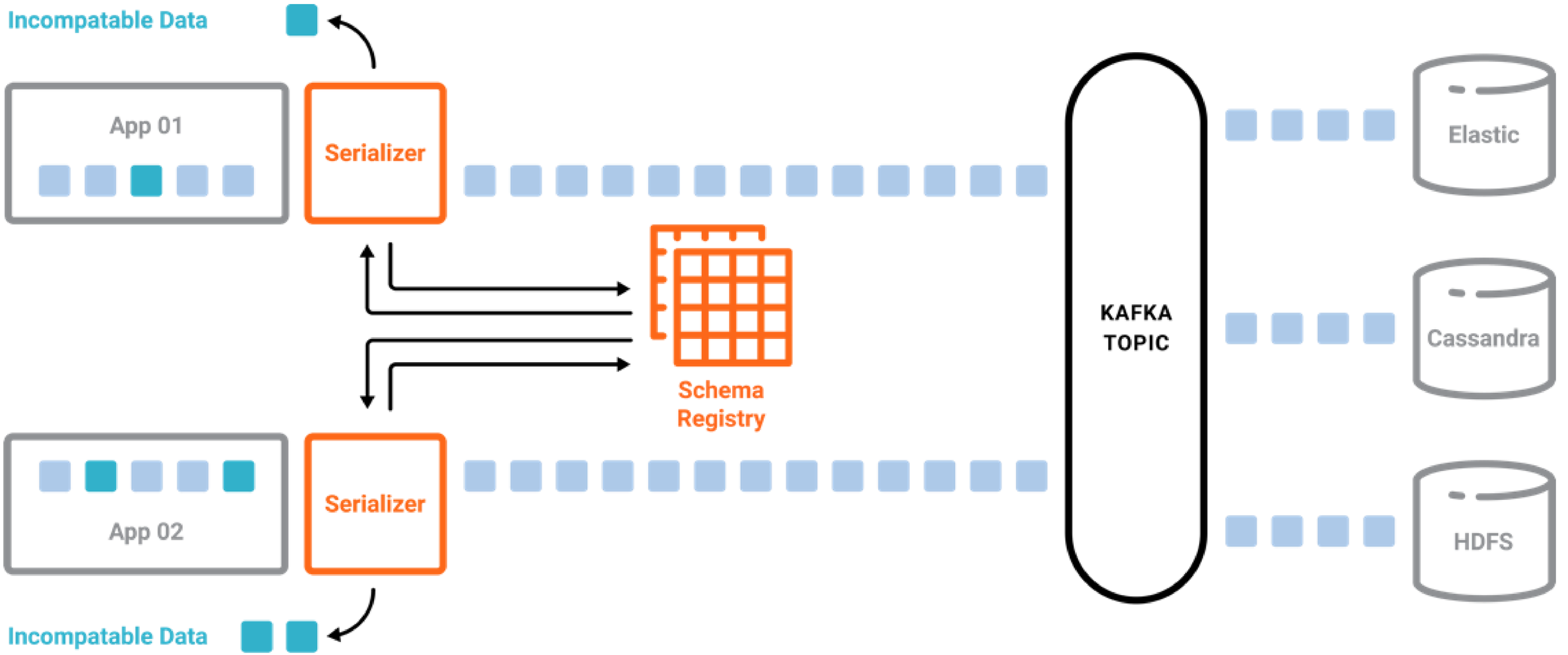
CONSUMERS, GROUPS, PARTITIONING, AND ALL THAT

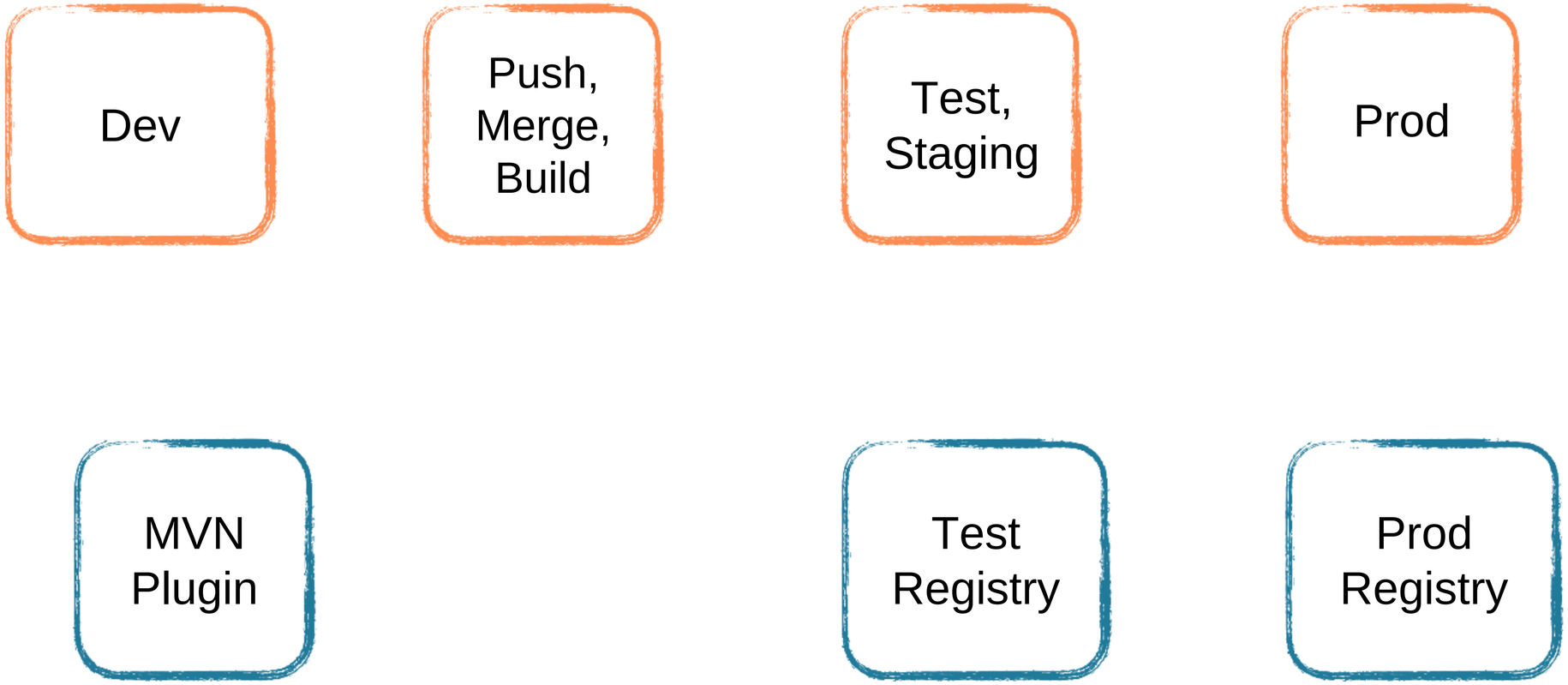




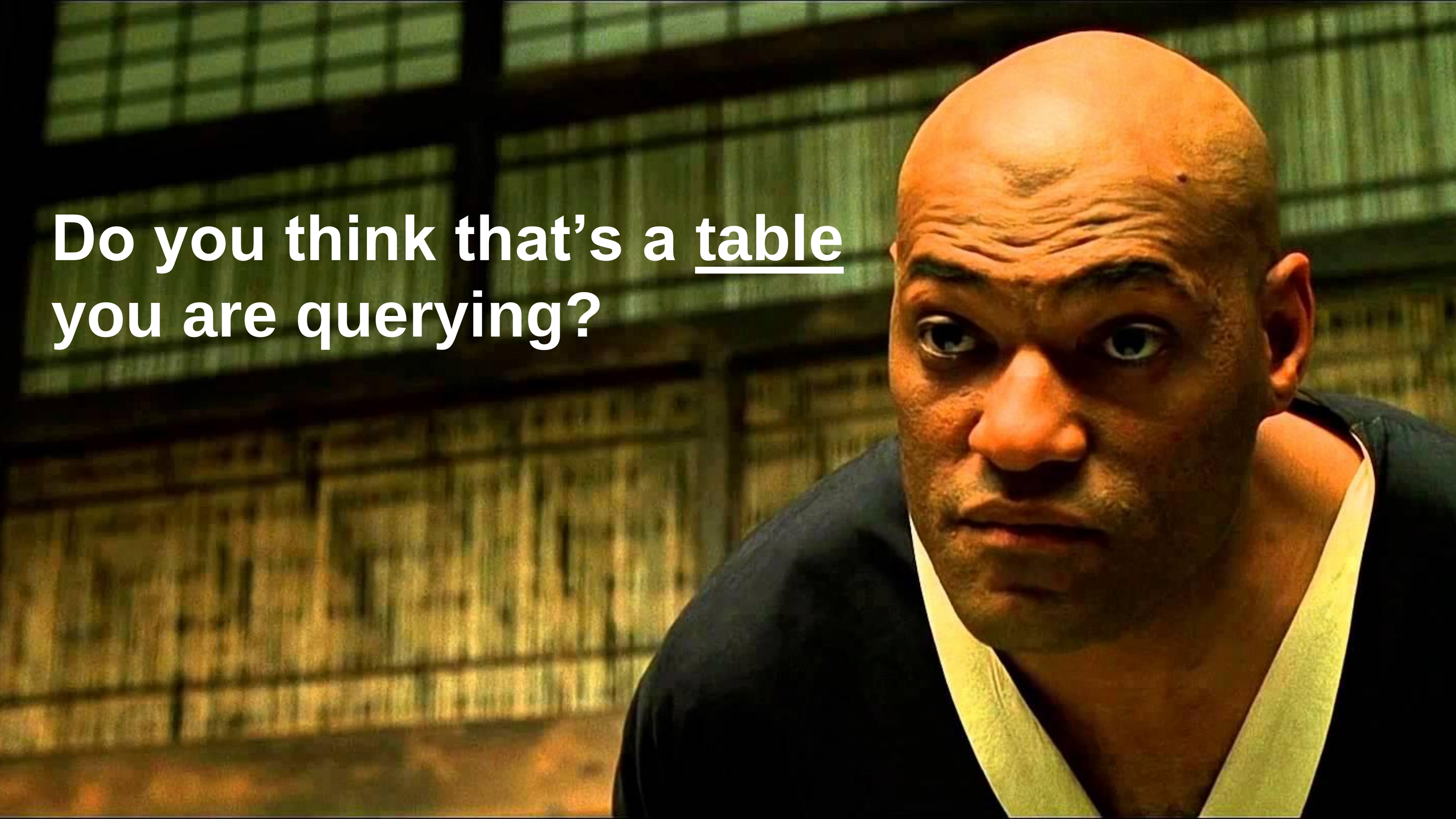
A stream fixes the madness

#2 Move Fast, Don’t Break Compatibility


Incompatable Data
App 01
Serializer
Schema
Registry
KAFKA
TOPIC
Elastic
Cassandra
HDFS
App 02
Serializer
Incompatable Data


Dev
Push, Merge, Build
Test, Staging
Prod
MVN Plugin
Test Registry
Prod Registry

#3 Integrate Databases and Applications at Scale


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

The Stream/Table Duality

Stream

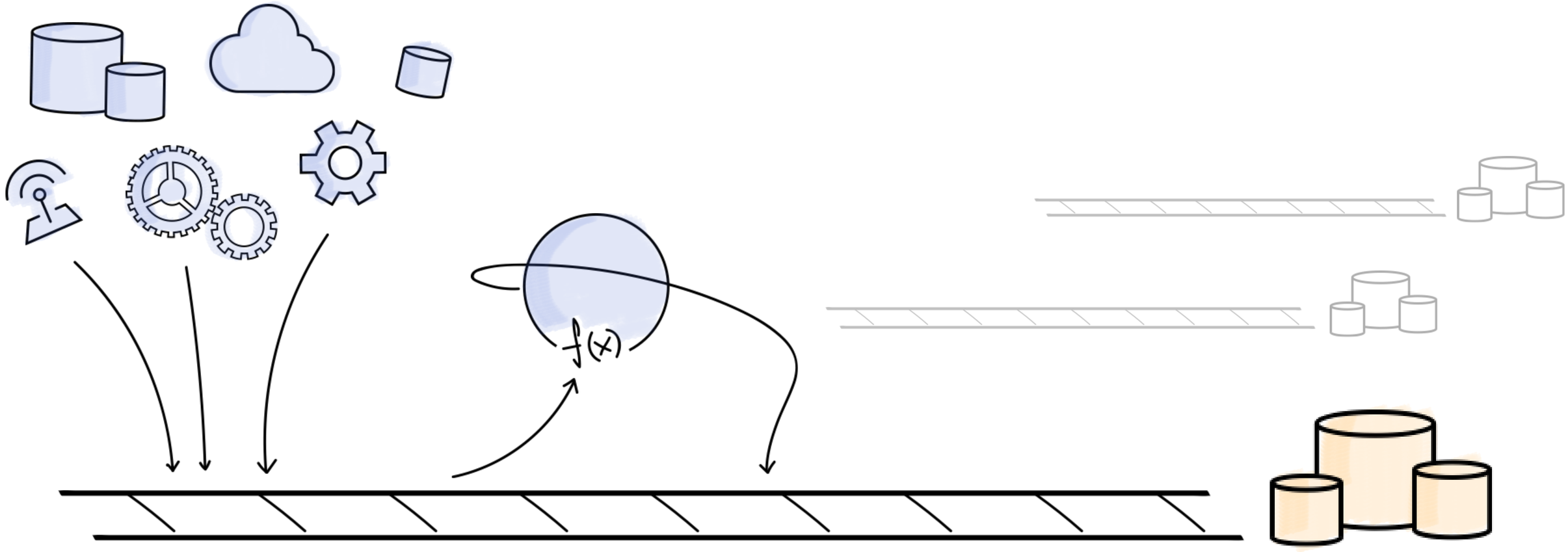
Time

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

Table

Account ID	Balance
Account ID	Balance
Account ID	Balance
12345	€15

http://docs.confluent.io/current/streams/concepts.html#duality-of-streams-and-tables



Full streaming platform with Connect + Streams

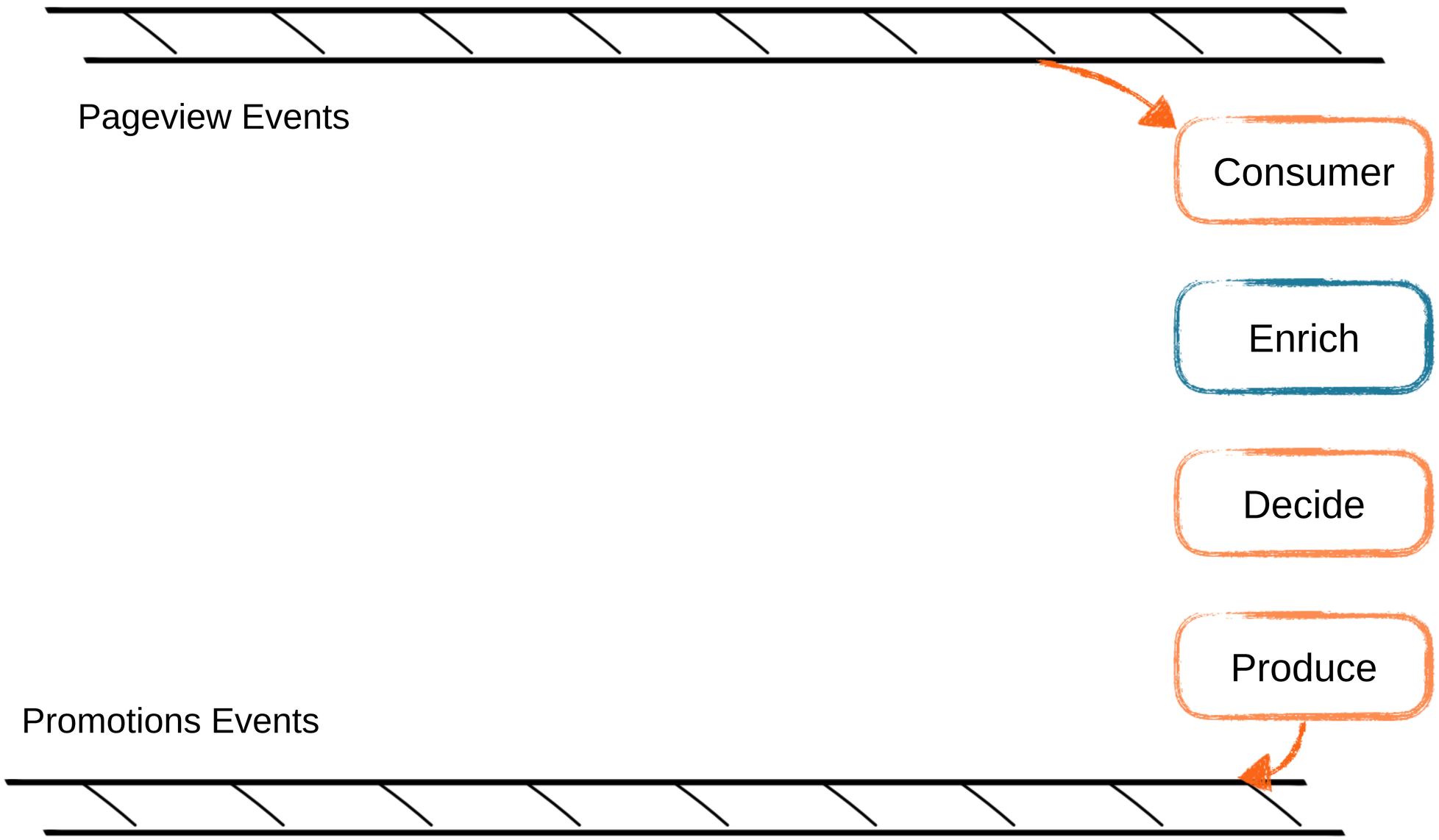
Lets look at an Example. Our marketing team has a small request:

We want to send Platinum members

Who are looking at beach properties

An email about

Discount package in new Florida hotel


Pageview Events
Consumer
Enrich
Decide
Produce
Promotions Events

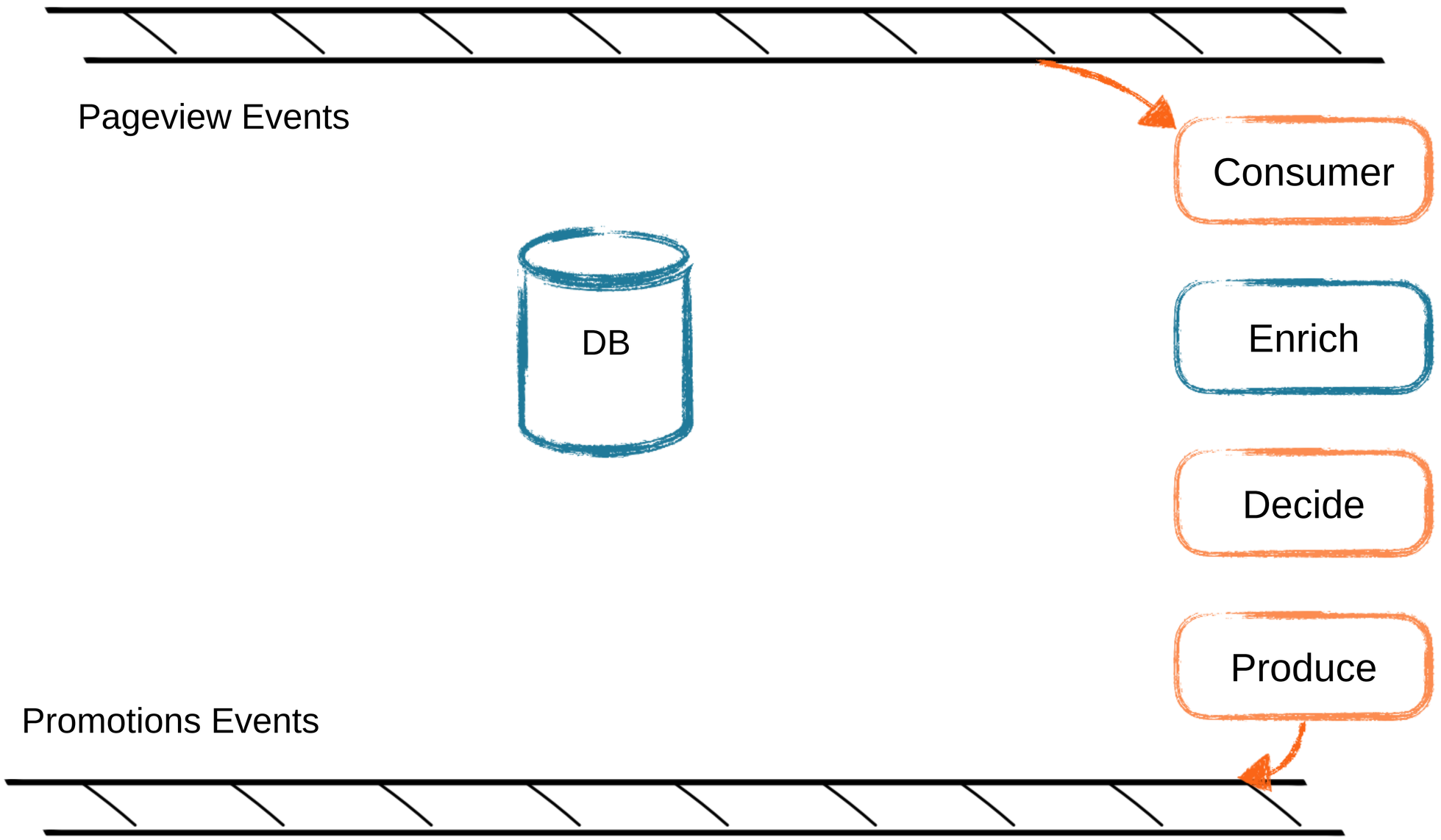
How to draw an owl

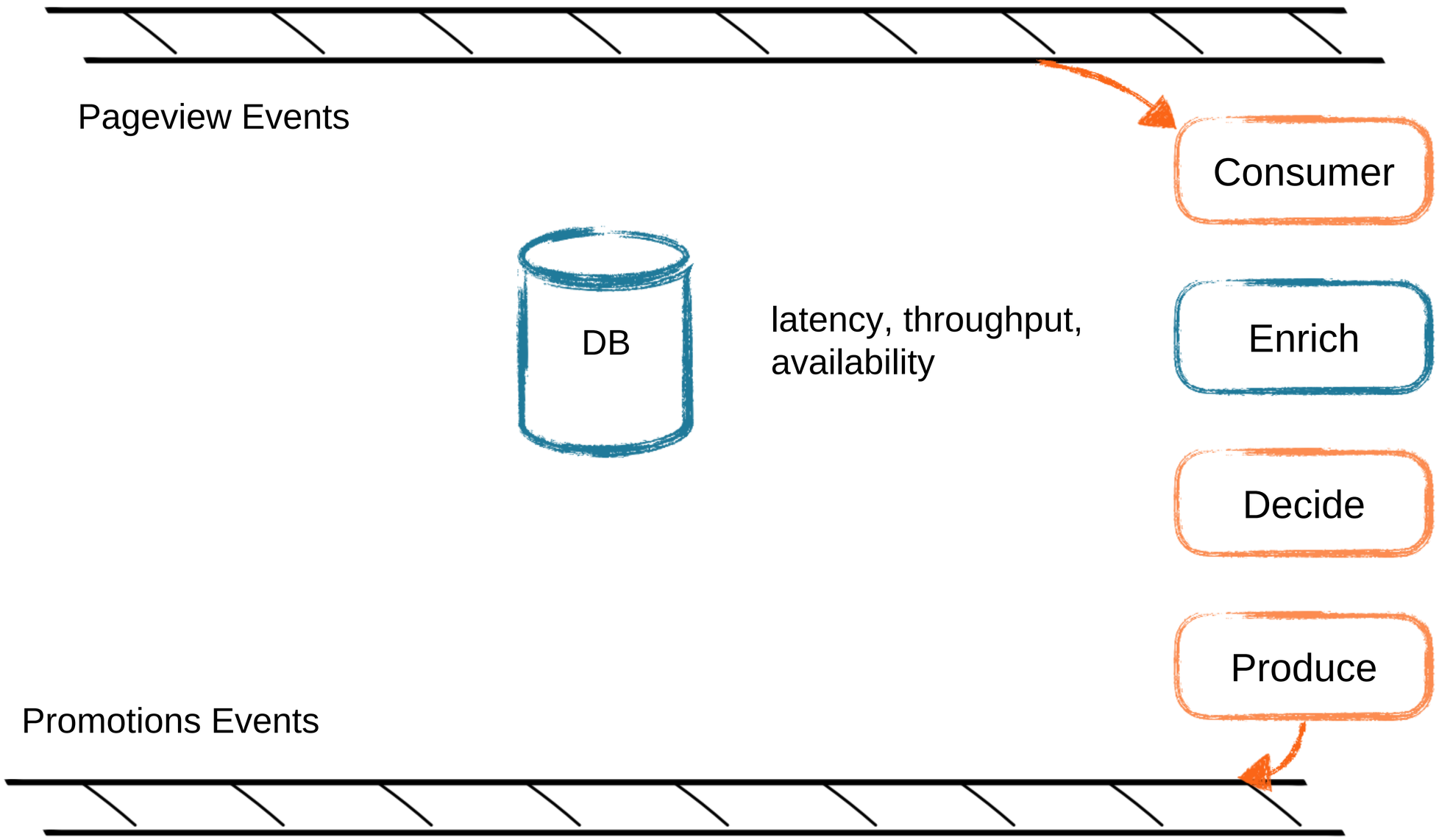
1.

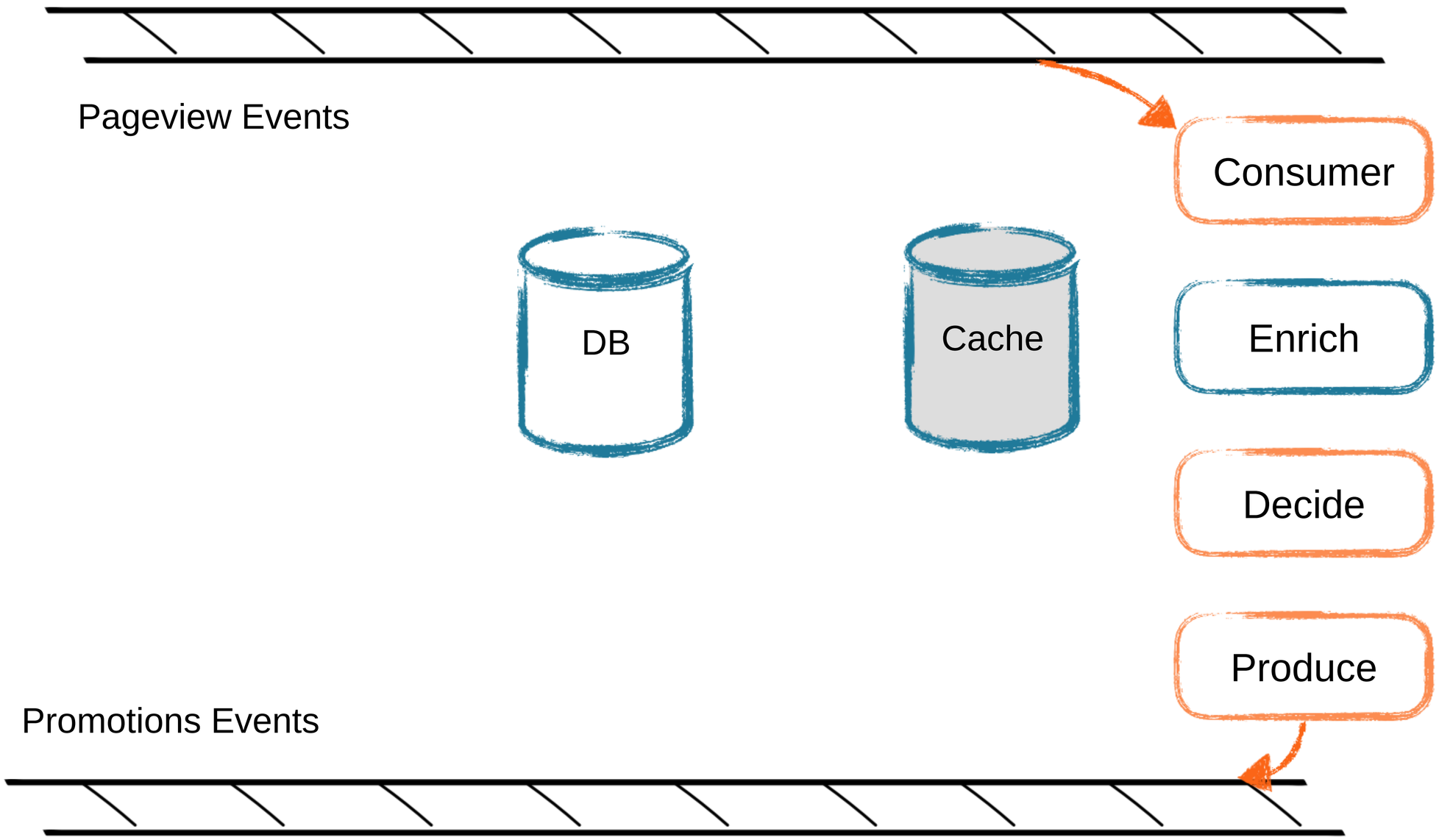
1. Draw some circles

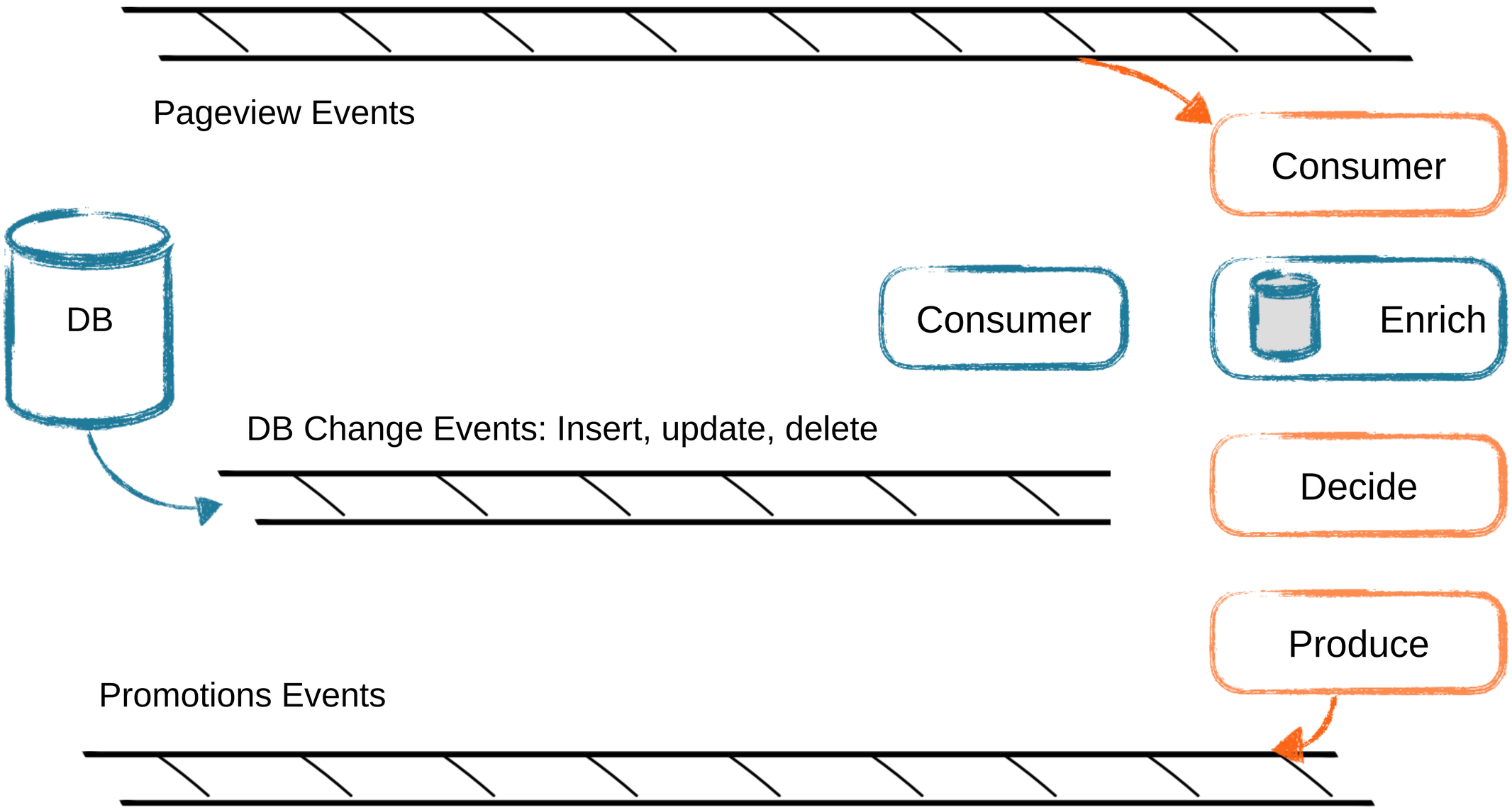
2.

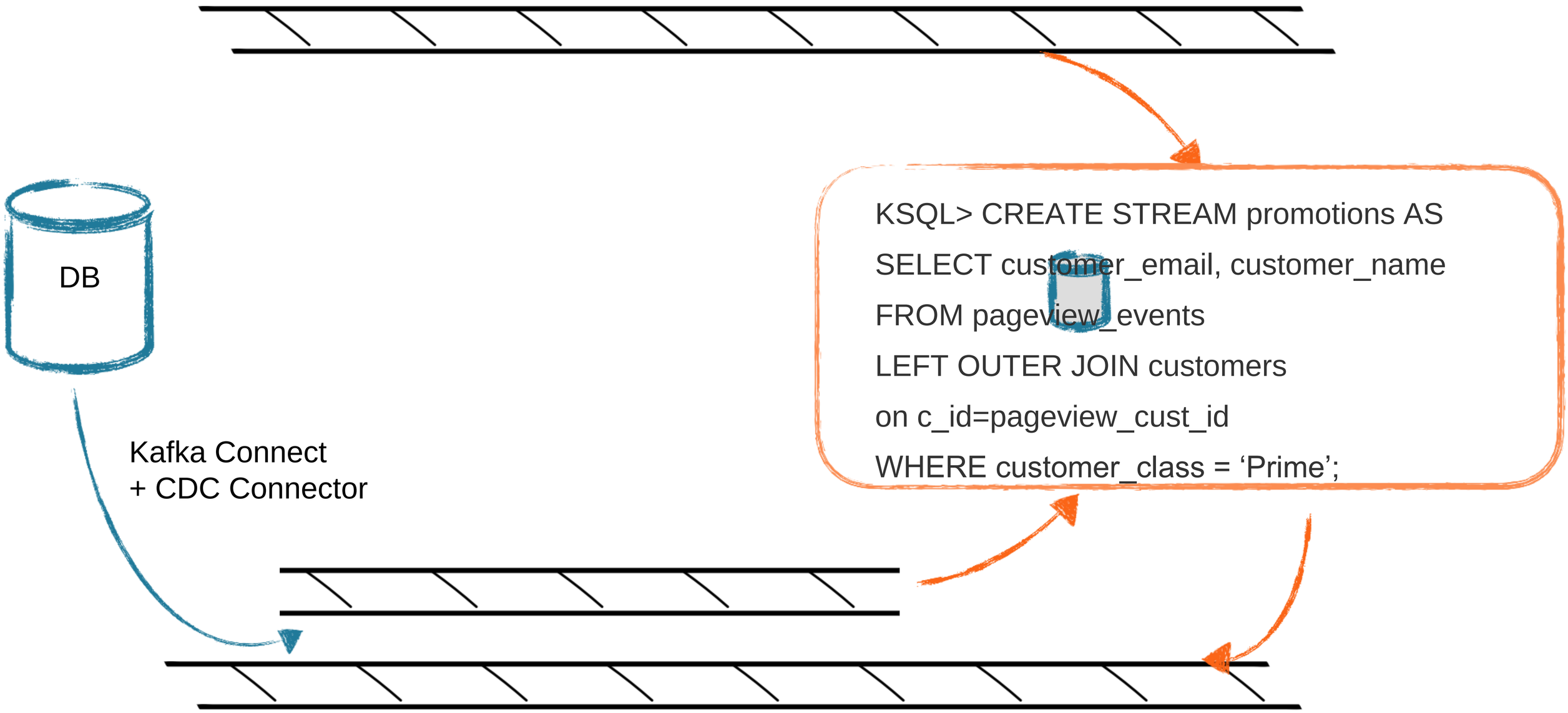
2. Draw the rest of the fucking owl


Pageview Events
Consumer
DB
Enrich
Decide
Produce
Promotions Events

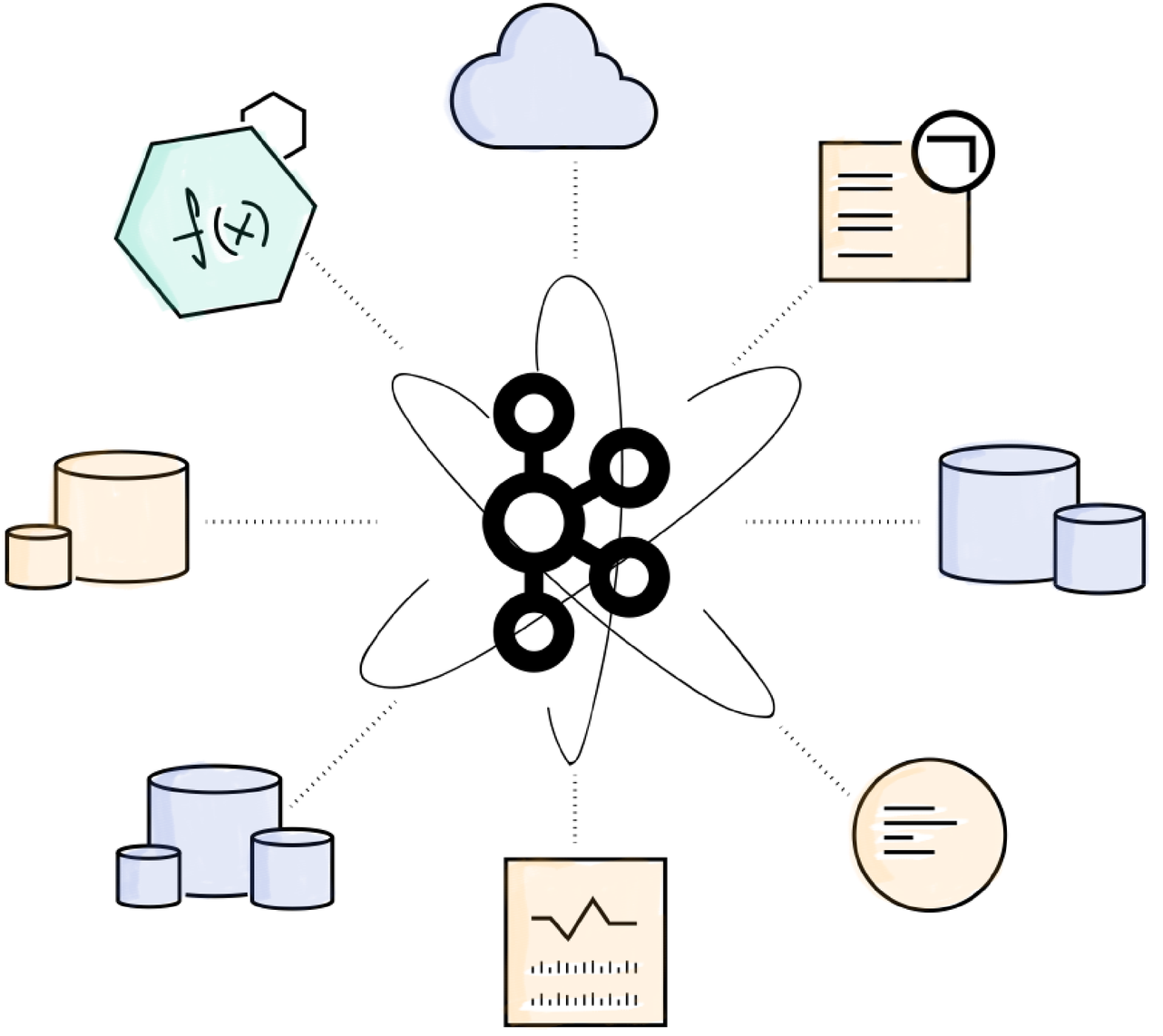

Pageview Events
Consumer
DB
latency, throughput,
availability
Enrich
Decide
Produce
Promotions Events


Pageview Events
Consumer
DB
Cache
Enrich
Decide
Produce
Promotions Events

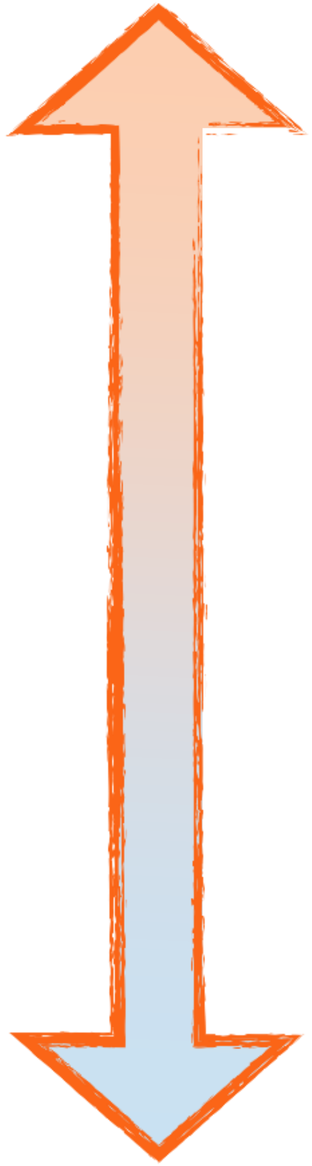

Pageview Events
Consumer
DB
Consumer
Enrich
DB Change Events: Insert, update, delete
Decide
Produce
Promotions Events


DB
Kafka Connect
+ CDC Connector
KSQL> CREATE STREAM promotions AS
SELECT customer_email, customer_name
FROM pageview_events
LEFT OUTER JOIN customers
on c_id=pageview_cust_id
WHERE customer_class = ‘Prime’;

What’s next?


f(x)

- Serverless
- Observability
- Operational Patterns
- Metadata for Engineers
- Stronger guarantees
- Standardization
- Return of ETL basics
- Metadata-driven, ML-driven ETL
- …
- …
- Organizational Enlightenment



Almost here?

Will this ever happen?

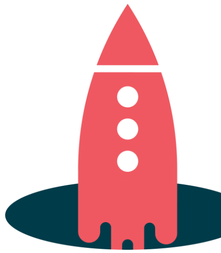
Is ETL dead?

Resources and Next Steps

https://github.com/confluentinc/ksql
https://github.com/confluentinc/cp-demo

https://www.confluent.io/download/
https://www.confluent.io/blog

https://slackpass.io/confluentcommunity



confluent

Questions?

@gwenshap
gwen@confluent.io