

# Как построить платформу для стриминга данных на Kafka?

Андрей Серебрянский

Streaming Platform Owner в Raiffeisen Bank



# А кто я?



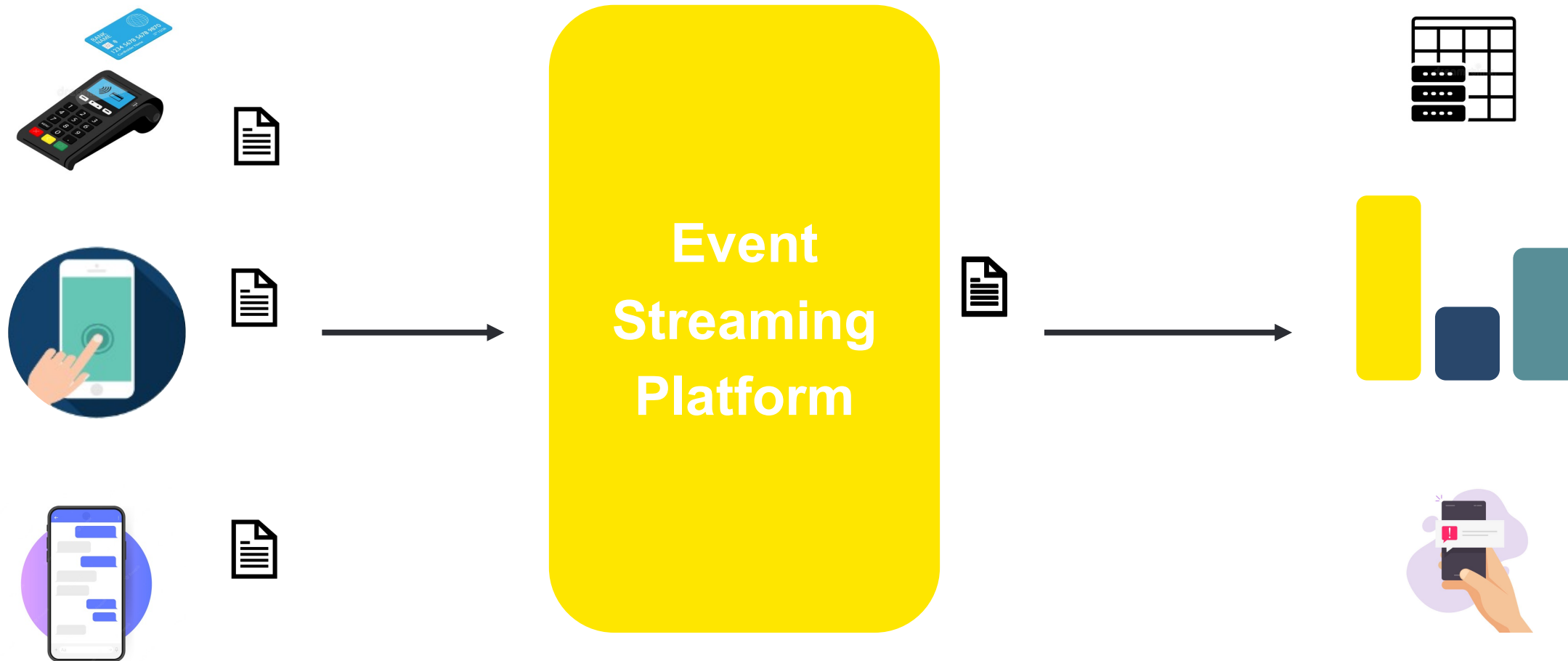
**Андрей  
Серебрянский**

Streaming Platform Owner в Raiffeisen  
Bank

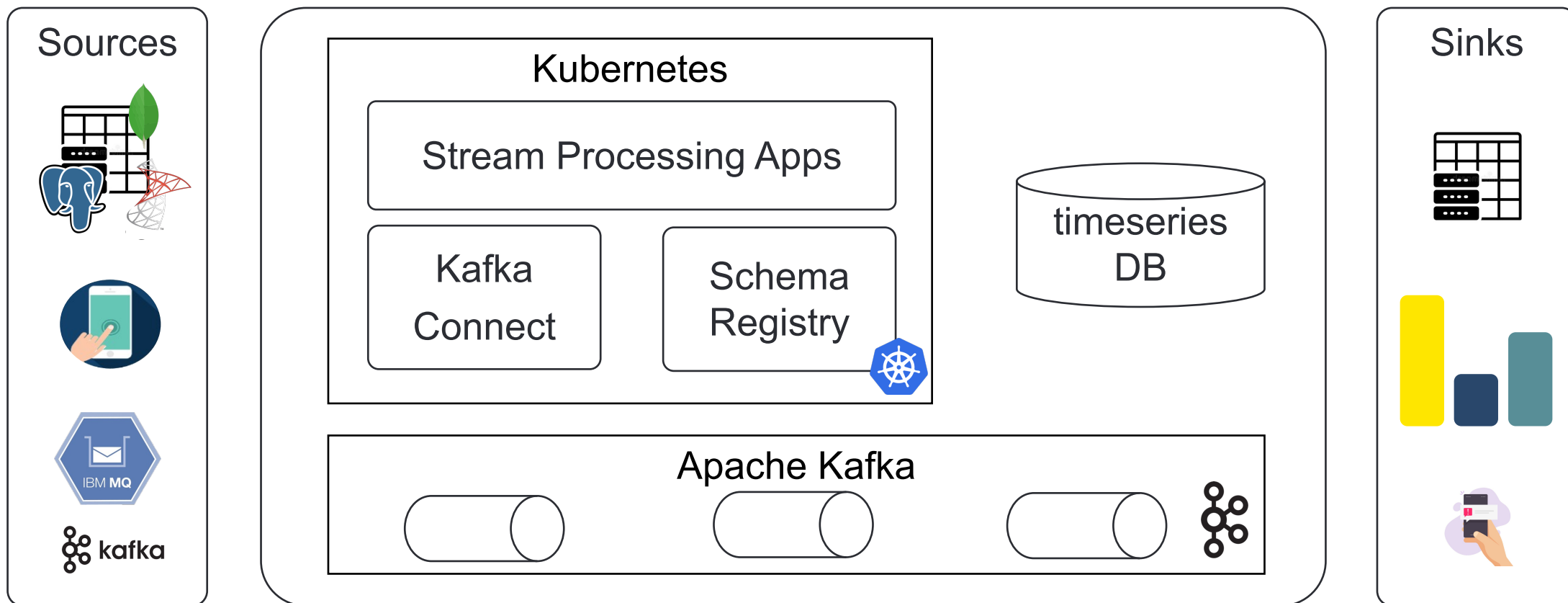
- спикер Kafka Summit, JPoint
- последние 3,5 года строю стриминговые платформы

# Зачем нужна стриминговая платформа?

Пара примеров

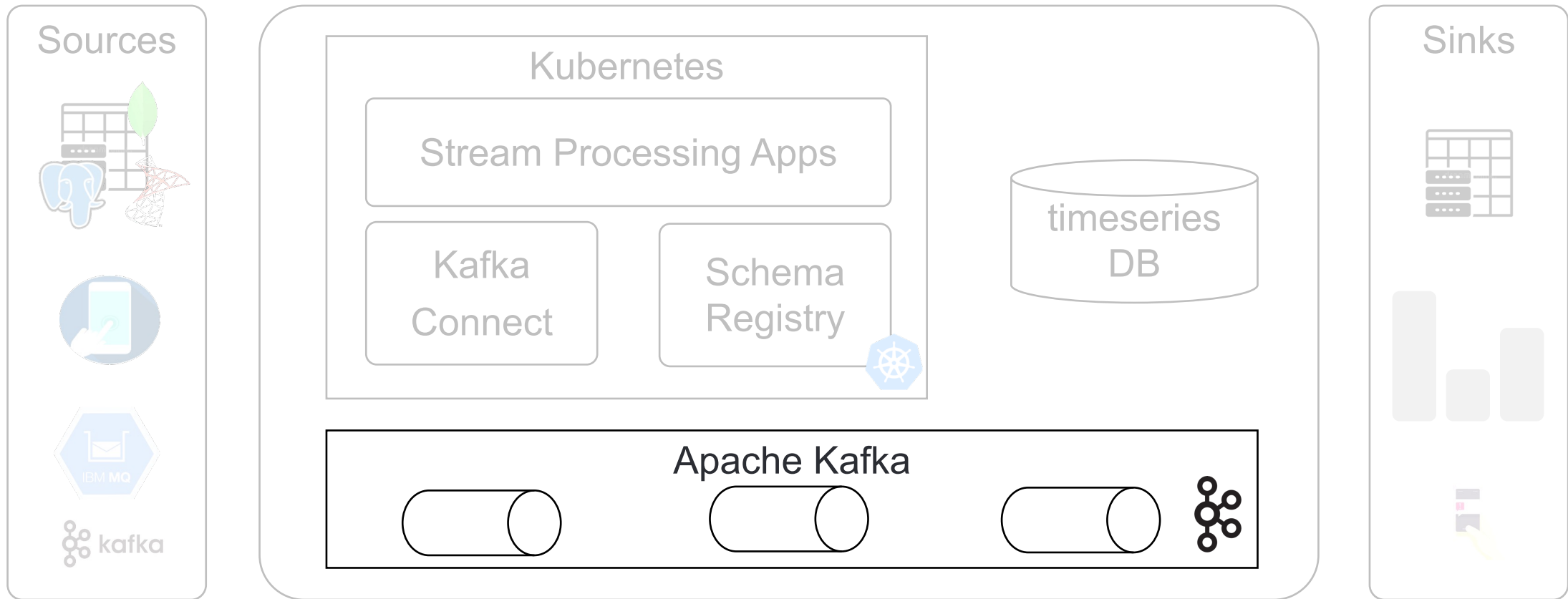


# Из чего собрать платформу?



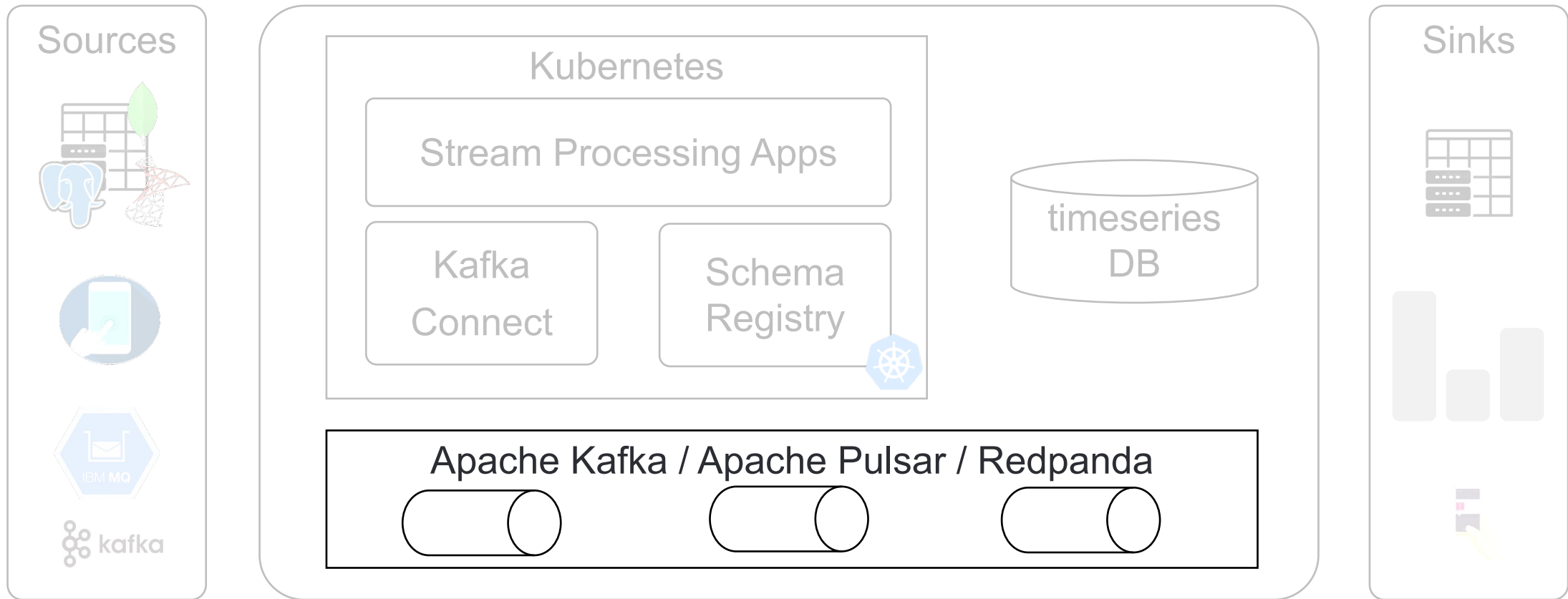


# Из чего собрать платформу?





# Из чего собрать платформу?



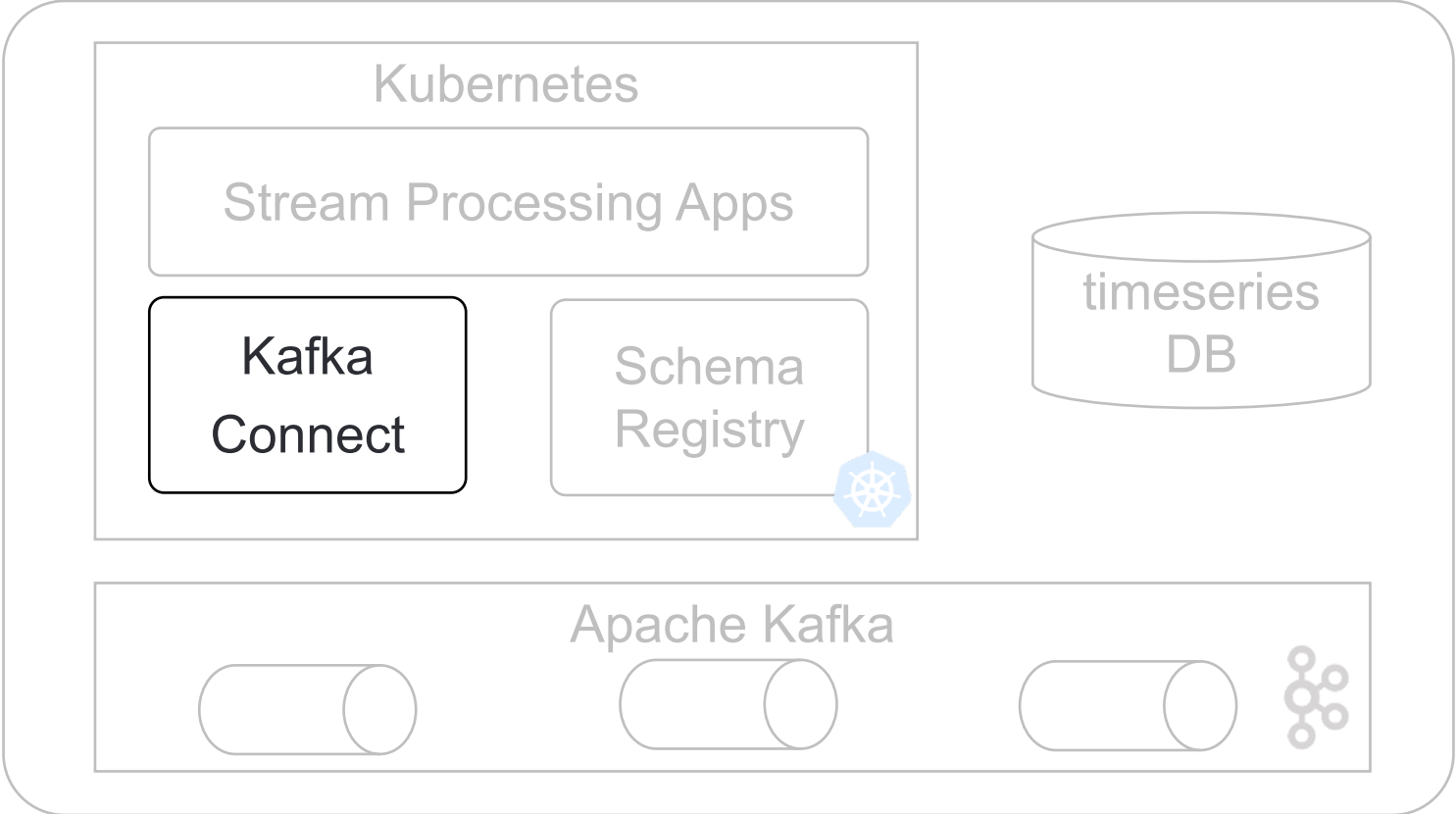


# Из чего собрать платформу?

Sources

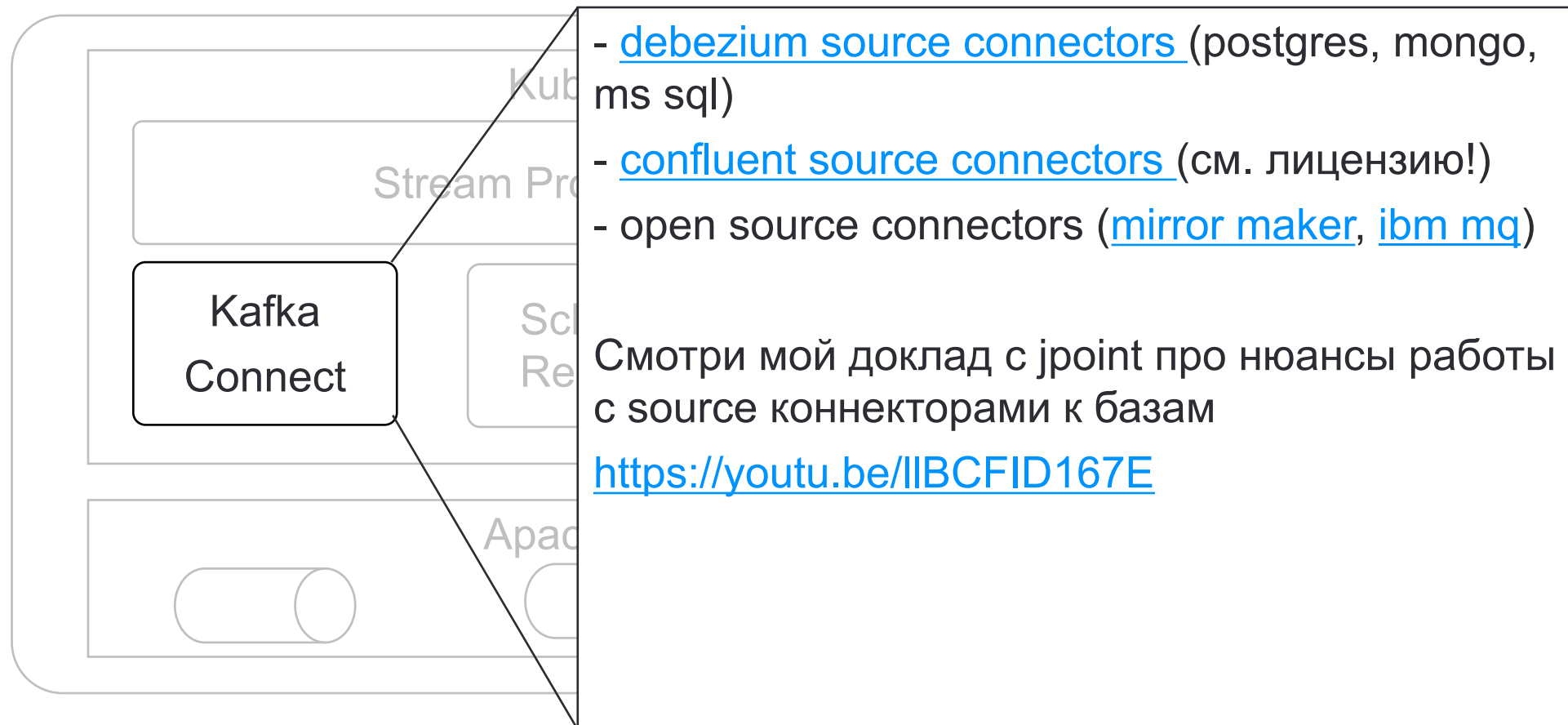
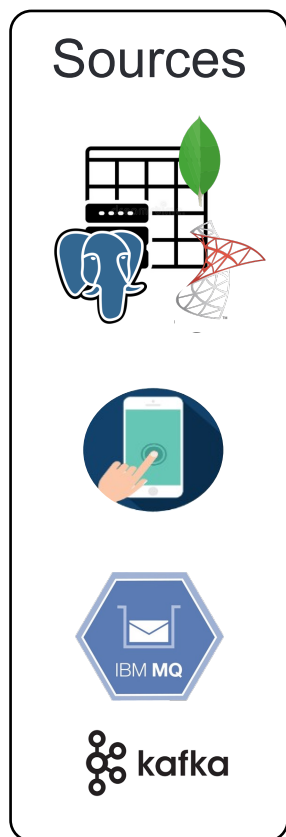
IBM MQ

kafka



Sinks

# Из чего собрать платформу?



- [debezium source connectors](#) (postgres, mongo, ms sql)
- [confluent source connectors](#) (см. лицензию!)
- open source connectors ([mirror maker](#), [ibm mq](#))

Смотри мой доклад с jpoint про нюансы работы с source коннекторами к базам

<https://youtu.be/IIBCFFID167E>



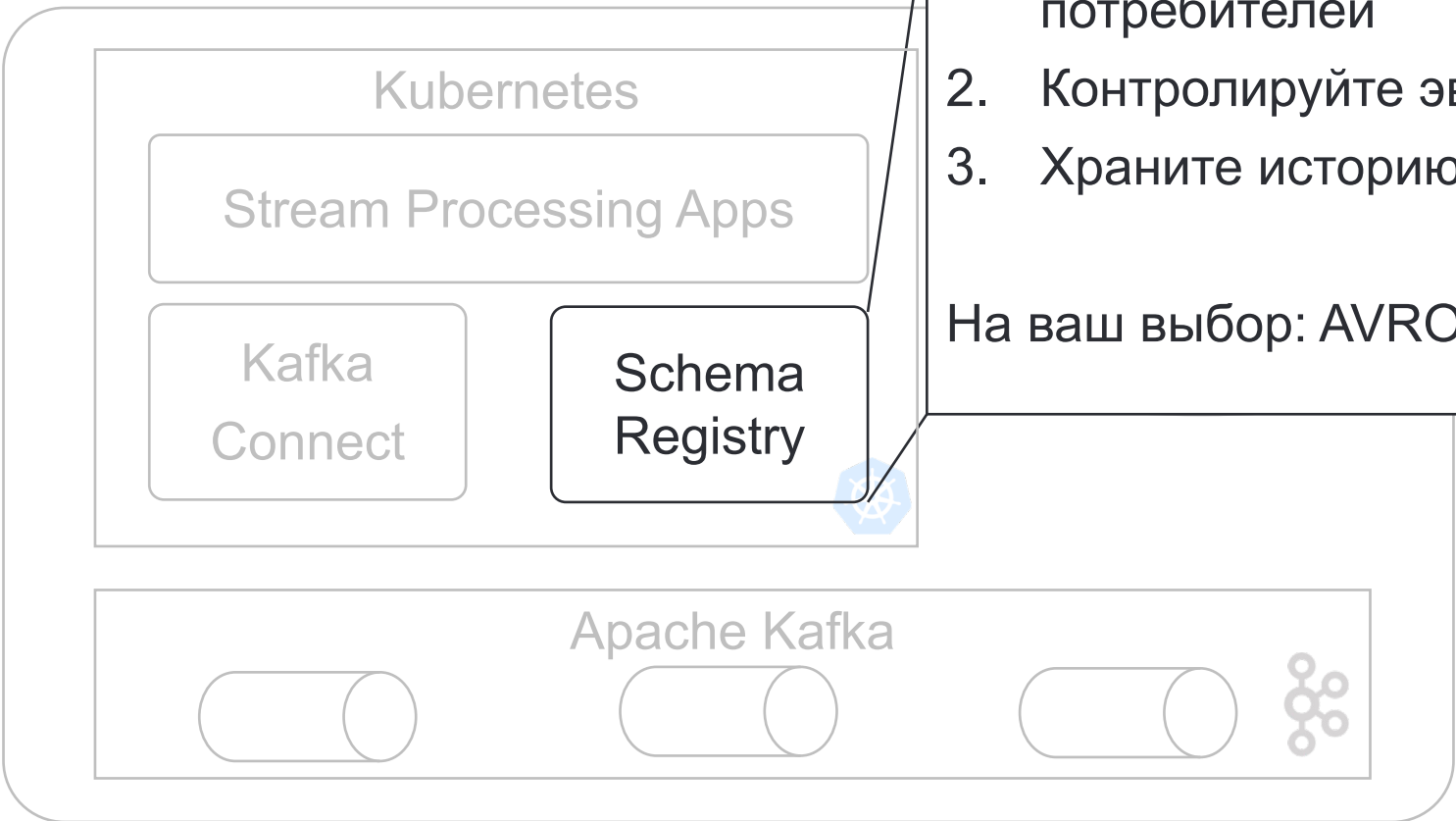


# Из чего собрать платформу?

Sources

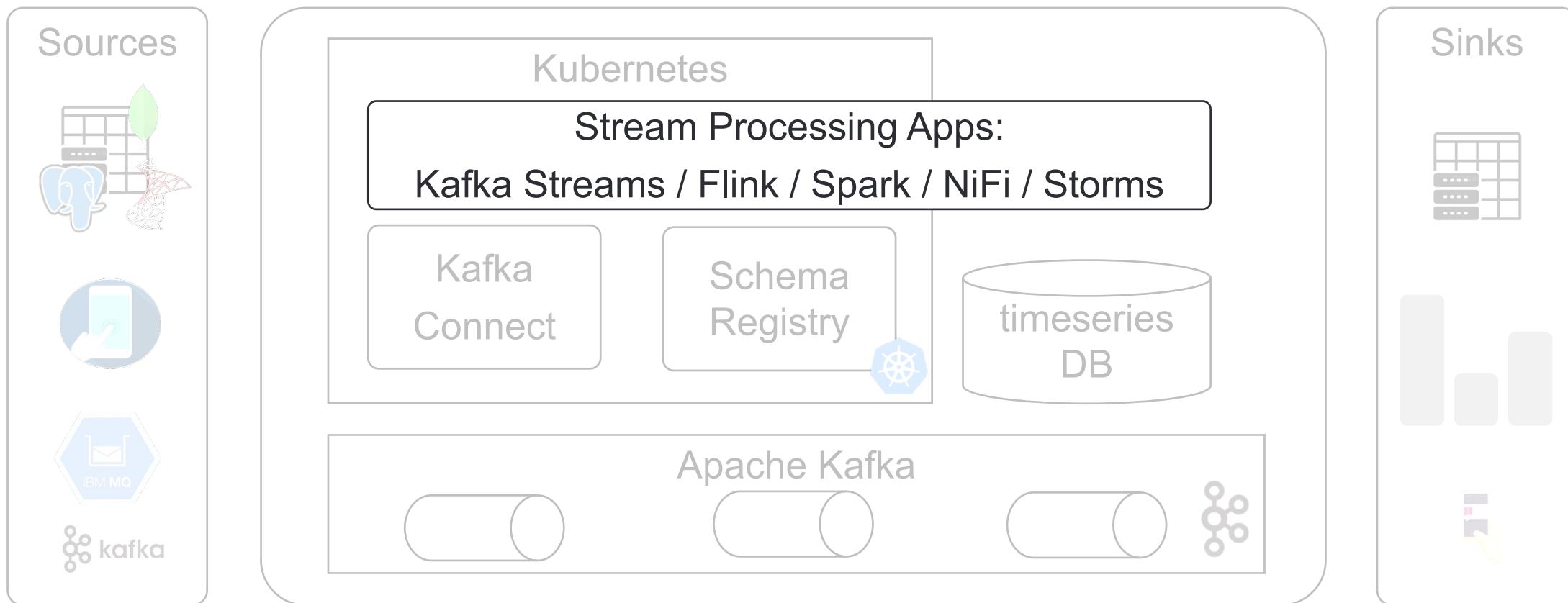
IBM MQ

kafka



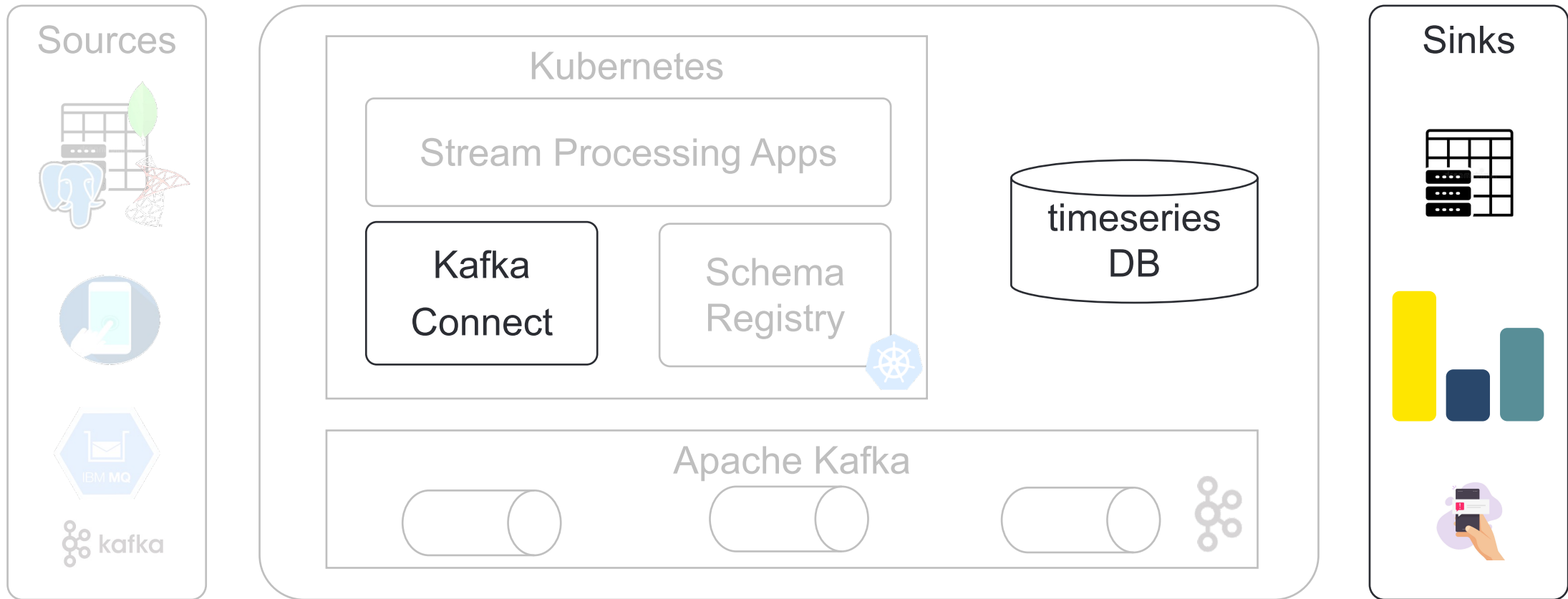
1. Гарантируйте схему для потребителей
  2. Контролируйте эволюцию
  3. Храните историю изменений схемы
- На ваш выбор: AVRO / JSON / protobuf

# Из чего собрать платформу?





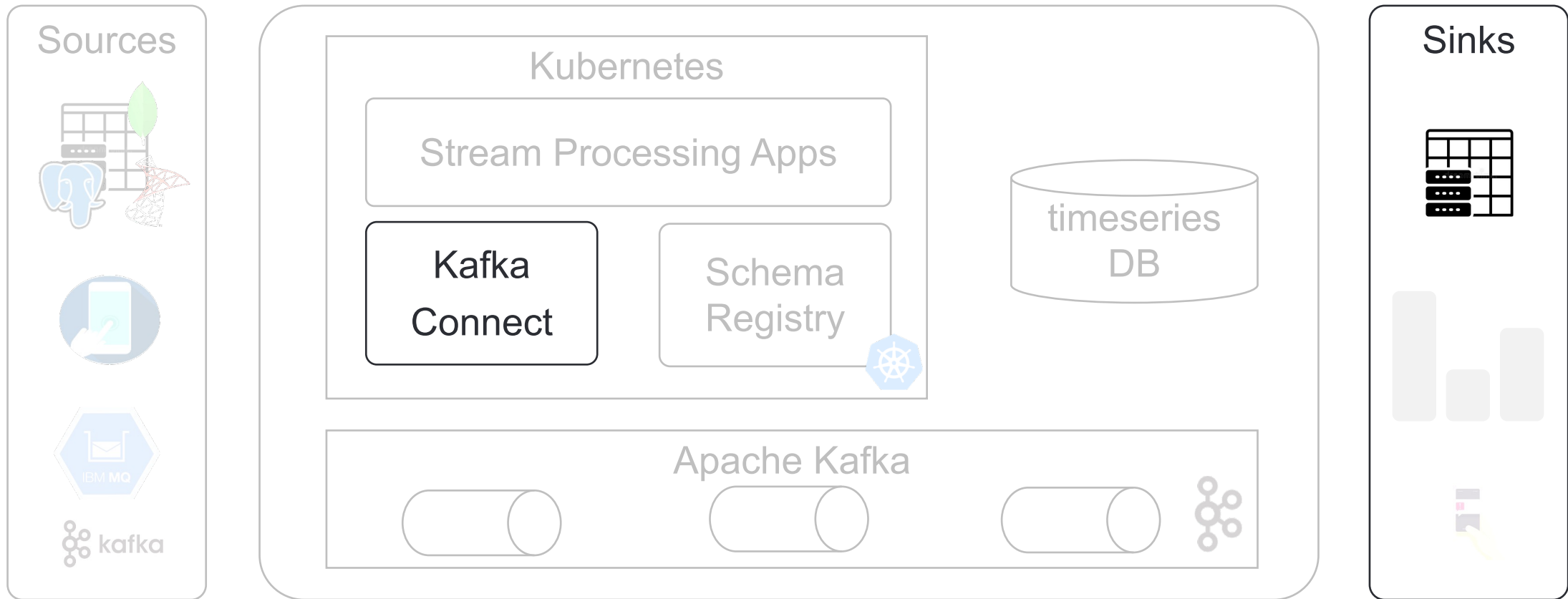
# Из чего собрать платформу?





# Из чего собрать платформу?

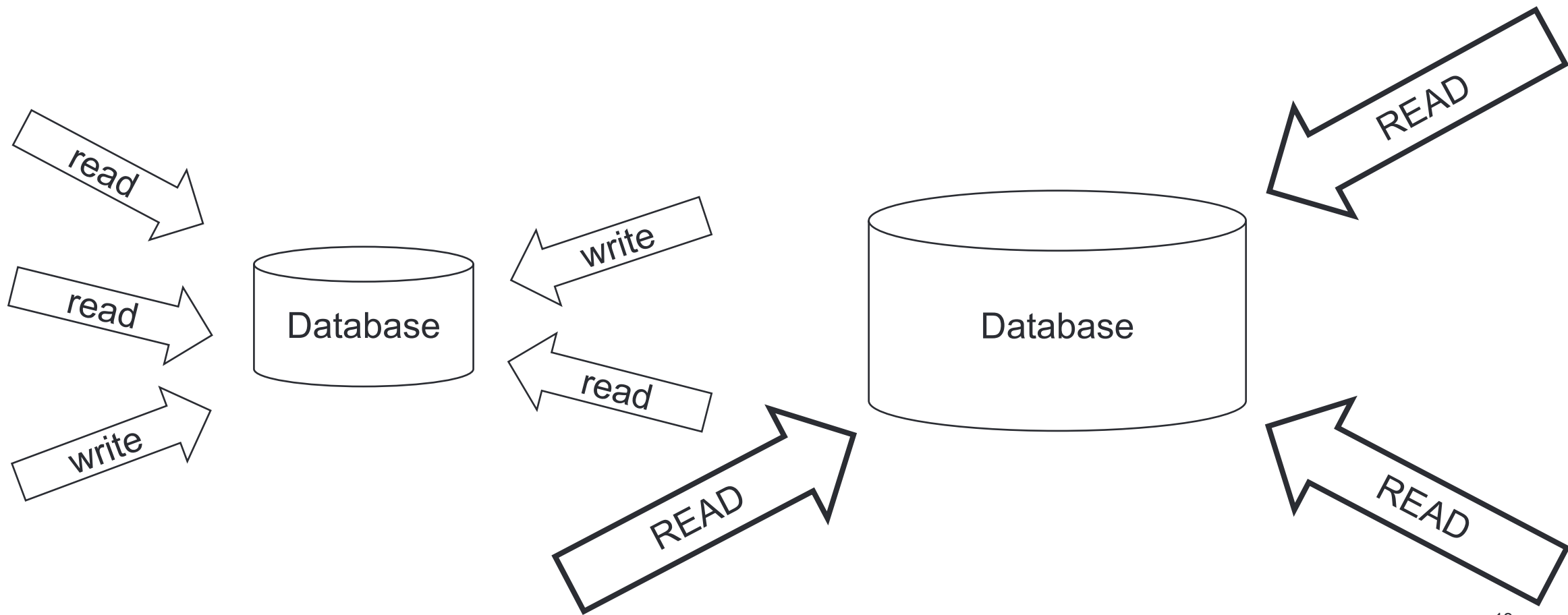
Настроить синк в архивную базу





# Из чего собрать платформу?

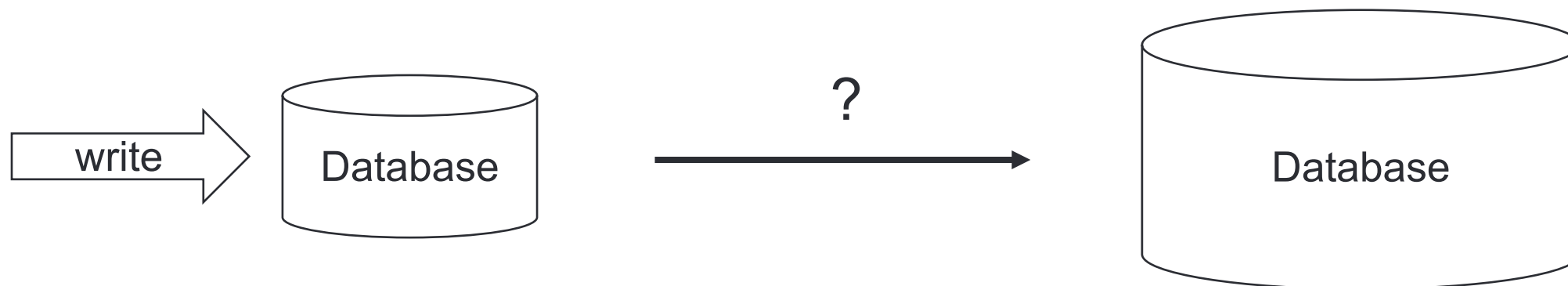
Настроить синк в архивную базу. Кейс команды рублевых платежей





# Из чего собрать платформу?

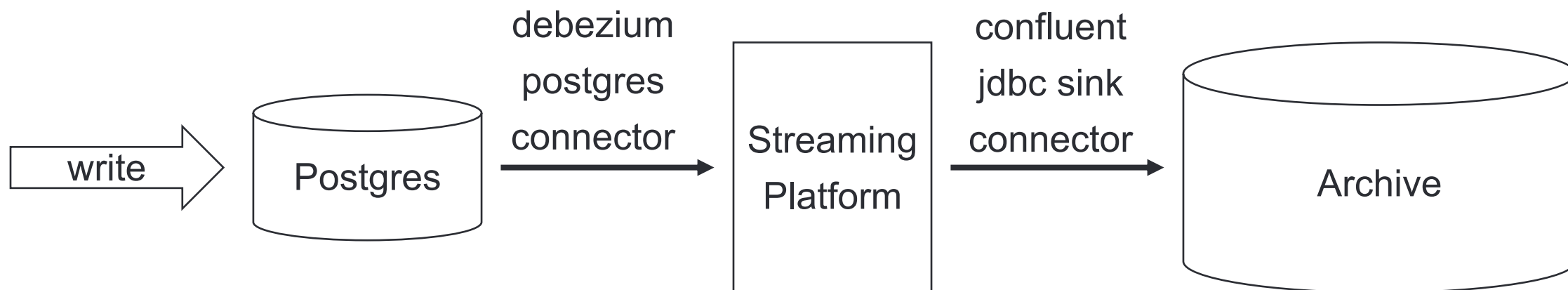
Настроить синк в архивную базу. Кейс команды рублевых платежей





# Из чего собрать платформу?

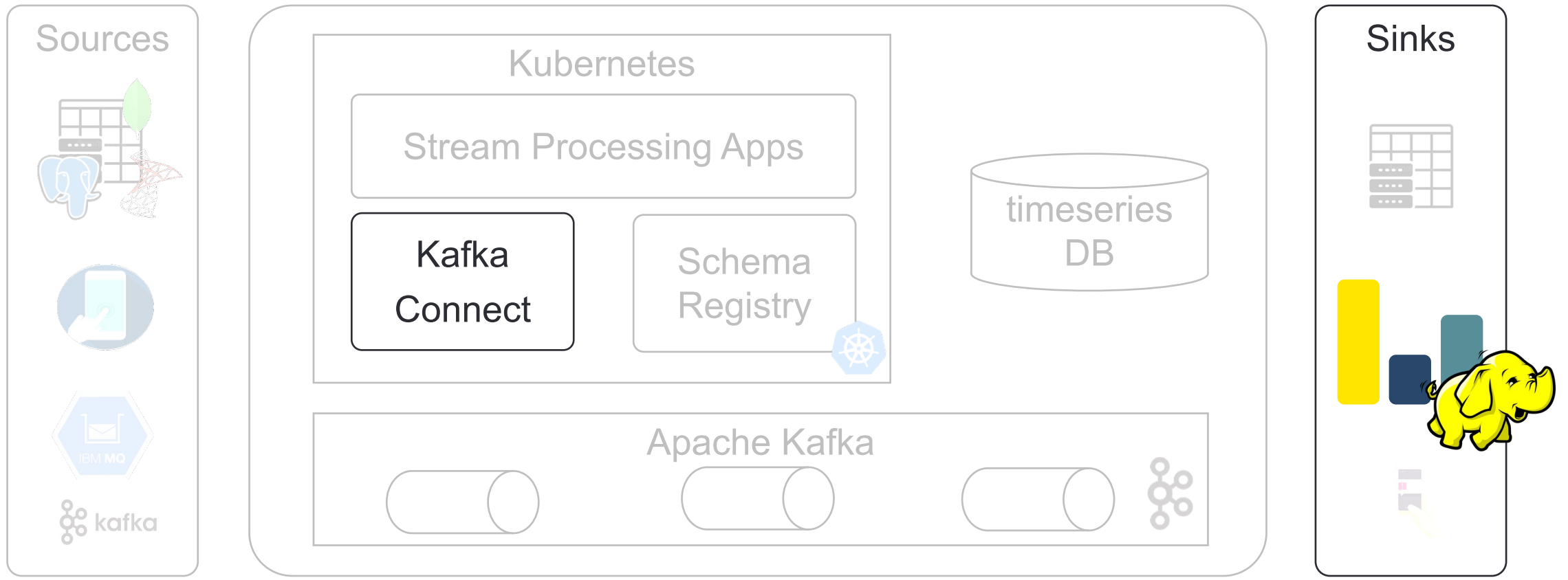
Настроить синк в архивную базу





# Из чего собрать платформу?

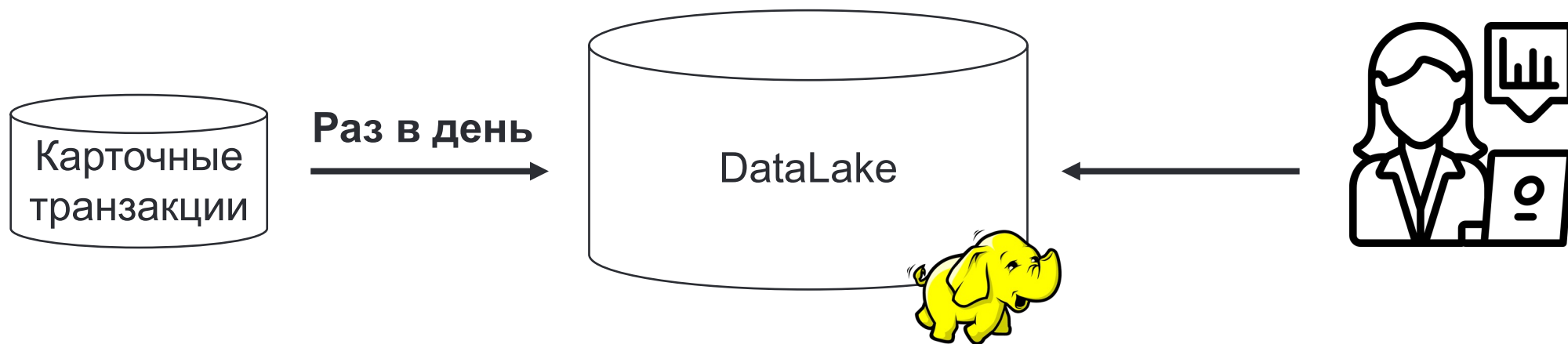
Настроить синк в Datalake (HDFS + Hive / S3). Parquet / Avro / ORC







# Как раньше грузились данные для аналитики?





# Почему так долго?

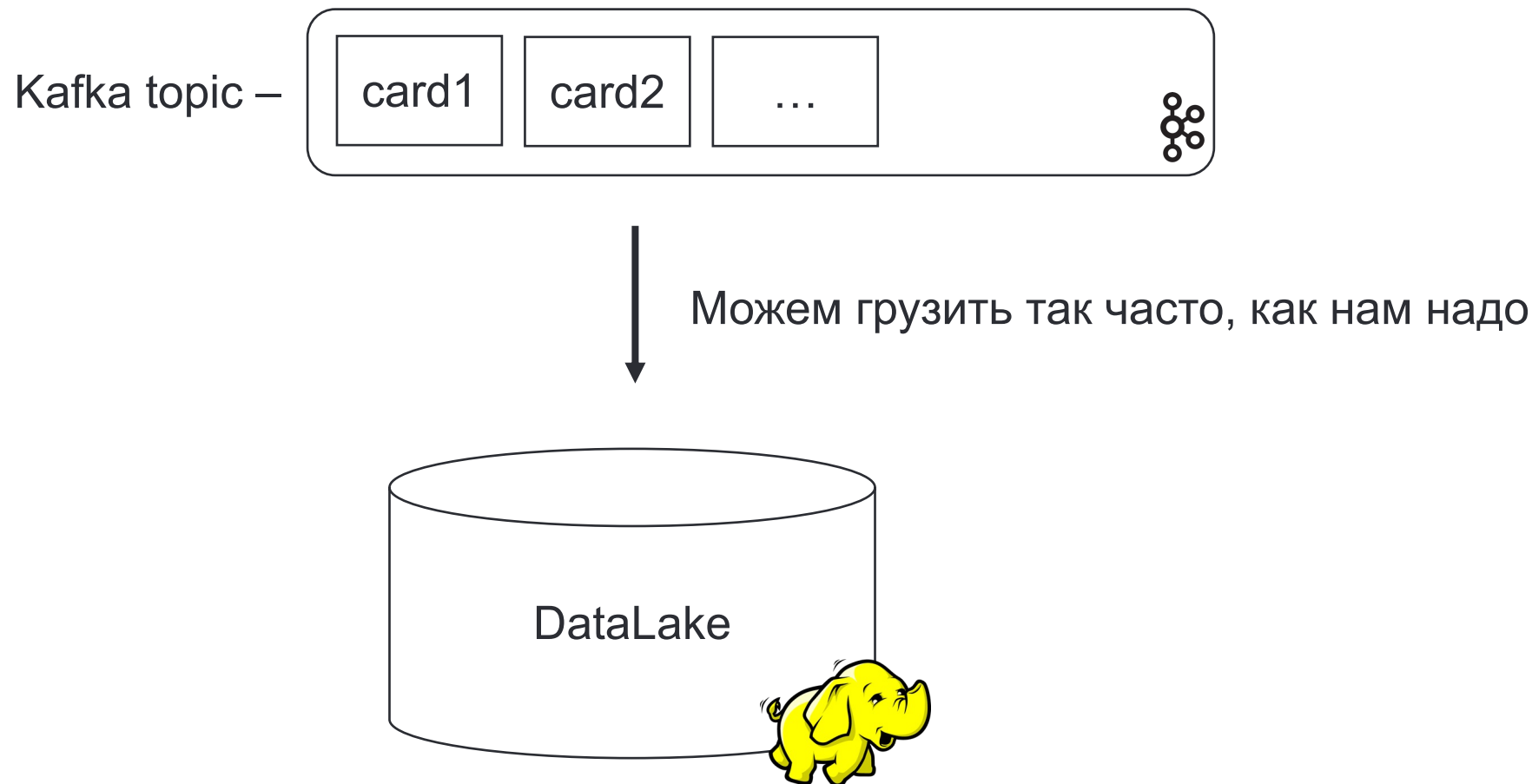


\* Проводится только ночью, чтобы не влиять на клиентов



# Из чего собрать платформу?

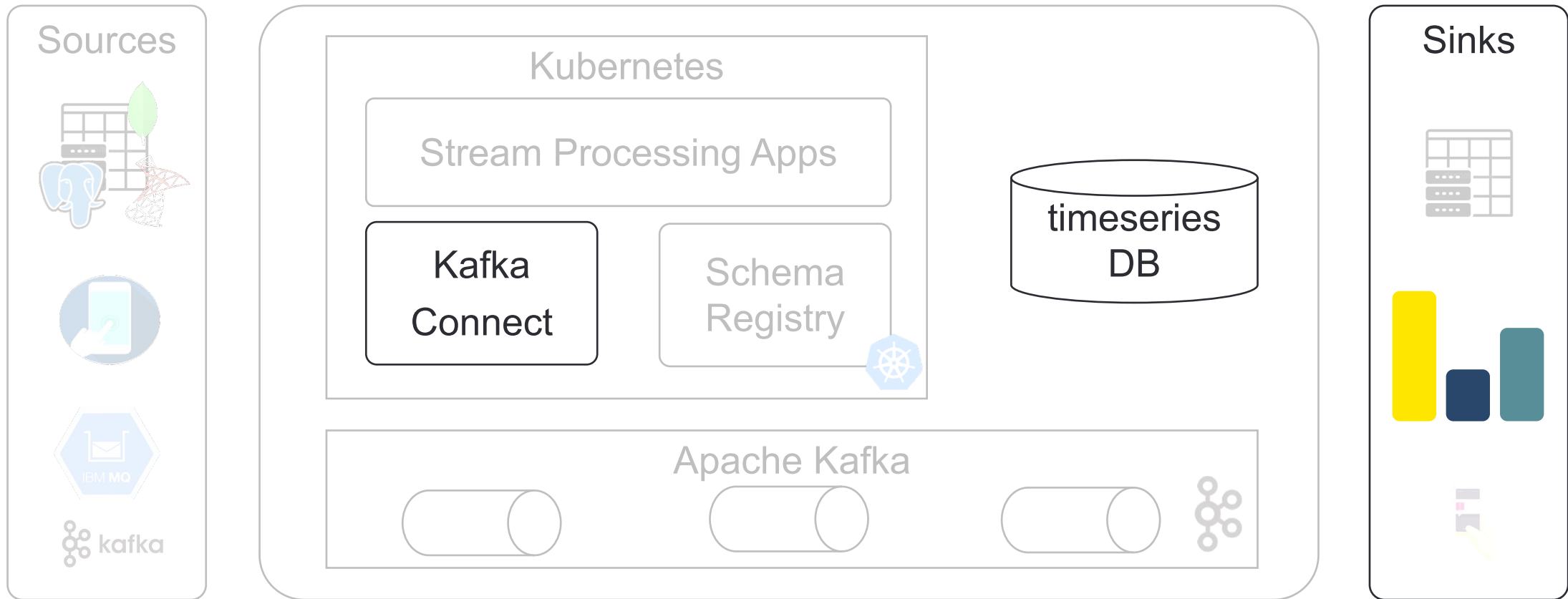
Настроить синк в Datalake (HDFS + Hive / S3). Parquet / Avro / ORC





# Из чего собрать платформу?

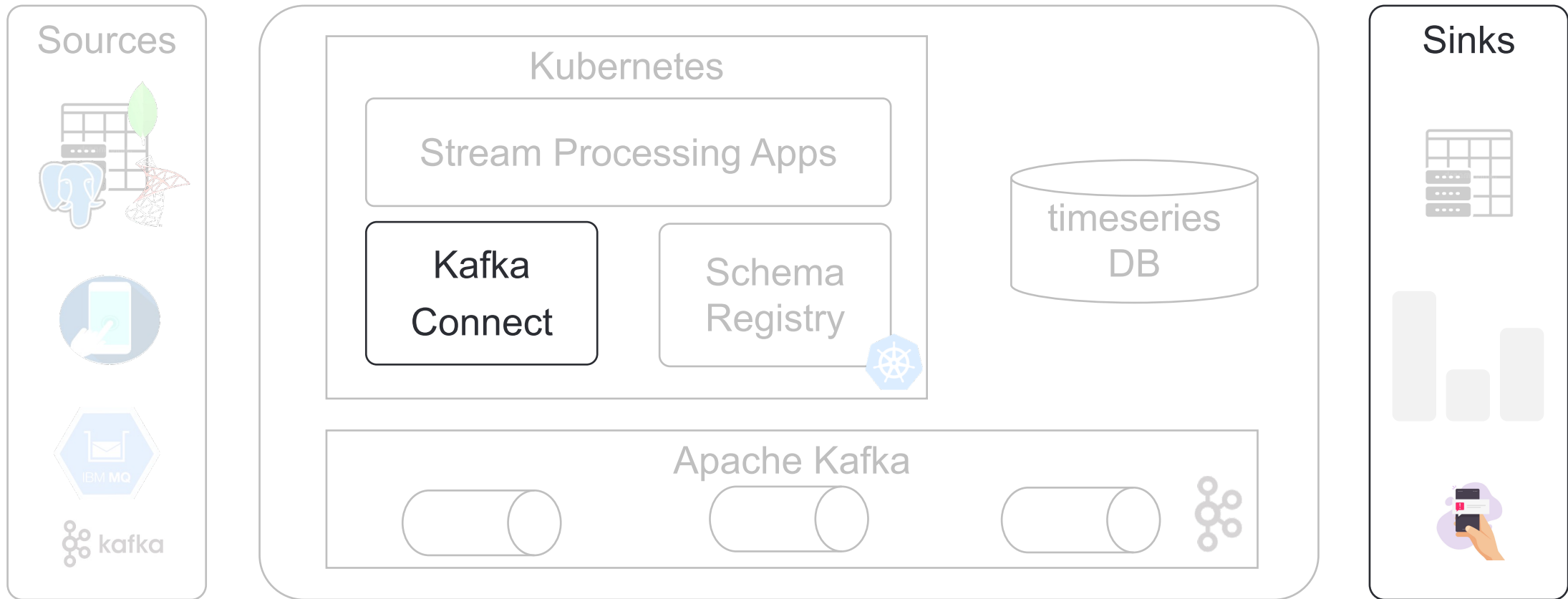
Добавить real-time аналитику





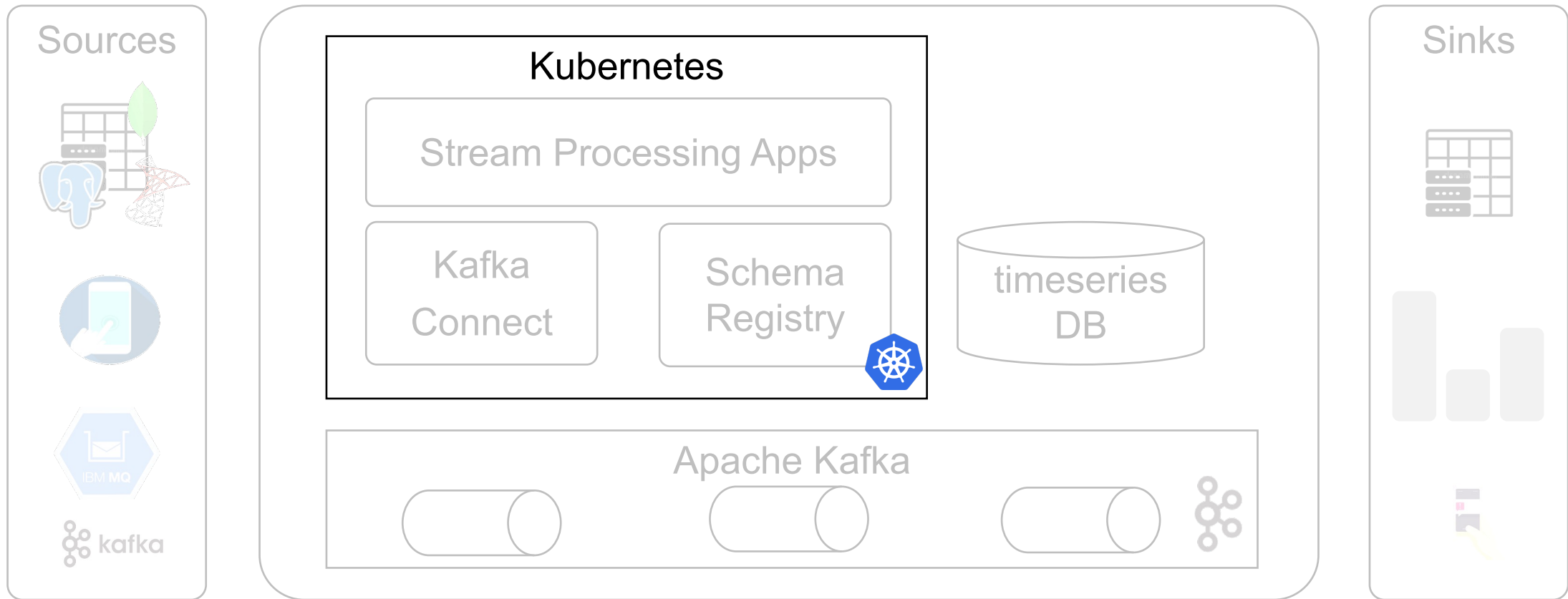
# Из чего собрать платформу?

Отправить события напрямую клиентам



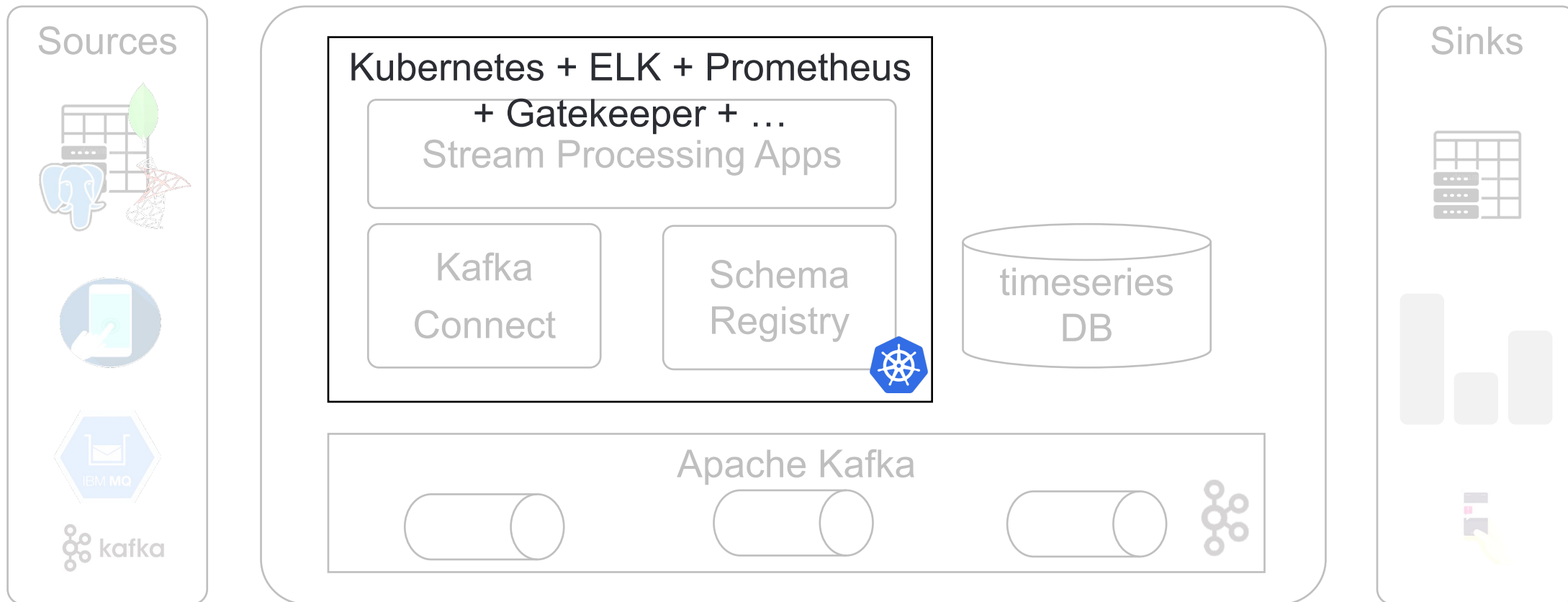


# Из чего собрать платформу?



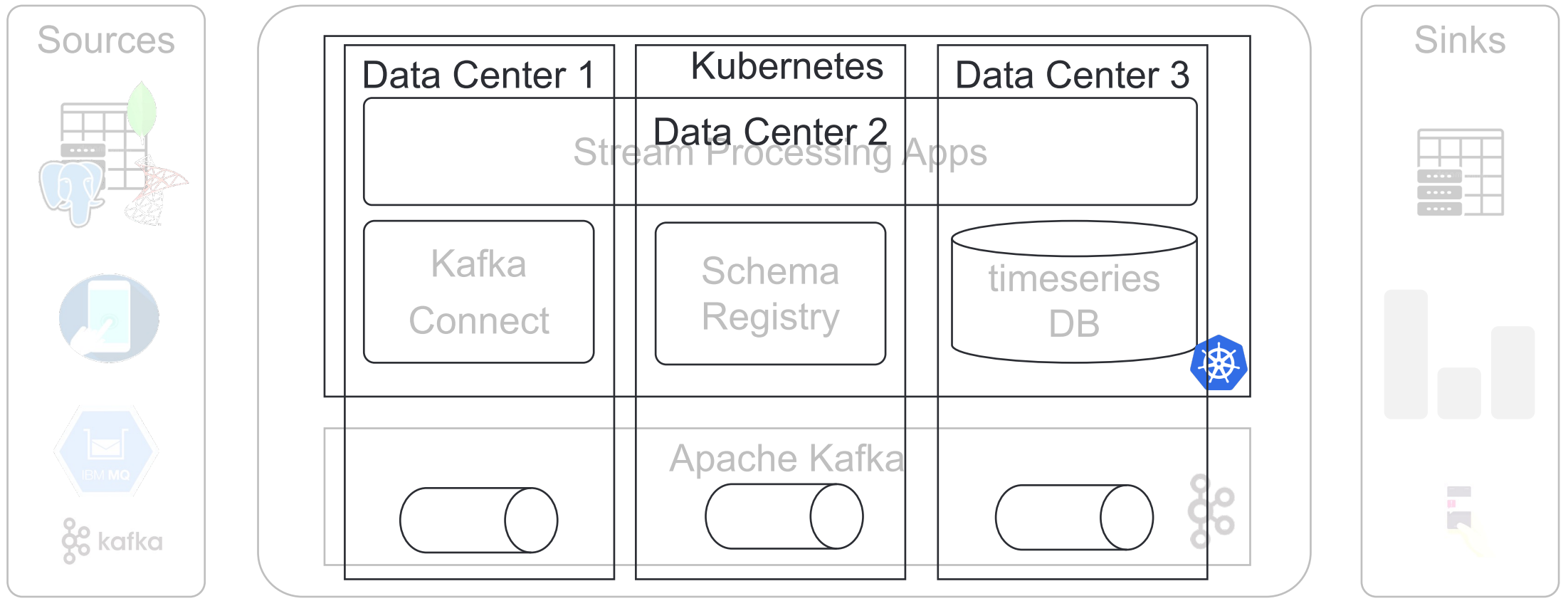


# Из чего собрать платформу?





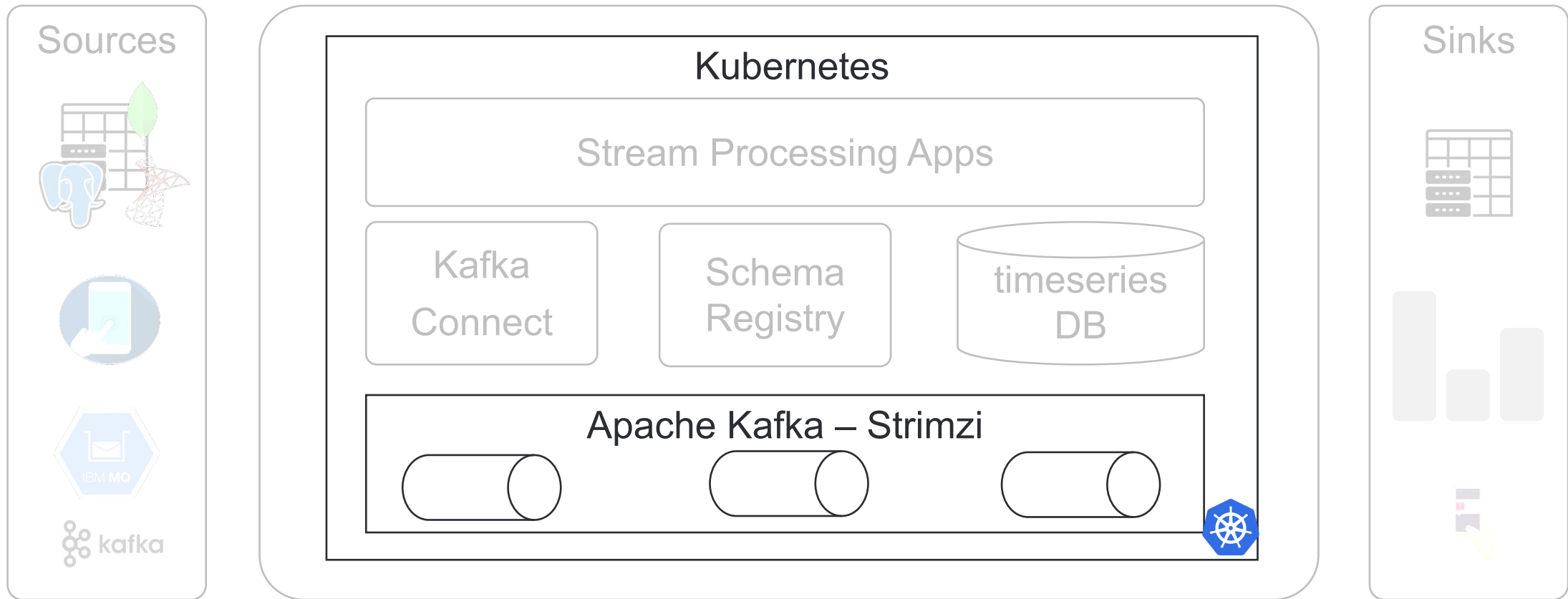
# Из чего собрать платформу?







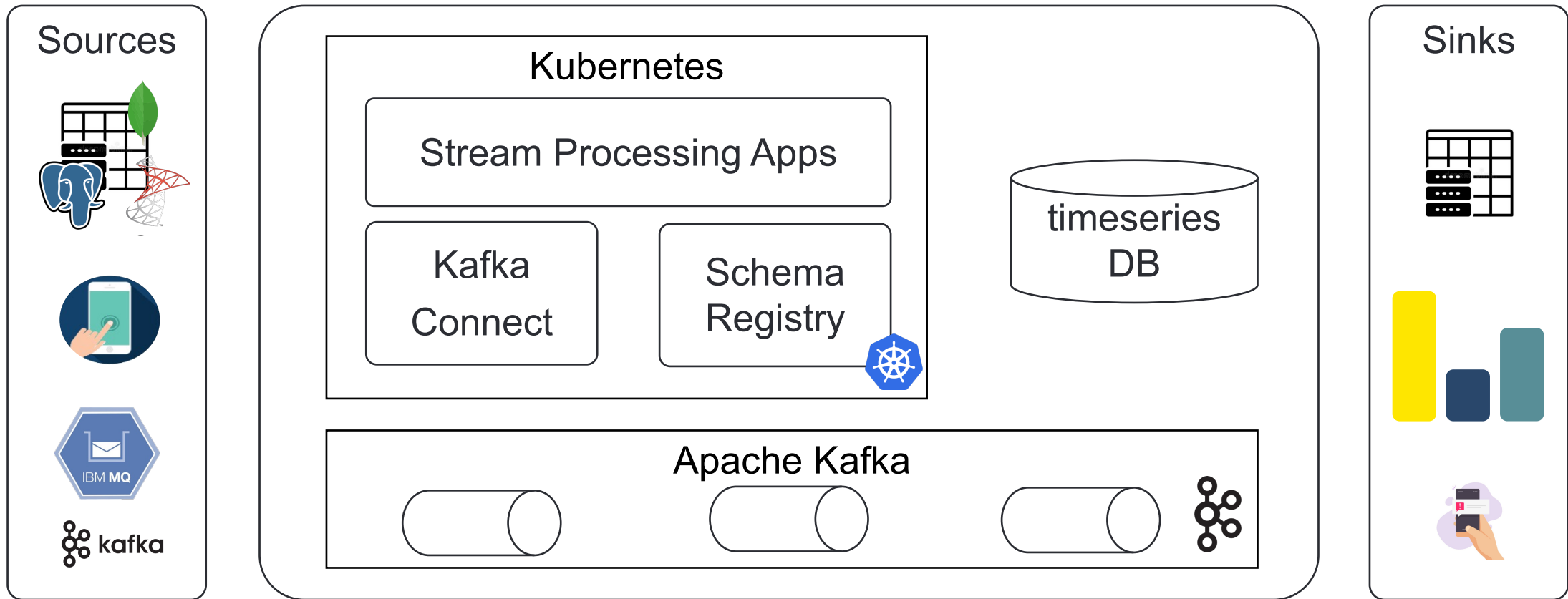
# Из чего собрать платформу?





# Из чего собрать платформу?

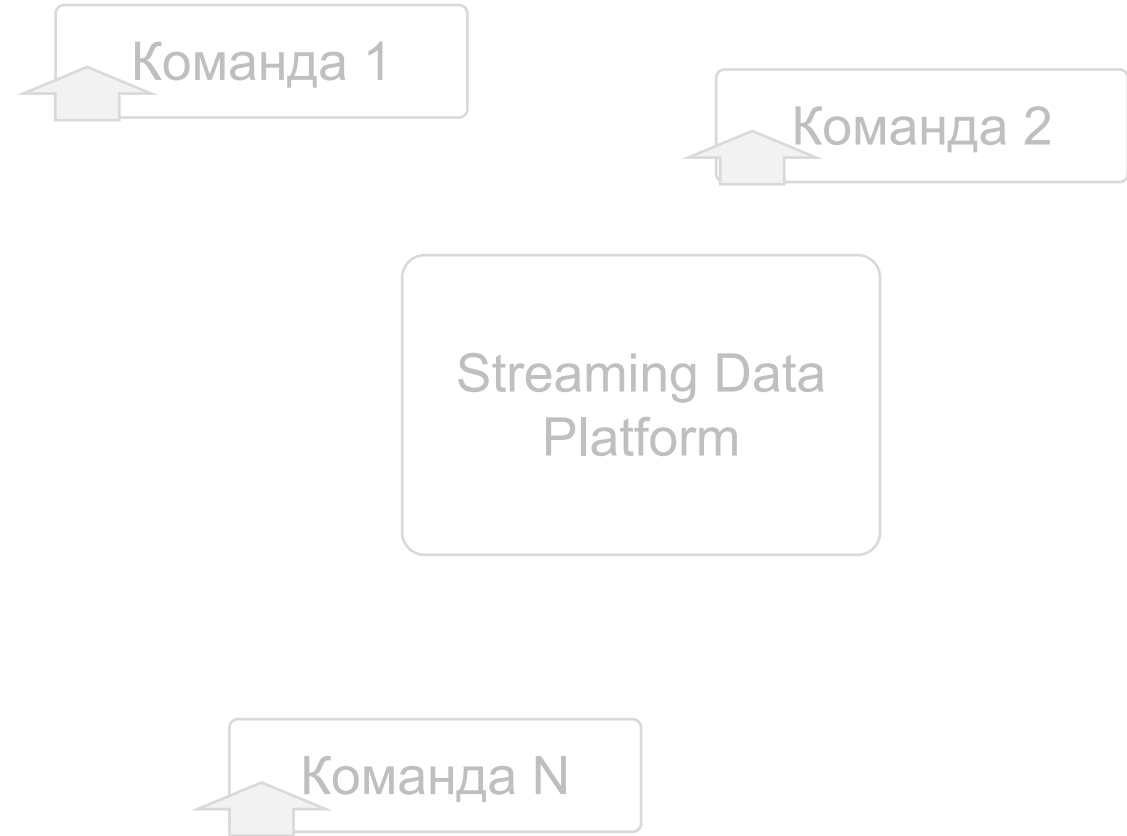
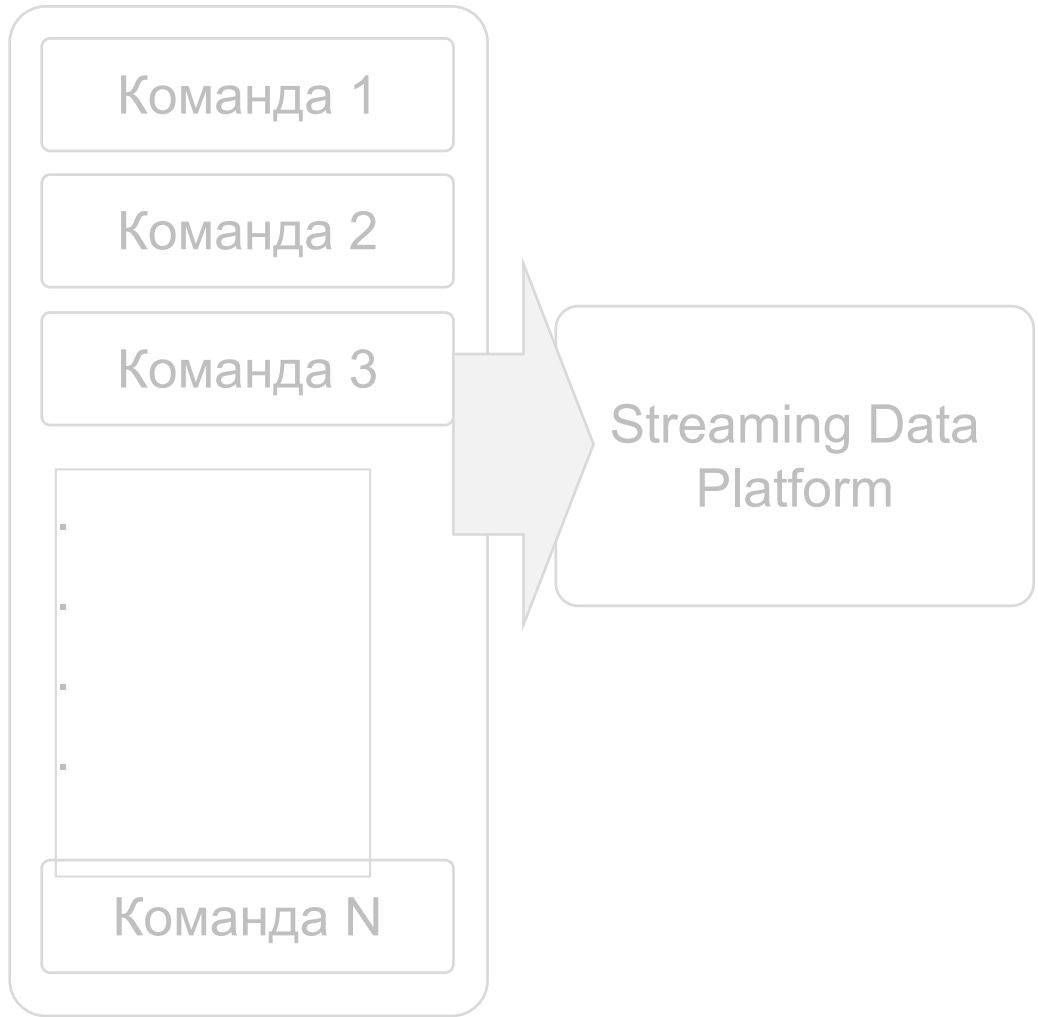
Все это – open source



**А что если юзеров много?**



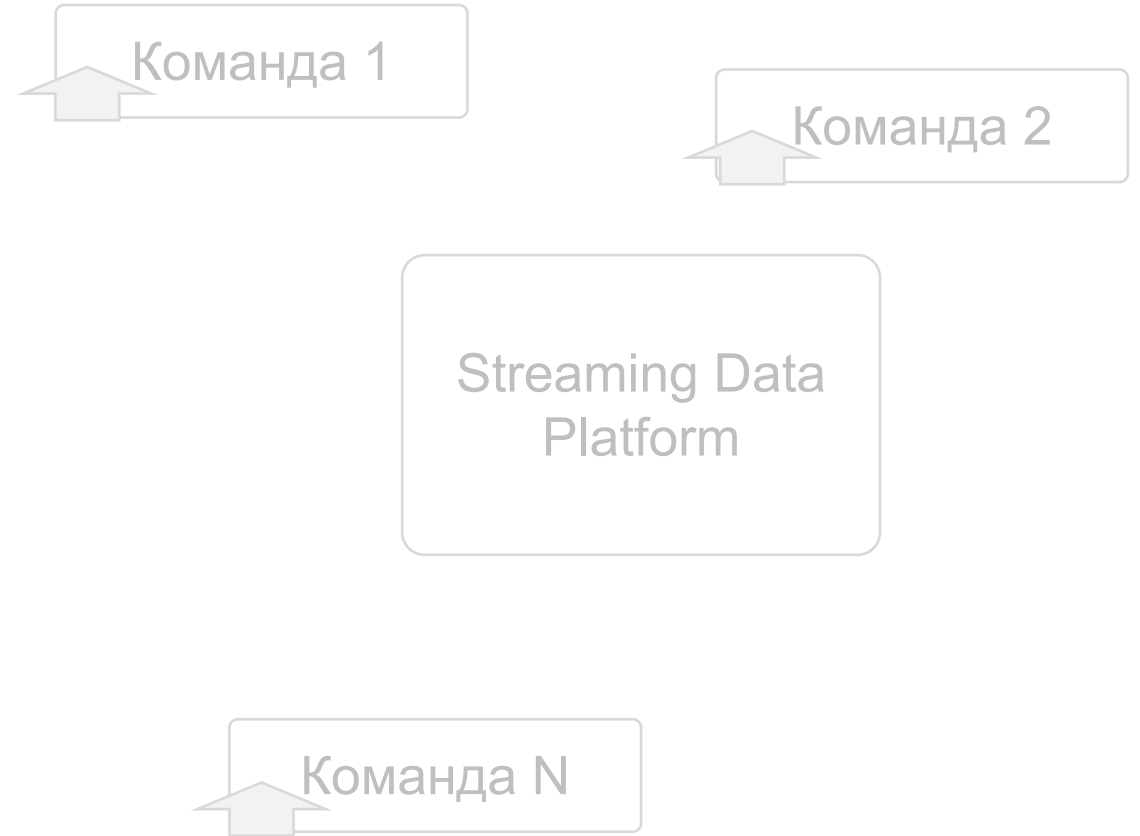
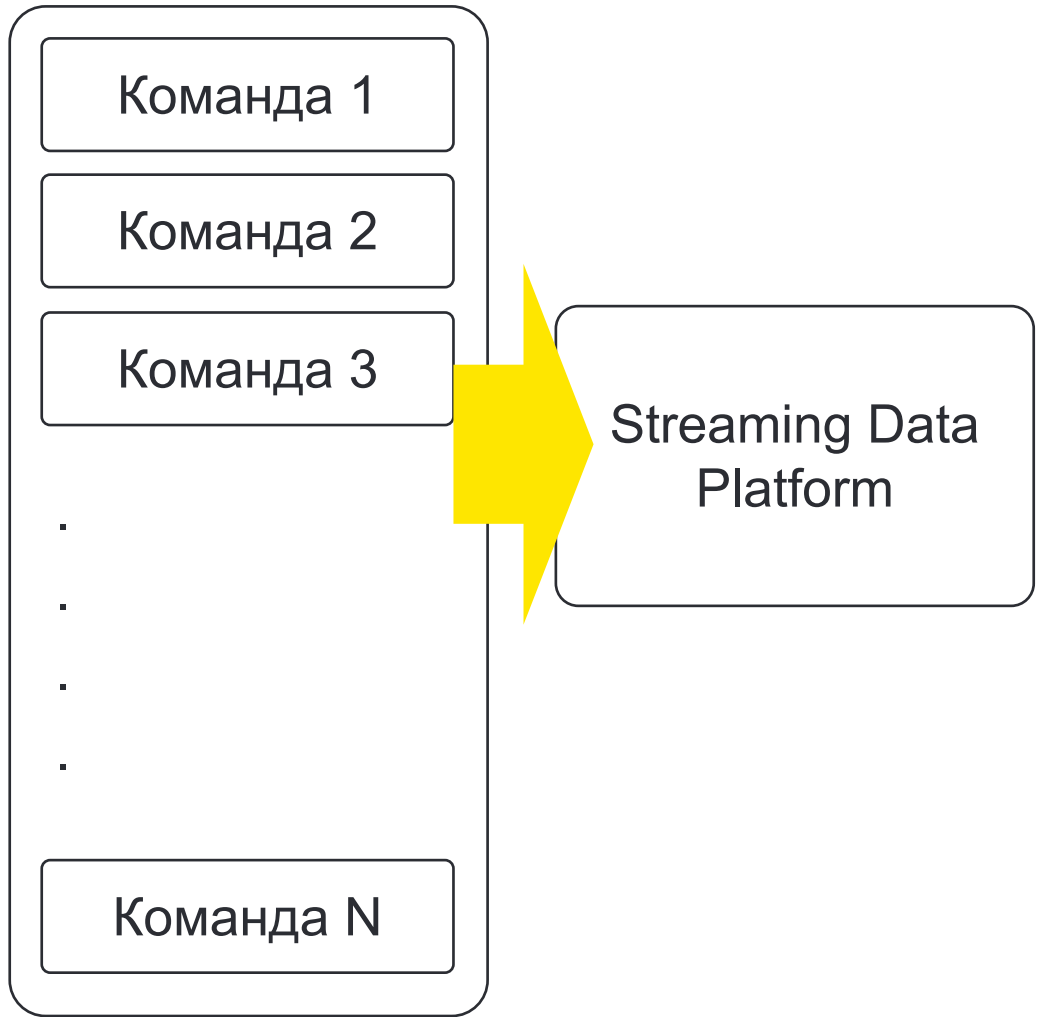
# Как организовать работу с платформой?





# Как организовать работу с платформой?

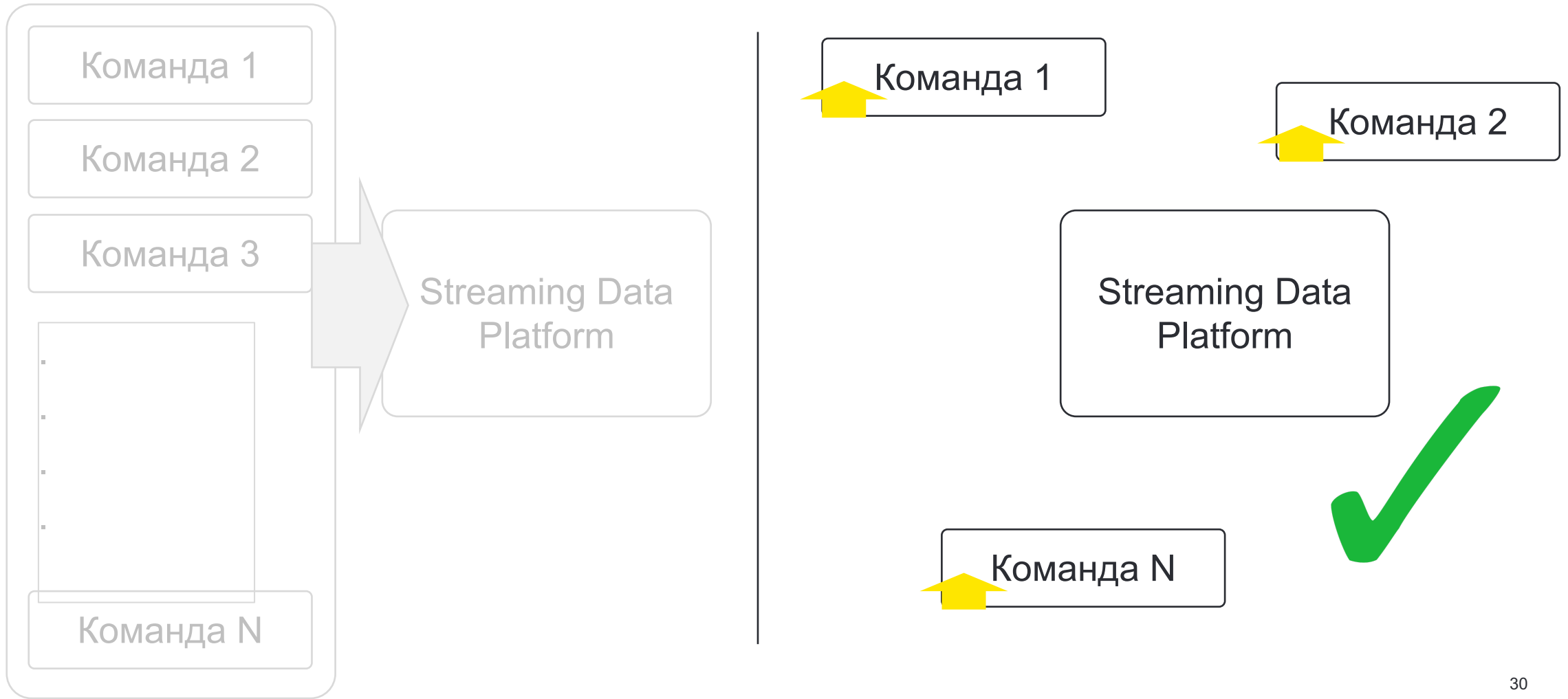
Вариант 1: Платформа работает как сервис





# Как организовать работу с платформой?

Вариант 2: Data Mesh – федеративное управление данными





# Как сделать ЭТОТ Data Mesh?

# Как сделать команды самостоятельными?



Обеспечить низкий порог входа



Решить общие проблемы



Сделать данные доступными



# Как сделать команды самостоятельными?



Обеспечить низкий порог входа



Решить общие проблемы



Сделать данные доступными

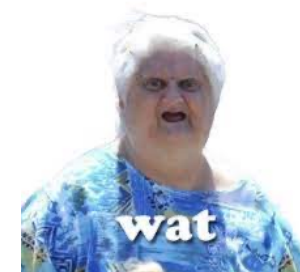


# Как сделать команды самостоятельными?

## Обеспечить низкий порог входа

```
{
  "database.hostname": "{{ DATABASE_HOST_NAME }}",
  "database.user": "${vault:kv/{{ PATH_IN_VAULT }}:username}",
  "database.password": "${vault:kv/{{ PATH_IN_VAULT }}:password}",
  "database.names": "{{ DATABASE_NAMES }}",
  "topic.prefix": "{{ TOPIC_PREFIX }}",
  "table.include.list": "{{ LIST_OF_TABLES_TO_REPLICATE }}",
  "connector.class": "io.debezium.connector.sqlserver.SqlServerConnector",
  "database.port": "1433",
  "tombstones.on.delete": "false",
  "database.trustServerCertificate": true,
  "schema.history.internal.kafka.bootstrap.servers": "...",
  "schema.history.internal.kafka.topic": "{{ CONNECTOR_NAME }}.schemahistory",
  "schema.history.internal.consumer.sasl.jaas.config": "...",
  "schema.history.internal.consumer.sasl.mechanism": "GSSAPI",
  "schema.history.internal.consumer.security.protocol": "SASL_SSL",
  "schema.history.internal.producer.sasl.jaas.config": "...",
  "schema.history.internal.producer.sasl.mechanism": "GSSAPI",
  "schema.history.internal.producer.security.protocol": "SASL_SSL",
  "heartbeat.interval.ms": 3600000,
  "key.converter": "io.confluent.connect.avro.AvroConverter",
  "key.converter.schema.registry.url": "https://...k8s.raiffeisen.ru",
  "value.converter": "io.confluent.connect.avro.AvroConverter",
  "value.converter.schema.registry.url": "https://...k8s.raiffeisen.ru",
  "transforms": "SetSchemaName",
  "transforms.SetSchemaName.type": "org.apache.kafka.connect.transforms.SetSchemaMetadata$Value",
  "transforms.SetSchemaName.schema.name": "ru.raif.sdp.{{ SHORT_TEAM_NAME }}.{{ TABLE_SCHEMA_NAME }}.{{ TABLE_NAME }}",

  "topicsToCreate": "{{ TOPICS_NAMES }}",
  "topicsToCreate.$TOPIC_NAME.partitions.count": "{{ TOPIC_PARTITIONS_COUNT }}",
  "topicsToCreate.$TOPIC_NAME.configs": "{{ TOPIC_CONFIGS }}",
  "topicsToCreate.$TOPIC_NAME.mim-roles": "{{ MIM_ROLES }}",
  "topicsToCreate.$TOPIC_NAME.description": "{{ DESCRIPTION }}",
  "topicsToCreate.$TOPIC_NAME.business-owner": "{{ BUSINESS_OWNER }}",
  "topicsToCreate.$TOPIC_NAME.source-name": "{{ SOURCE_NAME }}",
  "topicsToCreate.$TOPIC_NAME.source-ail-id": "{{ SOURCE_AIL_ID }}"
}
```



# Как сделать команды самостоятельными?

Обеспечить низкий порог входа

```
{
  "database.hostname": "{{ DATABASE_HOST_NAME }}",
  "database.user": "${vault:kv/{{ PATH_IN_VAULT }}:username}",
  "database.password": "${vault:kv/{{ PATH_IN_VAULT }}:password}",
  "database.names": "{{ DATABASE_NAMES }}",
  "topic.prefix": "{{ TOPIC_PREFIX }}",
  "table.include.list": "{{ LIST_OF_TABLES_TO_REPLICATE }}",

  "connector.class": "io.debezium.connector.sqlserver.SqlServerConnector",
  "database.port": "1433",
  "tombstones.on.delete": "false",
  "database.trustServerCertificate": true,
  "schema.history.internal.kafka.bootstrap.servers": "...",
  "schema.history.internal.kafka.topic": "{{ CONNECTOR_NAME }}.schemahistory",
  "schema.history.internal.consumer.sasl.jaas.config": "...",
  "schema.history.internal.consumer.sasl.mechanism": "GSSAPI",
  "schema.history.internal.consumer.security.protocol": "SASL_SSL",
  "schema.history.internal.producer.sasl.jaas.config": "...",
  "schema.history.internal.producer.sasl.mechanism": "GSSAPI",
  "schema.history.internal.producer.security.protocol": "SASL_SSL",
  "heartbeat.interval.ms": 3600000,
  "key.converter": "io.confluent.connect.avro.AvroConverter",
  "key.converter.schema.registry.url": "https://...k8s.raiffeisen.ru",
  "value.converter": "io.confluent.connect.avro.AvroConverter",
  "value.converter.schema.registry.url": "https://...k8s.raiffeisen.ru",
  "transforms": "SetSchemaName",
  "transforms.SetSchemaName.type": "org.apache.kafka.connect.transforms.SetSchemaMetadata$Value",
  "transforms.SetSchemaName.schema.name": "ru.raif.sdp.{{ SHORT_TEAM_NAME }}.{{ TABLE_SCHEMA_NAME }}.{{ TABLE_NAME }}"
}
```



# Как сделать команды самостоятельными?

Обеспечить низкий порог входа *и простые метаданные!*

```
{
  "database.hostname": "{{ DATABASE_HOST_NAME }}",
  "database.user": "${vault:kv/{{ PATH_IN_VAULT }}:username}",
  "database.password": "${vault:kv/{{ PATH_IN_VAULT }}:password}",
  "database.names": "{{ DATABASE_NAMES }}",
  "topic.prefix": "{{ TOPIC_PREFIX }}",
  "table.include.list": "{{ LIST_OF_TABLES_TO_REPLICATE }}",
  "connector.class": "io.debezium.connector.sqlserver.SqlServerConnector",
  "database.port": "1433",
  "tombstones.on.delete": "false",
  "database.trustServerCertificate": true,
  "schema.history.internal.kafka.bootstrap.servers": "...",
  "schema.history.internal.kafka.topic": "{{ CONNECTOR_NAME }}.schemahistory",
  "schema.history.internal.consumer.sasl.jaas.config": "...",
  "schema.history.internal.consumer.sasl.mechanism": "GSSAPI",
  "schema.history.internal.consumer.security.protocol": "SASL_SSL",
  "topicsToCreate": "{{ TOPICS_NAMES }}",

  "topicsToCreate.$TOPIC_NAME.partitions.count": "{{ TOPIC_PARTITIONS_COUNT }}",
  "topicsToCreate.$TOPIC_NAME.configs": "{{ TOPIC_CONFIGS }}",
  "topicsToCreate.$TOPIC_NAME.mim-roles": "{{ MIM_ROLES }}",
  "topicsToCreate.$TOPIC_NAME.description": "{{ DESCRIPTION }}",
  "topicsToCreate.$TOPIC_NAME.business-owner": "{{ BUSINESS_OWNER }}",
  "topicsToCreate.$TOPIC_NAME.source-name": "{{ SOURCE_NAME }}"
}
```

# Как сделать команды самостоятельными?

Нам есть куда расти в этом

```
{
  "database.hostname": "{{ DATABASE_HOST_NAME }}",
  "database.user": "${vault:kv/{{ PATH_IN_VAULT }}:username}",
  "database.password": "${vault:kv/{{ PATH_IN_VAULT }}:password}",
  "database.names": "{{ DATABASE_NAMES }}",
  "topic.prefix": "{{ TOPIC_PREFIX }}",
  "table.include.list": "{{ LIST_OF_TABLES_TO_REPLICATE }}",
  "connector.class": "io.debezium.connector.sqlserver.SqlServerConnector",
  "database.port": "1433",
  "tombstones.on.delete": "false",
  "database.trustServerCertificate": true,
  "schema.history.internal.kafka.bootstrap.servers": "...",
  "schema.history.internal.kafka.topic": "{{ CONNECTOR_NAME }}.schemahistory",
  "schema.history.internal.consumer.sasl.jaas.config": "...",
  "schema.history.internal.consumer.sasl.mechanism": "GSSAPI",
  "schema.history.internal.consumer.security.protocol": "SASL_SSL",
  "schema.history.internal.producer.sasl.jaas.config": "...",
  "schema.history.internal.producer.sasl.mechanism": "GSSAPI",
  "schema.history.internal.producer.security.protocol": "SASL_SSL",
  "heartbeat.interval.ms": 3600000,
  "key.converter": "io.confluent.connect.avro.AvroConverter",
  "key.converter.schema.registry.url": "https://...k8s.raiffeisen.ru",
  "value.converter": "io.confluent.connect.avro.AvroConverter",
  "value.converter.schema.registry.url": "https://...k8s.raiffeisen.ru",
  "transforms": "SetSchemaName",
  "transforms.SetSchemaName.type": "org.apache.kafka.connect.transforms.SetSchemaMetadata$Value",
  "transforms.SetSchemaName.schema.name": "ru.raif.sdp.{{ SHORT_TEAM_NAME }}.{{ TABLE_SCHEMA_NAME }}.{{ TABLE_NAME }}",

  "topicsToCreate": "{{ TOPICS_NAMES }}",
  "topicsToCreate.$TOPIC_NAME.partitions.count": "{{ TOPIC_PARTITIONS_COUNT }}",
  "topicsToCreate.$TOPIC_NAME.configs": "{{ TOPIC_CONFIGS }}",
  "topicsToCreate.$TOPIC_NAME.mim-roles": "{{ MIM_ROLES }}",
  "topicsToCreate.$TOPIC_NAME.description": "{{ DESCRIPTION }}",
  "topicsToCreate.$TOPIC_NAME.business-owner": "{{ BUSINESS_OWNER }}",
  "topicsToCreate.$TOPIC_NAME.source-name": "{{ SOURCE_NAME }}",
  "topicsToCreate.$TOPIC_NAME.source-ail-id": "{{ SOURCE_AIL_ID }}"
}
```

# Как сделать команды самостоятельными?



Обеспечить низкий порог входа



Решить общие проблемы



Сделать данные доступными



# Как сделать команды самостоятельными?

Решить общие проблемы – мониторинг

```
alertRules:  
  enabled: true  
  general:  
    forTeam: "atm-product"  
    defaultNotification:  
      slack: "~mon-bank-points-info"  
      chat: "rbru_L2_ATM..._Telegram"  
  
rules:  
  connectorHasFailedTask:  
    enabled: true  
    alerts:  
      - connectorRegEx: jdbc_snapshot-vv-v_cashout_.*  
        reactionTime: 24h  
        resolveTime: 24h  
        severity: warning
```



# Как сделать команды самостоятельными?

Решить общие проблемы – мониторинг




```
alertRules:  
  enabled: true
```

```
rules:  
  connectorHasFailedTask:  
    # ...  
  connectorHasNoTasks:  
    # ...  
  kafkaSdpTopicLag:  
    # ...  
  kafkaSdpTopicHasNoNewMessages:  
    # ...  
  kafkaSdpDlqTopics:  
    # ...
```



# Как сделать команды самостоятельными?

Решить общие проблемы – безопасность

1. Кто может читать топики по умолчанию? - никто
2. Где надо закрыть доступ к топику? - в UI (AKHQ), в Kafka
3. Как получить доступ в UI? - Получить одну из ролей:
  -  Data Owner;
  -  Sink Owner;
  -  Read-only (согласует Data Owner)
4. Как получить доступ в Kafka? - Попросить дежурного прописать ACL



# Как сделать команды самостоятельными?

Решить общие проблемы – безопасность

```
{{ kafka_install_dir }}/kafka-main/bin/kafka-acls.sh \  
  --bootstrap-server {% for kafka_host in kafka_cluster_hosts %}  
{{ kafka_host }}:9092{% if not loop.last %},{% endif %}{% endfor %} \  
  --command-config /opt/kafka/command-config.properties \  
  --add \  
  --allow-principal User:srv-p-kafka-sdp-voic \  
  --producer \  
  --resource-pattern-type prefixed \  
  --topic 'avaya.'
```



# Как сделать команды самостоятельными?

Решить общие проблемы – безопасность

```
{{ kafka_install_dir }}/kafka-main/bin/kafka-acls.sh \  
  --bootstrap-server {% for kafka_host in kafka_cluster_hosts %}  
{{ kafka_host }}:9092{% if not loop.last %},{% endif %}{% endfor %} \  
  --command-config /opt/kafka/command-config.properties \  
  --add \  
  --allow-principal User:srv-p-kafka-sdp-voic \  
  --producer \  
  --resource-pattern-type prefixed \  
  --topic 'avaya.'
```



# Как сделать команды самостоятельными?

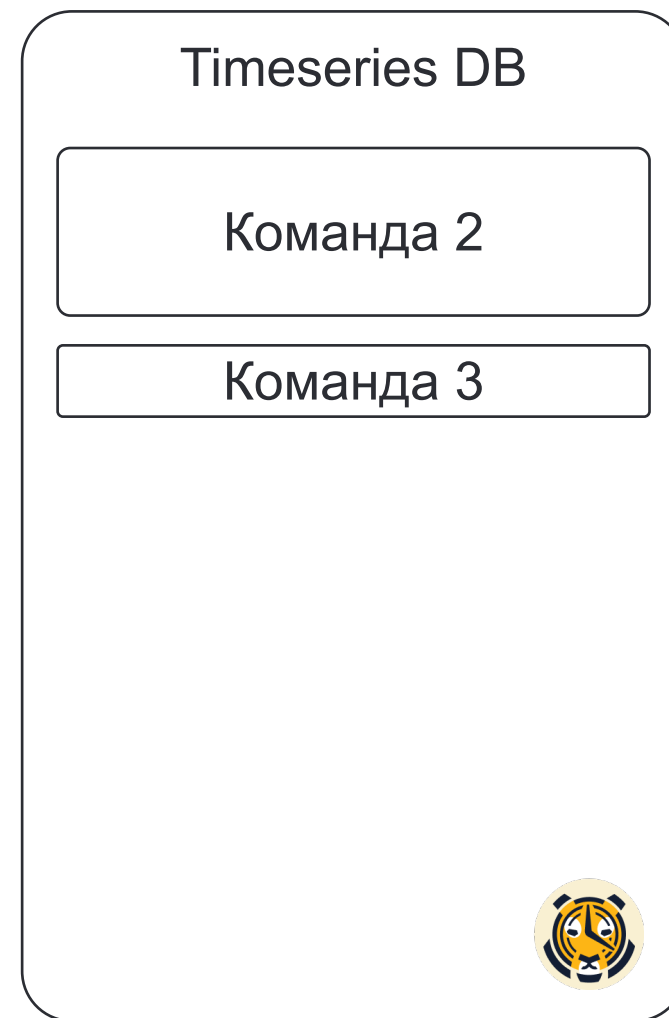
Решить общие проблемы – безопасность

```
{{ kafka_install_dir }}/kafka-main/bin/kafka-acls.sh \  
  --bootstrap-server {% for kafka_host in kafka_cluster_hosts %}  
{{ kafka_host }}:9092{% if not loop.last %},{% endif %}{% endfor %} \  
  --command-config /opt/kafka/command-config.properties \  
  --add \  
  --allow-principal User:srv-p-kafka-sdp-voic \  
  --producer \  
  --resource-pattern-type prefixed \  
  --topic 'avaya.'
```



# Как сделать команды самостоятельными?

Решить общие проблемы – инфраструктура



# Как сделать команды самостоятельными?

Решить общие проблемы – деплой



## 1. Заполни шаблон

- По инструкции в Confluence
- Большинство переменных предзаполнено



## 2. Пройди валидации

- Автоматические валидации
- Ревью дежурного



## 3. Посмотри статус своих пайплайнов в UI

- open source Kafka UI - AKHQ



# Наше решение можно сделать лучше

1. Создать коннектор

Вся работа в 1 репозитории



# Наше решение можно сделать лучше

1. Создать коннектор
2. Запустить приложение - дедупликатор

Вся работа в **2** репозиториях





# Наше решение можно сделать лучше

1. Создать коннектор
2. Запустить приложение - дедупликатор
3. Закрыть доступ к данным ролью

Вся работа в **3** репозиториях



# Наше решение можно сделать лучше

1. Создать коннектор
2. Запустить приложение - дедупликатор
3. Закрыть доступ к данным ролью
4. Настроить алерты

Вся работа в **4** репозиториях



# Наше решение можно сделать лучше

1. Создать коннектор
2. Запустить приложение - дедупликатор
3. Закрыть доступ к данным ролью
4. Настроить алерты
5. Запросить доступ на UI

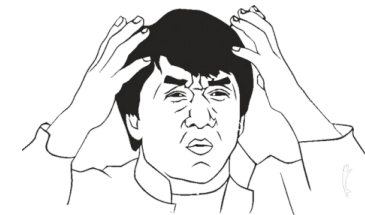
Вся работа в **4** репозиториях  
и 1 UI



# Наше решение можно сделать лучше

1. Создать коннектор
2. Запустить приложение - дедупликатор
3. Закрыть доступ к данным ролью
4. Настроить алерты
5. Запросить доступ на UI
6. Повторить на проде

Вся работа в **4** репозиториях  
и 1 UI



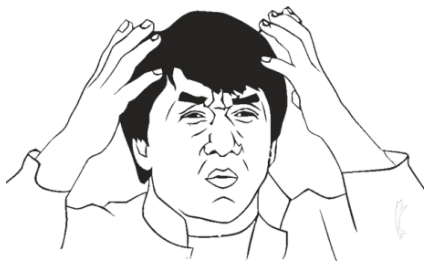


# Как сделать лучше?

Коннекторы

Права

Алерты



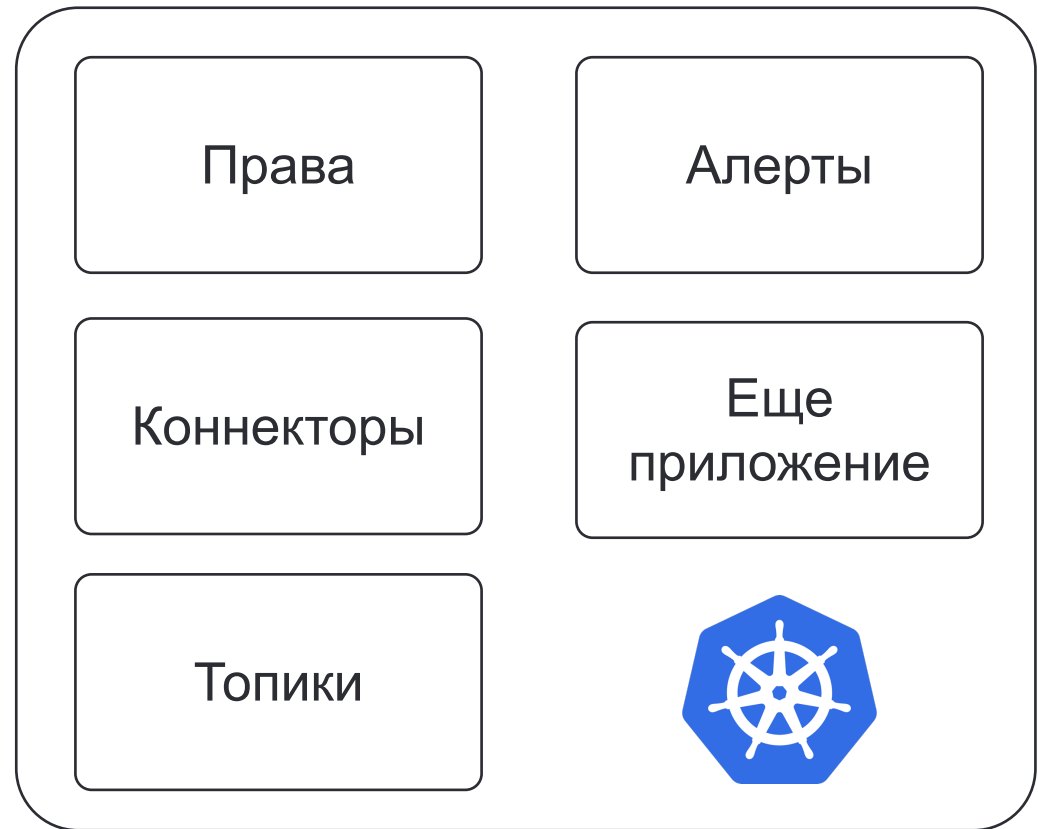
Топики

Еще приложение



# Как сделать лучше?

```
source: postgres  
db_address: 1.2.3.4  
tables: table1  
alerts: enabled  
roles: Role1
```





# Как сделать лучше?

Деплой через Strimzi (коннекторы)

```
apiVersion: kafka.strimzi.io/v1beta2
kind: KafkaConnector
metadata:
  name: my-source-connector
  labels:
    strimzi.io/cluster: my-connect-cluster
spec:
  class: org.apache.kafka.connect.file.FileStreamSourceConnector
  tasksMax: 2
  config:
    file: "/opt/kafka/LICENSE"
    topic: my-topic (7)
    # ...
```





# Как сделать лучше?

Деплой через Strimzi (топики)

```
apiVersion: kafka.strimzi.io/v1beta2
kind: KafkaTopic
metadata:
  name: my-topic
  labels:
    strimzi.io/cluster: my-cluster
spec:
  partitions: 1
  replicas: 1
  # ...
```



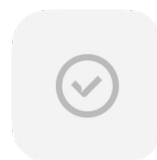




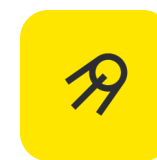
# Как сделать команды самостоятельными?



Обеспечить низкий порог входа



Решить общие проблемы



Сделать данные доступными



# Как сделать команды самостоятельными?

Сделать данные доступными – Data Catalog



Написать свой

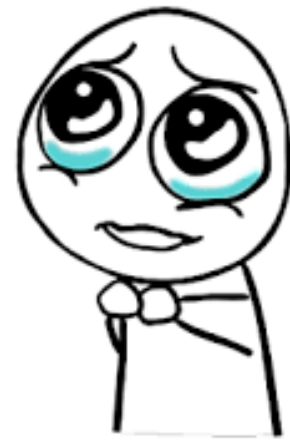
**А теперь команды будут  
счастливы?**



# Что еще можно добавить в платформу?

Рубрика вопросы от наших пользователей

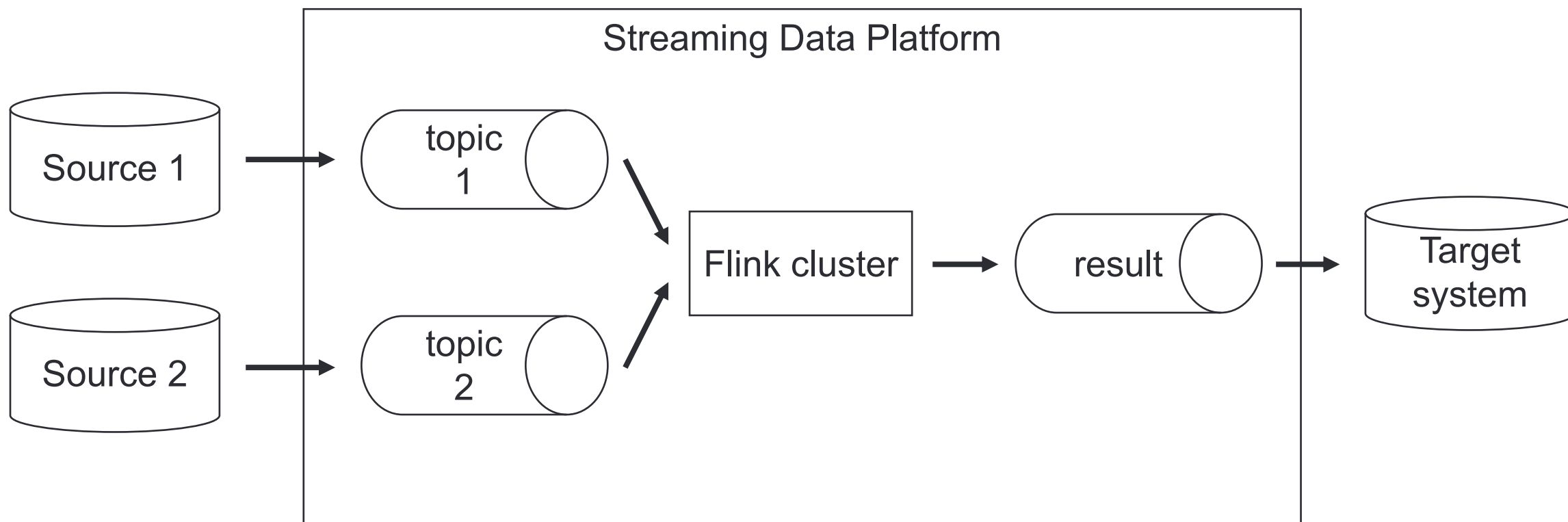
1. Эти данные нас не устраивают, можно мы их поменяем под себя?
2. Не хочу раскрывать детали своей БД, как их скрыть?
3. Мне бы объединить между собой данные из четырех баз в реальном времени
4. Как я могу эти данные вывести на график?





# Что еще можно добавить в платформу?

Преобразования над данными



# Как выглядит SQL?

```
insert into turbo_rep_assets
select R.counterparty_code as COUNTERPARTY_CODE,
       R.rep_asset_type as ASSET_TYPE,
       ...
       R.rep_asset_description as ASSET_DESCRIPTION,
from turbo_rfb_buh_ric R
inner join ods_dm_lv_lcif_map_id E
on R.counterparty_code = E.EXT_ID and E.EXT_SYSTEM = 'MIDAS'
inner join ri_s_org_ext_x_view SOX
on SO.ROW_ID=SOX.ROW_ID
where 1=1
and SO.X_CIF_ID is not null and SOX.X_RI_WO_STATUS = 'В работе'
and R.reporting_date =
      (select max(reporting_date) from turbo_rfb_buh_ric)
;
```



# Как выглядит SQL?

## SELECT

```
insert into turbo_rep_assets
select R.counterparty_code as COUNTERPARTY_CODE,
       R.rep_asset_type as ASSET_TYPE,
       ...
       R.rep_asset_description as ASSET_DESCRIPTION,
from turbo_rfb_buh_ric R
inner join ods_dm_lv_lcif_map_id E
on R.counterparty_code = E.EXT_ID and E.EXT_SYSTEM = 'MIDAS'
inner join ri_s_org_ext_x_view SOX
on SO.ROW_ID=SOX.ROW_ID
where 1=1
and SO.X_CIF_ID is not null and SOX.X_RI_WO_STATUS = 'В работе'
and R.reporting_date =
      (select max(reporting_date) from turbo_rfb_buh_ric)
;
```



# Как выглядит SQL?

## JOINS

```
insert into turbo_rep_assets
select R.counterparty_code as COUNTERPARTY_CODE,
       R.rep_asset_type as ASSET_TYPE,
       ...
       R.rep_asset_description as ASSET_DESCRIPTION,
from turbo_rfb_buh_ric R
inner join ods_dm_lv_lcif_map_id E
on R.counterparty_code = E.EXT_ID and E.EXT_SYSTEM = 'MIDAS'
inner join ri_s_org_ext_x_view SOX
on SO.ROW_ID=SOX.ROW_ID
where 1=1
and SO.X_CIF_ID is not null and SOX.X_RI_WO_STATUS = 'В работе'
and R.reporting_date =
      (select max(reporting_date) from turbo_rfb_buh_ric)
;
```





# Как выглядит SQL?

## WHERE

```
insert into turbo_rep_assets
select R.counterparty_code as COUNTERPARTY_CODE,
       R.rep_asset_type as ASSET_TYPE,
       ...
       R.rep_asset_description as ASSET_DESCRIPTION,
from turbo_rfb_buh_ric R
inner join ods_dm_lv_lcif_map_id E
on R.counterparty_code = E.EXT_ID and E.EXT_SYSTEM = 'MIDAS'
inner join ri_s_org_ext_x_view SOX
on SO.ROW_ID=SOX.ROW_ID
where 1=1
and SO.X_CIF_ID is not null and SOX.X_RI_WO_STATUS = 'В работе'
and R.reporting_date =
      (select max(reporting_date) from turbo_rfb_buh_ric)
;
```



# Как выглядит SQL?

## Подзапросы

```
insert into turbo_rep_assets
select R.counterparty_code as COUNTERPARTY_CODE,
       R.rep_asset_type as ASSET_TYPE,
       ...
       R.rep_asset_description as ASSET_DESCRIPTION,
from turbo_rfb_buh_ric R
inner join ods_dm_lv_lcif_map_id E
on R.counterparty_code = E.EXT_ID and E.EXT_SYSTEM = 'MIDAS'
inner join ri_s_org_ext_x_view SOX
on SO.ROW_ID=SOX.ROW_ID
where 1=1
and SO.X_CIF_ID is not null and SOX.X_RI_WO_STATUS = 'В работе'
and R.reporting_date =
    (select max(reporting_date) from turbo_rfb_buh_ric)
;
```

# Как выглядит SQL?

## INSERT

```
insert into turbo_rep_assets
```

```
select R.counterparty_code as COUNTERPARTY_CODE,  
       R.rep_asset_type as ASSET_TYPE,
```

```
...
```

```
R.rep_asset_description as ASSET_DESCRIPTION,
```

```
from turbo_rfb_buh_ric R
```

```
inner join ods_dm_lv_lcif_map_id E
```

```
on R.counterparty_code = E.EXT_ID and E.EXT_SYSTEM = 'MIDAS'
```

```
inner join ri_s_org_ext_x_view SOX
```

```
on SO.ROW_ID=SOX.ROW_ID
```

```
where 1=1
```

```
and SO.X_CIF_ID is not null and SOX.X_RI_WO_STATUS = 'В работе'
```

```
and R.reporting_date =
```

```
    (select max(reporting_date) from turbo_rfb_buh_ric)
```

```
;
```



# Как выглядит SQL?

INSERT в новый Kafka topic

```
CREATE TABLE turbo_rep_assets (  
  COUNTERPARTY_CODE STRING,  
  ASSET_TYPE STRING,  
  ...  
  ASSET_DESCRIPTION STRING  
) with (  
  'connector' = 'upsert-kafka',  
  'topic' = 'flink.test.turbo_rep_assets',  
  'properties.bootstrap.servers' = '...'  
);
```

```
insert into turbo_rep_assets  
...;
```



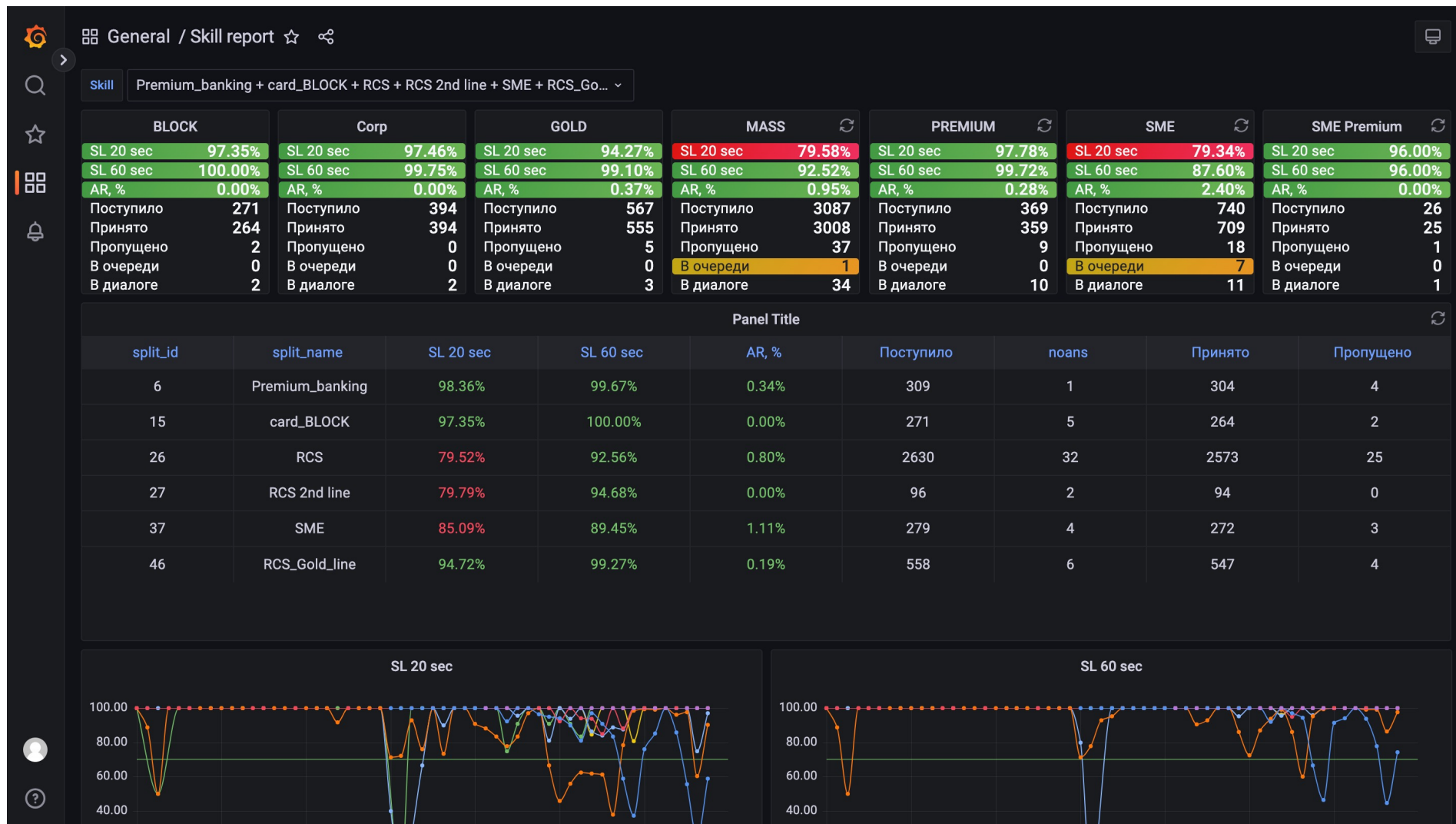
# Можно и без join-ов

Кейс справочника стран

```
INSERT INTO countries_out_state
SELECT inp.D_FROM AS D_FROM,
       CASE WHEN inp.DELETED = 0 THEN FALSE
            ELSE TRUE END as DELETED,
       CAST(inp.CORRUPTION_CRIMES AS BIGINT) AS CORRUPTION_CRIMES,
       ...
FROM countries_in_state inp;
```

# Что еще можно добавить в платформу?


Меняющиеся в real time графики





# Что еще можно добавить в платформу?

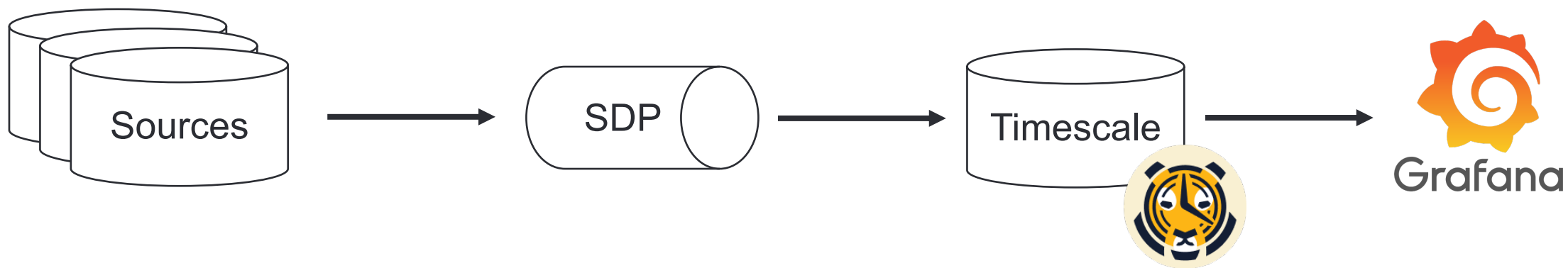
Меняющиеся в real time графики

MASS 	
SL 20 sec	79.58%
SL 60 sec	92.52%
AR, %	0.95%
Поступило	3087
Принято	3008
Пропущено	37
В очереди	1
В диалоге	34



# Что еще можно добавить в платформу?

Меняющиеся в real time графики



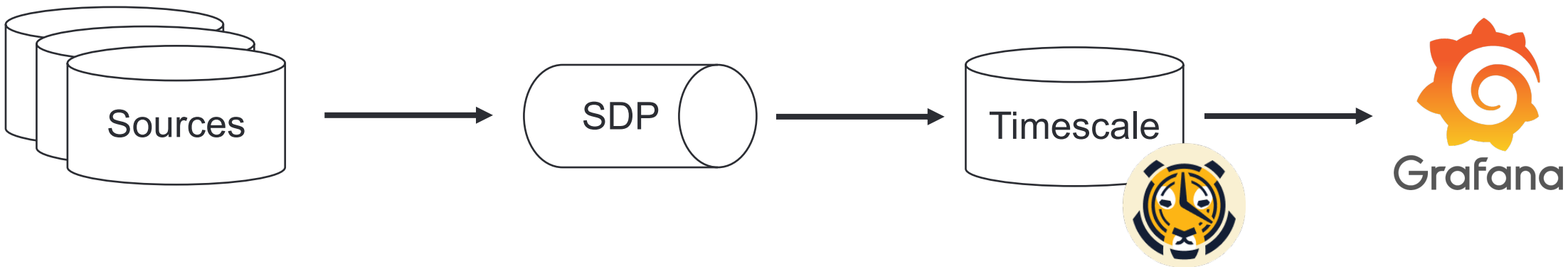
\* Timescale – это time series база на основе Postgres, которая быстро работает с данными, у которых есть timestamp





# Как это работает?

Меняющиеся в real time графики



ClickHouse  druid  ?  
influxdb  [ROCKSET] 



# Как выбрать time series базу?

## Наш алгоритм

high availability	2 - есть 1 - нужно проверять
Retention policy	1 - да 0 - нет
Загрузка данных	2 - нашими коннекторами 1 - есть коннекторы/jdbc драйвер, нужно проверять 0 - другими средствами
Интеграция с BI инструментами	2 - нативная 1 - нужны плагины/драйверы
Открытая лицензия	1 - да 0 - нет
Поддержка SQL	3 - нативный SQL 2 - близкий к нативному 1 - собственный диалект 0 - нет
Ролевая модель	2 - есть 1 - нужно проверять



 ClickHouse

 druid

 [ROCKSET]

 influxdb





# О чем еще не забыть?

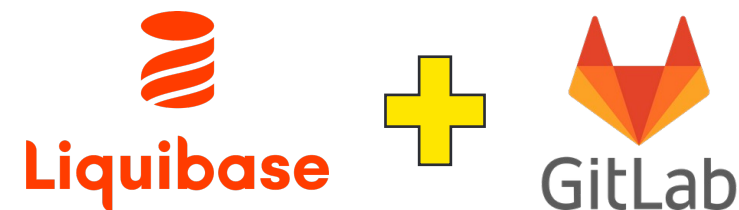
Меняющиеся в real time графики

1. Как пользователям задать индексы к своим данным?
2. Как указать retention на хранение?
3. Алерты на объем места и скорость работы запросов

# О чем еще не забыть?

Меняющиеся в real time графики

1. Как пользователям задать индексы к своим данным?
2. Как указать retention на хранение?
3. Алерты на объем места и скорость работы запросов



```
CREATE TABLE cca.avaya_cagent (  
    ...  
);
```

```
SELECT add_retention_policy(relation => 'cca.avaya_cagent',  
drop_after => INTERVAL '48 hours');
```



**Но теперь то все  
ГОТОВО?**

# Что можно сделать дальше?



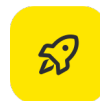
## Улучшать инструменты

- Деплой
- UI
- Автоматические валидации
- Простой дебаг



## Streaming ML Ops

- Стандартные функции
- Real time feature store



## Новые источники

- NoSQL databases
- JMS
- Vendor: Jira, Amplitude, ...



## Triggers

- Отправить email/push
- Запустить расчет
- Отправить алерт



# Спасибо!

<https://t.me/aserebryanskiy>

[a.serebrianskiy@gmail.com](mailto:a.serebrianskiy@gmail.com)

