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

USENIX Large Installation System Administration Conference (LISA)
October 31 2018
$ whoami

- Developer Advocate @ Confluent
- Working in data & analytics since 2001
- Oracle Developer Champion 🏆
- Twitter: @rmoff
  - Geek stuff
  - Beer & Fried Breakfasts

https://speakerdeck.com/rmoff/
Kafka
Kafka is a Streaming Platform

request-response
messaging
stream processing
streaming data pipelines

changelogs

Apache Kafka and KSQL in Action: Let's Build a Streaming Data Pipeline!
Streaming is not just for realtime
Streaming is for everyone
All data is events
A Dumb Pipeline

Logs -> HDFS / S3 / BigQuery etc
A Dumb Pipeline

Logs → HDFS / S3 / BigQuery etc

Apache Kafka and KSQL in Action: Let's Build a Streaming Data Pipeline!
Stream Processing with Apache Kafka and KSQL

- Logs
- All logs
- Errors
- HDFS / S3 / BigQuery etc
- Stream Processing

Apache Kafka and KSQL in Action: Let's Build a Streaming Data Pipeline!
Real-time Event Stream Enrichment

order events

CDC

customer

RDBMS

Stream Processing

customer orders

Apache Kafka and KSQL in Action: Let's Build a Streaming Data Pipeline!
Transform Once, Use Many

order events

RDBMS

customer

CDC

customer orders

Stream Processing

New App

Apache Kafka and KSQL in Action: Let’s Build a Streaming Data Pipeline!
Transform Once, Use Many

- order events
- customer
- CDC
- customer orders
- HDFS / S3 / etc
- RDBMS
- Stream Processing

Apache Kafka and KSQL in Action: Let's Build a Streaming Data Pipeline!
Let’s Build It!

Rating events

Join events to users, and filter

Push notification

Operational Dashboard

Data Lake

user data
Let’s Build It!

Join events to users, and filter

Rating events

Push notification

App

Consumer API

Operational Dashboard

Elasticsearch

Data Lake

S3/HDFS/Snowflake DB etc

User data

RDBMS

Let’s Build a Streaming Data Pipeline!
Kafka Connect

Join events to users, and filter

Kafka Connect

S3/HDFS/Snowflake DB etc

Rating events

Producer API

Consumers API

App

Push notification

Operational Dashboard

Data Lake

Elasticsearch

RDBMS

User data
Kafka Connect

An API of Apache Kafka, providing reliable and scalable integration of Kafka with other systems – no coding required.

```json
{
    "connector.class": "io.confluent.connect.jdbc.JdbcSourceConnector",
    "connection.url": "jdbc:mysql://localhost:3306/demo?user=rmoff&password=foo",
    "table.whitelist": "sales,orders,customers"
}
```

https://docs.confluent.io/current/connect/
Streaming Integration with Kafka Connect

Sources

Kafka Connect

Kafka Brokers

syslog
flat file
CSV
JSON
MQTT

Kafka Brokers

Tasks

workers
Streaming Integration with Kafka Connect

Kafka Connect

Kafka Brokers

Sinks

Tasks

workers

Amazon S3

MQT

elasticsearch

influxdb

Google BigQuery

Amazon S3

ORACLE

MQT
Streaming Integration with Kafka Connect

Sources

Tasks

Kafka Connect

Kafka Brokers

Workers

Sinks

Amazon S3

MQTT
Confluent Hub

• One-stop place to discover and download:
  • Connectors
  • Transformations
  • Converters

hub.confluent.io
Kafka Connect + Schema Registry = WIN
Kafka Connect + Schema Registry = WIN

```
mysql> describe rental;
+----------------+-----------------------------------+
| Field          | Type     |
+----------------+-----------------------------------+
| rental_id      | int(11)  |
| rental_date    | datetime |
| inventory_id   | mediumint(8) |
| customer_id    | int(11)  |
| return_date    | datetime |
| staff_id       | smallint(5) |
| last_update    | timestamp |
+----------------+-----------------------------------+
```

```
"mappings": {
  "type.name-kafka-connect": {
    "dynamic_templates": [],
    "properties": {
      "customer_id": {
        "type": "integer"
      },
      "inventory_id": {
        "type": "long"
      },
      "last_update": {
        "type": "date"
      },
      "rental_date": {
        "type": "date"
      },
      "rental_id": {
        "type": "integer"
      },
      "return_date": {
        "type": "date"
      }
    }
  }
}
```
curl -X "POST" "http://kafka-connect-cp:18083/connectors/" \
-H "Content-Type: application/json" \
-d '{
    "name": "es_sink_lisa18",
    "config": {
        "connector.class": "io.confluent.connect.elasticsearch.ElasticsearchSinkConnector",
        "key.converter": "org.apache.kafka.connect.storage.StringConverter",
        "value.converter": "org.apache.kafka.connect.json.JsonConverter",
        "value.converter.schemas.enable": false,
        "topics": "lisa18",
        "key.ignore": "true",
        "schema.ignore": "true",
        "type.name": "type.name=kafkaconnect",
        "connection.url": "http://elasticsearch:9200"
    }
}'
Demo Time!

Apache Kafka and KSQL in Action: Let’s Build a Streaming Data Pipeline!
Let’s Build It!

Join events to users, and filter

Push notification

App

Consumer API

Producer API

Kafka Connect

RDBMS

user data

Elasticsearch

App

Operational Dashboard

Data Lake

S3/HDFS/Snowflake DB etc

Rating events

Rating events

Let’s Build a Streaming Data Pipeline!
Join events to users, and filter

Rating

Push notification

App

Consumer API

Operational Dashboard

Elasticsearch

Kafka Connect

S3/HDFS/
Snowflake DB

Data Lake

RDBMS

user data

App

Producer API

Kafka Connect

Kafka Connect

Kafka Connect

Rating events

Apache Kafka and KSQL in Action: Let's Build a Streaming Data Pipeline!
KSQQL is a Declarative Stream Processing Language
KSQIL is the Streaming SQL Engine for Apache Kafka
KSQL for Real-Time Monitoring

- Log data monitoring, tracking and alerting
- syslog data
- Sensor / IoT data

CREATE STREAM SYSLOG_INVALID_USERS AS
SELECT HOST, MESSAGE
FROM SYSLOG
WHERE MESSAGE LIKE '%Invalid user%';

KSQL for Streaming ETL

Joining, filtering, and aggregating streams of event data

```
CREATE STREAM vip_actions AS
    SELECT userid, page, action
    FROM clickstream c
    LEFT JOIN users u
    ON c.userid = u.user_id
    WHERE u.level = 'Platinum';
```
KSQL for Anomaly Detection

Identifying patterns or anomalies in real-time data, surfaced in milliseconds

```
CREATE TABLE possible_fraud AS
    SELECT card_number, count(*)
    FROM authorization_attempts
    WINDOW TUMBLING (SIZE 5 SECONDS)
    GROUP BY card_number
    HAVING count(*) > 3;
```
KSQL for Data Transformation

Make simple derivations of existing topics from the command line

CREATE STREAM pageviews
WITH (PARTITIONS=4,
    VALUE_FORMAT='AVRO') AS
SELECT * FROM pageviews_json;
KSQL in Development and Production

Interactive KSQL for development and testing

Headless KSQL for Production

Desired KSQL queries have been identified

"Hmm, let me try out this idea..."
Demo Time!

Apache Kafka and KSQL in Action: Let's Build a Streaming Data Pipeline!
CREATE STREAM POOR_RATINGS AS
SELECT * FROM ratings WHERE STARS < 3
Do you think that’s a table you are querying?
### The Stream Table Duality

#### Stream

<table>
<thead>
<tr>
<th>Account ID</th>
<th>Amount</th>
</tr>
</thead>
<tbody>
<tr>
<td>12345</td>
<td>+ €50</td>
</tr>
<tr>
<td>12345</td>
<td>+ €25</td>
</tr>
<tr>
<td>12345</td>
<td>-€60</td>
</tr>
</tbody>
</table>

#### Table

<table>
<thead>
<tr>
<th>Account ID</th>
<th>Balance</th>
</tr>
</thead>
<tbody>
<tr>
<td>12345</td>
<td>€50</td>
</tr>
<tr>
<td>12345</td>
<td>€75</td>
</tr>
<tr>
<td>12345</td>
<td>€15</td>
</tr>
</tbody>
</table>

Read more: https://cnfl.io/stream-table-duality
The truth is the log.
The database is a cache of a subset of the log.

—Pat Helland
Immutability Changes Everything
```sql
CREATE STREAM RATINGS_WITH_CUSTOMER_DATA AS
SELECT *
FROM RATINGS
LEFT JOIN CUSTOMERS
ON R.ID=C.ID;
```
Apache Kafka and KSQL in Action: Let's Build a Streaming Data Pipeline!

Join each rating to customer data

Filter for just PLATINUM customers

CREATE STREAM UNHAPPY_PLATINUM_CUSTOMERS AS
SELECT * FROM RATINGS_WITH_CUSTOMER_DATA
WHERE STARS < 3
Confluent Open Source: Apache Kafka with a bunch of cool stuff! For free!

Database Changes ➔ Log Events ➔ IoT Data ➔ Web Events ➔ ...

**Confluent Platform**

- Monitoring & Administration
  - Confluent Control Center | Security
- Operations
  - Data Integration | Real-time Applications
- Development and Connectivity
  - Hadoop | Database | Data Warehouse | CRM
- Data Compatibility
  - Schema Registry
- SQL Stream Processing
  - KSQL
- Apache Kafka®
  - Core | Connect API | Streams API

**CUSTOMER SELF-MANAGED**
- Datacenter
- Public Cloud

**CONFLUENT FULLY-MANAGED**
- Confluent Cloud

---

Apache Kafka and KSQL in Action: Let’s Build a Streaming Data Pipeline!

@rmoff / http://cnfl.io/ksql
If you remember one thing... (or three)

• **Kafka Connect**
  • *Integration* between Kafka and other data stores

• **Kafka**
  • Provides *stream processing* natively

• **KSQL**
  • Build stream processing apps with *just SQL*
Free Books!

Try it out!

https://cnfl.io/kafka-ksql-elastic
Useful links

- Embrace the Anarchy: Apache Kafka's Role in Modern Data Architectures Recording & Slides
- Look Ma, no Code! Building Streaming Data Pipelines with Apache Kafka and KSQL
- Steps to Building a Streaming ETL Pipeline with Apache Kafka and KSQL Recording & Slides
- https://github.com/confluentinc/ksql/
#EOF