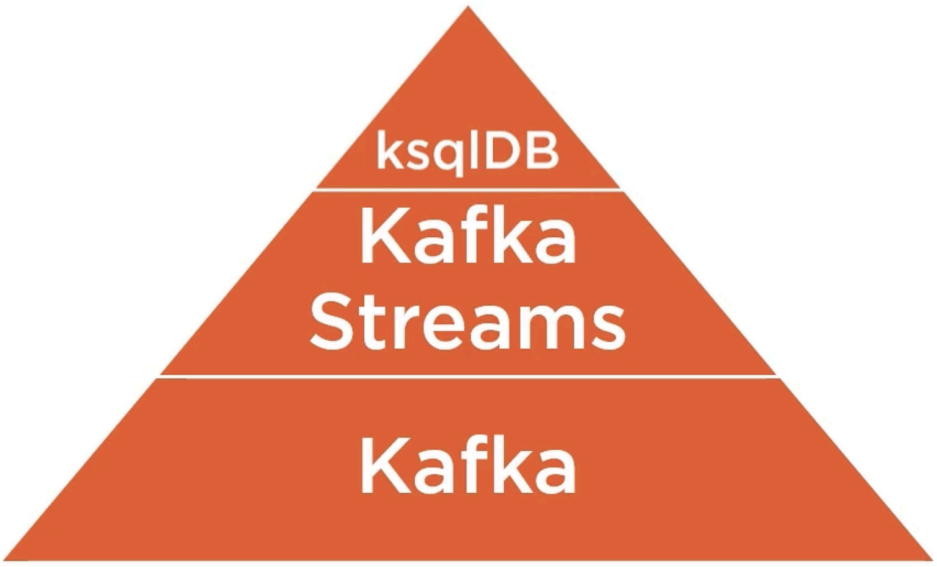
Overview



Kafka is an event streaming platform

Aggregating Data

Aggregating Data

Aggregating Data

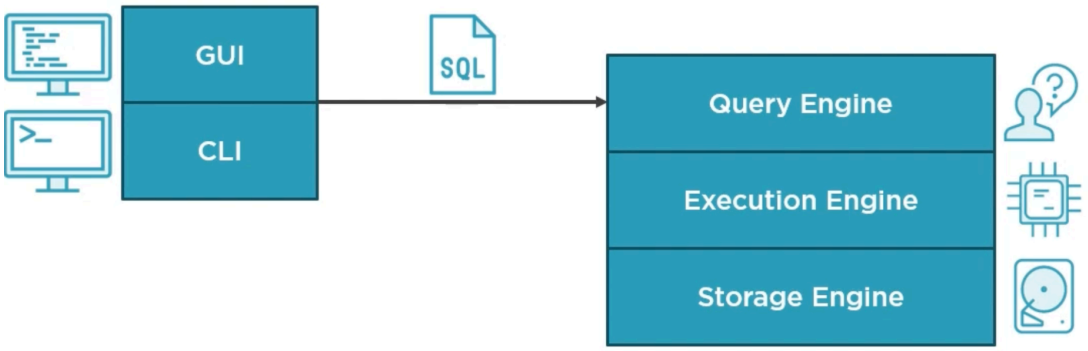
Aggregating Data

Processor Topology

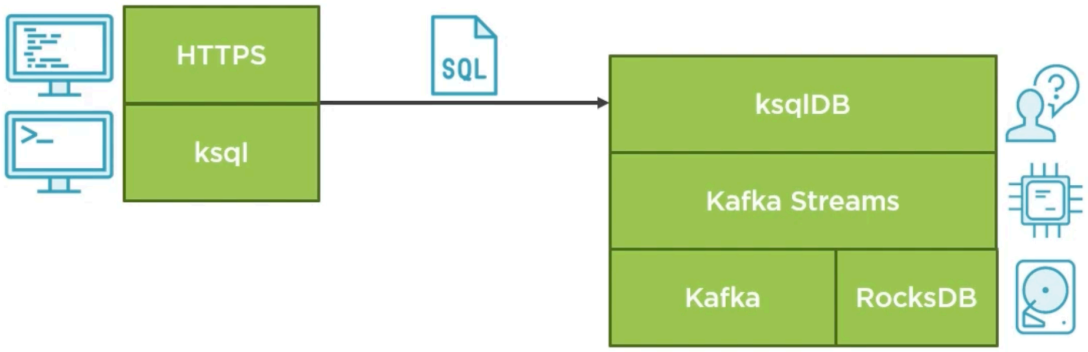
"ksqlDB is an event streaming database purpose-built to help developers create stream processing applications on top of Apache Kafka."

Confluent

What Is a Database?



What Is a Database?



What Is KsqlDB?

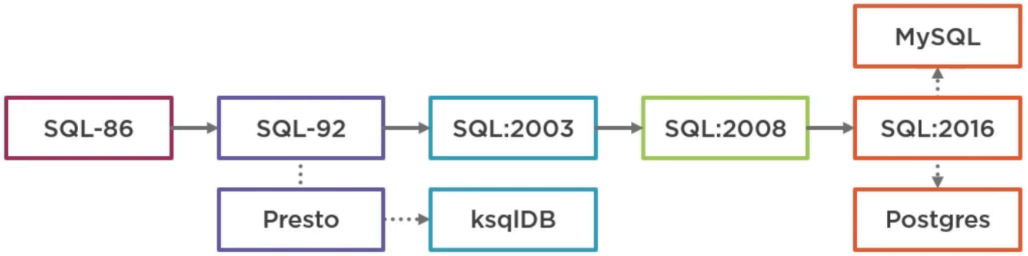
SQL Variant

Command Line Interface

Web Server

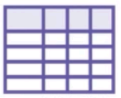
Top Layer

SQL Family Tree



Key Differences

Table



Stream

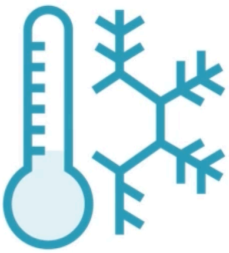
Modify

Append

Pull Query

Push Query

Weather Analysis



Hundreds of weather sensors

Thousands of readings

Stored in Kafka

Analysis with ksqlDB

Types of Queries

Push Query

Pull Query

Joins

Push Query

```
CREATE STREAM tempReadings (zipcode VARCHAR, sensortime BIGINT, temp DOUBLE)
  WITH (kafka_topic='readings', timestamp='sensortime', value_format='json',
  partitions=1);

SELECT zipcode, TIMESTAMPTOSTRING(WINDOWSTART, 'HH:mm:ss') AS windowtime,
       AVG(temp) as temp
FROM tempReadings
WINDOW TUMBLING (SIZE 1 HOURS)
GROUP BY zipcode EMIT CHANGES;
```

Pull Query

```
CREATE TABLE highsandlows WITH (kafka_topic='readings') AS
    SELECT MIN(temp) as min_temp, MAX(temp) as max_temp, zipcode
    FROM tempReadings GROUP BY zipcode;

SELECT min_temp, max_temp, zipcode
FROM highsandlows
WHERE zipcode = 25005
```

Join Query

```
SELECT tempReadings2.temp, CASE
        WHEN tempReadings2.temp - highsandlows.min_temp <= 5 THEN 'Low'
        WHEN highsandlows.max_temp - tempReadings2.temp <= 5 THEN 'High'
        ELSE 'Normal' END AS classification
   FROM tempreadings2
   LEFT JOIN highsandlows ON tempReadings2.zipcode = highsandlows.zipcode
   EMIT CHANGES;
```

Queries:

Deploying in Headless Mode

Kafka Connect

Queries File

Command Line Parameter (--queries-file)

Kafka Client

```
SET 'auto.offset.reset'='earliest';

-- Streams
SHOW ALL TOPICS;

CREATE STREAM tempReadings (zipcode VARCHAR, sensortime BIGINT, temp DOUBLE)
 WITH (kafka_topic='readings', timestamp='sensortime', value_format='json',
partitions=1);

SHOW TOPICS EXTENDED;
SHOW STREAMS EXTENDED;
```

```
INSERT INTO tempReadings (zipcode, sensortime, temp) VALUES ('25005',
UNIX_TIMESTAMP(), 40);
INSERT INTO tempReadings (zipcode, sensortime, temp) VALUES ('25005',
UNIX_TIMESTAMP(), 50);
INSERT INTO tempReadings (zipcode, sensortime, temp) VALUES ('25005',
UNIX_TIMESTAMP() + 60 * 60 * 1000, 60);


SELECT zipcode, TIMESTAMPTOSTRING(WINDOWSTART, 'HH:mm:ss') as windowtime,
      COUNT(*) AS rowcount, AVG(temp) as temp
FROM tempReadings
WINDOW TUMBLING (SIZE 1 HOURS)
GROUP BY zipcode EMIT CHANGES;


--Tables
CREATE TABLE highsandlows WITH (kafka_topic='readings') AS
   SELECT MIN(temp) as min_temp, MAX(temp) as max_temp, zipcode
 FROM tempReadings GROUP BY zipcode;

SELECT min_temp, max_temp, zipcode
FROM highsandlows
WHERE zipcode = '25005';

INSERT INTO tempReadings (zipcode, sensortime, temp) VALUES ('25005',
UNIX_TIMESTAMP() + 60 * 60 * 1000, 70);

SELECT min_temp, max_temp, zipcode
FROM highsandlows
WHERE zipcode = '25005';

-- Joins

CREATE STREAM tempReadings2 (zipcode VARCHAR, sensortime BIGINT, temp DOUBLE)
WITH (kafka_topic='readings2', timestamp='sensortime', value_format='json',
partitions=1);



SELECT tempReadings2.temp, CASE
                           WHEN tempReadings2.temp - highsandlows.min_temp <= 5
THEN 'Low'
                           WHEN highsandlows.max_temp - tempReadings2.temp <= 5
THEN 'High'
```

```
                            ELSE 'Normal'
                        END AS classification
FROM tempReadings2 LEFT JOIN highsandlows ON tempReadings2.zipcode =
highsandlows.zipcode
EMIT CHANGES;

-- docker exec -it ksqldb-cli ksql http://ksqldb-server:8088

INSERT INTO tempReadings2 (zipcode, sensortime, temp) VALUES ('25005',
UNIX_TIMESTAMP(), 40);
INSERT INTO tempReadings2 (zipcode, sensortime, temp) VALUES ('25005',
UNIX_TIMESTAMP(), 50);
INSERT INTO tempReadings2 (zipcode, sensortime, temp) VALUES ('25005',
UNIX_TIMESTAMP(), 68);
```