S1 = { id = int
       m = char
       age = int 10KB }

SR

10uB

10uB

S01 4 Byte

Cache 300KB 10KB

S01 | m1   10.004KB
             10KB

P

S01 | m1

C

S1 = { id: int
       m: char
       age: int }
10KB

m1 = { K: 101
       n: Bob
       age: 30 }
       10KB

Schema   Quality   msg

S01 | m1

Broker!
Kafka cluster

S1 = { id: int
       m: char   m1 = { K: 101
       age: int } ≅   n: Bob
                          age: 30 }

zookeeper

Kafka

Python producer
Java producer
C# producer

driver-n.csv
CSV

driver-positions-avro

schema-registry

Python consumer
Java consumer
C# consumer

Kafka consumer
webserver avro

web browser

Module 6 — Schema
Module 7 — Schema-Registry

Java API   Non-Java   SQL

Broker

Kafka Stream

Confluent

IOT Sensor

Temp

Temp

| 24 | 29 | 21 | 32 | 27 |

Kafka Stream

Stream

| 24 | 29 | 21 | 32 | 27 |

→ KSQLDB

BL → ○ > 25

Temp

| | 29 | 32 | 27 |

High-Temp

P

Zoo Loc

SQL
KSQL
J-6 Lim
SQL

---

zookeeper

Kafka

D1
D2
D3
D4

driver-positions-avro

driver-positions-string-avro

Kafka consumer

webserver-streams

web browser

CSV driver-1.csv — producer1
CSV driver-2.csv — producer2
CSV driver-3.csv — producer3
CSV driver-4.csv — producer4

D1 D2 D3 D4   stream()   != D2   D2   to()   D1 D3 D4

filter()
mapValues()

KStream positions   KStream positionsString

streams-app-1

schema-registry

D1 D3 D4

Hash(key) % of Ru
Ram   0,1,2 %3

0,1,2 %24

Hash(driver)

$40\%5 = 1\ P_1$

$40\%9 = 0\ P_0$

Profile
Po. 1

Position

Extra Challenges and Questions Solutions

1. When you use ksqlDB to join streaming data, you must ensure that your streams and tables are co-partitioned. To be considered co-partitioned we look at 3 properties:

   a. Records have the same keying scheme - both topics are keyed on `driver-id`.

   b. Records have the same number of partitions - both topics have 3 partitions.

   c. Records have the same partitioning strategy - the producers and Kafka Connect are using the default murmur2 hash partitioner.

These properties guarantee that records of the same key will always be delivered to the same partition number in both topics. For example, `driver-3` will be delivered to partition 1 for both `driver-profiles-ksql` and `driver-positions-avro`.

Extermal Slim / Source

Postgres

CSV

mqtt

1) Install Connector
2) Configure
   JDBC Source Connector

Source
Connector

TOPIC-A

Source
Connector   Kafka Connect

P

C

P

C

Sevem 70ay
5 for

SINK
Connector
Quot

SS

Hadoop

ELK

Spark
Extermal
Slim

SINK
Connector

Source

Postgres

Configure

zookeeper

kafka

JDBC source
connector

driver

Kafka Connect

schema-registry

driver-profiles-avro

SI

kafka-avro-
console-
consumer

14

15+

Order-1

Upsa 14

14