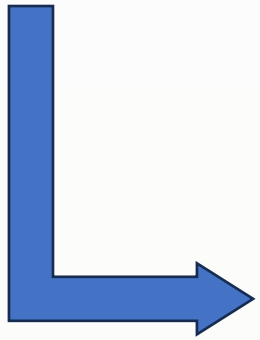
Join in Spark
Stream to static table

And a short intro to Cassandra

Spark Streaming Joins

Streaming Dataframe to Static Dataframe
Streaming Dataframe to Streaming Dataframe



Inner Joins with optional Watermarking

Outer Joins with Watermarking

Semi Joins with Watermarking

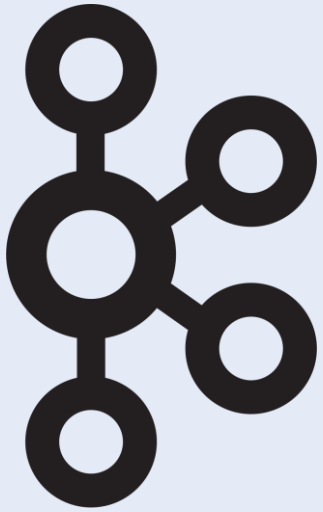
Support matrix for joins in streaming queries

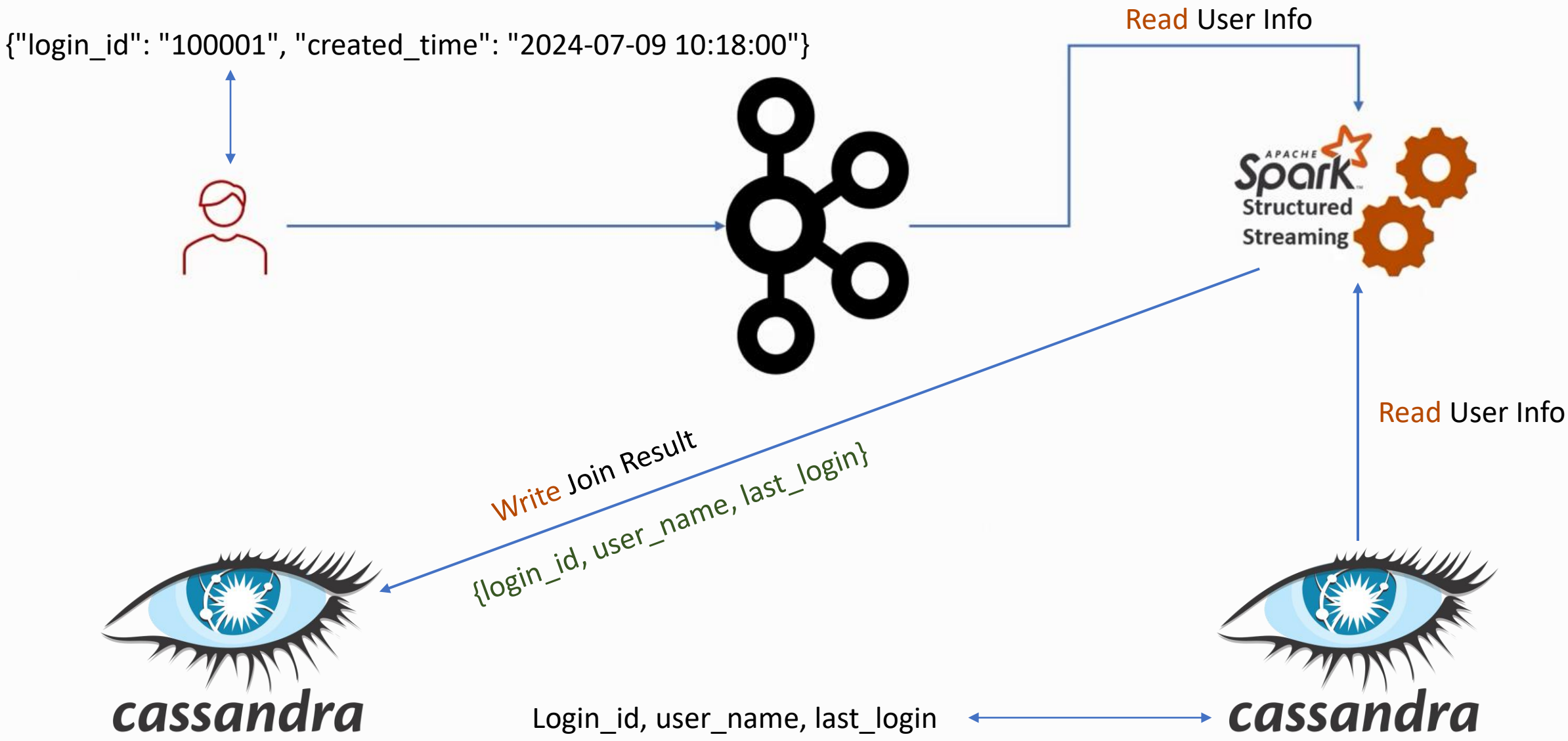
1. We have a Cassandra Database which includes all information regarding our bank users such as
   (Login_id, user_name, last_login)

2. We have a Kafka topic which includes the latest info of our users such as
   (login_id, created_time)

3. We wish our database has the latest user information to track users behaviour

   and that our concern in this video!




{"login_id": "100001", "created_time": "2024-07-09 10:18:00"}
Read User Info
APACHE Spark Structured Streaming
Read User Info
Write Join Result
{login_id, user_name, last_login}
cassandra
cassandra
Login_id, user_name, last_login
('100001', 'Ali', '2024-02-05 10:05:00')

Cassandra is an **open-source**, **distributed NoSQL** database management system designed to handle large amounts of data with the following characteristics:

- Distributed Architecture
- High Availability
- Linear Scalability
- No Single Point of Failure
- Flexible Data Model

Use Cases and Applications

- Time Series Data
- Real-Time Analytics
- Highly Available Web Applications
- Content Management Systems

Steps to create a database and table in Cassandra:

1. CREATE KEYSPACE spark_db WITH replication = {'class': 'SimpleStrategy', 'replication_factor': 1};

2. USE spark_db;

3. CREATE TABLE users(Login_id text PRIMARY KEY, user_name text, last_login timestamp);

4. INSERT INTO users (Login_id, user_name, last_login) VALUES('100001', 'Ali', '2024-02-05 10:05:00');

- CREATE KEYSPACE is similar to creating databases in RDBMS
- USE will use of the specified keyspace

Let’s go for coding!