CADEC 2018
KAFKA IN THE STREAM

TORBJÖRN CLAESSON

CADEC 2018.01.24 |
CALLISTAENTERPRISE.SE

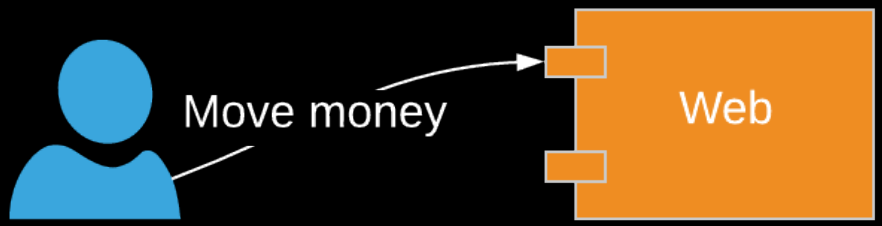
CALLISTA

— ENTERPRISE —

PARADIGMS OF PROGRAMMING

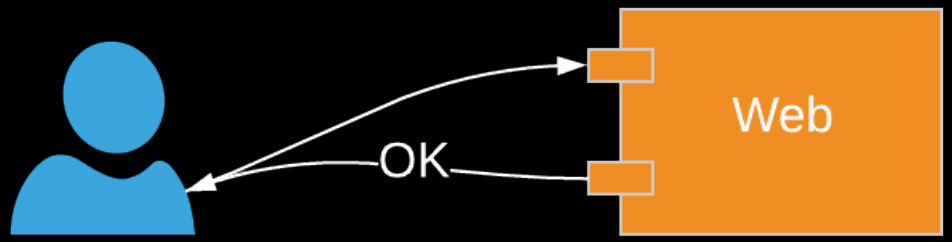
PARADIGMS OF PROGRAMMING

REQUEST / RESPONSE / RPC



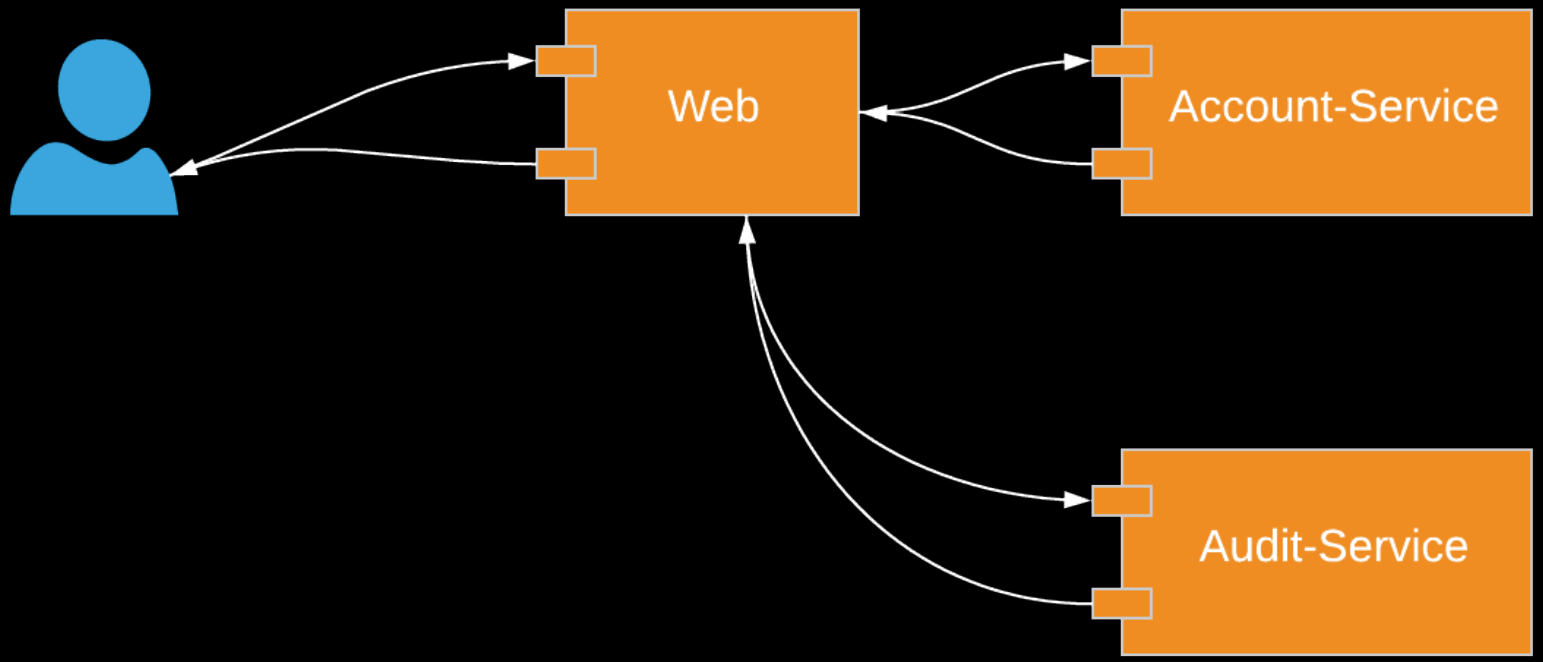
PARADIGMS OF PROGRAMMING

REQUEST / RESPONSE / RPC



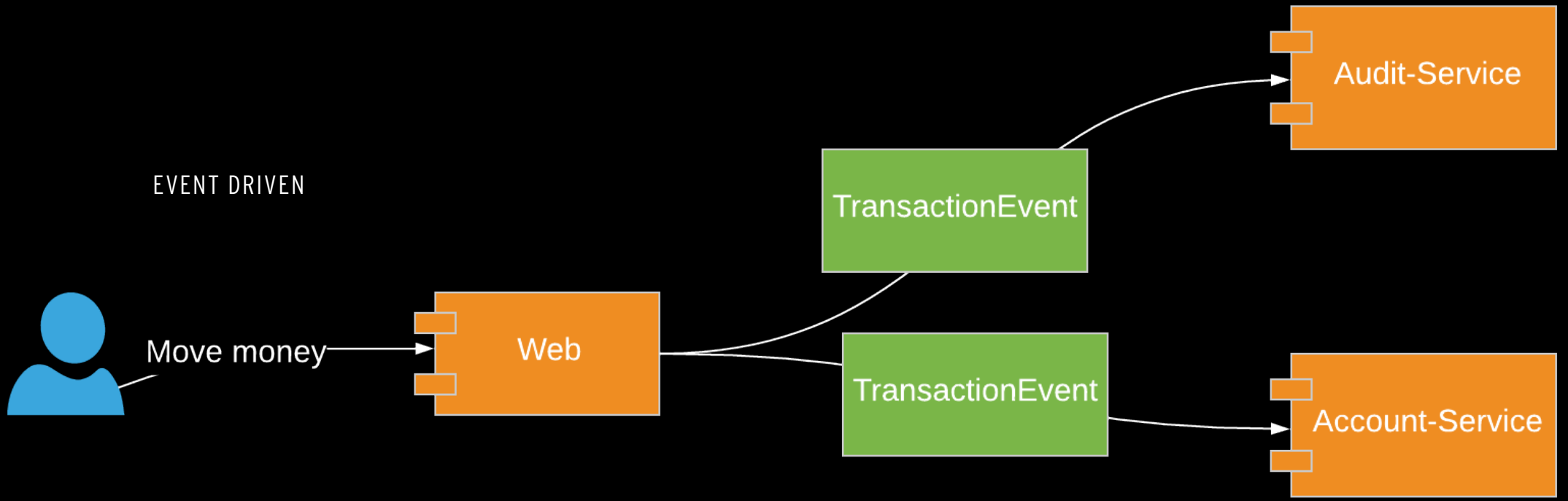
PARADIGMS OF PROGRAMMING

REQUEST / RESPONSE / RPC

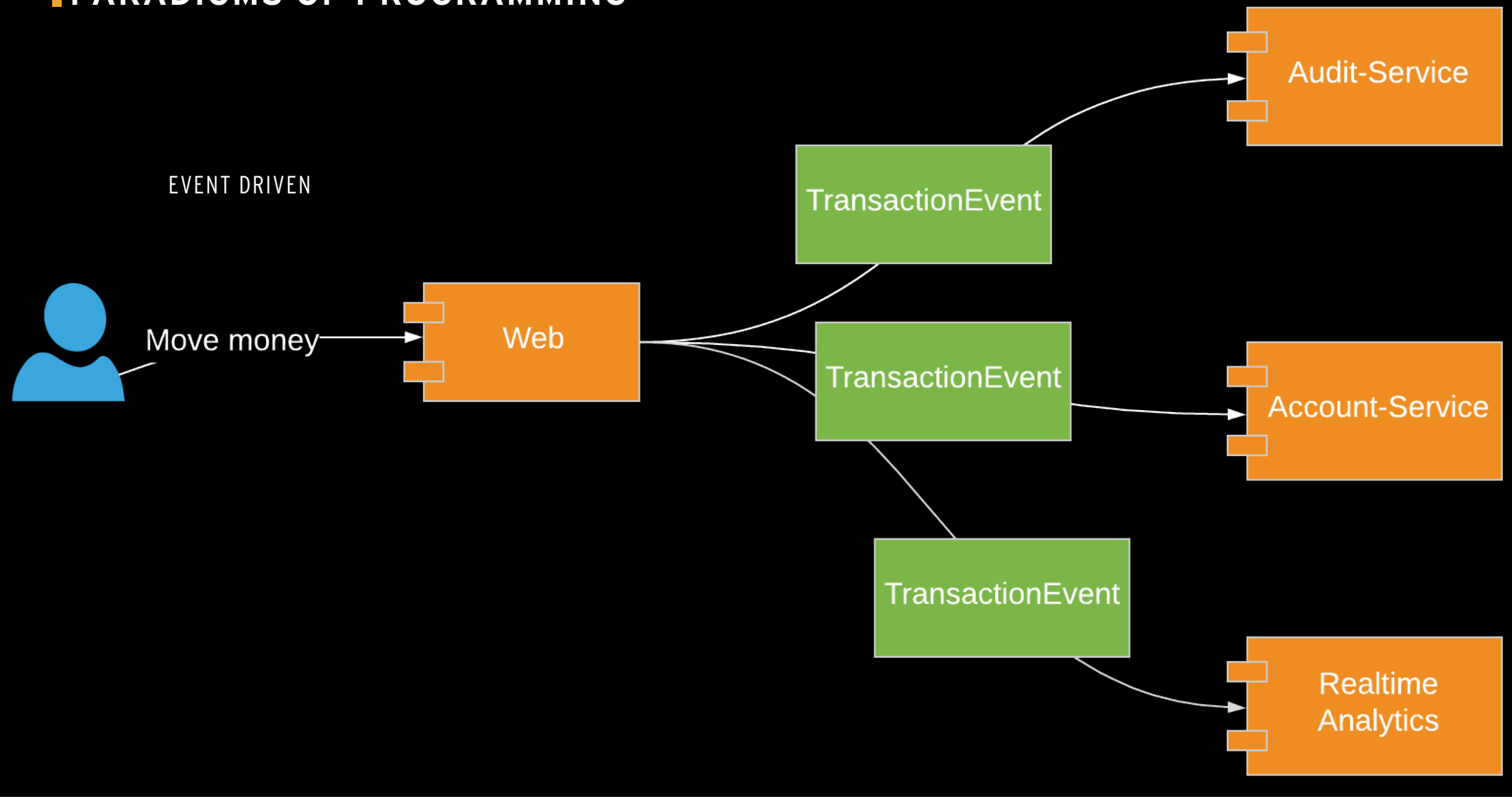


PARADIGMS OF PROGRAMMING

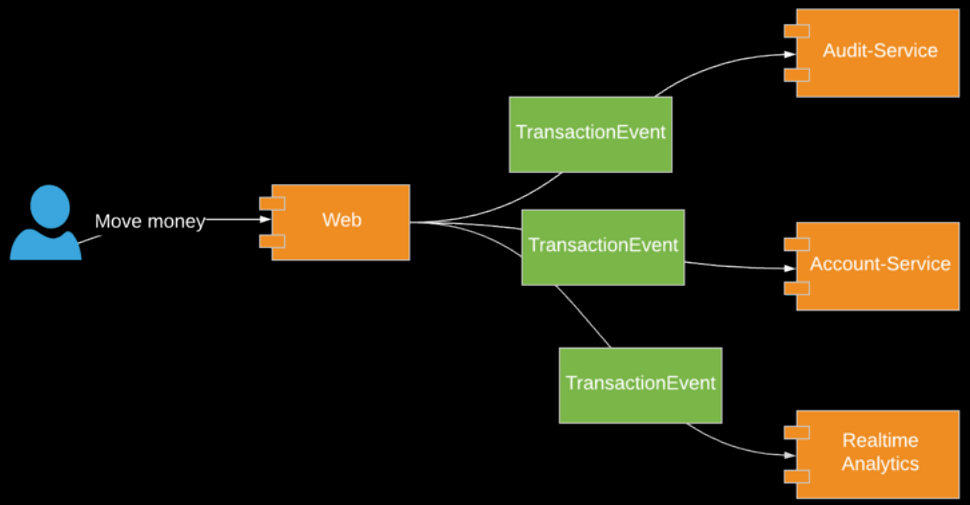
EVENT DRIVEN



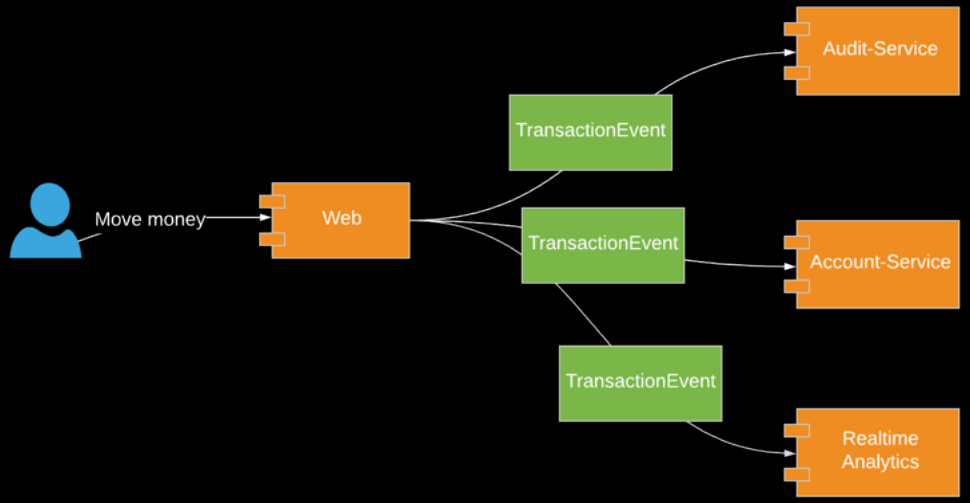
PARADIGMS OF PROGRAMMING

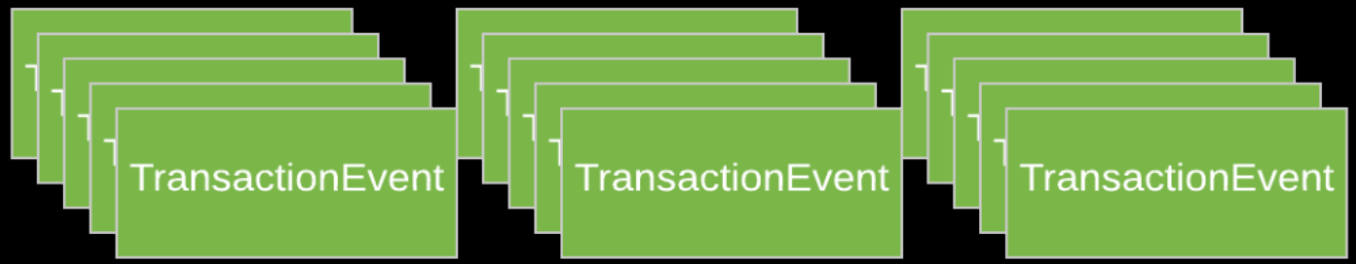


EVENT SOURCING

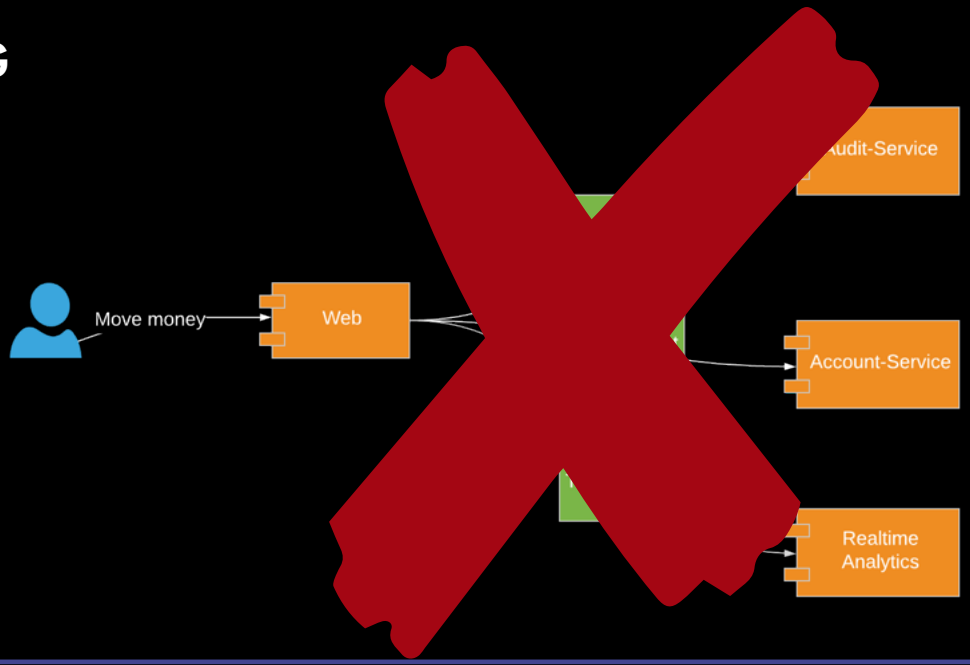


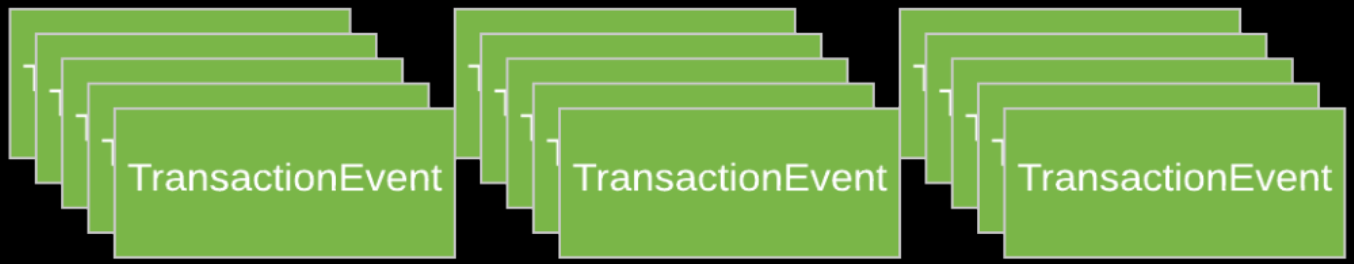
EVENT SOURCING



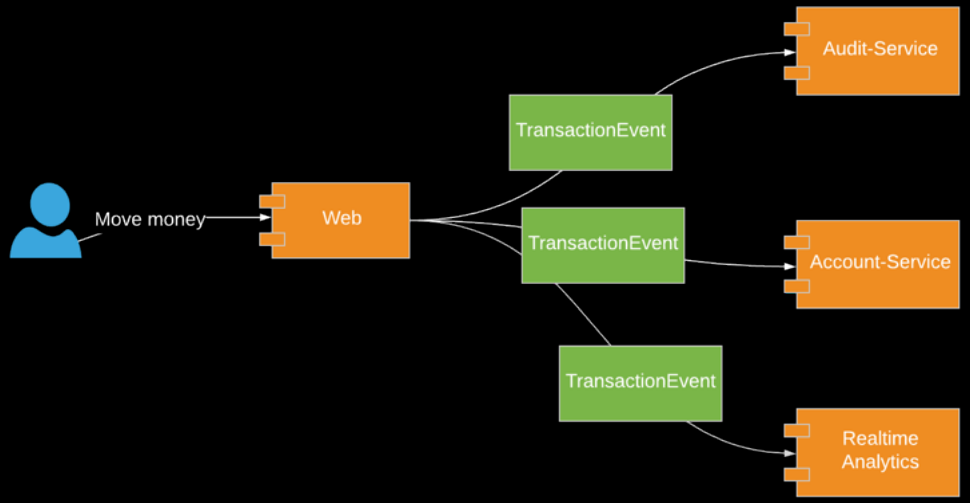


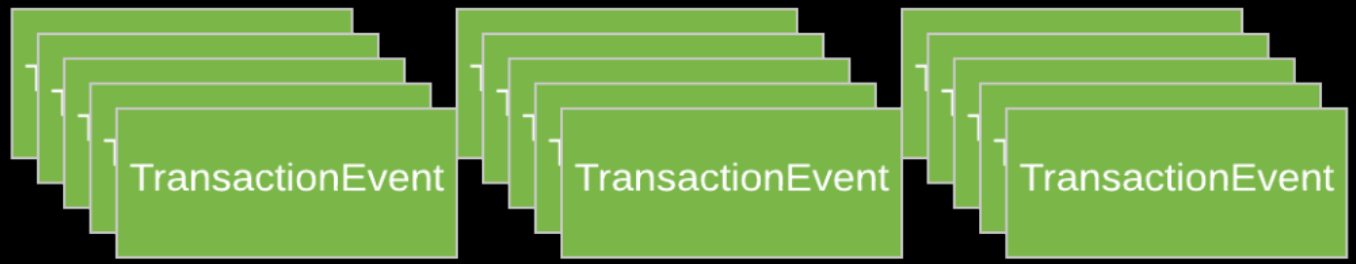
EVENT SOURCING





EVENT SOURCING





STREAMS

"STREAMING IS THE TOOLSET FOR DEALING WITH EVENTS AS THEY MOVE"

- BEN STOPFORD

STREAMS

ORDERED

STREAMS

ORDERED
IMMUTABLE

STREAMS

ORDERED
IMMUTABLE
RE-PLAYABLE

WHAT IS APACHE KAFKA?

WHAT IS APACHE KAFKA?

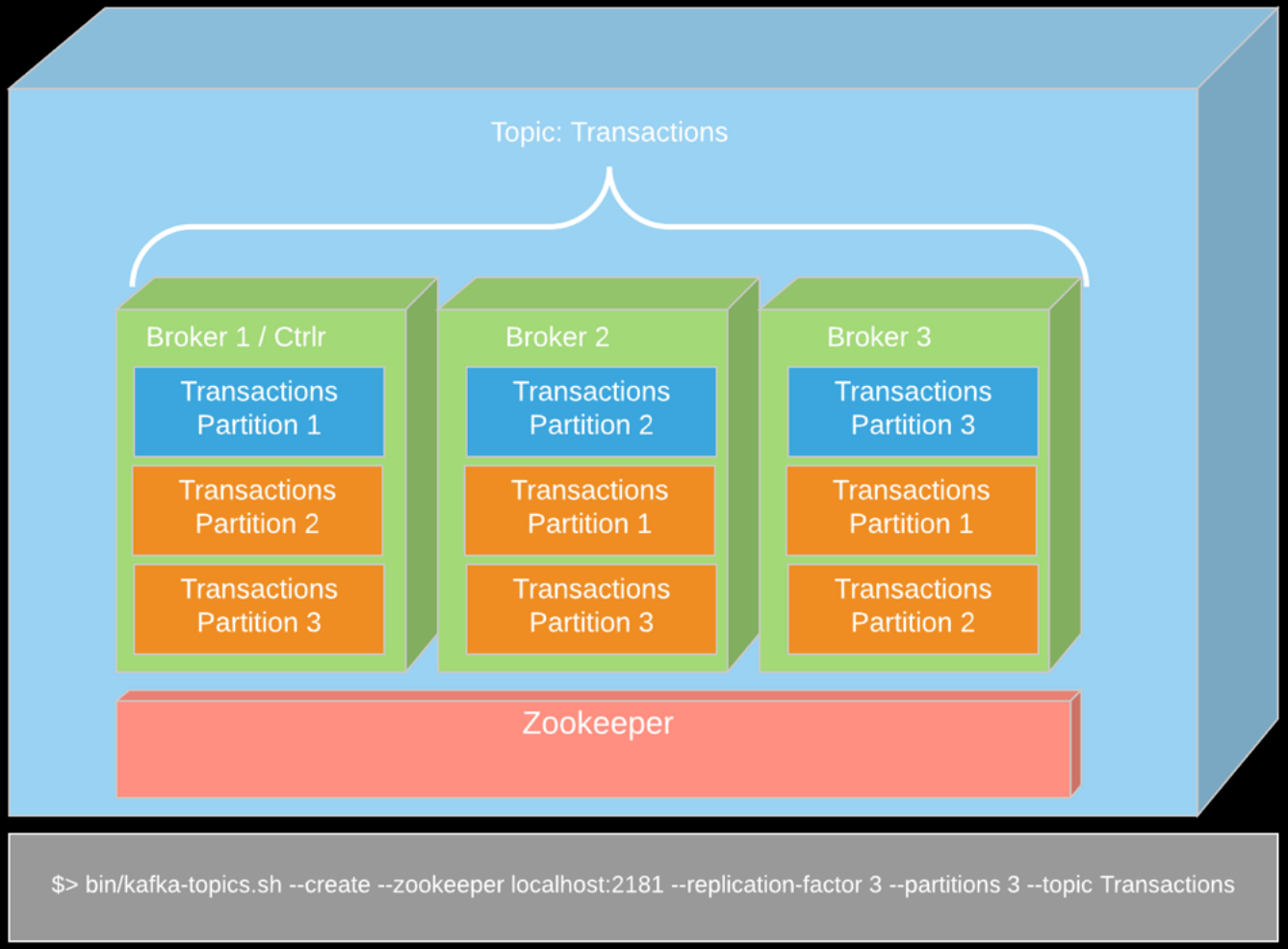
"Kafka is a distributed publish-subscribe messaging system that is designed to be fast, scalable, and durable."

kafka.apache.org

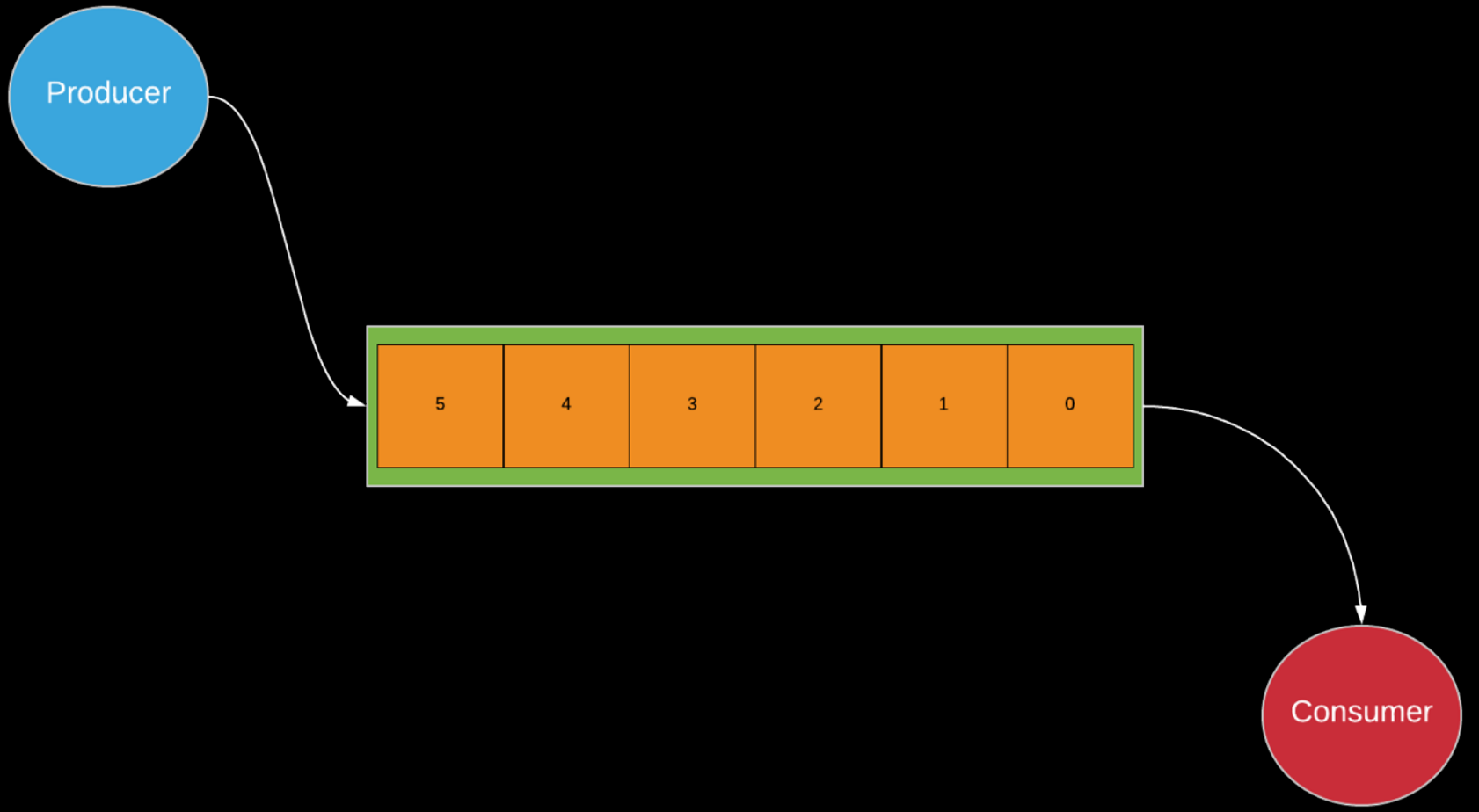
WHAT IS APACHE KAFKA?

a distributed commit log

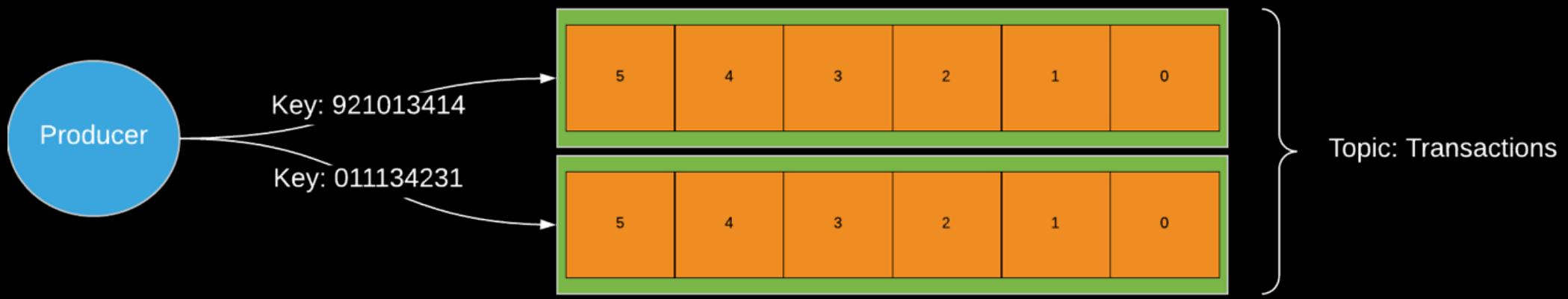
APACHE KAFKA CLUSTER



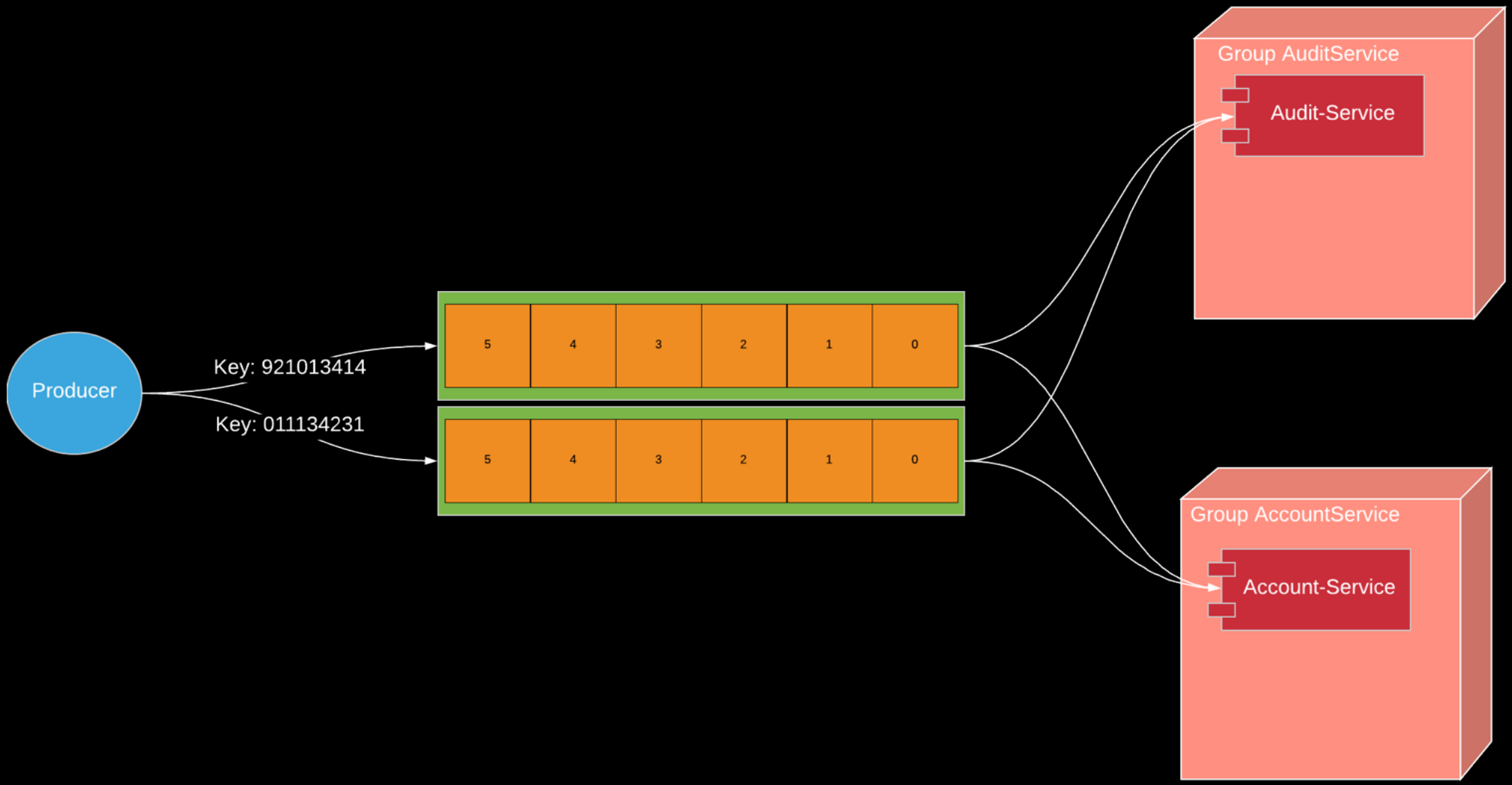
PARTITIONS



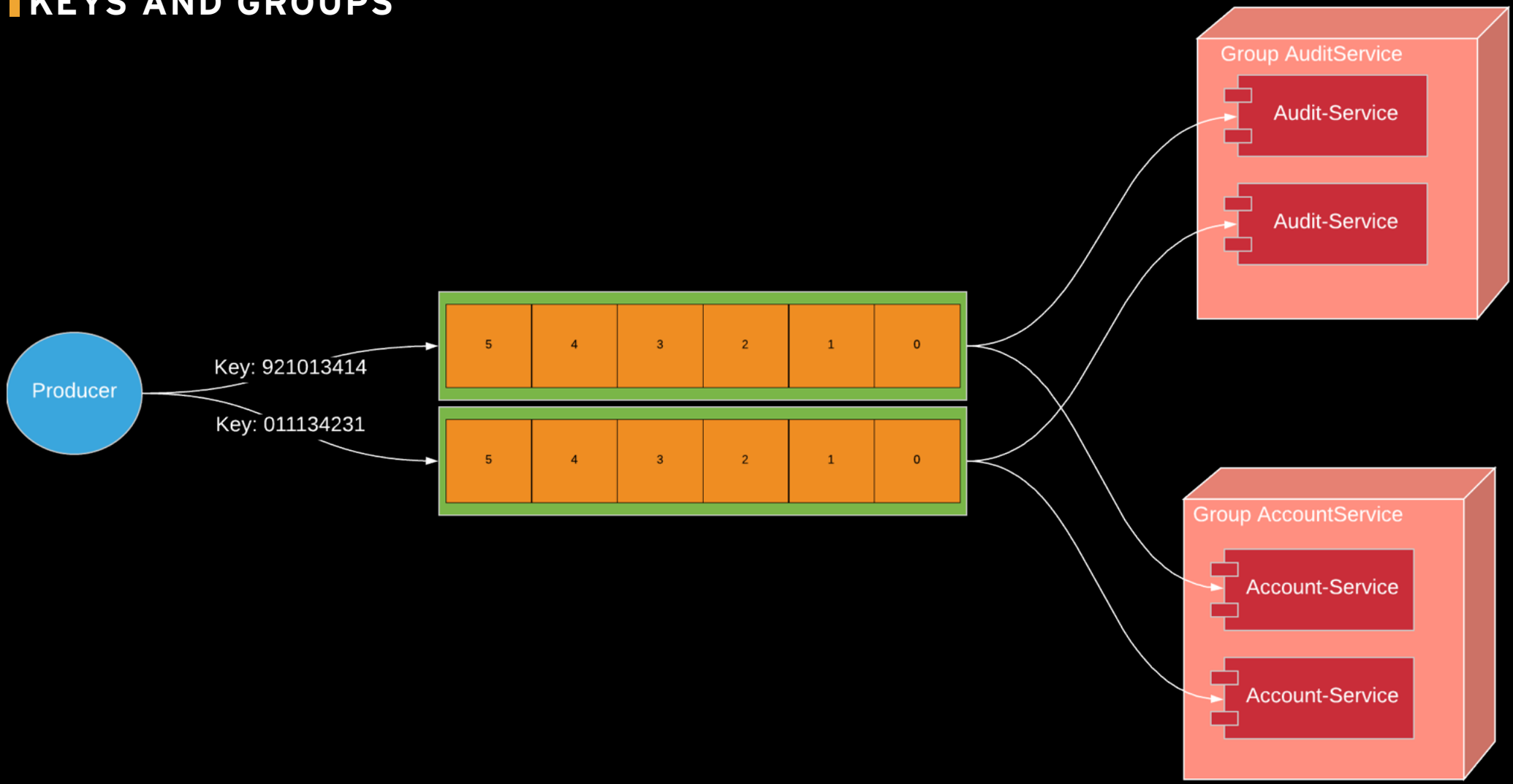
KEYS AND GROUPS



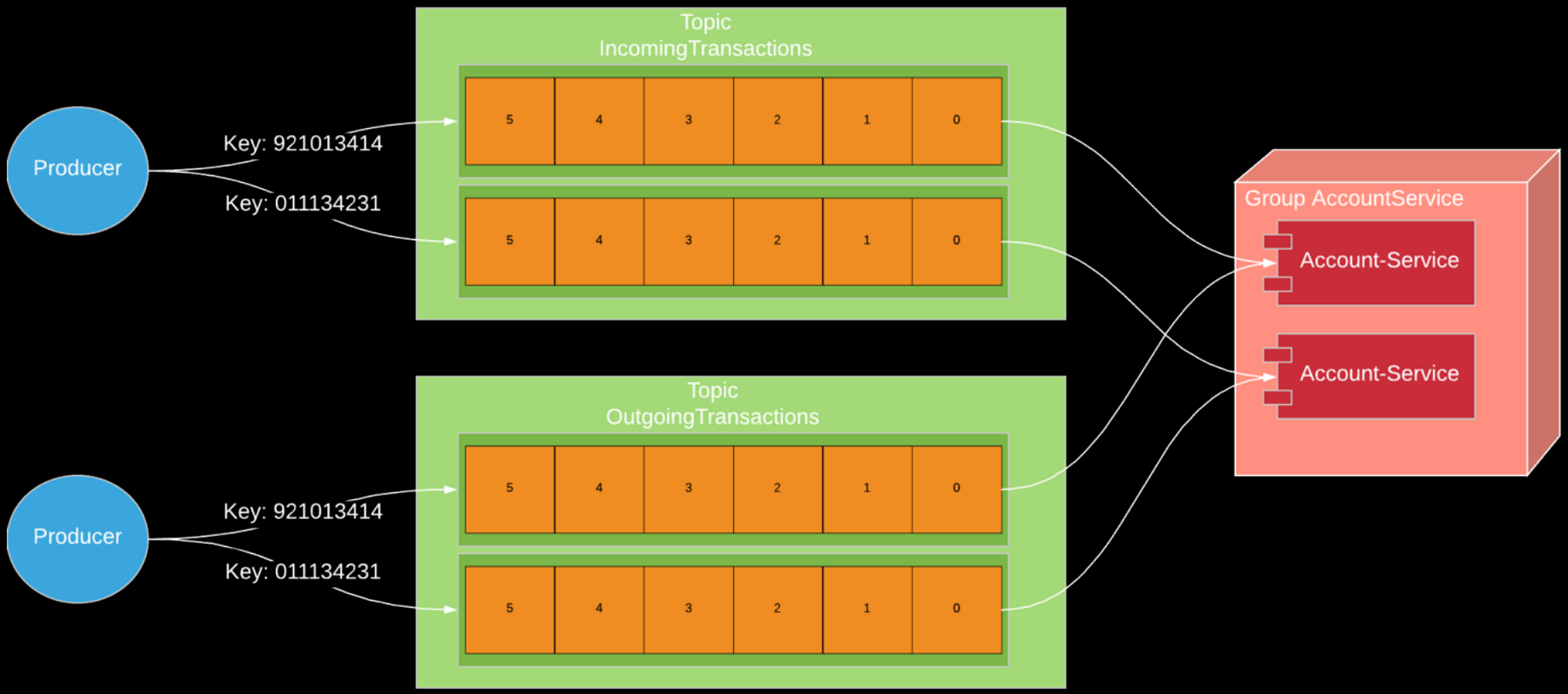
KEYS AND GROUPS



KEYS AND GROUPS



KEYS AND GROUPS



FRAMEWORKS

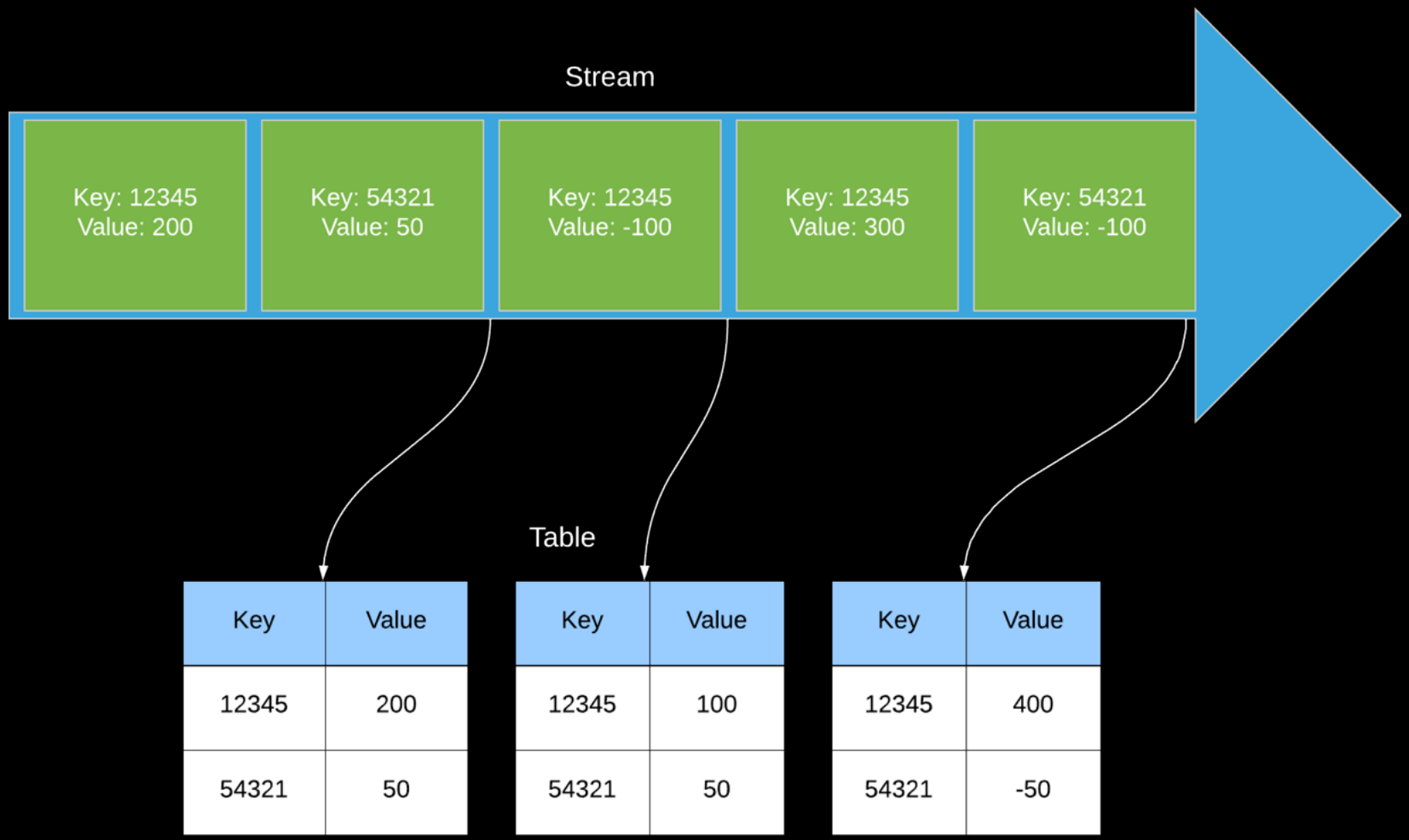
KAFKA STREAMS

KAFKA STREAMS

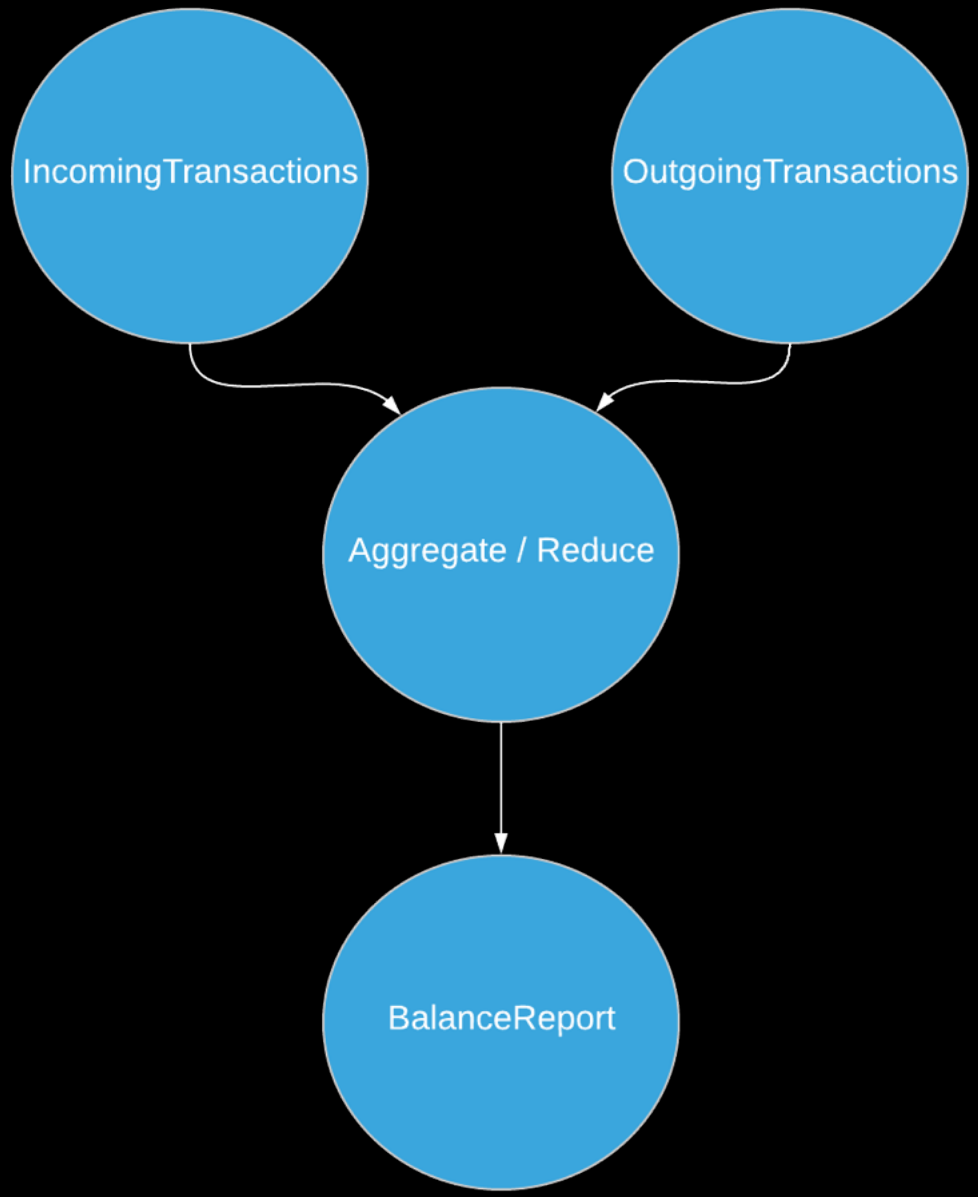
STREAMS

```
@SpringBootApplication
@EnableAutoConfiguration
@EnableKafka
@EnableKafkaStreams
public class App {

    public static void main(String... args) { SpringApplication.run(App.class, args); }

    @Bean(name = KafkaStreamsDefaultConfiguration.DEFAULT_STREAMS_CONFIG_BEAN_NAME)
    public StreamsConfig config() {
        final Map<String, Object> props = new HashMap<>();
        props.put(StreamsConfig.BOOTSTRAP_SERVERS_CONFIG, "localhost:9092");
        props.put(StreamsConfig.APPLICATION_ID_CONFIG, "firstStream");
        return new StreamsConfig(props);
    }

    @Bean
    public KStream<String, String> firstStream(final StreamsBuilder builder) {
        final KStream<String, String> stream =
                builder.stream("MightBeWords", Consumed.with(Serdes.String(), Serdes.String()));

        stream.filter((key, unknown) -> !StringUtils.isNumeric(unknown))
                .mapValues(String::toUpperCase)
                .selectKey((key, word) -> word.substring(0,1))
                .to("UpperCaseWords");

        return stream;
    }
}
```

TABLES

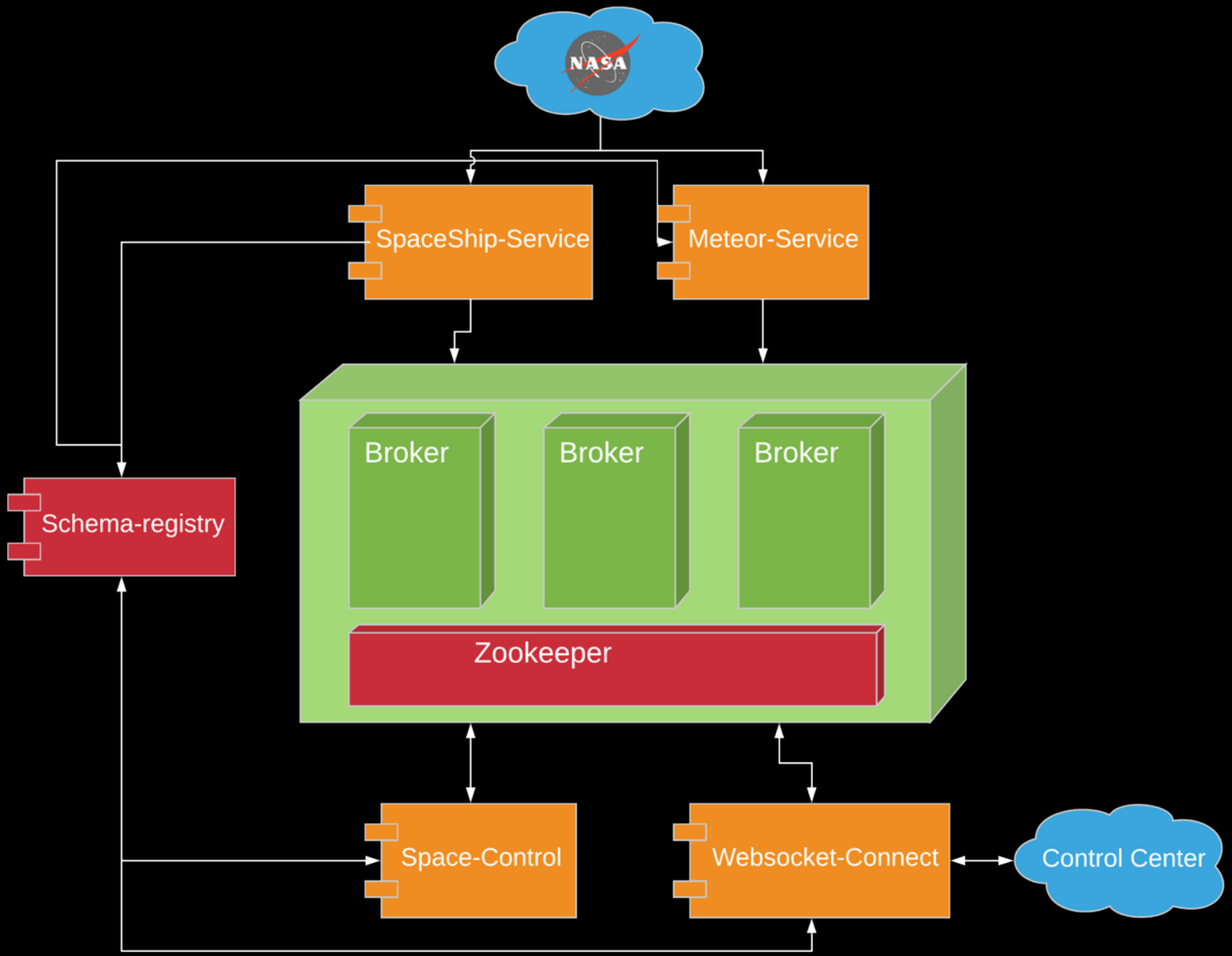

Stream
Key: 12345
Value: 200
Key: 54321
Value: 50
Key: 12345
Value: -100
Key: 12345
Value: 300
Key: 54321
Value: -100
Table
Key
Value
12345
200
54321
50
Key
Value
12345
100
54321
50
Key
Value
12345
400
54321
-50

JOINS


IncomingTransactions
OutgoingTransactions
Aggregate / Reduce
BalanceReport

A long time ago, in a galaxy far,
far away....

DEMO


NASA
SpaceShip-Service
Meteor-Service
Broker
Broker
Broker
Schema-registry
Zookeeper
Space-Control
Websocket-Connect
Control Center