IoT Devices

Kafka Broker
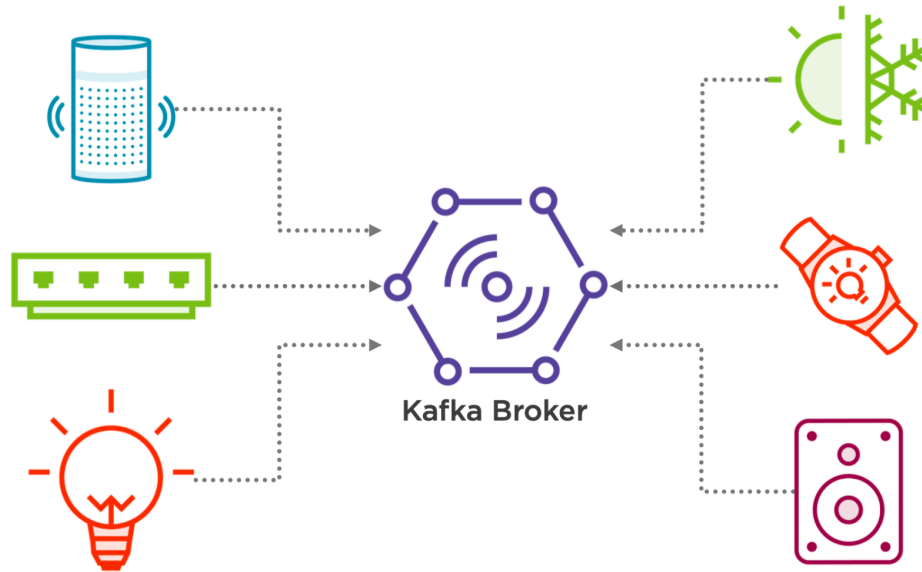
Data Serialization

Software Application

Machine

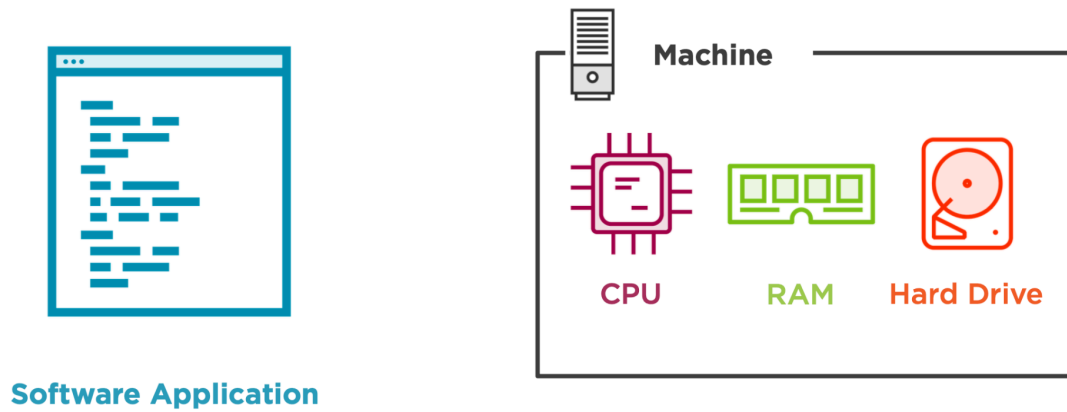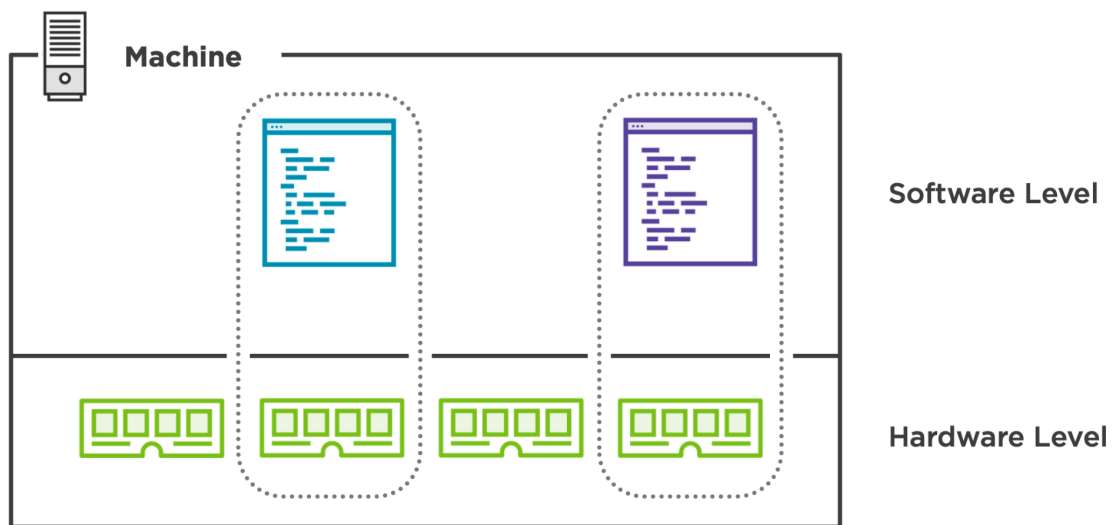| Bare Metal | Virtual Machine | Container |

Data Serialization

Software Application

Machine

CPU   RAM   Hard Drive

Data Serialization

Machine

Software Level

Hardware Level

Data Serialization


Machine 1   Machine 2
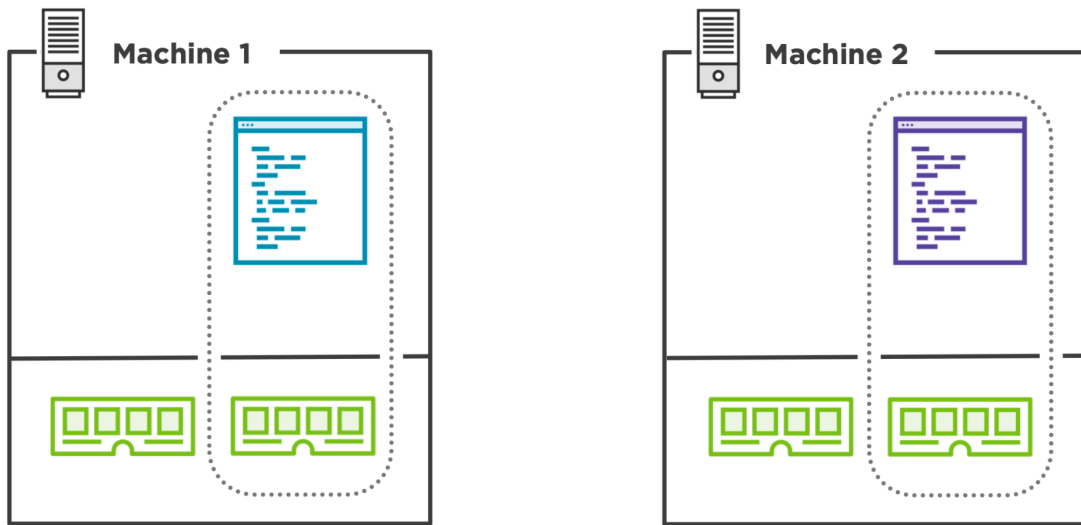
Data Serialization

Serialization

The process of translating **data structures** or **object states** into a format that can be **stored** or **transmitted** and reconstructed later, possibly in a **different** computer environment.

-Wikipedia

Data Serialization

Data Serialization

Data Serialization

Serializer — Kafka Producer → Kafka Broker(s) → Kafka Consumer — Deserializer

RAM

Zero Copy | Non-SSL
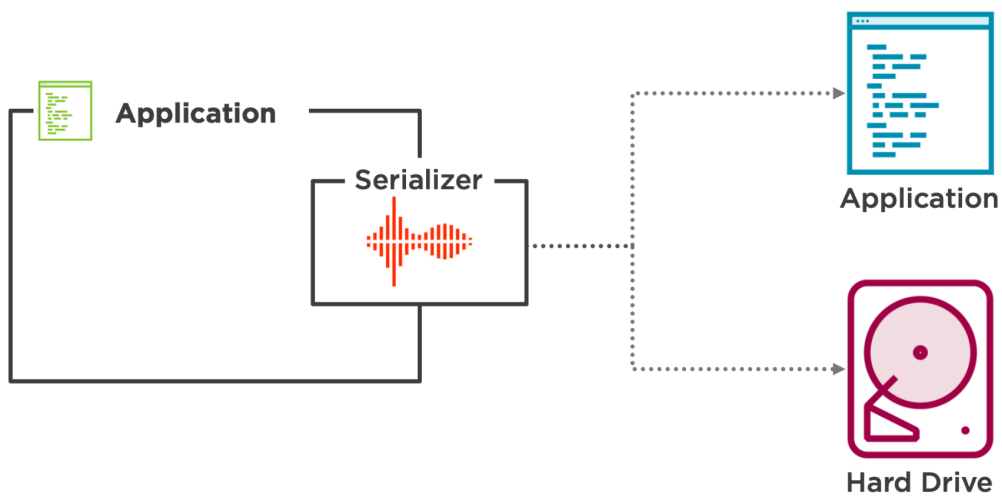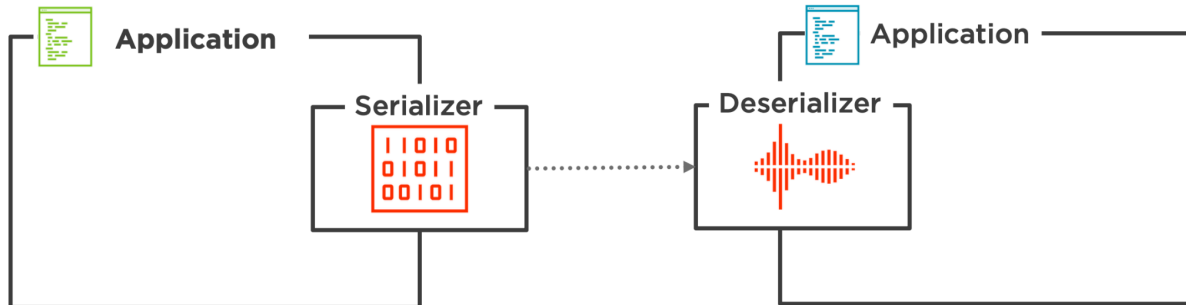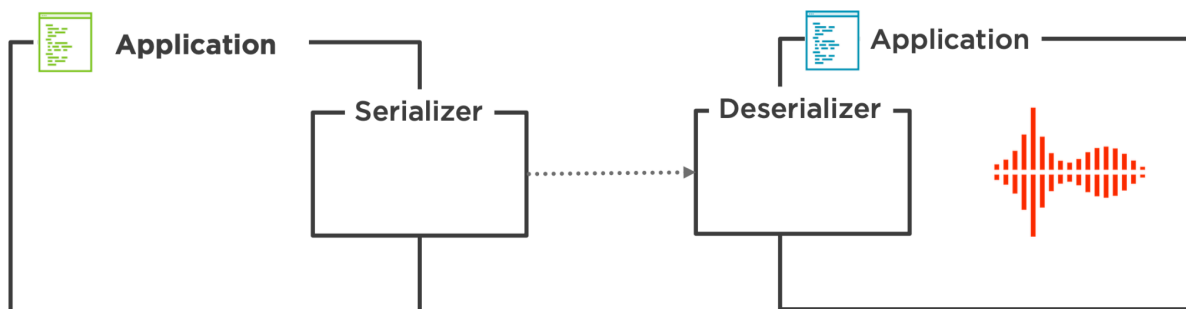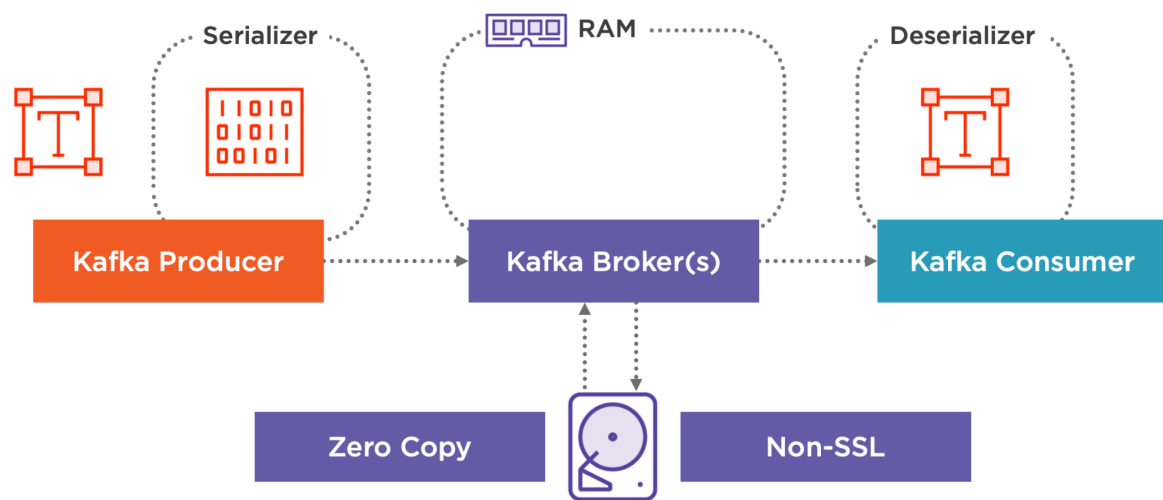
Serialization Formats

Binary?

Data is transmitted / stored as bytes or plaintext

Human-readable?

Can be read without referring to an encoding table?

Schema?

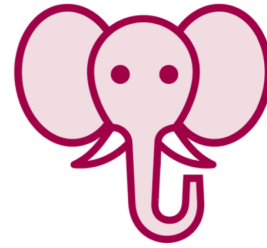Fixed data structure or Flexible

Serialization Formats

Speed
How fast can data be serialized?

Size
How much space will
serialized data take?

Serialization Formats

Name	Binary	Human-readable	Schema - IDL	
JSON	✗	✓	✗	✓
XML	✗	✓	✓	
YAML	✗	✓	✗	
AVRO	✓	✗	✓	
Protobuf	✓	✗	✓	
Thrift	✓	✗	✓	

Serialization Formats with Apache Kafka

| AVRO | - first class support using Schema Registry |
JSON	- easy, no learning curve, extremely common
Protobuf	- highly efficient, gRPC
Thrift	- Twitter Finagle, Apache Cassandra

Serialization Formats with Apache Kafka

AVRO		
JSON	Human-readable	Large Scale Adoption
Protobuf		
Thrift		

Serialization Formats with Apache Kafka

JSON	Human-readable	Large Scale Adoption

AVRO	Binary	Compact Data
Protobuf	Schema-IDL	Data Evolution
Thrift		
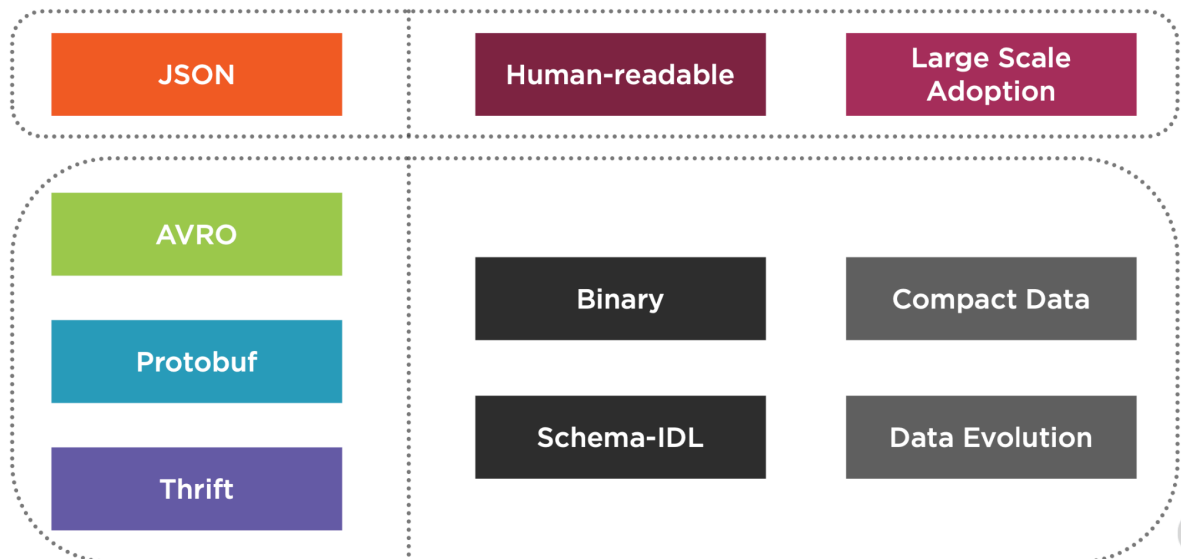
Schema-IDL

schema.json
```json
{
    "type" : "string"
}
```

schema.proto
```proto
syntax = "proto3";

message AudioCommand {
  string command = 1;
}
```

schema.avsc
```json
{
    "type" : "string",
    "name" : "command"
}
```

schema.thrift
```thrift
struct AudioCommand {
    1: string command
}
```
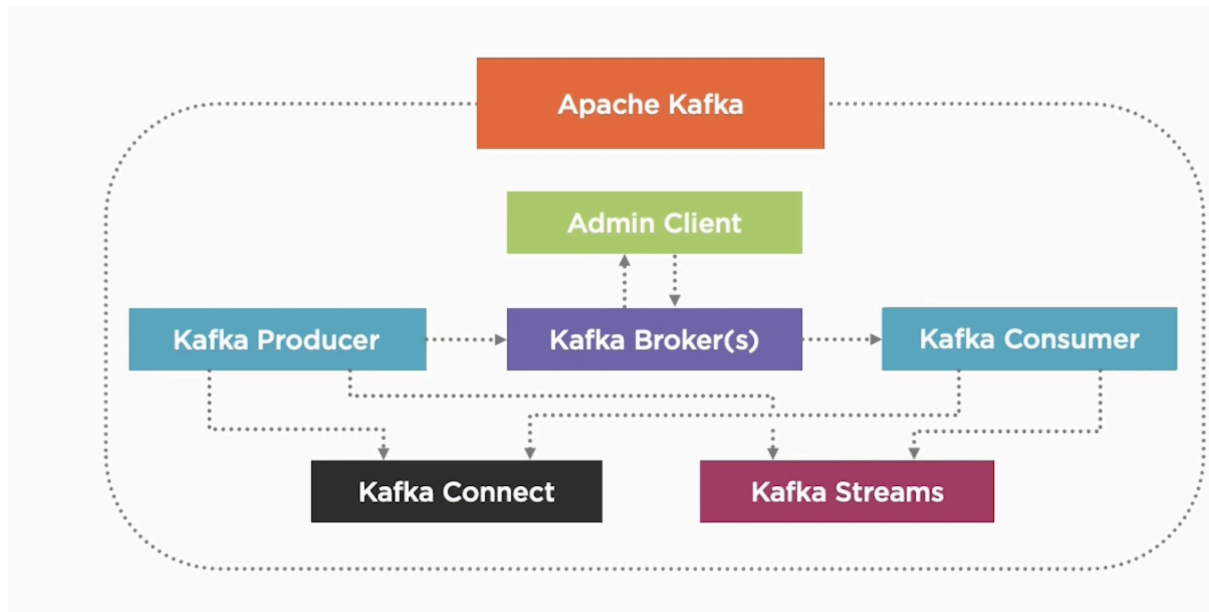
Schema-IDL

Fixed Data Structure

Fine-grained Control

Data Contract

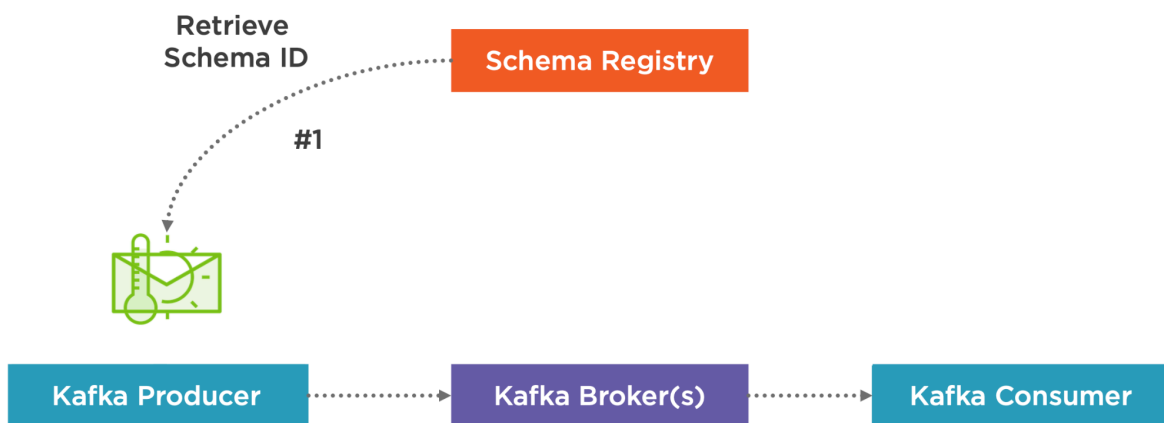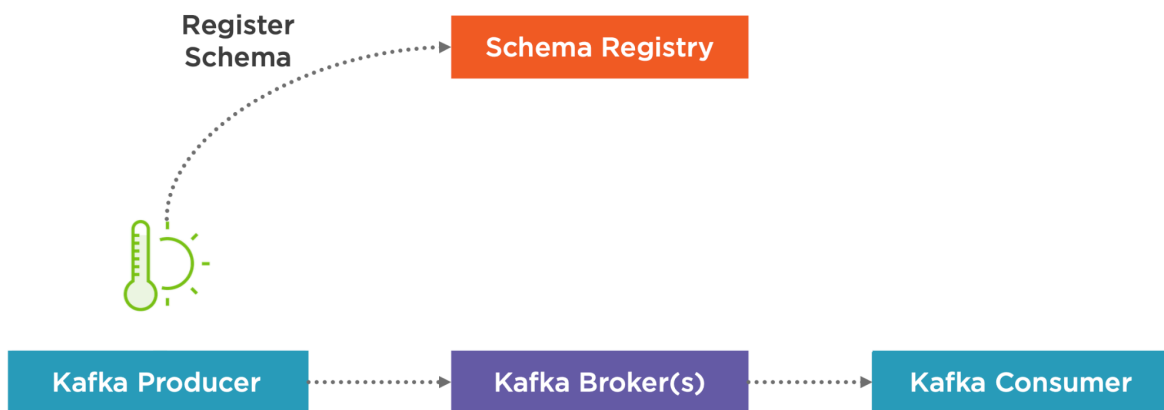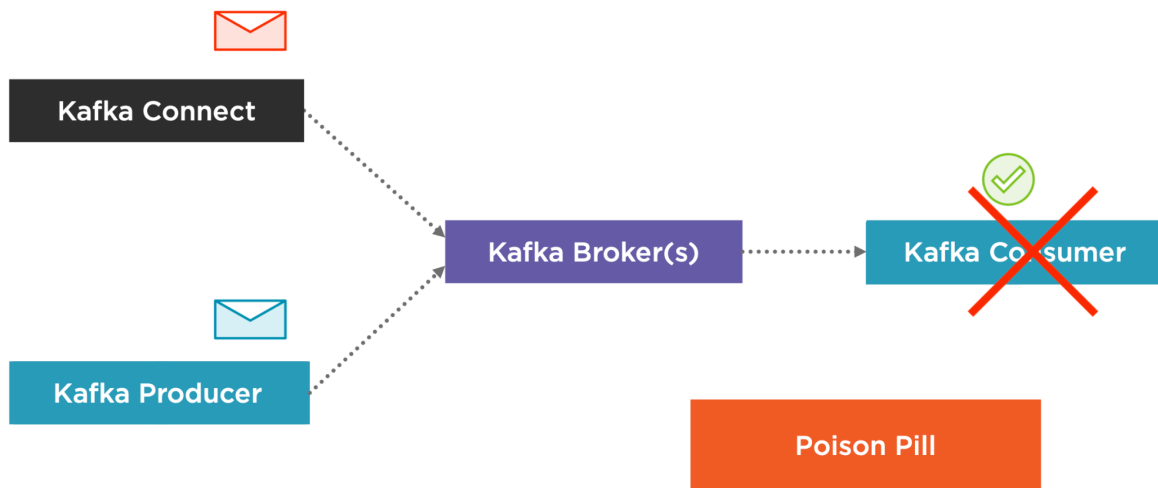A formal **agreement** between a service and a client that abstractly **describes the data** to be exchanged.

- Microsoft

Schema Registry Architecture

Schema Registry

Enforcing Data Contracts	**Handling Schema Evolution**

Kafka Connect

Kafka Producer

Kafka Broker(s)

Kafka Consumer

Poison Pill

Register Schema

Schema Registry

Kafka Producer

Kafka Broker(s)

Kafka Consumer

Retrieve Schema ID

Schema Registry

#1

Kafka Producer

Kafka Broker(s)

Kafka Consumer

Schema Registry

#1

Kafka Producer → Kafka Broker(s) → Kafka Consumer

Schema Registry

#1

Kafka Producer → Kafka Broker(s) → Kafka Consumer

GET Schema
By ID

Schema Registry

#1

Kafka Producer → Kafka Broker(s) → Kafka Consumer

Retrieve
Schema

Schema Registry

#1

Kafka Producer → Kafka Broker(s) → Kafka Consumer

Schema Registry

KafkaAvroSerializer   **KafkaAvroDeserializer**

Kafka Producer ·····▶ **Kafka Broker(s)** ·····▶ **Kafka Consumer**

Q: How does this process take place?

REST API   **Schema Registry**   REST API

KafkaAvroSerializer   **KafkaAvroDeserializer**

Kafka Producer ·····▶ **Kafka Broker(s)** ·····▶ **Kafka Consumer**

Q: How they communicate with the Schema Registry?

Schema Registry

Kafka
Producer

Kafka
Consumer
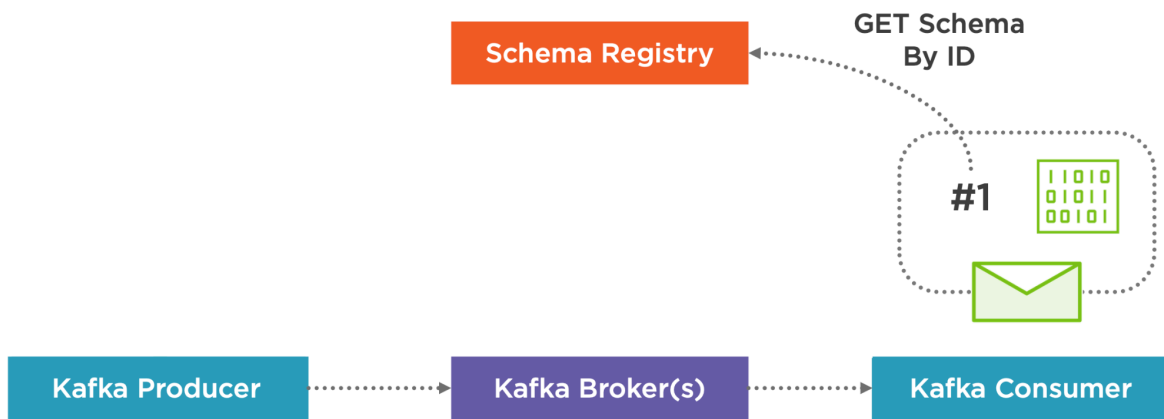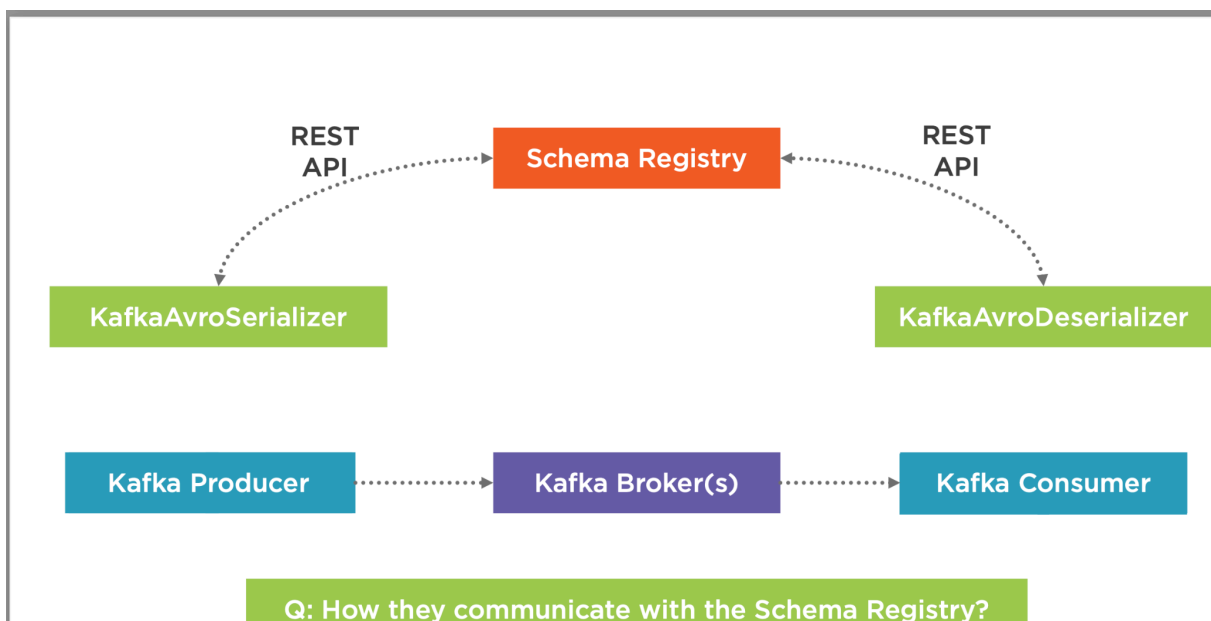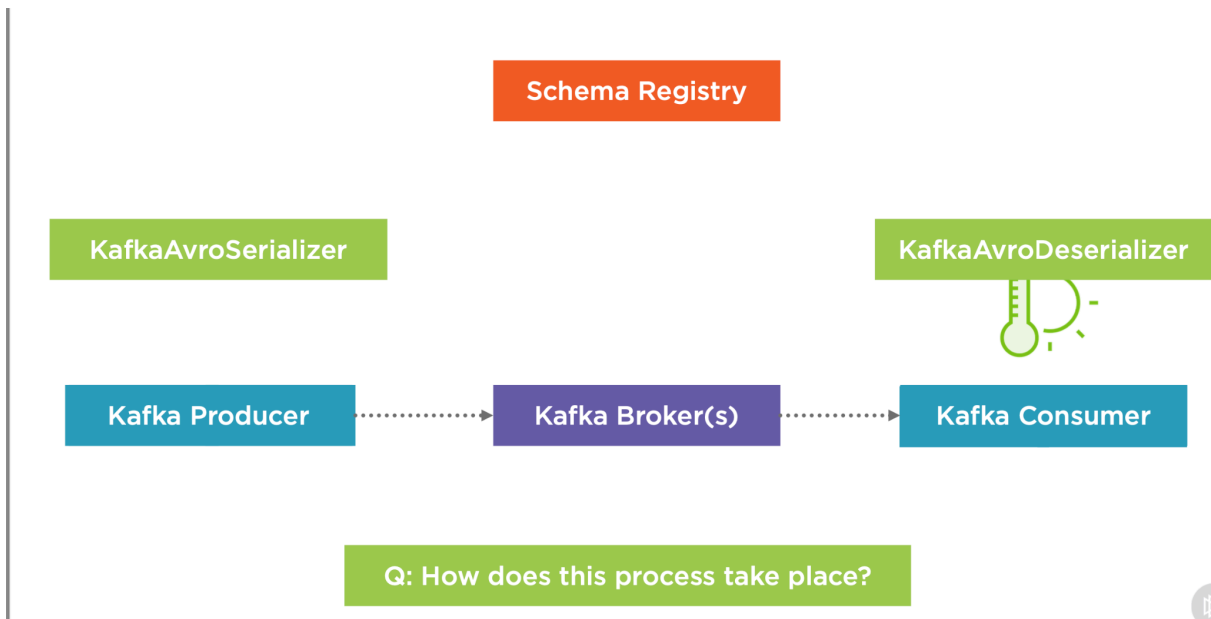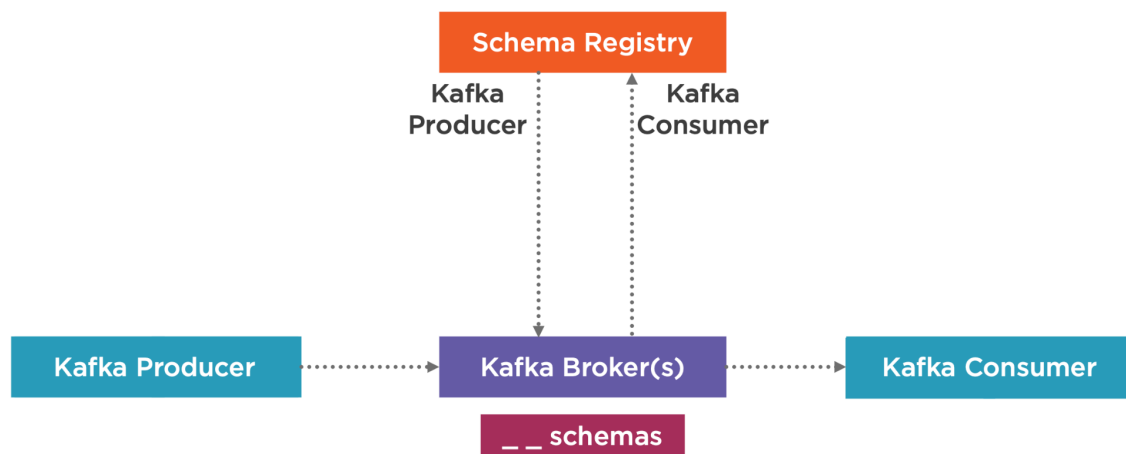
Kafka Producer ┄┄┄▶ **Kafka Broker(s)** ┄┄┄▶ **Kafka Consumer**

_ _ schemas

Q: Where does Schema Registry store the schemas?