Streaming Data Analysis

Streaming Message Brokers

Spark is intended to process data that was previously generated. It doesn't process data in real time. **Spark Streaming** gives us the option of processing data in near real-time. In this course, we won't go into Spark Streaming in-depth, but we'll cover it briefly for awareness.

Often data is generated in real-time and then is stored for later processing. A common example is **IoT** or Internet of Things, which consists of small devices sending internet-connected messages. These devices send messages to convey meaning about the world. A smart thermostat is an IoT device. It continually communicates back with a server to send usage statistics to the end user.

Because servers are not always designed to handle large volumes of real-time data, **message brokers** were created. They are intended to "broker" connections between systems and make near real-time processing of data possible.

Some examples of message brokers are:

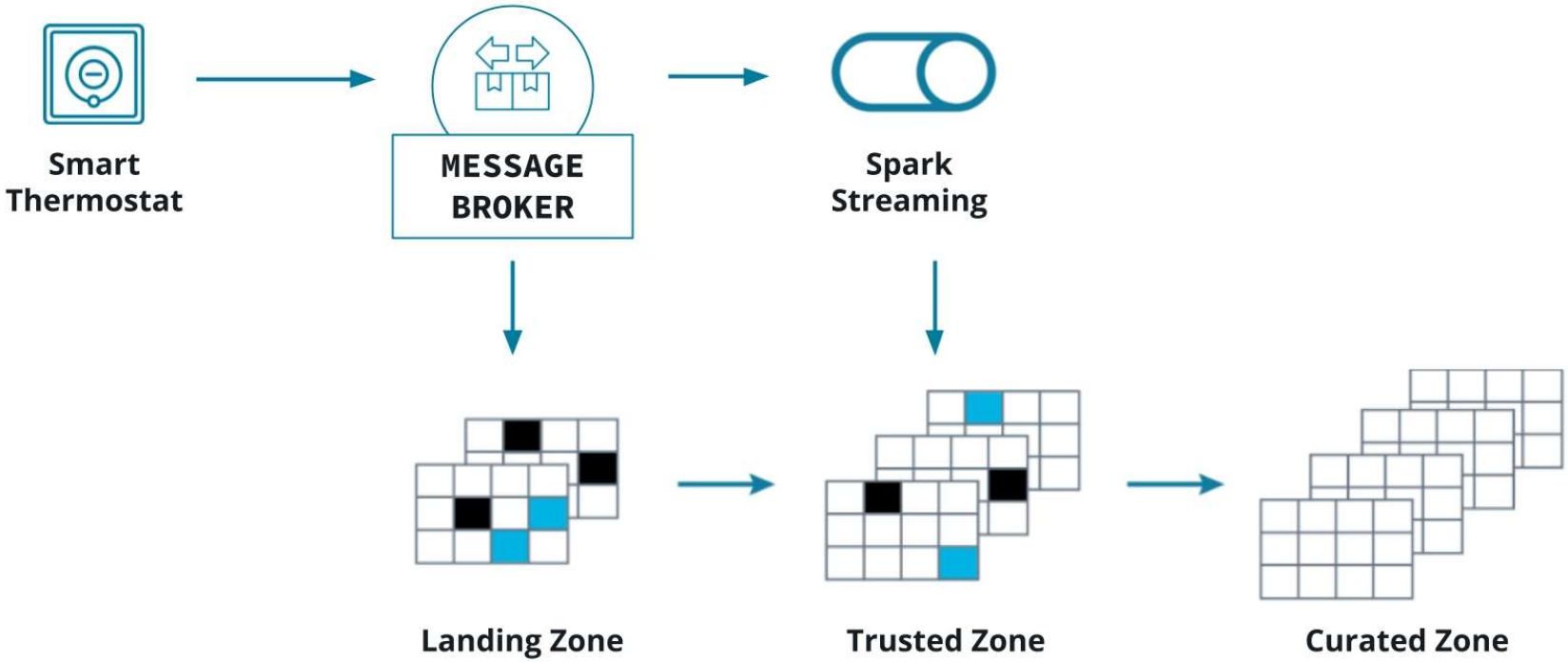
- Kafka
- Simple Queue Services (AWS SQS)
- Amazon Kinesis

Brokers aren't Forever

Unlike diamonds, message brokers don't last forever. Neither do the data they store. They are intended to facilitate a message being received and re-transmitted. This event typically should happen within seven days. Then the data will be deleted from the **Raw Zone**.

To keep messages longer, we move them into a **Landing Zone**. This is where the data can be loaded and transformed for later use in the **Trusted** and **Curated Zone**.

Streaming Data



Using Glue to Process Streaming Data

Glue can load data directly from Kafka or Kinesis. AWS doesn't offer Glue support for SQS at this time. Using Spark Streaming, we can load data from Message Brokers into a Spark DataFrame or Glue DynamicFrame.

We can then join data from the Message Broker with other data sources as part of the streaming job to create **Trusted** or **Curated data**. Kafka can be configured to load data

into S3 using a Kafka Connector as a **Landing Zone**, avoiding the need to connect Glue to Kafka directly.