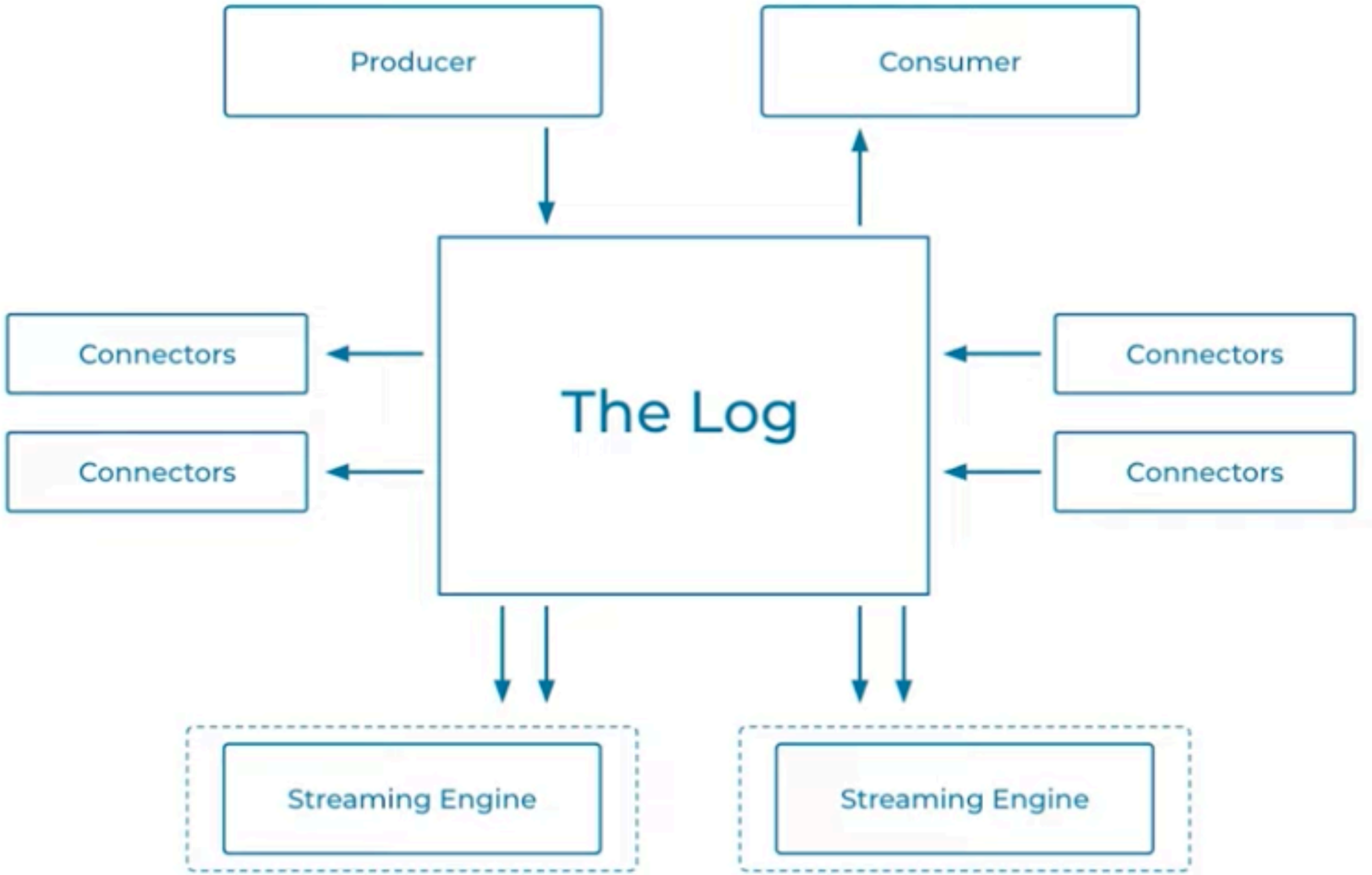
Apache Kafka



How to process events in Kafka?

```
{
 "reading_ts": "2020-02-14T12:19:27Z",
 "sensor_id": "aa-101",
 "production_line": "w01",
 "widget_type": "acme94",
 "temp_celcius": 23,
 "widget_weight_g": 100
}
```





Processing events with producer and consumer clients

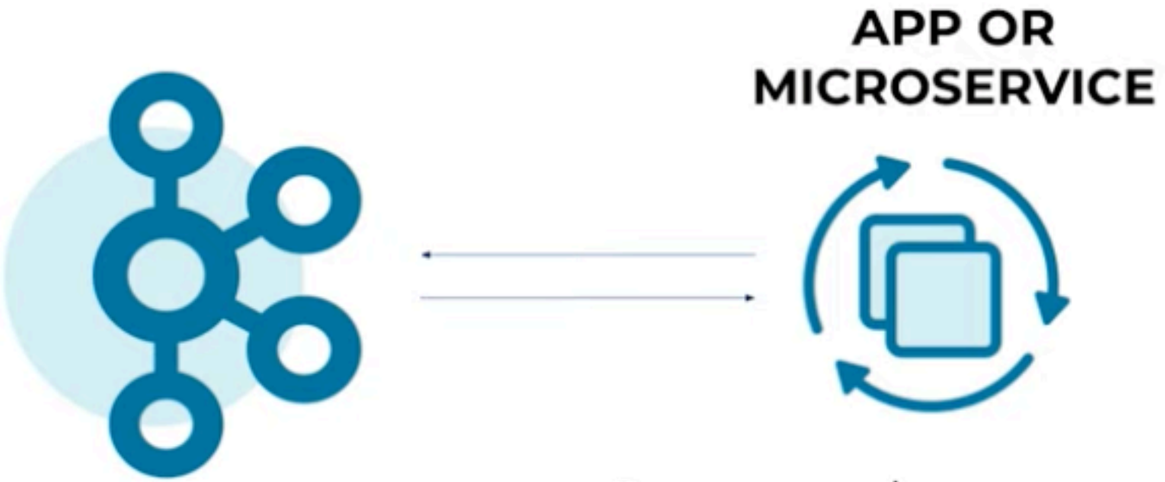
```
public static void main(String[] args) {
  try(Consumer<String, Widget> consumer = new KafkaConsumer<>(consumerProperties());
     Producer<String, Widget> producer = new KafkaProducer<>(producerProperties())) {
        consumer.subscribe(Collections.singletonList("widgets"));
        while (true) {
            ConsumerRecords<String, Widget> records =    consumer.poll(Duration.ofSeconds(5));
                for (ConsumerRecord<String, Wigdet> record : records) {
                    Widget widget = record.value();
                    if (widget.getColour().equals("red") {
                        ProducerRecord<String, Widget> producerRecord = new ProducerRecord<>(
                            "widgets-red", record.key(), widget);
                        producer.send(producerRecord, (metadata, exception)-> {…….} );
                ...
```

Processing events with Kafka Streams

```
final StreamsBuilder builder = new StreamsBuilder();

builder.stream(“widgets”, Consumed.with(stringSerde, widgetsSerde))
       .filter( (key, widget) -> widget.getColour.equals("red"))
       .to("widgets-red", Produced.with(stringSerde, widgetsSerde));
```

Kafka Streams is a Java library



Create, read, process events.

Use the library to write standard Java/JVM applications that process data in Kafka.
Kafka Streams makes your applications elastic, distributed, scalable, and fault-tolerant.