**Kafka Connect IBM MQ Source**

The Kafka Connect IBM MQ Source Connector is used to read messages from an IBM MQ cluster and write them to an Apache Kafka® topic.

***Features :***

***JMS Message types***

The connector currently supports TextMessage and BytesMessage. The connector does not currently support ObjectMessage or StreamMessage.

***Retries and Reconnection***

The connector can be configured to retry on retriable errors using the max.retry.time configuration property. This property sets the maximum time in milliseconds (ms) the connector will attempt to retry. The property defaults to 3600000 ms (1 hour). The connector uses exponential backoff after each retry attempt. That is, each subsequent retry attempt interval increases exponentially with jitter.

The following errors will trigger a retry:

MQRC\_GET\_INHIBITED

MQRC\_BACKED\_OUT

MQRC\_CHANNEL\_NOT\_AVAILABLE

MQRC\_CONNECTION\_BROKEN

MQRC\_HOST\_NOT\_AVAILABLE

MQRC\_NOT\_AUTHORIZED

MQRC\_Q\_MGR\_NOT\_AVAILABLE

MQRC\_Q\_MGR\_QUIESCING

MQRC\_Q\_MGR\_STOPPING

MQRC\_UNEXPECTED\_ERROR

**Topics**

This connector consumes messages from IBM MQ using the configured message selectors and writes them to a single Kafka topic. If you want to write messages to multiple topics, use a simple message transform that routes the messages based upon your criteria.

**Schemas**

io.confluent.connect.jms.Key

This schema is used to store the incoming MessageID on the message interface. This will ensure that when that if the same message id arrives it will end up in the same partition. In practice this should never occur.

io.confluent.connect.jms.Value

This schema is used to store the value of the JMS message.

io.confluent.connect.jms.Destination

This schema is used to represent a JMS Destination, and is either queue or topic.

io.confluent.connect.jms.PropertyValue

This schema is used to store the data that is found in the properties of the message. To ensure that the proper type mappings are preserved field propertyType stores the value type for the field. The corresponding field in the schema will contain the data for the property. This ensures that the data is retrievable as the type returned by Message.getObjectProperty().