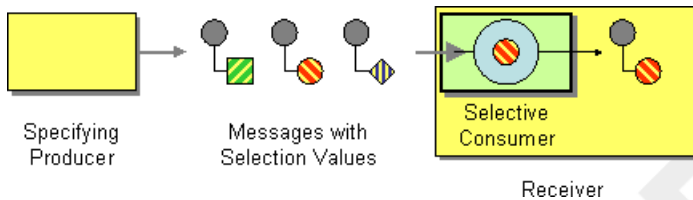


Selective Consumer

Overview

The *selective consumer* pattern describes a consumer that applies a filter to incoming messages, so that only messages meeting a specific selection criterion are processed.

Figure 30. Selective Consumer Pattern



You can implement the selective consumer pattern in Mediation Router using one of the following approaches:

- [JMS selector on page 96](#).
- [JMS selector in ActiveMQ on page 97](#)
- [Message filter on page 97](#).

JMS selector

A JMS selector is a predicate expression involving JMS headers and JMS properties: if the selector evaluates to `true`, the JMS message is allowed to reach the consumer; if the selector evaluates to `false`, the JMS message is blocked. For example, to consume messages from the queue, `selective`, and select only those messages whose country code property is equal to `US`, you could use the following Java DSL route:

```
from("jms:selective?selector=" + java.net.URLEncoder.encode("CountryCode='US'", "UTF-8"))
  to("cxf:bean:replica01");
```

Where the selector string, `CountryCode='US'`, must be URL encoded (using UTF-8 characters) in order to avoid trouble with parsing the query options. This example presumes that the JMS property, `CountryCode`, was set by the sender. For more details about JMS selectors, see [JMS selectors on page 93](#).



Note

If a selector is applied to a JMS queue, messages that are not selected remain on the queue (and are thus potentially available to other consumers attached to the same queue).

JMS selector in ActiveMQ

You can also define JMS selectors on ActiveMQ endpoints. For example:

```
from("activemq:selective?selector=" + java.net.URLEncoder.encode("CountryCode='US'", "UTF-8")).
to("cxf:bean:replica01");
```

For more details, see [ActiveMQ: JMS Selectors](http://activemq.apache.org/selectors.html) [http://activemq.apache.org/selectors.html] and [ActiveMQ Message Properties](http://activemq.apache.org/activemq-message-properties.html) [http://activemq.apache.org/activemq-message-properties.html].

Message filter

If it is not possible to set a selector on the consumer endpoint, you can insert a filter processor into your route instead. For example, you could define a selective consumer that processes only messages with a US country code using Java DSL, as follows:

```
from("seda:a").filter(header("CountryCode").isEqualTo("US")).process(myProcessor);
```

The same route can be defined using XML configuration, as follows:

```
<camelContext id="buildCustomProcessorWithFilter" xmlns="http://activemq.apache.org/camel/schema/spring">
  <route>
    <from uri="seda:a"/>
    <filter>
      <simple>header.CountryCode = 'US'</simple>
      <process ref="#myProcessor"/>
    </filter>
  </route>
</camelContext>
```

For more information about the Mediation Router filter processor, see [Message Filter on page 54](#).



Warning

Be careful about using a message filter to select messages from a JMS *queue*. When using a filter processor, blocked messages are simply discarded. Hence, if the messages are consumed from a queue

(which allows each message to be consumed only once—see [Competing Consumers on page 91](#)), blocked messages would not be processed at all. This might not be the behavior you want.

DRAFT