This PDF is generated from authoritative online content, and is provided for convenience only. This PDF cannot be used for legal purposes. For authoritative understanding of what is and is not supported, always use the online content. To copy code samples, always use the online content.

Working with iWD Capture Points

JMS Capture Point

4/25/2025

Contents

JMS Capture Point

The JMS Capture Point uses the Java Message Service (JMS) as a way to pass messages between the source system and iWD. JMS is a queue-based message system, and the iWD JMS Capture Point utilizes four queues:

- An inbound queue that the source system will use to submit XML messages. iWD will pick up the XML messages from this queue and use the information in the message to take some action such as creating a new task, completing or updating an existing task, and so on.
- A processed queue where iWD will place a copy of messages that were successfully processed.
- An error queue where iWD will place a copy of messages that were not successfully processed.
- A notification queue where iWD will place both replies to inbound requests, as well as unsolicited notification messages. The notifications received will be based on the notification filtering that is configured for the JMS Capture Point. It is possible, for example, to filter out certain types of notifications coming from iWD that may not be of interest to the source system.

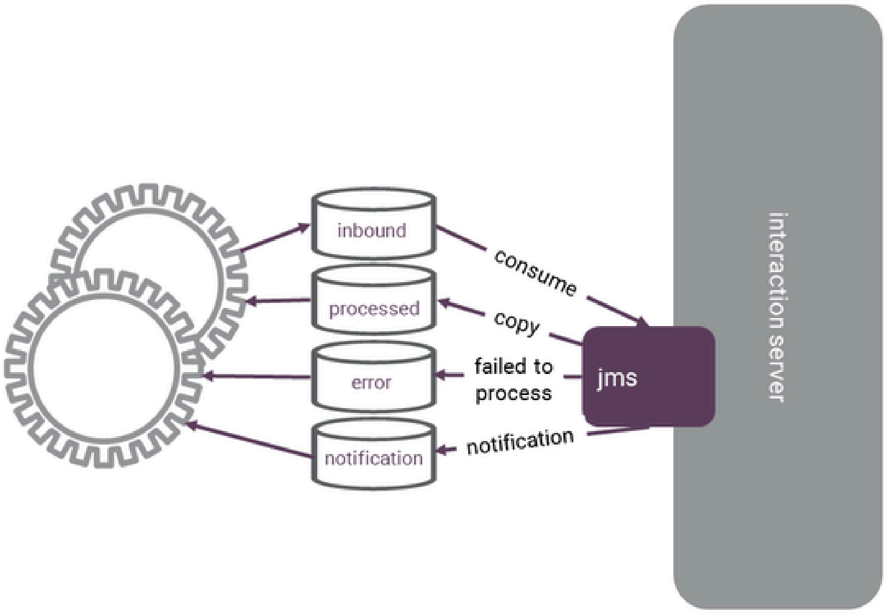


Figure 2: JMS Capture Point Queues

The source system is responsible for generating the XML messages and putting them in the inbound queue on the JMS provider. The JMS Capture Point is an excellent choice for all source systems that can support it, either directly or through some middleware such as WebSphere MQ Server or TIBCO Enterprise Messaging Server. Because it is an enterprise-class message bus, these providers that implement the JMS specification are able to leverage the persistence of the message queues. This means that if there is some service interruption on either JMS client (that is, the source system or iWD), any messages that have not yet been consumed will remain in the JMS queues to be retrieved when the JMS client is brought back into service. The JMS specification caters for this by using a correlation ID to map the outgoing request message with its corresponding reply.

The JMS Capture Point can also be configured to support SSL between the Genesys Interaction Server and the JMS provider.

Message Transformation

The JMS Capture Point supports optional message transformation. This is provided via the Groovy scripting language. That is, Groovy scripts may be written that describe to iWD how the source system's native XML message format should be converted into a message format that can be understood by iWD, and vice-versa for message replies and notifications coming from iWD back to the source system. For customers who want to use the message transformation feature, Interaction Server must load a JVM, which requires that the Java JDK (latest release of 5 or 6) be installed on the Interaction Server-accessible host.the

For more details about message transformation, please read the Transformation topics in the eServices Integrated Capture Points Guide.

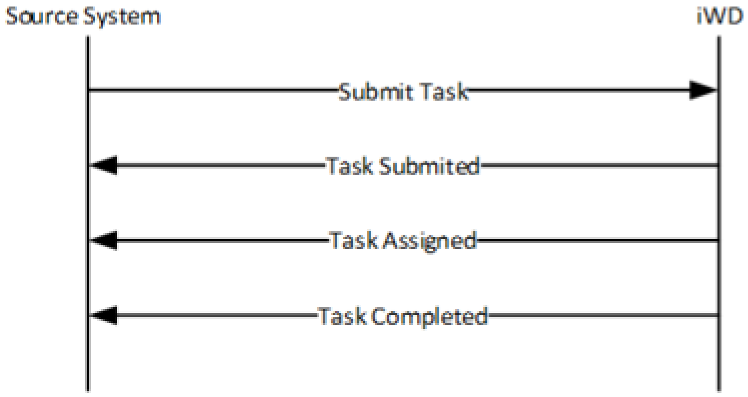
Example JMS Capture Point Request

Here is an example of a simple request using the iWD JMS Capture Point, which instructs iWD to create a new task.

```
<?xml version="1."? encoding="UTF-8"?>
<interaction operation="submit">
<properties>
<ExternalId>SomeExternalId</ExternalId>
<CustomerSegment>Gold</CustomerSegment>
<CustomerInfo>
<FirstName>William</FirstName>
<LastName>Bell</LastName>
</CustomerInfo>
</properties>
</interaction>
```

Simple JMS Capture Point Flow

Here is an example of a simple JMS Capture Point message flow.



Integration

To use the JMS Capture Point, the source system must be able to create XML files and put them into a particular JMS queue. The XML files must contain messages that are in an iWD-compatible (or Interaction Server-native) XML format. If not, then Groovy transformation scripts (inbound and outbound) must be created to transform the inbound XML message into the Interaction Server message format, and to transform the outbound reply or notification XML message from Interaction Server into the source system’s XML format. There are sample Groovy scripts provided with Interaction Server that can serve as starting points, if message transformation is required.

At a minimum, the source system must be integrated with the JMS provider to allow the source system to put the XML messages into the inbound queue. Optionally, the additional JMS queues can be leveraged as part of the iWD integration. The actual configuration of queues on the JMS provider should be trivial; the additional effort will be to read the messages from those queues in order to take some action on those replies or notifications, within the source system’s workflow.

Further Reading

There is more detailed technical reference information in the JMS Capture Point section of the eServices Integrated Capture Point Guide.