



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.

Integrated Capture Points Guide

Inbound Messages

5/8/2025

Inbound Messages

Operation Elements and Root Element

An inbound XML document can contain multiple operations, but only a single root element. For maximum flexibility the name of the root element can be anything, and it is not taken into account. The transformation scripts use `messages` as the name of the root element. If the document contains a single operation, this operation can be a root element. Any operation item is specified by an `interaction` element. The following are sample XML messages:

Sample 1

```
<?xml version="1.0" encoding="UTF-8"?>
<interaction operation="submit" ExternalId="SomeExternalId"/>
```

Sample 2

```
<?xml version="1.?" encoding="UTF-8"?>
<messages>
  <interaction operation="submit" ExternalId="ExternalId2"/>
  <interaction operation="submit" ExternalId="ExternalId3"/>
</messages>
```

Sample 3

```
<?xml version="1.?" encoding="UTF-8"?>
<myinteractions>
  <interaction operation="submit" ExternalId="ExternalId4"/>
  <interaction operation="submit" ExternalId="ExternalId5"/>
</myinteractions>
```

Operations

The operation type is specified by the `operation` attribute and can be one of the following:

- `submit`—Submit a new interaction
- `update`—Update or change interaction properties
- `hold`—Hold the interaction
- `resume`—Resume the interaction
- `stop`—Stop or delete the interaction
- `getinfo`—Request interaction properties

Properties Element

The `properties` element, which should be a direct child of the `interaction` element, specifies the interaction properties that are needed to perform the operation.

- For `submit`, the `properties` element specifies all of the interaction properties including any user data or custom properties. It also specifies standard attributes such as the tenant and queue to which interactions are submitted. Any attribute can have a default value specified in the capture point configuration.
- For `update`, the `properties` element specifies properties that need to be changed. This might include the `Queue` property, in which case the interaction will be moved into the specified interaction queue. For the update operation, configured default values are not used, and the attribute `InteractionId` or `ExternalId` must be specified.
- For `hold`, `resume`, and `stop`, the only attribute required or processed is `InteractionId` (or `ExternalId`), which specifies the interaction that is to be held, resumed, or stopped.

For simplicity, any child element of the `properties` element can be specified as an attribute of the `interaction` element. For example, to hold an interaction the following `interaction` element can be used:

```
<?xml version="1.0" encoding="UTF-8"?>
<interaction operation="hold" InteractionId="itx00777"/>
```

Interaction Server supports key-value lists (of any depth) as interaction properties. To specify such attributes, natural XML structure is used. Note the `CustomerInfo` group of properties in the following example:

```
<?xml version="1.0" encoding="UTF-8"?>
<interaction operation="submit">
  <properties>
    <ExternalId>SomeExternalId</ExternalId>
    <CustomerSegment>Gold</CustomerSegment>
    <CustomerInfo>
      <FirstName>William</FirstName>
      <LastName>Bell</LastName>
    </CustomerInfo>
  </properties>
</interaction>
```

Interaction Server supports spaces and some special characters in interaction property names. To allow for this in XML messages, any property can have a "real" name specified as a `name` attribute. For example:

```
<?xml version="1.0" encoding="UTF-8"?>
<interaction operation="submit">
  <properties>
    <property name="First Name">William</property>
    <property name="Last Name">Bell</property>
  </properties>
</interaction>
```

The following is a list of Interaction Server's predefined properties and their meanings. Custom properties can also be specified and attached to the interaction. `<InteractionId>`—Interaction Identifier. Can be omitted and generated by Interaction Server. `<ParentId>`—Parent interaction Identifier. `<ExternalId>`—Identifier used by the external system. `<TenantId>`—Tenant Identifier. `<MediaType>`—Interaction media type. `<InteractionType>`—Inbound, Outbound, or Internal.

<InteractionSubtype>—Interaction subtype, selected from the list defined for the tenant.
<IsOnline>—The interaction is (1) or is not (0) online. <IsHeld>—The interaction is (1) or is not (0) on hold. <Queue>—Name of the queue in which the interaction is initially placed. <Workbin>—Initial workbin name; optional. <WorkbinAgentId>—Initial workbin agent ID.
<WorkbinAgentGroupId>—Initial workbin agent group ID. <WorkbinPlaceId>—Initial workbin place ID. <WorkbinPlaceGroupId>—Initial workbin place group ID. <ReceivedAt>—Date and time received; format is YYYY-MM-DD HH:MM:SS. <Priority>—Initial interaction priority. <ServiceType>—Service type. <ServiceObjective>—Service objective in seconds.

Delete Element

The `delete` element, which must be a direct child of the `interaction` element, is used only for the `update` operation, and specifies the names of the properties that are to be deleted. For example:

```
<?xml version="1.0" encoding="UTF-8"?>
<interaction operation="update" InteractionId="itx00777">
  <properties>
    <property name="Last Name">Ball</property>
  </properties>
  <delete>
    <property name="Middle Name"/>
  </delete>
</interaction>
```

Reason Element

The `reason` element, which must be a direct child of the `interaction` element, can specify the reason for the operation. This attribute is optional and can be used with the `hold`, `resume` and `stop` operations. The `reason` element has the attributes `name` and `description`. For example:

```
<?xml version="1.0" encoding="UTF-8"?>
<interaction operation="hold" ExternalId="Loan1022011-02">
  <reason name="AwaitingInfo" description="Waiting for credit history report"/>
</interaction>
```