



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.

Composer Help

Queue Interaction Block

Queue Interaction Block

Contents

- **1 Queue Interaction Block**
 - 1.1 Use Case
 - 1.2 Name Property
 - 1.3 Block Notes Property
 - 1.4 Exceptions Property
 - 1.5 Condition Property
 - 1.6 Logging Details Property
 - 1.7 Log Level Property
 - 1.8 Enable Status Property
 - 1.9 Interaction Queue Name Property
 - 1.10 Interaction ID Property
 - 1.11 Hints Property
 - 1.12 Detach Property
 - 1.13 Detach Timeout Property
 - 1.14 Pass Context Property
 - 1.15 ORS Extensions Property

Use this block to place a multimedia (non-voice) interaction into an existing queue created with the [Interaction Queue](#) block. The generated SCXML code instructs Universal Routing Server to request Interaction Server to move the current interaction into the specified queue. You can select an existing interaction queue from within this block or specify a variable that contains the name of an interaction queue. You can also send an interaction to a queue in another IPD within the same Project. The selected interaction queue appears as a [Queue Reference block](#) in the [interaction process diagram](#). Important Note! Each interaction path in a workflow for multimedia interactions should end with one of these blocks: [Stop Interaction](#), [Queue Interaction](#), or [Route Interaction](#). Also see information on the [App Terminate Ixn On Exit](#) variable.

Use Case

The logic of a routing workflow may determine some attributes of an interaction, such as by looking at the Subject line of an inbound e-mail, and then use different interaction queues as a method of segmenting these different types of interactions. You could use the [Branching block](#) for this purpose. The Queue Interaction block has the following properties:

Name Property

Find this property's details under [Common Properties](#).

Block Notes Property

Find this property's details under [Common Properties](#).

Exceptions Property

Find this property's details under [Common Properties](#).

Condition Property

Find this property's details under [Common Properties](#).

Logging Details Property

Find this property's details under [Common Properties](#).

Log Level Property

Find this property's details under [Common Properties](#).

Enable Status Property

Find this property's details under [Common Properties](#).

Interaction Queue Name Property

Use this property to specify the queue where the interaction is to be placed.

1. Click under Value to display the  button.
2. Click the button to open the Interaction Queue dialog box. Your dialog box may include existing IRD business processes with associated interaction queues underneath.
3. Select an interaction queue to which the incoming interaction has to be sent.
4. Click OK.

Starting with Release 8.1.410.14, you can use a queue defined in referenced Projects. For more information, see [AddingQueues](#).

Starting with Release 8.1.440.18, the Interaction Queue property supports selecting a queue dynamically through a variable. The selected variable will be used in the generated code in the destQueue variable 'id' option. Example:

```
<state id="_reserved_QueueInteraction1_redirect">
    <onentry>
        <script>
            var destQueue = {'type':'IQ','id':var0};
        </script>
        <ixn:redirect detach="true"
requestid="App_QueueInteraction1['requestid']" interactionid="system.InteractionID"
from="system.InteractionID" to="destQueue"/>
    </onentry>
</state>
```

Interaction ID Property

Set to a meaningful value or keep the default value, which is the system variable InteractionId. Can be used for "interaction-less" processing for scenarios where the InteractionId variable is not automatically initialized, but instead must wait for an event. An example would be an SCXML application triggered by a Web Service that does not add an interaction. Background: Previous to 8.1.1, Composer did not expose an Interaction ID property. Instead, when ORS started processing an interaction, a generated SCXML application automatically initialized the system variable,

InteractionId. This variable was then used internally by Routing and certain eServices blocks when interacting with ORS. With the introduction of support for Interaction-less processing, you can now define a specific event (**IPD** Diagram Wait For Event property) to initialize InteractionId, or not define an event at all. For scenarios with an interaction (IPD Diagram/Wait For Event=interaction.present for example), you may keep the default value for the Interaction ID property. The default value is the system variable InteractionId, which is initialized automatically in this case. For other scenarios (any scenario where the system variable InteractionId is not set), you may choose to:

1. Not use blocks that require an Interaction ID
2. And/or set the Interaction ID property to a meaningful value
3. And/or assign a meaningful value to the InteractionId system variable

Hints Property

This property is for future use by Orchestration Server. Its use will be described in various action elements reference in the Orchestration Server wiki.

Detach Property

Controls whether the Orchestration Platform should <detach> an interaction from the current session before queueing it. When this property is set to true, the interaction is detached from the current session.

Note: A Project properties option, **Interaction Detach**, in the Orchestration Options dialog can generate the detach attribute in the <ixn:redirect> tag in the Routing blocks. See [Detaching Interactions from Sessions](#).

Detach Timeout Property

Use to specify how long to attempt to <detach> if an initial attempt fails with an invalidstate error. Specify the timeout in milliseconds. If set to 0, no further attempt to detach is made. After the timeout, if the <detach> is not successful, no further attempts will be made and the block will attempt to reclaim the interaction back into the current session using <attach>.

Pass Context Property

This property accepts true/false values. When set to true and Detach is also true:

- URL built with the block name is stored into this interaction's user data (user data key name is '_composer_originating_session') just before detaching the interaction. That URL will be used by the orchestration destination session (that is the new orchestration session started to handle the interaction after it was redirected to an other routing point) to request the context of the originating

session. After the processing for this block is over, the originating session is blocked until the destination session actually reads the context. The context consists of the system and user **variables**.

ORS Extensions Property

Starting with 8.1.4, Composer blocks used to build routing applications (with the exception of the Disconnect and EndParallel blocks) add a new **ORS Extensions** property.