



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

Route Interaction Block

# Route Interaction Block

## Contents

- **1 Route Interaction Block**
  - 1.1 Use Case
  - 1.2 Name Property
  - 1.3 Block Notes Property
  - 1.4 Condition Property
  - 1.5 Detach Property
  - 1.6 Detach Timeout Property
  - 1.7 Activity Property
  - 1.8 Clear Targets Property
  - 1.9 Cut Off Time Property
  - 1.10 Enable Status Property
  - 1.11 Exceptions Property
  - 1.12 Expected Wait Time Property
  - 1.13 Hints Property
  - 1.14 Include Requests from Previous Blocks Property
  - 1.15 Interaction ID Property
  - 1.16 Logged In Only Property
  - 1.17 Logging Details Property
  - 1.18 Login Size Property
  - 1.19 Log Level Property
  - 1.20 ORS Extensions Property
  - 1.21 Pass Context Property
  - 1.22 Position in Queue Property
  - 1.23 Priority Property (Queue)
  - 1.24 Priority Property (Target Selection)
  - 1.25 Queue Poll Rate Property
  - 1.26 Queue for Existing Interaction Property
  - 1.27 Queue for Outgoing Interaction Property

- [1.28 Request ID Property](#)
- [1.29 Route Property](#)
- [1.30 Statistic Property](#)
- [1.31 Statistics Order Property](#)
- [1.32 Target Component Selected](#)
- [1.33 Target Deviation From Ideal Agent](#)
- [1.34 Target Object Selected](#)
- [1.35 Target Queue Selected Property](#)
- [1.36 Total Size Property](#)
- [1.37 Targets Property](#)
- [1.38 Timeout Property](#)
- [1.39 Virtual Queue Property](#)
- [1.40 Workbin Property](#)
- [1.41 Workbin Type Property](#)

Use this block to route a multimedia (non-voice) interaction to one or more target objects (use the [Target block](#) for voice interactions). You can also route to a target based on the value of a [Skill expression](#). An [interaction process diagram](#) associates routing targets with queues, either [interaction queues](#) or [virtual queues](#). When defining Route Interaction block properties, you have the option of selecting queues for both the existing interaction and a new outgoing interaction that may be created. You can also define a new interaction queue from within the block so you don't have to navigate away. 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

- An inbound interaction hits a virtual route point, initiating a workflow routing strategy.
- The workflow strategy looks up data for the interaction in the customer database to determine the last agent who helped this customer and to determine the intent of the customer interaction.
- Next, the workflow sets priority to the interaction based on media type (for example, e-mail or chat) and customer segment (for example, Gold or Silver). Note: Setting the priority of an interaction is a Universal Routing Server function that is not directly related to target selection, but is normally done prior to target selection as a way to segment interactions.
- If the last agent exists, the workflow routes to the agent based on variable, setting a timeout of five seconds.
- If the last agent used is unavailable (timeout exceeded), the workflow routes to an Agent Group that is properly skilled to handle this type of interaction.
- There is an Escalation interaction queue configured as an outgoing interaction queue, so the agent can select this interaction queue from the desktop application in case he cannot handle the interaction himself and he needs to escalate it.

The Route 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](#).

## Condition Property

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

## Detach Property

Controls whether the Orchestration Platform should <detach> an interaction from the current session before routing to reserved targets. 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>.

## Activity Property

Optional. Click the down arrow and select the variable that contains the Activity to be used for Workforce Management-based routing.

## Clear Targets Property

Optional. Select **true** if targets listed in the object should be retained after the interaction moves on through the workflow and encounters other Route Interaction object. For more information, see the [Clear Targets](#) description for the voice interaction Target block.

## Cut Off Time Property

Optional. Click the down arrow and select the variable that contains the cut-off time (in seconds) that defines the time window in which a potential target must be assigned to the activity defined in the Activity property above.

## Enable Status Property

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

## Exceptions Property

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

## Expected Wait Time Property

To use this optional property, click the down arrow under Value and select a variable to contain the expected wait time that the interaction will be given in the queue. The selected variable will be populated with the respective output value obtained from the operations described next.

Starting with 8.1.4, the Target block and Route Interaction block provide an integrated mechanism to playback the estimated waiting time to customers, including defining the playback repeat interval. To implement this mechanism, the Target and Route Interaction blocks have new properties (variables) to capture the result of `queue.query.done`.

- [Expected Wait Time](#)
- [Login Size](#)
- [Position in Queue](#)
- [Request ID](#)
- [Total Size](#)
- [Queue Poll Rate](#)

If any of the above five properties are set, the application does a `queue:query` after the `queue.submit.requestid`.

The `queue:query` is repeated every x secs until `queue.submit.done` is received. There is no additional output.

The variables used for the five properties can be used in a Busy Treatment block, such as [Play Application](#). Or you may put the Target block in a parallel leg and use the variables (for example, to speak the estimate wait time) in another leg.

If the call is distributed right away, then the application is presented an `error.queue.query` event (the `queue:query` is processed after the `queue.submit.done`). If nothing is done, that unexpected error will terminate the application. As a result:

- When assigning a variable to any of the above five properties, you will be prompted to automatically add a target-less exception for `error.queue.query`.
- Diagram validation will show a warning if one of the five properties is set and no handler is defined in the Target block (or one of its parents, // or the [Entry block](#)).

### Hints Property

This property is for use by Orchestration Server. Its use is described in various action elements referenced in the *Orchestration Developer's Guide*.

It can be used to specify extension data in Treatment blocks. To do this, use an ECMAScript block to create an ECMAScript object and assign properties to it. Then specify this object in the Hints property by selecting the variable whose content is a JSON object. This object is passed to ORS at runtime.

For example, define variable myExtensions and myHints and set them as shown below in an ECMAScript block.

```
myExtensions={'NO_ANSWER_TIMEOUT':'10','NO_ANSWER_OVERFLOW':'recall','NO_ANSWER_ACTION':'notready'};
myHints = { 'extensions' : myExtensions }; Then specify myHints as the value of the Hints property in the Route Interaction object.
```

### Include Requests from Previous Blocks Property

This property controls whether the block should transition if a target from previous Target block is selected even though it may not be specified in the current block. Set it to true for cascaded target lookups. If set to false, the block will wait until a target specified in the current block is selected for routing.

### 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 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

## Logged In Only Property

Select **true** or **false** to indicate whether only logged in agents can pull interactions out of this Workbin. Use to prevent logged out agents from pulling interactions. Selecting true = logged in agents only.

## Logging Details Property

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

## Login Size Property

Select the variable to contain the number of agents logged in that can be targeted for the request (you must have URS 8.1.3+). The selected variable will be populated with the respective output value obtained from the operations described in the [Expected Wait Time](#) property.

## Log Level Property

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

## 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.

## 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](#).



## Position in Queue Property

Click the down arrow under Value and select the variable to contain the position that the interaction will be given in the queue. The selected variable will be populated with the respective output value obtained from the operations described in the **Expected Wait Time** property.

## Priority Property (Queue)

To use this optional property, click the down arrow under Value and select the variable, which contains a value expression returning the priority that the interaction will be given in the queue. For more information, see <http://www.w3.org/TR/scxml/#ValueExpressions>.

## Priority Property (Target Selection)


To use this optional property, click the down arrow under Value and select the variable, which contains a value expression returning the priority that the interaction will be given during target selection.

## Queue Poll Rate Property

Select the variable to contain how frequently the block will poll for latest values and populate selected variables.


## Queue for Existing Interaction Property

Optional. To specify a queue for an existing interaction:

1. Click under **Value** to display the  button.
2. Click **Add** to open the Queue for Existing Interaction dialog box. Do one of the following:
  - If you are **connected to Configuration Server**, select Configuration Server. Select the interaction queue from the **Value** field. This is the interaction queue to which the incoming interaction has to be sent. Queues listed here were previously defined with the **Interaction Queue** block.
  - Select **Literal** and enter the name of the interaction queue in the Value field.
  - Select a variable (introduced in 8.1.420).
3. Click **OK**.

## Queue for Outgoing Interaction Property

Use to create a new interaction. Only one new interaction may be created. The agent must close the existing interaction with no further processing allowed.



1. Click under **Value** to display the  button.
2. Click **Add** to open the Queue for Outgoing Interaction dialog box. Do one of the following:
  - If you are **connected to Configuration Server**, select **Configuration Server**.  
Select the interaction queue from the **Value** field. This is the interaction queue to which any new interaction generated has to be sent. For example, the agent might create a new e-mail to a supervisor as a result of handling another interaction. Queues listed here were previously defined with the **Interaction Queue** block.
  - Select **Literal** and enter the name of the interaction queue in the **Value** field.
  - Select a variable (introduced in 8.1.420).
3. Click **OK**.

## Request ID Property

Set to a meaningful value or keep the default value, which is the system variable system.ANI. The selected variable will be populated with the respective output value obtained from the operations described in the **Expected Wait Time** property.

## Route Property



Use this property to set the SCXML <queue:submit> route attribute.

1. Click under Value to display the  button.
2. Click the  button to open the Route dialog box.
3. Select one of the following:
  - **Variable**. Then for **Value**, select the variable that contains either True (default) or False.
  - **Literal**. Then for **Value**, enter either **True** or **False**.

When set to false, the functional module will not attempt to route the associated interaction.

## Statistic Property

Optional. If you wish to route based on the value of a statistic:

1. Click under Value to display the  button.
2. Click the  button to open the Statistic dialog box.
3. Select one of the following:
  - **Literal.** For Value, you can write in a URS-predefined statistic or a custom statistic created with **Statistics Builder**.
  - **Variable.**
  - **Configuration Server** to select a statistic. In order to select a statistic, you must be **connected to Configuration Server** and have set the option to use **URS Predefined statistics**.
4. Click OK when through in the dialog box.

The URS predefined statistics are described in the [Universal Routing 8.1 Reference Manual](#).

The statistic you select is used by Universal Routing Server to determine which target to route the interaction to if more than one target is available. After defining a complete set of available agents (taking agent capacity rules into consideration, if configured), URS applies the selection criteria specified in the Target block, which can include using the minimum or maximum value of the statistic (see Statistics Order property).

## Statistics Order Property

Optional. This property can work with the Statistics property. Select **Max** or **Min** to specify whether the interaction should be routed to the target with the maximum or the minimum value of the statistic.

## Target Component Selected

Select a variable containing the agent-level target to which the interaction was routed or should be routed to definitively.

- If the target specified in `<submit>` and selected for routing is of type Agent, Place, Queue, or Routing Point, this contains the target itself.
- If the desired target type is Agent Group, Place Group, or Queue Group, the function returns the agent, place, or queue from the corresponding group to which the interaction was sent.

The target format is `Name@StatServerName.Type`. The selected variable will be updated with the target information after receiving a `queue.submit.done`.

## Target Deviation From Ideal Agent

New with 8.1.410.14. Select a variable to hold the result of calling the Universal Routing Server (URS)

TargetListSelected function to calculate a value indicating an agent target's deviation from **ideal agent** to handle an interaction. When the property is assigned to a variable, Composer makes a call to URS right after getting `queue.submit.done`:

```
'urs/call/@' + system.InteractionID + '/func/TargetListSelected?params=[]'
```

URS returns a string similar to the sample below:

```
return:ok|dn:7001|rdn:7001|switch:SIP_Switch|agent:KSippola|place:SIP_Server_Place1|target_location:
```

If the **Set Ideal Agent block** is used before the Route Interaction block, the URS reply includes a mismatch key-value pair. The below sample was obtained with: `SetIdealAgent[Billing > 6]` and the target with `Skill Billing == 5`).

```
return:ok|dn:7001|rdn:7001|switch:SIP_Switch|agent:KSippola|place:SIP_Server_Place1|target_location:
```

In this sample, the Target Deviation From Ideal Agent variable is assigned the (integer) value (i.e., 2) of the mismatch key-value pair when available, undefined otherwise.

## Target Object Selected

Select a variable containing the high-level target (one that you specify in a `<submit>`) to which the interaction was routed or should be routed to definitively. If a skill expression is used, the function returns: `?SkillExpression@statserver.GA` or even `?GroupName:SkillExpression@statserver.GA`. The target format is `Name@StatServerName.Type`. The selected variable will be updated with the target information after receiving a `queue.submit.done`.

## Target Queue Selected Property

Select a variable containing the DN and the switch name of the target to which the interaction was routed or should be routed to definitively. The selected variable will be updated with the target information after receiving a `queue.submit.done`.

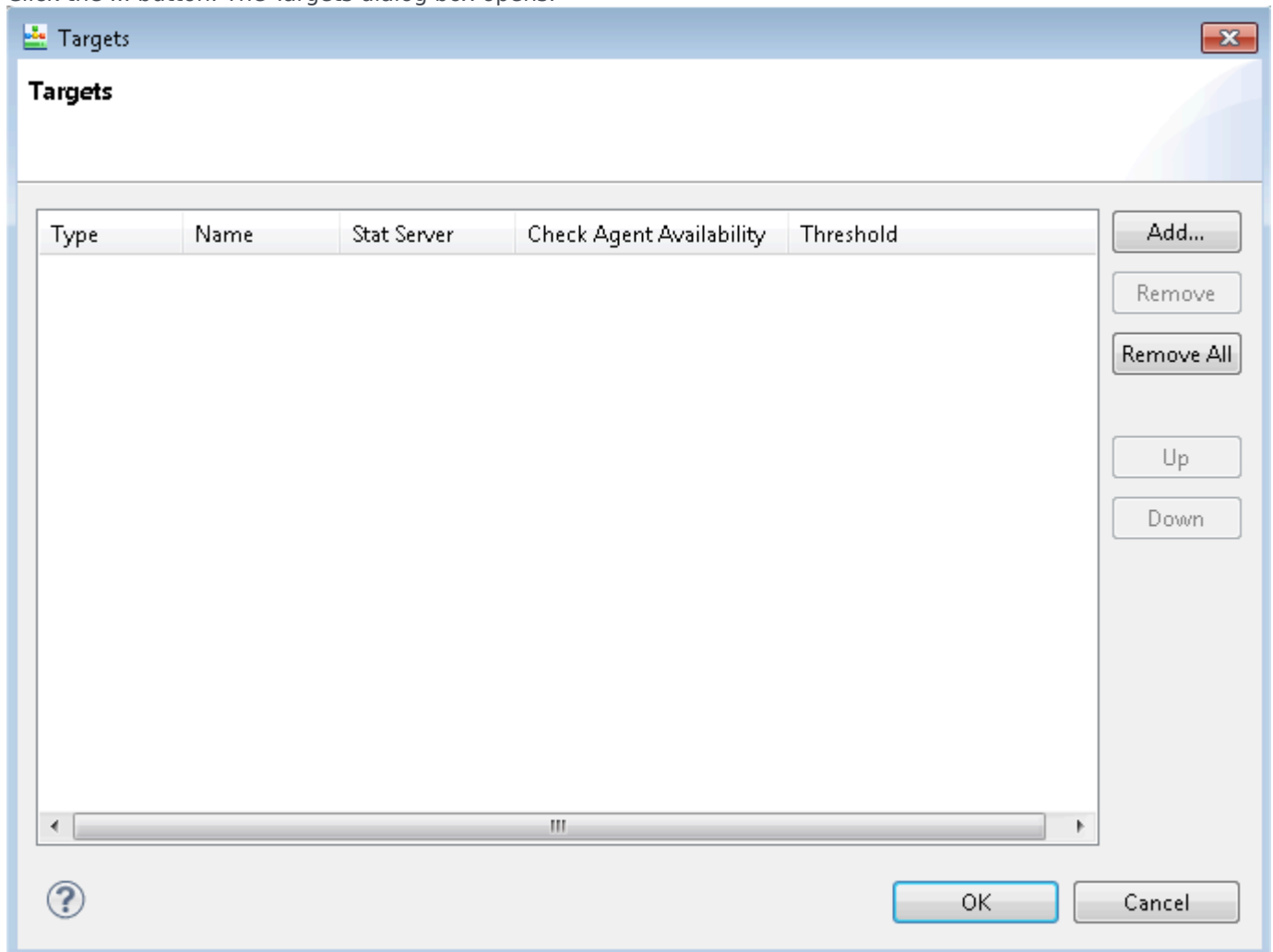
## Total Size Property

Select the variable to contain the total number of agents that can be targeted for the request. The selected variable will be populated with the respective output value obtained from the operations described in the **Expected Wait Time\_Property**. You must have installed Universal Routing Server 8.1.3+.

## Targets Property

Use this property to specify routing targets and for percentage or conditional routing.

1. If you have not already done so, [connect to Configuration Server](#).
2. Opposite the Targets property, click under **Value**.
3. Click the ... button. The Targets dialog box opens.



The figure above shows the dialog box when **Min** or **Max** is selected for the [Statistics Order](#) property. In this case, a Stat Server column appears. If you select **Percentage** for the Statistics Order property, a **Weight** column appears instead of a Stat Server column (see Statistics Order property above).

4. Click **Add** in the Targets dialog box.
5. Click under **Type** to display a down arrow.
6. You can route based on various criteria. Click the down arrow and select the target Type (defined in Genesys Administrator): Agent, AgentGroup, **LCA**, Place, PlaceGroup, Skill, List object or Variable. At

runtime, the application reads the key-value pairs in the targets section. The name part of the key-value pair can be used at your convenience. Only the value part of the key-value pair is read by the application and must be defined as indicated under <targetid> in the *Orchestration Developer's Guide*, [Parameter Elements](#). For example, a valid value is: ksippo@.A

7. Click under **Name** to display to bring up a dialog box. Targets of the selected **Type** appear for selection.
8. Select the name of routing target and click **OK**. You have the option to add another target.
9. If applicable, select a **Stat Server**. Or, if you selected **Percentage** for the Statistics Order property, enter the **Weight** column.
10. Starting with 8.1.440.18, you can check agent availability in the Route Interaction block before executing <queue:submit> by selecting true for **Check Agent Availability**. This "skip targets" feature can improve the efficiency of finding an agent by enhancing or relaxing the target criteria without waiting until the routing timeout is reached in the Route Interaction block. If multiple/mixed target types are used in the dialog, the skip target-enabled target types will take precedence and will cause Route Interaction block execution to be skipped if no agent is available. In this case, a warning message appears stating that the Route Interaction block uses different target types with skip targets enabled.
11. The Route Interaction block adds a new "Skip Target" output in the workflow diagram if **Check Agent availability** is enabled for one of the target types in the block. **Note:** This feature is not supported for the following target types: List Objects, Skill expressions, and Variables.
12. Starting with 8.1.440.18, you also have the option of using the [threshold functions](#) for conditional routing. You can use Threshold to define additional conditions the target must meet to be considered as valid routing target. Click the button under Threshold to open [Expression Builder](#) where you can create a threshold expression. See [Creating Threshold Expressions](#) in the Target block topic. Threshold is not supported for the following target types: List Objects and Variables.
13. Click **OK** when through in the Targets dialog box.

### Important



The option of routing an interaction to the last called agent (LCA) is available only from release 8.1.510.12

## Timeout Property

Optional. Enter an integer to specify the time in seconds an interaction waits for an available target or keep the default of 90 (added in 8.1.440.18). If the timeout expires before one of the targets is available, the interaction is routed to the error port (if the exception property is configured for the block).



## Virtual Queue Property

Optional. A virtual queue is a logical queue, not a physical queue. Interactions can be queued to a virtual queue if the specified targets are unavailable. To select a virtual queue.

1. Click under **Value** to display the  button.
2. Click the  button to open the Virtual Queue dialog box.
3. Select an Alias, Switch, and Number. For more information on these fields, start with the Framework Configuration Manager Help.
4. Click OK.

## Workbin Property

Use this property if you wish to route this interaction to an agent workbin. To select a workbin:

1. Click under Value to display the  button.
2. Click the  button to open the Workbin Properties dialog box.
3. Select a workbin previously defined with the **Workbin block**.
4. Optional. Click the **Show Unpublished Workbins** box.
5. Click **OK**.

Starting with 8.1.440.18, Composer adds support for dynamic Workbin selection.

- Composer adds variable support in Route interaction Block, Workbin property.
- A new Workbin Type property is added.
- You can use variables in the Route Interaction block Targets property when the Workbin property is used.

## Workbin Type Property

This property, added in 8.1.440.18, works along with the Workbin property. See the Workbin property above. Select one of the following from the dropdown: **Agent, Agent Group, Place, Place Group**.

- If a Workbin block object is selected in the Workbin property, this property is auto-populated with the Workbin Block Owner property value.
- If a variable is selected in the Workbin property, you can edit the Workbin Type property to set the corresponding Owner type.