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

Workbin Block

4/30/2025

Workbin Block

Contents

- 1 Workbin Block
  - 1.1 Input and Output Ports
  - 1.2 Name Property
  - 1.3 Block Notes Property
  - 1.4 Object Name Property
  - 1.5 Interaction Queue Property
  - 1.6 Views Property
  - 1.7 Description Property
  - 1.8 Display Name Property
  - 1.9 Enabled Property
  - 1.10 Order Property
  - 1.11 Owner Property
  - 1.12 Enabled
  - 1.13 Deleting Workbins

This block is used only for multimedia workflows. Use this block in an interaction process diagram to define a temporary storage area called a workbin, which is accessible from the agent desktop. You can then use the Workbin property in the Route Interaction block when redirecting interactions for continued processing.   Internally, a workbin has a queue and can support views. The workbin owner (agent, agent group, place, or place group) can view contents of the workbin and pull interactions out in any order. An agent associated with a workbin may get a notification when interactions are put into the workbin.

Input and Output Ports

A Workbin block:

- Does not have any input ports. You add Interactions to a workbin via the Route Interaction block in a workflow diagram.
- Has only one outport, which can be connected to a Workflow block in the IPD. Floating workbins are also allowed that are not required to be connected to Workflow blocks. This allows interactions routed to workbins to remain in workbins until they are pulled out by agents

The Interaction Workbin block has the following properties:

Name Property

Use this property to define the name of the workbin, which will appear on the block in the IPD (this is not the Configuration Database name). Find this property's details under Common Properties.

Block Notes Property

Click under Value to display the ... button. Enter text to describe the block.

Object Name Property

A unique Configuration Server Object Name will be created once a Publish operation is executed. No updates to the database are created until you invoke the Publish operation. Note: If you rename the block after its corresponding CfgScript object is created in Configuration Server, the original published object name in Configuration Server remains unchanged. For more information, see Publishing Updates.

Interaction Queue Property

Specifies the queue to be used with this workbin. This may be an existing queue in the IPD or it can be a private queue exclusive to this workbin. Multiple workbins can use the same queue.

1. Click under Value to display the ... button.
2. Click the ... button to open the Workbin Interaction Queue dialog box.
3. Select one of the following:
    - Use **Private Queue** for a queue exclusive to this workbin.
    - Use **Queue** to select an interaction queue already defined in this IPD.
4. Click OK.

Views Property

A workbin can have one or more views defined in it. Each view represents an exit channel from the workbin (or workbin’s queue). The criteria defined in the view must be satisfied before an interaction can exit out from the view. You can also use a view to enable certain types of interactions to be processed earlier than others. Note: Irrespective of the number of views defined in the Workbin block, the block will have only one outport and will feed interactions to only one workflow. To define a view:

1. Click under Value to display the ... button.
2. Click the ... button to open the View Properties dialog box.
3. Click Add to display Main, Parameterized Conditions, and Segmentation tabs. For information on these tabs, see Interaction Queue Views.
4. After you complete the applicable fields in these tabs, click OK to close the View Properties dialog box. Each view will be created as a separate CfgScript object of type Interaction Queue View.

Description Property

Enter text that describes the workbin.

Display Name Property

Enter a display name for the workbin to appear in the interaction process diagram.

Enabled Property

Select **true** or **false** to enabled or disabled this queue as ready to accept interactions. A queue may be enabled during design time. Note: Setting this property is equivalent to enabling or disabling the Workbin object from Genesys Administrator or Configuration Manager.

Order Property

Use this property to specify the WHERE clause of a SQL statement, which will determine the order in which interactions will be sorted in the workbin.

Owner Property

A workbin may be owned by (associated with) one or more agents. Click the down arrow and select one of the following:

- **Agent**
- **Agent Group**(default)
- **Place**
- **Place Group**

Agents associated with a workbin may get a notification if an item is placed in the workbin.

Enabled

This property controls whether or not a block contributes code to the application. You may wish to use this property if there is a need to temporarily remove a block during debugging or, for other reasons during development, temporarily disable a block. This saves the effort of having to remove the block and then add it back later.

Deleting Workbins

If you delete a Workbin block from an IPD in online mode, Composer displays a prompt asking if you wish to delete its Configuration objects. These will include:

- Interaction Workbin (CfgScript:: `InteractionWorkBin`)
- Interaction Queues (CfgScript::`InteractionQueue`)
- Interaction Queue Views (CfgScript:: `InteractionQueueView`)

If you indicate Yes," then the Interaction Queue object and view objects will be deleted from Configuration Server. If you indicate No," Composer will not delete these objects from Configuration Server. You must manually clean up these objects outside of Composer. In offline mode, if you delete a Workbin block from an IPD, the Configuration Server connection dialog is shown. If a connection cannot be established, object deletion will fail and you must manually delete the objects outside of Composer.