Composer Help

SCXML State block

5/10/2025

SCXML State block

Contents

Use to write custom SCXML code for Composer to include in the SCXML document that it generates based on the workflow diagram.

The SCXML State block has the following properties:

Name Property

Click under Value and enter the block name. Composer will use the name to identify the block in the diagram and as the state name in the SCXML code.

Block Notes Property

Find this property's details under Common Properties.

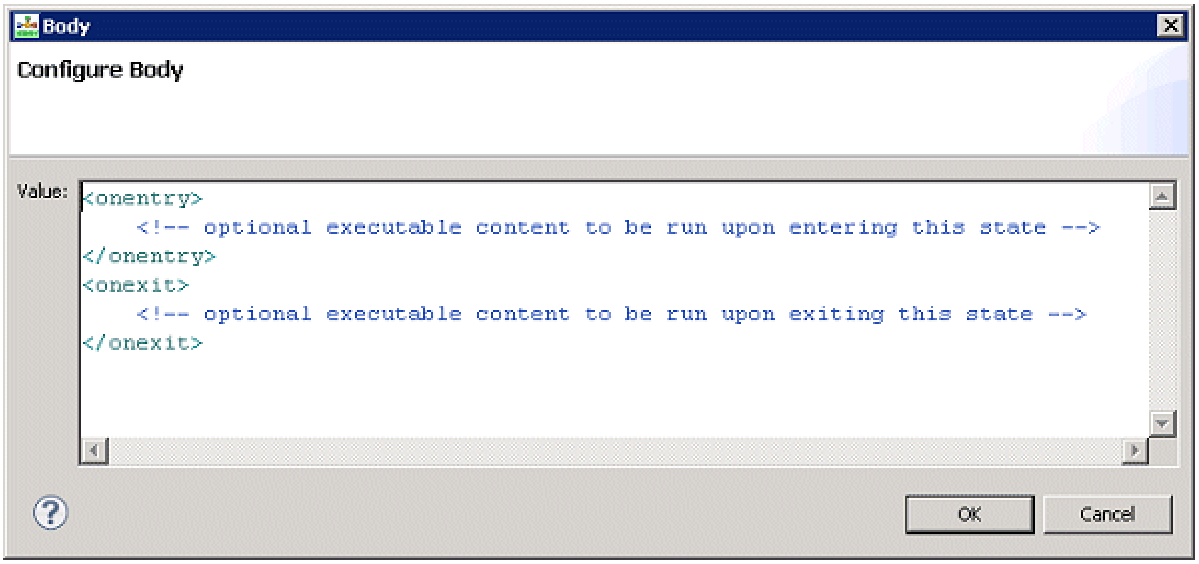
Exceptions Property

Use to configure the exception nodes, with each port being hooked up to an event configured by you and selectable using Add Custom Event. Find this property's details under Common Properties.

Body Property

This property contains all the executable content of the <state> element (<onentry>, <onexit>, <final>, ...).

1. Click opposite **Body** under Value. This brings up the ... button.
2. Click the ... button to bring up the Configure Body dialog box.



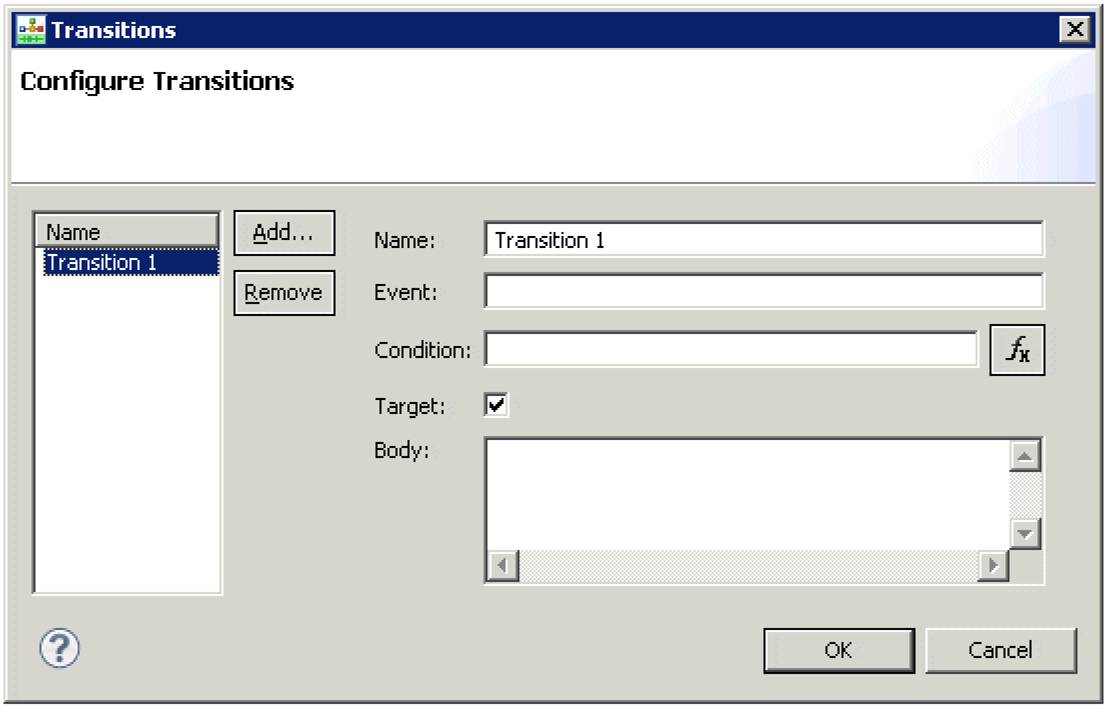
```
<onentry>
    <!-- optional executable content to be run upon entering this state -->
</onentry>
<onexit>
    <!-- optional executable content to be run upon exiting this state -->
</onexit>
```

3. Enter the executable content of the <state> element. . All content (children) of the state are editable. You also have the option of adding code to <onentry> and <onexit>.
4. When through, click **OK**. Note: The editor does not validate against the SCXML schema.

Transitions Property

Use this property to add additional outports (transitions) using the block's custom Transitions dialog.

1. Click opposite **Transitions** under **Value.** This brings up the ⋯ button.
2. Click the ⋯ button to bring up the Configure Transitions dialog box.
3. Click **Add**. The dialog box now appears as shown below.



Name: Transition 1

Target: ☑

4. For each transition, specify at least one name, event, condition, or target (you are not required to complete all three fields). An output port is created for every transition

   - **Name**—Composer uses the name of the transition to label the outport.
   - **Event**—Use to select the event that will trigger this transition.
   - **Condition**—The guard condition for this transition. The transition is selected only if the condition evaluates to true.
   - **Target**—If true, an output port is created and the user can connect it to the block this transition will transition to when it is executed. If false, the transition will not cause a change in the state configuration when it is executed. The executable content contained in the transition will still be executed, so the transition will function as a simple event handler.

If a target is set, an outport for that transition will appear and you can connect it to other blocks. If a target is not set, an outport for that transition does not appear; in this case, you can add some SCXML code to handle the event.

When through in the dialog box, click **OK**.

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.