Composer Help

ECMAScript Block

5/10/2025

ECMAScript Block

Contents

Orchestration Server (ORS) 8.0+ supports SCXML plus ECMAScript as a routing language. While the core SCXML provides State Chart functionality, you can specify ORS-specific instructions, such as conditions that can be used for routing decisions, in the form of ECMAScript. The Script property brings up Composer's Expression Builder for creating those conditions in the form of expressions. Use the ECMAScript block to build an ECMAScript expression. Notes:

- The ECMAScript block supports general ECMAScript in addition to ORS-specific ECMAScript functionality.
- If the Composer Project contains a folder at include/user, then any files with extension .js will be included in the generated SCXML. This allows you to write custom ECMAScript and include it in the application.
- To support creating multiple views per interaction queue, the ECMAScript block is available when creating an IPD.

The ECMA Script 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.

- For callflows, invalid ECMAScript expressions may raise the following exception event: `error.semantic`
- For workflows, invalid ECMAScript expressions may raise the following exception events: `error.script.SyntaxError` and `error.script.ReferenceError`

You can use custom events to define the ECMAScript exception event handling.

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.

Script Property

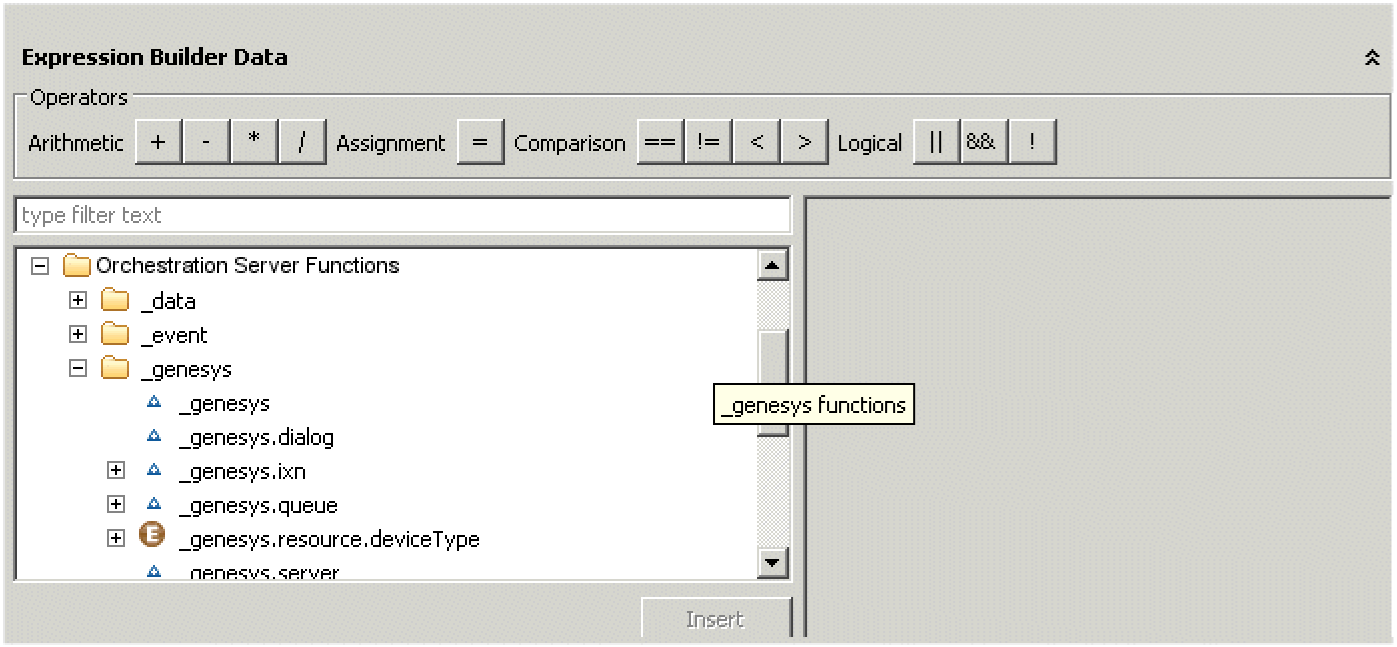
To create an ECMAScript expression in Expression Builder:

1. Click opposite Script under Value. This brings up the button.
2. Click the button to bring up Expression Builder.

Expression Builder gives access to various categories of data, which can be used in expressions. To create an expression, follow the instructions in the Creating Expressions topic.

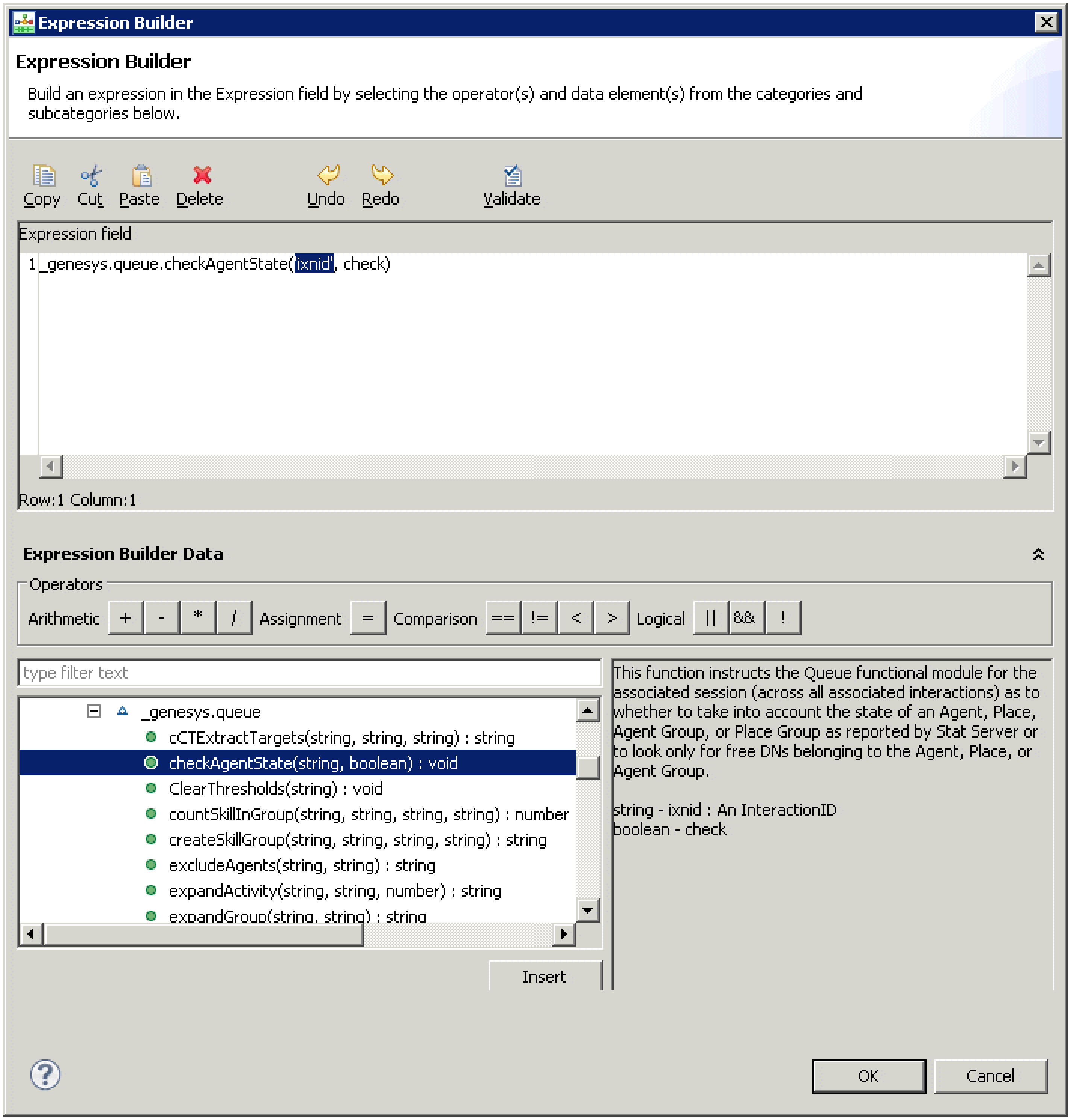
Using Genesys Functional Modules

Assume you expand Orchestration Server Functions in step 2 above. The lower Expression Builder Data area appears as shown below.



ECMAStart.gif

Orchestration Server Functions shows different categories of Genesys-supplied Functional Modules. For more information, see the Orchestration Server information on the Genesys Documentation Wiki. For example, assume you double-click genesys.queue.checkAgentState(check). Expression Builder now appears as shown below.



```
_genesys.queue.checkAgentState('ixnid', check)
```

```
_genesys.queue
    cCTExtractTargets(string, string, string) : string
    checkAgentState(string, boolean) : void
    ClearThresholds(string) : void
    countSkillInGroup(string, string, string, string) : number
    createSkillGroup(string, string, string, string) : string
    excludeAgents(string, string) : string
    expandActivity(string, string, number) : string
    expandGroup(string, string) : string
```

This function instructs the Queue functional module for the associated session (across all associated interactions) as to whether to take into account the state of an Agent, Place, Agent Group, or Place Group as reported by Stat Server or to look only for free DNs belonging to the Agent, Place, or Agent Group.

string - ixnid : An InteractionID
boolean - check

ECMA1.gif

In this case, the genesys.queue module implements the target selection functionality of URS (finding resources for interactions and delivering interactions to the resource). The Interaction Routing Designer Function object precursor (not necessarily equivalent) is described in the Universal Routing

8.1 Reference Manual, CheckAgentState function. When URS executes Functional Modules, it returns events back to the instance of logic running the SCXML document that requested the action.