



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.

Orchestration Server Developer's Guide

ECMAScript

4/28/2025

Contents

- 1 ECMAScript
 - 1.1 SCXML Elements

ECMAScript

SCXML Elements

<anchor>

The <anchor> module is *not supported* by the SCXML engine. This element would otherwise be used for providing 'go back' or 'redo'-like functionality for applications.

<cancel >

TTTTFor <cancel>, either one of the attributes `id` and `sendid` may be used. However, both cannot be defined at the same time.

When using <send> to generate events, if there is an intention to cancel the event sent, it is recommended to use the attribute `idlocation` instead of `id`. The `sendid` stored at the location specified by `idlocation` may then be used in <cancel>.

When the <cancel> request has been processed, the SCXML engine will send back the "cancel.successful" event if the event was successfully removed, or "error.notallowed" if there was a problem, along with the attribute `sendid` in the event.

<data>

The following are the additional Genesys attributes for <data> element. They are strictly used to help define and administer the provisioning of this data from the appropriate source.

Attribute Details

Name	Required	Type	Default Value	Valid Values	Description
<code>_type</code>	false	NMTOKEN	data	<p>The following is the set of valid values:</p> <ul style="list-style-type: none">• <code>data</code>• <code>parameter</code>	This allows the developer to identify the data elements that are to be parameters that the platform must obtain values for when the session is initiated. Note that this does not impact the way in which the SCXML document is executed.

Name	Required	Type	Default Value	Valid Values	Description
_desc	false	string	none	Any valid string	This allows the developer to provide a description of the parameter that is to be supplied at session initiation. Note that this does not impact the way in which the SCXML document is executed.

src Attribute

The currently supported URI schema types for the src attribute are:

- HTTPS
- HTTP
- FILE

id Attribute

The value of this attribute must be a valid ECMAScript variable name. This means that variable semantics that include elements like "." (for example, foo.foo) and "-" (for example, foo-foo) are not allowed. The rule is that the variable name must be able to be processed on its own in an ECMAScript snippet. If not, then a TypeError event is generated.

For example,

Valid element

```
<data id="foo" expr="'value1'"/>
```

Invalid element

```
<data id="foo.foo" expr="'value2'"/> <!--TypeError event generated -->
```

If you need to create complex objects you can always create them with the `<script>` element as a child of the `<scxml>` element with the src attribute where the src attribute value points to a valid JSON object with a mime type of application/json.

<foreach> (Since ORS 8.1.200.40, SCXML 8.1.000.77)

This element is an extension to the W3C Working Draft 7 May 2009. However, it has been formally added to the W3C SCXML specification since the [W3C Working Draft 26 April 2011](#). `<foreach>` is an Executable Content element (like `<if>`, or `<log>`) and can be used to create iterators. The behaviour of `<foreach>` is similar to that of the C# and Perl 'foreach' construct, which traverses items in a collection. This implementation differs from the W3C specification in that the SCXML engine behaves

as though a deep copy of each item in the collection is created during iteration as opposed to a shallow copy. Nevertheless, iteration behaviour will remain unaffected by changes to the collection.

Attribute Details

Name	Required	Type	Default Value	Valid Values	Description
array	true	Value expression	none	A value expression that evaluates to an iterable collection.	The <foreach> element will iterate over a deep copy of this collection.
item	true	string	none	Any variable name that is valid in the specified data model.	A variable that stores a different item of the collection in each iteration of the loop.
index	false	string	none	Any variable name that is valid in the specified data model.	A variable that stores the current iteration index upon each iteration of the foreach loop.

<history>

The <history> element is *not supported* by the SCXML engine. This element would otherwise be used for allowing 'pause and resume' control flows.

<invoke>

The child element <content> of <invoke> is *not supported*.

<scxml>

The following are the additional Genesys attributes for the <scxml> element:

Attribute Details

Name	Required	Type	Default Value	Valid Values	Description
_type	false	string	combination	Any valid string	This is set by the developer at the beginning of the SCXML document to define what type of SCXML logic has been defined. Composer sets this property based on

Name	Required	Type	Default Value	Valid Values	Description
					the type of logic you are building. It is used for reporting purposes.
_persist	false	boolean	false true (prior to 8.1.2)	<p>The following is the set of valid values:</p> <ul style="list-style-type: none"> • true • false 	<p>This allows the developer to suppress all persistence capabilities.</p> <p>Persistence is not always desired, due to the associated performance overhead. For instance, in Orchestration, current voice-related routing strategies normally run to completion in a reasonable amount of time, and in the event of a failure, restarting the routing strategy may not be problematic. Therefore, this attribute allows sessions to suppress all use of persistence, which prevents the orchestration platform from ever persisting the session. (Note that this does <i>not</i> preclude the orchestration platform from employing other techniques, such as hot standby servers, to achieve fault tolerance for these types of session.</p>
_statePersistDefault	false	string	"may"	<p>The following is the set of valid values:</p> <ul style="list-style-type: none"> • must • may • no 	To ensure proper session persistence during High Availability recovery, the _statePersistDefault may be used as an attribute to the top-level <scxml> element.

Name	Required	Type	Default Value	Valid Values	Description
					<p>Orchestration Server uses the value of <code>_statePersistDefault</code> as the default for the <code><state></code> <code>_persist</code> attribute, if it is not specified at the <code><state></code> level.</p> <ul style="list-style-type: none">• <code>may</code>—Default value. ORS will persist the SCXML session in the entered state once the event queue becomes empty.• <code>must</code>—ORS will immediately persist the SCXML session in the entered state.• <code>no</code>—ORS will not persist the SCXML session in the entered state.
<code>_maxtime</code> (Since ORS 8.1.300.03, SCXML 8.1.300.00)	false	integer	"604800"	Any valid positive integer, inside double quotes.	<p>Specifies the maximum age in seconds that an ORS session should exist. If this age is reached, ORS shall attempt to exit the session.</p> <p>If specified, this overrides the value specified in configuration for ORS under <code>scxml/max-session-age</code>.</p> <p>To disable this feature, set the</p>

Name	Required	Type	Default Value	Valid Values	Description
					<p>_maxtime to "0".</p> <p>As of ORS 8.1.300.13, SCXML 8.1.300.13, an available Cassandra data store will be required for this functionality.</p>
<code>_microStepLimit</code> (Since ORS 8.1.300.11, SCXML 8.1.300.10)	false	integer	1000	Any valid positive integer, inside double quotes.	<p>Specifies the maximum number of microsteps allowed to be taken following the processing of one event. Subsequent transitions may arise from the processing of one event if the following transitions are eventless. If this number is reached, ORS shall attempt to exit the session. To use ORS configured default, leave <code>_microStepLimit</code> undefined. To disable this feature, set <code>_microStepLimit="0"</code>.</p>
<code>_stateEntryLimit</code> (Since ORS 8.1.300.11, SCXML 8.1.300.10)	false	integer	100	Any valid positive integer, inside double quotes.	<p>Specifies the maximum number of times that a state may be entered as the target of a transition. States entered indirectly as the result of a transition element or initial attribute are not considered for this limit (for example, ancestors of the target state that must be entered before entering the target state). If this number is reached, ORS shall attempt to exit the session. To use ORS configured default, leave <code>_stateEntryLimit</code></p>

Name	Required	Type	Default Value	Valid Values	Description
					undefined. To disable this feature, set <code>_stateEntryLimit="0"</code> .
<code>_maxPendingEvents</code> (Since ORS 8.1.300.11, SCXML 8.1.300.10)	false	integer	100	Positive integer between 30 to 100000, inclusive, inside double quotes.	Specifies the maximum number of events allowed to be queued to a session (inclusive of internal, external, delayed and undelivered events). If this number is reached, ORS shall attempt to exit the session. This feature cannot be disabled.
<code>_processEventTimeout</code> (Since ORS 8.1.300.11, SCXML 8.1.300.10)	false	integer	10000	Any valid positive integer, inside double quotes.	Specifies the maximum time allotted for the processing of the event queue. The processing of one event may lead to additional events being queued. Processing of the event queue does not complete until the event queue is empty. This feature sets an upper bound to the amount of time dedicated to processing these events. If the timeout is reached, ORS shall attempt to exit the session. To use ORS configured default, leave <code>_processEventTimeout</code> undefined. To disable this feature, set <code>_processEventTimeout="0"</code> .
<code>_sendSessionRecovered</code> (Since ORS 8.1.300.13, SCXML 8.1.300.13) <code>_recoveryEnabled</code> (Since ORS 8.1.300.12, SCXML 8.1.300.12)	false	boolean	false	The following is the set of valid values: <ul style="list-style-type: none"> • true • false 	Specifies whether or not this strategy is eligible for proactive recovery. If set to true, the session will be explicitly restored by ORS when an ORS node performs switch-over to

Name	Required	Type	Default Value	Valid Values	Description
					Primary.
<code>_debug</code> (Since ORS 8.1.300.11, SCXML 8.1.300.10)	false	boolean	false	<p>The following is the set of valid values:</p> <ul style="list-style-type: none"> • <code>true</code> • <code>false</code> 	Specifies whether or not debugging of SCXML strategy is required. When set to true, the session will save a copy of the fully assembled SCXML strategy to disk (working directory).
<code>_transitionStyle</code> (Since ORS 8.1.300.28, SCXML 8.1.300.38)	false	string	legacy	<p>The following is the set of valid values:</p> <ul style="list-style-type: none"> • <code>legacy</code> • <code>genesys</code> • <code>w3c</code> 	<p>Specifies the order in which the <code><transition></code> executable content is to be executed in the scenario where there are two or more selected transitions (only in <code><parallel></code> regions).</p> <ul style="list-style-type: none"> • <i>legacy</i> setting dictates that transitions are executed by line order (lowest line number first) • <i>genesys</i> setting orders transitions by reverse scope order. Transitions of deepest scope (most nested) are executed first. Ties for scope are broken

Name	Required	Type	Default Value	Valid Values	Description
					by lowest line number first. <ul style="list-style-type: none">w3c setting adheres to the ordering prescribed by the W3C Working Draft for SCXML. Transitions are executed in the scope order of the states which selected them. Ties for scope are broken by lowest line number first.

<log>

<log> has three attributes (expr, label, level). For attribute details, please refer to State Chart XML (SCXML): State Machine Notation for Control Abstraction W3C Working Draft 7 May 2009 (www.w3.org). As of version 8.1.200.46, specifying a level of 5 with a label of 22000 to 22020 will result in behavior equivalent to that for URS/IRD.

<state>

The following are the additional Genesys attributes, children, and behavior for the <state> element:

Attribute Details

Name	Required	Type	Default Value	Valid Values	Description
_type	false	NMTOKEN	normal	The following is the set of valid values: <ul style="list-style-type: none">normal	This allows the developer to control how the platform is to handle this state and is a place

Name	Required	Type	Default Value	Valid Values	Description
					holder for future support.
_persist	false	NMTOKEN	may	<p>The following is the set of valid values:</p> <ul style="list-style-type: none">• no• may• must	<p>Long-running sessions typically experience concentrated time windows in which active processing is performed, followed by a relatively long time window during which the system awaits follow-up by a customer or potentially by the agent. This attribute is used to indicate to the platform whether a session can or must be persisted:</p> <ul style="list-style-type: none">• no - Used to indicate a state is transitional, or is not meaningful for recovery purposes over the last persisted state.• may - The platform may persist the session in this state at its discretion. This is the default value.• must - The platform must persist the session as part of entry processing

Name	Required	Type	Default Value	Valid Values	Description
					<p>of this state (before the <onentry> elements are executed). This is used to guarantee that the session can be recovered from this point in the event of failure (that is, the ability to reenter the session at this state).</p> <p>Note: Orchestration Server uses the value of <scxml>_statePersistDefault as the default for the <state>_persist attribute, if it is not specified at the <state> level.</p>
_deactivate	false	string	no	<p>The following is the set of current valid values:</p> <ul style="list-style-type: none"> • now • no 	<p>This attribute defines whether the session that enters this state and is waiting for a transition should be immediately persisted, removed from platform memory, and marked as inactive. This attribute is valid only if the "_persist" attribute is set to "may" or "must", since only persistable sessions can be de-activated. This attribute is treated purely as a hint to the platform about how meaningful it</p>

Name	Required	Type	Default Value	Valid Values	Description
					is to persist the session.
src	<i>Not Supported</i>				
Persist/Deactivate option matrix		_persist			
no	may	must			
_deactivate	no	Persistence disabled. _deactivate attribute is ignored.	The session may be persisted in this state at the platform's discretion.	The platform guarantees that this state will be persisted upon entry.	
	now	Persistence disabled. _deactivate attribute is ignored.	The session may be persisted in this state at the platform's discretion and will be marked as inactive when waiting for transition.	The session that enters this state and is waiting for transition will be immediately persisted and marked as inactive.	

<param>

The <param> element has the following restriction:

- When using the <param> element with any action element, you must specify both the name and expr attributes. Because of this, the platform *does not support* the name attribute value as a data model location expression if the expr attribute is missing.

<transition>

The attribute anchor of the <transition> is *not supported*.

ORS Version 8.1.200.60/8.1.300.01

The behaviour of transitions in the SCXML engine is different from that which is described in the W3C Working Draft 7 May 2009 [1]. This draft spec explains the following:

The **LCA** is the innermost <state>, <parallel>, or <scxml> element that is a proper ancestor of the transition's source state and its target state(s).

During a transition, all active states that are proper descendants of the LCA are exited.

The new transition behaviour of the SCXML engine shares greater similarities with that of the W3C Working Draft 16 December 2010 [2], in that in the case of a transition whose source state is a compound state and whose target(s) is a descendant of the source, the transition will not exit and re-enter its source state. In addition, the notion of the LCA is replaced by the LCCA:

The **LCCA** is the innermost compound `<state>` or `<scxml>` element that is a proper ancestor of the transition's source state and its target state(s).

During a transition, all active states that are proper descendants of the LCCA are exited.

`<validate>`

The `<validate>` element is *not supported* by the SCXML engine. This would otherwise be used to invoke a validation of the datamodel.

`<xi:include>`

The xinclude recommendation (<http://www.w3.org/TR/xinclude/>) is used for inlining of ECMAScripts (`<script>`) and states (`<state>`). An application developer may specify scripts, states, and other content separately from the main SCXML document. The included document or fragment can be text or xml. See section below on using `<xi:include>`.

When using xinclude, the following are important considerations to keep in mind:

- Developers should ensure that the 'resolveid' attribute value is unique within a document. This is necessary when the same included document is used multiple times within the including document since, the IDs of `<state>`, `<parallel>`, and so on, must be unique across the entire document.
- Included documents must NOT transition back to states that are defined in an including document. This does not work.

Xinclude can also be used to provide a subroutine-like capability within an SCXML application by using it like a macro facility. This replaces all `<xi:include>` elements with the referenced state content during the initial document fetch and load. Once the SCXML application is fully assembled, it is compiled and validated before sessions can be created based on this application.

In addition to the considerations above, the following guidelines must be followed when using xinclude as a macro style "subroutine":

- For the included document:
 - The document must be a valid `<state>` fragment that specifies the complete behavior of the subroutine. The document can contain an `<scxml>` document, but if it does, the xinclude declaration must use `xpointer` to reference the `<state>` that is to be used as the subroutine.
 - The referenced `<state>` can be a simple or a compound state. If it is a compound state, it must define `<initial>` as well as `<final>` states.
 - An atomic state must use `<raise>/<event>` to return the appropriate output parameters. A compound state, on the other hand, can use either `<raise>`, `<event>`, or the `<donedata>` element of the `<final>` states to perform this function.
 - The included `<state>` must be self-contained: it should not have transitions to states in the including document or outside of itself.
 - The included `<state>` must not use datamodel elements from the included document, unless the `<data>` elements are defined within the `<state>` or one of its children. Using `<data>` elements defined elsewhere in the included document will likely result in an error since they are not defined in the including document.
 - The included `<state>` should not use datamodel elements from the including document. Doing so

makes subroutine information global to the application. It is recommended that data should be passed to a subroutine via an event, or through variables defined via `<script>`.

- The included `<state>` must not rely on events from the including document other than the transition to the included state.
- For the including document:
 - The document must have a `<transition>` for the event generated by the included state or subroutine. This event will contain the results from the subroutine.
 - The document must have a `<transition>` to the included state. The event can contain the input parameters for the subroutine. Alternately, the including state can use a `<script>` element in its `<onentry>` element to define and initialize a set of parameters that are passed to the included `<state>`. The included state can access these parameters through the variable scoping that ECMAScript provides.
 - When using `<xi:include>` elements, the namespaces used in the included document need to be declared in the including document.

The following are the additional Genesys attributes for the `<xi:include>` element as well as existing attribute limitations.

Attribute Details

Name	Required	Type	Default Value	Valid Values	Description
accept	false	string			<i>Not supported</i>
accept-language	false	string			<i>Not supported</i>
encoding	false	string			<i>Not supported</i>
href	true	URL	none		<p>The URI of the resource to include.</p> <p>As of ORS 8.1.300.27, this attribute also supports substitution by session start parameters. These parameters may come from the ApplicationParms section of an Enhanced Routing Script, URL-encoded parameters of a web-started session, or the <code><param></code> elements nested within a <code><session:start></code></p>

Name	Required	Type	Default Value	Valid Values	Description
					<p>For example:</p> <pre><xi:include href="http://appsrv:80/ scxml/ subroutine_routing.scxml" /></pre> <p>It is possible to parametrize the URI as follows:</p> <pre><xi:include href="http://appsrv:80/ scxml/\$\$MY_SUBROUTINE\$\$" /></pre> <p>When a special token of the form <code>\$\$parameter_name\$\$</code> is provided, it will be automatically substituted with the value of the matching session start parameter (case-sensitive).</p> <p>If the session start parameters are as follows:</p> <pre>MY_SUBROUTINE = subroutine_chat.scxml</pre> <p>Resulting URI:</p> <pre><xi:include href="http://appsrv:80/ scxml/ subroutine_chat.scxml" /></pre>
parse	false	string	"xml"	"xml", "text"	<p>See the following for details: http://www.w3.org/TR/xinclude/#include_element</p>
resolveid	false	string	none	Any value string	<p>In order to support subroutines and avoid issues with duplicate SCXML element IDs (for example, <code><state id=x></code>), this Genesys extension</p>

Name	Required	Type	Default Value	Valid Values	Description
					<p>attribute must be used. If this attribute is specified, then ID modifications will occur. All the SCXML elements with an ID attribute (<state>, <parallel>, <final>, <history>, <send>, <invoke>, <cancel>, and <data>) in the included document are prefixed by the value of this attribute, and a separating dot. In addition, the IDREF attributes in the included document (the initial attribute in the <state> element and the target and event attributes in the <transition> element) can also be modified as long as the following wildcard substitution key is specified in the value. Otherwise they will not be changed when included. The substitution key is the string "\$\$_MY_PREFIX_\$\$_". If specified, it is replaced by the value of this attribute for the included document.</p> <p>IMPORTANT NOTE: Developers must ensure that each use of the 'resolveid' attribute value is unique across the chain of included documents.</p>
xmlns	false	string	none	Any value string	Used to provide namespaces for the included document. This is necessary for

Name	Required	Type	Default Value	Valid Values	Description
					fragments. If subroutines include subroutines, this attribute must be set to the appropriate namespace for the including element. For example, <code>xmlns:xi="http://www.w3.org/2001/XInclude"</code>
xpointer	false	string	none		When <code>parse="xml"</code> , xpointer may be used to specify a particular element and its children to include. The value of xpointer must be a literal ID. The first node in the included document that matches that ID is included. When xpointer is omitted, the entire resource is included. Note: XPath is not supported.

When `resolveid` is used, two additional items can be used to handle the prefix provided by this attribute:

Name	Valid locations	Description
<code>\$_MY_PREFIX_</code>	<ul style="list-style-type: none"> initial attribute of <code><state></code> event and target attribute of <code><transition></code> 	During document assembly, <code>\$_MY_PREFIX_</code> is replaced with the value of <code>resolveid</code> only in the defined locations. The engine does not perform global search and replace with this token.
<code>_my_prefix</code>	Any ECMAScript expression	During document assembly, <code><state></code> , <code><parallel></code> , and <code><final></code> is given a <code>_my_prefix</code> attribute extension that contains the value of <code>resolveid</code> . This allows the prefix value to be used in ECMAScript expressions within these states.

Children

The child element `<fallback>` is *not supported*.