GENESYS™

Orchestration Server Developer's Guide

Interaction Interface Action Elements

5/2/2025

Contents

Interaction Interface Action Elements

Common Actions

The following are the common actions across interactions:

<terminate>

This is the action to terminate the interaction. It is equivalent to the `TClearCall` request in T-Server. For non-voice interactions, this action can only be done when the interaction is "presented" with *RequestStopProcessing*.

Attribute Details

Name	Required	Type	Default Value	Valid Values	Description
requestid	false	location expression	none	Any valid location expression	This is the location for the request ID that is returned as part of this request. Any data model expression evaluating to a data model location. See SCXML Location Expressions for details. The location's value will be set to an internally generated unique string identifier to be associated with the action being sent. If this attribute is not specified, the event identifier is dropped. This identifier can be tested by the completion event handler to distinguish among several outstanding requests. If this attribute is not specified, the identifier can be acquired from the completion event. Every request must receive a unique

Name	Required	Type	Default Value	Valid Values	Description
					identifier.
interactionid	true	value expression		Any value expression that returns a valid string	A value expression which returns the _genesys.ixn.interactions[x].g_u associated with this request. See SCXML Legal Data Values and Value Expressions for details.
reason	false	value expression	none	Any value expression that returns a valid string	A value expression which returns a character string which identifies the reason why the interaction is being terminated. See SCXML Legal Data Values and Value Expressions for details.
resource	false	value expression	none	Any value expression that returns a valid string	A value expression which returns the resource address for the call to be terminated. Has no meaning when used with a multimedia interaction. See SCXML Legal Data Values and Value Expressions **Note**: Since 8.1.100.19
hints	false	value expression	none	Any valid ECMAScript object	A value expression which returns the ECMAScript object containing information which may be used by the implementing functional module when redirecting this interaction. This information may consist of protocol-specific parameters, protocol selection guidelines, and so on. **Note:** The meaning of these hints is specific to the implementing functional module. See SCXML Legal

Name	Required	Type	Default Value	Valid Values	Description
					Data Values and Value Expressions for details.

hints Attribute Considerations

- When the interaction is non-voice, properties "ucs" and "delete" could be defined in "hints" object.

  - "ucs" - if true, OCS will also send ESP request with Method=StopProcessing via Ixn Server to UCS.

  - "delete" - value of this property will be attached as "Delete" parameter to ESP request.

The following is an example:

```
<state id="do_terminate">
<datamodel>
        <data id="reqid"/>
</datamodel>
<onentry>
        <ixn:terminate requestid="_data.reqid" interactionid="_data.ixnid"
              reason="'finished service X'"
resource="_genesys.ixn.interactions[_data.ixnid].voice.dnis"/>
</onentry>
<transition event="interaction.terminate.done" target="statex"/>
<transition event="error.interaction.terminate" target="statey"/>
</state>
```

Children

None

Events

The following events can be generated as part of this action:

- `interaction.terminate.done` - This event is sent when the request has been accepted by the system and the interaction has started the termination process.

- `error.interaction.terminate` - This event is sent when the request itself has failed for some reason.

- `interaction.deleted` - This event is sent when the interaction is actually terminated.

<clear>

This is the action to clear a given party from the interaction. Note that in cases of conference interactions, the interaction will remain alive until there is no more than one party in the interaction. This is equivalent to the `TReleaseCall` request in T-Server. **Note:** This action is only for voice interactions at this time, but will apply to other media in the future.

Attribute Details

Name	Required	Type	Default Value	Valid Values	Description
requestid	false	location expression	none	Any valid location expression	This is the location for the request ID that is returned as part of this request. Any data model expression evaluating to a data model location. See SCXML Location Expressions for details. The location's value will be set to an internally generated unique string identifier to be associated with the action being sent. If this attribute is not specified, the event identifier is dropped. This identifier can be tested by the completion event handler to distinguish among several outstanding requests. If this attribute is not specified, the identifier can be acquired from the completion event. Every request must receive a unique identifier.
interactionid	true	value expression	none	Any value expression that returns a valid string	A value expression which returns the _genesys.ixn.interactions[x].g_u associated with this request. See SCXML Legal Data Values and Value Expressions for details.
resource	true	value expression	none	Any valid string or Resource Object	A value expression which returns the resource address currently involved in the interaction for which the connection will be cleared. See SCXML Legal Data

Name	Required	Type	Default Value	Valid Values	Description
					Values and Value Expressions for details.
reason	false	value expression	none	Any value expression that returns a valid string	A value expression which returns a character string which identifies the reason why the resource connection with the interaction is being cleared. See SCXML Legal Data Values and Value Expressions for details.

The following is an example:

```
<state id="do_clear">
<datamodel>
        <data id="reqid"/>
        <data id="ixnid"/>
        <data id="rscid"/>
</datamodel>
<onentry>
        <ixn:clear requestid="_data.reqid" interactionid="_data.ixnid" resource="_data.rscid"
reason="'finished service X'"/>
</onentry>
<transition event="interaction.clear.done" target="statex"/>
<transition event="error.interaction.clear" target="statey"/>
</state>
```

Children

None

Events

The following events can be generated as part of this action:

- `interaction.clear.done` - This event is sent when the request has been accepted by the system and the resource's connection to the interaction has started the clearing process.
- `error.interaction.clear` - This event is sent when the request itself has failed for some reason.
- `interaction.partydeleted` - This event is sent when the resources's connection to the interaction is actually cleared.

<redirect>

This is the action to redirect the interaction to another resource. This action is used for the following media server actions:

- T-Server - `TRouteCall`, (IRD Functions - DeliverCall, DeliverToIVR, RouteCall, TRoute)

- Interaction Server - `RequestPlaceInQueue, RequestPlaceInWorkbin, RequestDeliver` (IRD Function Blocks - Queue Interaction, IRD Functions - RouteCall, TRoute)

When using the `<redirect>` call, the redirected interaction may or may not start a new session depending on whether or not the redirecting session is already terminated. In the event that the interaction arrives on the target resource after the redirecting session has terminated, a new session will be started. Otherwise, if the interaction arrives on the target resource prior to the termination of the redirecting session, no new session will be created. To guarantee the creation of a new session for the redirected interaction, it is suggested to use `<detach>` to dissolve the interaction/session association.

Attribute Details

Name	Required	Type	Default Value	Valid Values	Description
detach	false	Boolean expression	false	true, false	ORS Release 8.1.400.21 introduced a detach attribute for the <ixn:redirect> action. The default value is false to comply with existing/previous ORS behavior. The attribute controls whether ORS should detach an interaction from the current session before routing to the specified target, which can allow ORS to start processing the next session. As further detailed below, if the value is true, then the interaction is detached from the session before routing to the specified target. If the value is false, then no detach occurs before routing. **Succesful Redirect/ detach=true** In the case of a successful execution of a redirect action with detach=true, a routed interaction (previously belonging to the

Name	Required	Type	Default Value	Valid Values	Description
					session that invoked this action) will be automatically detached. The SCXML session invoking this action receives interaction.notcontrolled and interaction.deleted events with resultof=detaching, and then interaction.redirect.done event. **Unsuccesesful Redirect/ detach=true** In the case of a failed <ixn:redirect> action (with detach=true), the session invoking this action receives only the error.interaction.redirect event. The interaction remains in the same attached/detached state as it was before invoking <ixn:redirect> action with detach=true. **Note:** If the <ixn:redirect> action is provided with the ID of an interaction that is already detached, then the detach attribute has no effect. In this case, ORS produces warning IxnVMMThreadData::CommonPartHan WARNING - ixn action attribute 'detach' has no effect, interaction 'ZZZ' is already detached and operates as if detach=false. If a redirect error happens, the interaction remains in the same state

Name	Required	Type	Default Value	Valid Values	Description
					(is still detached) and the application developer must decide the next step for the interaction.
requestid	false	location expression	none	Any valid location expression	This is the location for the request ID that is returned as part of this request. Any data model expression evaluating to a data model location. See SCXML Location Expressions for details. The location's value will be set to an internally generated unique string identifier to be associated with the action being sent. If this attribute is not specified, the event identifier is dropped. This identifier can be tested by the completion event handler to distinguish among several outstanding requests. If this attribute is not specified, the identifier can be acquired from the completion event. Every request must receive a unique identifier.
interactionid	true	value expression	none	Any value expression that returns a valid string	A value expression which returns the _genesys.ixn.interactions[x].g_u associated with this request. See SCXML Legal Data Values and Value Expressions for details.
from	true	value expression	none	Any valid string or Resource Object	A value expression which returns the address that this interaction is to be

Name	Required	Type	Default Value	Valid Values	Description
					redirected from. For details, on the format, see Addressing Resources. See SCXML Legal Data Values and Value Expressions
to	true	value expression	none	Any valid string or Resource Object	A value expression which returns the destination address for this interaction. For details, on the format, see Addressing Resources. See SCXML Legal Data Values and Value Expressions
type	false	NMTOKEN	_genesys.queue.rType.RouteTypeDefaultType	Values from the RouteTypeDefaultType enumeration	This defines the type of redirection processing that is to be done.
hints	false	value expression	none	Any valid ECMAScript object	A value expression which returns the ECMAScript object containing information which may be used by the implementing functional module when redirecting this interaction. This information may consist of protocol-specific parameters, protocol selection guidelines, and so on. **Note:** The meaning of these hints is specific to the implementing functional module. See SCXML Legal Data Values and Value Expressions for details.

hints Attribute Considerations Note: you can use a special interaction queue name (__STOP__) which will do the processing required to stop the interaction.

- When the interaction is non-voice, inPersistentQueues and OutPersistentQueues properties of the Interaction Server interaction will be set automatically to the appropriate queues defined for the session (this information comes from the trigger information). So these properties do not need to be

exposed. BUT if the developer wants to explicitly specify these properties, they can supply them through the hints attribute object. The object should have the following properties

- "inqueues" - This is an ECMAScript object with a collection of string properties. The name of the property is an interaction queue name and the value is a description of that queue.

- "outqueues" - This is an ECMAScript object with a collection of string properties. The name of the property is an interaction queue name and the value is a description of that queue.

- Property **extensions** of **hints** object allows to specify the content of `AttributeExtension` for the corresponding `TRequestRouteCall` request. A valid value for property **extensions** is an ECMAScript object. (Since 8.1.2)

- Property **reasons** of **hints** object allows to specify the content of `AttributeReason` for the corresponding `TRequestRouteCall` request. A valid value for property **reasons** is an ECMAScript object. (Since 8.1.4)

The following are some examples:

```
<state id="do_redirect_voice_interaction_to_an_agent_dn">
    <onentry>
        <ixn:redirect interactionid="_data.ixnid" from="'1010'" to="'1111'"/>
    </onentry>
    <transition event="interaction.redirect.done" target="statex"/>
    <transition event="error.interaction.redirect" target="statey"/>
</state>
<state id="do_redirect_mm_interaction_to_an_agent">
    <onentry>
        <ixn:redirect interactionid="_data.ixnid" from="'1010'" to="'1111'"/>
    </onentry>
    <transition event="interaction.redirect.done" target="statex"/>
    <transition event="error.interaction.redirect" target="statey"/>
</state>
<state id="do_redirect_mm_interaction_to_an_interaction_queue">
    <onentry>
        <ixn:redirect requestid="_data.reqid" to="'queueF'"/>
    </onentry>
    <transition event="interaction.redirect.done" target="statex"/>
    <transition event="error.interaction.redirect" target="statey"/>
</state>
<state id="do_redirect_multi-resource_interaction_to_another_agent">
    <onentry>
        <ixn:redirect requestid="_data.reqid" interactionid="_data.ixnid"
                      from="'3444'" to="'6555'"/>
    </onentry>
    <transition event="interaction.redirect.done" target="statex"/>
    <transition event="error.interaction.redirect" target="statey"/>
</state>
<state id="do_redirect_voice_interaction_to_another_queue-route_point">
    <onentry>
        <ixn:redirect requestid="_data.reqid" to="'6666'"/>
    </onentry>
    <transition event="interaction.redirect.done" target="statex"/>
    <transition event="error.interaction.redirect" target="statey"/>
</state>
<state id="do_redirect_with_resource_object">
    <onentry>
        <script>
            var dest = {type:"A",
                        agent:"702_sip",
                        dn:"702",
                        id:"702_sip",
```

```
                        place:"702",
                        'switch':"SipSwitch"};
              var myhints = { extensions : { ext1:"extval1" }, reasons : {reason1:2} };
      </script>
      <ixn:redirect requestid="_data.reqid" interactionid="_data.ixnid"
                    from="'RP_sip1'" to="dest" type="1" hints="myhints" />
  </onentry>
  <transition event="interaction.redirect.done" target="statex"/>
  <transition event="error.interaction.redirect" target="statey"/>
</state>
```

Children

None

Events

The following events can be generated as part of this action:

- `interaction.redirect.done` - This event is sent when the request has been accepted by the system and the interaction has started the redirection process.

- `error.interaction.redirect` - This event is sent when the request itself has failed for some reason.

- `interaction.ondivert` - This event is sent when the interaction has been moved to the defined destination.

<singlesteptransfer>

This is the action to transfer the interaction to another resource in a single step. This is equivalent to the `TSingleStepTransfer` request in T-Server. **Note:** This action is only for voice interactions at this time

Attribute Details

Name	Required	Type	Default Value	Valid Values	Description
requestid	false	location expression	none	Any valid location expression	This is the location for the request ID that is returned as part of this request. Any data model expression evaluating to a data model location. See SCXML Location Expressions for details. The location's value will be set to an internally generated unique string identifier to be associated with the action being sent. If this attribute is not

Name	Required	Type	Default Value	Valid Values	Description
					specified, the event identifier is dropped. This identifier can be tested by the completion event handler to distinguish among several outstanding requests. If this attribute is not specified, the identifier can be acquired from the completion event. Every request must receive a unique identifier.
interactionid	true	value expression	none	Any value expression that returns a valid string	A value expression which returns the _genesys.ixn.interactions[x].g_u associated with this request. See SCXML Legal Data Values and Value Expressions for details.
from	true	value expression	none	Any valid string or Resource Object.	A value expression which returns the address that this interaction is to be transferred from. For details, on the format, see Addressing Resources. See SCXML Legal Data Values and Value Expressions
to	true	value expression	none	Any valid string or Resource Object.	A value expression which returns the destination address for this interaction. For details, on the format, see Addressing Resources. See SCXML Legal Data Values and Value Expressions
hints	false	value expression	none	Any valid ECMAScript object	A value expression which returns the ECMAScript object

Name	Required	Type	Default Value	Valid Values	Description
					containing information which may be used by the implementing functional module when transferring this interaction. This information may consist of protocol-specific parameters, protocol selection guidelines, and so on. **Note:** The meaning of these hints is specific to the implementing functional module. See SCXML Legal Data Values and Value Expressions for details.

The following are some examples:

```
<state id="dosingle_step_transfer_voice_interaction_to_an_agent_dn">
<datamodel>
        <data id="reqid"/>
        <data id="ixnid"/>
        <data id="rscid"/>
</datamodel>
<onentry>
        <script>
                var myhints = { extensions:{ key_1:'Value1', key_2:200*200, key_3:300*300 } };
        </script>
        <ixn:singlesteptransfer requestid="_data.reqid" interactionid="_data.ixnid"
from="_data.rscid" to="'1111'" hints="myhints" />
</onentry>
<transition event="interaction.singlesteptransfer.done" target="statex"/>
<transition event="error.interaction.singlesteptransfer" target="statey"/>
</state>
```

Children

None

Events

The following events can be generated as part of this action:

- `interaction.singlesteptransfer.done` - This event is sent when the request has been accepted by the system and the interaction has started the single step transfer process.
- `error.interaction.singlesteptransfer` - This event is sent when the request itself has failed for some reason.
- `interaction.onsinglesteptransfer` - This event is sent when the interaction has been transferred to the defined destination.

\<singlestepconference\>

This is the action to conference another resource into the interaction in a single step. It is equivalent to the TSingleStepConference request in T-Server. **Note:** This action is only for voice interactions at this time

Attribute Details

Name	Required	Type	Default Value	Valid Values	Description
requestid	false	location expression	none	Any valid location expression	This is the location for the request ID that is returned as part of this request. Any data model expression evaluating to a data model location. See SCXML Location Expressions for details. The location's value will be set to an internally generated unique string identifier to be associated with the action being sent. If this attribute is not specified, the event identifier is dropped. This identifier can be tested by the completion event handler to distinguish among several outstanding requests. If this attribute is not specified, the identifier can be acquired from the completion event. Every request must

Name	Required	Type	Default Value	Valid Values	Description
					receive a unique identifier.
interactionid	true	value expression	none	Any value expression that returns a valid string	A value expression which returns the `_genesys.ixn.interactions[x].g_u` associated with this request. See SCXML Legal Data Values and Value Expressions for details.
from	true	value expression	none	Any valid string or Resource Object	A value expression which returns the address that this interaction is to be conferenced from. For details, on the format, see Addressing Resources. See SCXML Legal Data Values and Value Expressions
to	true	value expression	none	Any valid string or Resource Object.	A value expression which returns the destination address for the resource that is to be conferenced into the interaction. For details, on the format, see Addressing Resources. See SCXML Legal Data Values and Value Expressions

The following is an example:

```
<state id="dosingle_step_conference_voice_interaction_to_an_agent_dn">
<datamodel>
        <data id="reqid"/>
        <data id="ixnid"/>
        <data id="rscid"/>
</datamodel>
<onentry>
        <ixn:singlestepconference requestid="_data.reqid" interactionid="_data.ixnid"
from="_data.rscid" to="'1111'"/>
</onentry>
<transition event="interaction.singlestepconference.done" target="statex"/>
<transition event="error.interaction.singlestepconference" target="statey"/>
</state>
```

Children

None

Events

The following events can be generated as part of this action:

- `interaction.singlestepconference.done` - This event is sent when the request has been accepted by the system and the interaction has started the single step conference process.

- `error.interaction.singlestepconference` - This event is sent when the request itself has failed for some reason.

- `interaction.onsinglestepconference` - This event is sent when the specified resource has been conferenced into the interaction.

<associate>

This is the action to associate an interaction with the target SCXML session and un-associate it with the current session. This is implicitly done when a session is triggered by an interaction event, but there are cases where an interaction that is currently not associated with another SCXML session needs to be associated with it by the session that currently owns the interaction. The association process will add the interaction to the target session's `_genesys.ixn.interactions[]` array. In addition, the functional module will do the necessary media-specific processing to ensure that this session will be able to receive events associated with this interaction as well as be able to take action against it. This may also include the taking of ownership of this interaction from another SCXML session or 3rd party application. This level of processing will depend on the capabilities of the underlying functional module and its associated system.

Attribute Details

Name	Required	Type	Default Value	Valid Values	Description
requestid	false	location expression	none	Any valid location expression	This is the location for the request ID that is returned as part of this request. Any data model expression evaluating to a data model location. See SCXML Location Expressions for details. The location's value will be set to an internally generated unique string identifier to be associated with the action being sent. If this attribute is not specified, the event identifier is

Name	Required	Type	Default Value	Valid Values	Description
					dropped. This identifier can be tested by the completion event handler to distinguish among several outstanding requests. If this attribute is not specified, the identifier can be acquired from the completion event. Every request must receive a unique identifier.
category	true	value expression	none	Any value expression that returns one of the following values: voice, msgbased, chat, web	A value expression which returns the category for this interaction. See SCXML Legal Data Values and Value Expressions for details.
Interactionid	true	value expression	none	Any value expression that returns a valid string	A value expression which returns the interaction id that is to be associated with this session. See SCXML Legal Data Values and Value Expressions for details.
sessionid	true	value expression	none	Any value expression that returns a valid string	A value expression which returns the session id to associate interaction with. See SCXML Legal Data Values and Value Expressions for details.

The following is an example:

```
<state id="do_associate">
<datamodel>
        <data id="reqid"/>
</datamodel>
<onentry>
        <ixn:associate requestid="_data.reqid" category="'voice'"
interactionid="_event.data.id" sessionid="'12345678'"/>
</onentry>
<transition event="interaction.associate.done" target="statex"/>
<transition event="error.interaction.associate" target="statey"/>
```

```
</state>
```

Children

None

Events

The following events can be generated as part of this action:

- `interaction.associate.done` - This event is sent when the request has been initiated to associate the interaction with the SCXML session.
- `error.interaction.associate` - This event is sent when the request itself has failed for some reason.
- `interaction.deleted` and `interaction.notcontrolled` - These events indicate that the interaction has been associated with the originating session.
- `interaction.added` and `interaction.present` - These events indicate that the interaction is now associated with the new session.

Note: A current limitation is that using the association function with an incorrect 'sessionid' results in the event 'interaction.associate.done' being generated, but the interaction remains with the current session.

<accept>

This is the action that accepts and answers an interaction at the specific resource. This is used to answer a voice interaction or accept a multi-media-based interaction. It is equivalent to the `TAnswerCall` request in T-Server. **Note:** This action is only for voice interactions at this time.

Attribute Details

Name	Required	Type	Default Value	Valid Values	Description
requestid	false	location expression	none	Any valid location expression	This is the location for the request ID that is returned as part of this request. Any data model expression evaluating to a data model location. See SCXML Location Expressions for details. The location's value will be set to an internally generated unique string identifier to be associated with the action being sent. If this attribute is not specified, the

Name	Required	Type	Default Value	Valid Values	Description
					event identifier is dropped. This identifier can be tested by the completion event handler to distinguish among several outstanding requests. If this attribute is not specified, the identifier can be acquired from the completion event. Every request must receive a unique identifier.
interactionid	true	value expression		Any value expression that returns a valid string	A value expression which returns the _genesys.ixn.interactions[x].g_u associated with this request. See SCXML Legal Data Values and Value Expressions for details.
resource	true	value expression	none	Any valid string or Resource Object.	A value expression which returns the address that this interaction is to be accepted for. For details, on the format, see Addressing Resources. See SCXML Legal Data Values and Value Expressions

The following is an example:

```
<state id="do_accept">
<datamodel>
        <data id="reqid"/>
</datamodel>
<onentry>
        <ixn:accept requestid="_data.reqid"/>
</onentry>
<transition event="interaction.accept.done" target="statex"/>
<transition event="error.interaction.accept" target="statey"/>
</state>
```

Children

None

Events

The following events can be generated as part of this action:

- `interaction.accept.done` - This event is sent when the request has been initiated to accept theinteraction by the resource.
- `error.interaction.accept` - This event is sent when the request itself has failed for some reason.
- `Interaction.partystatechanged` - This event is sent when the resource accepts or answers the interaction for processing.

<attach>

This action (as well as the 2 other related actions <detach> and <associate>) change the ownership relation between sessions and interactions. It is used to associate an **ownerless** interaction with current session. Attempting to apply this action to an interaction owned by another session will fail - taking the interaction from another session is supposed to be done by agreement with this session the (see <associate> action). For multimedia interactions <attach> will not pull an interaction - it must be already pulled to work. Typically, when using this action for multimedia, the <detach> would be used prior to <attach>.

Attribute Details

Name	Required	Type	Default Value	Valid Values	Description
requestid	false	location expression	none	Any valid location expression	This is the location for the request ID that is returned as part of this request. Any data model expression evaluating to a data model location. See SCXML Location Expressions for details. The location's value will be set to an internally generated unique string identifier to be associated

Name	Required	Type	Default Value	Valid Values	Description
					with the action being sent. If this attribute is not specified, the event identifier is dropped. This identifier can be tested by the completion event handler to distinguish among several outstanding requests. If this attribute is not specified, the identifier can be acquired from the completion event. Every request must receive a unique identifier.
Interactionid	true	value expression	none	Any value expression that returns a valid string	A value expression which returns the interaction id that is to be attached to this session. See SCXML Legal Data Values and Value Expressions for details.

Example

```
<state id="do_attach">
```

```
    <datamodel>
        <data id="reqid"/>
    </datamodel>
    <onentry>
        <ixn:attach requestid="_data.reqid" interactionid="_event.data.id"/>
    </onentry>
    <transition event="interaction.attach.done" target="statex"/>
    <transition event="error.interaction.attach" target="statey"/>
</state>
```

Children

None

Events

The following events can be generated as part of this action:

- `interaction.attach.done` - Success; will be accompanied with asynchronous events interaction.added and interaction.present.

- `error.interaction.attach` - This event is sent when the request has failed for some reason (for example - interaction does not exist or interaction is owned by another session).

<detach>

This action (as well as the 2 other related actions <attach> and <associate>) change the ownership relation between sessions and interactions. It is the opposite of <attach> and is used to dis-associate an interaction from the current session. A session that <detach>'s for some interaction is still responsible for this interaction, or ensuring that the interaction is handled by some other session. This applies to both voice and multimedia interactions (in the case of multimedia, a detached interaction will not be returned back into queue). Normally after <detach> the session should either move the interaction to another session via <associate> or <attach> back to the interaction. Note. As <detach> action relies on interaction attached data it cannot be done immediately after assigning the interaction to the session. If invoked before interaction data properly updated the action will result error.interaction.detach with error invalidstate. In such cases application should repeat the action after short delay

Attribute Details

Name	Required	Type	Default Value	Valid Values	Description
requestid	false	location expression	none	Any valid location expression	This is the location for the request ID that is returned as part of this request. Any data model expression

Name	Required	Type	Default Value	Valid Values	Description
					evaluating to a data model location. See SCXML Location Expressions for details. The location's value will be set to an internally generated unique string identifier to be associated with the action being sent. If this attribute is not specified, the event identifier is dropped. This identifier can be tested by the completion event handler to distinguish among several outstanding requests. If this attribute is not specified, the identifier can be acquired from the completion event. Every request must receive a unique identifier.
Interactionid	true	value expression	none	Any value expression that	A value expression which

Name	Required	Type	Default Value	Valid Values	Description
				returns a valid string	returns the interaction id that is to be attached to this session. See SCXML Legal Data Values and Value Expressions for details.

Children

None

Events

The following events can be generated as part of this action:

- `interaction.detach.done` - Success; will be accompanied with asynchronous events interaction.notcontrolled and interaction.deleted.

- `error.interaction.detach` - This event is sent when the request has failed for some reason (for example - interaction does not exist or interaction is owned by another session).

- `interaction.notcontrolled` - This event is sent when the interaction is no longer associated with a session.

- `interaction.deleted` - This event is sent when the interaction is deleted from the current session.

Note: SCXML application may have transition for events interaction.notcontrolled and interaction.deleted in some of outer states. In this case, if those events, raised after ixn:detach action, will not be handled in a state where ixn:detach has been executed, they may cause undesired transitions. To avoid that, add target-less transitions for those events into state where ixn:detach is executed. In Composer, it may be done by adding exceptions for those events in "properties" of corresponding block (for example "Detach" or "ForceRoute" with detach=true), with "Target" checkbox un-checked.

Voice Interaction Actions

<createcall>

This is the initiation or creation of a voice interaction between a customer or resource and a resource. This is equivalent to the T-Server `TMakeCall` or `TMakePredictiveCall` functions.

Attribute Details

Name	Required	Type	Default Value	Valid Values	Description
requestid	false	location expression	none	Any valid location expression	This is the location for the request ID that is returned as part of this request. Any data model expression evaluating to a data model location. See SCXML Location Expressions for details. The location's value will be set to an internally generated unique string identifier to be associated with the action being sent. If this attribute is not specified, the event identifier is dropped. This identifier can be tested by the completion event handler to distinguish among several outstanding requests. If this attribute is not specified, the identifier can be acquired from the completion event. Every request must receive a unique identifier.
from	true	value expression	none	Any valid string or Resource Object	A value expression which returns the address that this call is to be made from. See SCXML Legal Data Values and Value Expressions
to	true	value expression	none	Any valid string or Resource Object	A value expression which returns the destination address for this call. See SCXML Legal Data Values and Value Expressions

Name	Required	Type	Default Value	Valid Values	Description
type	false	NMTOKEN	regular	regular, agent, supervisor, priority, agentpriority, predictive	This defines the type of call that is to be created. • Regular - is a call to a customer. • Agent - is a call to another agent. • Supervisor - is a call to a supervisor. • Priority - is a priority call to a customer. • Agentpriority - is a priority call to another agent. • Predictive - is a predictive call to a customer.
reason	false	value expression	none	Any valid string	A value expression which returns the reason for making this call. See SCXML Legal Data Values and Value Expressions for details.
udata	false	value expression	none	Any valid ECMAScript object	A value expression which returns a valid ECMAScript object. This object's data will become part of the call's interaction object. See SCXML Legal Data Values and Value Expressions for details.

Name	Required	Type	Default Value	Valid Values	Description
hints	false	value expression	none	Any valid ECMAScript object	A value expression which returns the ECMAScript object containing information which may be used by the implementing functional module when establishing this call. This information may consist of protocol-specific parameters, protocol selection guidelines, and so on. **Note:** The meaning of these hints is specific to the implementing functional module. See SCXML Legal Data Values and Value Expressions for details.
timeout	false	value expression	none	A value expression which returns an integer	A value expression which returns an integer which represents the number of seconds to wait. See SCXML Legal Data Values and Value Expressions for details. The integer returned is interpreted as a time interval. This interval begins when `<createcall>` is executed. The `<createcall>` must fail if not completed by the end of this interval. Completion is defined as the call getting to a CONNECTED state as signaled by an interaction.created event. A failed `<createcall>` must return either error.voice.createcall event or the interaction.partystatechanged event. This attribute is only used when the type attribute is set to "predictive".

The following is an example:

```
<state id="do_createcall">
  <datamodel>
        <data id="reqid"/>
  </datamodel>
  <onentry>
        <ixn:createcall requestid="_data.reqid" from="'1234'" to="'+1919466600'" />
  </onentry>
  <transition event="voice.createcall.done" target="statex"/>
  <transition event="error.voice.createcall" target="statey"/>
</state>
```

hints Attribute Considerations When property type of `<createcall>` action is set to predictive, property **extensions** of **hints** object allows to specify the content of `AttributeExtension` for the corresponding `TMakePredictiveCall` request. This is used to control call progress detection (CPD) processing. A valid value for property **extensions** is an ECMAScript object. Valid extensions to use for CPD purposes are described in the Deployment Guide for a specific T-Server. The following is an example using CPD to detect answering machine:

```
<state id="do_createcall">
  <datamodel>
        <data id="reqid"/>
  </datamodel>
  <onentry>
        <script>
                var cpdExtensions = {"extensions":{
                                        "cpd-record":"off",

"call_answer_type_recognition":"positive_am_detection",
                                        "cpd-on-connect":"off",
                                        "call_timeguard_timeout":"10000"}};
        </script>
        <ixn:createcall requestid="_data.reqid" from="'1234'" to="'+1919466600'"
type="predictive" hints="cpdExtensions" />
  </onentry>
  <transition event="voice.createcall.done" target="statex"/>
  <transition event="error.voice.createcall" target="statey"/>
</state>
```

Children

None

Events

The following events can be generated as part of this action:

- `voice.createcall.done` - This event is sent when the request has been accepted by the system and the call has started the creation process.

- `interaction.added` - This event is sent when either party is connected to the call.

- `interaction.partystatechanged` - This event is sent when the call has been created and either party is connected to the call.

- `interaction.partystatechanged` - This event is sent when the call has failed for some reason. This includes timeouts based on the timeout attribute.

- `error.voice.createcall` - This event is sent when the request itself has failed for some reason.

<hold>

This action puts a voice interaction on hold for a specific resource's device. This is equivalent to the T-Server `THoldCall` function.

Attribute Details

Name	Required	Type	Default Value	Valid Values	Description
requestid	false	location expression	none	Any valid location expression	This is the location for the request ID that is returned as part of this request. Any data model expression evaluating to a data model location. See SCXML Location Expressions for details. The location's value will be set to an internally generated unique string identifier to be associated with the action being sent. If this attribute is not specified, the event identifier is dropped. This identifier can be tested by the completion event handler to distinguish among several outstanding requests. If this attribute is not specified, the identifier can be acquired from the completion event. Every request must receive a unique identifier.
interactionid	true	value expression		Any value expression that returns a valid string	A value expression which returns the `_genesys.ixn.interactions[x].g_u` associated with this request. See SCXML Legal Data Values and Value Expressions for details.
resource	true	value expression	none	Any valid string or Resource Object	A value expression which returns the

Name	Required	Type	Default Value	Valid Values	Description
					holding device address for this call. See SCXML Legal Data Values and Value Expressions

The following is an example:

```
<state id="do_holdcall">
<datamodel>
        <data id="reqid"/>
        <data id="ixnid"/>
</datamodel>
<onentry>
        <ixn:hold requestid="_data.reqid" interactionid="_data.ixnid" resource="'1234'" />
</onentry>
<transition event="voice.hold.done" target="statex"/>
<transition event="error.voice.hold" target="statey"/>
</state>
```

Children

None

Events

The following events can be generated as part of this action:

- `voice.hold.done` - This event is sent when the request has been accepted by the system and the interaction has started the hold process.
- `interaction.partystatuschanged` - This event is sent when the actual resource is on hold for the call.
- `<samp>error.voice.hold</samp>` - This event is sent when the request itself has failed for some reason.

<retrieve>

This action retrieves a voice interaction from hold for a specific resource's device. This is equivalent to the T-Server `TRetrieveCall` function.

Attribute Details

Name	Required	Type	Default Value	Valid Values	Description
requestid	false	location expression	none	Any valid location expression	This is the location for the request ID that is returned as part of this request. Any data

Name	Required	Type	Default Value	Valid Values	Description
					model expression evaluating to a data model location. See SCXML Location Expressions for details. The location's value will be set to an internally generated unique string identifier to be associated with the action being sent. If this attribute is not specified, the event identifier is dropped. This identifier can be tested by the completion event handler to distinguish among several outstanding requests. If this attribute is not specified, the identifier can be acquired from the completion event. Every request must receive a unique identifier.
interactionid	true	value expression		Any value expression that returns a valid string	A value expression which returns the `_genesys.ixn.interactions[x].g_u` associated with this request. See SCXML Legal Data Values and Value Expressions for details.
resource	true	value expression	none	Any valid string or Resource Object	A value expression which returns the retrieved device address for this call. See SCXML Legal Data Values and Value Expressions

The following is an example:

```
<state id="do_retrievecall">
<datamodel>
        <data id="reqid"/>
        <data id="ixnid"/>
</datamodel>
```

```
<onentry>
        <ixn:retrieve requestid="_data.reqid" interactionid="_data.ixnid" resource="'1234'" />
</onentry>
<transition event="voice.retrieve.done" target="statex"/>
<transition event="error.voice.retrieve" target="statey"/>
</state>
```

Children

None

Events

The following events can be generated as part of this action:

- `voice.retrieve.done` - This event is sent when the request has been accepted by the system and the interaction has started the retrieve process.

- `interaction.partystatuschanged` - This event is sent when the actual resource is retrieved from hold for the call.

- `error.voice.retrieve` - This event is sent when the request itself has failed for some reason.

<consultation>

This action extends an existing voice interaction to consult with a new resource's device. This is equivalent to the T-Server `TInitiateTransfer` and `TInitiateConference` functions.

Attribute Details

Name	Required	Type	Default Value	Valid Values	Description
requestid	false	location expression	none	Any valid location expression	This is the location for the request ID that is returned as part of this request. Any data model expression evaluating to a data model location. See SCXML Location Expressions for details. The location's value will be set to an internally generated unique string identifier to be associated with the action being sent. If this attribute is not specified, the event identifier is dropped. This identifier can be tested by the completion event

Name	Required	Type	Default Value	Valid Values	Description
					handler to distinguish among several outstanding requests. If this attribute is not specified, the identifier can be acquired from the completion event. Every request must receive a unique identifier.
interactionid	true	value expression		Any value expression that returns a valid string	A value expression which returns the _genesys.ixn.interactions[x].g_u associated with this request. See SCXML Legal Data Values and Value Expressions for details.
from	true	value expression	none	Any valid string or Resource Object	A value expression which returns the address that this consultation is to be made from. See SCXML Legal Data Values and Value Expressions
to	true	value expression	none	Any valid string or Resource Object	A value expression which returns the destination address for this consultation See SCXML Legal Data Values and Value Expressions
udata	false	value expression	none	Any valid ECMAScript object	A value expression which returns a valid ECMAScript object. This object's data will be come part of the consultation call's interaction object. See SCXML Legal Data Values and Value Expressions for details.

The following is an example:

```
<state id="do_consultationcall">
```

```
<datamodel>
        </datamodel>
<onentry>
        <ixn:consultation requestid="_data.reqid" interactionid="_data.ixnid" from="'1234'"
to="'5678'" />
</onentry>
<transition event="voice.consultation.done" target="statex"/>
<transition event="error.voice.consultation" target="statey"/>
</state>
```

Children

None

Events

The following events can be generated as part of this action:

- `voice.consultation.done` - This event is sent when the request has been accepted by the system and it has started the consultation process.

- `error.voice.consultation` - This event is sent when the request itself has failed for some reason.

- `interaction.partyadded` - This event is sent when the consulting or consulted party has been added to the consultation interaction.

- `interaction.partystatechanged` - This event is sent when the consulting or consulted party has changed states.

<alternate>

This action alternates a resource from a held call and an active call. This is equivalent to the T-Server `TAlternateCall` function.

Attribute Details

Name	Required	Type	Default Value	Valid Values	Description
requestid	false	location expression	none	Any valid location expression	This is the location for the request ID that is returned as part of this request. Any data model expression evaluating to a data model location. See SCXML Location Expressions for details. The location's value will be set to an internally generated unique string identifier to be associated with the action being sent. If this

Name	Required	Type	Default Value	Valid Values	Description
					attribute is not specified, the event identifier is dropped. This identifier can be tested by the completion event handler to distinguish among several outstanding requests. If this attribute is not specified, the identifier can be acquired from the completion event. Every request must receive a unique identifier.
heldinteractionid	true	value expression		Any value expression that returns a valid string	A value expression which returns the `_genesys.ixn.interactions[x].g_u` associated with this request. See SCXML Legal Data Values and Value Expressions for details.
heldresource	true	value expression	none	Any valid string or Resource Object	A value expression which returns the address that is the held call. For details, see Addressing Resources. See SCXML Legal Data Values and Value Expressions
activeinteractionid	true	value expression		Any value expression that returns a valid string	A value expression which returns the `_genesys.ixn.interactions[x].g_u` associated with this request. See SCXML Legal Data Values and Value Expressions for details.
activeresource	true	value expression	none	Any valid string or Resource Object	A value expression which returns the address that is in the active call. For details, see Addressing Resources. See SCXML Legal Data

Name	Required	Type	Default Value	Valid Values	Description
					Values and Value Expressions

The following is an example:

```
<state id="do_alternatecall">
<datamodel>
        <data id="reqid"/>
        <data id="Aixnid"/>
        <data id="Hixnid"/>
</datamodel>
<onentry>
        <ixn:alternate requestid="_data.reqid" heldinteractionid="_data.Hixnid"
                heldresource="'1234'" activeinteractionid="_data.Aixnid"
activeresource="'5678'" />
</onentry>
<transition event="voice.alternate.done" target="statex"/>
<transition event="error.voice.alternate" target="statey"/>
</state>
```

Children

None

Events

The following events can be generated as part of this action:

- `voice.alternate.done` - This event is sent when the request has been accepted and the system has started the alternate process.

- `error.voice.alternate` - This event is sent when the request itself has failed for some reason.

- `interaction.partystatechanged` - This event is sent when the held and active resource in the calls changes state (held to active and active to held).

<reconnect>

This action drops the active call in the consultation and retrieves the held call for a specific resource. This is equivalent to the T-Server `TReconnectCall` function.

Attribute Details

Name	Required	Type	Default Value	Valid Values	Description
requestid	false	location expression	none	Any valid location expression	This is the location for the request ID that is returned as

Name	Required	Type	Default Value	Valid Values	Description
					part of this request. Any data model expression evaluating to a data model location. See SCXML Location Expressions for details. The location's value will be set to an internally generated unique string identifier to be associated with the action being sent. If this attribute is not specified, the event identifier is dropped. This identifier can be tested by the completion event handler to distinguish among several outstanding requests. If this attribute is not specified, the identifier can be acquired from the completion event. Every request must receive a unique identifier.
heldinteractionid	true	value expression		Any value expression that returns a valid string	A value expression which returns the _genesys.ixn.interactions[x].g_u associated with this request. See SCXML Legal Data Values and Value Expressions for details.
heldresource	true	value expression	none	Any valid string orResource Object	A value expression which returns the address that is the held call. For details, see Addressing Resources. See SCXML Legal Data Values and Value Expressions
activeinteractionid	true	value expression		Any value expression that returns a valid	A value expression which returns the _genesys.ixn.interactions[x].g_u

Name	Required	Type	Default Value	Valid Values	Description
				string	associated with this request. See SCXML Legal Data Values and Value Expressions for details.
activeresource	true	value expression	none	Any valid string or Resource Object	A value expression which returns the address that is in the active call. For details, see Addressing Resources. See SCXML Legal Data Values and Value Expressions

The following is an example:

```
<state id="do_reconnectcall">
<datamodel>
        <data id="reqid"/>
        <data id="Aixnid"/>
        <data id="Hixnid"/>
</datamodel>
<onentry>
        <ixn:reconnect requestid="_data.reqid" heldinteractionid="_data.Hixnid"
heldresource="'1234'"
                activeinteractionid="_data.Aixnid" activeresource="'5678'" />
</onentry>
<transition event="voice.reconnect.done" target="statex"/>
<transition event="error.voice.reconnect" target="statey"/>
</state>
```

Children

None

Events

The following events can be generated as part of this action:

- `voice.reconnect.done` - This event is sent when the request has been accepted by the system, which has started the reconnect process.

- `interaction.partystatuschanged` - This event is sent when the held resource in the calls changes state.

- `interaction.partydeleted` - This event is sent when the parties of the active call are dropped.

- `interaction.deleted` - This event is sent when the active call is terminated.

- `error.voice.reconnect` - This event is sent when the request itself has failed for some reason.

<conference>

This action conferences a consultation call from a specific resource. This is equivalent to the T-Server `TCompleteConference` function.

Attribute Details

Name	Required	Type	Default Value	Valid Values	Description
requestid	false	location expression	none	Any valid location expression	This is the location for the request ID that is returned as part of this request. Any data model expression evaluating to a data model location. See SCXML Location Expressions for details. The location's value will be set to an internally generated unique string identifier to be associated with the action being sent. If this attribute is not specified, the event identifier is dropped. This identifier can be tested by the completion event handler to distinguish among several outstanding requests. If this attribute is not specified, the identifier can be acquired from the completion event. Every request must receive a unique identifier.
heldinteractionid	true	value expression		Any value expression that returns a valid string	A value expression which returns the _genesys.ixn.interactions[x].g_u associated with this request. See SCXML Legal Data Values and Value Expressions for details.
activeinteractionid	true	value expression		Any value expression that	A value expression which returns the

Name	Required	Type	Default Value	Valid Values	Description
				returns a valid string	_genesys.ixn.interactions[x].g_ associated with this request. See SCXML Legal Data Values and Value Expressions for details.
resource	true	value expression	none	Any valid string or Resource Object	A value expression which returns the address that is conferencing the call. For details, see Addressing Resources. See SCXML Legal Data Values and Value Expressions

The following is an example:

```
<state id="do_conferencecall">
<datamodel>
        <data id="reqid"/>
        <data id="Aixnid"/>
        <data id="Hixnid"/>
</datamodel>
<onentry>
        <ixn:conference requestid="_data.reqid" heldinteractionid="_data.Hixnid"
                activeinteractionid="_data.Aixnid" resource="'1234'"/>
</onentry>
<transition event="voice.conference.done" target="statex"/>
<transition event="error.voice.conference" target="statey"/>
</state>
```

Children

None

Events

The following events can be generated as part of this action:

- `voice.conference.done` - This event is sent when the request has been accepted by the system, which has started the conference process.

- `error.voice.conference` - This event is sent when the request itself has failed for some reason.

- `interaction.onmerge` - This event is sent when the active call is merged with the held call.

- `interaction.partydeleted` - This event is sent when the party on the active call is dropped.

- `interaction.deleted` - This event is sent when the active call is terminated.

<transfer>

This action transfers a consultation call from a specific resource. This is equivalent to the T-Server `TCompleteTransfer` function.

Attribute Details

Name	Required	Type	Default Value	Valid Values	Description
requestid	false	location expression	none	Any valid location expression	This is the location for the request ID that is returned as part of this request. Any data model expression evaluating to a data model location. See SCXML Location Expressions for details. The location's value will be set to an internally generated unique string identifier to be associated with the action being sent. If this attribute is not specified, the event identifier is dropped. This identifier can be tested by the completion event handler to distinguish among several outstanding requests. If this attribute is not specified, the identifier can be acquired from the completion event. Every request must receive a unique identifier.
heldinteractionid	true	value expression		Any value expression that returns a valid string	A value expression which returns the `_genesys.ixn.interactions[x].g_u` associated with this request. See SCXML Legal Data Values and Value Expressions for details.
activeinteractionid	true	value expression		Any value expression that	A value expression which returns the

Name	Required	Type	Default Value	Valid Values	Description
				returns a valid string	_genesys.ixn.interactions[x].g_u associated with this request. See SCXML Legal Data Values and Value Expressions for details.
resource	true	value expression	none	Any valid string or Resource Object	A value expression which returns the address that is transferring the call. For details, see Addressing Resources section. See SCXML Legal Data Values and Value Expressions

The following is an example:

```
<state id="do_transfercall">
<datamodel>
        <data id="reqid"/>
        <data id="Aixnid"/>
        <data id="Hixnid"/>
</datamodel>
<onentry>
        <ixn:transfer requestid="_data.reqid" heldinteractionid="_data.Hixnid"
                activeinteractionid="_data.Aixnid" resource="'1234'"/>
</onentry>
<transition event="voice.transfer.done" target="statex"/>
<transition event="error.voice.transfer" target="statey"/>
</state>
```

Children

None

Events

The following events can be generated as part of this action:

- `voice.transfer.done` - This event is sent when the request has been accepted by the system, which has started the transfer process.

- `error.voice.transfer` - This event is sent when the request itself has failed for some reason.

- `interaction.onmerge` - This event is sent when the held call and consult call on the transferring party are merged.

- `interaction.partydeleted` - This event is sent when the transferring party drops.

- `interaction.deleted` - This event is sent when the consult call is terminated.

\<privateservice\>

This action enables an application to pass data and request services (such as Set Feature, SIP Advice of Charge, change T-Server behavior, and so on) that are supported only by certain T-Servers and which are not covered by general feature requests. This is equivalent to the T-Server `TPrivateService` function and the applicable T-Server documentation should be consulted for the applicability of this request.

Attribute Details

Name	Required	Version	Type	Default Value	Valid Values	Description
requestid	false	8.1.1	location expression	none	Any valid location expression	This is the location for the request ID that is returned as part of this request. Any data model expression evaluating to a data model location. See SCXML Location Expressions for details. The location's value will be set to an internally generated unique string identifier to be associated with the action being sent. If this attribute is not specified, the event identifier is dropped. This identifier can be tested by the completion event handler to distinguish among several outstanding requests. If this attribute is not specified, the identifier can be acquired from the completion event. Every request must receive a unique identifier.
serviceid	true	8.1.1	value expression	none	Any value expression that returns a valid integer	A value expression which returns an integer to indicate the

Name	Required	Version	Type	Default Value	Valid Values	Description
						type of information being passed or the service being requested. This is specific to the T-Server handling the call. Please refer to the T-Server documentation for your switch when setting this value. See SCXML Legal Data Values and Value Expressions for details.
interactionid	true	8.1.1	value expression	none	Any value expression that returns a valid string	A value expression which returns the _genesys.ixn.interactions[x]. associated with this request. Non voice interactions will result in an error.voice.privateservice being generated. See SCXML Legal Data Values and Value Expressions for details.
resource	false	8.1.1	value expression	none	Any valid string or Resource Object	A value expression which returns the DN of the controlling agent or route point on whose behalf the information is provided. For details, see Addressing Resources. See SCXML Legal Data Values and Value Expressions **Note:** This corresponds to the thisDN parameter within the TLib

Name	Required	Version	Type	Default Value	Valid Values	Description
						TPrivateService method. Please refer to the T-Server documentation for your switch when setting this value
udata	false	8.1.1	ECMAScript Object	none	Any valid ECMAScript object	An ECMAScript Object which contains the list of key/value pairs which should be attached to the call in question. See SCXML Legal Data Values and Value Expressions for details.
reasons	false	8.1.1	ECMAScript Object	none	Any valid ECMAScript object	An ECMAScript Object which contains the list of key/value pairs which provide additional information associated with this private service request intended to specify reasons for and results of actions taken by the user. See SCXML Legal Data Values and Value Expressions for details.
extensions	false	8.1.1	ECMAScript Object	none	Any valid ECMAScript object	An ECMAScript Object which contains the list of key/value pairs which provides an additional data structure intended to take account of switch-specific features that cannot be described by other parameters or

Name	Required	Version	Type	Default Value	Valid Values	Description
						in the original structure of user data associated with this private service request. See SCXML Legal Data Values and Value Expressions for details.

Note: When submitting the Private Service request the target T-Server to which this request will be submitted will be determined based upon the following;

- If resource attribute contains both a switch and DN the switch will be used to locate the T-Server to submit the request to.

    - - If resource attribute contains only a switch part, the switch will be used to locate the T-Server to submit the request to and the thisDN value of the underlying TLib TPrivateService will not be populated

- If resource attribute contains only a DN part, then the switch and associated T-Server will be determined from the accommodating interactionid. The switch will be determined based upon the first party that has the DN resource referenced.

- If resource is not provided then the T-Server will be determined by the last party within the associated parties entries for the supplied interaction. No resource (thisDN) will be provided in this case to the target T-Server.

In cases where by the target T-Server cannot be determined from the information provided then an error.voice.privateservice will be generated. The following is an example, please consult your T-Server manual for explicit information and appropriate examples on `TPrivateService` and applicable parameters:

```
<state id="do_private_service">
<datamodel>
        <data id="reqid"/>
</datamodel>
<onentry>

        <script>
                var myuserdata = { details : { name : "Smith, John", age : 45 } };
                var myreasons = { code : "New Update"};
                var myextensions = { keyname : "Its value" };
        </script>

        <ixn:privateservice requestid="_data.reqid"
        serviceid="1234" interactionid="_data.ixnid"
        resource="'9000'"
        udata = "myuserdata"
        reasons = "myreasons"
        extensions = "myextensions"/>
</onentry>

<transition event="voice.privateservice.done"
```

```
        cond="_event.data.requestid==_data.reqid"   target="statex"/>
<transition event="error.voice.privateservice"
        cond="_event.data.requestid==_data.reqid"   target="statey"/>

</state>
```

Children

None

Events

The following events can be generated as part of this action, please refer to your specific T-Server manual for details of how and when these events are to be generated as they are specific to the service and T-Server implementation:

- `voice.privateservice.done` - This event is sent when the request has been accepted by the orchestration system and sent. It is not an indication that the T-Server has handled or accepted the event.

- `error.voice.privateservice` - This event is sent when the request itself has failed for some reason.

<userevent>

This action provides access to the Tlib function TSendUserEvent and allows distributing of specified user events to registered clients of the TServer. ORS only supports sending of user events and cannot be the recipient of such events from other sources, to interact with sessions directly from an external source it is recommended to use the External Interfaces which are described here External Interfaces

Attribute Details

Name	Required	Version	Type	Default Value	Valid Values	Description
requestid	false	8.1.2	location expression	none	Any valid location expression	This is the location for the request ID that is returned as part of this request. Any data model expression evaluating to a data model location. See SCXML Location Expressions for details. The location's value will be set to an internally generated unique string identifier to be associated with the action being sent. If this attribute is not

Name	Required	Version	Type	Default Value	Valid Values	Description
						specified, the event identifier is dropped. This identifier can be tested by the completion event handler to distinguish among several outstanding requests. If this attribute is not specified, the identifier can be acquired from the completion event. Every request must receive a unique identifier.
resource	true	8.1.2	value expression	none	Any valid string or resource object	A value expression which returns the Switch and DN that will be used send message. The receving client for the user event should be this DN See SCXML Legal Data Values and Value Expressions for details.
event	false	8.1.2	value expression	none	Any valid ECMAScript object containing the zero one or more of the following properties: • AttributeCallID • AttributeCallType • AttributeCustomerID • AttributeConnID • AttributeThisDN • AttributeOtherDN • AttributeThirdPartyDN	An ECMAScript Object which contains the list of key/value pairs which describe the content of sent event, with the exception of the user data, extensions and reasons which are provided if required in additional attributes

Name	Required	Version	Type	Default Value	Valid Values	Description
					• AttributeDNIS • AttributeANI • AttributeCollectedDigits • AttributeMediaType	for this action.
interactionid	false	8.1.2	value expression	none	interaction id	Id of interaction which properties will be partially used to populate attributes of distributed user event.
udata	false	8.1.2	ECMAScript Object	none	Any valid ECMAScript object	An ECMAScript Object which contains the list of key/value pairs which should be sent as user data with the user event See SCXML Legal Data Values and Value Expressions for details.
reasons	false	8.1.2	ECMAScript Object	none	Any valid ECMAScript object	An ECMAScript Object which contains the list of key/value pairs which should be sent as reason information with the user event See SCXML Legal Data Values and Value Expressions for details.
extensions	false	8.1.2	ECMAScript Object	none	Any valid ECMAScript object	An ECMAScript Object which contains the list of key/value pairs which should be sent as extensions with the user event. See SCXML Legal

Name	Required	Version	Type	Default Value	Valid Values	Description
						Data Values and Value Expressions for details.

Note 1. If the **interactionid** is provided in teh action then following 7 attributes will be copied from the specified interaction if they are not explcitely defined in **event** (and only them): AttributeCallID, AttributeCallType, AttributeCustomerID, AttributeConnID, AttributeDNIS, AttributeANI, AttributeCollectedDigits. **Note 2.** User Event Attributes type
AttributeCallID must be provided as string or number. If taken from interaction then matches to **voice.callid** property.

- **AttributeCallType** must be provided as string. If taken from interaction then matches to **voice.type** property.

- **AttributeCustomerID** must be provided as string. If taken from interaction then matches to **tenantid** property.

- **AttributeConnID** must be provided as string. If taken from interaction then matches to **voice.connid** property.

- **AttributeThisDN** must be provided as string. Cannot be taken from interaction.

- **AttributeOtherDN** must be provided as string. Cannot be taken from interaction.

- **AttributeThirdPartyDN** must be provided as string. Cannot be taken from interaction.

- **AttributeDNIS** must be provided as string. If taken from interaction then matches to **voice.dnis** property.

- **AttributeANI** must be provided as string. If taken from interaction then matches to **voice.ani** property.

- **AttributeCollectedDigits** must be provided as string. If taken from interaction then matches to **voice.ced** property.

- **AttributeMediaType** must be provided as number. Cannot be taken from interaction.

Sample:

```
<state id="do_userevent">
    <datamodel>
        <data id="reqid"/>
    </datamodel>
    <onentry>
        <script>
            var myuserdata = { "details": { "name": "Smith, John", "age": 45 } };
            var myreasons = { "code": "New Update" };
            var myextensions = { "keyname": "Its value" };
            var myevent = { "AttributeThisDN": "4000", "AttributeConnID":
genesys.ixn.interactions[0].voice.connid };
            var dest={ "switch":"target_switch", "dn" : "4000" };
        </script>
        <ixn:userevent requestid="_data.reqid"
            resource = "dest"
            event = "myevent"
```

```
        udata = "myuserdata"
        reasons = "myreasons"
        extensions = "myextensions"/>
    </onentry>
    <transition event="voice.userevent.done" cond="_event.data.requestid==_data.reqid"
target="statex"/>
    <transition event="error.voice.userevent" cond="_event.data.requestid==_data.reqid"
target="statey"/>
</state>
```

Children

None

Events

The following events can be generated as part of this action :

- `voice.userevent.done` - This event is sent when the request has been accepted by the orchestration system and sent. It is not an indication that the T-Server has handled or accepted the event.

- `error.voice.userevent` - This event is sent when the request itself has failed for some reason.

<querycalls>

This action queries a call from a specific resource.

Attribute Details

Name	Required	Type	Default Value	Valid Values	Description
requestid	false	location expression	none	Any valid location expression	This is the location for the request ID that is returned as part of this request. Any data model expression evaluating to a data model location. See SCXML Location Expressions for details. The location's value will be set to an internally generated unique string identifier to be associated with the action being sent. If this attribute is not specified, the event identifier is dropped. This identifier can be tested by the completion event handler to

Name	Required	Type	Default Value	Valid Values	Description
					distinguish among several outstanding requests. If this attribute is not specified, the identifier can be acquired from the completion event. Every request must receive a unique identifier.
resource	true	value expression	none	Any valid string or Resource Object	A value expression which returns the DN of the controlling agent or route point on whose behalf the information is provided. For details, see Addressing Resources. See SCXML Legal Data Values and Value Expressions

The following is an example:

```
<state id="do_querycalls">
<datamodel>
        <data id="reqid"/>
</datamodel>
<onentry>
        <ixn:querycalls requestid="_data.reqid" resource="'600'" />
</onentry>

<transition event="voice.querycalls.done"
        cond="_event.data.requestid==_data.reqid"  target="statex"/>
<transition event="error.voice.querycalls"
        cond="_event.data.requestid==_data.reqid"  target="statey"/>

</state>
```

Children

None

Events

The following events can be generated as part of this action:

- `voice.querycalls.done` - This event is sent when the request has been accepted by the orchestration system and sent.

- `error.voice.querycalls` - This event is sent when the request itself has failed for some reason.

Message Based Actions

The following are the msgbased-specific interaction actions.

<createmessage>

This action creates a new message and an associated interaction which can be used for the following purposes:

- Acknowledgements (source interaction - e-mail and open media)

- AutoResponses (source interaction - e-mail and open media)

- New e-mail notifications

- SMS messages (e-mail and native)

- Forwarded e-mail

- Reply e-mail from an external resource

- Redirect e-mail

- E-mail a chat transcript

- Create outbound e-mail

This action will also give the developer the options to send the message after it is created.

Attribute Details

Name	Required	Type	Default Value	Valid Values	Description
requestid	false	location expression	none	Any valid location expression	This is the location for the request ID that is returned as part of this request. Any data model expression evaluating to a data model location. See SCXML Location Expressions for details. The location's value will be set to an internally generated unique string identifier to be associated with the action being sent. If this attribute is not specified, the event identifier is dropped. This identifier can be tested by the completion event

Name	Required	Type	Default Value	Valid Values	Description
					handler to distinguish among several outstanding requests. If this attribute is not specified, the identifier can be acquired from the completion event. Every request must receive a unique identifier.
type	false	NMTOKEN	emailout_reply	acknowledgement, auto_response, customer_reply, outbound_new, emailout_reply, inbound_collaboration_reply, ndr_interaction, outbound_collaboration_invite, outbound_notification, redirect, forwarded	This specifies the type of message header that is to be used for the message.
media	false	NMTOKEN	_genesys.ixn.mediaType.TMedia.EMail	Values from the `_genesys.ixn.mediaType` enumeration object: TMediaAny, TMediaEMail, TMediaFax, TMediaNativeSMS, TMediaSMail, TMediaSMS, TMediaWebForm, TMediaOpenMedia	This is the media type of the message. The TMediaOpenMedia is only valid when the type attribute value is "acknowledgement" or "autoresponse".
server	false	value expression	If not supplied, the functional module will determine the server based on the type attribute	Any value expression that returns a valid string	A value expression which returns the name of the msgbased server to create this message on. See SCXML Legal Data Values and Value Expressions for details.
relatedixnid	false	value expression	none	Any value expression that returns a valid string	A value expression which returns the `_genesys.ixn.interactions[x].g_u` of the interaction that is associated with the interaction being created with this request. There are special values that can be returned:

Name	Required	Type	Default Value	Valid Values	Description
					• **"$NOT"** - means that there is not an interaction that is to be related to the new one. See SCXML Legal Data Values and Value Expressions for details.
thread	false	boolean expression	true	Any boolean expression that returns a true or false	A boolean expression which returns a boolean that identifies whether this new message is going to be part of the related interaction's thread. See SCXML Conditional Expressions for details.
includeorig	false	boolean expression	false	Any boolean expression that returns a true or false	A boolean expression which returns a boolean that identifies whether the original message is going to be added to this new message. See SCXML Conditional Expressions for details.
msgsrc	false	value expression	none	Any of the following valid URI schemes: • gdata	A value expression which returns a URI that identifies the location of the source message (suggested response, and so on) to use for this new message. The following are the URI schemes that are supported: • gdata • Configuration

Name	Required	Type	Default Value	Valid Values	Description
					data (for example, gdata:config\CA.name or gdata:config\SR.name) - This will be the name (id) of the suggested responses or categories from the Configuration Server. • User data (for example, gdata:udata) - this indicates that the message will be taken from the user data of the related interaction. So the related interaction must have key/value pairs for the CategoryId key. If SRLid and

Name	Required	Type	Default Value	Valid Values	Description
					CtgId are in the related interaction's message and the CategoryId in udata is null, then an error is generated (error.msgbased.create See SCXML Legal Data Values and Value Expressions for details.
to	false	value expression	none	Any list of valid addresses. This will be a single-quoted string with the address URIs separated by either a "," or ";".	A value expression which returns a list of addresses to send this message to. An example is to='joe@cox.com;joe@coy.com' or to="_origin.all". The following are the valid values for this attribute: • _origin • _origin.all • _udata - if this URI value is used, then this can be the only entry in list. You can also specify a specific key name to be using _udata\\<keyname>. • List of addresses See SCXML Legal Data Values and Value Expressions for details.

Name	Required	Type	Default Value	Valid Values	Description
from	false	value expression	none	Any valid e-mail address(es). If multiple addresses are supplied, the string with the URIs will be single-quoted and will be separated by either a "," or ";".	A value expression which returns the address(es) that this message will be from. An example is from="joe@cox.com" or from="_origin". The following are the valid values for this attribute: • _origin • List of addresses **Note:** This attribute is required when the type attribute value is: • outbound_new See SCXML Legal Data Values and Value Expressions
cc	false	value expression	none	Any list of valid addresses. This will be a single-quoted string with the address URIs separated by either a "," or ";".	A value expression which returns a list of addresses to send a copy of this message to. An example is cc='joe@cox.com;joe@coy.com'. cc="_origin" The following are the valid values for this attribute: • _origin • List of addresses See SCXML Legal Data Values and Value Expressions
exclude	false	value expression	none	Any list of valid addresses. This will be a single-quoted string with the address URIs separated by either a "," or ";".	A value expression which returns a list of addresses to exclude from the "to" or "cc" list of addresses. An example is exclude='joe@cox.com;joe@coy.com'. The following are the valid values for this attribute:

Name	Required	Type	Default Value	Valid Values	Description
					• List of addresses See SCXML Legal Data Values and Value Expressions
subject	false	value expression	none	Any valid string	A value expression which returns the subject of the new message. There are special values that can be returned: • **"$USESRL"** means the functional module will use the subject from the suggested response template. If not specified, there will be no subject for the new message. See SCXML Legal Data Values and Value Expressions for details.
queue	false	value expression	none	Any valid string	A value expression which returns the queue that the new interaction or message should be put into when it is created. The SCXML session associated with this queue will be responsible for explicitly sending this message or interaction later.The following are the values that can be provided and what processing will be done: • **"$CURQUEUE"** means that the current

Name	Required	Type	Default Value	Valid Values	Description
					queue associated with the interaction is used. This value cannot be used if the relatedixnid value is "$NOT". • If no value is specified the message will be sent by platform automatically. The session will not get any other events (except masgbased.createmessage related with this message. It is recommended that you do not use this attribute and allow the functional module to send the interaction or message. See SCXML Legal Data Values and Value Expressions for details.
chattranscript	false	boolean expression	false	Any valid boolean value	A boolean expression which identifies whether the chat transcript of the related interaction (must be a chat interaction) should be attached to the message. This attribute is only valid when the relatedixn attribute does not have a value of "NOT". See SCXML

Name	Required	Type	Default Value	Valid Values	Description
					Conditional Expressions for details.
delivery	false	boolean expression	false	Any boolean expression that returns true or false	A boolean expression which returns a boolean that identifies whether this message being sent should include a request for a return message indicating whether and how the original message was delivered. • If one of the SMTP servers involved in the transport of the original message fails to deliver it, the return message comes into the system with subtype InboundNDR. It contains no additional information. • If the original message is successfully delivered, the return message comes into the system with subtype InboundReport. It uses attached data to

Name	Required	Type	Default Value	Valid Values	Description
					indicate delivery statuses such as delayed, delivered, relayed, and so on. This attribute is only used when the "send" attribute is equal to "true". User's Guide. See SCXML Conditional Expressions for details.
disposition	false	boolean expression	false	Any boolean expression that returns true or false	A boolean expression which returns a boolean that identifies whether the message should include a request for a return message indicating what happened to the original message after it was delivered; for example, whether it was displayed, printed, deleted without displaying, and so on. The return message comes into the system with subtype InboundDisposition and it provides the delivery status in attached data. This attribute is only used when the "send" attribute is equal to "true". See SCXML Conditional Expressions for details.
codefields	false	ECMAScript object	none	Any valid ECMAScript object with a set of key/value properties	An ECMAScript Object which contains the list of key/value pairs which should be used to substitute the values into the

Name	Required	Type	Default Value	Valid Values	Description
					suggested response message that is created. See SCXML Legal Data Values and Value Expressions for details.
associated	false	boolean expression	true	Any valid boolean value	Release 8.1.400.36 introduces an optional Boolean attribute, associated, for the <ixn:createmessage> action. The default value is true to comply with previous behavior. If the associated attribute is set to false, a new interaction is not associated with the session that created it. In this case, the creating session should not expect any asynchronous events (such as interaction.added, and so on) that are related to the new interaction and the only published event will be msgbased.createmessage.done.

The following is an example:

```
<state id="do_simple_autorsp_or_Ack">
<datamodel>
        <data id="reqid"/>
</datamodel>
<onentry>
        <ixn:createmessage requestid="_data.reqid" type="customer_reply"
server="'JEmailServer1'"
                        msgsrc="'gdata:config\CA.SR27'" includeorig="true" to="'_origin.all'"
                        from="'_origin'" subject="'$USESRL'">
                <ixn:field key="'$servicename'" value="_data.service"/>
        </ixn:createmessage>
</onentry>
<transition event="msgbased.createmessage.done" target="statex"/>
<transition event="error.msgbased.createmessage" target="statey"/>
</state>
<state id="do_new_email">
<datamodel>
        <data id="reqid"/>
</datamodel>
<onentry>
        <ixn:createmessage requestid="_data.reqid" type="emailout_reply"
```

```
server="'JEmailServer1'"
                        msgsrc="'gdata:config\CA.SR45'" relatedixn="'$NOT'"
to="'joee@abc.com'"
                        from="'doitco@xyz.com'" subject="'$USESRL'">
            <ixn:field key="'$servicename'" value="_data.service"/>
        </ixn:createmessage>
</onentry>
<transition event="msgbased.createmessage.done" target="statex"/>
<transition event="error.msgbased.createmessage" target="statey"/>
</state>
<state id="do_new_sms">
<datamodel>
        <data id="reqid"/>
</datamodel>
<onentry>
        <ixn:createmessage requestid="_data.reqid" type="outbound_notification"
server="'JEmailServer1'"
                        msgsrc="'gdata:config\CA.SR888'" relatedixn="'$NOT'"
to="'joee@abc.com'"
                        from="'doitco@xyz.com'" subject="'Hello to Bank xyz'">
            <ixn:field key="'$custname'" value="_data.custname"/>
        </ixn:createmessage>
</onentry>
<transition event="msgbased.createmessage.done" target="statex"/>
<transition event="error.msgbased.createmessage" target="statey"/>
</state>
<state id="do_forward">
<datamodel>
        <data id="reqid"/>
</datamodel>
<onentry>
        <ixn:createmessage requestid="_data.reqid" type="forwarded" server="'JEmailServer1'"
                thread="false" to="'joee@abc.com'" from="'doitco@xyz.com'"/>
</onentry>
<transition event="msgbased.createmessage.done" target="statex"/>
<transition event="error.msgbased.createmessage" target="statey"/>
</state>
<state id="do_redirect">
<datamodel>
        <data id="reqid"/>
</datamodel>
<onentry>
        <ixn:createmessage requestid="_data.reqid" type="redirect" server="'JEmailServer1'"
                thread="false" to="'joee@abc.com'" from="'doitco@xyz.com'"/>
</onentry>
<transition event="msgbased.createmessage.done" target="statex"/>
<transition event="error.msgbased.createmessage" target="statey"/>
</state>
<state id="do_reply_from_extresource">
<datamodel>
        <data id="reqid"/>
</datamodel>
<onentry>
        <ixn:createmessage requestid="_data.reqid" type="inbound_collaboration_reply"
                server="'JEmailServer1'" thread="false" to="'joee@abc.com'"
                from="'doitco@xyz.com'"/>
</onentry>
<transition event="msgbased.createmessage.done" target="statex"/>
<transition event="error.msgbased.createmessage" target="statey"/>
</state>
<state id="do_chat_transcript">
<datamodel>
        <data id="reqid"/>
```

```
</datamodel>
<onentry>
        <ixn:createmessage requestid="_data.reqid" type="emailout_reply"
                server="'JEmailServer1'" msgsrc="'gdata:config\CA.SR988'"
                thread="false" relatedixnid="_genesys.ixn.interactions[x].g_uid"
                to="'joee@abc.com'" from="'doitco@xyz.com'" chattranscript="true"/>
</onentry>
<transition event="msgbased.createmessage.done" target="statex"/>
<transition event="error.msgbased.createmessage" target="statey"/>
</state>
<state id="do_new_task">
<datamodel>
        <data id="reqid"/>
</datamodel>
<onentry>
        <ixn:createmessage requestid="_data.reqid" type="outbound_new"
                server="'JEmailServer1'" msgsrc="'gdata:config\CA.SR999'"
                thread="false" relatedixnid="'$NOT'" to="'agent1'"
                from="'doitco@xyz.com'"/>
</onentry>
<transition event="msgbased.createmessage.done" target="statex"/>
<transition event="error.msgbased.createmessage" target="statey"/>
</state>
```

Required Attributes based on Request and Media Type

Note: For <createmessage>, all request types except outbound_new require a related interaction (relatedixnid). For send, all requests require a related interaction (relatedixnid). The related interaction id defaults to _genesys.ixn.interactions[main_ixnid].g_uid if none is specified on the request (so it's not listed as a required attribute in the following table).

Type	Media	Required Attributes	Comments
acknowledgement, auto_response	TMediaEMail, TMediaOpenMedia	msgsrc	msgsrc may be gdata:config\ CA.id or gdata:config\SA.id or gdata:udata A valid related interaction is required.
acknowledgement, auto_response	TMediaNativeSMS	msgsrc	msgsrc contains text string of message. A valid related interaction is required.
forwarded	TMediaEMail	msgsrc, to	msgsrc must be gdata:config\ SA.id A valid related interaction is required.
inbound_collaboration_reply	TMediaEMail		no required attributes A valid related interaction is required. The related interaction must be of type "InboundCollaborationReply"
outbound_new	TMediaEMail, TMediaSMS	msgsrc, from	If 'to' is not specified then 'ContactId' must be in user

Type	Media	Required Attributes	Comments
			data of related interaction. msgsrc may be gdata:config\ CA.id or gdata:config\SA.id or gdata:udata
outbound_new	TMediaNativeSMS	msgsrc, to, from	msgsrc contains text string of message.
outbound_notification	TMediaEMail	msgsrc	A valid related interaction is required.
redirect	TMediaEMail	to	A valid related interaction is required.

Children

None

Events

The following events can be generated as part of this action:

- `msgbased.createmessage.done` - This event is sent when the request has been accepted by the system, creating the msgbased interaction.

- `error.msgbased.createmessage` - This event is sent when the request has failed for some reason.

- `interaction.present` - This event is sent when the actual new msgbased interaction is created and is available to the session (if the queue is not specified).

<sendmessage>

This action sends a message that was created either by the `<createmessage>` action or by an outside source (for example, agent desktop).

Attribute Details

Name	Required	Type	Default Value	Valid Values	Description
requestid	false	location expression	none	Any valid location expression	This is the location for the request ID that is returned as part of this request. Any data model expression evaluating to a data model location. See SCXML Location

Name	Required	Type	Default Value	Valid Values	Description
					Expressions for details. The location's value will be set to an internally generated unique string identifier to be associated with the action being sent. If this attribute is not specified, the event identifier is dropped. This identifier can be tested by the completion event handler to distinguish among several outstanding requests. If this attribute is not specified, the identifier can be acquired from the completion event. Every request must receive a unique identifier.
server	false	value expression	If not supplied, the functional module will determine the server based on the media type of the interaction	Any value expression that returns a valid string	A value expression which returns the name of the msgbased server to create this message on. See SCXML Legal Data Values and Value Expressions for details.
interactionid	false	value expression	none	Any value expression that returns a valid string	A value expression which returns the `_genesys.ixn.interactions[x].g_u` of the interaction that is associated with this request. There is a special value that can be returned: • ECMAScript **Null** means the functional module will not use an interaction for the request.

Name	Required	Type	Default Value	Valid Values	Description
					See SCXML Legal Data Values and Value Expressions for details.
delivery	false	boolean expression	false	Any boolean expression that returns a true or false	A boolean expression which returns a boolean which identifies if the message being sent should include a request for a return message indicating whether and how the original message was delivered. • If one of the SMTP servers involved in the transport of the original message fails to deliver it, the return message comes into the system with subtype InboundNDR. It contains no additional information. • If the original message is successfully delivered, the return message comes into the system with subtype InboundReport. It uses attached data to

Name	Required	Type	Default Value	Valid Values	Description
					indicate delivery statuses such as delayed, delivered, relayed, and so on. See SCXML Conditional Expressions for details.
disposition	false	boolean expression	false	Any boolean expression that returns a true or false	A boolean expression which returns a boolean that identifies whether the message should include a request for a return message indicating what happened to the original message after it was delivered; for example, whether it was displayed, printed, deleted without displaying, and so on. The return message comes into the system with subtype InboundDisposition, and it provides the delivery status in attached data. See SCXML Conditional Expressions for details.
from	false	value expression	none	A valid address URI	A value expression which returns the address that this message will be from. If specified, it will overwrite the current "from" address in the message. An example is from="joe@cox.com" The following are the valid values for this attribute: • address

Name	Required	Type	Default Value	Valid Values	Description
					See SCXML Legal Data Values and Value Expressions
cc	false	value expression	none	Any list of valid addresses. This will be a single-quoted string with the address URIs separated by either a "," or ";".	A value expression which returns a list of additional addresses to send a copy of this message to. These addresses are added to the current list of carbon copy addresses in the message being sent. An example is cc='joe@cox.com;joe@coy.com'. The following are the valid values for this attribute: • List of addresses See SCXML Legal Data Values and Value Expressions
exclude	false	value expression	none	Any list of valid addresses. This will be a single-quoted string with the address URIs separated by either a "," or ";".	A value expression which returns a list of additional addresses to exclude from the "to" or "cc" list of addresses. An example is exclude='joe@cox.com;joe@coy.com'. The following are the valid values for this attribute: • List of addresses See SCXML Legal Data Values and Value Expressions
subject	false	value expression	none	Any valid string	A value expression which returns the subject of the message that is to be sent. This will overwrite the subject currently assigned to the message. See SCXML Legal Data

Name	Required	Type	Default Value	Valid Values	Description
					Values and Value Expressions for details.
headerfields	false	ECMAScript object	none	Any valid ECMAScript object with a set of key/value properties.	An ECMAScript Object which contains the list of key/value pairs which should be used to add headers to the message when it is sent. See SCXML Legal Data Values and Value Expressions for details.

The following is an example:

```
<state id="do_send">
<datamodel>
        <data id="reqid"/>
</datamodel>
<onentry>
        <ixn:sendmessage requestid="_data.reqid" server="'JEmailServer1'"
                delivery="true"/>
</onentry>
<transition event="msgbased.sendmessage.done" target="statex"/>
<transition event="error.msgbased.sendmessage" target="statey"/>
</state>
```

Children

None

Events

The following events can be generated as part of this action:

- `msgbased.sendmessage.done` - This event is sent when the request has been accepted by the system and the msgbased message is being sent.

- `error.msgbased.sendmessage` - This event is sent when the request has failed for some reason.

<getcontent >

This action gets the latest version of the message and updates the `_genesys.ixn.interactions[].msgbased.content` property

Attribute Details

Name	Required	Type	Default Value	Valid Values	Description
requestid	false	location expression	none	Any valid location expression	This is the location for the request ID that is returned as part of this request. Any data model expression evaluating to a data model location. See SCXML Location Expressions for details. The location's value will be set to an internally generated unique string identifier to be associated with the action being sent. If this attribute is not specified, the event identifier is dropped. This identifier can be tested by the completion event handler to distinguish among several outstanding requests. If this attribute is not specified, the identifier can be acquired from the completion event. Every request must receive a unique identifier.
interactionid	false	value expression	none	Any value expression that returns a valid string	A value expression which returns the `_genesys.ixn.interactions[x].g_u` of the interaction that is associated with this request. There is a special value that can be returned: • ECMAScript **Null** means the functional module will not use an interaction for the

Name	Required	Type	Default Value	Valid Values	Description
					request. See SCXML Legal Data Values and Value Expressions for details.

The following is an example:

```
<state id="do_getContent">
<datamodel>
        <data id="reqid"/>
        <data id="ixnid"/>
        <data id="currentixn"/>
</datamodel>
<onentry>
        <ixn:getcontent requestid="_data.reqid"
                interactionid="'_genesys.ixn.interactions[_data.ixnid].g_uid"/>
</onentry>
<transition event="msgbased.getcontent.done" target="statex"/>
<transition event="error.msgbased.getcontent" target="statey"/>
</state>
```

Children

None

Events

The following events can be generated as part of this action:

- msgbased.getcontent.done - This event is sent when the request has been accepted by the system and the content has been refreshed.

- error.msgbased.getcontent - This event is sent when the request has failed for some reason.

Chat Actions

The following are the chat-specific interaction actions.

<gettranscript >

This action gets the latest version of the chat transcript and returns it in the chat.gettranscript.done event.

Attribute Details

Name	Required	Type	Default Value	Valid Values	Description
requestid	false	location expression	none	Any valid location expression	This is the location for the request ID that is returned as part of this request. Any data model expression evaluating to a data model location. See SCXML Location Expressions for details. The location's value will be set to an internally generated unique string identifier to be associated with the action being sent. If this attribute is not specified, the event identifier is dropped. This identifier can be tested by the completion event handler to distinguish among several outstanding requests. If this attribute is not specified, the identifier can be acquired from the completion event. Every request must receive a unique identifier.
interactionid	false	value expression	none	Any value expression that returns a valid string	A value expression which returns the `_genesys.ixn.interactions[x].g_u` of the interaction that is associated with this request. There is a special value that can be returned: • ECMAScript **Null** means the functional module will not use an interaction for the

Name	Required	Type	Default Value	Valid Values	Description
					request. See SCXML Legal Data Values and Value Expressions for details.

The following is an example:

```
<state id="do_getTranscript">
<datamodel>
        <data id="reqid"/>
        <data id="currentixn"/>
</datamodel>
<onentry>
        <ixn:gettranscript requestid="_data.reqid"
                interactionid="'_genesys.ixn.interactions[1].g_uid"/>
</onentry>
<transition event="chat.gettranscript.done" target="statex"/>
<transition event="error.chat.gettranscript" target="statey"/>
</state>
```

Children

None

Events

The following events can be generated as part of this action:

- `chat.gettranscript.done` - This event is sent when the request has been accepted by the system and the transcript has been refreshed.
- `error.chat.gettranscript` - This event is sent when the request has failed for some reason.