



This PDF is generated from authoritative online content, and is provided for convenience only. This PDF cannot be used for legal purposes. For authoritative understanding of what is and is not supported, always use the online content. To copy code samples, always use the online content.

Orchestration Server Deployment Guide

Enhanced Routing and Interaction Submitter Script Options

12/19/2025

Contents

- 1 Enhanced Routing and Interaction Submitter Script Options
 - 1.1 Enhanced Routing Script Options
 - 1.2 Parameter Substitution
 - 1.3 Interaction Submitter Script Options

Enhanced Routing and Interaction Submitter Script Options

Enhanced Routing Script Options

Application Section

This section describes Enhanced Routing Script object Application section options.

alternate-url

Option section: Application

Configuration object: Script object (CfgEnhancedRouting)

Default value: none

Valid values: any valid URL

Value changes: For the next interaction that hits this resource

This option specifies the URL of the SCXML document to load. This is attempted if the value of the url option cannot be fetched or compiled (for example: application server is temporarily down, network error, script is malformed, and so on).

This is the option to use for <callback> URI functionality, as a failover mechanism to provision the SCXML application from an ORS Web Server.

The URL can be any one of the following protocols:

- file:<path>
- http://<url>
- https://<url>

The URL can also contain parameters which will be passed to the Application server. The values shown below are substituted at run-time based on the information in the interaction.

Parameter Element Descriptions

- [DNIS] The DNIS attribute of the interaction
- [ANI] The ANI attribute of the interaction
- [DN] The ThisDN attribute of the interaction

- [CED] The CollectedDigits attribute of the interaction
- [EMAILFROM] E-mail from address
- [EMAILTO] E-mail to address
- [EMAILSUBJECT] E-mail subject
- [UDATA] Expanded to the entire user data of the interaction in the format: name1=value1&name2=value2&...
- [UDATA.*] Expanded to the entire user data of the interaction in the format:name1:value1,name2:value2,...
- [UDATA.name] Expanded to the value of a specific user data key of the interaction value. For example: &servicetype=[UDATA.ServiceType] could resolve to &servicetype=CreditCards

Simple case

alternate-url = http://xserver.genesyslab.com:80/NewCallReq.asp

With parameters:

alternate-url = https://xserver.genesyslab.com:80/
NewCallReq.asp?ani=[ANI]&dnis=[DNIS]&servicetype=[UDATA.ServiceType]

Parameter Substitution

Orchestration Server allows substitution of parameters, provided in the URL of an SCXML application, with predefined values. There are several ways to utilize this functionality.

- Configure parameters in an Enhanced Routing Script object in Configuration Layer.
The URL of the SCXML application specified in the url option or alternate-url option may contain parameters in the format \$\$parameter-name\$\$ where parameter-name should match the name of the option configured in the ApplicationParms section. When this parameter-name is found, the parameter will be substituted for its value. For example, configure the following options in the ApplicationParms section:

```
APPSERVER1=http://appserver:8080 REGION1=CA REGION2=NY
```

- Configure the url and alternate-url options in the Application section:

```
url=$$APPSERVER1$$/service/$$REGION1$$/Workflow1.scxml
```

```
alternate-url=$$APPSERVER1$$/service/$$REGION2$$/Workflow2.scxml
```

The url will be substituted at run-time with:

```
http://appserver:8080/service/CA/Workflow1.scxml
```

The alternate-url will be substituted at run-time with:

```
http://appserver:8080/service/NY/Workflow2.scxml
```

- Sessions started using an HTTP request with parameters. For example:

```
http://localhost:17011/scxml/session/start
```

```
src=http://appserver:8080/service/CA/Workflow1.scxml&APPSERVER1=http://appserver:8080@I0N1=CA
```

These parameters can be utilized in a started SCXML application as a value of the href attribute in an `<xi:include>` action element. For example:

```
<xi:include href="$$APPSERVER1$$/service/$$REGION1$$/RouteToAgGroup.scxml" .../>
```

At run-time, the href value will be substituted with:

```
"http://appserver:8080/service/CA/RouteToAgGroup.scxml"
```

- Session started with parameters specified in `<session:start>` action element. For example:

```
<session:start sessionid="_data.NewID" src="_data.Path + '/Workflow1.scxml'" >
```

```
<param name="APPSERVER1" expr="'http://appserver:8080'"/>
```

```
<param name="REGION1" expr="'CA'"/>
```

```
</session:start>
```

Notes:

- Orchestration Server now sends the HTTP fetch request to the address specified in the `alternate-url` option after the `fetch-timeout` has expired.
- `parameter-name` can be used in all subroutines started from the main SCXML session. To avoid unexpected outcomes, Genesys recommends avoiding a `parameter-name` with a space at the beginning, end, or in the middle. `parameter-name` is case-sensitive.

fetch-timeout

Option section: `scxml` in an ORS Application object; Application in an Enhanced Routing Script object

Configuration object: ORS Application object; Enhanced Routing Script object

Default value: 5000

Valid values: Any integer greater than 0

Value changes: After restart

This option specifies the time (in milliseconds) before a document fetch operation is abandoned. If the SCXML document cannot be retrieved within this timeout value, the fetch operation is abandoned and a fetch operation for the `alternate-url` is attempted.

Orchestration Server sends the HTTP fetch request to the address specified in the `alternate-url` option after `fetch-timeout` has expired.

http-useragent

Option section: Application

Configuration object: Script object (CfgEnhancedRouting)

Default value: G0ES/8.0

Valid values: Any string

Value changes: For the next document fetch

Specifies the value of the User-Agent header that ORS includes into originating HTTP requests that it sends when fetching an SCXML document (script) from an application server. For example: http-User-Agent = MYAGENT

http-version

Option section: Application

Configuration object: Script object (CfgEnhancedRouting)

Default value: 1.1

Valid values: 1.0, 1.1

Value changes: For the next document fetch

This option specifies the HTTP version to use when fetching an SCXML document from an application server.

For example: http-version = 1.1

ixnfm-idle--session-ttl

Option section: orchestration (in Application) or application (in Enhanced Routing Script)

Configuration Object: Enhanced Routing Script, ORS Application

Default value: 0

Valid values: Any non-negative integer between 0 and 3600

Value changes: Immediately upon notification.

Defines the maximum time that an SCXML session may exist only if interactions have been successfully redirected and there are no other interactions subsequently attached to that session. If configured in Enhanced Routing Script, the value will override value from the ORS Application for sessions created from that Script.

Use to eliminate the possibility of the scenario where an SCXML session session gets "stuck" in memory caused by an unsuccessful detach action execution and/or incorrect strategy design. This scenario can prevent the creation of new sessions if the number of sessions reaches the limit of concurrent active sessions per ORS Application.

Value 0 disables this feature.

max-age

Option section: `Application`

Configuration object: `Script (CfgEnhancedRouting)` object

Default value: 0

Valid values: Any integer greater than or equal to 0

Value changes: For the next change of the SCXML document

Defines time in seconds for how long a fetched SCXML application is cached. When the same SCXML application is to be used for the new session within the configured timeout, the cached version of the document will be used instead of fetching it from the application server.

- If the option value is empty or set to 0, fetching this document from the cache will be disabled.
- If max-age is configured for an Enhanced Routing Script object, the value from max-age takes precedence over the http-max-age **Application level** option.
- If the URL of SCXML application contains parameters that are unique for each invocation, then the application will be fetched each time regardless of the option value.

max-script-duration

Option section: `scxml` (in the ORS Application object) or `Application` (in an Enhanced Routing Script object)

Configuration object: `ORS Application` object and `Enhanced Routing Script` object

Default value: 2,000 milliseconds

Valid values: 500 - 10,000 milliseconds

Value changes: When configured on an ORS Application object, takes effect during startup. When configured on an Enhanced Routing Script object, takes effect for sessions that started after that update. When configured in an Enhanced Routing Script object, that option takes precedence over the option configured in the ORS Application object.

If the execution time for JavaScript within a `<Script>` element exceeds the configured limit, execution will be interrupted, an Alarm-level message will be printed in ORS log, and the `error.script` Script event will be raised for the corresponding session.

max-stale

Option section: `Application`

Configuration object: `Script (CfgEnhancedRouting)` object

Default value: 0

Valid values: Any integer greater than or equal to 0

Value changes: For the next fetch of the SCXML document

Defines the time in seconds to extend the lifetime of a cached SCXML application. For example, if the value of the max-age option is set to 60 seconds and the value of the max-stale option is set to 30 seconds, then the fetched document will be cached for the timeframe of 90 seconds.

- If the option value is empty or set to 0, fetching this document from the cache will be disabled.
- If max-stale is configured for an Enhanced Routing Script object, the value from max-stale takes precedence over the http-max-stale Application level option.

url

Option section: `Application`

Configuration object: `Script (CfgEnhancedRouting)` object

Default value: none (this is a required option)

Valid values: any valid URL

Value changes: For the next interaction that hits this resource

This option specifies the URL of the SCXML document to load. The URL can be any one of the following protocols:

- `file:<path>`
- `http://<url>`

The URL can also contain parameters which will be passed to the Application server. The values shown in Parameter Element Descriptions below are substituted at run-time based on the information in the interaction.

Parameter Element Descriptions

- `[DNIS]` The DNIS attribute of the interaction
- `[ANI]` The ANI attribute of the interaction
- `[DN]` The ThisDN attribute of the interaction
- `[CED]` The CollectedDigits attribute of the interaction
- `[EMAILFROM]` E-mail from address
- `[EMAILTO]` E-mail to address
- `[EMAILSUBJECT]` E-mail subject
- `[UDATA]` Expanded to the entire user data of the interaction in the format: `name1=value1&name2=value2&...`
- `[UDATA.*]` Expanded to the entire user data of the interaction in the format: `name1:value1,name2:value2,...`
- `[UDATA.name]` Expanded to the value of a specific user data key of the interaction value. For example:

`&servicetype=[UDATA.ServiceType]` could resolve to `&servicetype=CreditCards`

For example:

`url = http://xserver.genesyslab.com:80/NewCallReq.asp`

`url = http://xserver.genesyslab.com:80/
NewCallReq.asp?ani=[ANI]&dnis=[DNIS]&servicetype=[UDATA.ServiceType]`

Orchestration Server allows substitution of parameters, provided in the URL of an SCXML application, with predefined values. For more information, see the `alternate-url` option.

{Parameter Name}

Option section: `ApplicationParms`

Configuration object: `Script object (CfgEnhancedRouting)`

Default value: none

Valid values: Any string

Value changes: For the next interaction that hits this resource

This option specifies a string that represents a parameter value to be passed to the application.

The `ApplicationParms` section contains the values for data elements that may be referred to within the SCXML application. The parameters can be statically defined for each application (for example, `Service = Sales`) or contain substitution values that will be substituted at run-time based on the information in the interaction.

For example: `Segment = [UDATA.CustomerSegment]`

Parameter Element Descriptions

- `[DNIS]` The DNIS attribute of the interaction
- `[ANI]` The ANI attribute of the interaction
- `[DN]` The `ThisDN` attribute of the interaction
- `[CED]` The `CollectedDigits` attribute of the interaction
- `[EMAILFROM]` E-mail from address
- `[EMAILTO]` E-mail to address
- `[EMAILSUBJECT]` E-mail subject
- `[UDATA.name]` Expanded to the value of a specific user data key of the interaction value. For example:
`&servicetype=[UDATA.ServiceType]` could resolve to `&servicetype=CreditCards`

For example:

`ApplicationParms =`

`Service = Sales`

```
EmailSubject = [EMAILSUBJECT]
Segment = [UDATA.CustomerSegment]
AppServer = http://myappserver:8080
```

Interaction Submitter Script Options

This section describes `Interaction Submitter Script` object options.

Orchestration Section

`cluster-id`

Option section: Orchestration in Annex of the `Interaction Submitter Script` object

Configuration object: `Interaction Submitter Script` object

Default value: None

Valid values: List of ORS clusters separated by ‘;’

Value changes: Immediately upon notification. Applicable to subsequent pull requests.

Starting with release 8.1.400.36, ORS provides the capability to define which ORS cluster will process interactions. For deployments with **Enhanced Pulling** of multimedia interactions from interaction queues, you configure the `cluster-id` option.

When option `cluster-id` is configured, it restricts usage of the Submitter to only the ORS nodes that belong to the clusters listed in this option. If this option is not configured in a Submitter, then the Submitter can be used by any ORS node. Also see Submitter Objects in the Composer **Interaction Queue Block** topic.