



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.

# Callback User's Guide

Accept Inbound Calls

12/18/2025

# Accept Inbound Calls

For some voice scenarios, you must create an SCXML scenario to manage inbound calls related to Callback. The customer retrieves a number to call that is associated with a list of DNS dedicated to the Callback scenarios. This page details how you create configuration objects and scripts related to these inbound calls.

## Contents

- [1 Accept Inbound Calls](#)
  - [1.1 Setting up Acceptance of an Inbound Call](#)
  - [1.2 Setting up Acceptance of a Classic Inbound Call](#)

## Setting up Acceptance of an Inbound Call

To enable the Callback service to accept an inbound call, Orchestration Server (ORS) requires a SCXML file to manage inbound calls. On this page, samples use the `inbound.scxml` available in the <GMS installation directory>\service\_templates\callback.zip file. Unzip the file. The `inbound.scxml` file is located in the <GMS installation directory>\service\_templates\callback\src directory.

### Tip

You can customize the `inbound.scxml` file or create new SCXML files dedicated to Callback in this directory.

## Create a Callback Inbound Script Resource

The screenshot shows the GAX Configuration page for the `GMSInboundUISample Properties`. The `Options` tab is selected, displaying a table of configuration options. A handwritten note "URL of the inbound.scxml file" with an arrow points to the `url` key value.

Key	Value
Application	
url	http://135.39.45.125:8010/genesys/1/document/service_template/callback...
ApplicationParms	
_treatment_customer_connect	http://localhost:8010/genesys/1/document/service_template/callback/Re...
_treatment_find_agent_fail	http://localhost:8010/genesys/1/document/service_template/callback/Re...
_treatment_waiting_for_agent	http://localhost:8010/genesys/1/document/service_template/callback/Re...
app_find_agent_timeout	120
app_gms_offer_callback	true
app_gms_service_name	samples
app_gms_service_uri	http://135.39.45.125:8010/genesys/1/service/samples
app_selected_agent_group	Customer_Service

Navigate to the **Environment > Scripts** section of Genesys Administrator Extension and click **New** to create the script resource.

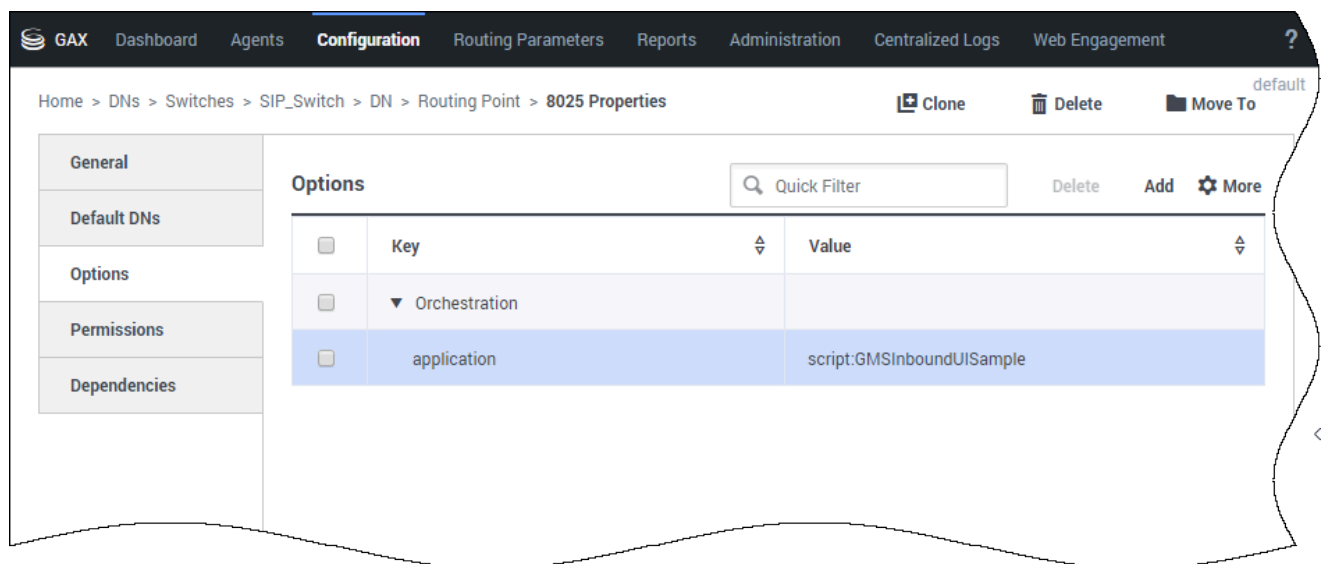
1. Enter a name, for instance GMSInboundUISample, select the **SCXML** type, and click **Save**.
2. In the **Options** tab, enter the Application > url of your SCXML file:  
`http://<gmshost>:<gmsport>/genesys/1/document/service_template/callback/src/inbound.scxml`

3. Then, create the following parameters in the ApplicationParms section:

```
app_find_agent_timeout = <timeout in seconds when routing to agent for GMS match fails>
app_selected_agent_group = <agent_group for GMS match fails>
treatment_for_no_match = <treatment_file prior to routing on GMS match fail>
require_access_code = <true/false>
require_ani = <true/false>
```

4. Save your script object.

## Create DNs for your Callback Inbound Script Resource



Browse or create DNs of type Routing Point in **Switching > DNs > Switches > SIP\_Switch > DN > Routing Point**. **Check the Options** tab. The Orchestration.application value must be set to script:<gms-inbound-sample-script> where gms-inbound-sample-script is the name of the inbound script that you have just created. In our example, Orchestration.application=script:GMSInboundUISample.

## Setting up Acceptance of a Classic Inbound Call

### What are the differences between an Inbound Call and a Classic Inbound Call?

In a typical Callback scenario, a service is created first and then, the call arrives/initiates. In a Classic call scenario, the arrival of the call results in the creation of a Callback service.

The Classic inbound service does the following:

1. Creates a Callback (user originated) service.
2. Transfers the call to the Callback service created in the previous step.

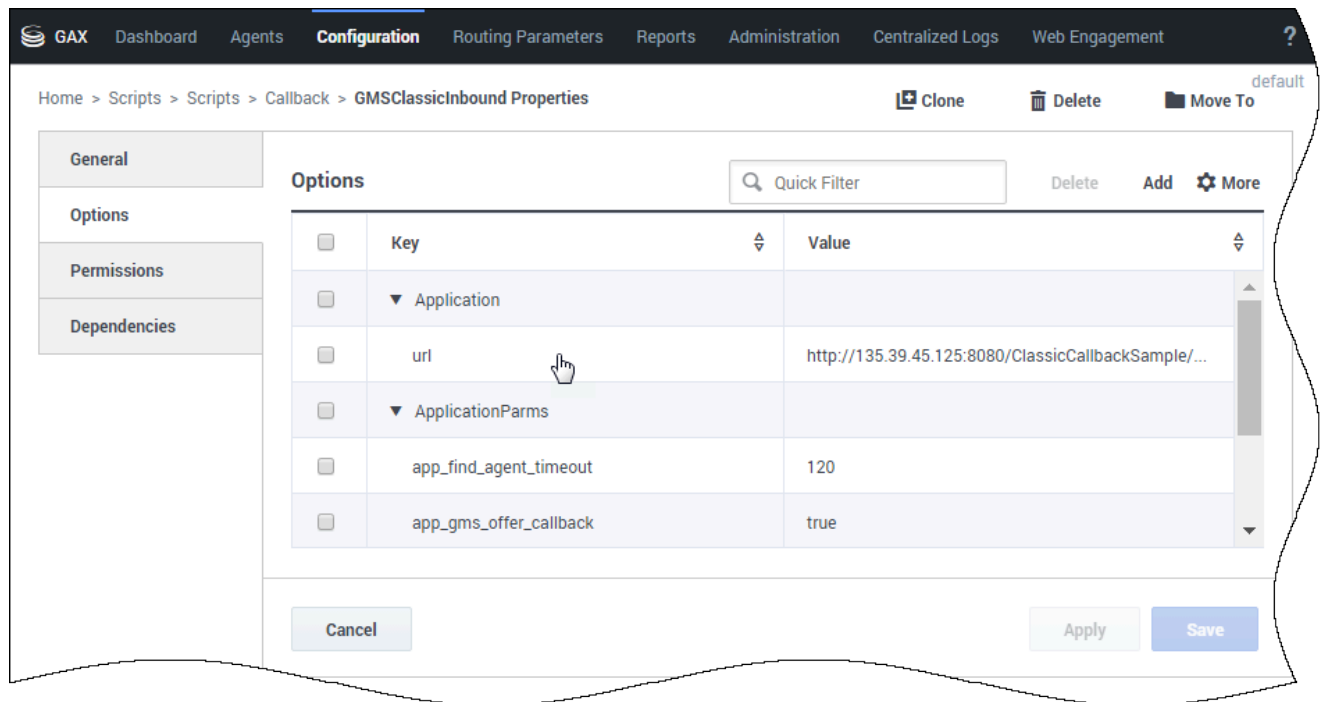
To create and accept an inbound call, you must set up a routing point to load the Classic Inbound workflow.

#### Important

The `IPD_ClassicCallInbound_Entry.sxml` file can be downloaded from the [Classic Callback Sample](#) page.

Before configuring your script, you must first setup the classic callback sample: [See: Running the Sample](#).

## Setting up your Classic Inbound script



Create an **Enhanced Routing** object. Navigate to the **Environment > Scripts** section of Genesys Administrator Extension and click **New** to create the script resource.

1. Enter a name, for instance GMSClassicInbound, select the **SCXML** type, and click **Save**.
2. In the **Options** section, enter the URI of your SCXML file in the Application.url option:  
`http://<gmshost>:<gmsport>/ClassicCallbackSample1/src-gen/IPD_ClassicCallInbound_Entry.scxml`

3 Then, create the following parameters in the ApplicationParms section:

```
app_find_agent_timeout = <timeout in seconds when routing to agent if GMS Callback fails>
app_selected_agent_group = <agent_group to route to if GMS Callback fails>
app_gms_offer_callback = true
app_gms_service_uri = http://<gmshost>:<gmsport>/genesys/1/service/<callback_service_name>
```

4. Save your script object.

### Important

The `<callback_service_name>` parameter must match the name of the Callback service that you will **add** to the Service Admin UI.

## Create DNs for your Classic Callback Inbound Script Resource

The screenshot shows the GAX Configuration page for a Routing Point. The breadcrumb trail is: Home > DNs > Switches > SIP\_Switch > DN > Routing Point > 8214 Properties. The 'Options' tab is selected, showing a table with the following data:

Key	Value
Orchestration	
application	script:GMSClassicInbound

At the bottom of the page, there are buttons for 'Cancel', 'Apply', and 'Save'.

Browse or create DNs of type Routing Point in **Switching > DNs > Switches > SIP\_Switch > DN > Routing Point**. Check the **Options** tab. The Orchestration.application value must be set to script:<gms-classic-sample-script> where gms-classic-sample-script is the name of the classic script that you have just created. In our example, Orchestration.application=script:GMSClassicInbound.