GENESYS™

Callback User's Guide

Genesys Callback 8.5.1

12/29/2021

Table of Contents

Callback Solution Guide

Welcome to the Callback Solution Guide! You can use this guide as a starting point to configure Callback services.

Find information to help plan Callback

- Solution Overview: Features, Use Cases, Scenarios
- Architecture: Components, Key Advantages
- Prerequisites:
  - Hardware, OS, Browser, Java, Genesys Requirements
  - Integration with GVP or Media Server

Find instructions to configure Callback

Once you have set up the Prerequisites, find procedures to configure Genesys Callback.

- Strategy for queuing
- Accept inbound calls
- Enable outbound calls
- Enable Call Progress Detection
- Enable Callback UI for Administrators

Provision your Callback Services

Find procedures to create and configure the Callback services required to implement scenarios.

- Access UIs
- Add a Callback Service
- Manage Callbacks
- Set up Play Treatments
- Set up Office Hours
- Define Capacity
- Add Exception Patterns

- Enable Status Notifications for Callback

> **Important**
>
> Provisioning should be performed through the GMS Service Management UI. Read the complete help here.

Learn about scenarios

Find more details about the available scenarios and Service Configuration

- User Terminated Immediate
- User Terminated Delayed
- User Terminated Scheduled
- User Terminated Delayed Agent Preview

Learn about options

- Callback Service Options Reference

Learn about Callback and Service APIs

Once your Callback Service is ready, you can start implementing your application.

APIs related to Callback Services, builtin services, and ORS scenarios are detailed in GMS API References.

- Callback Services API
- Calendar API
- Capacity API

For custom samples, see:

- ClassicCallbackSample illustrates how to implement an IVR (Genesys Voice Platform VoiceXML) application that communicates with GMS and performs classic Callback scenarios.
- Custom Callback Sample implements an On-Dial plugin to interface with the GMS Callback service. Developers should use this sample as a reference to build a Composer application that is invoked as a plugin from GMS Callback.

The Service Management UI also includes a Sample panel to test your Callback Services.

Solution Overview

Callback is built on top of the Genesys Mobile Services (GMS) platform, which enables mobile and other applications to use Genesys callback capabilities from their applications. Integration with the contact center callback functionality is accomplished through APIs that can be accessed through the internet and deployed into your network infrastructure.

Scope of Use

Typical usage scenarios of callback include:

- Schedule, immediate, or delayed callback requests.
- Preview callback requests.
- Proactive notification.
- Schedule callback with enhancement multimedia confirmation.
- Schedule an immediate return call or a callback at a convenient time of the customer's choosing, based on operating business hours.
- Check and display agent availability by providing estimated wait times.
- Support for mobile push notification to provide an alert when agent is available.

In addition, Callback includes:

- Samples to integrate with Genesys Web Desktop.
- APIs to integrate with your own environment.
- An intuitive interface to simplify management and reporting of callbacks.
- RESTful APIs, optimized for mobile, web, and IVR.

Use Case: IVR Callback

Requirement: If your contact center is busy, use IVR to offer an option for Callback while retaining the caller's position in queue, in order to free up valuable IVR resources and optimize contact center resources.

Callback Solution

- Use GME Stats API to check the Estimated Wait Time for the caller and determine if the caller should be offered a callback option.
- Use Callback to offer the caller an option to hang up and call back when an agent is available, while retaining the caller's position in queue.
- The call is virtually queued and called back when an agent is available

> **Tip**
>
> Callbacks can also be scheduled at the caller's convenience.

Use Case: Credit Card Use Verification

Scenario: When a credit card is used abroad without a previous arrangement, the bank must verify that the user is authorized to accept or reject the transaction.

Solution:

- Send native Push notification to a registered banking app on the customer's mobile device.
- Offer Click-To-Connect to an agent with contextual information, such as the credit card number and the user location.
- Offer help via callback from an agent.

> **Tip**
>
> You can also offer to contact the user with backup methods, such as email and SMS.

Use Case: Vehicle Roadside Assistance

Scenario: Scenario: An insurance company offers a Vehicle Roadside Assistance service. When the car breaks down, the driver uses the mobile app to get assistance.

Solution:

- Offer Connect Me to deliver the driver's location and account information to the contact center that can forward the information to a towing company.
- Offer callback when all agents are busy.
- Send native Push notification to provide status of assistance. For example, tow truck is dispatched with ETA

Supported Callbacks

Callback includes pre-defined scenarios that you can easily implement through the Callback

There are two types of scenarios for Callback:

- Immediate: callback happens as soon as an agent is available.
- Scheduled: callback occurs at an agreed to time in the future.

In the graphic shown here, the app offers the customer a scheduled callback.

For instance, the following basic scenarios are available:

Call flow for User Terminated Immediate
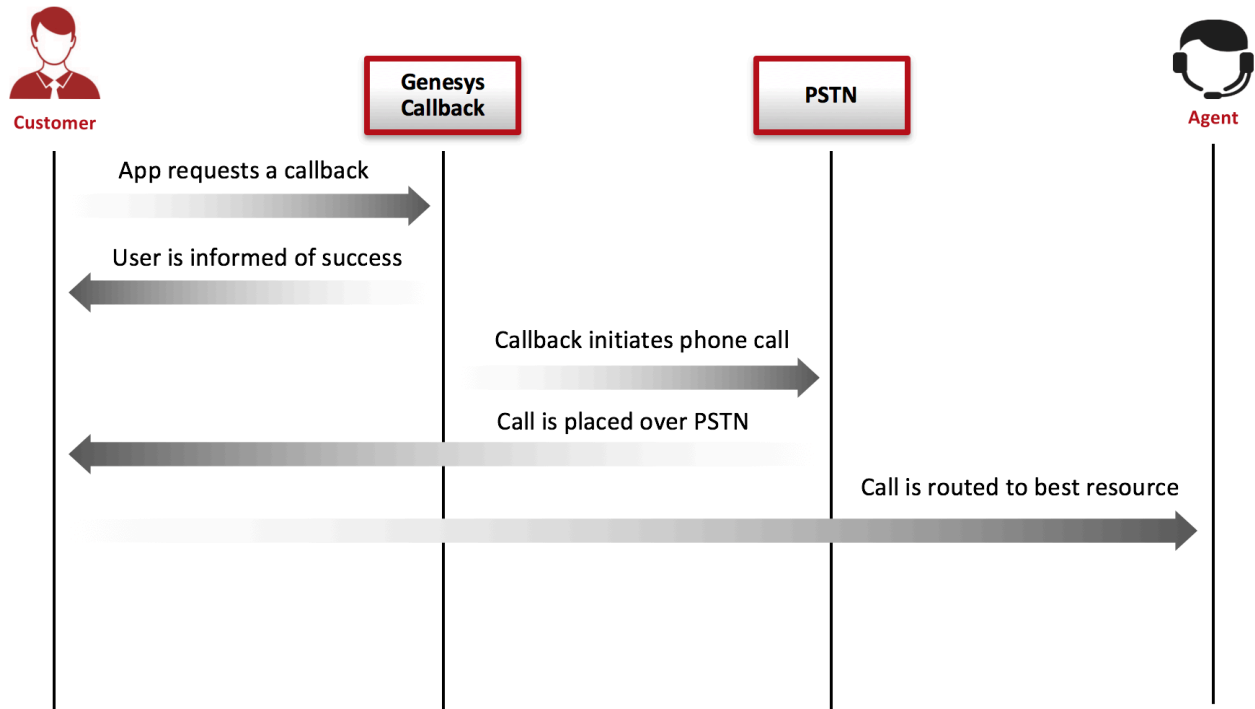
In this scenario, the customer requests an immediate callback, that is, as soon as the agent i available.

Start Callback

[+] Start Callback

- Callback service: Sends a message to expect a call, to the mobile device immediately.

- Callback service: Calls the mobile device.

- Mobile device: Accepts the call.

- Callback service: Identifies that a human has answered the call.
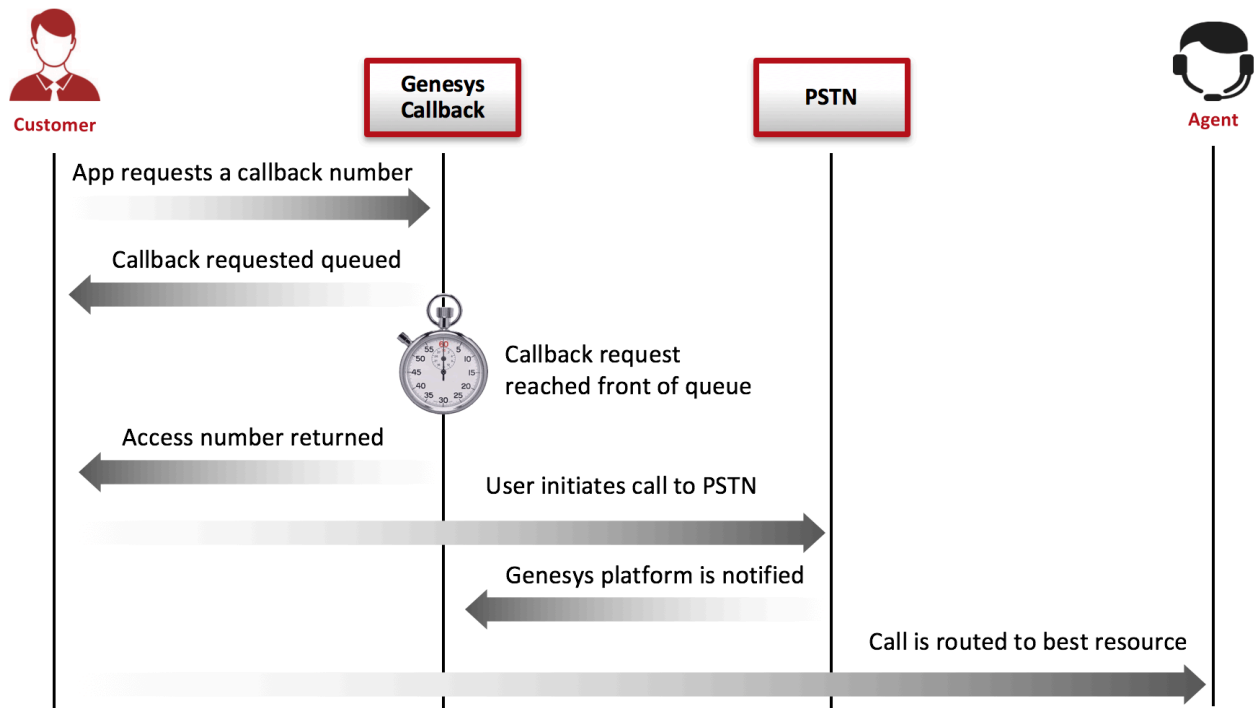
Dial Access Number

[+] Connect Agent

Connect to Agent

- Callback service: Plays treatment until target is available.

- Callback service: Reserves target to route call.

- Callback service: Routes the call to the target agent.

- Callback service terminates.

Call flow for User Terminated Delayed

In this scenario, the customer schedules a callback.

Start Callback

[+] Start Callback
*Callback service: Returns session ID to the user.

- Callback service: Waits for an agent to be available.

- Callback service: Notifies mobile device when an agent is available.

- Next: Mobile device is expected to send a connect request to confirm the user's availability.

Connect

[+] Connect

- Callback service: Immediately returns access information to the mobile device.

- Callback service: Waits for the voice call to arrive.

- Next: Mobile device is expected to dial the access number.

Dial Access Number

[+] Dial Access Number

- Inbound service: Locates the GMS service associated with the arrived voice call.

- Inbound service: Delegates the call to be processed by the Callback service.

- Callback service: Reserves the target to route the call.

- Callback service: Routes the call to the target.

- Callback service terminates.

Architecture

Callback is deployed on top of the Genesys Mobile Engagement Platform that allows mobile and other applications, for example, IVR and Web, to connect and use Genesys Callback. The Callback complexity of integrating Callback with the contact center is hidden and embedded through simple REST APIs that are:

- Optimized for the Mobile and Internet Data network
- Easily deployed in existing customer network infrastructures

Components
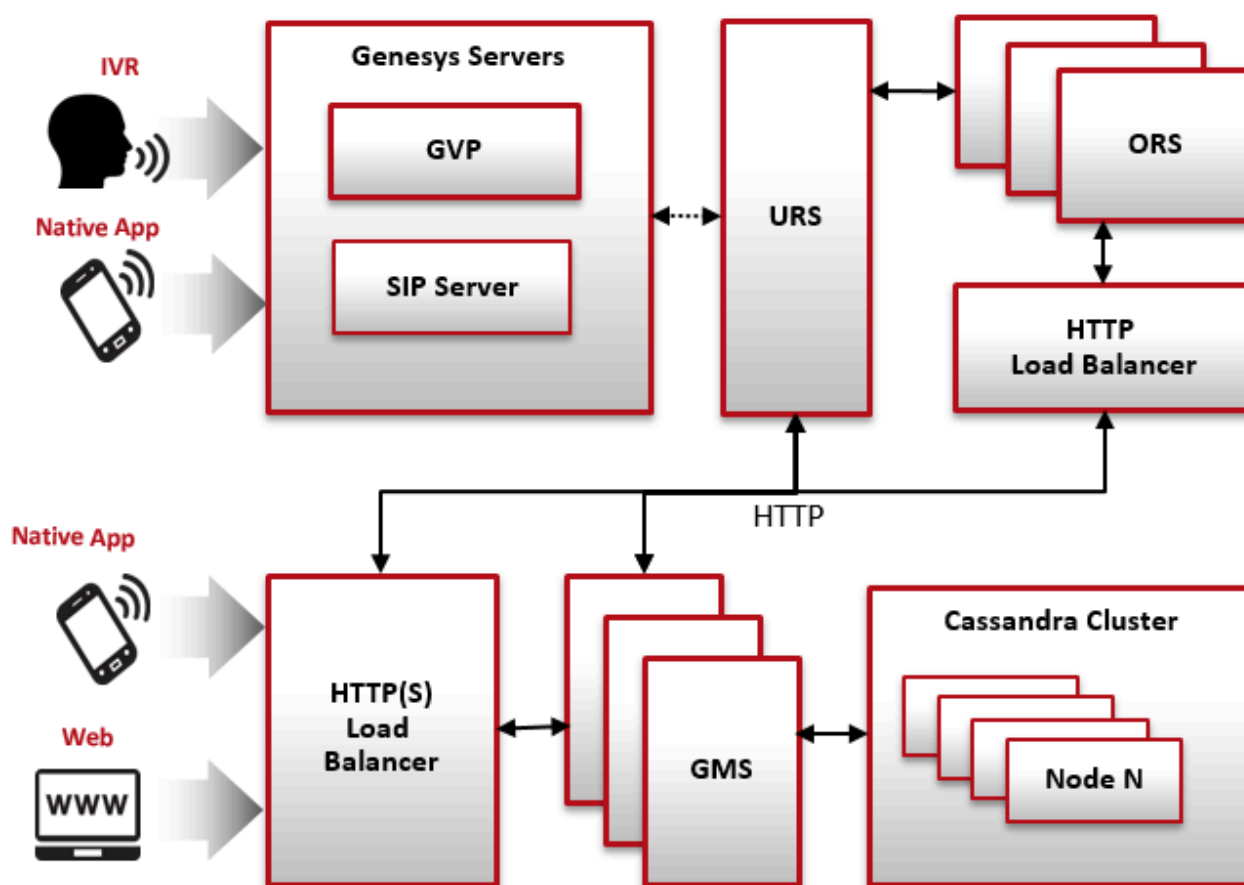
The main components for Callback are:

- Genesys Mobile Services (GMS) – This component controls and exposes the Genesys API functi
  external applications by REST APIs, and provides critical callback services (for example, callba
  management). GMS provides a user interface to manage the provisioning and deployment of c
  services. This same user interface also provides basic callback service monitoring capability.

- Orchestration Server (ORS) – This component provides various callback communication proces
  callback inbound treatments, and matching services.

- Genesys Voice Platform - This component provides custom treatments through SCXML or Voice
  plugins as configurable options for the Callback Service template. If you don't need IVR, you ca
  replace it with a Media Server.

- SIP Server - This component provides play treatments, Call Progress Detection (CPD), and outb
  management.

- Universal Routing Server – This component routes interactions.

Also consider reading the Universal Routing Voice Call Back White Paper that details two types of
Voice Call Back (VCB) solutions managed by Universal Routing Server (URS):

- Pre-emptive Agent Reservation
- Dialing Notifications

Key Advantages

- A multi-channel, integrated solution that provides customer service access to context, such as customer
  profile, history, and location
- Tightly integrated with Genesys routing and does not disrupt queuing
- Optimized REST and Ajax Push interfaces for mobile, web, and IVR
- Flexible connection for adding callback anywhere in the interaction: click-to-connect voice, push
  notification, chat, delayed or immediate
- Native Estimated Wait Time, based on URS/Stats Server

Prerequisites

Modified in 8.5.114

Callback is provided through Genesys Mobile Environment components and requires that your system meets the software requirements established in the Genesys Supported Operating Environment Reference Manual for Genesys Mobile Services (GMS), as well as meeting the following minimum requirements:

Hardware Requirements

The following are minimum requirements:

- CPU: Quad Core
- Memory: 4GB
- Disk: 160GB
- At least 2-3 nodes recommended for redundancy and availability

OS Requirements

- Genesys Supported Operating Environment Reference Guide

> ### Important
> For Linux installations, the Linux compatibility packages must be installed prior to installing the Genesys IPs.

Browser Support

- Genesys Supported Operating Environment Reference Guide

Java Requirements

- Before 8.5.103, Callback requires JDK 7.

- Starting in 8.5.103, Callback requires JDK 8.

> ### Tip
> Edit JAVA_HOME to point to the JDK installation folder, for example,
> C:\Program Files\Java\<your JDK>.

Cassandra Support

- Cassandra 2.x: Tested version is 2.2.9

Genesys Environment

In addition to having a Genesys Management Framework 8.1 environment installed and running, the following table lists the Genesys components that are required for a Callback installation.

Genesys Component	Minimum Version Required	Comments
Genesys Mobile Services (GMS)	8.5.114.09	Installed and running: you must use the Service Administrator UI to manage Callback services.
Orchestration Server (ORS)	8.1.400.26	Optional, installed and running: • An HTTP port must be enabled in the related Application object. • The ORS server must use the Orchestration Server type in Configuration Manager.
Universal Routing Server (URS)	8.1.400.39	Mandatory. Required for GMS and URS-based dialing in Callback applications.
Interaction Routing Designer (IRD)	8.1.400.26	Mandatory, required for strategies running on URS.
SIP Server	8.1.100.67	• SIP or Inbound Voice is required for agents. • SIP Server is recommended for outbound calling for Callback.

Genesys Component	Minimum Version Required	Comments
Stat Server	8.x	Required to obtain statistics.
Media Server	8.1.410.33	Required to play treatments and use Call Progress Detection (CPD) for outbound calls.
Resource Manager	8.1.410.13	Required to play treatments and use Call Progress Detection (CPD) for outbound calls.
Workspace Desktop Edition	**(optional)** 8.5.111.21	Support for Genesys Callback. This component is not mandatory.

Integrating with GVP or Media Server

Genesys Voice Platform (GVP) provides VoiceXML features in addition to the Media Server features. You need GVP only if you need VoiceXML. In any other cases, Media Server is sufficient and will provide better performances without GVP.

See GVP Documentation for further details.

Configure Callback

> **Important**
>
> Before you start configuring Callback, make sure that you setup and installed your Genesys Environment.

Configuring a Callback service involves:

- Importing a URS strategy.
- Setting up inbound and outbound calls.
- Setting Roles for the Callback UI.

Then, you will be able to create a Callback service in the Callback UI, associated with one of the Callback scenarios detailed in the Scenarios section.

Set up URS Strategy for Queuing

Enable HTTP Interfaces in URS

Create a Listening HTTP Port in URS

URS will listen on this port for incoming HTTP requests. Basically, this steps turns URS into an HTTP server.
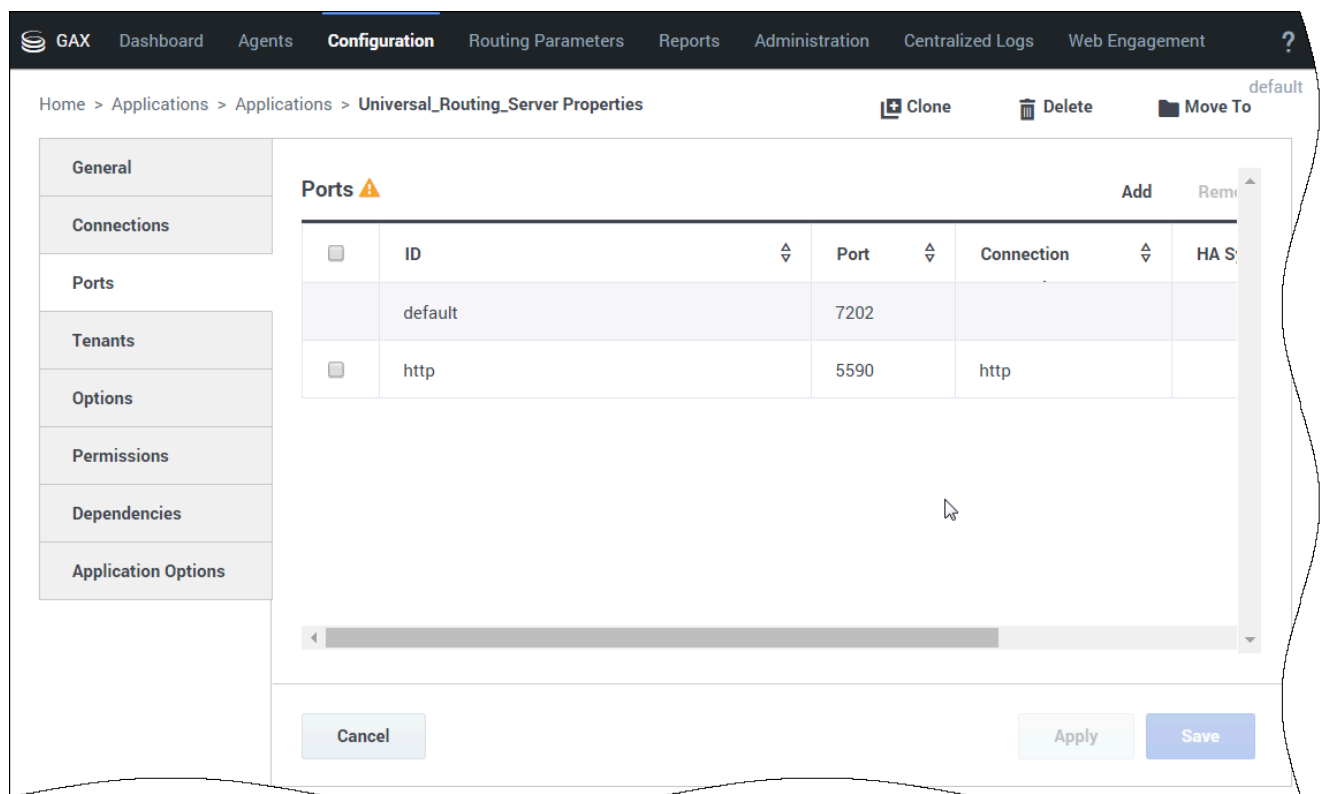
In Genesys Administrator Extension, edit your URS application.

Add an HTTP listening port with a port ID `http` in the **Ports** tab. Make a note of this port number as you will need it later when configuring GMS and ORS-based services.
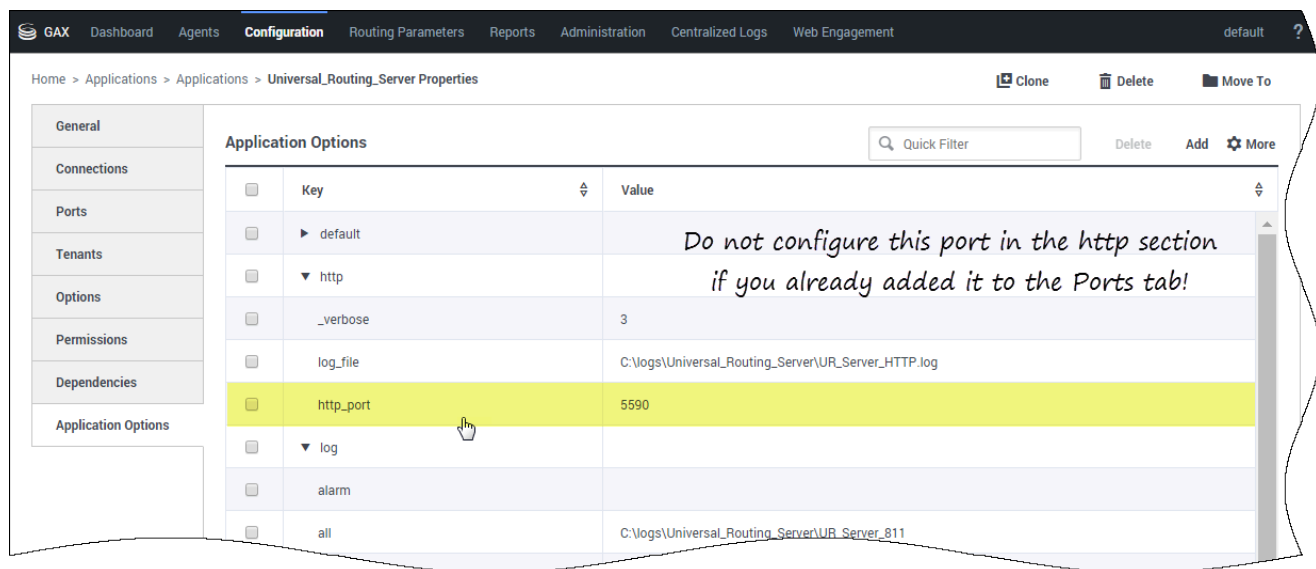
You can also do this by creating the `http_port` option in the `http` section of your **Application Options** tab.
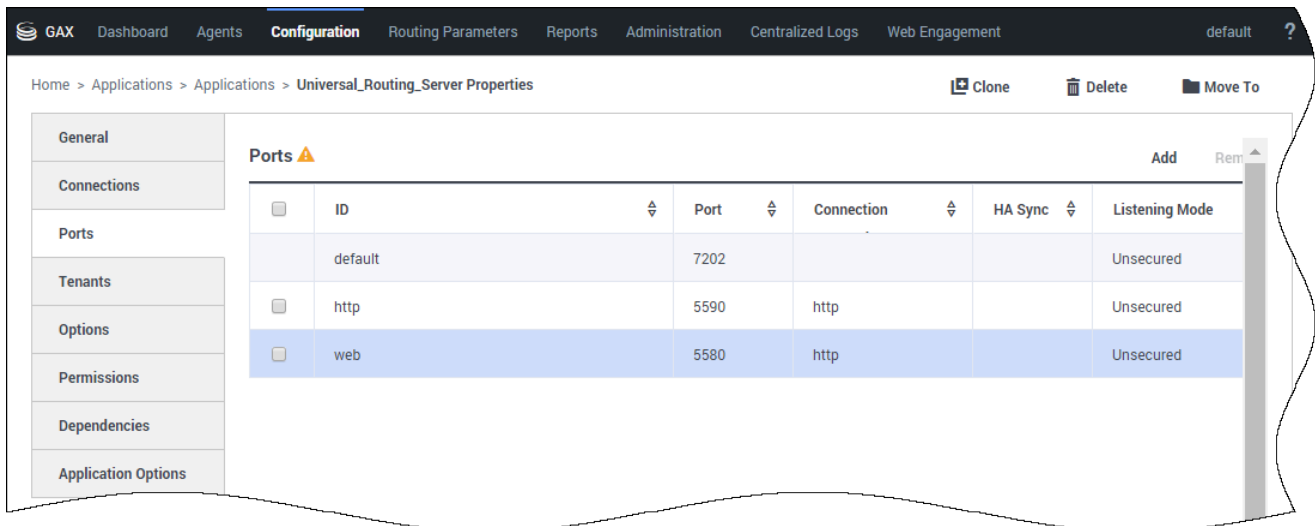

Warning

This HTTP port needs to be created in one place only.

Enable Web HTTP Replies in URS

URS uses the `httpbridge` module to send target information back to GMS. To make this possible, create a web HTTP port that will be used to reply. URS will be able to perform external HTTP requests, for example, to submit `timetodial` events to GMS, and so on.

Important

The listening **http** port created in the previous section and the **web** port defined below MUST have different values.

In Genesys Administrator Extension, edit your URS application. Add an HTTP port with a port ID web in the **Ports** tab.

You can also do this by creating the `http_port` option in the **web** section of your **Application Options** tab.

- `http_port` = 5580 (or some other port, used internally)

Warning
This HTTP port needs to be created in one place only.

Configure URS Delay Strategies

You must deploy URS delay strategies. This step is required because when a service request is received by GMS, the request is sent to ORS for execution. ORS then sends a request to URS to create a virtual interaction and to place it in the specified virtual queue. When an agent is available, URS sends an asynchronous response containing the selected target information to GMS, via a URL specified at the time of creation of the virtual interaction. For samples, you will create a new virtual queue in which to place the interactions, however, for a real-world scenario, the virtual queue must be selected appropriately.

To deploy URS delay strategies, open Genesys Administrator Extension.

Create a dedicated Virtual Queue

Navigate to '*Switching > DNs > Switches > SIP_Switch > DN > Virtual Queue* and create a virtual queue GMS_VQ. Save and configure the alias GMS_VQ_SIP_Switch. You will need this alias when you

will configure the `_urs_virtual_queue` option of your Callback service.

Enable ORS to pull interactions and URS to receive routing requests.
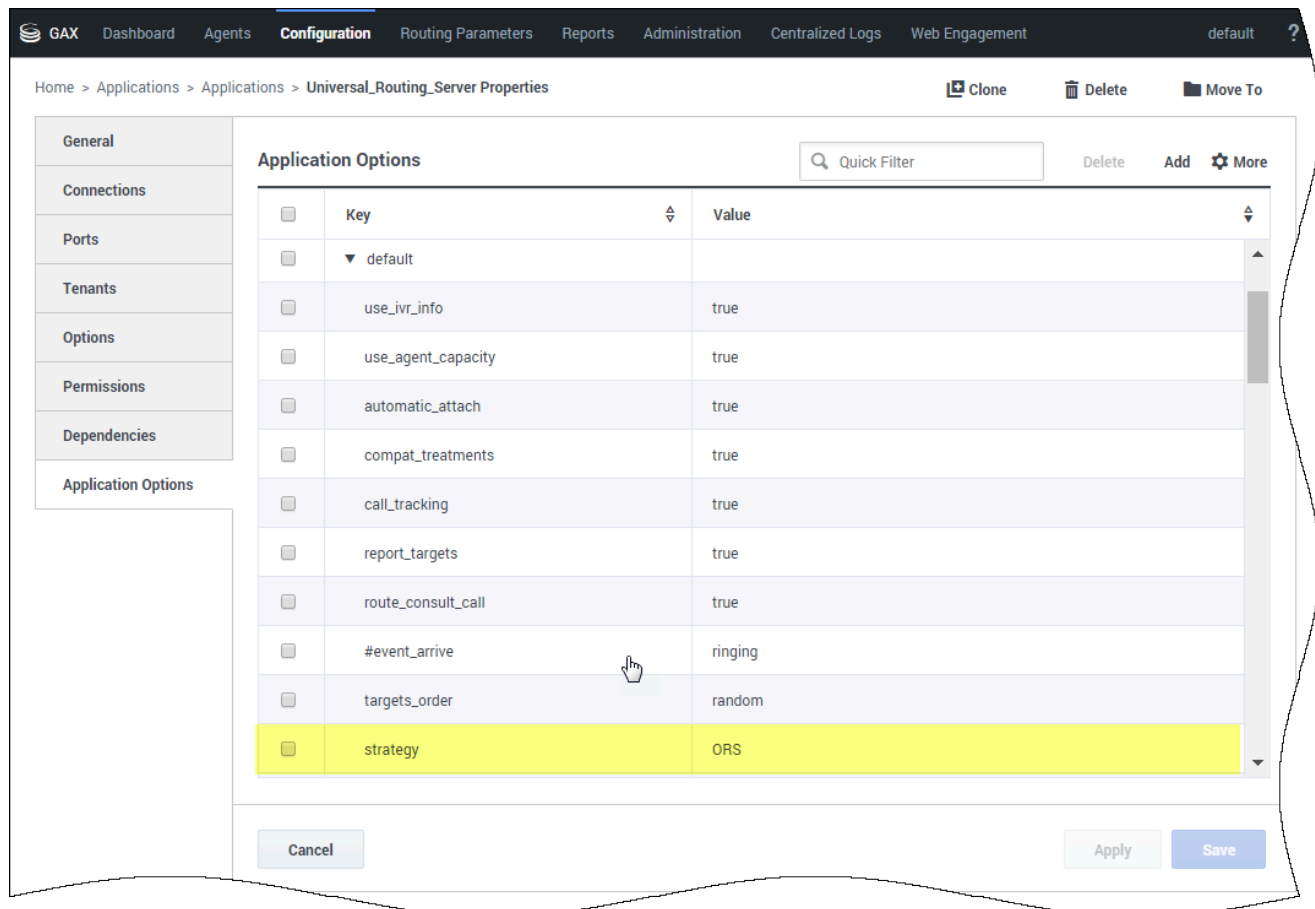
Edit the Universal Routing Server Application and select the **Options** tab. Enable ORS to pull interactions by setting the `strategy` option to `ORS` in the `default` section.

Now, you can download the URS Strategies and import them into IRD. See the download section below to manage the downloadable files.

Configure Multiple Targets

The option _target in the URS Queuing section of your callback service allows you to configure one or

more URS targets (as detailed in the option's description).

To configure multiple targets, you must create a JSON array of targets (maximum 5) and for each target, you provide a statistic condition that the system will check. This condition specifies when to switch to the next target. If the condition is not matched, the interaction will be queued. Otherwise, the system will test the condition of the next target, as detailed in the diagram below.

The condition is defined by the parameters stat_to_check, stat_operator (< or >) and stat_value.

For example, if you set:

```
{
        "target": "GMS_AG_Kilfoil@Stat_Server.GA",
        "timeout": "15",
        "clear": false,
        "stat_to_check": "StatAgentsAvailable",
        "stat_operator": "<",
        "stat_value": "1"
}
```

- The system will change to the next target in the list if StatAgentsAvailable < 1 (no agent is available).

- The system will queue the interaction to the target GMS_AG_Kilfoil@Stat_Server.GA only if StatAgentsAvailable >= 1 (which means that at least one agent is available). If the duration specified in the target timeout has passed, and if the agent is not selected, then the statistic condition for the next target is checked prior to queuing the interaction for the next target. You can also expand to the next target if clear = true.

> ## Important
>
> If you do not set a condition by using `stat_to_check`, `stat_operator`, and `stat_value`, then the tested condition is set by default to `no logged in agents`. That means that the interaction is queued to a target only if at least one agent belonging to the target is logged in.

The `stat_to_check` property can be set to any of the values supported by the Statistics parameter passed to the IRD function `SData(Target, Statistics)`, unless `target` is a skill expression. If `target` is a skill expression, you must choose one of the following values:

- RStatAgentsReadyvoice—agents ready for voice media.

- RStatAgentsReady—agents ready for any media.

- RStatAgentsTotal—agents logged in.

The JSON code below is a sample of value for the _target option and includes 5 different targets.

```
[
    {
        "target": "GMS_AG_Kilfoil@Stat_Server.GA",
        "timeout": "15",
        "clear": false,
        "stat_to_check": "StatAgentsAvailable",
        "stat_operator": "<",
        "stat_value": "1"
    },
    {
        "target": "GMS_AG_Milburn@Stat_Server.GA",
        "timeout": "10",
        "clear": false,
        "stat_to_check": "StatAgentsAvailable",
        "stat_operator": "<",
        "stat_value": "1"
    },
    {
        "target": "GMS_AG_Monique@Stat_Server.GA",
        "timeout": "30",
        "clear": false,
        "stat_to_check": "StatAgentsAvailable",
        "stat_operator": "<",
        "stat_value": "1"
    },
    {
        "target": "GMS_AG_Oladipo@Stat_Server.GA",
        "timeout": "15",
        "clear": false,
        "stat_to_check": "StatAgentsAvailable",
        "stat_operator": "<",
        "stat_value": "1"
    },
    {
        "target": "GMS_AG_Sippola@Stat_Server.GA",
        "timeout": "15",
        "clear": false,
        "stat_to_check": "StatAgentsAvailable",
        "stat_operator": "<",
```

```
      "stat_value": "1"
   }
]
```

URS Strategies for Download

> **Important**
>
> When you upgrade GMS, you need to import the Callback Template from your GMS installation directory. Start the Service Management UI, upload the <GMS Installation Directory>/service_templates/callback.zip file, and restart ORS.

GMS version	ZIP	Instructions
8.5.004.xx and earlier	GMS_URS_Strategy.zip	1. Download and unzip the zip file containing the URS strategies.
between 8.5.005.xx and 8.5.101.10	GMS_URS_Strategy_85005.zip	
between 8.5.101.10 and 8.5.107.19	GMS_URS_Strategy_85010_v2.3.zip	2. Open Interaction Routing Designer (IRD).
8.5.108.02 and higher	GMS_URS_Strategy_85108_v2.4.zip	3. Import the strategy **WaitForTarget.zcf,** and subroutine **SetRouteDelay.zcf,** using *File > Import From File* on the respective tabs.
8.5.109.08 and higher	GMS_URS_Strategy_85109_v2.58.zip	
8.5.114.09 and higher	GMS_URS_Strategy_85114_v2.63.1.zip GMS_URS_Strategy_852000_v2.64.1.zip GMS_URS_Strategy_85200_v2.66.zip	
8.5.2 and higher	See URS Strategies in 8.5.2	4. Open the strategy and subroutine. 5. Compile and save.

- Starting in 8.5.109.08, the URS Dial Success Rate is set to 85% when new callbacks are created to improve the callback performance.

- Starting in 2.64.1, the DialOutSuccessRate function of the WaitForTarget strategy is no longer invoked to allow the enhanced VCB algorithm within Universal Routing Server to work properly. If your application requires the legacy VCB algorithm to work, change the strategy to invoke the DialOutSuccessRate function as in earlier versions of the strategy.

> **Important**
>
> You do not need to load the strategy in ORS because ORS will request it when needed. See the Interaction Routing Designer help file for information about using IRD.

Additional Deployment Steps for GMS 8.5.005.xx and Higher

Prerequisites:

If you are upgrading Callback from GMS 8.5.004.xx and earlier, make sure that you have the following components and versions installed:

- Interaction Routing Designer (IRD) 8.1.400.26
- Universal Routing Server (URS) 8.1.400.39

Deployment Changes

1. Uninstall your existing IRD.
2. Install IRD 8.1.400.26.
3. Delete the existing `WaitForTarget` strategy.
4. Delete the existing SetRouteDelay subroutine.
5. Download the strategies for GMS versions 8.5.114.09 and higher.
6. Import into IRD.
7. Compile both strategy and subroutine.
8. Uninstall your existing URS.
9. Install URS 8.1.400.39.

> ### Important
> Do not skip step 7. This step is mandatory to ensure that the strategy and subroutine are properly saved and loaded into IRD.

Accept Inbound Calls

For some voice scenarios, you must create a 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.

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

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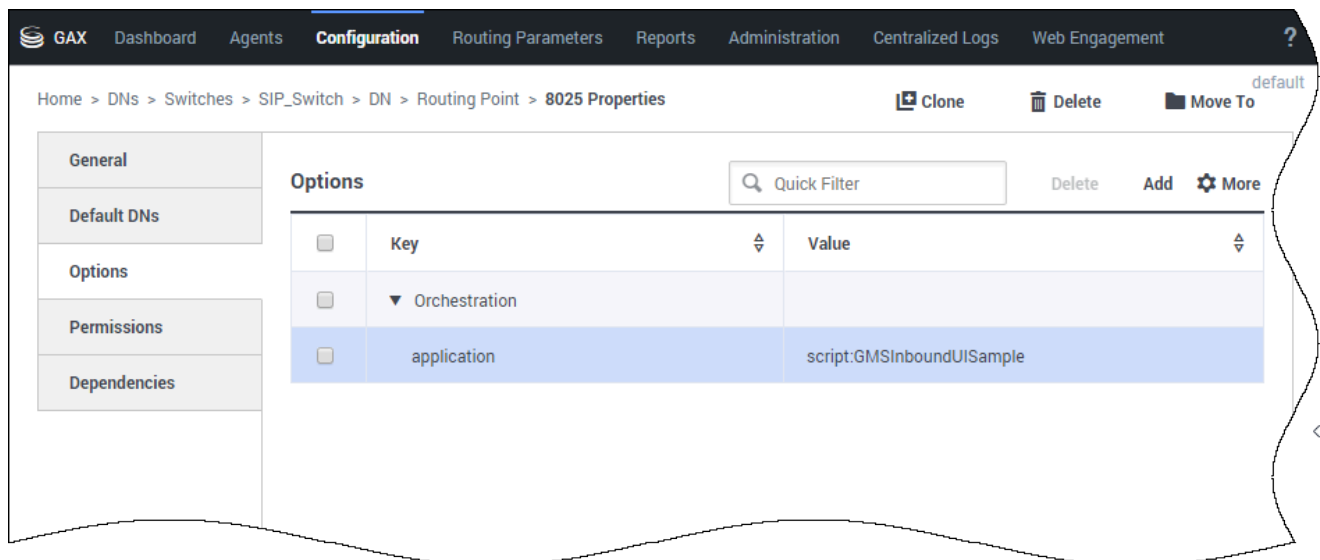
Browse or create DNs of type Routing Point in **Switching > DNs > Switches > SIP_Switch > DN > Routing Point'**. *Check the* Options tab. The Orchestration.application value must be set to script:<gms-inbound-sample-script> where gms-inbound-sample-script is the name of the inbound script that you have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

Setting up Acceptance of a Classic Inbound Call

What are the Difference 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.scxml 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

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`.
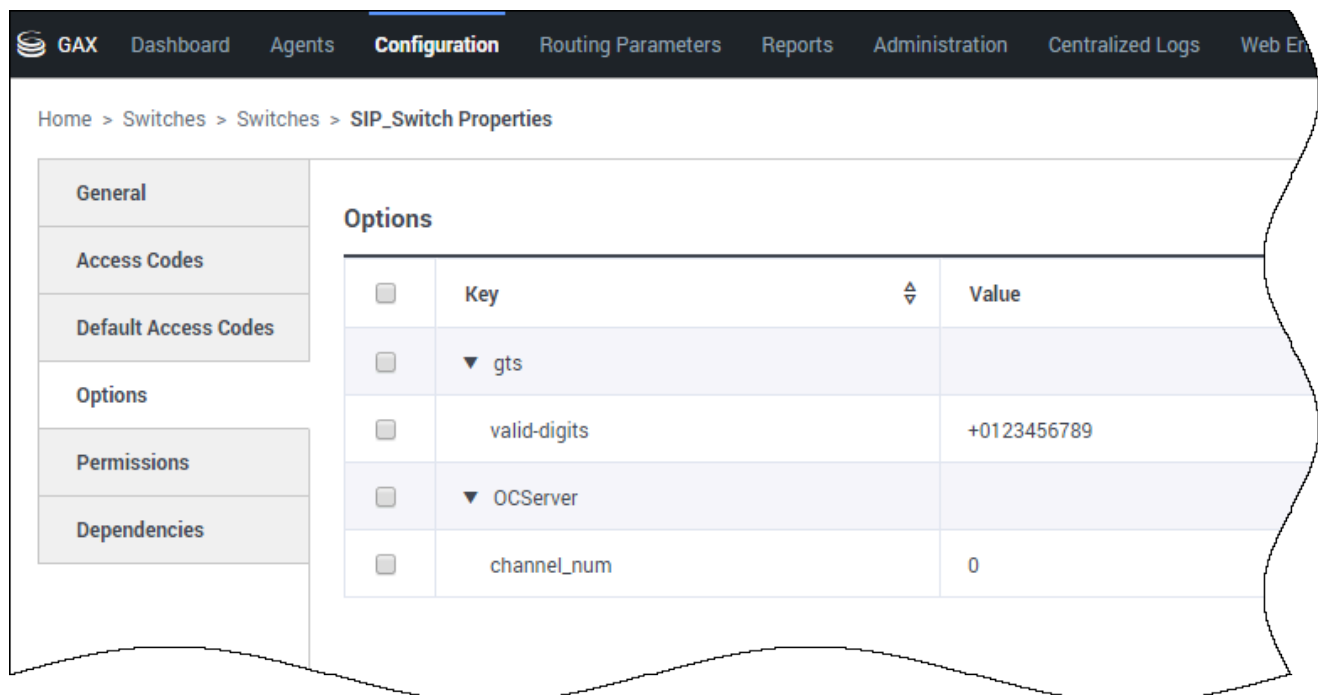
Enable Outbound Calls

Modified in 8.5.108.02

This configuration is required for voice scenarios. See the Genesys Voice Platform Deployment Guide for additional details.

Callback uses Media Server via SIP Server to make outbound calls. SIP Server communicates with Media Server using MSML and requires the following configuration to enable outbound calls.

Set Valid Digits (Optional)

Valid customer numbers should include a + sign if needed. If true, edit the `valid-digits` option in the **gts** section of your SIP Switch object:

```
[gts] valid-digits = +0123456789
```

Refer to ORS documentation for further details.

Set the Prefix Dial Out Option

To make sure that the system will be able to call, configure the _prefix_dial_out option in your callback service with the Service Management UI.

Set the Default Country Option

By default starting in 8.5.108, callbacks for unreachable phone numbers and premium numbers are disabled (see _disallow_impossible_phone_numbers and _disallow_premium_phone_numbers options). Therefore, you must configure the _default_country option in your Callback service.

- Phone numbers are tested against Google's library for parsing, formatting, and validating international phone numbers.

- GMS 8.5.108.02 integrates version 7.2.8 and uses the Apache License Version 2.0.

- The list of premium numbers is available in Wikipedia.

How to Configure the MSML Service

Open Genesys Administrator:

- Navigate to **Switching > Switches > SIP_Switch > DN > VOIP Service** and edit the MSML_Service object.

- Make sure that the following options are configured for **MSML_Service** to enable outbound:

```
make-call-rfc3725-flow=1
refer-enabled=false
ring-tone-on-make-call=false
userdata-map-filter=*
```

Create a Routing Point DN Dedicated to Outbound Calls

Navigate to **Switching > Switches > SIP_Switch > DN > Routing Point** and create a **Routing Point** object with, for instance, **name** set to 8999 and **alias** set to 8999_SIP_Switch.

Then, use this DN to set the option _route_point in your Callback service. For example, _route_point = 8999_SIP_Switch.

How to Configure Calls Placed from Agent DNs

Added in 8.5.108.02

Outbound calls will be placed from agent DNs if you configure the following options in your callback service:

```
_userterminated_first_connect_party=AGENT
_agent_preview_via_rp=false
_agent_first_via_rp=false
```

Additionally, for agents involved in this callback scenario, set the following configuration in each agent DN Annex:

```
section TServer
refer-enabled=false
make-call-rfc3725-flow=1
```

Advanced Settings for Agents on External Switch

For a user-terminated callback with option `_userterminated_first_connect_party` set to CUSTOMER, the outbound call will be placed from the route point specified by option _route_point. This route point must be on a **SIP Server** type switch.

If the agents are on a different switch, you must set the callback service option `_ixn_redirect_confirm` to `false`. This is due to a limitation in how the routing to agent operates. In this scenario, a new call is created, the callback SCXML will not receive the events for this call and will not be able to confirm that the agent answered.

To handle the case where the agent does not answer, you can set the option `_ixn_redirect_hints` option to enable particular handling by the other switch. For example, you can set the following value for a Cisco switch.

```
_ixn_redirect_hints = {"extensions" : {"NO_ANSWER_TIMEOUT" : "5","NO_ANSWER_ACTION" :
"notready","NO_ANSWER_OVERFLOW" : "some DN"}}
```

This configuration enables a **no answer** timeout of 5 seconds, sets the agent to Not Ready Upon No Answer, and, upon no answer, routes the call to the DN specified.

Call Progress Detection (CPD)

You must enable the answering machine connection for user-terminated scenarios with Call Progress Detection (CPD) capability. This feature is tied to SIP Server and MSML Service.

Enable CPD in your SIP Server
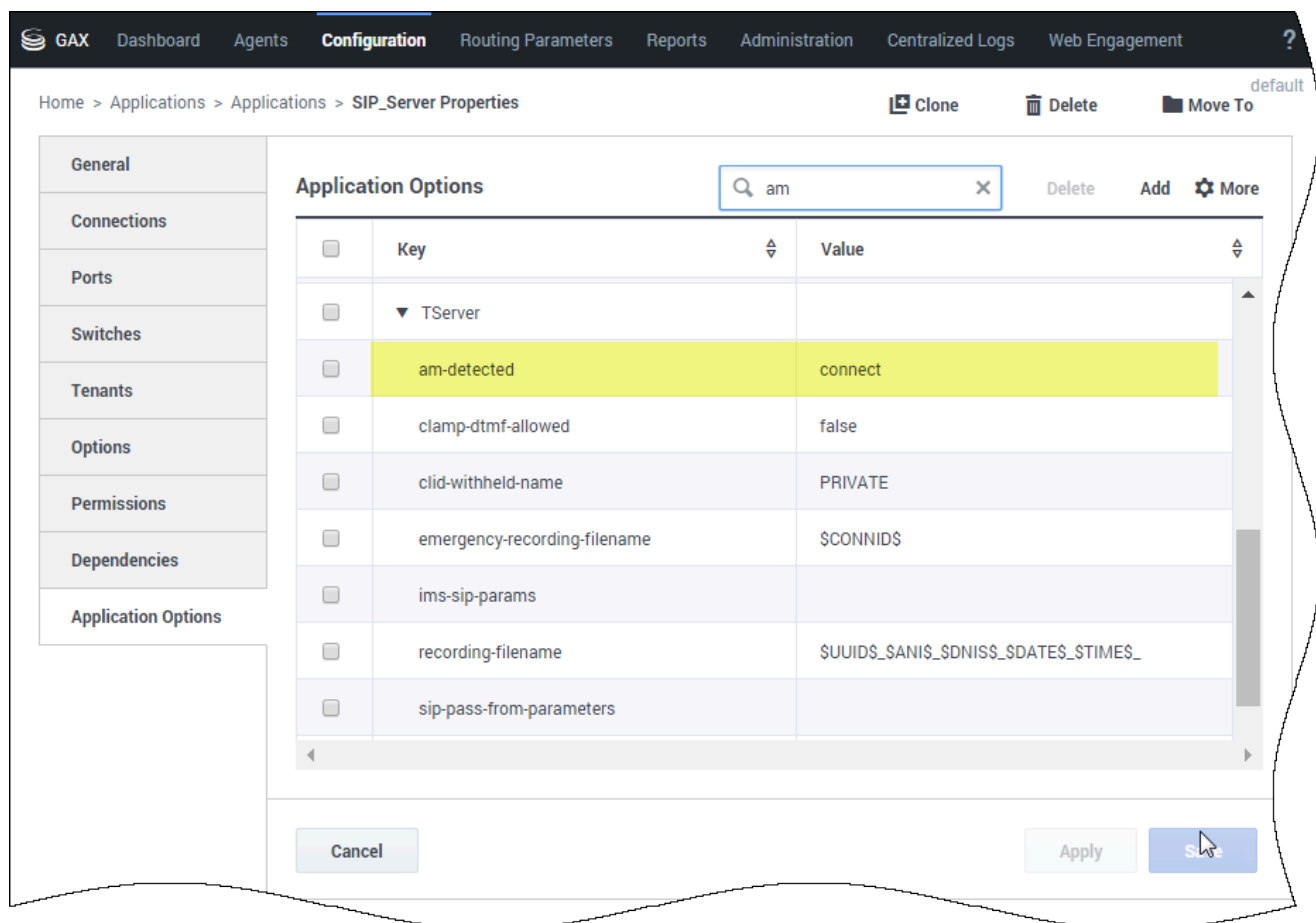
1. Edit your SIP Server Application, and set `TServer/am-detected = connect`.

2. Enable MSML, which is required so SIP Server can communicate with GVP as a Media Server to delegate outbound calls, play treatments, and CPD.

   - `TServer/msml-support=true`
   - `TServer/refer-enable=true`

Enable Callback UI for Administrators

The Callback UI enables you to create services based on the provided Scenarios and is embedded in the Service Management UI of Genesys Mobile Environment. To access the Callback UI, make sure that GMS is started and that your user owns `Supervisor` or `Administrator` permissions.

Setting permissions to Callback Administrators
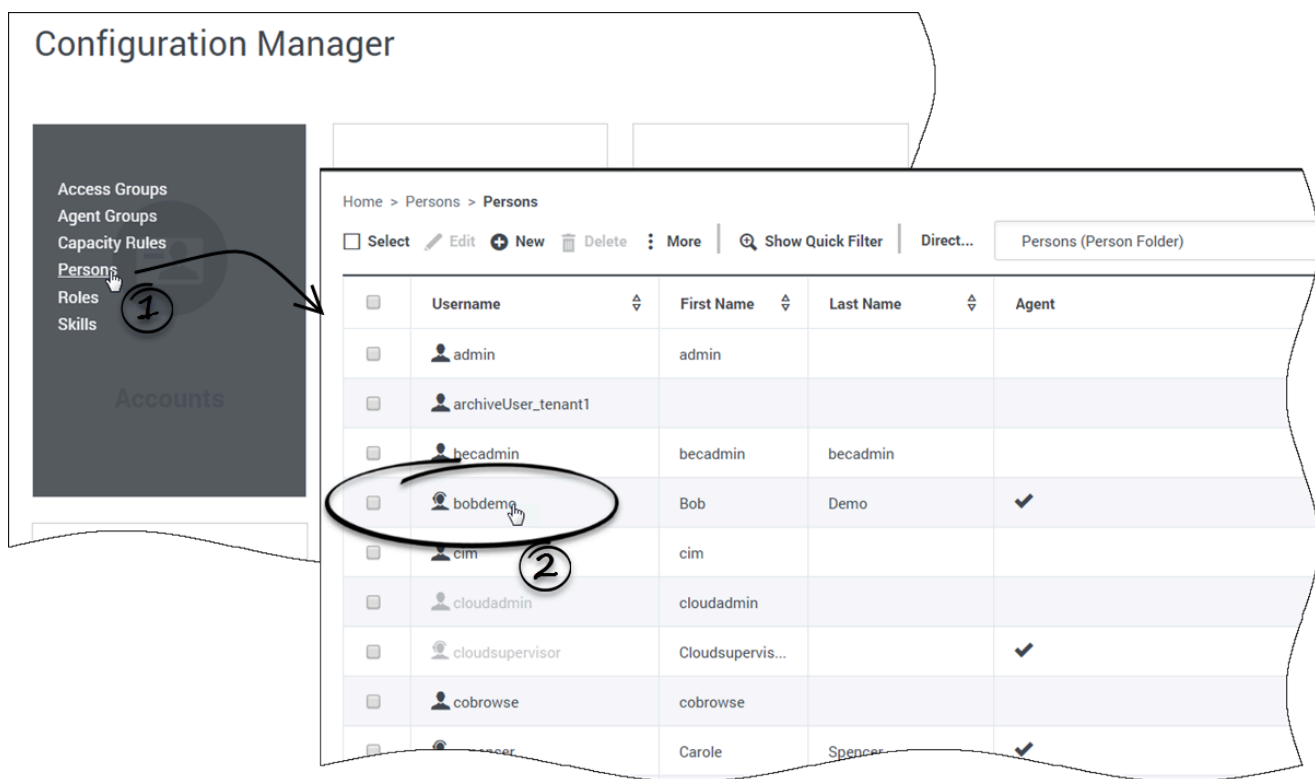
Open GAX and select the **Configuration Manager** section. In the **Accounts** menu, select **Persons** to get the list of configured users.

Edit the person who will be logging into the Service Management UI. You are going to give this user the permissions to read/write data into Callback related configuration objects (for example, GMS Application, Business Attributes, Transaction Lists for Resources/Patterns, and so on).
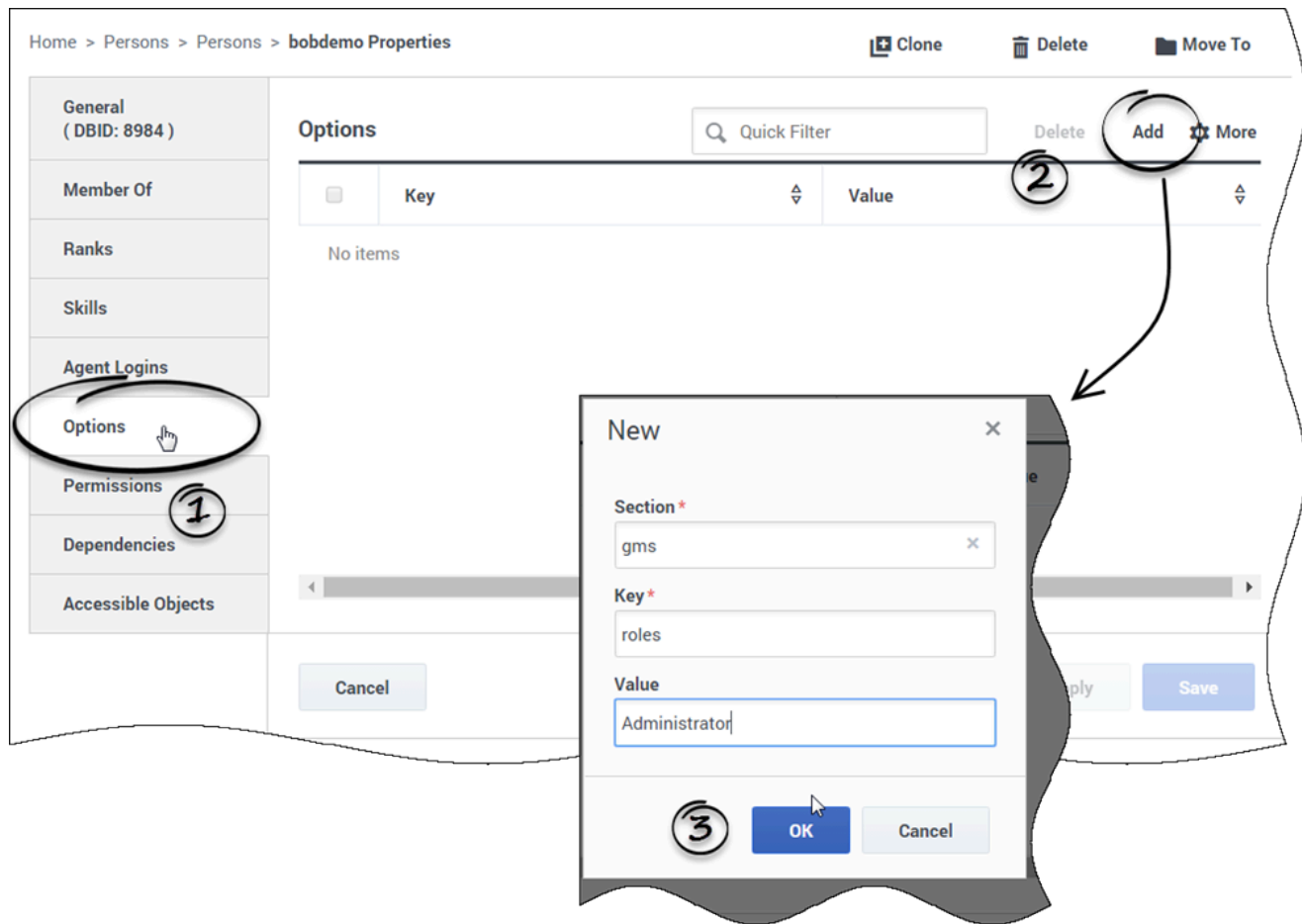
In the **Options** tab, click **Add** to set the gms/roles option:

- Enter gms for **Section**,

- Enter roles for **Key**,

- Enter one of the following values: and its value to:

  - Supervisor: Role used to monitor and configure Callbacks only.

  - Administrator: Role used to administer the creation of Callback Services. This role provides access to all panels and includes the **Supervisor** role.

Set up Historical Reporting

> **Important**
>
> Starting in 8.5.105.12, Genesys Callback reports callback metrics through `UserEvents`. You can enable this feature in your callback service. When enabled, GMS sends the UserEvents to the configured DN. You can then configure your reporting tools to listen to the User Events for this DN and report on callback details.

Prerequisites

Mandatory Genesys Components

Component	Minimum Version
Orchestration Server	8.1.400.24
Universal Routing Server	8.1.400.22
Interaction Concentrator	8.1.506.07
Genesys Info Mart	8.5.005 (GA)
Reporting and Analytics Aggregates (RAA)	8.5.000.02
Genesys CX Insights (GCXI)	9.0.007.03

Historical Reporting Architecture

Reporting on Genesys Callback relies on the user-event mechanism to provide Callback-related metrics and requires Interaction Concentrator, Genesys Info Mart, and Reporting and Analytics Aggregates (RAA) to collect and organize data to produce a database from which Genesys CX Insights (GCXI) can rapidly extract the needed data.
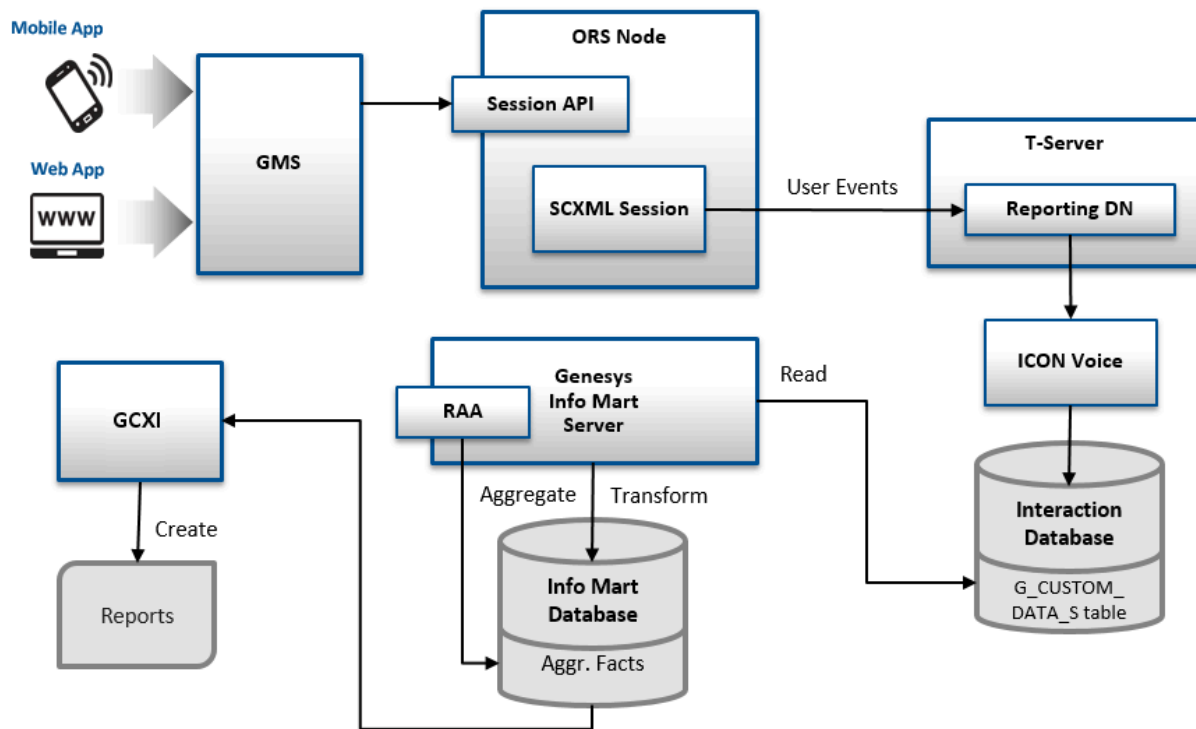
1. Genesys Callback reports callback metrics through `UserEvents` to the configured DN. SCXML strategies that you load through the templates in the Service Management UI collect metrics, and then pass the metrics as user data (KVPs) with two `UserEvent` events, one sent at the start of the session and another, at the end of the session. Genesys Info Mart has certain minimum requirements for the KVPs that must be sent. The out-of-box templates include these KVPs and other KVPs that Genesys Info Mart requires for meaningful reporting. See Genesys Info Mart KVP Requirements for details.

2. Interaction Concentrator (ICON) stores the user data (KVPs) attached to these events into the `G_CUSTOM_DATA_S` table of the Interaction Database (IDB).

3. Genesys Info Mart transforms the data into the `CALLBACK_FACT` table of the Info Mart database; this format can be more quickly loaded into reports.

4. Reporting and Analytics Aggregates (RAA) aggregates the data; in other words, RAA summarizes and organizes the data from Genesys Info Mart in such a way that Genesys CX Insights (GCXI) is able to extract meaning.

5. Genesys CX Insights (GCXI) then presents two out-of-box callback reports: Callback Summary Report and Callback Details Report.

For example, for reporting purposes, the following are some of the keys that GMS sends in
`UserEvents` related to Outbound calls:

- _CB_T_SERVICE_START
- _CB_SERVICE_ID
- _CB_D_CALLBACK_OFFER
- _CB_N_CALLBACK_OFFERED
- _CB_T_CALLBACK_OFFERED
- _CB_T_CALLBACK_ACCEPTED
- _CB_T_CUSTOMER_CONNECTED
- _CB_N_IS_SNOOZED
- _CB_T_NEXT_REDIAL_ATTEMPT
- _CB_N_CALLBACK_MEDIA_ATTEMPTS
- _CB_T_LAST_DIAL_ATTEMPT
- _CB_N_AGENT_ADDED_TO_IXN

The keys that GMS sends depend on the scenario. To get a complete list of the keys that might be
sent, refer to the Callback KVPs reference in this page.

Genesys Info Mart KVP Requirements

The following KVPs are mandatory. Genesys Info Mart will not create a record for the callback event if
the KVP is missing from the `UserEvent`.

- _CB_SERVICE_ID

- _CB_T_SERVICE_START

- _CB_D_CALLBACK_OFFER

- _CB_N_CALLBACK_OFFERED

- _CB_T_CALLBACK_OFFERED

The following four KVPs need to be sent in both `UserEvents` and as call-based attached data in `TEvents`.The duplicated KVPs enable Genesys Info Mart to associate the callback event with interaction data.

- _CB_T_CALLBACK_ACCEPTED

- _CB_T_SERVICE_START

- _CB_SERVICE_ID

- _CB_T_CUSTOMER_CONNECTED

For meaningful reporting, Genesys Info Mart requires a number of other KVPs, depending on the callback scenario. See the Callback KVPs reference, below, for the complete list.

Important

- The _CB_SERVICE_ID is returned by the GMS API in response to the callback request.

- For Inbound Calls, where the in-queue callback offer was presented and accepted, _CB_T_CALLBACK_ACCEPTED, _CB_T_SERVICE_START, and _CB_SERVICE_ID must be attached at the time at which the callback was accepted.

- For Virtual and Outbound Calls, _CB_T_CUSTOMER_CONNECTED must be attached at the time at which the customer was connected.

Virtual Queues

As a best practice, Genesys recommends creating virtual queues associated with the following interaction types:

- Virtual Queue for Inbound calls—This queue is where the regular inbound calls are going to be reported. Those calls are callbacks that were not offered or, offered and rejected.

- Virtual Queue for Virtual callbacks—This queue is where the virtual callbacks are going to be waiting for an agent.

- Virtual Queue for Outbound calls—This is where the callback application will place the real outbound call when it gets confirmation that the right person is connected. The call is removed from this queue after it is successfully delivered to an agent or is abandoned by the customer.

Important

> Virtual queues (VQ) that are used for reporting will make metrics effective, but they are not used for routing in this context.

Related Resources for Historical Reporting

You may also be interested in reading:
  *

  * The Genesys Info Mart Physical Data Model documentation for your RDBMS.

  * The Reporting and Analytics Aggregates Physical Data Model documentation for your RDBMS.

Configure Historical Reporting

Important
Genesys Info Mart and Genesys CX Insights (GCXI) support for callback offered through GMS is provided out-of-box, with no additional configuration required. To see callback data in GCXI reports, however, you need to modify the configuration for other products as explained in this section.

Configure a Reporting DN

Open Genesys Administrator or Configuration Manager and create a new DN of type **Trunk Group DN**. The name of the DN is used inside SCXML scripts, so it should be meaningful and recognizable. For example: `Sip_Switch > DN > REPORTING`

Configure your Callback Service

Edit your callback service in **Callback and Mobile Engagement > Configured Services**, expand the **Reporting** section:

- Set the _rep_userevent_enable option to true to enable reporting.
- Set the _rep_userevent_dn option to the Trunk Group DN that you created previously, used as destination DN of the reporting events.
- Set the _rep_userevent_switch option to the Switch name where you created this DN. This is Switch used to report the events.

Configure Orchestration Server

In the connections of your Orchestration Server application, add the T-Server used to define the reporting Switch and DN in the GMS service configuration. For example, `Sip_Switch`.

Configure Interaction Concentrator

To make Callback reporting work, you need to configure Interaction Concentrator (ICON) for Voice. See here for details.

Set the KVP list

- Configure ICON to store the KVP data provided in the `UserData` section of `EventUserEvents`. ICON will store this data in the G_CUSTOM_DATA_S table of the Interaction Database (IDB):

      ICON > Options > custom-states/store-event-data=all

  By default, `store-event-data` is set to none.

- Configure ICON to store required duplicate KVP data provided in the `UserData` attribute of `TEvents`. ICON will store this data in the G_USERDATA_HISTORY table. To enable this storage, modify your `ccon_adata_spec.xml` file to capture the four `TEvents` KVPs described in the Callback KVPs reference below:

  - `_CB_T_CALLBACK_ACCEPTED`

  - `_CB_T_CUSTOMER_CONNECTED`

  - `_CB_T_SERVICE_START`

  - `_CB_SERVICE_ID`

> ### Tip
> See the `ccon_adata_spec_GIM_example.xml` file in the Genesys Info Mart installation package for an example of the required modification.

Check Interaction Concentrator Connections

Make sure that Interaction Concentrator is connected to the T-Server that is servicing the switch specified in the Callback Service.
For example: `Sip_Switch`.

Start Interaction Concentrator and use logs to verify that it registered on the `REPORTING DN`.

> ### Important
> Interaction Concentrator does not produce historical records for virtual interactions.

Configure Reporting and Analytics Aggregates

Edit the Genesys Info Mart application to enable the `agg-feature\enable-callback` option:
`agg-feature\enable-callback=yes`

> ### Tip
> See here for details about the configuration of your RAA application.

Verify Reporting Data

1. Run your scenario by triggering Genesys Mobile Services and Orchestration Server (ORS) APIs directly.

2. Make sure user events are being delivered to Interaction Concentrator applications by checking T-Server logs. You should see something like this:

```
00:34:20.757 Int 04543 Interaction message "RequestDistributeUserEvent" received from
516 ("OrchestrationServer")
  -- Absent ThisDN, REPORTING was used
  @00:34:20.7570 [0] 8.1.000.62 send_to_client: message EventACK
          AttributeEventSequenceNumber        0000000000000ef8
          AttributeCustomerID          'Environment'
          AttributeTimeinuSecs         757000
          AttributeTimeinSecs          1348817660 (00:34:20)
          AttributeReferenceID         431
          AttributeThisDN          'REPORTING'
          AttributeUserEvent          RequestDistributeUserEvent
  00:34:20.757 Trc 04542 EventACK sent to [516] (00000003 OrchestrationServer 192.168.27.50:40727)
  @00:34:20.7570 [0] 8.1.000.62 distribute_user_event: message EventUserEvent
          AttributeEventSequenceNumber        0000000000000ef9
          AttributeCustomerID          'Environment'
          AttributeTimeinuSecs         757000
          AttributeTimeinSecs          1348817660 (00:34:20)
          AttributeUserEvent          EventUserEvent
          AttributeUserData          [347] 00 0c 00 00..
               'gms_AgentAvailable'         '1348817660755'
               'gms_AgentConnected'          ''
               'gms_IxnCompleted'         ''
               'gms_ServiceName'          'inbound-delay'
               'gms_ServiceStartAt'          '1348817660553'
               'gms_ServiceStoppedAt'          ''
               'gms_SessionEventSeq'          3
               'gms_SessionId'          '65UA6ISSJH76R340BNDQ2DG0DG000036'
               'gms_UserConnected'          ''
               'gms_UserId'          ''
               'gms_WaitingForAgent'          '1348817660744'
               'gms_externalId'          ''
          AttributeANI          '777'
          AttributeDNIS          '333'
          AttributeReferenceID          431
          AttributeThisDN          'REPORTING'
  00:34:20.758 Trc 04542 EventUserEvent sent to [508] (0000000c Icon_Voice 192.168.27.50:42678)
  00:34:20.758 Trc 04542 EventUserEvent sent to [588] (00000004 Stat_Server 192.168.27.50:40728)
  00:34:20.758 Trc 04542 EventUserEvent sent to [592] (00000005 Universal_Routing_Server
192.168.27.50:40744)
```

3. Check your Interaction Concentrator logs and the G_CUSTOM_DATA_S table in Interaction Database and make sure that data is recorded properly.

For example, you should see in Interaction Concentrator logs:

```
00:39:19.569 Int 04543 Interaction message "EventUserEvent" received from 65200 ("SIP_Server@REPORTING")
00:39:19.751 Int 04543 Interaction message "EventUserEvent" received from 65200 ("SIP_Server@REPORTING")
00:39:19.766 Int 04543 Interaction message "EventUserEvent" received from 65200 ("SIP_Server@REPORTING")
00:39:19.987 Trc 25016 Persistent Queue GUD: transaction 10929 is committed. 5 records written
into the queue
00:39:19.987 Trc 25003 Database queue [GUD]: persistent queue transaction 10929 is being processed.
00:39:20.001 Trc 25004 Database queue [GUD]: persistent queue transaction 10929 is processed, committed
 and removed. 5 records are written.
```

4. Optionally, you can also check the content of the CALLBACK_FACT table in the Info Mart database to make sure that the transformation process is correctly executed as well. For example, you can try the following query:

```
SELECT * FROM dbo. CALLBACK_FACT
```

	ADDED_TS	DS_AUDIT_KEY	EVENT_SEQUENCE	CREATE_AUDIT_KEY	TENANT_KEY	SERVICE_ID	FINAL_RECORD	EWT_READY_TO_START_IXN	EWT_WHEN_OFFERED	POS_READY_TO_START_IXN	POS_WHEN_OFFERED	CALLBACK_
1	1465324803	9864	3174	10002	1	445-17a6a647-0fcd-498a-adda-c586922efff1	0	0	0	0	0	0
2	1465324821	9864	3232	10002	1	445-17a6a647-0fcd-498a-adda-c586922efff1	1	0	0	0	0	0
3	1465327890	9864	3952	10008	1	445-5938c861-6f94-48fc-a3a3-81ca343c9cc2	0	0	0	0	0	0
4	1465327908	9864	4011	10008	1	445-5938c861-6f94-48fc-a3a3-81ca343c9cc2	1	0	0	0	0	0
5	1465331525	9864	4842	10008	1	445-16a902ff-4447-40f3-9b8c-4125c7aa489c	0	0	0	0	0	0
6	1465331543	9864	4901	10008	1	445-16a902ff-4447-40f3-9b8c-4125c7aa489c	1	0	0	0	0	0
7	1465395725	9953	3169	10018	1	445-aa8095bc-c556-42b6-b045-a17b53c04641	0	0	0	0	0	0
8	1465395743	9953	3228	10018	1	445-aa8095bc-c556-42b6-b045-a17b53c04641	1	0	0	0	0	0
9	1465396434	9953	3465	10018	1	445-fa6e76ff-20fc-4cb5-82c2-7a31e59646c6	0	0	0	0	0	0
10	1465396452	9953	3524	10018	1	445-fa6e76ff-20fc-4cb5-82c2-7a31e59646c6	1	0	0	0	0	0
11	1465398916	9953	4122	10020	1	445-f3348ed6-1f0d-467d-a525-4e7faf2c39e9	0	0	0	0	0	0
12	1465398934	9953	4180	10020	1	445-f3348ed6-1f0d-467d-a525-4e7faf2c39e9	1	0	0	0	0	0
13	1465400440	9953	4583	10020	1	445-ab1af229-bb0a-4f70-bf4d-1680e29cf581	0	0	0	0	0	0
14	1465400458	9953	4641	10020	1	445-ab1af229-bb0a-4f70-bf4d-1680e29cf581	1	0	0	0	0	0
15	1465408962	9953	6465	10021	1	445-7bc50c56-ab8e-4e7e-8aa7-f2e52ca44abf	0	0	0	0	0	0
16	1465408980	9953	6523	10021	1	445-7bc50c56-ab8e-4e7e-8aa7-f2e52ca44abf	1	0	0	0	0	0
17	1465411643	9978	3209	10021	1	445-df4ad696-d70d-4092-b95c-a7cd76e13287	0	0	0	0	0	0
18	1465411665	9978	3268	10021	1	445-df4ad696-d70d-4092-b95c-a7cd76e13287	1	0	0	0	0	0
19	1465411915	10023	3416	10095	1	445-e1345be8-81d0-43eb-8b0e-bee831237109	0	0	0	0	0	0
20	1465411933	10023	3475	10095	1	445-e1345be8-81d0-43eb-8b0e-bee831237109	1	0	0	0	0	0
21	1465481078	50016	17609	50098	1	445-659ab04d-97e5-41be-8c74-5c1244a062f5	0	0	0	0	0	0
22	1465481096	50016	17668	50165	1	445-659ab04d-97e5-41be-8c74-5c1244a062f5	1	0	0	0	0	0

How to Pass Reporting KVPs of the Inbound Call in the Callback Request

Some historical reporting KVP values are known only by the IVR or application that requests the callback service. Including these KVPs in the historical reporting is optional. If you want to include them, the values can be passed in the HTTP request that starts the Callback service. The following is the list of the KVP parameters that can be passed in the HTTP request. Each maps to the corresponding _CB_X KVP.

- _cb_t_callback_offered
- _cb_d_callback_offer
- _cb_ewt_when_callback_was_offered
- _cb_pos_when_callback_was_offered
- _cb_t_callback_accepted
- _cb_dim_channel
- _cb_dim_callback_offer_type
- _cb_dim_offer_timing

- _cb_n_callback_offers_per_session
- _cb_d_last_callback_offer

> **Important**
> If the agent submits the completed reason in the disposition result, the system will set the reporting key _CB_DISPOSITION to the provided COMPLETED reason.

Reference: Callback KVPs

The following table describes the KVPs that, if sent by GMS in UserEvents, Genesys Info Mart uses to enable Callback reporting.

The following four KVPs must also be sent as call-based attached data.

- _CB_SERVICE_ID
- _CB_T_SERVICE_START
- _CB_T_CALLBACK_ACCEPTED
- _CB_T_CUSTOMER_CONNECTED

> **Important**
> The sample attached-data specification file in the Genesys Info Mart IP includes these four KVPs by default.

KVP	Description	Info Mart Database Target
_CB_TENANT_DBID	The Tenant DBID.	CALLBACK_FACT.TENANT_KEY
_CB_DISPOSITION	Callback state using the format `<state>.<sub state>` where: • `<state>` can be set to: `SCHEDULED, QUEUED, ROUTING, PROCESSING, COMPLETED`. • `<sub state>` can be set: `REDIAL_LIMIT_REACHED, CANCELLED, AGENT, ABANDONED_IN_QUEUE, REJECTED, PUSH_SEND, PUSH_DELIVERY_CONFIRMED, PUSH_SEND_ERROR, FAILED, CONNECTED, TRANSFERRED_TO_RP`.	CALLBACK_DIM_3.DISPOSITION (referenced through CALLBACK_FACT.CALLBACK_DIM_3_KEY)
_CB_SERVICE_ID*	The ID of the callback service request. Depending on the scenario, the value equals the ID of the GMS service instance or ID of the ORS session.	CALLBACK_FACT.SERVICE_ID
_CB_ORIGINATION_IXN_ID **Introduced:** GMS 8.5.200.07	The ID of the inbound call where the callback was originally offered and accepted. You must pass the _cb_origination_ixn_id parameter in your Start Callback query when creating a callback request. If you do not pass the _cb_origination_ixn_id parameter, the value of _CB_ORIGINATION_IXN_ID will be undefined. For chat scenarios, this ID should be the chat interaction ID.	CALLBACK_FACT.ORIGINATION_IXN_ID
_CB_FIRST_OUT_IXN_ID **Introduced:** GMS 8.5.200.07	The call ID of the first outbound call that the callback service created.	CALLBACK_FACT.FIRST_OUT_IXN_ID
_CB_LAST_OUT_IXN_ID **Introduced:** GMS 8.5.200.07	The call ID of the last outbound call that the callback service created.	CALLBACK_FACT.LAST_OUT_IXN_ID

KVP	Description	Info Mart Database Target
_CB_DIAL_1_RESULT **Introduced:** GMS 8.5.200.07	The result of the first callback dialing attempt. One of the following values: • CREATE_CALL_ERROR • BUSY • NO_ANSWER • ANSWERING_MACHINE • ERROR_TONE • FAX • PERSON • CONNECTED • FAILED_TO_ESTABLISH_CUSTOMER_ORIGINATED_MEDIA • PUSH_DELIVERY_CONFIRMED • PUSH_SEND_ERROR • PUSH_DELIVERY_NOT_CONFIRMED • USERORIGINATED_CONNECTED **Notes:** FAILED_TO_ESTABLISH_CUSTOMER_ORIGINATED_MEDIA is a result that must be reported by the user application; otherwise, there is no CTI data that will enable Genesys Callback to identify this result.	CALLBACK_DIAL_RESULTS.DIAL_1_RESULT (referenced through CALLBACK_FACT.CALLBACK_DIAL_RESULTS_KEY)
_CB_DIAL_2_RESULT **Introduced:** GMS 8.5.200.07	The result of the second callback dialing attempt. See _CB_DIAL_1_RESULT for possible values.	CALLBACK_DIAL_RESULTS.DIAL_2_RESULT (referenced through CALLBACK_FACT.CALLBACK_DIAL_RESULTS_KEY)
_CB_DIAL_3_RESULT	The result of the third callback dialing attempt. See _CB_DIAL_1_RESULT for possible values.	CALLBACK_DIAL_RESULTS.DIAL_3_RESULT (referenced through

KVP	Description	Info Mart Database Target
Introduced: GMS 8.5.200.07		CALLBACK_FACT.CALLBACK_DIAL_RESULTS_KEY)
_CB_DIAL_4_RESULT **Introduced:** GMS 8.5.200.07	The result of the fourth callback dialing attempt. See _CB_DIAL_1_RESULT for possible values.	CALLBACK_DIAL_RESULTS.DIAL_4_RESULT (referenced through CALLBACK_FACT.CALLBACK_DIAL_RESULTS_KEY)
_CB_DIAL_5_RESULT **Introduced:** GMS 8.5.200.07	The result of the fifth callback dialing attempt. See _CB_DIAL_1_RESULT for possible values.	CALLBACK_DIAL_RESULTS.DIAL_5_RESULT (referenced through CALLBACK_FACT.CALLBACK_DIAL_RESULTS_KEY)
_CB_T_DIAL_1 **Introduced:** GMS 8.5.200.07	UTC Timestamp of the first dialing attempt.	CALLBACK_FACT.DIAL_1_TS
_CB_T_DIAL_2 **Introduced:** GMS 8.5.200.07	UTC Timestamp of the second dialing attempt.	CALLBACK_FACT.DIAL_2_TS
_CB_T_DIAL_3 **Introduced:** GMS 8.5.200.07	UTC Timestamp of the third dialing attempt.	CALLBACK_FACT.DIAL_3_TS
_CB_T_DIAL_4 **Introduced:** GMS 8.5.200.07	UTC Timestamp of the fourth dialing attempt.	CALLBACK_FACT.DIAL_4_TS
_CB_T_DIAL_5 **Introduced:** GMS 8.5.200.07	UTC Timestamp of the fifth dialing attempt.	CALLBACK_FACT.DIAL_5_TS
_CB_IXN_START_IGNORING_AVAILABILITY	For premise callback, _CB_IXN_START_IGNORING_AVAILABILITY will	CALLBACK_DIM_4.DIAL_IGNORING_AVAILABILITY

KVP	Description	Info Mart Database Target
Introduced: GMS 8.5.200.07	always be 0.	
_CB_FINAL_RECORD	Indicates whether this is a final record about this callback service: 0 = No, 1 = Yes.	CALLBACK_FACT.FINAL_RECORD
_CB_EWT_WHEN_READY_TO_START_MEDIA_IXN	The value of Expected Wait Time (EWT), in seconds, for the service request when the contact center was ready to start the first callback interaction, such as an outbound dialing attempt.	CALLBACK_FACT.EWT_READY_TO_START_IXN
_CB_EWT_WHEN_CALLBACK_WAS_OFFERED	The value of EWT, in seconds, at the time the callback was offered.	CALLBACK_FACT.EWT_WHEN_OFFERED
_CB_POS_WHEN_READY_TO_START_MEDIA_IXN	The customer position in the queue when the contact center was ready to start the first callback interaction, such as an outbound dialing attempt.	CALLBACK_FACT.POS_READY_TO_START_IXN
_CB_POS_WHEN_CALLBACK_WAS_OFFERED	The customer position in the queue when callback was offered.	CALLBACK_FACT.POS_WHEN_OFFERED
_CB_D_CALLBACK_OFFER	The duration of the callback offer, in seconds.	CALLBACK_FACT.CALLBACK_OFFER_TIME
_CB_OFFER_EWT_INBOUND_VQ **Introduced:** GMS 8.5.111.04	Estimated Wait Time for the queue where rejected calls and not offered callbacks are being placed. This value is identical to _CB_EWT_WHEN_CALLBACK_WAS_OFFERED if the same Virtual Queue is used to place accepted callbacks.	CALLBACK_FACT.EWT_WHEN_REJECTED
_CB_N_ABANDONED_DURING_CALLBACK_OFFER **Introduced:** GMS 8.5.111.04	Indicates whether the caller dropped the call without explicitly accepting or rejecting the callback offer: 0 = No, 1 = Yes.	CALLBACK_DIM_4.ABANDONED_DURING_CB_OFFER (referenced through CALLBACK_FACT.CALLBACK_DIM_4_KEY)
_CB_CUSTOMER_ANI	ANI of the customer for in-queue scenarios. This value can match _CB_CUSTOMER_PHONE_NUMBER if the same	CALLBACK_FACT.CUSTOMER_ANI

KVP	Description	Info Mart Database Target
Introduced: GMS 8.5.111.04	number is confirmed or entered. Could also be empty if the ANI is not detected.	
_CB_T_SERVICE_END **Introduced:** GMS 8.5.111.04	UTC timestamp for when service was completed or terminated.	CALLBACK_FACT.SERVICE_END_TS
_CB_D_CUSTOMER_WAITED_BEFORE_OFFER **Introduced:** GMS 8.5.106.14	The amount of time, in seconds, the customer waited in the queue before a callback was offered.	CALLBACK_FACT.WAITED_BEFORE_OFFER_TIME
_CB_D_WAITING_FOR_AGENT_OFFLINE	The amount of time, in seconds, the customer was waiting offline for an agent to become available.	CALLBACK_FACT.WAIT_AGENT_OFFLINE_TIME
_CB_D_ESTABLISH_MEDIA_IXN	The amount of time, in seconds, it took to establish the callback interaction, such as an outbound call.	CALLBACK_FACT.ESTABLISH_MEDIA_IXN_TIME
_CB_D_CUSTOMER_CONNECTED_WAITING_FOR_AGENT	The amount of time, in seconds, the customer was waiting to be connected to the agent after the callback interaction was established.	CALLBACK_FACT.CONN_WAITING_AGENT_TIME
_CB_T_CALLBACK_ACCEPTED*	The UTC timestamp when the callback offer was accepted.	CALLBACK_FACT.CALLBACK_ACCEPTED_TS
_CB_T_CALLBACK_OFFERED	The UTC timestamp when the callback was offered.	CALLBACK_FACT.CALLBACK_OFFERED_TS
_CB_T_READY_TO_START_MEDIA_IXN	The UTC timestamp when the contact center was ready to start the callback interaction. The value matches the time of either an outbound dialing attempt or a push notification prompting the customer to start a call or chat session. **Note:** Set this value only once, before the first dial attempt.	CALLBACK_FACT.READY_START_MEDIA_IXN_TS
_CB_T_CUSTOMER_CONNECTED*	The UTC timestamp when the customer was reconnected to the contact center and started	CALLBACK_FACT.CUSTOMER_CONNECTED_TS

KVP	Description	Info Mart Database Target
	waiting for an agent to be connected.	
_CB_N_AGENT_ADDED_TO_IXN	Indicates whether the agent was successfully added to the callback interaction: 0 = No, 1 = Yes.	CALLBACK_FACT.AGENT_ADDED_TO_IXN
_CB_N_TRANSFER_TO_AGENT_FAILED	Number of times the callback interaction failed to transfer to the agent.	CALLBACK_FACT.XFER_TO_AGENT_FAILED
_CB_N_CUSTOMER_ABANDONED_WHILE_WAITING_FOR_AGENT	Indicates whether the customer abandoned the callback interaction while waiting to be connected to an agent: 0 = No, 1 = Yes.	CALLBACK_FACT.ABANDONED_WAITING
_CB_N_TIMEOUT_WHILE_WAITING_FOR_AGENT	Indicates whether the customer was disconnected because the timeout for waiting for an agent was reached: 0 = No, 1 = Yes.	CALLBACK_FACT.TIMEOUT_WAITING
_CB_N_IXN_REQ_AGENT	Indicates whether the interaction required agent assistance: 0 = No, 1 = Yes.	CALLBACK_FACT.IXN_REQ_AGENT
_CB_N_CALLBACK_OFFERED	Indicates whether callback was offered, at least once, during the session: 0 = No, 1 = Yes.	CALLBACK_FACT.CALLBACK_OFFERED
_CB_N_CALLBACK_ACCEPTED	Indicates whether a callback offer was accepted: 0 = No, 1 = Yes.	CALLBACK_FACT.CALLBACK_ACCEPTED
_CB_N_CALLBACK_MEDIA_ATTEMPTS	The total number of callback attempts or notifications, both successful and unsuccessful.	CALLBACK_FACT.CALLBACK_ATTEMPTS
_CB_T_SERVICE_START*	The UTC timestamp when the callback service started. This value represents either the time of the callback request or the time that the callback offer was played, depending on deployment.	CALLBACK_FACT.SERVICE_START_TS, CALLBACK_FACT.START_DATE_TIME_KEY
_CB_DIM_VQ_DBID	The DBID of the virtual queue used to find the target agent. Genesys Info Mart uses this value in combination to identify the RESOURCE_KEY to use.	CALLBACK_FACT.RESOURCE_KEY
VQ_CFG_TYPE_ID	The configuration type ID of the virtual queue used to find the target agent. Genesys Info Mart uses this value in combination to identify the	CALLBACK_FACT.RESOURCE_KEY

KVP	Description	Info Mart Database Target
	RESOURCE_KEY to use.	
VQ_CFG_TYPE	The configuration type of the virtual queue used to find the target agent. Genesys Info Mart uses this value in combination to identify the RESOURCE_KEY to use.	CALLBACK_FACT.RESOURCE_KEY
_CB_DIM_VQ	The virtual queue used to find the target agent. Genesys Info Mart uses this value in combination to identify the RESOURCE_KEY to use.	CALLBACK_FACT.RESOURCE_KEY
_CB_DIM_CHANNEL	The interaction channel from which the callback originated. One of the following values: • IVR • WEB • MOBILE	CALLBACK_DIM_1.CHANNEL (referenced through CALLBACK_FACT.CALLBACK_DIM_1_KEY)
_CB_DIM_CALLBACK_OFFER_TYPE	The type of callback offer that was presented to the customer. For example, after business hours, SCHEDULED is the only available option; during business hours, business rules might allow only the WAIT_FOR_AGENT option or a combination of SCHEDULED and WAIT_FOR_AGENT. One of the following values: • SCHEDULED • WAIT_FOR_AGENT • COMBINED_SCHEDULED_AND_WAIT_FOR_AGENT • IMMEDIATE	CALLBACK_DIM_1.CALLBACK_OFFER_TYPE (referenced through CALLBACK_FACT.CALLBACK_DIM_1_KEY)
_CB_DIM_TYPE	The type of callback the customer requested. One of the following values:	CALLBACK_DIM_1.CALLBACK_TYPE (referenced through CALLBACK_FACT.CALLBACK_DIM_1_KEY)

KVP	Description	Info Mart Database Target
	• IMMEDIATE - The interaction is created right away while the customer is waiting for the agent (in an online chat session or waiting for a voice call). • WAIT_FOR_AGENT - The interaction is delayed until the agent is about to become available or actually becomes available (as in an agent first scenario). • SCHEDULED - The time for the callback interaction is negotiated with the customer.	
_CB_DIM_CONNECT_ORDER	The order in which the final callback interaction was connected. One of the following values: • CUSTOMER_FIRST • AGENT_FIRST_PREVIEW • AGENT_FIRST_NO_PREVIEW	CALLBACK_DIM_1.CONNECT_ORDER (referenced through CALLBACK_FACT.CALLBACK_DIM_1_KEY)
_CB_DIM_DIAL_DIALOG_RESULT	The result of the final dialog for the callback. One of the following values: • RIGHT_PERSON • RESCHEDULED • CANCELLED • TRANSFERRED_TO_RP	CALLBACK_DIM_2.DIAL_DIALOG_RESULT (referenced through CALLBACK_FACT.CALLBACK_DIM_2_KEY)
_CB_DIM_CALL_DIRECTION	The direction of the final callback interaction. One of the following values:	CALLBACK_DIM_2.CALL_DIRECTION (referenced through CALLBACK_FACT.CALLBACK_DIM_2_KEY)

KVP	Description	Info Mart Database Target
	• CUSTOMER_TERMINATED - Outbound Callback scenarios in which the contact center is dialing out to the customer's number. • CUSTOMER_ORIGINATED - Inbound Callback scenarios in which the contact center notifies the customer-facing application that it is time for the callback interaction, after which the application creates the interaction (such as a call or chat), obtaining the phone number if necessary. In this scenario, a customer call comes into the contact center as a regular inbound call, but it is recognized as the callback interaction.	
_CB_DIM_FINAL_DIAL_RESULT	The result of the final callback dialing attempt. One of the following values: • CREATE_CALL_ERROR • BUSY • NO_ANSWER • ANSWERING_MACHINE • ERROR_TONE • FAX • PERSON • CANCEL • CONNECTED • FAILED_TO_ESTABLISH_CUSTOMER_ORIGINATED_MEDIA	CALLBACK_DIM_2.FINAL_DIAL_RESULT (referenced through CALLBACK_FACT.CALLBACK_DIM_2_KEY)

KVP	Description	Info Mart Database Target
	• PUSH_DELIVERY_CONFIRMED • PUSH_SEND_ERROR • PUSH_DELIVERY_NOT_CONFIRMED • USERORIGINATED_CONNECTED **Notes:** 1. FAILED_TO_ESTABLISH_CUSTOMER_ORIGINATED_MEDIA is a result that must be reported by the user application; otherwise, there is no CTI data that will enable Genesys Callback to identify this result. 2. CANCEL is set when the on_dial plugin returned action=CANCEL.	
_CB_DIM_OFFER_TIMING	Specifies whether the callback offer was made during operational (business) or non-operational hours. One of the following values: • ON-HOURS • OFF-HOURS	CALLBACK_DIM_2.OFFER_TIMING (referenced through CALLBACK_FACT.CALLBACK_DIM_2_KEY)
_CB_DIM_FINAL_TARGET	The routing target that was used to find the agent.	CALLBACK_DIM_3.FINAL_TARGET (referenced through CALLBACK_FACT.CALLBACK_DIM_3_KEY)
_CB_ORS_SESSION_ID **Introduced:** GMS 8.5.114.09	The Orchestration Server (ORS) session ID used to manage the callback. If multiple sessions were used (for example, because an ORS session terminated unexpectedly during the callback), the last session ID is reported.	CALLBACK_FACT.ORS_SESSION_ID
_CB_EWT_WHEN_READY_TO_START_LAST_MEDIA_IXN **Introduced:** GMS 8.5.200.07	Estimated Wait Time in seconds when the last dial attempt was made or the last push notification sent.	CALLBACK_FACT.EWT_WHEN_LAST_DIAL

KVP	Description	Info Mart Database Target
_CB_POS_WHEN_READY_TO_START_LAST_MEDIA_IXN **Introduced:** GMS 8.5.200.07	Position in queue when the last dial attempt was made or the last push notification sent.	CALLBACK_FACT.POS_WHEN_LAST_DIAL
_CB_PRIORITY_WHEN_CALLBACK_ACCEPTED **Introduced:** GMS 8.5.200.07	Priority of the interaction (real or virtual) when the callback offer was accepted.	CALLBACK_FACT.PRIORITY_WHEN_CB_ACCEPTED
_CB_PRIORITY_WHEN_CUSTOMER_CONNECTED **Introduced:** GMS 8.5.200.07	Priority of the virtual interaction when the customer was connected.	CALLBACK_FACT.PRIORITY_WHEN_C_CONNECTED
_CB_PRIORITY_AT_THE_END_OF_ONLINE_WAIT **Introduced:** GMS 8.5.200.07	Priority of the virtual interaction when the customer was connected to the agent. If the customer abandoned while waiting in queue, then this value is the priority of the call when the customer disconnected.	CALLBACK_FACT.PRIORITY_WHEN_A_CONNECTED
_CB_EWT_THRESHOLD_WHEN_OFFERED **Introduced:** GMS 8.5.200.07	Value of the EWT threshold used to decide whether the callback offer should be made or not. Pass this value as an argument of the application that is responsible for making the callback offer.	CALLBACK_FACT.EWT_THRESHOLD_WHEN_OFFERED

*This KVP must be sent twice -- as call-based attached data in a TEvent and as UserEvent-based user data.

Restore Virtual Queue Position Upon Resubmit

In some scenarios, the Orchestration Server (ORS) fails to handle a callback session that is in the QUEUED state and GMS will resubmit the callback service request to another ORS.

Starting in **8.5.106.16**, GMS can maintain the queue priority and interaction age by passing in calculated settings. This is possible only if you enable the Restore Virtual Queue Position Upon Resubmit feature by configuring your GMS application and callback service as detailed in this page.

Enable the Restoring of Virtual Queue Position Upon Resubmit

To enable this feature, set the `callback/enable-restore-vq-position` to `true` in your GMS application. By default, the option is `true` and enables the feature. As result, both the virtual call **priority** and **interaction age** will be restored when GMS will resubmit the callback.

How to set your Virtual Queue Priority

The Virtual Queue Priority depends on the following callback service options:

- `_urs_vq_priority`—Priority to be set for the virtual interaction when submitting to `_urs_virtual_queue`. If you leave this option blank, no priority will be set.
- `_urs_vq_priority_increment`—The amount by which the priority will be incremented. For example, 10 to increment the priority by 10 each time. If you leave this option blank, the incrementation of the priority is disabled.
- `_urs_vq_priority_increment_interval`—Number of seconds between each incrementation of the priority. For example, 60 to increment the priority every 60 seconds. If you leave this option blank, the incrementation of the priority is disabled.

You can either set these options in your callback service through the Service Management UI, or pass them the HTTP request that starts your callback Service.

These values will be used to calculate an updated `_urs_vq_priority` value using the following formula:

```
_urs_vq_priority + _urs_vq_priority_increment * floor( (current_time - callback_start_time) /
urs_vq_priority_increment_interval )
```

> **Important**
> If one of the _urs_vq_priority, _urs_vq_priority_increment, or
> _urs_vq_priority_increment_interval options is not set, GMS does not update the
> urs_vq_priority parameter when resubmitting the callback.

Interaction Age

If the _urs_call_interaction_age parameter is neither set in the callback service nor passed in the
HTTP request that starts the service, it is set upon resubmit to a value based on the callback start
time. Otherwise, the originally provided value of _urs_call_interaction_age is maintained upon
resubmit.

Callback and Mobile Engagement UI

> **Important**
>
> Before you start using this interface, make sure that your GMS application is up and running, and that you properly configured your user permissions.

Scope of Use
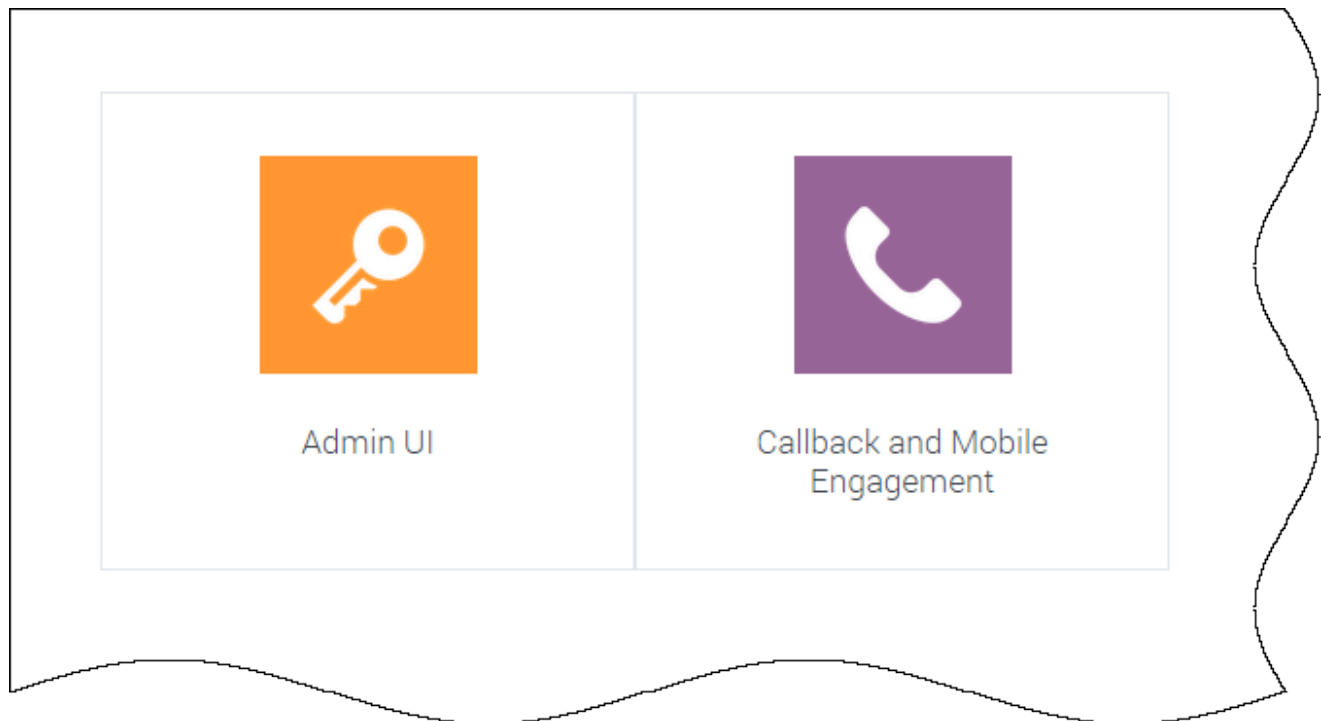
Callback UI is provided through the Service Management User Interface of the Genesys Mobile Services (GMS) at the following location: `<GMS Local Host>:8080/genesys`

- Admin UI
  - Access Reporting tools.
  - Test your Callback Scenario with a Sample interface.
  - Manage Callbacks, and manage the exception list for Callback services.
- Mobile Engagement and Callback
  - Service Templates: Upload the Callback templates.

- Configured Services:

    - Create a Callback service implementing a given scenario.

    - Manage Call Center operating hours and holiday schedules by implementing an Office Hour service.

  - Callback UI: Manage completed and scheduled Callbacks.

> ### Important
>
> For further information and detailed help about the Admin UI, refer to the Service Management Help.

Implement a Callback Service

If you already have configured Callback as detailed in the Configuration chapter, you can now create your callback service:

- First, add a Callback Service using the Admin UI. This Callback service will implement the scenario of your choice.

- Once the Callback Service is instantiated, you must configure some of its parameters according to the chosen scenario.

- You can implement additional Callback features by configuring sections such as Treatments, Reporting, or Exception Patterns in your service's details.

- Once you are done with your service configuration, you can start using it:

    - You can create and manage callbacks in the Callback UI.

    - You can use the Callback Services API to implement an application.

Scenario Sample

The **Sample** tab in Admin UI supports the scenarios described in the Callback Scenarios. This sample can be used to test that your Callback configuration works.

For more information about the sample, as well as a download option, see the Lab Sample in the *Service Management Help*.

Add a Callback Service

After Callback is configured, you must create a Callback Service for each Callback scenario that you want to implement. Then, you can manage Callback interactions in the Callback UI.

- To create a Callback service, you need Administrator permissions.

- You must set up Office Hours if you want to implement **Scheduled Callback**.

Load the Callback Service Template
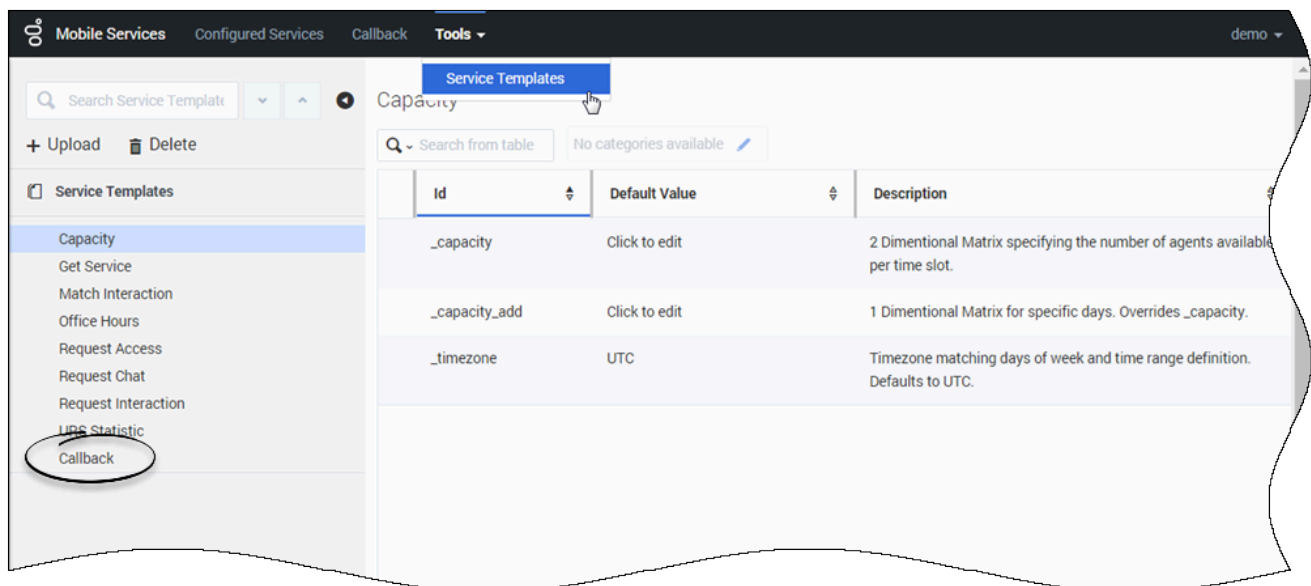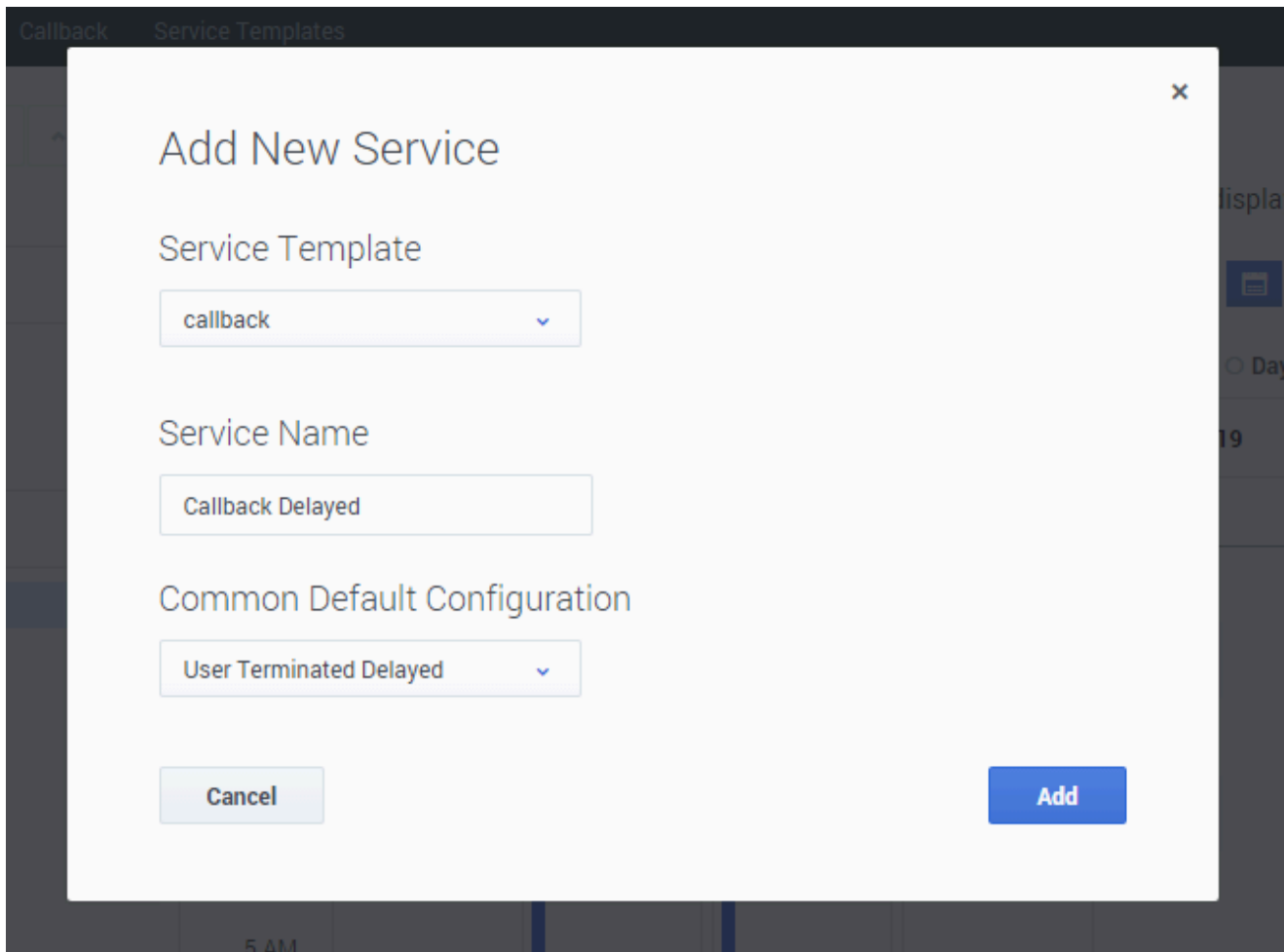
Open the Service Management UI, and navigate to **Callback and Mobile Engagement > Service Templates**. By default, **callback** should be part of the **Service Templates** list.

If not, you must load the Callback service template before you can create a Callback service. The `callback.zip` template is located in the *<GMS installation directory>/service_templates* directory.

Once the callback template has been loaded, it is available in the filter drop-down list of the **Services** tab.

Create a Callback Service

Navigate to the **Services > Configured Services** tab. There, you can manage your services based on the templates. The services and categories can be collapsed or expanded for easier viewing.

1. Click **Create**.

2. Enter a **Service Name**: This name will be used as the {callback-execution-name} parameter in your Callback queries.

3. Select your **Common Default Configuration**. These configuration scenarios are detailed in the Callback Scenarios section of this guide.

4. Click **Add**. An information message confirms the service creation.
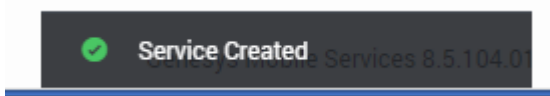
The new Callback service appears in the *Configured Services* list.

> **Important**
>
> The service is also created in the `service.{callback-execution-name}` section of your GMS configuration.

Configured Callback Services

Key parameters for the service are automatically populated with the appropriate default values. For cluster configurations, all changes made in a service will be replicated into the entire GMS cluster.

> **Important**
>
> To configure your Callback service, refer to the configuration options described in the associated scenario page. See the Callback scenarios for further details.

The parameters have the following characteristics:

- Mandatory parameters - are identified with a lock icon; you cannot rename them or remove them.

- Optional parameters - you can rename them by hovering your cursor over the value field (you will see a pencil icon), click, and then enter the new value. You can delete optional parameters by clicking the *Delete* button.

- Advanced parameters - can be displayed by selecting the **Advanced** button in the upper right.

- Request parameters - are identified with an ***Request Parameter*** label in the Description.

Important

Request parameters are usually provided within the request itself. Genesys recommends that you do not configure the Request parameters through this UI, otherwise, the parameter within the request will be overridden. If a value was entered through this UI and you wish to remove it at some later date, you can click the garbage can icon and the value becomes *not specified.*

Accessing Callback Services and Records with the Callback API

The URLs used by the Callback API are dependent on the name of the Callback service that you have just created. Callback services are available at the following URL:

```
http://<host>:<port>/{base-web-application}/service/callback/{callback-execution-name}
```

For instance, if you create a callback service named callback-for-mobile, then {callback-execution-name} is callback-for-mobile and the callback service is available at:

```
http://<host>:<port>/{base-web-application}/service/callback/callback-for-mobile
```

For further details, see Callback API.

Manage Callbacks

Updated in 8.5.110, 8.5.111

The *Callback* UI displays the list of Callbacks that are still alive. After the Callback due date (which is the _desired_time parameter), the service record will appear in the UI for the amount of time configured in the _ttl parameter (in seconds).

Callback records can have the following states:

- SCHEDULED - Request is handled by Callback Management service (there are no sessions started in ORS). While in this state, the request will be handled by Management when the specified `desired_time` is upcoming.

- QUEUED - Callbacks actively waiting for agent in ORS/URS; agent not assigned yet.

- ROUTING - Agent is reserved but the call is not yet routed to the agent.

- PROCESSING - Callback being handled by assigned agents.

- COMPLETED - Callback was completed with `_callback_reason`, for example, timed-out, cancelled, and so on.

- PAUSED - Callback was paused. See Pausing Callback for details.

> **Important**
> You must have the appropriate Supervisor role in order to use the Callback UI.

Callback UI Overview

The following features are available:

- The dropdown at the top left gives you the option to display Callbacks using a pre-defined range for the past day, week, or month; or for the next day, week, or month.

- Refresh – You can force the interface to refresh the list of Callbacks.

- Search Box – You can select a column field in the dropdown at the right of the Search Box. Then, the interface filters the results dynamically as you type.

Limitations of the Interface

- The total number of callbacks that the table can display is 100000.
- Callbacks are displayed in pages of 100 items for Internet Explorer and 250 for other browsers.
- There is a dropdown which you can select to get pre-defined ranges or set a custom range.

Advanced Options

Advanced Options - Opens a new window where you can customize how the Callback Management table displays:

- **Add Column Field** - You can add your own custom column field to display in the table. This option can be useful when you have created your own properties that you want to display in the table. You can add more than one field, which will display in the same column. In this case, to enhance readability in the single column, you can use the Alias option, and create a short name to display.

- **Filter Table by States and Service** - You can include one or more states to display in the table, and/ or you can select a service to display in the table. Note: You can only select one service at a time to display.

- Max # of Callbacks per Service - Default is 500.

Create or Schedule a Callback

You can submit or schedule a callback request by clicking **Create Callback**. A new dialog opens and you can add a new Callback to be displayed. You can select the **Callback Type** and **Service Name**, add your own properties to the Callback, and enter the following Callback Parameters:

- *No results* - Your Callback number.

- _urs_virtual_queue - Queue to use for this Callback if several virtual queues are used for Callback with identical configuration. If you are defining the `_urs_virtual_queue` here, you must then remove this option from the Callback Service (through Genesys Administrator), because it cannot be defined in both places.

- _request_queue_time_stat - Queue statistics. For example, "ExpectedWaitTime;Queue;8999@SIP_Server;Environment".

> ### Important
> Make sure to set _wait_for_user_confirm to `false` in the selected service.

Edit a Callback

In the Callback table, for Callbacks that can be edited, a blue pencil displays in the last column. Clicking this pencil displays the edit options for that Callback.

- You can choose to reschedule the Callback (only for SCHEDULED Callbacks).

- You can choose to cancel the Callback.

Get Additional Details about the Queued State

You can expand the QUEUED state to get ORS-level diagnostics about the queued Callback.

The following information can be displayed:

- **Estimated wait time**: The estimated time that the customer will wait for the callback. For example: 46.5 seconds
- **Position in queue**: The callback's current position in the queue. For example: 3
- **Agents Logged in**: The number of agents that have logged in. For example: 3
- **Callback type**: The type of callback. For example: WAIT_FOR_AGENT
- **Callback version**: The version of the ORS Callback strategy. For example: v2.39
- **Channel**: The callback channel. For example: WEB
- **Dial attempt number**: The number of dials that the agent has attempted. For example: 2
- **Dial Result**: The result of the callback dial. For example: PUSH_DELIVERY_NOT_CONFIRMED

- **EWT at first outbound call**: The estimated wait time when the first outbound call happened. For example: 22.5

- **EWT at callback offer**: The estimated wait time when the callback is offered. For example: 0

- **Is snoozed**: Shows whether the callback is snoozed or not. For example: `false`

- **ORS session ID**: ORS session ID of the callback. For example: 00NEEH6C74C6NAC01G015B5AES000005

- **Position at first outbound call**: The callback's position in the queue when the first outbound call happened. For example: 3

- **Position at callback offer**: The callback's position in the queue when the callback is offered. For example: 3

- **Current priority**: The callback's priority. For example: 0

- **Routing target**: The callback's target or skill expression. For example: GMSCallbackAgents@stat.GA

- **Time of first outbound call**: The time when the first outbound call happened. For example: 1/3/2017 11:26:21

- **Time callback was accepted**: The time when the callback is accepted. For example: 1/3/2017 11:26:02

- **Time of next outbound call**: The time when the next outbound call happened. For example: 1/3/2017 11:27:24

Bulk Cancel and Export of Callback Records

Added in: 8.5.110, updated in 8.5.111

Important

This feature is enabled by default in your GMS configuration. To disable it, set disable-bulk-cancel-and-export-callback to `true`.

Bulk Cancel

Make sure to select an appropriate Time Range to filter callbacks, then make a callback selection to activate the **Cancel Callbacks** button. When you click this button, the **Cancel Callbacks Confirmation** dialog opens, displaying the selected callback cancellations.
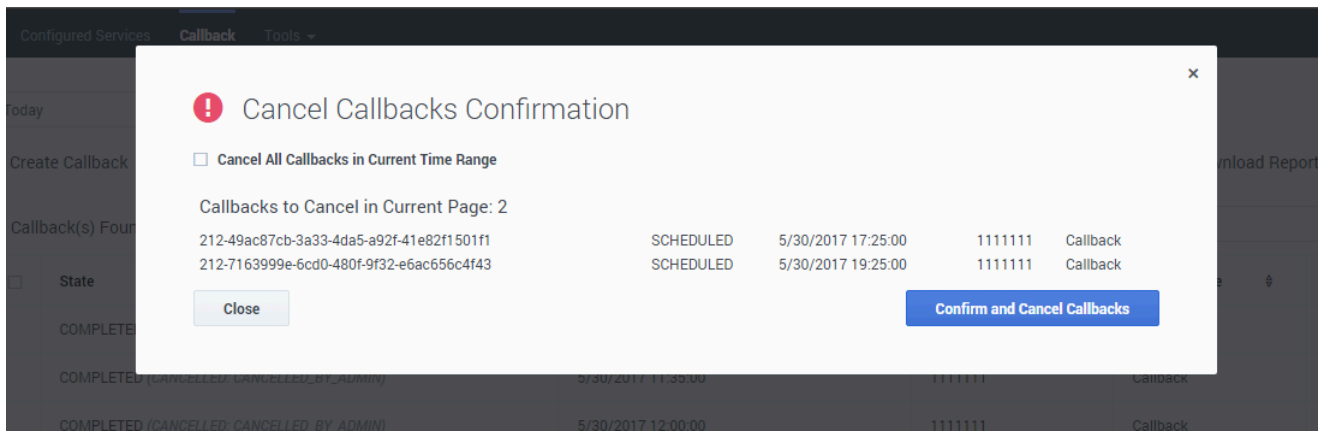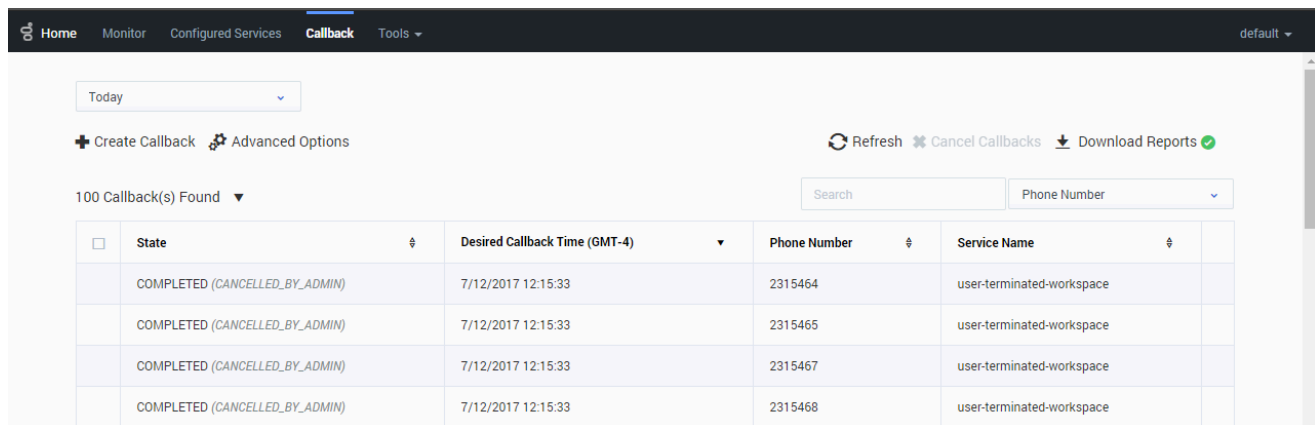
You can either:

- Continue with the selections you have made and click confirm to cancel them.
- Check **Cancel All Callbacks In Current Time Range**, to override your previous callback selections and delete all callbacks in the current time range. In that case, the list of displayed callbacks disappears from the dialog window. Confirm to cancel the records.
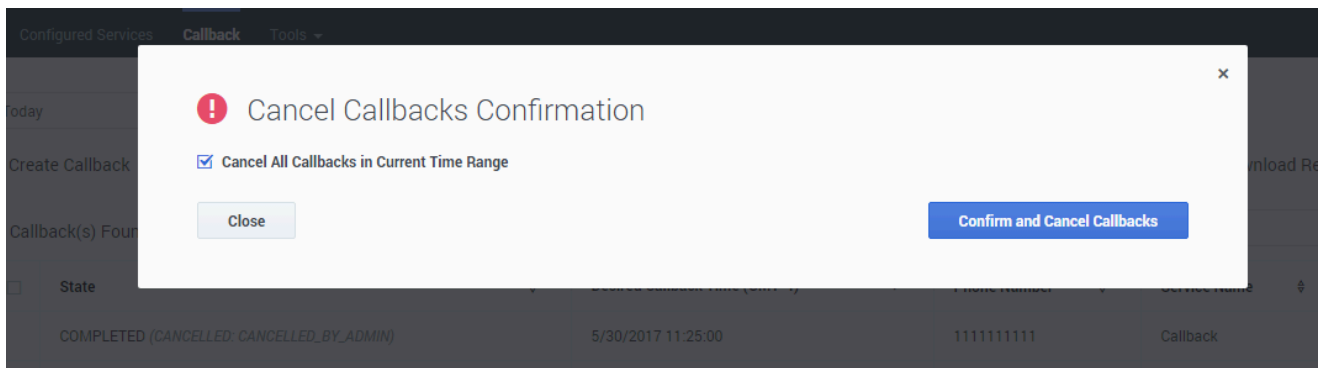
The resulting display shows all the callbacks that you have selected to cancel and that are now in COMPLETED(CANCELLED_BY_ADMIN) state.

Download the Reports

Then, Download Reports dialog can provide two files:

- **Export Cancelled Callbacks** allows you to export a CSV file that includes all of the recently cancelled Callbacks. Click the Refresh button beside the label to get an updated report if the one downloaded seems out of date.

- **Cancellation Summary Report** is available only if you recently cancelled some callbacks. This report shows the cancellation status of the recently Cancelled Callbacks.
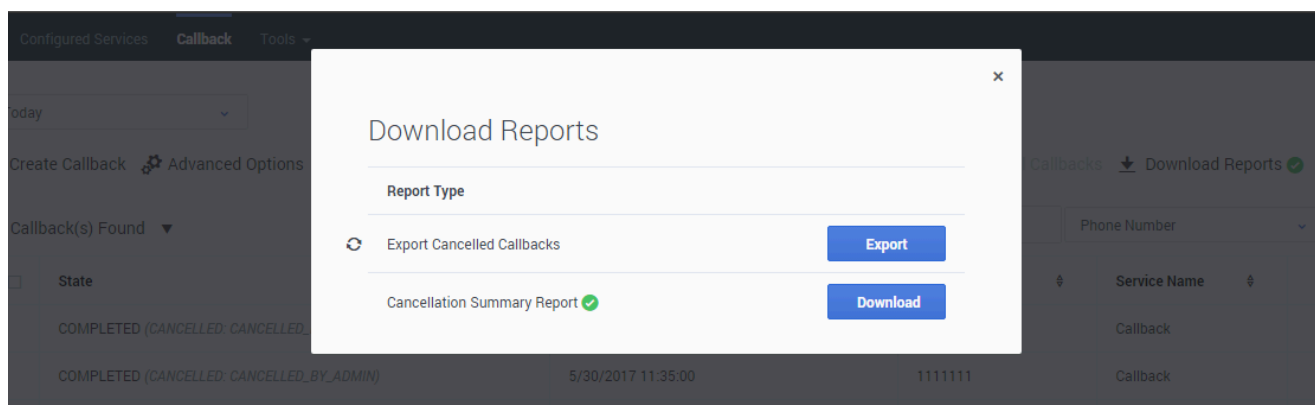
Configure Custom Date Range in Advanced Options

As detailed previously, the **Custom Date Range** selection in the drop-down list enables you to specify a start time and end time for the displayed Callbacks. If you are often using the same date filters, consider saving these filters in advanced options.

1. In the **Callback** Panel, click **Advanced Options**, then **Add Custom Date Range**.

2. Add your date filters with appropriate names.

3. Once saved, they will appear at the bottom of the drop down list.
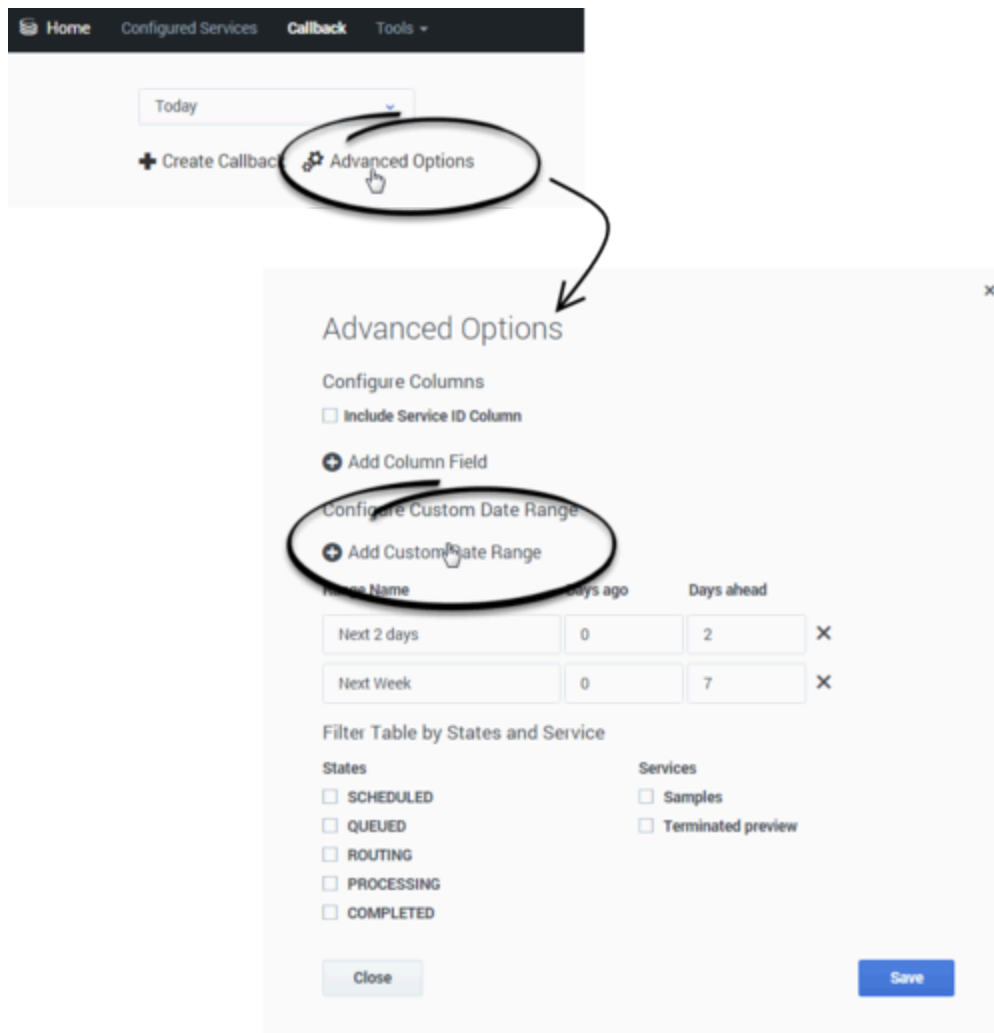
Configure Aliases to Display Custom Fields

Introduced in: 8.5.111

You can configure aliases for additional Column Names that display the custom fields passed in your Callback queries parameters. To do so:

1. Make sure that *No results* is not set to true in your GMS configuration.

2. Allow the list of the parameters that you wish to see displayed by using the filter-keys and returned-keys options.

3. Create a list of aliases for the parameters that you wish to see displayed by using the callback_column_alias option.

For example, if you pass the _L0B parameter in your callback queries, enable its alias as follow in the

callback section:

```
filter-keys=_callback_state,_callback_reason,_request_queue_time_stat,
_request_ewt_service,_vq,_LOB
```

```
returned-keys=_desired_time,_callback_state,_callback_state,_callback_reason,
_ors_session_id,_LOB
```

```
callback_column_alias = {"_LOB": "Line Of Business"}
```

☐	State ⇅	Desired Callback Time (GMT-4) ▼	Phone Number ⇅	Service Name ⇅	Line Of Business ⇅	
☐	SCHEDULED	7/25/2017 14:00:00	10015	user-terminated-workspace	Secondary	✎
☐	SCHEDULED	7/25/2017 14:00:00	10016	user-terminated-workspace	Special Projects	✎
☐	SCHEDULED	7/25/2017 14:00:00	10013	user-terminated-workspace	Primary	✎
☐	SCHEDULED	7/25/2017 14:00:00	10017	user-terminated-workspace	Not Specified	✎
☐	SCHEDULED	7/25/2017 14:00:00	10011	user-terminated-workspace	Primary	✎
☐	SCHEDULED	7/25/2017 14:00:00	10020	user-terminated-workspace	Not Specified	✎
☐	SCHEDULED	7/25/2017 14:00:00	10014	user-terminated-workspace	Not Specified	✎
☐	SCHEDULED	7/25/2017 14:00:00	10019	user-terminated-workspace	Not Specified	✎
☐	SCHEDULED	7/25/2017 14:00:00	10012	user-terminated-workspace	Not Specified	✎
☐	SCHEDULED	7/25/2017 14:00:00	10018	user-terminated-workspace	Not Specified	✎

Custom Date Range | 7/25/2017 | 7/26/2017 | Submit

+ Create Callback ⚙ Advanced Options   ↻ Refresh ✖ Cancel Callbacks ↓ Download Reports

10 Callback(s) Found ▼   Search   Phone Number ⌄

▲ Back to top

Then, the Callback UI will display the _LOB values in the **Line Of Business** column.

If you wish to display multiple custom values, the UI will group them in the **Custom Fields** column. For example, if you wish to display the values for the _LOB and _service_type query parameters, configure the alias as follow:

```
filter-keys=_callback_state,_callback_reason,_request_queue_time_stat,
_request_ewt_service,_vq,_LOB,_service_type
```

```
returned-keys=_desired_time,_callback_state,_callback_state,_callback_reason,
_ors_session_id,_LOB,_service_type
```

```
callback_column_alias = {"_LOB": "Line Of Business", "_service_type": "Service Type"}
```

Then, the Callback UI will display the `_LOB` and `_service_type` values in the **Custom Field** column. The aliases will identify each custom field.

Note that you can also use the aliases to search and filter the list of displayed Callbacks.

Tip

Keys for Custom Fields do not need to start with underscores: you can use _LOB or LOB for the name of your Custom Field.

Play Treatments

> **Important**
>
> The following configuration is required for voice scenarios.

Set up Play Treatments

You can configure the treatments to play in the **Callback and Mobile Engagement > Configured Services** tab. Expand your Callback service and then, expand **Voice Treatments**.

- If you are using Genesys Media Server, place the treatment files that will be used by the Callback service in the *<MCP Install Dir>/GMSApplications* directory and enter the name of your file.

- For other media servers, enter the URL of your files.

In voice scenarios, the customer call can be placed in a virtual queue, waiting for an agent. You can

set up custom treatments for calls waiting for an agent:

- To play some music while the customer is waiting for an agent, set the _treatment_waiting_for_agent option to your music file location (as stated above).

- If you configure the _max_time_to_wait_for_agent_on_the_call option for the waiting for agent state and if this timeout occurs, the service will play the music file set by the option _treatment_find_agent_fail option.

Customize Treatments for User Terminated Calls

The outbound call has Call Progress Detection (CPD) enabled. If a human answers the call, the Media Server provides the CPD result in the _call_state parameter of the request submitted to your SCXML plugin. The interaction ID is provided in the _interaction_id parameter, which is set to undefined if the call is not answered.

If you implement one of the user terminated callback scenarios, you can define treatments for outbound calls by using an SCXML plugin, a VXML application, or by setting Callback built-in treatments through your service options. Callback built-in treatments are part of the the the Callback template strategy and they will be executed if you set options that start with **_treatment**, such as _treatment_customer_connect, _treatment_call_failure_answering_machine, _treatment_find_agent_fail, and so on.

The figure below is a simplified diagram of the flow for the treatments. The table that follows details the Callback service options that configure the condition points in the diagram.
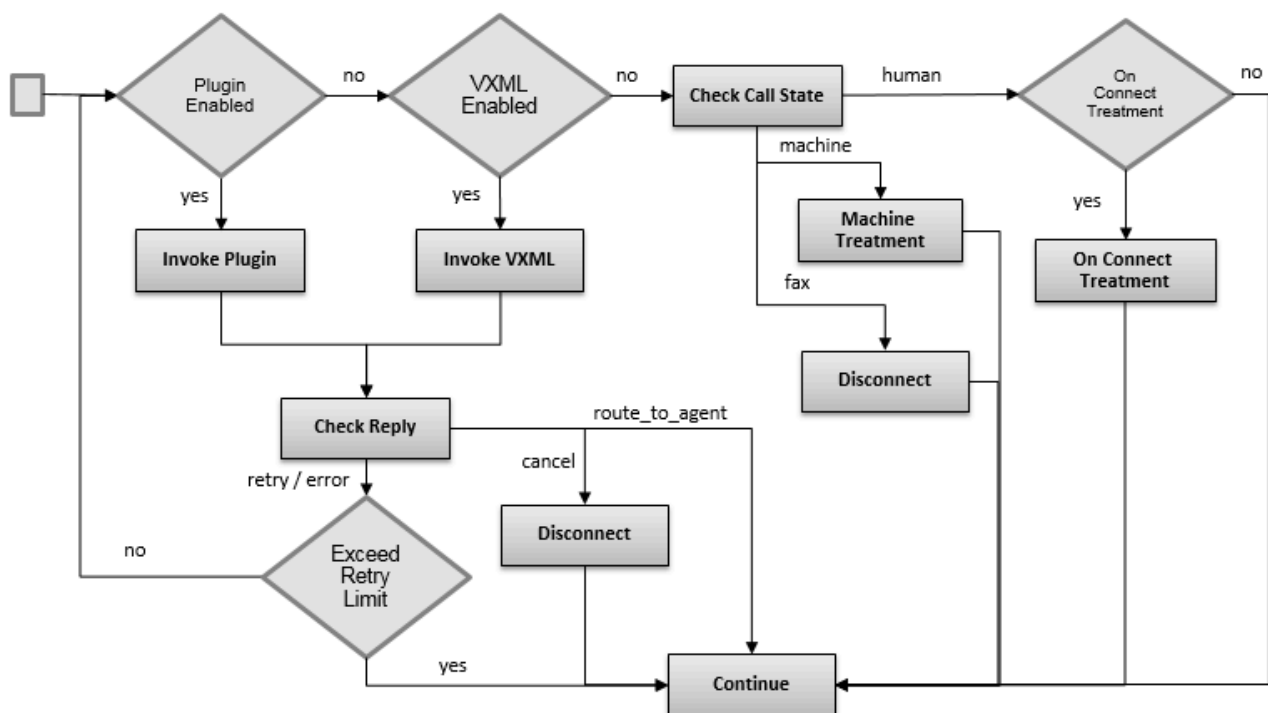
Customization	Answer type	Options
Plugin Enabled = yes	machine, human	`_plugin_on_dial_url` = <URL of your SCXML strategy> _plugin_on_dial_invoke_on_call_failed = true
	human only	`_plugin_on_dial_url` = <URL of your SCXML strategy> _plugin_on_dial_invoke_on_call_failed = true or false
VXML Enabled = yes	human only	`_on_customer_connect_treatment_url` = <URL of your VoiceXML application>
On Connect Treatment = yes (optional)	human only	`_treatment_customer_connect` = <music file location>
Machine Treatment	answering machine	`_treatment_call_failure_answering_machine` = <music file location>
Retry Limit Exceeded = yes	all	Number of dial attempts > `_max_dial_attempts`

Implement an SCXML Plugin
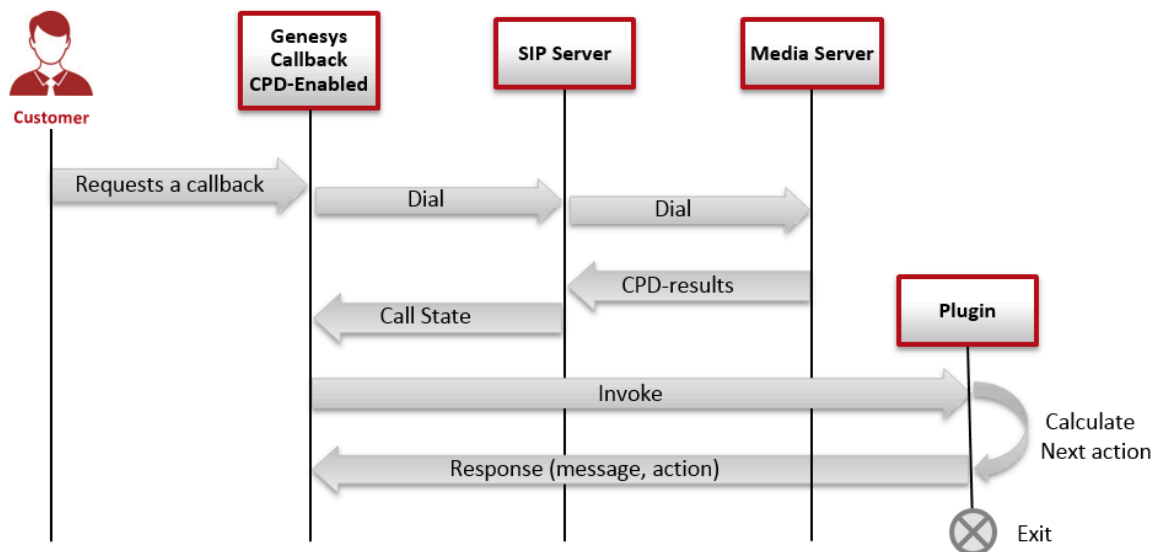
If a user requests a new Callback Service which is CPD-enabled, Callback sends a request to the SIP Server to dial a call and waits for CPD-Results to invoke the SCXML plugin. The SCXML plugin performs the custom treatments. Callback waits for the plugin response, which includes the following

values:

- Mandatory: `action` and `message`.
- Optional: `customer_number`, `customer_number_prefix`, and `delay`.

Then, Callback executes the action.

> ### Important
> If the customer ends the call while the plugin is executing, the plugin should detect hangup by the customer and return the response with its action set to `cancel` or `retry`, and its `message` set to `interaction_deleted`. Otherwise, the plugin message value should always be set to OK.

Invoke your Plugin

Callback invokes the SCXML plugin by starting an SCXML session with the full path to an SCXML entry page provided in the _plugin_on_dial_url option, for example:
_plugin_on_dial_url = http://server:port/StrategyName/src-gen/IPD_Pluging_Entry.scxml

The Callback strategy generates the Callback parameters listed in the table below, and adds them to the **<session:start>** request, in the `request-specific` attribute of the request's body:

Parameter name	Parameter description
_call_state	JSON-formatted call state object returned by ORS on a successful CPD. Set to a value from the `_genesys.ixn.callState` documented enumeration. Possible values are: - Ok=0 (human answer) - NoAnswer=7 - SilDetected=8 - AnsweringMachineDetected=9 - FaxDetected=17
_interaction_id	Set to the interactionID, or `undefined` for the no answer case.
_customer_number	Customer number dialed prior to the plugin invocation.
_gms_service_id	GMS Service ID responsible for this Callback.
_ors_service_id	ORS Service ID responsible for this Callback.
_user_data	User data passed to the user request when starting Callback.

_dial_attempt	Number of outbound attempts made to connect to the customer (includes the current attempt).
_ttl	Time To Live (in seconds) that Callback waits to receive a response. Callback must receive the response before TTL expires to continue the processing of the Callback service request; otherwise, the Callback session will exit with an error. The TTL value is set by the _plugin_on_dial_timeout option of your Callback service in the Admin UI.
_reply_url	HTTP URL to which the plugin must send the asynchronous response after the execution of the plugin logic.

The plugin strategy needs to use these parameters for various purposes. You can get examples of how you can use them in the code of the Custom Callback Plugin Sample.

Invoke Plugin For No Answer or Machine Answer

By default, the _plugin_on_dial_invoke_on_call_failed option of your callback service is true and invokes the plugin if the outbound call is not answered or is answered by machine.

> **Important**
>
> To disable the plugin invoke in that scenario, you must set the _plugin_on_dial_invoke_on_call_failed option to false.

In a **no answer** scenario:

- The _interaction_id parameter is set to undefined.
- The _call_state parameter is set to NoAnswer (7).

> **Important**
>
> In this scenario, the plugin should return a reply message with its action parameter set to cancel or retry.

In a **machine answer** scenario:

- The _call_state parameter is set to AnsweringMachineDetected (9).
- If the SIP Server configuration has the **TServer / am-detected** option set to connect, the call remains connected and _interaction_id is set to the corresponding voice interaction ID. In that case, the plugin can play a message to be recorded.
- Otherwise, SIP Server hangs up the call and _interaction_id is set to undefined.

Invoke Plugin with Attached Call

By defaut, the _plugin_on_dial_associate_ixn option is set to true in your Callback service's configuration and passes the control of the voice interaction (call) to the plugin.

- Before returning the reply, the plugin must detach the call to allow Callback to regain control of the interaction. See the Custom Callback Plugin Sample example for further details. The Plugin sample includes a re-usable sub-workflow that performs the detach interaction step and returns the plugin response.

- The plugin must monitor the interaction.deleted event; if this event occurs, the plugin must return a reply with its message parameter set to interaction_deleted and its action parameter set to either cancel or retry.

Disabling the plugin's interaction control (by setting _plugin_on_dial_associate_ixn to false) provides compatibility with the earlier release of the plugin feature.

> **Important**
>
> Genesys recommends to pass the interaction's control to the SCXML plugin to ensure the best compatibility with Composer strategies. Composer's various interaction-related blocks are configured by default to handle the interaction.added and interaction.present events that occur when the interaction is associated to the plugin session.

Invoke Plugin with no Attached Call

Composer strategies are built to start processing when an interaction is attached to the Orchestration session, but the plugin is invoked with no attached interaction if you set the following options in your Callback service configuration:

- _plugin_on_dial_invoke_on_called_failed=true and the call failed

- _plugin_on_dial_associate_ixn=false

If there is no attached call, the Callback session sends an event named plugin.start to the plugin session and you must force the strategy to start by adding the following event to the interaction process diagram of the strategy. Refer to the Custom Callback Plugin for an example of the Composer plugin strategy that includes this event handling in the interaction process diagram.

```
Name = plugin start
Event = plugin.start
Body:
<log expr="'plugin.start in IPD event handler. Starting application with no interaction'"/>
 <assign location="system.SessionID" expr="_sessionid"/>
 <assign location="system.StartEvent" expr="_event"/>
 <assign location="App_StartEvent" expr="_event"/>
 <raise event="application.start">
        <param name="interactionid" expr="undefined"/>
 </raise>
```

Return the Plugin Response

By sending a request to the existing session (**<session:fetch>**), the plugin returns an asynchronous response to the Callback URL received in the _reply_url parameter of the query. The response is included in parameters named action and message. For example:

```
<session:fetch requestid="requestId"  srcexpr="_data._reply_url"
                    method="'post'" timeout="30">
            <param name="action" expr="replyAction"/>
            <param name="message" expr="replyMesg"/>
</session:fetch>
```

Action returned	Type of answer	Callback behavior
cancel	all	Callback cancels the request.
route_to_agent	human or machine	Callback routes the call to an agent. If an agent is not immediately available, the music file configured by the _treatment_waiting_for_agent parameter is played.
route_to_rp	human or machine	Callback routes the customer call to the destination specified by the message parameter. The destination can be any valid value for the destination of a Composer "Force Route" block, for example an **ACD Queue**, **Destination Label**, or **Routing Point**. The router priority and age of the call are respectively set in the GMS_URS_PRIORITY and RouterData70 attached data keys. • You can retrieve and use the GMS_URS_PRIORITY value to set the interaction priority if the interaction is added to another queue. • The RouterData70 value will automatically be used by URS to set the interaction age if the interaction is added to another queue.
retry	all	Callback dials out again unless the maximum number of retry attempts has been reached, in which case the request is cancelled.

The plugin can optionally return a customer number, a customer number prefix, or a retry time delay in seconds. If included, these values override the service configuration values.

If the callback service has the _ixn_redirect_hints parameter set to {"extensions":

{"CPNDigits": "5557777"}}, the value of the CPNDigits key is updated in the returned customer_number value. See the Callback Service Options Reference for a description of the _ixn_redirect_hints parameter.

The following example includes the optional parameters:

```
<session:fetch requestid="requestId" srcexpr="_data._reply_url"
method="'post'" timeout="30">
<param name="action" expr="'retry'"/>
<param name="message" expr="'Ok'"/>
<param name="delay" expr="300"/>
<param name="customer_number" expr="'555-2336'"/>
<param name="customer_number_prefix" expr="91"/>
</session:fetch>
```

Invoke a VXML Interface

The sequence for executing a VXML application is the same as that for the plugin detailed above, except that the VXML application is only invoked for human answers. The built-in treatment path is taken for the other cases.

The requirements for the VXML response are identical to the plugin case, but the response must be returned in an exit namelist object named treatment_result. For example:

```
<var name="treatment_result" expr="{'action':'route_to_agent', 'message': 'Ok'}"/>
…
<exit namelist="treatment_result"/>
```

Create Office Hours

> ### Important
>
> You must set up Office Hours if you want to implement **Scheduled Callback** and **Disposition Dialog**.

Upload the Office Hours Template

In the **Callback and Mobile Engagement > Service Templates** tab, make sure that the **Office Hours** template is available. Otherwise, click **Upload** to add the `office-hours.zip` to the **Templates** list.

Create the Office Hours Service

Click **Create** in **Configured Services**, select `office-hours` in the filter and configure the `business-hours` service.

Set your Calendar Timezone

The **Display Timezone** is the timezone of your **Calendar** view and is set to the Browser timezone by default. The **Office Timezone** is bound to the `_timezone` parameter of the Office Hours service.

The **Calendar** view is in Read-Only mode if your Display Timezone and Office Timezone are different.

In that case, the interface shows a warning message and does not allow you to edit the **Calendar** view.

[+] Show me the message

🛈 The calendar is in read-only mode because the office timezone is not equal to the displayed timezone

To fix this issue, click **Settings** and fix your **Office Timezone** and **Display Timezone** by setting identical timezones.

Add Office Hours to your Callback Service

Select your Callback Service and expand the **General** category. Find the `_business_hours_service` parameter and select your Office Hours instance in the drop down list.

Create Regular Office Hours

Select your Office Hours service in the **Configured Service** tab.

In the **Calendar** view, make sure that **Regular Office Hours** is checked.

Select your Office Hours service in **Configured Services**. You can add Office Hours by selecting a timezone in your **Calendar** view; the Office Hours interface shows up. Or, you can switch to the **Table** view and edit options there.

Switch Views to Check Office Hours

If you add Office Hours to one or more days in the week, the results apply to all months and weeks. Switch to the **Month** view to see the actual Office Hours of the month.

Add Holidays and Exceptions

You can set certain dates as Holidays so they will not be used for Callback and certain dates as Exceptions (or Special Office Hours) for days that require additional hours.

To add these special events, uncheck the **Regular Office Hours** parameter, then click in the **Calendar** view on the appropriate date to open the dialog box.

Exceptions Always Apply

If you set a Holiday and an Exception on the same day, then the Exception applies. The office is opened during the hours configured in the Exception.

If Regular Hours, Holiday, and Exception are set on the same day, then Holiday blocks Regular Hours, but the Exception still applies and the office is opened during the Exception's hours.

Add Holidays and Exceptions from the Table view

You can also add Holidays and Exceptions by editing the associated options in the **Table** view.

Import Rules in Office Hours

The **Import Rules** feature enables you to set up Regular Hours, Holidays, and Exceptions (Special Office Hours) that you want to apply to several Office Hours services. The imported content (options and setup) becomes part of the Office Hours service.

- Further changes to the imported services (Regular Hours, Holidays, Exceptions) instantly apply to all the Office Hours services that imported the rules.

- The **Import Rules** feature ignores the timezones defined in the imported service.

> **Tip**
> The **Import Rules** feature allows you to create and manage a single Office Hours, Holidays, and Exceptions calendar that is applicable to multiple team calendars.

Use Case: Import Holiday Rules

Let's consider that you have several offices in the United States, sharing the same federal holidays and vacations. In that scenario, you can create a dedicated Office Hours called `Federal USA Holidays`, and then import it in your U.S. offices, instead of manually adding the same holidays to your **Calendar** views for each service (`Office New York` and `Office SFO` in the example below).

Use Case: Import Exception Rules

Let's consider that, every two weeks during spring, all of your offices are opened on Wednesday evenings from 6-9 p.m. for a special event called *Special Deal*. In that scenario, you can create an Office Hours service called `Special Deal Events` where you create the needed extra hours. Then, you can import this rule in each concerned office (`Office New York`, `Office SFO`, and `Office Amsterdam` in the example below).

Note that `Special Deal Events` will apply even if Regular Hours and Holiday are set for a given date, as explained above.

Set up Rules in a Dedicated Office Hours Service

To set up these rules, create a new Office Hours service. In this example, two services are created: `Special Deal Events` and `Federal USA Holidays`.

Import Rules in Office Hours Service

Switch to **Table** view and expand **Imported Rule Calendars**. Click to add a new rule line, then select the appropriate Office Hours service.

View Imported Rules

Switch to the **Calendar** view to display the results of the imports. In this example, you can see that the `Special Deal Events` and `Federal USA Holidays` are imported in the `Office SFO` service.

If you update an Office Hour service (for instance, `Special Deal Events` or `Federal USA Holidays`), you will see these updates replicated to the Office Hours services importing the modified calendars (here, `Office SFO`).

You can also click **Settings** and see the imported Offices Hours in the list of **Displayed Calendars**.

Import Schedules in Office Hours

You can import **Schedules** from other Office Hours Services to combine working hours of multiple teams and shifts in the same Office Hours service. In this scenario, you can create a main Office Hours service (`All Offices` in the example below) into which you can import all the Office Hours schedules (Regular Hours, Holidays, Exceptions, and timezones). This results in the union of the imported Office Hours schedules within the timezone of the main Office Hours.

> ## Tip
>
> The **Import Schedules** feature helps you to manage teams dispatched in different timezones.

Import Schedules in Office Hours Service

Switch to **Table** view and expand **Imported Schedule Calendars**. Click to add a new rule line, then select the appropriate Office Hours service.

Display Imaged Schedules
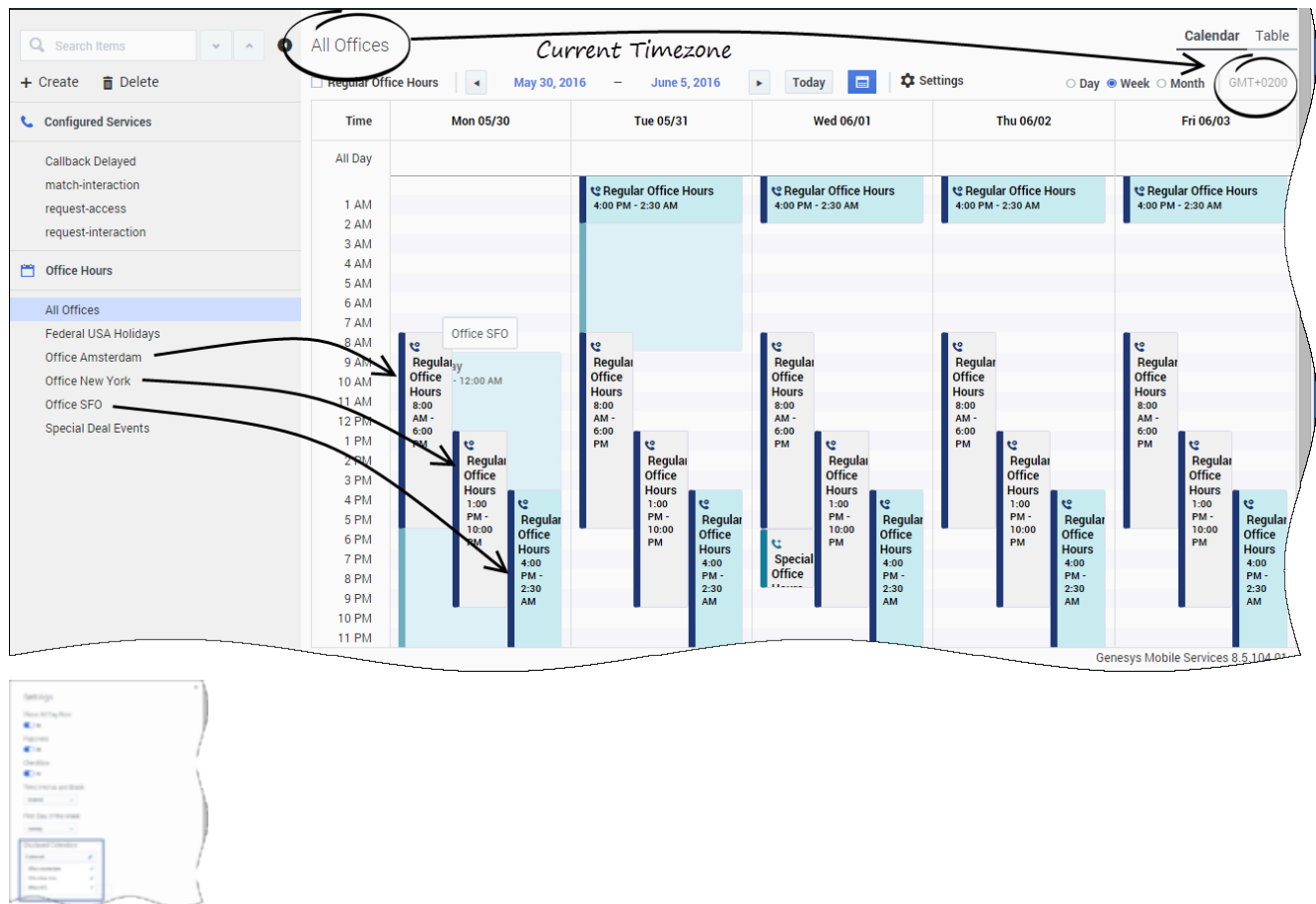
Switch to the **Calendar** view to see the results of the imports.

In this example, the `All Offices` service combines the schedules of the `Office SFO`, `Office New York`, and `Office Amsterdam` services.

The timezones, Regular Hours, Holidays, and Exceptions are aggregated in the **Calendar** view that shows the callback coverage for the three offices.

You can also click **Settings** and see the imported Offices Hours in the list of **Displayed Calendars**.

Office Hours Options Reference

_include_schedule<n>

Section: office-hours
Default Value:
Valid Values: service.<office hours ID>
Changes Take Effect: Immediately

The name of another office-hours service to include in the current service. If you need to combine the schedules of several teams, use this option to import schedules from other Office Hours Services to combine working hours of multiple teams and shifts in the same Office Hours service. In this scenario, you can create a main Office Hours service (All Offices in the example below) into which you can import all the Office Hours schedules (Regular Hours, Holidays, Exceptions, and timezones). This results in the union of the imported Office Hours schedules within the timezone of the main Office Hours.

For example, create and configure two office Hours services, one named Office SFO, the other Office Amsterdam. Then create the All Offices service and set _include_schedule parameter to combine schedules of both teams. A service referenced in _include_schedule parameter can include additional _include_rule and _include_schedule option(s) and in its configuration.

_include_rule<n>

Section: office-hours
Default Value:
Valid Values: service.<national_holidays>
Changes Take Effect: Immediately

ID of another office-hours service. All rules, such as _bh_regular<n>, _bh_add<n>, _holiday<n> from the specified service (and it's predecessors) are included into the current service. The timezone specified in the included service is ignored. Instead the timezone specified in the current service, applies to all of the rules configured in the current service.

_holiday<n>

Section: office-hours
Default Value:
Valid Values: "YYYY-MM-dd" or "MM-dd"
Changes Take Effect: Immediately

Holidays are specified either by full date (including the year), or only by month and day. In the latter case it is treated as a recurrent event, which is repeated annually. The name of the parameter should be unique within this service (section of options). For example, `2016-07-01` or `07-01`.

The parameter name should be unique within this service. For instance, _holiday1, _holiday2, and so on.

_bh_regular<n>

Section: office-hours
Default Value:
Valid Values: `DDD-DDD HH:mm-HH:mm`
Changes Take Effect: Immediately

Days and hours of the week where regular office hours apply; for example, `Mon-Fri 9:00-17:00`. The range of days can overrun the end of the week, for example, you can specify `Sat-Mon` that includes 3 days: Saturday, Sunday, Monday. The hours are specified in 24-hour format. The range of hours can overrun the end of the day, for example `19:00-7:00` to model the shift that starts at 7 pm and ends at 7 am.

The parameter name should be unique within this service. For instance, _bh_regular2, _bh_regular2, and so on.

_bh_add<n>

Section: office-hours
Default Value: No default value
Valid Values: `YYYY-MM-dd HH:mm-HH:mm` or `MM-dd HH:mm-HH:mm`
Changes Take Effect: Immediately

Specifies a special date and time period when the office will be open.

Additional days can be specified by providing either full date (including the year) or only the month and the day. For example, `2014-11-30 10:00-14:00` or `11-30 10:00-14:00`

If you do not specify the year, the event is considered as recurrent and will be repeated every year. The name of the parameter should be unique within this service (section of options). For instance, _bh_add1, _bh_add2, and so on.

_use_alternative<n>

Section: office-hours
Default Value:
Valid Values: String
Changes Take Effect: Immediately

A comma-separated list of one or more alternative `office-hours` services that should be used during specified periods instead of the current (master) `office-hours` service configuration, for example: `office-hours-summer, holidays`.
If you add an `office-hours` service to this list (for instance, `office-hours-summer`), you must configure its _effective_dates parameter. During the effective dates, the master service uses the configuration of the alternative services instead of its own configuration.
If you set more than one service in this option, the (master) `office-hours` service uses the _timezone parameter of this list's first service (`office-hours-summer` in our example) and will use the "include rule" approach to combine the effective hours.

> ### Important
> Not supported by the Service Management UI in 8.5.106.

_effective_dates<n>

Section: office-hours
Default Value:
Valid Values: `YYYY-MM-dd`
or `YYYY-MM-dd to YYYY-MM-dd`
Changes Take Effect:

Period while the `office-hours` service applies. If the service has at least one _effective_dates parameter configured, then the office is considered closed outside of the configured _effective_dates<n> ranges.
Use this parameter to restrict the service to a given period of time (a few weeks or months). For example, 2016-04-01 or `2016-04-01 to 2016-10-31`.
This parameter is mandatory if you set the _use_alternative<n> parameter.
_effective_dates<n> parameters apply to all of the parameters configured in the current office-hours service and in the other included services (for example `_include_rule, _include_schedule, _use_alternative`) services.

Important

Not supported by the Service Management UI in 8.5.106

Create Capacity Service
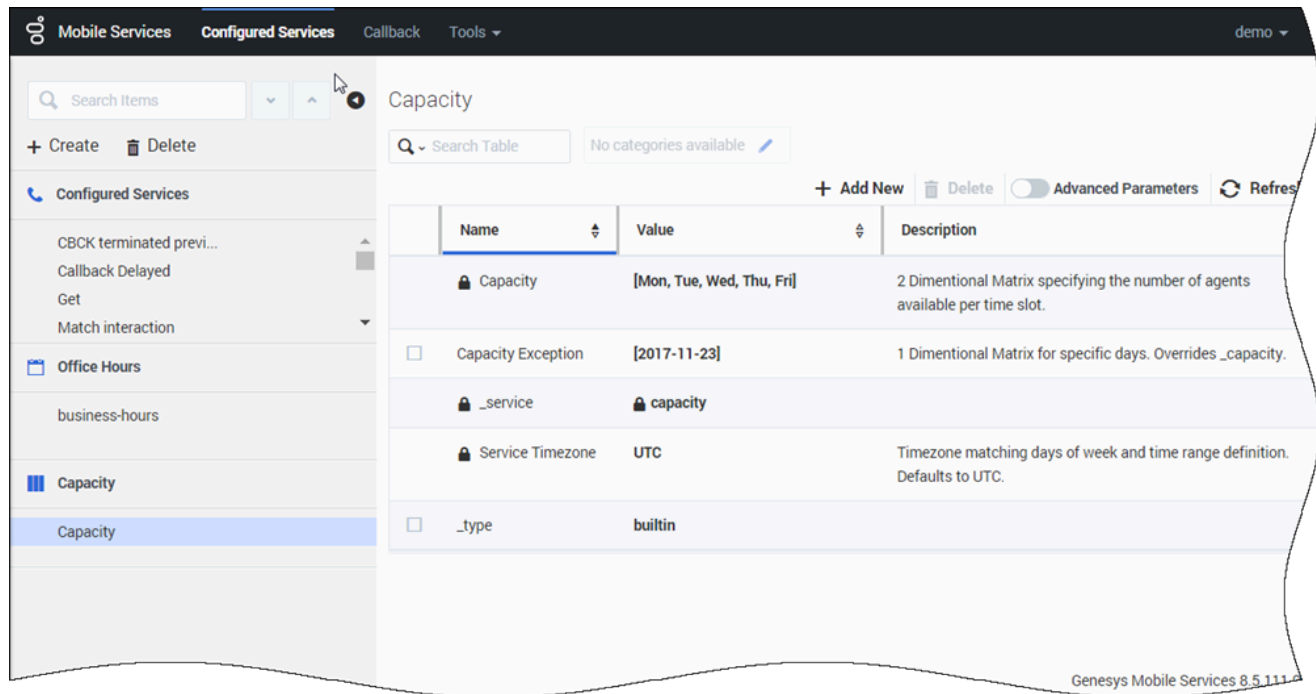
The Capacity Service enables you to define the number of scheduled callbacks that are allowed for Callback for a given time slot in the week. Then, your Callback service refers to your Capacity service and to your Office Hours service to adjust the agent availability and the number of scheduled callbacks.

You can define exceptions for dates when fewer or more scheduled callbacks are available, and you can define as many Capacity services that you need to match your Callback services.

Make sure to update the existing calendar configuration to set the correct timezone for your Capacity service. For instance, if you configured "EST", or "PST" timezones with the configuration, your parameters must use the timezones defined for Java such as "America/Toronto", or "Europe/Paris". See Wikipedia to get the list of correct timezones.

Create a Capacity Service

Create a new service and select **capacity** in the list of templates.

Once this service is created, you can use the Capacity API.

> **Important**
>
> You also need to create an Office Hours service.

Link your Capacity to your Callback Service
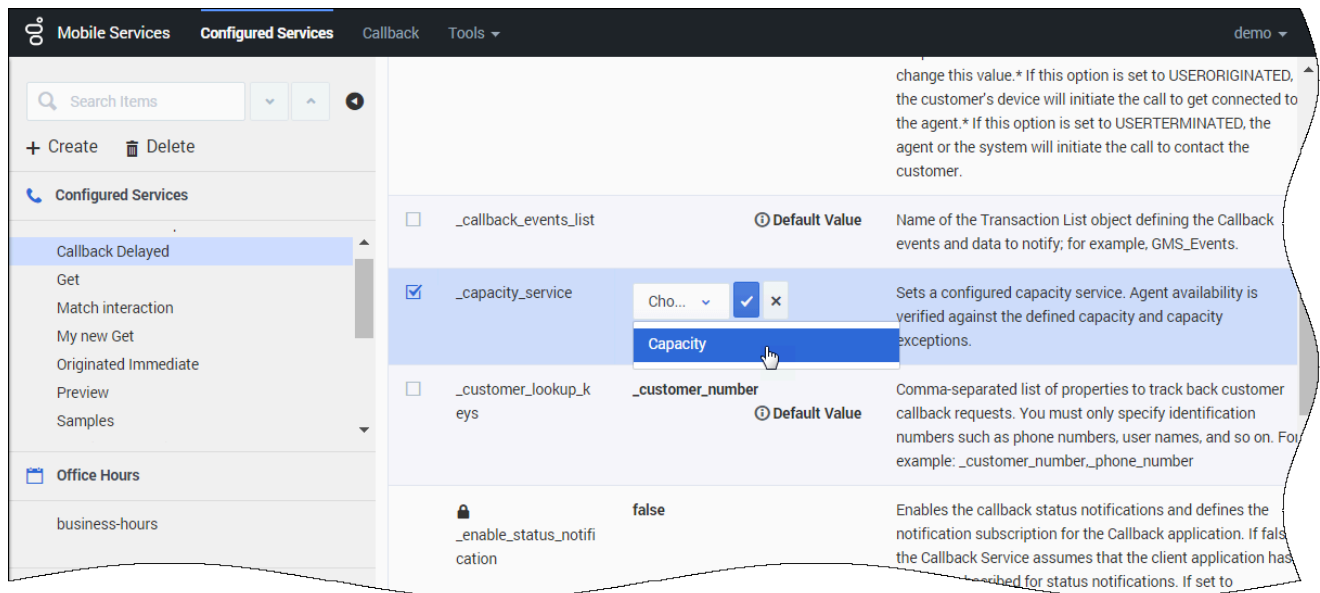
If your Callback service needs variable capacity levels, you must map its `_capacity_service` parameter value with the name of the Capacity service that you have created.

> **Important**
>
> Callback services that need fixed capacity levels can continue to use the `_max_request_by_time_bucket` option. But, if your Callback service includes both `_capacity_service` and `_max_request_by_time_bucket` options, then `_max_request_by_time_bucket` is ignored.

Add Capacity

Capacity

Q ▾ Search Table   No categories available ✏

＋ Add New   🗑 Delete   ⬤ Advanced Parameters   ↻ Ref

Name	Value	Description
🔒 Capacity	[] ✏	2 Dimentional Matrix specifying the number of agents available per time slot.
☐ Capacity Exception	{}	1 Dimentional Matrix for specific days. Overrides _capacity.
🔒 _service		
🔒 Service Timezone		
☐ _type		

Capacity ✕
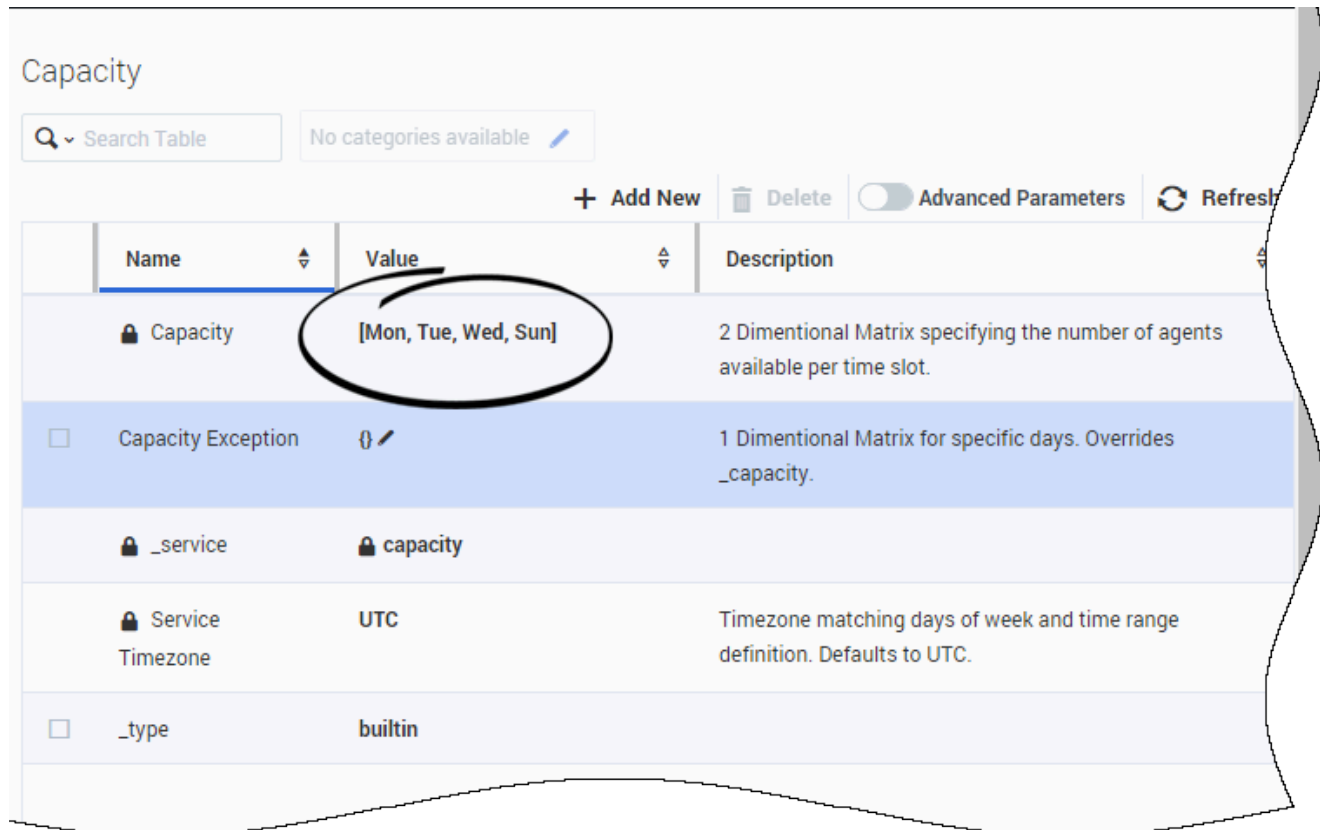
Timestamp	Sun	Mon	Tue	Wed	Thu	Fri	Sat	Set Slot
00:00	0	0	0	0	0	0	0	
01:00	0	0	0	0	0	0	0	
02:00	0	0	0	0	0	0	0	
03:00	0	0	0	0	0	0	0	
04:00	0	0	0	0	0	0	0	
05:00	0	0	0	0	0	0	0	
06:00	0	0	0	0	0	0	0	
07:00	0	0	0	0	0	0	0	
08:00	0	0	0	0	0	0	0	
09:00	5	10	10	10	0	0	0	
10:00	5	10	10	10	0	0	0	
11:00	5	10	10	│	0	0	0	
12:00	5	10	10	0	0	0	0	
13:00	5	10	10	0	0	0	0	
14:00	5	10	10	0	0	0	0	
15:00	5	10	10	0	0	0	0	
16:00	5	10	10	0	0	0	0	
17:00	5	10	10	0	0	0	0	
18:00	5	10	10	0	0	0	0	
19:00	0	0	0	0	0	0	0	
20:00	0	0	0	0	0	0	0	
21:00	0	0	0	0	0	0	0	
22:00	0	0	0	0	0	0	0	
23:00	0	0	0	0	0	0	0	
Set Day								

Cancel   Reset   Save

Edit the **Capacity** value to open the Capacity grid widget. Enter your capacity per Day or Hours, then **Save**.

Capacity

Name	Value	Description
🔒 Capacity	[Mon, Tue, Wed, Sun]	2 Dimentional Matrix specifying the number of agents available per time slot.
☐ Capacity Exception	{} ✎	1 Dimentional Matrix for specific days. Overrides _capacity.
🔒 _service	🔒 capacity	
🔒 Service Timezone	UTC	Timezone matching days of week and time range definition. Defaults to UTC.
☐ _type	builtin	

You can see for which days of the week the Capacity service is defined.

Add Exceptions

For more flexibility, you can set exceptions for the Capacity service. You can enter dates with different capacities for federal holidays, vacations, and so on.

Edit the **Capacity Exception** value to open the interface.

You can add as many exceptions as you need. You can also even update later to modify your capacity.

Click **Save** to update your Capacity data. A popup message displays the operation result.

Display Sub-capacities and Sub-exceptions

To display the list of sub-capacities and sub-exceptions, enable **Advanced Parameters**.

Exception Patterns

Genesys Callback allows implementing patterns to be checked before performing the callback attempt. The **Patterns** tab enables you to configure exceptions; for example, phone numbers. You can define as many patterns as you wish, then add them to your Callback services.

Create a Pattern Group
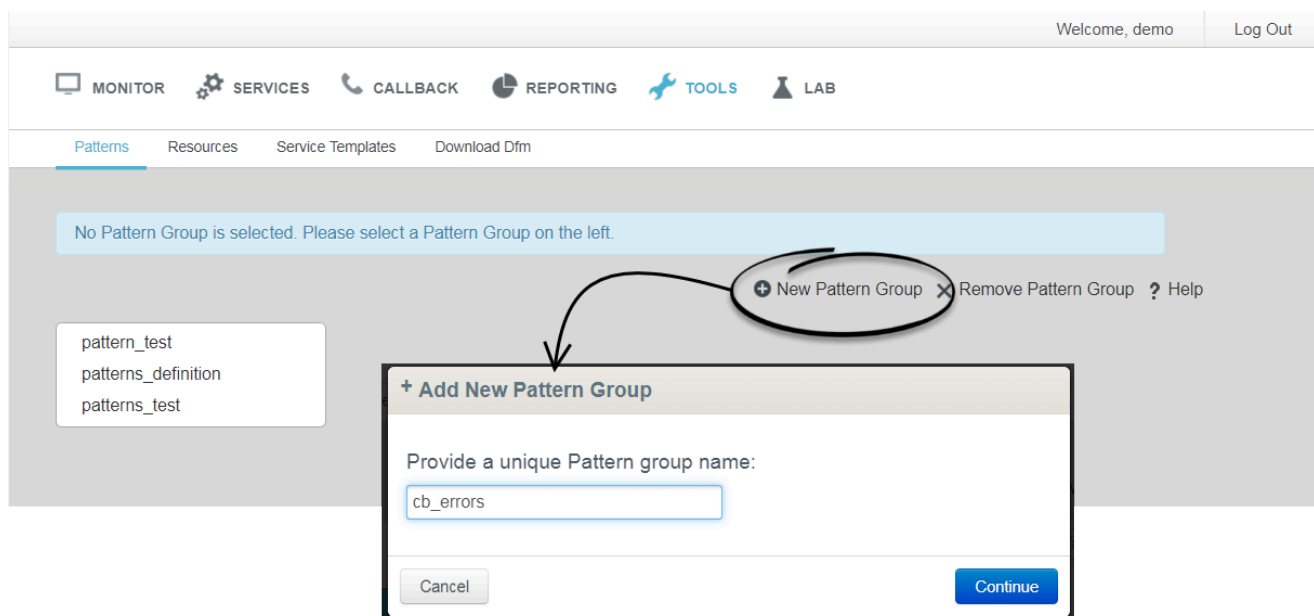
Navigate to **Admin UI > Tools > Patterns**. Click **New Pattern Group** to define a new group of patterns. Enter a name.

Create an Exception Pattern

Select a group. Then, in this group, click **Add Pattern** to create as many patterns than you need.

Test your Exception Pattern

Select a group. You can test the value of a pattern against this entire group. Type a value in the input field, and if a match is found, the corresponding table row will be highlighted.

Add the Pattern Group to your Callback Service
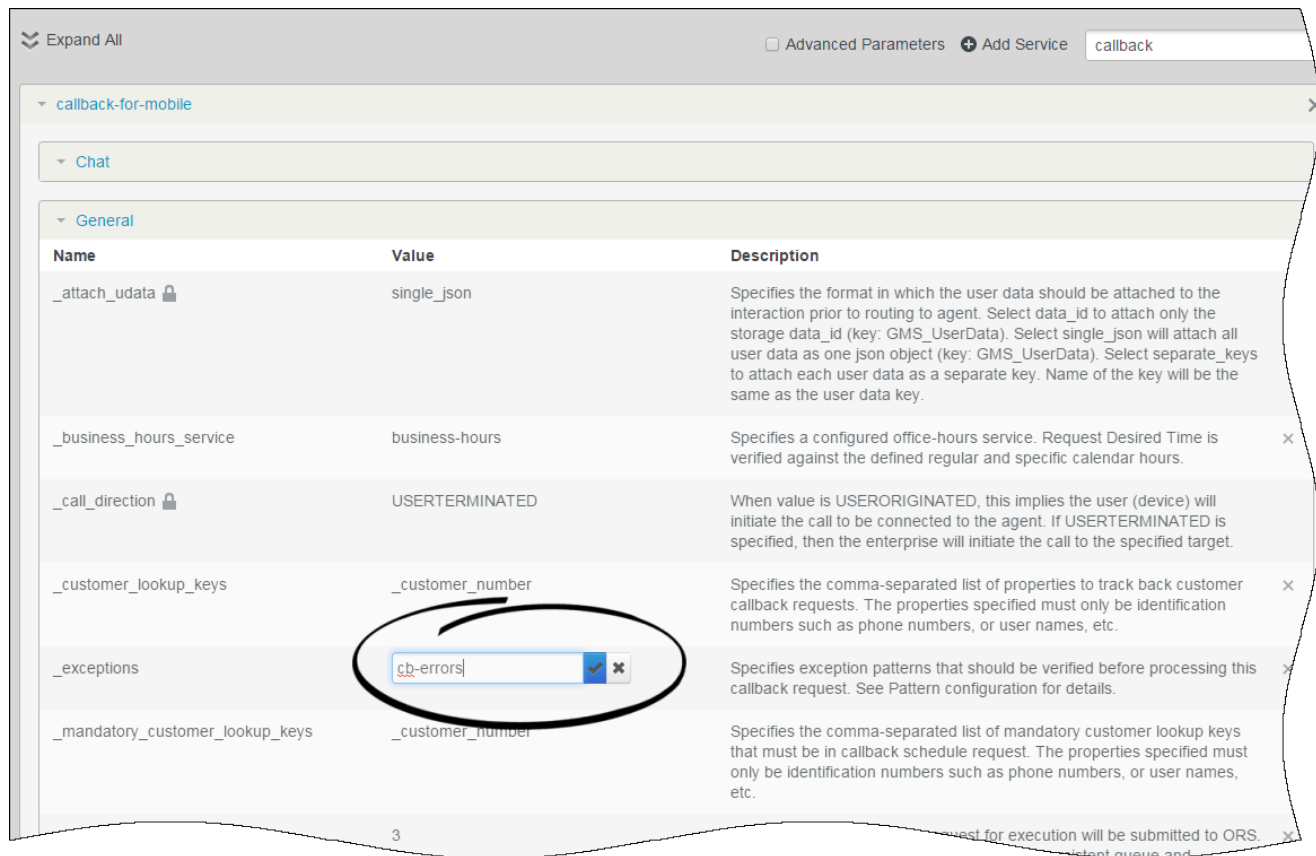
Navigate to **Admin UI > Services** and expand your Callback service. In the **General** section, set the name of your Pattern Group as the value of the _exceptions parameter.

How to get Help about Supported Patterns?

A regular expression defines a search pattern for strings. Regular expressions can be used to search, edit and manipulate text.	
Regular Expression	**Description**
.	Matches any character
^regex	Regex must match at the beginning of the line
regex$	Regex must match at the end of the line
[abc]	Set definition, can match the letter a or b or c
[abc] [vz]	Set definition, can match a or b or c followed by either v or z
[^abc]	When a "^" appears as the first character inside [] when it negates the pattern. This can match any character except a or b or c
[a-d1-7]	Ranges, letter between a and d and figures from 1 to 7, will not match d1
X|Z	Finds X or Z
\\([0-9][0-9][0-9]\\)[0-9][0-9][0-9]-[0-9][0-9][0-9][0-9]	Telephone Number in the format of (XXX)XXX-XXXX
[0-9a-zA-Z]+@[0-9a-zA-Z]+\\.[0-9a-zA-Z]+	Simple Email Format: X@X.X where X can be any number or letters
[A-Z][0-9][A-Z]\\ [0-9][A-Z][0-9]	Canada postal code: A0A 0A0

For more details please refer to Java Regular Expression Lesson by Oracle

Close

Click the *Help* button to get more information about the supported patterns. For additional details, refer to Oracle Java Lesson: Regular Expressions.

Enable Status Notifications

Modified in 8.5.107

Starting in 8.5.105, the Callback service can publish notifications to GMS that distributes these notifications to the target specified in the callback's service request, and consequently, to the subscribers of these notifications. The possible targets can be an ORS session of an existing GMS service (`orscb` notification type) or any URL (`httpcb` notification type).

Starting in 8.5.107, you can receive two types of notifications: Callback SCXML and additional GMS Callback notifications.

> ### Important
> By default, this feature is turned off for all callback services.

To enable Callback Status Notifications (SCXML), you can either:

- Enable the Default Status Notifications (from SCXML).
- Create a Transaction Event object that overwrites the list of default notifications and assign it to your Callback Service. You can configure additional GMS Callback Status notifications by using the Transaction List entries which override the defaults. In that scenario, the notifications will only report the events specified in this Transaction List.
- Add notifications parameters to your Callback Services query.

Callback will send the notification events and provides two subscription modes to receive them:

- `subscribe_notify`—Callback subscribes for your application to the notifications.
- `notify`—Your application must subscribe to receive events.

Enable Default Status Notifications in a Callback Service

To receive default callback status notifications (SCXML), open the Service Management User Interface and navigate to your Callback Service (in the **Configured Services** panel).

Enable **Advanced Parameters** and configure the following options in the **General** section:

- _enable_status_notification= subscribe_notify
- _status_notification_type= httpcb (or orscb)
- _status_notification_target = Target URL (or the ORS session id if _status_notification_type = orscb)

You can add the following additional parameters to your Callback queries:

- _status_notification_debug = false—Set to true to enable the debug mode for notification.
- _status_notification_language = <language> where the language matches one of the supported languages used for push notifications.

The _status_notification_debug option defines the URL where the notifications will be pushed using HTTP POST requests.

> ### Tip
> The `orscb` notification type should be used for advanced ORS customization only.

Overwrite Default Notifications with a Transaction List

Start by defining a Transaction List object that includes the notifications and the associated events triggering notifications.

Open Genesys Administrator. In `PROVISIONING > Routing/eServices > Transactions`, click **New** to create the `GMS_Events` list.

In the **Options** tab, create a `properties` section with:

- `_enable_status_notification = subscribe_notify`
- `_status_notification_provider = <customerprovider>` or blank for default provider
- `_status_notification_type = httpcb`
- `_status_notification_target=<Target URL>`

- `_status_notification_debug= false`
- `_status_notification_language = <language>` where the language matches one of the supported languages used for push notifications.

Then, create a section for each subscribed event and define the data that your application needs to receive in the notification event.

- `notify_params`—The comma-separated list of callback parameters to retrieve. See the reference to get the list of parameters that can be retrieved.
- `notify_custom`—(Optional) A JSON object of the custom attached data to send in the notification in addition to the callback parameters set in `notify_params`.

> ## Tip
>
> - Either click **New** to add the following options or copy the source below to a `GMS_Events.cfg` file that you can import in your Transaction List.
> - You do **not** have to include all the events listed below.
> - The `notify_custom` parameter should suite your use case or can be removed if not needed.

In the XML sample below, `c_target` must match `c_target` as provided by URS.

```
[properties]
_enable_status_notification = notify
_status_notification_provider =
_status_notification_type = httpcb
_status_notification_target =<your URL>

[_cbe_on_service_create]
notify_params = _service_id, _service_name, _customer_number, _urs_virtual_queue
notify_custom = {"name1":"value1", "name2": "value2"}

[_cbe_on_virtual_ixn_create]
notify_params = _service_id, _service_name
notify_custom = {"name1":"value1", "name2": "value2"}

[_cbe_on_target_found]
notify_params = _service_id, _service_name, c_target, _urs_virtual_queue
notify_custom = {"name1":"value1", "name2": "value2"}

[_cbe_on_dial_init]
notify_params = _service_id, _service_name, _customer_number, c_dialed_number
notify_custom = {"name1":"value1", "name2": "value2"}

[_cbe_on_dial_done]
notify_params = _service_id, _service_name, _customer_number, c_dialed_number, c_call_result, c_call_num_attempt
notify_custom = {"name1":"value1", "name2": "value2"}

[_cbe_on_connect_treatment_start]
notify_params = _service_id, _service_name, _customer_number, _vq_for_outbound_calls, c_dialed_number
notify_custom = {"name1":"value1", "name2": "value2"}

[_cbe_on_customer_queued]
notify_params = _service_id, _service_name, _customer_number, _vq_for_outbound_calls, c_dialed_number
notify_custom = {"name1":"value1", "name2": "value2"}

[_cbe_on_route_to_agent]
notify_params = _service_id, _service_name, _customer_number, _urs_virtual_queue, c_agent_id,
c_agent_extension
notify_custom = {"name1":"value1", "name2": "value2"}

[_cbe_on_service_exit]
notify_params = _service_id, _service_name, _customer_number, c_last_dialed_number, c_termination_type
notify_custom = {"name1":"value1", "name2": "value2"}
```

```
[_cbe_on_callback_scheduled]
notify_params=_customer_number,_phone_number,_desired_time
notify_custom={"state":"scheduled"}

[_cbe_on_callback_rescheduled]
notify_params=_customer_number,_phone_number,_desired_time
notify_custom={"state":"rescheduled"}
```

Add the Event Transaction List to the Callback Service

Edit the **Advanced Parameters** in the **General** section of your Callback Service.

Set the _callback_events_list to the name of the Transaction List created above, GMS_Events in our example.

> ## Important
>
> If you set other status notification parameters (_status_notification_type, _status_notification_target, _status_notification_provider) in your Callback service configuration or in your REST queries, they override the values set in the Transaction List object.

Callback Status Notifications Events

Callback notifications consist in a JSON object that contains:

- deviceId—The custom id provided at subscription time by the subscriber.
- message—The notification message as defined in the Callback Events Transaction List.

The following JSON code is an event sample.

```
{
        "event_id": "_cbe_on_service_create",
        "timestamp": "1467575991",
        "_service_id": "445-20e740d3-8458-43d6-834d-3713c3385bac",
        "_service_name": "samples_dev",
        "_callback_state": "QUEUED",
        "_customer_number": "5115",
        "_urs_virtual_queue": "SIP_VQ_SIP_Switch",
        "name1": "value1",
        "name2": "value2"
}
{
        "event_id": "_cbe_on_virtual_ixn_create",
        "timestamp": "1467575992",
        "_service_id": "445-20e740d3-8458-43d6-834d-3713c3385bac",
        "_callback_state": "QUEUED",
        "name1": "value1",
        "name2": "value2"
}
{
        "event_id": "_cbe_on_dial_init",
        "timestamp": "1467575992",
        "_service_id": "445-20e740d3-8458-43d6-834d-3713c3385bac",
        "_service_name": "samples_dev",
        "_callback_state": "QUEUED",
        "_customer_number": "5115",
        "c_dialed_number": "5115",
        "name1": "value1",
        "name2": "value2"
}
{
        "event_id": "_cbe_on_dial_done",
        "timestamp": "1467576012",
        "_service_id": "445-20e740d3-8458-43d6-834d-3713c3385bac",
        "_service_name": "samples_dev",
        "_callback_state": "QUEUED",
        "_customer_number": "5115",
        "c_dialed_number": "5115",
        "c_call_result": 0,
        "c_call_num_attempt": 1,
        "name1": "value1",
        "name2": "value2"
}
{
        "event_id": "_cbe_on_connect_treatment_start",
        "timestamp": "1467576012",
        "_service_id": "445-20e740d3-8458-43d6-834d-3713c3385bac",
        "_service_name": "samples_dev",
        "_callback_state": "QUEUED",
        "_vq_for_outbound_calls": "VQ_GMS_REP_SIP_Switch",
        "c_dialed_number": "5115",
```

```
        "name1": "value1",
        "name2": "value2"
}
{

        "event_id": "_cbe_on_customer_queued",
        "timestamp": "1467576016",
        "_service_id": "445-20e740d3-8458-43d6-834d-3713c3385bac",
        "_service_name": "samples_dev",
        "_callback_state": "QUEUED",
        "_vq_for_outbound_calls": "VQ_GMS_REP_SIP_Switch",
        "c_dialed_number": "5115",
        "name1": "value1",
        "name2": "value2"
}
{

        "event_id": "_cbe_on_target_found",
        "timestamp": "1467576016",
        "_service_id": "445-20e740d3-8458-43d6-834d-3713c3385bac",
        "_service_name": "samples_dev",
        "_callback_state": "QUEUED",
        "_urs_virtual_queue": "SIP_VQ_SIP_Switch",
        "c_target": {
                "agent": "KSippola",
                "dn": "7001",
                "id": "Customer_Service",
                "place": "SIP_Server_Place1",
                "resource": "7001",
                "return": "target",
                "stat_value": "0",
                "switch": "SIP_Switch",
                "type": "GA",
                "vq": "SIP_VQ_SIP_Switch"
        },
        "name1": "value1",
        "name2": "value2"
}
{

        "event_id": "_cbe_on_service_exit",
        "timestamp": "1467576291",
        "_service_id": "445-20e740d3-8458-43d6-834d-3713c3385bac",
        "_service_name": "samples_dev",
        "_callback_state": "QUEUED",
        "c_termination_type": "COMPLETED.AGENT_CONNECTED",
        "name1": "value1",
        "name2": "value2"
}
```

Reference for Notification Events

Event Name	When this event is triggered	List of attributes that you can retrieve
_cbe_on_service_create	As soon as the callback service (ORS session) is started.	_service_id _service_name _customer_number _urs_virtual_queue
_cbe_on_virtual_ixn_create	When the virtual interaction is successfully created in URS.	_service_id _service_name
_cbe_on_target_found	When the callback has found the target and URS reports the target to ORS.	_service_id _service_name _urs_virtual_queue c_agent_id c_agent_extension
_cbe_on_dial_init	When the dialing to the customer is started. Note: This behavior applies for both standard and preview callback.	_service_id _service_name _customer_number c_dialed_number
_cbe_on_dial_done	When the dialing result is known.	_service_id _service_name _customer_number c_dialed_number c_call_result c_call_num_attempt
_cbe_on_connect_treatment_start	When the greeting treatment is started right after the successful CPD.	_service_id _service_name _vq_for_outbound_calls c_dialed_number
_cbe_on_customer_queued	In User Terminated scenarios, as soon as the onconnect treatment is over, the virtual interaction becomes routable and the customer is	_service_id _service_name

Event Name	When this event is triggered	List of attributes that you can retrieve
	placed into a queue to wait for an agent.	_vq_for_outbound_calls c_dialed_number
_cbe_on_route_to_agent	When the call is transferred from Routing Point to the agent.	_service_id _service_name c_agent_id c_agent_extension
_cbe_on_service_exit	In all exit scenarios.	_service_id _service_name c_dialed_number c_termination_type
_cbe_on_callback_scheduled	When a callback in SCHEDULE status is created.	_service_id _service_name _desired_time _customer_number _v_queue
_cbe_on_callback_rescheduled	When a callback in SCHEDULE status is re-scheduled.	_service_id _service_name _desired_time _customer_number _v_queue
_cbe_on_callback_cancelled	When the callback is canceled.	_service_id _service_name _desired_time _customer_number _v_queue
_cbe_on_callback_status_updated	When the _callback_state field is updated by a REST query. This can be due to ORS updates.	_service_id _service_name _desired_time _customer_number _v_queue
_cbe_on_callback_submitted	When the callback is submitted for ORS execution.	_service_id

Event Name	When this event is triggered	List of attributes that you can retrieve
		_service_name _desired_time _customer_number _v_queue
_cbe_on_callback_resubmitted	When the callback is re-submitted for ORS execution.	_service_id _service_name _desired_time _customer_number _v_queue
_cbe_on_callback_submit_failed	When submit for execution fails.	_service_id _service_name _desired_time _customer_number _v_queue
_cbe_on_callback_processing_failed	When the callback processing fails.	_service_id _service_name _desired_time _customer_number _v_queue
_cbe_on_callback_queued	When the callback is successfully submitted and its state changed to QUEUED.	_service_id _service_name _desired_time _customer_number _v_queue

Configure Agent Reject

To ensure that agents will be able to reject callbacks, Genesys recommends to configure the following values by using the Genesys Administrator Extension to edit your configuration.

1. In your T-Server application:
    ```
    divert-on-ringing=false

    after-routing-timeout=30
    ```

2. In the **gts** section of your ORS application:
    ```
    cti-transaction-timeout=45
    ```

3. In the **General** section of your Callback service:
    ```
    _agent_transfer_confirm_timeout=0
    ```

Get EWT for Callback

Modified in 8.5.109

In your Callback service, you can configure some options to specify how Callback should query Estimated Wait Time.

- See set up an EWT method for Callback.

To retrieve the Expected Waiting Time (EWT), you have three possibilities:

- Directly query URS.

- Use the EWT API for virtual queues (added in 8.5.109).

- Use the urs-stat Service.

Set up an EWT method for Callback

In your Callback service, you can configure some options to specify how Callback should query Estimated Wait Time. The following options are used to define when a request should be submitted to the Callback Orchestration execution service. You can either:

- Set the _request_queue_time_stat parameter to provide the Stat Server statistics. For example, you can use the ExpectedWaitTime statistic to set this option:
  "ExpectedWaitTime;Queue;8999@SIP_Server;Environment"

- Configure the Callback service option named `_request_ewt_service` to specify the `urs-stat` service to use to retrieve EWT value from URS stats.

- Configure the option _urs_virtual_queue in your Callback service to be able to use the new REST API as detailed below.

Query URS

To retrieve the Expected Waiting Time (EWT) for callback, you can use the lvq method of the URS Web API and directly query the following URS URL:

`http://<urshost>:<ursport>/urs/call/max/lvq?name=VQ_Name&aqt=urs&tenant=TenantName`

To view additional URS `lvq` input parameters and output information, open a browser with URS running and run the help method for `lvq` as follows:

`http://<urshost>:<ursport>/urs/help/call/lvq`

The help method is described in the Universal Routing 8.1 Reference Manual, Appendix C, "Supported Methods."

Use the Callback EWT API for Virtual Queues

Added in 8.5.109
You can query Estimated Wait Time statistics if you configure a Virtual Queue for your Callback service by using the service option _urs_virtual_queue. Then, you can use the new REST API, Query EWT, as detailed in the Stat Service API page.

> ### Important
> In your GMS configuration, add a connection to an active URS to enable this service.

Use the urs-stat Service

Create a GMS built-in service using the **urs-stat** template that provides the following benefits:

- Statistics caching of the statistic to reduce load on URS. The _caching_policy parameter sets the cache period in seconds (see below).

- Load balancing and scaling across multiple GMS nodes.

- A single point of contact for your app.

Create a urs-stat Service

To create this GMS built-in service, select the **urs-stat** template when creating a new service.
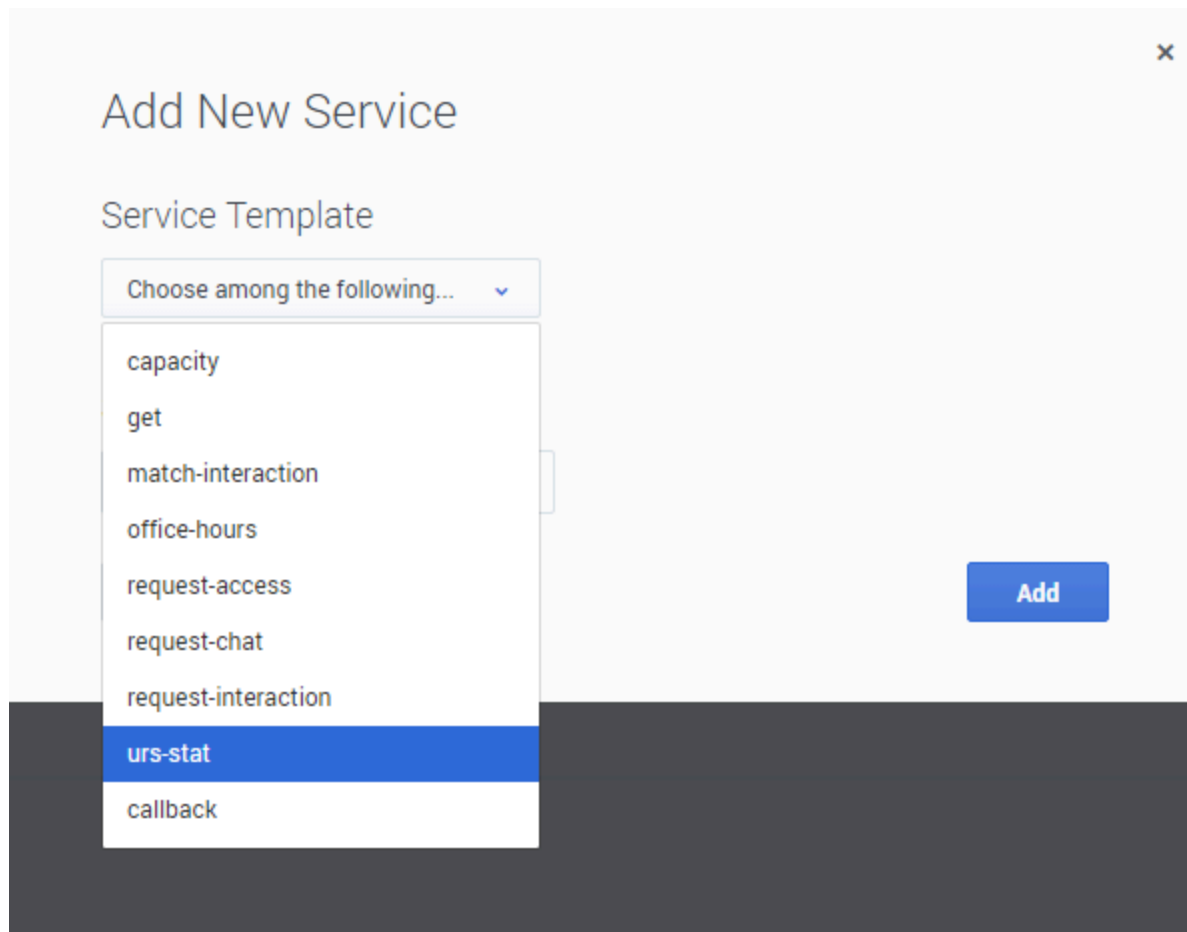
Configure urs-stat parameters

Configure the following parameters in your `<name-of-urs-stat-service>` service:

```
_caching_policy=30 # Cache refresh time in seconds
_service=urs-stat
_type=builtin
_urs_stat_url_parameters=name=<VQ_Name>&tenant=<Tenant_Name>&aqt=urs
_urs_url=http://<urshost>:<ursport>/urs/call/max/lvq
```

Where: VQ_Name, Tenant_Name, `urshost`, and `ursport` match the environment and Callback service's Virtual Queue (VQ). The following screenshot shows the creation and configuration of the **my-urs-stat** service.

> **Important**
>
> The `_urs_url` option can point to the load balancer in front of the URS that should be configured as part of the GMS provisioning steps in that scenario.

Query EWT using the urs-stat service

The following query example shows the resulting response that you get when you call the service:

```
GET http://<gmshost>:<gmsport>/genesys/1/service/<name-of-urs-stat-service>

Response:
{ "wcalls" : 20, "wpos" : 21, "time" : 1467922222, "hit" : 95, "calls" : 20,
 "wt" : 0, "ewt" : 300, "pos" : 21, "aqt" : 300 }
```

> **Important**
>
> - The value of interest here is ewt: the time unit is seconds and can be a float value.
>
> - An empty object will be returned if there is no activity for the VQ.

You can use a single service for multiple VQs by omitting the _urs_stat_url_parameters option from the service and including the value for that option (for example, name of virtual queue, tenant ID, or statistical method) in the HTTP request as follows:

```
http://<gmshost>:<gmsport>/genesys/1/service/<name-of-urs-stat-service>
?name=<one-of-the-callback-VQs>&tenant=<tenant-name>&aqt=urs
```

The URS stat service will append the content of the _urs_stat_url_parameters option and the HTTP request parameters to the URS query. To view additional URS lvq input parameters and output information, open a browser with URS running and run the help method for lvq as follows:

```
http://<urshost>:<ursport>/urs/help/call/lvq
```

The help method is described in the Universal Routing 8.1 Reference Manual, Appendix C, "Supported Methods."

If, for example, you set the following configuration for the <name-of-urs-stat-service> service:

```
_caching_policy=5
_service=urs-stat
_type=builtin
_urs_stat_url_parameters=scale=true&tenant=Environment&aqt=urs
_urs_url=http://<ursloadbalancer>:<ursport>/urs/call/max/lvq
```

You can can use this service for multiple VQs by specifying only the name of a virtual queue in the HTTP request as follows:

```
http://<gmshost>:<gmsport>/genesys/1/service/<name-of-urs-stat-service>
?name=<one-of-the-callback-VQs>
```

Callback Scenarios and Configuration

This chapter details the scenarios that you can implement based on the provided templates and options.

- **User Terminated Immediate**: The customer requests that an agent immediately makes a callback.
- **User Terminated Delayed**: The customer delays the callback and an agent will call him.
- **User Terminated Scheduled**: The customer schedules the callback and an agent will call him.
- **User Terminated Delayed Agent Preview**: The customer schedules the callback and an agent will preview the callback before calling.
- **Capacity**: You define the number of agents that are available for Callback for a given time slot during the week.
- **IVR Classic Callback**: The IVR handling an inbound call has logic to check for a long waiting time and offers to call back the caller.
- **Preview and Disposition Scenarios**: Integrate the Preview and Disposition scenarios to your Callback application.

Additional User Originated scenarios are covered in GMS Documentation. See User Originated Delayed and User Originated Immediate.

Also consider reading the Universal Routing Voice Call Back White Paper that details two types of Voice Call Back (VCB) solutions managed by Universal Routing Server (URS):

- Pre-emptive Agent Reservation
- Dialing Notifications

Learn about the Callback Templates

Callback provides a set of templates that you can use to create your customized workflow. Templates are already loaded at the product installation and show up in the **Templates** panel.
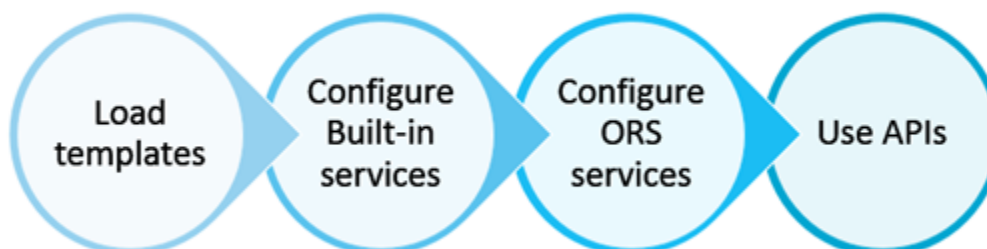
Templates include two types of services:

- Built-in services of type `builtin` that are basic services executed in the Genesys Mobile Services server. They provide fixed functionality that you can tune only through configuration options in the **Configured services** section.

- Orchestration Server-based (ORS) services of type `ors` that implement ORS scenarios. The implemented scenarios depend on the ORS service configuration.

These templates enable you to create Callback services that provide the GMS APIs.

Note that `ors` and `builtin` services may implement some cross-dependencies.

> ## Important
>
> The Callback services are executed in the Orchestration Server, and managed in the Genesys Mobile Services server. See the Scenarios section of the *Callback User's Guide* for more information.

Relationship between Configured Services and API queries

When you create a new `{service-execution-name}` service in the **Configured Services** section of the Service Management UI, this service is also created in the `service.{service-execution-name}` section of your GMS configuration.

The URLs used by the Service API are dependent on the execution name of the service that you have just created. Services are available at the following URL:

```
http://<host>:<port>/genesys/1/service/{service-execution-name}
```

For instance, if you create a service named `match-interaction`, then `{service-execution-name}` is `match-interaction` and the service is available at:

```
http://<host>:<port>/genesys/1/service/match-interaction
```

> **Important**
>
> To use a service, you should start by allocating resources to this service with a create service request. Note that for some `builtin` services, this may not be necessary.

Customized ORS Scenarios
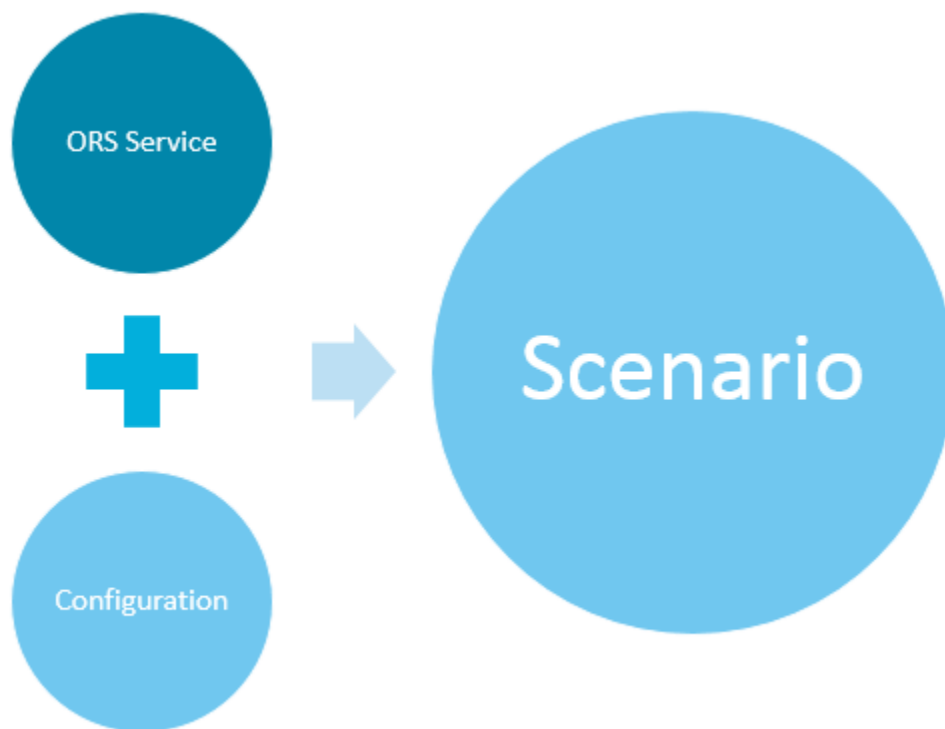
To implement a customized ORS scenario, you will need to **create** an ORS service and select the **Callback** template.

Choose your scenario in the list that the template displays. Then, refer to the pages of this chapter for configuration details and sequence diagrams.

[+] See the list.

Advanced Customization

If you are an advanced user of Composer, you can customize the SCXML and VXML of the Classic Callback sample that includes a Composer project.

Services Cross-Dependencies

According to the services that you plan to use, you may need to create and configure the following services in the **Configured Services** panel, even for services of type `builtin`.

- *request-access* — Use the `request-access` template to create a **request-access**.

- *match-interaction* — Use the `match-interaction` template to create a **match-interaction** service.

The following table presents the builtin services and the ors scenario that require one of these services (or both) before you can start using them.

Builtin or Scenario name	request-access	match-interaction
match-interaction	✓	
request-interaction	✓	✓
User Terminated Delayed Voice (notification)	✓	✓
User Terminated Immediate Voice ORS Service	✓	✓
User Terminated Scheduled Voice	✓	✓
User Terminated Delayed Voice Agent Preview	✓	✓

User Terminated Immediate

Call flow
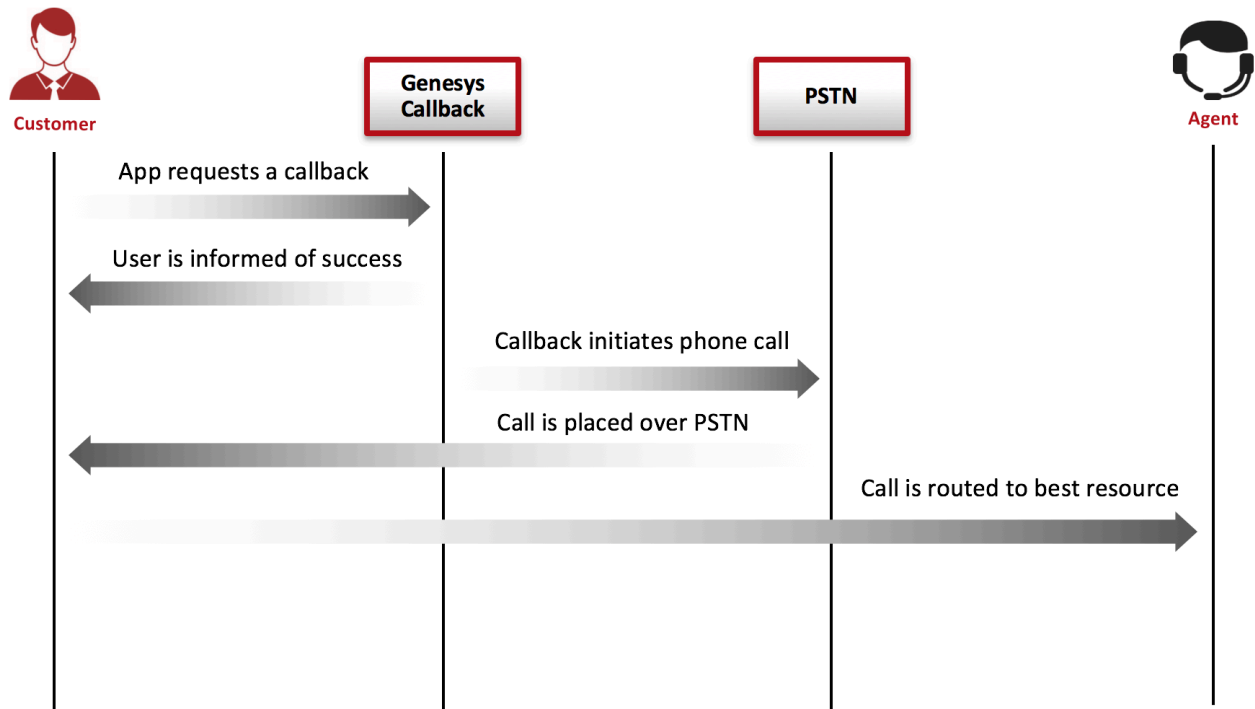
In this scenario, the customer requests an immediate callback, that is, as soon as the agent is available. This Callback scenario is an outbound voice service that goes through the following stages:

Start Callback
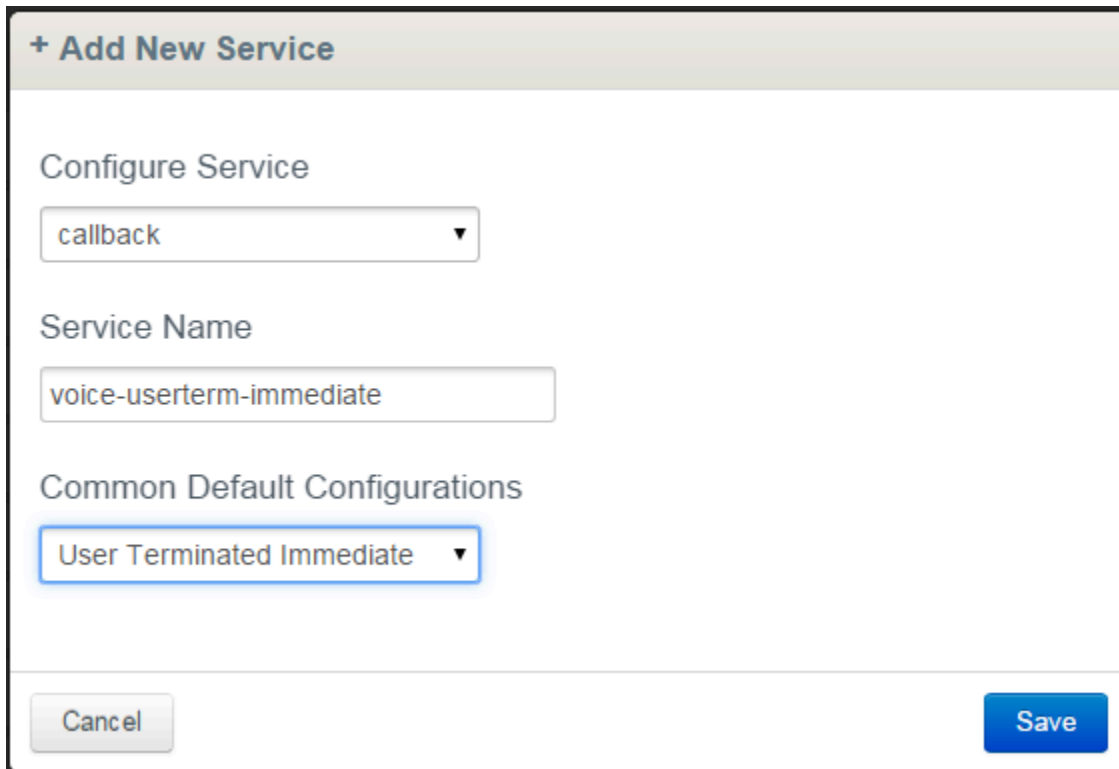
- Callback service: Returns a message to expect a call, immediately to the mobile device.

- Callback service: Calls the mobile device.

- Mobile device: Accepts the call.

- Callback service: Identifies that a human has answered the call.

Connect to Agent

- Callback service: Plays treatment until the target is available.

- Callback service: Reserves target to route call.

- Callback service: Routes the call to the target agent.

- Callback service terminates.

Create your Scenario

In the **Admin UI > Services > Configured Services** tab, add a Callback service with User-Terminated-Immediate as the **Common Default Configuration** (see Adding a Service for details).

Enter a service name. This name is the callback execution name of your service and will be used in URLs to access this service. For example, if you set this name to `voice-userterm-immediate`, your service URL will be:
`http://host:port/{base-web-application}/service/callback/voice-userterm-immediate`

When you add this service and default configuration, many options are automatically populated with the appropriate default values.

Configuration Options

The table below lists the key options applicable to this scenario. Some options, however, will require

you to enter your own values. See the **Description** column in the following table for these details.

Predefined Values

These are the default values, which are automatically populated when using the pre-defined User-Terminated-Immediate service. You do not need to change these values.

Option	Description
_media_type=voice	Media type of the interaction that the service is expected to handle. This option enables URS to select an agent who has the appropriate media capabilities. This is a default value, automatically populated when using the predefined User-Terminated scenario. You do not need to change this value. **This option is mandatory.**
_wait_for_agent = false	True to wait for an agent to connect. If this option is set to true, the service will wait for the agent to initiate the interaction and to send the notification to the customer. If the option is set to false, the interaction can start right after the creation of the service instance. In voice scenarios, the access information will be returned immediately with the service ID. **This option is mandatory.**
_wait_for_user_confirm = false	True to wait for confirmation of the customer's availability. If this option is set to true, the service sends a push notification to the customer's device to get confirmation that the customer is ready to have a conversation with the agent. This scenario is possible only if the _wait_for_agent option is set to true.
_max_transfer_to_agent_attempts = 5	Maximum number of attempts to transfer the call to the agent. If greater than 1, set the URS option on_route_error=try_other.
_call_direction = USERTERMINATED	This is a default value, automatically populated when using the predefined User-Terminated scenario. You do not need to change this value. • If this option is set to USERORIGINATED, the customer's device will initiate the call to get

Option	Description
	connected to the agent. • If this option is set to USERTERMINATED, the agent or the system will initiate the call to contact the customer.
_userterminated_first_connect_party = CUSTOMER	First party to connect when _call_direction is set to USERTERMINATED. Set this option to CUSTOMER to call the customer first; set this option to AGENT to call the agent first. **This option is mandatory.**
_ttl = 86400	Duration (in seconds) for which the service will be kept in storage after the Desired Time is passed (Time To Live). Once expired, the service is removed from the system. For example, if you want the callbacks to be visible in the Service Management UI for one week past the execution time, then you should set 7 days of Time To Live, which means _ttl=604800. **This option is mandatory.**
_type = ors	• For Genesys Mobile Services-based services: builtin • For Orchestration Server-based services: ors
_provide_code= false	If true, returns a randomly generated code to be used for the authentication of the user originated (inbound) call. **This option is mandatory.**
_use_debug_push_certificate = false	Use debug certificates for the push notification provider

Additional Required Options

You must enter a string value for the following options:

Option	Description
_route_point= "{Route Point}@{Telephony Switch}"	Route point from which the system can create a user-terminated (outbound) call. **This option is mandatory.**
_resource_group="{name of the resource pool configured under Transactions/GMS_Resources/Annex}"	Resource group from which access number is to be allocated. **This option is mandatory.**
_urs_virtual_queue = "MyVirtualQueue"	Virtual queue (alias) to which the service request will be added.
_target = "MyTarget@StatServer.GA"	Routing target that specifies the agent/queue resource that will process this request. • Starting in 8.5.108.02, you can set multiple targets in this option, limited to 5. • Starting in 8.5.114.09, the limit is increased to 15. ### Single Target For a **single** target, format the string according to the URS target specification: `<Target String>@<StatServer name>.<Target Type>` where `Target Type` is one of the following: • A (Agent) • AP (Agent Place) • GA (Group of Agents) • GP (Group of Places) • GC (Campaign Group) `<Target String>` can be a skill expression. In that case, `<Target String>` must start with '?:'. For example: • `Billing@StatServer.GA`—Routes to Agent Group "Billing". • `?:English=20&;Loans=2@StatServer.GA`—Routes to any agent matching the skill expression. See the Universal Routing Server (URS) documentation for additional information about URS targets.

Option	Description

	Multiple Targets
	
	To set multiple targets, create a JSON-formatted string array of maximum 15 elements as follows:
	
	``
	[
	    {
	        "target": "<Target String>@<StatServer name>.<Target Type>",
	        "timeout": "<integer>",
	        "clear":<true/false>,
	        "stat_to_check": "<stat name>",
	        "stat_operator": "< or >",
	        "stat_value": "1"
	    }
	]
	``
	
	• The `timeout` property specifies how long to wait in seconds before switching of targets.
	
	• The `stat_to_check` property can be set to any of the values supported by the Statistics parameter passed to the IRD function `SData(Target, Statistics)`, unless `target` is a skill expression. If `target` is a skill expression, you must choose one of the following values:
	
	    • RStatAgentsReadyvoice—agents ready for voice media.
	
	    • RStatAgentsReady—agents ready for any media.
	
	    • RStatAgentsTotal—agents logged in.
	
	• The `stat_value` property specifies the threshold for the statistic passed in `stat_to_check`. If the condition set by the combination of `stat_to_check`, `stat_operator`, and `stat_value` is met, the current target is skipped, except if it is the last target of the list.
	
	• If `clear=true`, the target will be overridden when switching to the next target; if `clear=false`, the target will be expanded with the next target.
	
	> **Important**
	> If you set multiple targets in this option, then _urs_queued_ttl should be set to the total queue time across all targets.
	
	more...

`_urs_prioritization_strategy = WaitForTarget`	By default, these options respectively match the

Option	Description
_urs_strategy_update_sub_routine = SetRouteDelay	names of the URS strategy and subroutine that you imported into IRD. If you changed one of these names, update the corresponding option to reflect the correct name.

Customization

All of the options in the Voice-User Terminated section are applicable. You can use the default values, or you can set your own values. For the route point option, you must select a value from the drop-down list.

Option	Description
Section: Voice-User Terminated	
_prefix_dial_out = 91	Prefix required to perform a user-terminated (outbound) call from the system.
Section: Voice Treatment	
_treatment_find_agent_fail = GMSApplications/<treatmentfile1>	Music file to be played when the service fails to find the agent in the time specified by the Max Time To Wait For Agent on the Call parameter. This parameter accepts a URI as a string or as a JSON-formatted string. See also _treatment_waiting_for_agent. By default, this option has an empty value and Callback will use the <GMS installation>/Resources/SampleTreatments/all_agents_busy.wav file available in the callback template.
_treatment_waiting_for_agent = GMSApplications/<treatmentfile2>	Music file to play when the customer is waiting for an agent. This parameter accepts a URI as a string or as a JSON-formatted string. If you do not set this option, Callback will use the default <GMS installation>/Resources/SampleTreatments/next_customer_rep.wav file of the callback template.
_treatment_customer_connect = GMSApplications/<treatmentfile3>	URI of the music file to play when the customer answers the callback. The JSON-formatted strings can be used to specify hints to the RequestApplyTreatment. For example: `{ "file": "file_url", "hints": {"hint1":"value"}}`
_treatment_call_failure_answering_machine =	URI of the music file to play when a call is not

Option	Description
GMSApplications/<treatmentfile4>	answered by the customer and is forwarded to the answering machine. JSON-formatted strings can be used to specify hints to the RequestApplyTreatment. The following example makes the music start playing after the answering machine beep is detected: { "file": "file_url", "hints":{ "am-beep-detection":"on"} } By default, the value of this option is empty and Callback uses the <GMS installation>/../Resources/SampleTreatments/ call_fail_ans_machine.wav file from the Callback template. To deactivate the play treatment, set the value of this option to { }.

Important

In the Voice Treatment section, the GMSApplications/<treatmentfile> path is applicable if you are using the treatments builtin to the Callback strategy. If you are not using the builtin treatments, enter the path where you have placed your voice treatment files.

Sample Request and Response Sequence

Create outbound immediate service

For instance, if your callback service is named voice-userterm-immediate, create the following POST request:

Request URL:http://localhost:8080/genesys/1/service/callback/voice-userterm-immediate
Request Method:POST
Status Code:200 OK
Request Headersview source
Accept:*/*
Accept-Encoding:gzip,deflate,sdch
Accept-Language:en-US,en;q=0.8
Connection:keep-alive
Content-Length:660
Content-Type:multipart/form-data; boundary=----WebKitFormBoundaryIWtKHpA86nG3FsWy
Cookie:JSESSIONID=4xjf734hb3pcnh5wd515j6f4; BAYEUX_BROWSER=86721orubxagcqhw0hj14cpyaqk2
gms_user:b16416334828b1d26ef14f329628b55b5a8c631d8928a371a5584722dd7fb673
Host:localhost:8080
Origin:http://localhost:8080
Referer:http://localhost:8080/gmstester/chat.html
User-Agent:Mozilla/5.0 (Windows NT 6.1; WOW64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/27.0.1453.110 Safari/537.36
Request Payload
------WebKitFormBoundaryIWtKHpA86nG3FsWy
Content-Disposition: form-data; name="_customer_number"

916504661232
------WebKitFormBoundaryIWtKHpA86nG3FsWy
Content-Disposition: form-data; name="usr_customer_name"

Bob Markel
------WebKitFormBoundaryIWtKHpA86nG3FsWy
Content-Disposition: form-data; name="usr_reason"

billing question
------WebKitFormBoundaryIWtKHpA86nG3FsWy
Content-Disposition: form-data; name="_device_notification_id"

b16416334828b1d26ef14f329628b55b5a8c631d8928a371a5584722dd7fb673
------WebKitFormBoundaryIWtKHpA86nG3FsWy
Content-Disposition: form-data; name="_device_os"

iOS
------WebKitFormBoundaryIWtKHpA86nG3FsWy--
Response Headersview source
Cache-Control:no-cache
Cache-Control:no-store
Content-Type:application/json;charset=UTF-8

```
Content-Type:application/json;charset=UTF-8
Date:Tue, 30 Jul 2013 07:02:36 GMT
Expires:Thu, 01 Jan 1970 00:00:00 GMT
Pragma:no-cache
Set-Cookie:JSESSIONID=4ieeqn8sa8ni1o2u2nd1br8a4;Path=/genesys
Transfer-Encoding:chunked

Response Body:
{
    "_dialog_id": "0",
    "_action": "ConfirmationDialog",
    "_text": "You will receive the call shortly",
    "_ok_title": "Ok",
    "_id": "369-f5d50ce1-488e-4db1-a472-8c1560b621b6"
}
```

Sequence Diagram

Click the diagram to access full resolution.

User Terminated Immediate - High Level
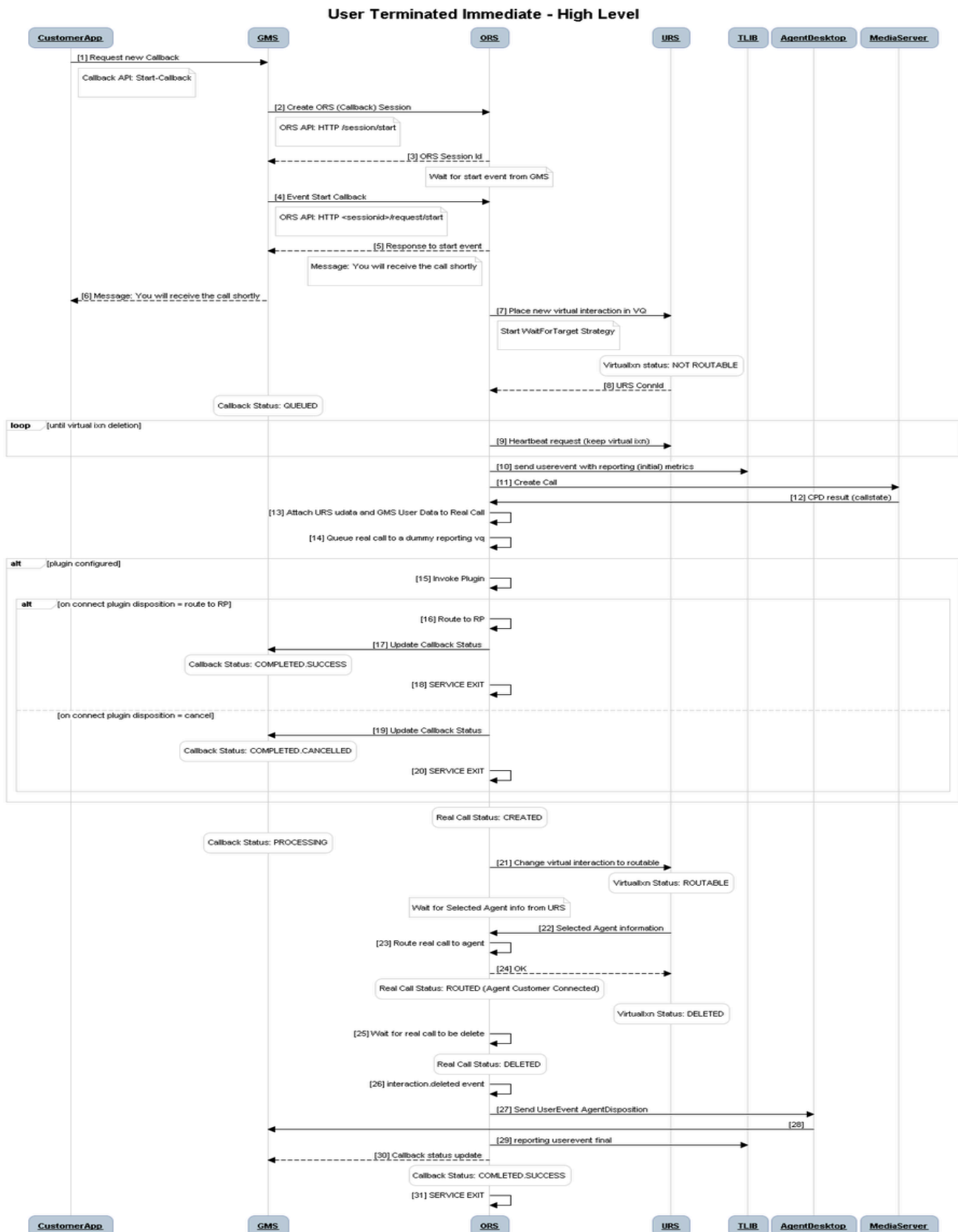
User Terminated Delayed

Call flow

This Callback scenario is an outbound voice service that goes through the following stages:
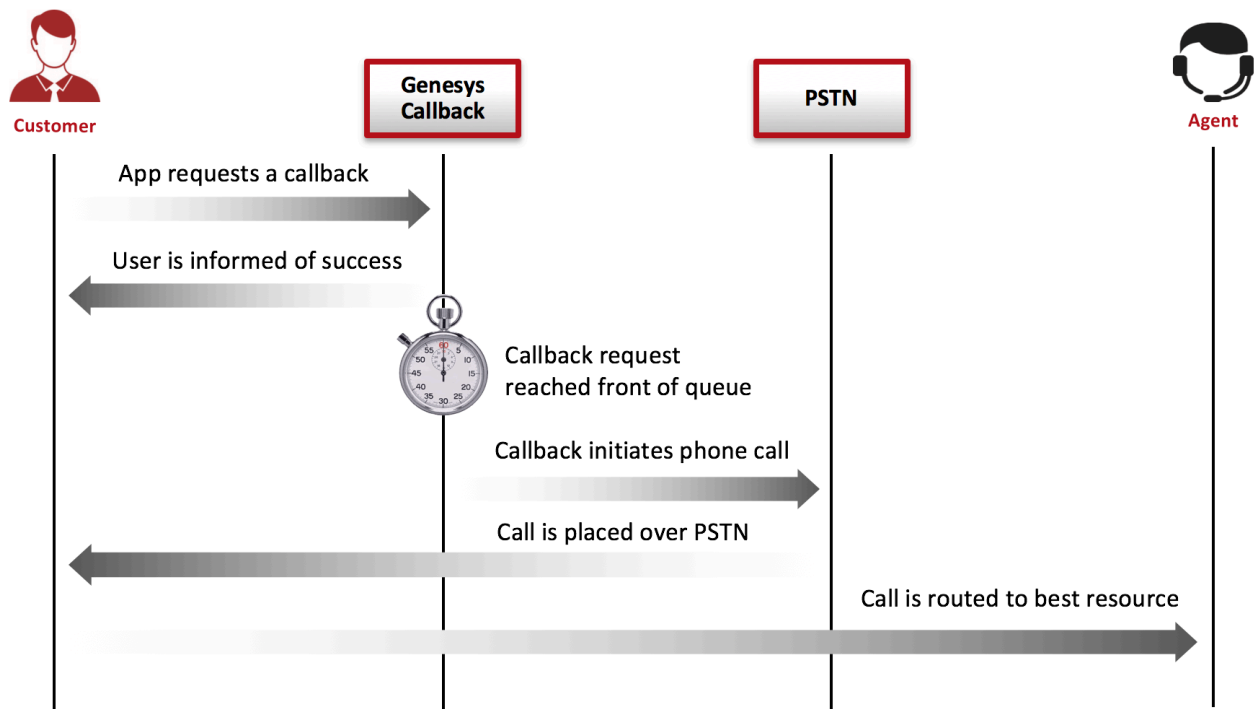
Start Callback
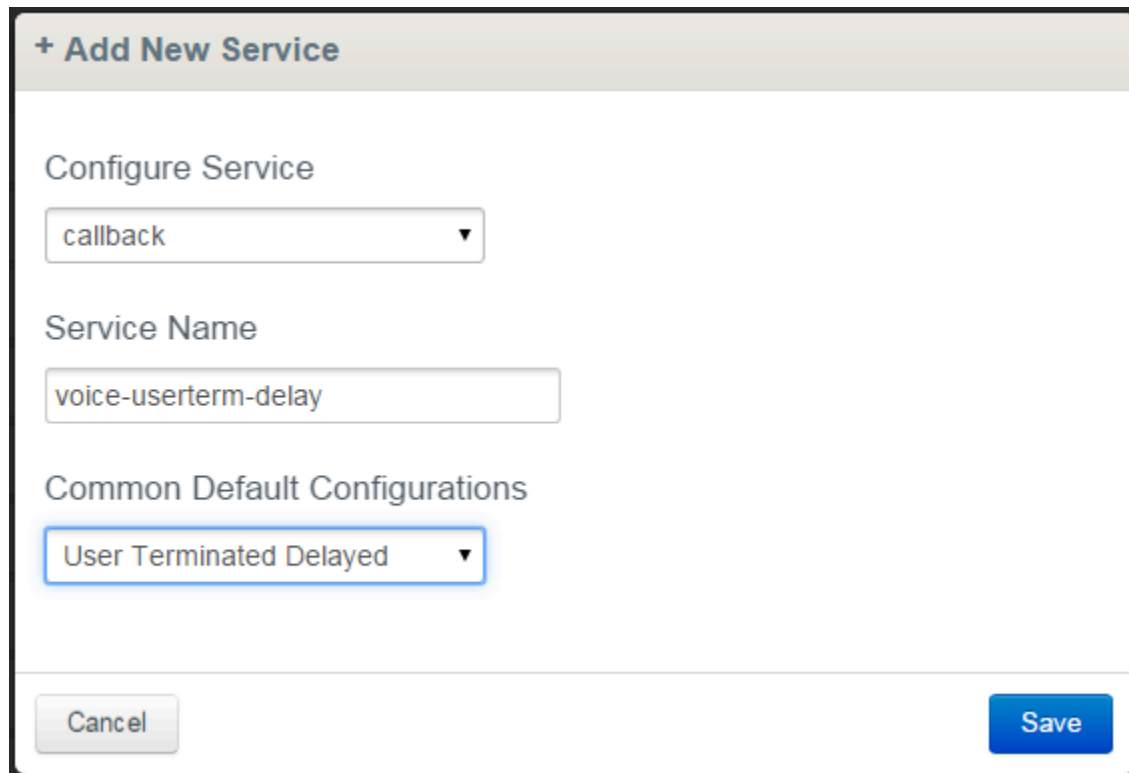
- Callback service: Returns session id to the user.

- Callback service: Waits for an agent to be available.

- Callback service: When an agent is available, notifies mobile device that agent is available.

- Next: Mobile device is expected to send connect request to confirm user's availability.

Connect

- Callback service: Returns a message to expect a call, to the mobile device.

- Callback service: Calls the mobile device.

- Mobile device: Accepts the call.

- Callback service: Identifies that a human has answered the call.

- Callback service: Reserves target to route call.

- Callback service: Routes the call to the target.

- Callback service terminates.

Create your Scenario

In the **Admin UI > Services > Configured Services** tab, add a Callback service with `User-Terminated-Delayed` as the **Common Default Configuration** (see Adding a Service for details).

Enter a service name. This name is the callback execution name of your service and will be used in URLs to access this service. For example, if you set this name to `voice-userterm-delay`, your service URL will be:
`http://host:port/genesys/1/service/callback/voice-userterm-delay`

When you add this service and default configuration, many options are automatically populated with the appropriate default values.

Configuration Options

The table below lists the key options applicable to this scenario. Some options, however, will require you to enter your own values. See the **Comments** column in the following table for these details.

Pre-defined Values

The following options are the default values, which are automatically populated when selecting the pre-defined *User-Terminated-Delayed* service. You do not need to change their values.

Option	Description
_call_direction = USERTERMINATED	This is a default value, automatically populated when using the predefined User-Terminated scenario. You do not need to change this value. • If this option is set to USERORIGINATED, the customer's device will initiate the call to get connected to the agent. • If this option is set to USERTERMINATED, the agent or the system will initiate the call to contact the customer.
_media_type = voice	Media type of the interaction that the service is expected to handle. This option enables URS to select an agent who has the appropriate media capabilities. This is a default value, automatically populated when using the predefined User-Terminated scenario. You do not need to change this value. **This option is mandatory.**
_wait_for_agent = true	True to wait for an agent to connect. If this option is set to true, the service will wait for the agent to initiate the interaction and to send the notification to the customer. If the option is set to false, the interaction can start right after the creation of the service instance. In voice scenarios, the access information will be returned immediately with the service ID. **This option is mandatory.**
_wait_for_user_confirm = true	True to wait for confirmation of the customer's availability. If this option is set to true, the service sends a push notification to the customer's device to get confirmation that the customer is ready to

Option	Description
	have a conversation with the agent. This scenario is possible only if the _wait_for_agent option is set to true.
_ttl = 86400	Duration (in seconds) for which the service will be kept in storage after the Desired Time is passed (Time To Live). Once expired, the service is removed from the system. For example, if you want the callbacks to be visible in the Service Management UI for one week past the execution time, then you should set 7 days of Time To Live, which means _ttl=604800. **This option is mandatory.**
_type = ors	• For Genesys Mobile Services-based services: builtin • For Orchestration Server-based services: ors
_userterminated_first_connect_party=CUSTOMER	First party to connect when _call_direction is set to USERTERMINATED. Set this option to CUSTOMER to call the customer first; set this option to AGENT to call the agent first. **This option is mandatory.**

Additional Required Options

You must configure the following options. See the Universal Routing Server (URS) documentation for additional information about URS targets.

Option	Description
_target="MyTarget@StatServer.GA"	Routing target that specifies the agent/queue resource that will process this request. • Starting in 8.5.108.02, you can set multiple targets in this option, limited to 5. • Starting in 8.5.114.09, the limit is increased to 15. Single Target For a **single** target, format the string according to the URS target specification: <Target String>@<StatServer name>.<Target Type> where Target Type is one of the

Option	Description

	following:
	
	- A (Agent)
	- AP (Agent Place)
	- GA (Group of Agents)
	- GP (Group of Places)
	- GC (Campaign Group)
	
	`<Target String>` can be a skill expression. In that case, `<Target String>` must start with '?:'.
	For example:
	
	- `Billing@StatServer.GA`—Routes to Agent Group "Billing".
	- `?:English=20&;Loans=2@StatServer.GA`—Routes to any agent matching the skill expression.
	
	See the Universal Routing Server (URS) documentation for additional information about URS targets.
	
	Multiple Targets
	
	To set multiple targets, create a JSON-formatted string array of maximum 15 elements as follows:
	
	``[
	    {
	        "target": "<Target
	String>@<StatServer name>.<Target Type>",
	        "timeout": "<integer>",
	        "clear":<true/false>,
	        "stat_to_check": "<stat name>",
	        "stat_operator": "< or >",
	        "stat_value": "1"
	    }
	]```
	
	- The `timeout` property specifies how long to wait in seconds before switching of targets.
	- The `stat_to_check` property can be set to any of the values supported by the Statistics parameter passed to the IRD function `SData(Target, Statistics)`, unless `target` is a skill expression. If `target` is a skill expression, you must choose one of the following values:
	
	  - `RStatAgentsReadyvoice`—agents ready for voice media.
	  - `RStatAgentsReady`—agents ready for any media.
	  - `RStatAgentsTotal`—agents logged in.

Option	Description
	• The `stat_value` property specifies the threshold for the statistic passed in `stat_to_check`. If the condition set by the combination of `stat_to_check`, `stat_operator`, and `stat_value` is met, the current target is skipped, except if it is the last target of the list. • If `clear=true`, the target will be overridden when switching to the next target; if `clear=false`, the target will be expanded with the next target. **Important** If you set multiple targets in this option, then `_urs_queued_ttl` should be set to the total queue time across all targets. more...
_urs_virtual_queue="MyVirtualQueue"	Virtual queue (alias) to which the service request will be added.
_urs_prioritization_strategy = WaitForTarget _urs_strategy_update_sub_routine = SetRouteDelay	By default, these options respectively match the names of the URS strategy and subroutine that you imported into IRD. If you changed one of these names, update the corresponding option to reflect the correct name.

Customization

The options shown here and all of the options of the Voice User Terminated section are applicable for this scenario. You can use the default values, or you can set your own values. For the route point option, you must select a value from the drop-down list in the UI.

Option	Description
Section: Voice User Terminated	
_on_user_confirm_timeout=CONNECT-ANYWAY	Selects the action to perform if the user does not submit his or her confirmation in response to the push notification. • `CONNECT-ANYWAY` will continue with the call. • `CANCEL` cancels the service request.
_prefix_dial_out=91	Prefix required to perform a user-terminated (outbound) call from the system.

Option	Description
_route_point={Route Point}@{Telephony Switch}	Route point from which the system can create a user-terminated (outbound) call. **This option is mandatory.**
Section: Voice Treatment	
_treatment_call_failure_answering_machine	URI of the music file to play when a call is not answered by the customer and is forwarded to the answering machine. JSON-formatted strings can be used to specify hints to the RequestApplyTreatment. The following example makes the music start playing after the answering machine beep is detected: { "file": "file_url", "hints":{ "am-beep-detection":"on"} } By default, the value of this option is empty and Callback uses the <GMS installation>/../Resources/SampleTreatments/ call_fail_ans_machine.wav file from the Callback template. To deactivate the play treatment, set the value of this option to { }.
_treatment_find_agent_fail = GMSApplications/<treatmentfile1>	Music file to be played when the service fails to find the agent in the time specified by the Max Time To Wait For Agent on the Call parameter. This parameter accepts a URI as a string or as a JSON-formatted string. See also _treatment_waiting_for_agent. By default, this option has an empty value and Callback will use the <GMS installation>/Resources/ SampleTreatments/all_agents_busy.wav file available in the callback template.
_treatment_waiting_for_agent = GMSApplications/<treatmentfile2>	Music file to play when the customer is waiting for an agent. This parameter accepts a URI as a string or as a JSON-formatted string. If you do not set this option, Callback will use the default <GMS installation>/Resources/SampleTreatments/ next_customer_rep.wav file of the callback template.
_treatment_customer_connect = GMSApplications/<treatmentfile3>	URI of the music file to play when the customer answers the callback. The JSON-formatted strings can be used to specify hints to the RequestApplyTreatment. For example:

Option	Description
	`{ "file": "file_url", "hints": {"hint1":"value"}}`

> **Important**
>
> In the Voice Treatment section, the GMSApplications/<treatmentfile> path is applicable if you are using the treatments builtin to the Callback strategy. If you are not using the builtin treatments, enter the path where you have placed your voice treatment files.

Sample Request and Response Sequence

Create outbound delay service

For instance, if your callback service is named `voice-userterm-delay`, create the following POST request:

```
Request URL:http://localhost:8080/genesys/1/service/callback/voice-userterm-delay
Request Method:POST
Status Code:200 OK
Request Headersview source
Accept:*/*
Accept-Encoding:gzip,deflate,sdch
Accept-Language:en-US,en;q=0.8
Connection:keep-alive
Content-Length:662
Content-Type:multipart/form-data; boundary=----WebKitFormBoundaryABpcDouIWQ5inBWl
Cookie:JSESSIONID=4ieeqn8sa8ni1o2u2nd1br8a4; BAYEUX_BROWSER=86721orubxagcqhw0hj14cpyaqk2
gms_user:b16416334828b1d26ef14f329628b55b5a8c631d8928a371a5584722dd7fb673
Host:localhost:8080
Origin:http://localhost:8080
Referer:http://localhost:8080/gmstester/chat.html
User-Agent:Mozilla/5.0 (Windows NT 6.1; WOW64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/27.0.1453.110 Safari/537.36
Request Payload
------WebKitFormBoundaryABpcDouIWQ5inBWl
Content-Disposition: form-data; name="_customer_number"

916504661232
------WebKitFormBoundaryABpcDouIWQ5inBWl
Content-Disposition: form-data; name="usr_customer_name"

Bob Markel
------WebKitFormBoundaryABpcDouIWQ5inBWl
Content-Disposition: form-data; name="usr_reason"

billing question
------WebKitFormBoundaryABpcDouIWQ5inBWl
Content-Disposition: form-data; name="_device_notification_id"

b16416334828b1d26ef14f329628b55b5a8c631d8928a371a5584722dd7fb673
------WebKitFormBoundaryABpcDouIWQ5inBWl
Content-Disposition: form-data; name="_device_os"

comet
------WebKitFormBoundaryABpcDouIWQ5inBWl--
Response Headersview source
Cache-Control:no-cache
Cache-Control:no-store
Content-Type:application/json;charset=UTF-8
```

```
Content-Type:application/json;charset=UTF-8
Date:Tue, 30 Jul 2013 07:04:31 GMT
Expires:Thu, 01 Jan 1970 00:00:00 GMT
Pragma:no-cache
Set-Cookie:JSESSIONID=1b81btxjbrb1wybz5a93i24io;Path=/genesys
Transfer-Encoding:chunked

Response Body:
{
    "_id": "369-b100700a-4ce8-48f7-b1b0-1944b12359b9",
    "_text": "We will notify you when agent is available"
}

Push notification data:
{
    "_dialog_id": "0",
    "_action": "ConfirmationDialog",
    "_text": "You will receive the call shortly",
    "_ok_title": "Ok",
    "_id": "369-b100700a-4ce8-48f7-b1b0-1944b12359b9"
}
```

Connect (user confirmation)

```
Request URL:http://localhost:8080/genesys/1/service/369-b100700a-4ce8-48f7-b1b0-1944b12359b9/connect
Request Method:POST
Status Code:200 OK
Request Headersview source
Accept:*/*
Accept-Encoding:gzip,deflate,sdch
Accept-Language:en-US,en;q=0.8
Connection:keep-alive
Content-Length:44
Content-Type:multipart/form-data; boundary=----WebKitFormBoundaryNY84ld7wm7oHB9fp
Cookie:JSESSIONID=1b81btxjbrb1wybz5a93i24io; BAYEUX_BROWSER=86721orubxagcqhw0hj14cpyaqk2
gms_user:b16416334828b1d26ef14f329628b55b5a8c631d8928a371a5584722dd7fb673
Host:localhost:8080
Origin:http://localhost:8080
Referer:http://localhost:8080/gmstester/chat.html
User-Agent:Mozilla/5.0 (Windows NT 6.1; WOW64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/27.0.1453.110 Safari/537.36
Request Payload
------WebKitFormBoundaryNY84ld7wm7oHB9fp--
Response Headersview source
Cache-Control:no-cache
Cache-Control:no-store
Content-Length:26
Content-Type:application/json;charset=UTF-8
Content-Type:application/json;charset=UTF-8
Date:Tue, 30 Jul 2013 07:04:35 GMT
Expires:Thu, 01 Jan 1970 00:00:00 GMT
Pragma:no-cache
Set-Cookie:JSESSIONID=mjjvtphwb8lpce7io23ggxcu;Path=/genesys

Response Body:
{
    "_dialog_id": "0",
    "_action": "ConfirmationDialog",
    "_text": "You will receive the call shortly",
    "_ok_title": "Ok",
    "_id": "369-b100700a-4ce8-48f7-b1b0-1944b12359b9"
}
```

Sequence Diagram

Click the diagram to access full resolution.

User Terminated Delay - High Level

CustomerApp | GMS | ORS | URS | TLIB | AgentDesktop | MediaServer

[1] Request new Callback

Callback API: Start-Callback

[2] Create ORS (Callback) Session

ORS API: HTTP /session/start

[3] ORS Session Id

Wait for start event from GMS

[4] Event Start Callback

ORS API: HTTP <sessionid>/request/start

[5] Response to start event

Message: We will notify you when agent is available

[6] Message : We will notify you when agent is available

[7] Place new virtual interaction in VQ

Start WaitForTarget Strategy

VirtualIxn status: NOT ROUTABLE

[8] URS.ConnId

Callback Status: QUEUED

loop [until virtual ixn deletion]

[9] Heartbeat request (keep virtual ixn)

[10] send userevent with reporting (initial) metrics

Wait for Agent available notification
Key differentiator in delay scenario
options _wait_for_agent=true

alt [notifications enabled]

loop [for configured confirmation attempts with retry delay interval]

[11] Agent available notification

[12] Push notification

Confirmation options : Connect/Snooze/Cancel

end loop

alt confirmation response: Cancel

ORS-->GMS: Callback status update

note left of ORS

COMLETED.CANCELLED

[13] service.exit

CALLBACK EXIT

[alt confirmation response: Snooze]

[14] Wait for snooze duration and retry confirm loop

[15] Agent will be available, dial now

confirmation response: Connect

[16] connect request

Callback API: HTTP <gmsserviceid>/connect

[17] Event Connect Callback

ORS API: HTTP <sessionid>/request/connect

[18] Create Call

[19] CPD result (callstate)

[20] Attach URS udata and GMS User Data to Real Call

[21] Queue real call to a dummy reporting vq

alt [plugin configured]

[22] Invoke Plugin

alt [on connect plugin disposition = route to RP]

[23] Route to RP

[24] Update Callback Status

Callback Status: COMPLETED.SUCCESS

[25] SERVICE EXIT

[on connect plugin disposition = cancel]

[26] Update Callback Status

Callback Status: COMPLETED.CANCELLED

[27] SERVICE EXIT

Real Call Status: CREATED

Callback Status: PROCESSING

[28] Change virtual interaction to routable

VirtualIxn Status: ROUTABLE

Wait for Selected Agent info from URS

[29] Selected Agent information

[30] Route real call to agent

[31] OK

Real Call Status: ROUTED (Agent Customer Connected)

VirtualIxn Status: DELETED

[32] Wait for real call to be delete

Real Call Status: DELETED

[33] interaction.deleted event

[34] Send UserEvent AgentDisposition

[35]

[36] reporting userevent final

[37] Callback status update

Callback Status: COMLETED.SUCCESS

[38] SERVICE EXIT

CustomerApp | GMS | ORS | URS | TLIB | AgentDesktop | MediaServer

User Terminated Scheduled Scenarios

Overview

This page describes two Callback scenarios:

- **User Terminated Scheduled**, also known as *User Terminated at Desired Time*.
- **User Terminated Delayed Scheduled**, also known as *User Terminated Delayed at Desired Time*.

The _wait_for_agent option selects between **Delayed** (_wait_for_agent=true) and **Not Delayed** (_wait_for_agent=false). The scenarios go through the following stages:

Find Available Slot

- Mobile device: Request for availability at the desired time.
- Availability API: Provides available time slots during the open business hours encompassing the desired time.
- Mobile device: Starts Callback with user selected available time slot.

Start Callback

Start Callback Not Delayed _wait_for_agent **= false**	Start Callback Delayed _wait_for_agent **= true**
     - Callback service: Returns session id to the user. - Callback service: Starts execution at time determined by the scheduled time minus the current EWT - Callback service: Initiates the outbound call after execution starts.	- Callback service: Returns session id to the user. - Callback service: Starts execution at time determined by the scheduled time minus the current EWT - Callback service: Waits for an agent to be available. - Callback service: When an agent is available, notifies mobile device that agent is available. Confirm user's availability and Connect - Callback service: Returns a message to expect a call, to the mobile device. - Callback service: Calls the mobile device. - Mobile device: Accepts the call.

Start Callback Not Delayed	Start Callback Delayed
_wait_for_agent = **false**	_wait_for_agent = **true**
	- Callback service: Identifies that a human has answered the call. - Callback service: Reserves target to route call. - Callback service: Routes the call to the target. - Callback service terminates.

Create your Scenario

In the Service Management UI, select **Callback and Mobile Engagement"' and in the** Configured Services **tab, add a Callback service with** User-Terminated **(Immediate or Delayed) as the** Common Default Configuration **(see Adding a Service for details).**

Enter a service name. This is the callback execution name of your service that will be used in URLs to access this service. For example, if you set this name to user-scheduled, your service URL will be: http://host:port/genesys/1/service/callback/user-scheduled

Configuration

When you add a service and default configuration, many options are automatically populated with the appropriate default values. Refer to the User Terminated Delayed and the User Terminated Immediate scenarios for further details. You will also need to create and configure an Office Hours service that you will use to configure the _business_hours_service option in your callback scenario: *No results*

The _wait_for_agent option selects between Delayed (_wait_for_agent=true) and not Delayed (_wait_for_agent=false). If you choose a User Terminated Delayed scenario, the outbound call will not occur until an agent is available.

- User Terminated Scheduled scenario:
    1. Start the callback service at the scheduled time minus the current EWT defined for the _target.
    2. The outbound call occurs immediately after starting the service.
- User Terminated Delayed Scheduled
    1. Start the callback service at the scheduled time minus the current EWT defined for the _target.
    2. Wait for the time dial notification from URS before making the outbound call.

User Terminated Scheduled scenarios might create a longer delay between the time of the outbound call and being connected to an agent if the EWT is not accurate.

Sample Request and Response Sequence

Get timeslots

```
Request URL:http://192.168.184.128:8080/genesys/1/service/callback/user-scheduled/
availability?timestamp=2014-07-30T14:00:00.000Z&_=1406647599350
Request Method:GET
Status Code:200 OK
Request Headersview source
Accept:application/json, text/javascript, */*; q=0.01
Accept-Encoding:gzip,deflate,sdch
Accept-Language:en-US,en;q=0.8
Cache-Control:no-cache
Connection:keep-alive
Cookie:JSESSIONID=1mmvrc06utu9l1s9azezm677ns; BAYEUX_BROWSER=db7a1s13ad7b0xvghy78iy6y17cq
Host:192.168.184.128:8080
Pragma:no-cache
Referer:http://192.168.184.128:8080/genesys/admin/js/sample/cb/index.html
User-Agent:Mozilla/5.0 (Windows NT 6.1; WOW64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/31.0.1650.57 Safari/537.36
Query String Parametersview sourceview URL encoded
timestamp:2014-07-30T14:00:00.000Z
_:1406647599350
Response Headersview source
Cache-Control:no-cache
Cache-Control:no-store
Content-Type:application/json;charset=UTF-8
Date:Wed, 30 Jul 2014 14:36:01 GMT
Pragma:no-cache
Transfer-Encoding:chunked

Response Body:
{
    "2014-07-30T06:00:00.000Z": 100,
    "2014-07-30T07:00:00.000Z": 100,
    "2014-07-30T08:00:00.000Z": 100,
    "2014-07-30T09:00:00.000Z": 100,
    "2014-07-30T10:00:00.000Z": 100,
```

```
    "2014-07-30T11:00:00.000Z": 100,
    "2014-07-30T12:00:00.000Z": 100,
}
```

Create outbound scheduled delay service

```
Request URL:http://192.168.184.128:8080/genesys/1/service/callback/user-scheduled
Request Method:POST
Status Code:200 OK
Request Headersview source
Accept:application/json, text/javascript, */*; q=0.01
Accept-Encoding:gzip,deflate,sdch
Accept-Language:en-US,en;q=0.8
Cache-Control:no-cache
Connection:keep-alive
Content-Length:915
Content-Type:multipart/form-data;boundary=AaB03x;charset=UTF-8
Cookie:JSESSIONID=1mmvrc06utu9l1s9azezm677ns; BAYEUX_BROWSER=db7a1s13ad7b0xvghy78iy6y17cq
gms_user:jdoe_desktop
Host:192.168.184.128:8080
Origin:http://192.168.184.128:8080
Pragma:no-cache
Referer:http://192.168.184.128:8080/genesys/admin/js/sample/cb/index.html
User-Agent:Mozilla/5.0 (Windows NT 6.1; WOW64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/31.0.1650.57 Safari/537.36
Request Payload
--AaB03x
Content-Disposition: form-data; name="first_name"

John
--AaB03x
Content-Disposition: form-data; name="last_name"

Doe
--AaB03x
Content-Disposition: form-data; name="_provide_code"

false
--AaB03x
Content-Disposition: form-data; name="_desired_time"

2014-07-30T15:00:00.000Z
```

```
--AaB03x
Content-Disposition: form-data; name="_customer_number"

5115
--AaB03x
Content-Disposition: form-data; name="location_lat"

37.8197
--AaB03x
Content-Disposition: form-data; name="location_long"

-122.4786
--AaB03x
Content-Disposition: form-data; name="_device_os"

comet
--AaB03x
Content-Disposition: form-data; name="_call_direction"

USERTERMINATED
--AaB03x
Content-Disposition: form-data; name="_wait_for_agent"

true
--AaB03x
Content-Disposition: form-data; name="_wait_for_user_confirm"

true
--AaB03x
Content-Disposition: form-data; name="_media_type"

voice
--AaB03x
Response Headersview source
Access-Control-Allow-Credentials:true
Access-Control-Allow-Origin:http://192.168.184.128:8080
Access-Control-Expose-Headers:
Cache-Control:no-cache
Cache-Control:no-store
Content-Type:application/json;charset=UTF-8
Date:Wed, 30 Jul 2014 14:36:16 GMT
Pragma:no-cache
Transfer-Encoding:chunked
```

```
Response Body:
{"_id":"440-96984577-a614-4f18-b80e-aaf91e78b178"}

Push notification data:
{
    "_dialog_id": "0",
    "_action": "ConfirmationDialog",
    "_text": "You will receive the call shortly",
    "_ok_title": "Ok",
    "_id":"440-96984577-a614-4f18-b80e-aaf91e78b178"
}
```

Connect (user confirmation)

```
Request URL:http://localhost:8080/genesys/1/service/440-96984577-a614-4f18-b80e-aaf91e78b178/connect
Request Method:POST
Status Code:200 OK
Request Headersview source
Accept:*/*
Accept-Encoding:gzip,deflate,sdch
Accept-Language:en-US,en;q=0.8
Connection:keep-alive
Content-Length:44
Content-Type:multipart/form-data; boundary=----WebKitFormBoundaryNY84ld7wm7oHB9fp
Cookie:JSESSIONID=1b81btxjbrb1wybz5a93i24io; BAYEUX_BROWSER=86721orubxagcqhw0hj14cpyaqk2
gms_user:b16416334828b1d26ef14f329628b55b5a8c631d8928a371a5584722dd7fb673
Host:localhost:8080
Origin:http://localhost:8080
Referer:http://localhost:8080/gmstester/chat.html
User-Agent:Mozilla/5.0 (Windows NT 6.1; WOW64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/27.0.1453.110 Safari/537.36
Request Payload
------WebKitFormBoundaryNY84ld7wm7oHB9fp--
Response Headersview source
Cache-Control:no-cache
Cache-Control:no-store
Content-Length:26
Content-Type:application/json;charset=UTF-8
Content-Type:application/json;charset=UTF-8
Date:Tue, 30 Jul 2013 07:04:35 GMT
Expires:Thu, 01 Jan 1970 00:00:00 GMT
```

```
Pragma:no-cache
Set-Cookie:JSESSIONID=mjjvtphwb8lpce7io23ggxcu;Path=/genesys

Response Body:
{
    "_dialog_id": "0",
    "_action": "ConfirmationDialog",
    "_text": "You will receive the call shortly",
    "_ok_title": "Ok",
    "_id": "440-96984577-a614-4f18-b80e-aaf91e78b178"
}
```

Sequence Diagram

Click on the diagram to access full resolution.

User Terminated Delay Scheduled – High Level
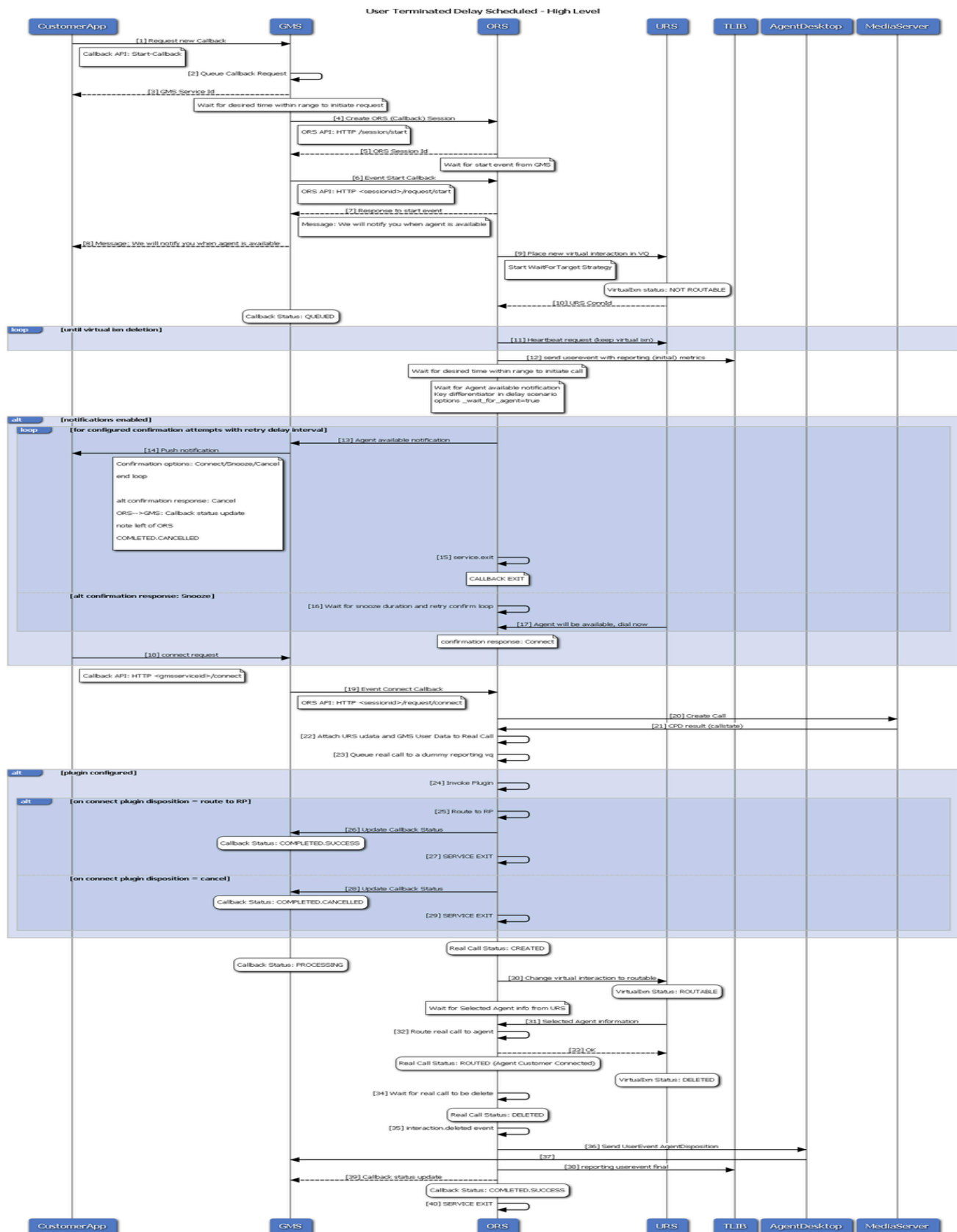
User Terminated Delayed Agent Preview

Call flow
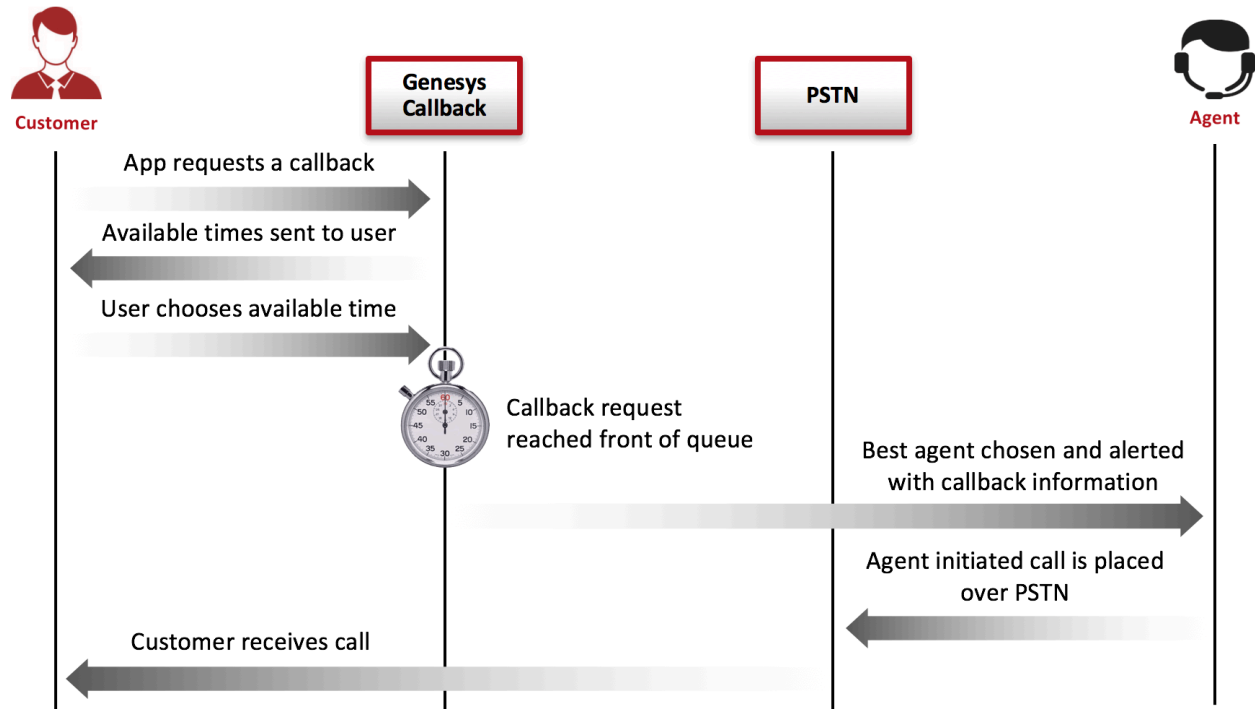
This Callback scenario is an outbound service that goes through the following stages:

Start Callback

- Callback service: Returns session id to the user.
- Callback service: Waits for an agent to be available.
- Callback service: Reserves agent.
- Callback service: Sends preview invite to agent.
- Agent Desktop: Displays customer attached data with actions `Accept` and `Cancel`.
- Agent: Accepts the invitation by clicking the `Accept` button.
- Callback service: Receives the agent acceptance request.
- Callback service: Calls the mobile device from reserved agent's DN.
- Mobile device: Accepts the call.

- Callback service: Waits for interaction to be deleted.
- Callback service: Terminates after interaction is deleted.

Create your Scenario

In the **Admin UI > Services > Configured Services** tab, add a Callback service with User Terminated-Preview as the **Common Default Configuration** (see Adding a Service for de

Enter a service name. This name is the callback execution name of your service and will be u URLs to access this service. For example, if you set this name to user_term_preview, your s URL will be:
http://host:port/{base-web-application}/service/callback/user_term_preview

When you add this service and default configuration, many options are automatically popula the appropriate default values.

Configuration Options

Predefined Values

These are the default values, which are automatically populated when using the pre-defined User-Terminated-Immediate service. You do not need to change these values.

Option	Description
_media_type=voice	Media type of the interaction that the service is expected to handle. This option enables URS to select an agent who has the appropriate media capabilities. This is a default value, automatically populated when using the predefined User-Terminated scenario. You do not need to change this value. **This option is mandatory.**
_wait_for_agent = true	True to wait for an agent to connect. If this option is set to true, the service will wait for the agent to initiate the interaction and to send the notification to the customer. If the option is set to false, the interaction can start right after the creation of the service instance. In voice scenarios, the access information will be returned immediately with the service ID. **This option is mandatory.**
_wait_for_user_confirm = false	True to wait for confirmation of the customer's availability. If this option is set to true, the service sends a push notification to the customer's device to get confirmation that the customer is ready to have a conversation with the agent. This scenario is possible only if the _wait_for_agent option is set to true.
_agent_preview = true	Enables Agent Preview. If set to true, the Preview Dialog with caller information is displayed to the agent.
_call_direction = USERTERMINATED	This is a default value, automatically populated when using the predefined User-Terminated scenario. You do not need to change this value.

Option	Description
	- If this option is set to USERORIGINATED, the customer's device will initiate the call to get connected to the agent. - If this option is set to USERTERMINATED, the agent or the system will initiate the call to contact the customer.
_ttl = 86400	Duration (in seconds) for which the service will be kept in storage after the Desired Time is passed (Time To Live). Once expired, the service is removed from the system. For example, if you want the callbacks to be visible in the Service Management UI for one week past the execution time, then you should set 7 days of Time To Live, which means _ttl=604800. **This option is mandatory.**
_type = ors	- For Genesys Mobile Services-based services: builtin - For Orchestration Server-based services: ors
_provide_code= false	If true, returns a randomly generated code to be used for the authentication of the user originated (inbound) call. **This option is mandatory.**
_cpd_enable = false	Enables CPD. If this option is set to true, CPD will be performed on a callback made to the customer. - If CPD results in a human or silence detection, the call will be routed to the agent. - If a fax is detected, the call will be disconnected and marked complete. - If an answering machine is detected, the answering machine treatment is played. **This option is mandatory.**
_use_debug_push_certificate = false	Use debug certificates for the push notification provider

Option	Description

Additional Required Options

You must enter a string value for the following options:

Option	Description
_route_point= "{Route Point}@{Telephony Switch}"	Route point from which the system can create a user-terminated (outbound) call. **This option is mandatory.**
_urs_virtual_queue = "MyVirtualQueue"	Virtual queue (alias) to which the service request will be added.
_resource_group="{name of the resource pool configured under Transactions/GMS_Resources/ Annex}"	Resource group from which access number is to be allocated. **This option is mandatory.**
_target = "MyTarget@StatServer.GA"	Routing target that specifies the agent/queue resource that will process this request. • Starting in 8.5.108.02, you can set multiple targets in this option, limited to 5. • Starting in 8.5.114.09, the limit is increased to 15. **Single Target** For a **single** target, format the string according to the URS target specification: `<Target String>@<StatServer name>.<Target Type>` where `Target Type` is one of the following: • A (Agent) • AP (Agent Place) • GA (Group of Agents) • GP (Group of Places) • GC (Campaign Group) `<Target String>` can be a skill expression. In that case, `<Target String>` must start with '?:'. For example:

Option	Description

	• `Billing@StatServer.GA`—Routes to Agent Group "Billing".
	
	• `?:English=20&;Loans=2@StatServer.GA`—Routes to any agent matching the skill expression.
	
	See the Universal Routing Server (URS) documentation for additional information about URS targets.
	
	## Multiple Targets
	
	To set multiple targets, create a JSON-formatted string array of maximum 15 elements as follows:
	
	```[{ "target": "<Target String>@<StatServer name>.<Target Type>", "timeout": "<integer>", "clear":<true/false>, "stat_to_check": "<stat name>", "stat_operator": "< or >", "stat_value": "1" }]```
	
	• The `timeout` property specifies how long to wait in seconds before switching of targets.
	
	• The `stat_to_check` property can be set to any of the values supported by the Statistics parameter passed to the IRD function `SData(Target, Statistics)`, unless `target` is a skill expression. If `target` is a skill expression, you must choose one of the following values:
	
	   • `RStatAgentsReadyvoice`—agents ready for voice media.
	
	   • `RStatAgentsReady`—agents ready for any media.
	
	   • `RStatAgentsTotal`—agents logged in.
	
	• The `stat_value` property specifies the threshold for the statistic passed in `stat_to_check`. If the condition set by the combination of `stat_to_check`, `stat_operator`, and `stat_value` is met, the current target is skipped, except if it is the last target of the list.
	
	• If `clear=true`, the target will be overridden when switching to the next target; if `clear=false`, the target will be expanded with the next target.

Option	Description
	Important If you set multiple targets in this option, then _urs_queued_ttl should be set to the total queue time across all targets. more...
_urs_prioritization_strategy = WaitForTarget _urs_strategy_update_sub_routine = SetRouteDelay	By default, these options respectively match the names of the URS strategy and subroutine that you imported into IRD. If you changed one of these names, update the corresponding option to reflect the correct name.

Customization

All of the options in the Voice-User Terminated section are applicable. You can use the default values, or you can set your own values. For the route point option, you must select a value from the drop-down list.

Option	Description
Section: Voice-User Terminated	
_prefix_dial_out = 91	_prefix_dial_out
_userterminated_first_connect_party = CUSTOMER	First party to connect when _call_direction is set to USERTERMINATED. Set this option to CUSTOMER to call the customer first; set this option to AGENT to call the agent first. **This option is mandatory.**
_agent_preview_allow_reject = false	Allows the agent to reject the call in the preview dialog. • If the option is set to 0, the preview dialog does not display the reject button. • If the option is greater than 0, its value determines the number of times that an agent can reject the service request; the reject option will not be displayed to the next agent.
_agent_preview_timeout = 30	Duration in seconds that the agent has to preview the callback information and submit a reply. The Preview dialog will automatically close after this timeout and submit a reject from the agent. In this scenario, the call will go back in the queue. During this period, the agent is reserved for the Callback interaction and is not an eligible target for other

Option	Description
	interactions; therefore, Genesys recommends to evaluate carefully when extending this timeout beyond 30 seconds (default).
Section: Voice Treatment	
_treatment_find_agent_fail = GMSApplications/<treatmentfile1>	Music file to be played when the service fails to find the agent in the time specified by the Max Time To Wait For Agent on the Call parameter. This parameter accepts a URI as a string or as a JSON-formatted string. See also _treatment_waiting_for_agent. By default, this option has an empty value and Callback will use the <GMS installation>/Resources/ SampleTreatments/all_agents_busy.wav file available in the callback template.
_treatment_waiting_for_agent = GMSApplications/<treatmentfile2>	Music file to play when the customer is waiting for an agent. This parameter accepts a URI as a string or as a JSON-formatted string. If you do not set this option, Callback will use the default <GMS installation>/Resources/SampleTreatments/ next_customer_rep.wav file of the callback template.
_treatment_customer_connect = GMSApplications/<treatmentfile3>	URI of the music file to play when the customer answers the callback. The JSON-formatted strings can be used to specify hints to the RequestApplyTreatment. For example: { "file": "file_url", "hints": {"hint1":"value"}}
_treatment_call_failure_answering_machine = GMSApplications/<treatmentfile4>	URI of the music file to play when a call is not answered by the customer and is forwarded to the answering machine. JSON-formatted strings can be used to specify hints to the RequestApplyTreatment. The following example makes the music start playing after the answering machine beep is detected: { "file": "file_url", "hints":{ "am-beep-detection":"on"} } By default, the value of this option is empty and Callback uses the <GMS installation>/../Resources/SampleTreatments/ call_fail_ans_machine.wav file from the Callback template. To deactivate the play treatment, set the value of this option to { }.

Option	Description

> **Important**
>
> In the Voice Treatment section, the GMSApplications/<treatmentfile> path is applicable if you are using the treatments builtin to the Callback strategy. If you are not using the builtin treatments, enter the path where you have placed your voice treatment files.

Sample Request and Response Sequence

Request

```
URL:http://192.168.184.128:8080/genesys/1/service/callback/user_term_preview
Request Method:POST
Status Code:200 OK
Request Headersview source
Accept:application/json, text/javascript, */*; q=0.01
Accept-Encoding:gzip,deflate,sdch
Accept-Language:en-US,en;q=0.8
Cache-Control:no-cache
Connection:keep-alive
Content-Length:513
Content-Type:multipart/form-data;boundary=AaB03x;charset=UTF-8
Cookie:JSESSIONID=1q2r9rzpkelzas5uh21vrubqd; BAYEUX_BROWSER=c397-1v94320p0gfmni2a3d4xs12mv
gms_user:jdoe_desktop
Host:192.168.184.128:8080
Origin:http://192.168.184.128:8080
Pragma:no-cache
Referer:http://192.168.184.128:8080/genesys/admin/js/sample/cb/index.html
User-Agent:Mozilla/5.0 (Windows NT 6.1; WOW64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/
31.0.1650.57 Safari/537.36
Request Payload
--AaB03x
Content-Disposition: form-data; name="first_name"

John
--AaB03x
Content-Disposition: form-data; name="last_name"

Doe
--AaB03x
Content-Disposition: form-data; name="_provide_code"

false
--AaB03x
Content-Disposition: form-data; name="_customer_number"

5125
--AaB03x
```

```
Content-Disposition: form-data; name="location_lat"

37.8197
--AaB03x
Content-Disposition: form-data; name="location_long"

-122.4786
--AaB03x
Content-Disposition: form-data; name="_device_os"

comet
--AaB03x
Content-Disposition: form-data; name="_agent_preview_data"

Field1Val, Field2Val,Field3Val,Field4Val
--AaB03x
```

Response

```
HTTP/1.1 200 OK
{"_dialog_id":"0","_action":"ConfirmationDialog","_text":"You will receive a call from the
agent","_ok_title":"Ok","_id":"440-f86e4dff-2c00-4753-a876-5b52354566de"}
```

Sequence Diagram

Click on the diagram to access full resolution. For a more detailed diagram, click here.

User Terminated Delay Preview (Agent First Connect Party) - High Level
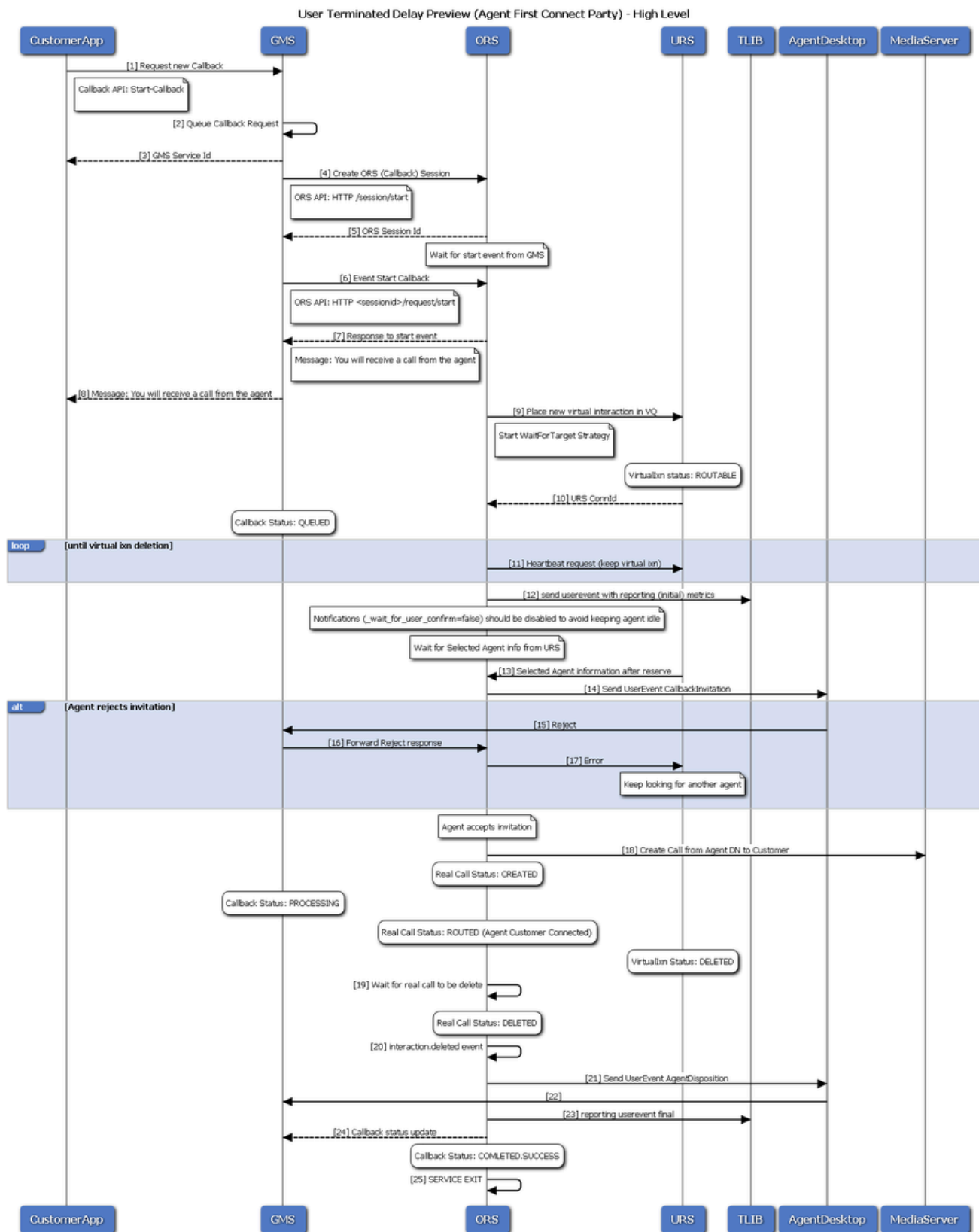
IVR Classic Callback

In this scenario, the IVR handling an inbound call has logic to check for a long waiting time and offers to call back the caller. To see an implementation example of this IVR, refer to the Classic Callback Sample.

Call Flow

IVR Check Wait Time

- IVR: Requests a statistic to determine the estimated wait time for the target.
- IVR: If the waiting time is above the configured threshold, offers a callback.

Start Callback

- IVR: Callback is accepted and IVR submits a request to the callback service.
- Callback service: Returns GMS service ID to the IVR.
- Callback service: If _wait_for_agent = true, waits for an agent to be available.
- Callback service: Calls the customer number.
- Customer: Accepts the call.

Connect to Agent

- Callback service: Identifies that a human has answered the call.
- Callback service: Reserves the target to route the call.
- Callback service: Routes the call to the target.
- Callback service terminates.

Create your Scenario

In the **Admin UI > Services > Configured Services** tab, add a Callback service with one of the `User-Terminated` scenario as the **Common Default Configuration** (see Adding a Service for details).

Enter the service name, which is the callback execution name of your service. It will be used in URLs

to access this service. For example, if you set this name to `voice-userorig-immediate`, your service URL will be:
`http://host:port/{base-web-application}/service/callback/voice-userterm-immediate`

When you add this service and default configuration, many options are automatically populated with the appropriate default values.

Configure the Scenario

The following parameters are mandatory in the configuration of your callback service and in the parameters of the requests that your application submits:

Option	Description
_customer_number Normally provided as a request parameter	Customer's phone number. The parameter is mandatory to match the call with service data when the call direction is set to `USERORIGINATED`. Also used to establish the connection with the customer when the call direction is `USERTERMINATED`. **Important** This is a request parameter that you can use in REST queries. **This option is mandatory.**
_call_direction = USERTERMINATED	This is a default value, automatically populated when using the predefined User-Terminated scenario. You do not need to change this value. • If this option is set to `USERORIGINATED`, the customer's device will initiate the call to get connected to the agent. • If this option is set to `USERTERMINATED`, the agent or the system will initiate the call to contact the customer.
_wait_for_user_confirm = false	True to wait for confirmation of the customer's availability. If this option is set to `true`, the service sends a push notification to the customer's device to get confirmation that the customer is ready to have a conversation with the agent. This scenario is possible only if the _wait_for_agent option is set to true.

Include the Originating Call in Callback Historical Reporting

To connect the originating call to the Callback Historical Reporting, the IVR should add the following Callback KVPs to the attached data of the originating call.

- _CB_SERVICE_ID: Set this value to the GMS Service ID. This value is returned in the _id key in the response to the callback service request.

- _CB_T_SERVICE_START:

  - If the callback was not scheduled, set this value to the time at which the callback service was requested.

  - For a scheduled callback, set this value to the scheduled time. The required format is number of seconds since January 1, 1970, 00:00:00 UTC; for example, you can get this value in javascript as follows: Math.floor((new Date()).getTime()/1000)

- _CB_T_CALLBACK_ACCEPTED: Set to the time at which the caller accepted the callback. The format is the same as for _CB_T_SERVICE_START.

> ### Important
>
> The callback strategy includes optional support for adding these attached data keys to the originating call. To enable this support, include the parameter _originating_interaction_id (set to the originating call interaction ID) in the HTTP request that starts the Callback service.

Terminate the Originating Call

After you have requested a callback and set the required attached data, the originating call might be set to the ABANDONED state. To avoid this issue, apply the following customization code to the SCXML of the Classic Callback sample that includes a Composer project.

```
<script>
    var reason = { ReasonSystemName : "Callback" };
    var hints = { 'reasons' : reason };
</script>
<ixn:redirect requestid="somevariable" interactionid="InteractionID"
from="routingPoint" to="'__STOP__'"
type="_genesys.queue.rType.RouteTypeReject" hints="hints"/>
```

Capacity

The Capacity Service enables you to define the number of scheduled callbacks that are allowed for Callback for a given time slot in the week. Then, your Callback service refers to your Capacity service and to your Office Hours service to adjust the agent availability and the number of scheduled callbacks. You can also implement exceptions that allow you to set a specific capacity for a given date.

If your Callback Service needs to define its scheduled callback capacity, you must map the _capacity_service parameter value with the name of the Capacity service that you have created. Depending on the defined capacity and on the defined business hours, the number of scheduled callbacks during certain days or hours will increase or decrease.

> ### Important
>
> Callback services that need fixed capacity levels can continue to use the _max_request_by_time_bucket option. But, if your Callback service includes both _capacity_service and _max_request_by_time_bucket options, then _max_request_by_time_bucket is ignored.

REST API

The Capacity service is similar to Office Hours and is accessible through REST API for external queries.

Refer to the API Reference for further details.

Configuration

Option	Value	Comment
_type	builtin	Mandatory option.
_service	capacity	Mandatory
_capacity_*	JSON-formatted String	Multiple properties that start with a prefix _capacity_ and describe capacity allocation through the course of the week. The JSON structure specifies the day of the week, and capacity for hourly slots during that day. Days of the week are numbered as recommended by ISO-8601 from

Option	Value	Comment

		1 (Monday) to 7 (Sunday).
		
		`_capacity_1 : {`
		`  1 : { // Monday`
		`    "0900" : 5,`
		`    "1000" : 7,`
		`    "1100" : 10,`
		`    "1200" : 10 }`
		`}`
		`_capacity_2 : {`
		`  2 : { // Tuesday`
		`    "0900" : 3,`
		`    "1000" : 5,`
		`    "1100" : 7,`
		`    "1200" : 7 }`
		`}`

_capacity_add*	JSON-formatted String	Multiple properties that start with the _capacity_add_ prefix and describe the capacity exceptions for additional working days.
		
		The format is similar to the format of _capacity_* properties but instead of a weekday, the full date for the extra day is used to prefix the capacity exception. This date is entered in the format of yyyyMMdd (year, month, day of the month).
		
		`_capacity_add_20160508 : {`
		`  20160508 : { // May 8, 2016`
		`    "0900" : 5,`
		`    "1000" : 7,`
		`    "1100" : 10 }`
		`}`
		`_capacity_add_20161111 : {`
		`  20161111 : { // November 11, 2016`
		`    "0900" : 3,`
		`    "1000" : 5,`
		`    "1100" : 7 }`
		`}`

_timezone		Timezone for your capacity service. For instance, if you configured "EST", or "PST" timezones with the CME, your parameters must use the timezones defined for Java such as "America/Toronto", or "Europe/Paris". See here Wikipedia to get the list of correct timezones.

Pausing Callback

For certain situations, you can temporarily pause outbound calling and routing to agents with callbacks that have reached the QUEUED state. You can set this up globally to suspend all callback services, or individually for particular services. This feature is useful, for example, to deal with a fire alarm where all agents need to exit the facility.

Consider the following:

- For callbacks where the outbound call has **not yet** connected to the customer, pausing these calls will disable outbound calling altogether.

- For callbacks where the outbound call has **already** connected to the customer, pausing the callback disables routing the call to the agent; the customer will continue listening to the "waiting for next available agent" prompt. As soon as you unsuspend the callback service, normal operation resumes.

- The state of paused callbacks will be **PAUSED** in the Callback UI.

Set up the list for the callback pause service

You can control how and when callbacks are paused using a transaction list entry in the **Transactions** folder of your tenant configuration.

Enter the name you want to use for the list of paused services by setting the _paused_services_list option in your callback service. If you don't specify a value for this option, the name GMS_Paused_Services is used by default.

Enable callback pause (globally or per individual service)

The **Transactions** folder of your tenant configuration includes an object called GMS_Paused_Services, of type list. In the **Annex** of this object, add the following settings to the section called **services**.

- Pause callback **globally** for all services by adding the following key:
    key: all

    value: true (pauses calling), false (enables calling)

- Pause callback for individual services based on the service name, by adding a key for each service:
    key: The service name.

    value: true (pauses calling), false (enables calling)

- Pause callback for **individual** services based on IDs, by adding an identification text passed in the HTTP request that starts the callback:
    key: Identification text, where the text value is passed in the option _paused_services_id of the HTTP request. This key can contain any alphanumeric characters, except the blank character.

value: `true` (pauses calling), `false` (enables calling)

If any of the keys ("all", callback service name, or _paused_services_id) is found in the transaction list with its value set to `true`, the system pauses the service.

> ## Tip
>
> If you anticipate using this feature, be careful that active callbacks do not expire while in the PAUSED state. For example, make sure that the callback service options values are all set to a longer time period than you expect the callbacks to be paused: _ttl, _urs_queued_ttl, and _max_time_to_wait_for_agent_on_the_call.

Preview and Disposition Scenarios

If you implement a custom agent desktop and wish to integrate the Preview and Disposition scenarios into your Callback application, you need to configure preview and disposition options in your callback service. If you implement the Disposition scenario, you also need to create an Office Hours service.

After you do this, your custom agent application will receive the following `UserEvent` events from Orchestration Server:

- `CallbackInvitationEvent`—The callback invitation that contains the attached data for the preview. The invitation includes the list of actions that the agent can perform–accept, reject, or cancel. Your Agent application displays the actions and the attached data for the preview to the agent, then submits a Preview Response to your Callback service.

- `CallbackDispositionEvent`—The callback disposition event that provides the URL to which you submit the disposition selected by the agent. Your Agent application then submits a Disposition Response to your Callback service through this URL.

If the agent needs to reschedule the callback, he or she can select the `retry_later` disposition option provided in the disposition event. This event also provides the `business_hours_url` that lists the available timeslots to reschedule.

Preview and Disposition Event Flow

This diagram shows the sequence of events involved in your callback service and in your Agent Desktop application if the agent receives a preview invitation and submits a disposition call after the call is done.
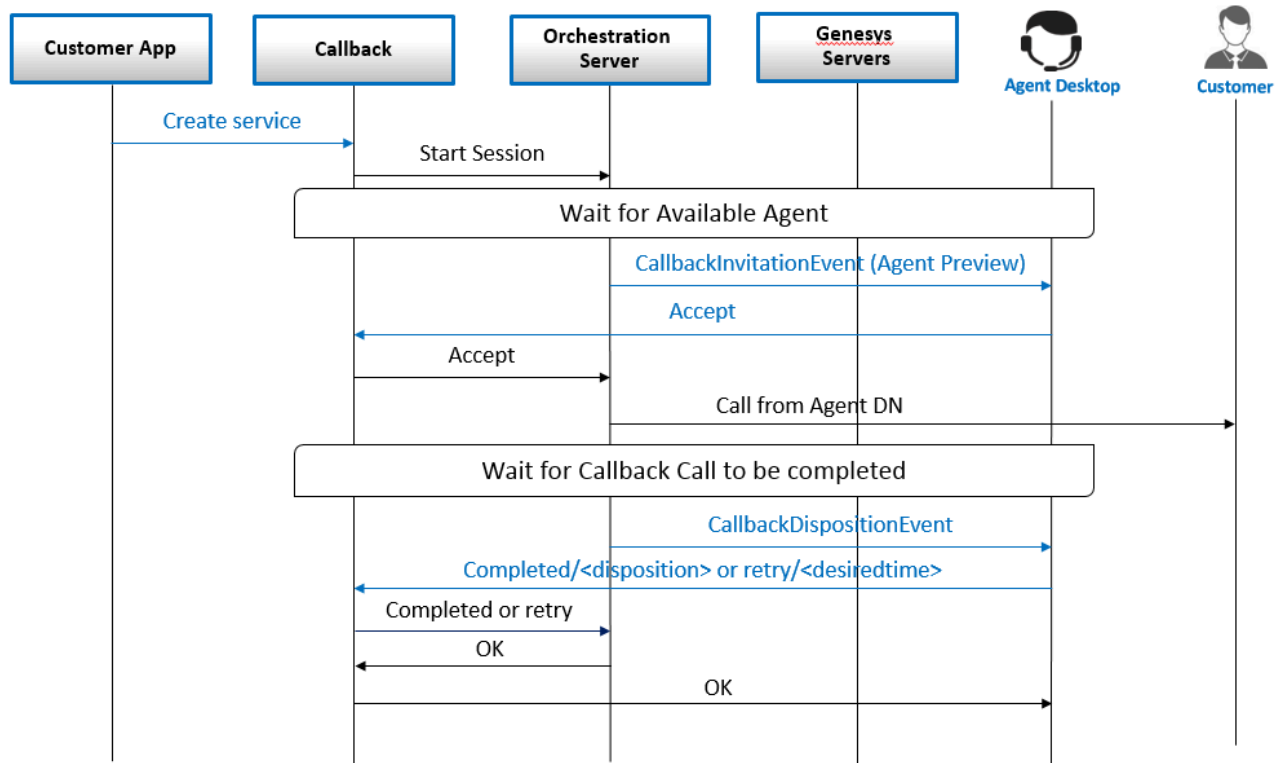
Steps to Handle a Preview Request

If you configure the preview feature, as detailed below, the Callback service sends a preview request to your Agent Desktop application. This request is a user event that includes the preview-specific data in the event's udata and extensions attributes. Your Agent Desktop application can use this data to:

- Show the available action(s) in the preview dialog.

- Use the provided URLs to submit their response(s) to the action(s) they select.

Configure the Preview Request Feature

To implement a preview scenario, check your settings:

1. Add an ORS Application (CME) connection to the agent's switch T-Server by using Configuration Manager or Genesys Administrator Extension (GAX).

2. In the Service Management UI, set the callback service options to the following values:

    - _agent_preview=true to enable the preview feature

    - _preview_userevent_mediatype = 0 to set the voice media (default)

    - _agent_preview_allow_reject = [0..n]

- If _agent_preview_allow_reject=0, the reject option will not be displayed to agents.

  - Any other values indicate how many times the call can be rejected by agents, to make sure that the call cannot be rejected indefinitely.

- _agent_preview_timeout = 30—Specify the number of seconds that the agent will be given before the preview times out. If the limit is exceeded, GMS assumes that the agent has rejected the request. (Note that the value shown here is only an example.)

- _urs_udata_xfer_keys = "" and _attach_udata = ""–List of KVPs available within the user-data key of the CallbackInvitationEvent event.

> **Important**
>
> Starting in 8.5.105, the user data contains all of the KVPs attached to the real call.

Wait for the CallbackInvitationEvent

If the Preview Request Feature is configured, ORS will use the Tlib/UserEvent protocol to send a preview request invitation for Callback as a CallbackInvitationEvent UserEvent to your desktop application.

- **UserEvent Name:** CallbackInvitationEvent

- **Limitations:** The expected target is an Agent DN.

This event includes the response-options list of actions that the preview invitation dialog should display to the agent: accept, reject, or cancel (xcancel). For each action, the url property indicates the URL that the event response will be submitted to.

Example of CallbackInvitationPreview UserEvent

```
@12:38:49.5410 [0] 8.1.101.10 distribute_user_event: message EventUserEvent
   AttributeEventSequenceNumber    00000000000018eb
   AttributeCustomerID    'Environment'
   AttributeTimeinuSecs    541000
   AttributeTimeinSecs    1467229129 (12:38:49)
   AttributeReferenceID    2223
   AttributeUserData    [1438] 00 03 03 00..
      'display-data'(list) '1'    'v1'
                           '2'    'v2'
                           '3'    'v3'
      'response-options'(list) 'accept'(list) 'button'    'accept'
         'url'    'http://2XX.XX.1XX.1XX:8080/genesys/1/ors//scxml/session/
UGHKD84S9H6M995N2PQV7OFN3000002F/request/accept_preview'
      'reject'(list) 'button'    'reject'
      'url'    'http://2XX.XX.1XX.1XX:8080/genesys/1/ors//scxml/session/
UGHKD84S9H6M995N2PQV7OFN3000002F/request/reject_preview'
      'xcancel'(list) 'button'    'xcancel'
      'url'    'http://2XX.XX.1XX.1XX:8080/genesys/1/ors//scxml/session/
UGHKD84S9H6M995N2PQV7OFN3000002F/request/cancel_preview'
      'user-data'(list) 'GMS_Call_Direction'    'USERTERMINATED'
         'GMS_Cb_Desired_Time'    '2016-06-29T19:38:39.788Z'
         'GMS_Cb_Type'    'immediate'
         'GMS_Customer_Number'    '5115'
```

```
        'GMS_Orig_Service_ID'    'test123'
        'GMS_ROUTABLE'    '0'
        'GMS_RP_Used'    '8999@SIP_Switch'
        'GMS_ServiceName'    'samples_dev'
        'GMS_Service_Data_ID'    '445-748c2c27-2010-47f7-91cc-49d19a7734c3'
        'GMS_Service_ID'    '445-36922082-f275-4d44-8200-92bc5c49a965'
        'GMS_Target_Selected'
            '{"agent":"KSippola","dn":"7001","id":"Customer_Service",

"place":"SIP_Server_Place1","resource":"7001","return":"target","stat_value":"0",
            "switch":"SIP_Switch","type":"GA","vq":"SIP_VQ_SIP_Switch"}'
        'GMS_UserTerminated_First_Connect_Party'    'AGENT'
        'GMS_VQ_Used'    'SIP_VQ_SIP_Switch'
        'RPVQID'    'I5Q83URTDD0FT2BTG0M03PT0P40000DT'
        'RTargetAgentGroup'    'Customer_Service'
        'RouterData70'    '("t"="1467229122 158")'
        'first_name'    'John'
        'last_name'    'Doe'
        'location_lat'    '37.2638324'
        'location_long'    '-122.02301459999998'
    AttributeExtensions    [118] 00 04 00 00..
        'event'    'CallbackInvitationEvent'
        'invitation-timeout'    '60'
        'session'    'UGHKD84S9H6M995N2PQV7OFN3000002F'
        'source'    'ORS'
    AttributeMediaType    '0'
    AttributeUserEvent    EventUserEvent
    AttributeThisDN    '7001'
```

[+] Show CallbackInvitationPreview as a JSON schema

```
// JSON Schema for udata passed in CallbackInvitationPreview userevent
{
    "title": "CallbackInvitationPreview Udata",
    "type": "object",
    "properties": {
        "response-options": {
            "type": "object",
            "properties": {
                "responseOption1": {
                    "type": "object",
                    "properties": {
                        "button": "accept",
                        "url": "…/request/accept_preview"
                    }
                },
                "responseOption2": {
                    "type": "object",
                    "properties": {
                        "button": "reject",
                        "url": "…/request/reject_preview"
                    }
                },
                "responseOption3": {
                    "type": "object",
                    "properties": {
                        "button": "xcancel",
                        "url": "…/request/cancel_preview"
                    }
                },
                "required": ["responseOption1", "responseOption3"]
            }
```

```
        },
        "display-data": {
            "type": "array",
            "items": {
                        "type": "string"
                }
        },
        "user-data": {
            "type": "object",
            "properties": {
                        . . .
                }
        }

    },
    "required": ["response-options", "display-data"]
}

// JSON Schema for extentions passed in CallbackInvitationPreview userevent
{
    "title": "CallbackInvitationPreview Extensions",
    "type": "object",
    "properties": {
        "source": {
            "enum": ["ORS"]
        },
        "event": {
            "enum": ["CallbackInvitationEvent"]
        },
        "session": {
            "type": "string"
        },
        "invitation-timeout": {
            "type": "string" // _agent_preview_timeout
        }
    },
    "required": ["source", "event", "session", "invitation-timeout"]
}
```

Send Preview Response

The Agent Desktop application must send the Preview Response after the agent selects one of the
actions presented in the agent preview dialog.

- **Description:** Preview response

- **Protocol:** HTTP

- **Method:** POST

- **URL:** Value of the url property received in the CallbackInvitationEvent user event corresponding to
  the agent-selected option.

- **Content-type:** application/json

- **Body**: None

- Response from callback: HTTP 200 - Response received: (Accept|Reject|Cancel)

Reduce Processing Delays in Scheduled Preview Callback Scenario

When GMS schedules a preview callback, at a given time T, GMS sends the request to ORS at time T and ORS sends the CallbackInvitationEvent request to SIP Server at T+60 up to T+90 seconds. You can use the following options to reduce the processing delay:

- Configure _wait_for_user_confirm = `false` to disable the user confirmation for this service.

- Configure _max_number_of_user_availability_confirmation_attempts to limit the number of retries in your Callback service.

- Configure _user_confirm_timeout to a smaller value to reduce the waiting time for the user confirmation timeout.

Steps to Handle the Disposition Response

If you configure the Disposition feature, the Callback service will send a disposition request to the Agent Desktop application. The Agent Desktop application uses information retrieved from the user event data and the extension attributes to display the disposition dialog to the agent. It then replies with the disposition selected by the agent, as discussed in greater detail in the Disposition response section.

> ### Important
> You must create an Office Hours service before configuring the Disposition feature.

Configure the Disposition Feature

To enable the Disposition feature, configure the following settings:

1. Add an ORS Application connection to the Agent Switch T-Server Application in Configuration Manager or in Genesys Administrator Extension.

2. In the Service Management UI, set the callback service options to the following values:

   - _enable_disposition_dialog = `true` to enable the disposition dialog; this option is `false` by default to disable the feature.

   - _disposition_userevent_mediatype = 0 (voice is default)

   - _agent_disposition_timeout = 45 to set the duration in seconds for receiving an agent response. If this time elapses with no reply from the agent, GMS sets the disposition to COMPLETED with the reason NO_AGENT_DISPOSITION.

   - Set the _business_hours_service option with the name of your Office Hours service.

Wait for the Disposition Request

Once the callback is completed, if the disposition feature is enabled, GMS sends a Disposition request as a CallbackDispositionEvent UserEvent using the Tlib protocol:

- **Description:** Disposition request

- **Protocol:** Tlib/UserEvent

- **UserEvent Name:** CallbackDispositionEvent

- **Limitations:** The expected target is an Agent DN

- **Body:** Contains udata and extensions.

```
12:39:05.224 Trc 04541 RequestDistributeUserEvent received from [628] (00000004
Orchestration_Server 216.38.144.133:34716)
message RequestDistributeUserEvent
    AttributeThisDN      '7001'
    AttributeUserEvent      EventUserEvent
    AttributeCommunicationDN     '7001'
    AttributeMediaType      '0'
    AttributeExtensions     [121] 00 04 00 00..
        'disposition-timeout'    '120'
        'event'      'CallbackDispositionEvent'
        'session'     'UGHKD84S9H6M995N2PQV7OFN3000002F'
        'source'      'ORS'
    AttributeUserData     [665] 00 03 03 00..
        'config'(list) '_gms_external_base_url'    'http://216.38.144.133:8080/'
                       '_service_name'     'samples_dev'
        'display-data'(list) 'business_hours_url'     '$_gms_external_base_url$genesys/1/
service/callback/$_service_name$/availability?start=$desired_time$#ber-of-days=1&max-time-
slots=5'
                            'completed_dispositions'
'SUCCESS,BUSY,NO_ANSWER,SIT_TONE,ANSWERING_MACHINE,CUSTOMER_NOT_KNOWN,CUSTOMER_NOT_PRESENT,WRONG_CUSTOMER_NUMBE
                            'customer_number'    '5115'
        'response-options'(list) 'done'(list) 'button'     'done'
            'options'     '["completed","retry_now","retry_later"]'
            'url'     '$_gms_external_base_url$genesys/1/ors/scxml/session/
UGHKD84S9H6M995N2PQV7OFN3000002F/request/callback_disposition'
    AttributeReferenceID     2228
```

In this UserEvent:

- `display-data` contains the information that the Agent Desktop application needs to present the agent disposition dialog.

- `business_hours_url` provides the list of available timeslots in case the agent selects the `retry_later` disposition option.

- `completed_dispositions` contains the list of common dispositions. The Agent Desktop can provide its own list of dispositions.

- `url` contains the URL to be used for submitting the disposition response.

[+] Show CallbackDispositionEvent as a JSON schema

Note that the JSON schema below is provided for the sake of clarity, as this event is **always** sent as a UserEvent.

```
// JSON Schema for udata passed in CallbackDispositionEvent UserEvent
{
    "title": "CallbackDispositionEvent Udata",
    "type": "object",
    "properties": {
        "response-options": {
            "type": "object",
```

```
            "properties": {
                "responseOption1": {
                    "type": "object",
                    "properties": {
                        "button": "done",
                        "url": "…/request/callback_disposition",
                        "options": {
                            "enum": ["completed", "retry_now", "retry_later"]
                        }
                    },
                    "required": ["responseOption1"]
                }
            },
            "display-data": {
                "type": "object",
                "properties": {
                    "customer_number": "string",
                    "business_hours_url": "/service/
callback/$_service_name$/availability?start=$desired_time$&number-of-days=1&max-time-slots=5",
                    "completed_dispositions": {
                        "enum": "array",
                        "properties": {
                            "button": "done",
                            "url": "…/request/callback_disposition",
                            "options": {
                                "enum": ["SUCCESS", "BUSY", "NO_ANSWER", "SIT_TONE",
"ANSWERING_MACHINE", "CUSTOMER_NOT_KNOWN", "CUSTOMER_NOT_PRESENT", "WRONG_CUSTOMER_NUMBER"]
                        }
                    },
                    "required": ["customer_number", "business_hours_url",
"completed_dispositions"]
                }
            },
            "config": { // to be used to evaluate business_hours_url specified earlier
                "type": "object",
                "properties": {
                    "_gms_external_base_url": "string",
                    "_service_name": "string"
                },
                "required": ["_gms_external_base_url", "_service_name"]
            }
        },
        "required": ["response-options", "display-data"]
}
// JSON Schema for extensions passed in CallbackDispositionEvent UserEvent
{
    "title": " CallbackDispositionEvent Extensions",
    "type": "object",
    "properties": {
        "source": {"enum": ["ORS"]},
        "event": {"enum": ["CallbackDispositionEvent "]},
        "session": {"type": "string"},
        "disposition-timeout": {"type": "string"}},
    "required": ["source", "event", "session", "disposition-timeout"]
}
```

Send a Disposition Response

The Agent Desktop application must send the Disposition Response after the agent selects one of the dispositions presented in the agent disposition dialog and before the _agent_disposition_timeout

timeout occurs.

- **Description:** Disposition response

- **Protocol:** HTTP

- **Method:** POST

- **URL:** Value of the url property received in the CallbackDispositionEvent UserEvent

- **Content-type:** application/json

- **Body:**

```
// JSON Schema for agent response from callback disposition dialog
{
        "title": "HTTP POST body received from callback disposition dialog",
        "type": "object",
        "properties": {
                "disposition": {
                        "enum": ["retry_later","retry_now", "completed"],
                },
                "customer_number": "string",
                "desired_time": "string"
        },
        "required": ["disposition"] // desired_time required when disposition = "retry_later"
}
```

Callback Response:

```
HTTP 200 Response received: <agent response from body of the request>
```

Callback Service Options Reference

Last update: 8.5.109.05, 8.5.110.00

Generic Options

These options are shared for some GMS Services, including Callback.

- _type
- _service
- _service_type
- _agent_timeout_notification_message
- _booking_expiration_timeout
- _chat_400_response_on_disconnected
- _client_timeout_notification
- _client_timeout_notification_message
- _filtering_chat_events
- _mandatory_customer_lookup_keys
- _return_pool_health

> **Important**
>
> In 8.5.109, new options have been added to the Callback Options Reference. This content is under development and might not be comprehensive or completely up to date.

General

_paused_services_id

Section: General
Default Value:
Valid Values: string

Changes Take Effect: Immediately
Introduced: 8.5.109.05

ID of the service in the Paused Service Transaction List. If you set this option, its value is used to look up the service in the Paused Service Transaction List. If you do not set any value, the service name is used instead.

> **Tip**
>
> This is an advanced parameter. To modify the value of an advanced parameter, you must enable **Advanced Parameters** in the Service Management UI.

_paused_services_list

Section: General
Default Value: GMS_Paused_Services
Valid Values: string
Changes Take Effect: Immediately
Introduced: 8.5.109.05

Name of the Transaction list entry that controls pausing of callbacks.

> **Tip**
>
> This is an advanced parameter. To modify the value of an advanced parameter, you must enable **Advanced Parameters** in the Service Management UI.

_max_transfer_to_agent_attempts

Section: General
Default Value: 5
Valid Values: integer
Changes Take Effect: Immediately

This option is mandatory.

Maximum number of attempts to transfer the call to the agent. If greater than 1, set the URS option on_route_error=try_other.

_agent_transfer_confirm_timeout

Section: General
Default Value: 0
Valid Values: integer
Changes Take Effect: Immediately

Maximum time to wait for call events (`interaction.partystatechanged`) related to the transfer of the call to the agent. If a transfer confirming or failing event is not received within this period, the transfer is assumed to have succeeded. Set this option value to zero to disable the timer. This timer should be enabled in rare conditions where the switch does not send confirmation events after the call redirection to an agent.

> ### Tip
> This is an advanced parameter. To modify the value of an advanced parameter, you must enable **Advanced Parameters** in the Service Management UI.

This option is mandatory.

_user_confirm_timeout

Section: General
Default Value: 30
Valid Values: integer
Changes Take Effect: Immediately

Period in seconds to receive the customer response. If the customer does not reply in time, the system assumes that the customer did not reply: the notification was not received or was received but ignored.

This option is mandatory.

_orig_service_id

Section: General
Default Value:

Valid Values: string
Changes Take Effect: Immediately

Id of the service that originated the callback service request. For example: IVR service, web session service, and so on.

_snooze_duration

Section: General
Default Value: 300
Valid Values: integer
Changes Take Effect: Immediately

Period in seconds to snooze agent notifications. When the customer receives the agent availability notification, he or she can snooze the notification for the specified duration.

This option is mandatory.

_ixn_redirect_confirm

Section: General
Default Value: true
Valid Values: boolean
Changes Take Effect: Immediately

Enables agent confirmation for accepting the call. If you set this option to `false`, the system will not wait for confirmation when a call is redirected. You should set this option to false in rare conditions where the switch does not send confirmation events (such as `interaction.partystatechanged`) after the call redirection to an agent.

> Tip
> This is an advanced parameter. To modify the value of an advanced parameter, you must enable **Advanced Parameters** in the Service Management UI.

This option is mandatory.

_max_time_to_wait_for_ixn_delete

Section: General
Default Value: 3600
Valid Values: integer
Changes Take Effect: Immediately

Duration (in seconds) that the callback in PROCESSING state will keep alive the interaction to be deleted. This value should be greater than 5 seconds and less than the specified _ttl value. To set this value, consider the requirements for features like the disposition dialog and the agent notifications that need GMS Callback to be running while callback is still in PROCESSING state.

> ### Tip
> This is an advanced parameter. To modify the value of an advanced parameter, you must enable **Advanced Parameters** in the Service Management UI.

_retain_session_until_ixn_deleted

Section: General
Default Value: false
Valid Values: boolean
Changes Take Effect: Immediately
Modified: 8.5.108.02

Retains the session until the interaction is deleted. Setting this parameter to true will enable the callback SCXML session to remain until the callback call is terminated. If set to false, and if the agent disposition is not enabled, the callback session will end after the call is connected to an agent. The latter is required if the agent will be transferring the call to another SCXML strategy.

_ttl

Section: General
Default Value: 86400
Valid Values: integer
Changes Take Effect: Immediately
Modified: 8.5.107.15, 8.5.109.05

Duration (in seconds) for which the service will be kept in storage after the Desired Time is passed (Time To Live). Once expired, the service is removed from the system. For example, if you want

the callbacks to be visible in the Service Management UI for one week past the execution time, then you should set 7 days of Time To Live, which means `_ttl=604800`.

This option is mandatory.

_capacity_service

Section: General
Default Value:
Valid Values: string
Changes Take Effect: Immediately

Sets a configured capacity service. Agent availability is verified against the defined capacity and capacity exceptions.

_mandatory_customer_lookup_keys

Section: General
Default Value: _customer_number
Valid Values: string
Changes Take Effect: Immediately

Sets the comma-separated list of mandatory customer lookup keys that must be provided in the callback schedule request. This list can contain only identification keys such as phone numbers, user names, and so on.

_business_hours_service

Section: General
Default Value: No default value
Valid Values: String
Changes Take Effect: Immediately

Name of the office hours service used to provide the available time slots for Callback. The Request Desired Time is verified against the defined regular and specific calendar hours.

_call_direction

Section: General
Default Value: USERORIGINATED
Valid Values: USERORIGINATED, USERTERMINATED
Changes Take Effect: Immediately

This is a default value, automatically populated when using the predefined User-Terminated scenario. You do not need to change this value.

- If this option is set to USERORIGINATED, the customer's device will initiate the call to get connected to the agent.

- If this option is set to USERTERMINATED, the agent or the system will initiate the call to contact the customer.

_media_type

Section: General
Default Value: voice
Valid Values: String
Changes Take Effect: Immediately

Media type of the interaction that the service is expected to handle. This option enables URS to select an agent who has the appropriate media capabilities. This is a default value, automatically populated when using the predefined User-Terminated scenario. You do not need to change this value.

This option is mandatory.

_wait_for_agent

Section: General
Default Value: true
Valid Values: Boolean
Changes Take Effect: Immediately

True to wait for an agent to connect. If this option is set to true, the service will wait for the agent to initiate the interaction and to send the notification to the customer. If the option is set to false, the interaction can start right after the creation of the service instance. In voice scenarios, the access information will be returned immediately with the service ID.

This option is mandatory.

_wait_for_user_confirm

Section: General
Default Value: false
Valid Values: Boolean
Changes Take Effect: Immediately

True to wait for confirmation of the customer's availability. If this option is set to true, the service sends a push notification to the customer's device to get confirmation that the customer is ready to have a conversation with the agent. This scenario is possible only if the _wait_for_agent option is set to true.

_max_ors_submit_attempts

Section: General
Default Value: 3
Valid Values: Integer
Changes Take Effect: Immediately

Maximum number of times that the request for execution will be submitted to ORS. When this number is reached, the request is removed from the persistent queue and discarded.

_exceptions

Section: General
Default Value:
Valid Values: String
Changes Take Effect: Immediately

List of the exception patterns that should be verified before processing this callback request. See the Pattern configuration for details.

_customer_lookup_keys

Section: General
Default Value: _customer_number
Valid Values: String
Changes Take Effect: Immediately

Comma-separated list of properties to track back customer callback requests. You must only specify identification numbers such as phone numbers, user names, and so on.
For example: `_customer_number,_phone_number`

_ixn_redirect_hints

Section: General
Default Value:
Valid Values: JSON-formatted string
Changes Take Effect: Immediately

Sets the `hints` parameter of the scxml `ixn:redirect` request for routing the callback call to the agent. You can use this parameter to override the customer phone number presented to the agent. Use this parameter if the prefix applied to outbound numbers should not be displayed to the agent. For example, the outbound number displayed to the agent can be set as follows:

`{"extensions": {"CPNDigits": "5551234"}}`

> Tip
>
> This is an advanced parameter. To modify the value of an advanced parameter, you must enable **Advanced Parameters** in the Service Management UI.

No results

_attach_udata

Section: General
Default Value: single_json
Valid Values:
Changes Take Effect: Immediately

- `separate_keys` – Each KVP data is attached as a separate key-value pair to the user event.

- `single_json` – All KVPs are attached as a single stringified-JSON object to the `GMS_UserData` key in the user event.

- `gms_storage_id` – Callback will attach the GMS service ID to the `GMS_UserData` userdata key and let the agent application retrieve the data from GMS.

_callback_events_list

Section: General
Default Value:
Valid Values: String
Changes Take Effect: Immediately

Name of the Transaction List object defining the Callback events and data to notify; for example, `GMS_Events`.

_enable_status_notification

Section: General
Default Value: false
Valid Values: false, subscribe_notify, notify
Changes Take Effect: Immediately

Enables the callback status notifications and defines the notification subscription for the Callback application.

- `false` – The Callback Service assumes that the client application has already subscribed for status notifications.

- `subscribe_notify` – The callback application will subscribe for status notification on behalf of the client application.

- `notify` – The client application has already subscribed for status notifications.

This option will override the settings in the GMS events transaction list.

> **Tip**
>
> This is an advanced parameter. To modify the value of an advanced parameter, you must enable **Advanced Parameters** in the Service Management UI.

This option is mandatory.

_status_notification_type

Section: General
Default Value:
Valid Values: httpcb, orscb
Changes Take Effect: Immediately

Notification type to use when publishing the notification. You should use `httpcb` in most cases. Use `orscb` to publish notifications in Orchestration Server for advanced ORS-based customizations.

> **Tip**
>
> This is an advanced parameter. To modify the value of an advanced parameter, you must enable **Advanced Parameters** in the Service Management UI.

_status_notification_target

Section: General
Default Value:
Valid Values: String
Changes Take Effect: Immediately

Target to which to publish notifications according to the `_status_notification_type` notification type:

- If `_status_notification_type` = httpcb, then set `_status_notification_target` to the target URL to which to push the notifications.

- If `_status_notification_type` = orscb, then set `_status_notification_target` to the target

ORS session ID.

> **Tip**
>
> This is an advanced parameter. To modify the value of an advanced parameter, you must enable **Advanced Parameters** in the Service Management UI.

This option is mandatory.

_status_notification_provider

Section: General
Default Value: ""
Valid Values: "" or <provider name>
Changes Take Effect: Immediately

Custom HTTP notification provider name. Leave blank to set the default provider.

> **Tip**
>
> This is an advanced parameter. To modify the value of an advanced parameter, you must enable **Advanced Parameters** in the Service Management UI.

This option is mandatory.

_max_time_to_wait_for_agent_on_the_call

Section: General
Default Value: 3600
Valid Values: Integer (seconds)
Changes Take Effect: Immediately

Maximum period of time (seconds) to wait for the agent to accept and answer the call after the customer is connected.

Chat section

_chat_endpoint

Section: Chat
Default Value:
Valid Values: string
Changes Take Effect: Immediately

Chat Server endpoint to create the chat interaction and to handle the service request. This endpoint should exist and should be specified in the following format: <tenant>:<endpoint>. The specified endpoint should be configured to trigger the GMS Chat Inbound Service. If you do not set this option, the system uses the `default_chat_endpoint` option of the chat section as the endpoint.

Notification section

_max_notify_delivery_attempts

Section: Notification
Default Value: 3
Valid Values: integer
Changes Take Effect: Immediately

Maximum number of attempts to send notifications to the user before the contact request is considered failed. Ignored if push notification properties are not specified.

> ### Tip
> This is an advanced parameter. To modify the value of an advanced parameter, you must enable **Advanced Parameters** in the Service Management UI.

_max_number_of_user_availability_confirmation_attempts

Section: Notification
Default Value: 3
Valid Values: integer
Changes Take Effect: Immediately

Maximum number of times the system will send agent availability notifications to the customer before the contact request is considered failed. Ignored if the push notification properties are not specified.

> **Tip**
>
> This is an advanced parameter. To modify the value of an advanced parameter, you must enable **Advanced Parameters** in the Service Management UI.

_notification_message_file

Section: Notification
Default Value: /genesys/1/document/service_template/callback/Resources/Strings/messages.json
Valid Values: url
Changes Take Effect: Immediately

Specifies the URL of the notification message file which contains the externalized strings to be displayed to the customer's mobile.

> **Tip**
>
> This is an advanced parameter. To modify the value of an advanced parameter, you must enable **Advanced Parameters** in the Service Management UI.

more...

_use_debug_push_certificate

Section: Notification
Default Value:
Valid Values: boolean
Changes Take Effect: Immediately

Use debug certificates for the push notification provider

_device_os

Section: Notification
Default Value:
Valid Values: string
Changes Take Effect: Immediately

OS running on the Mobile device. This is used for push notifications.

_provider_name

Section: Notification
Default Value:
Valid Values: string
Changes Take Effect: Immediately

Name of the notification provider. This is used for push notifications.

_device_notification_id

Section: Notification
Default Value:
Valid Values: string
Changes Take Effect: Immediately

Notification Id of the device to which the notifications need to be pushed.

_agent_availability_notification_delay

Section: Notification
Default Value: 30
Valid Values: integer
Changes Take Effect: Immediately

Duration to wait prior to sending the agent availability notification. This duration allows the customer application to display the user confirmation before the push notification dialog takes over the screen.

This option is mandatory.

Queue Management section

_enable_throttle_header_parameters

Section: Queue Management
Default Value: true
Valid Values: boolean
Changes Take Effect: Immediately
Introduced: 8.5.111.04

If true, applies the throttling to the header parameters that are configured in the
_throttle_request_parameters option. You can configure two types of parameters in the
_throttle_request_parameters option:

- Callback request body's parameters

- Callback request header's parameters.

If the same parameter name exists in both header and body with different values, and if this
option is set to true (default), the system applies the throttling to the header value and ignores
the body value.

For example, let's consider _throttle_request_parameters=_customer_number, user_id,
where user_id is a parameter passed in the Callback request header and where
_customer_number is passed in the request as a body parameter. If
_enable_throttle_header_parameters=true, by default, the system will allow a maximum of 8
immediate callback requests issued from the same user_id per day (24 hours). If set to false,
there is no restriction for the callback requests issued from the same user_id.

_enable_in_queue_checking

Section: Queue Management
Default Value: true
Valid Values: true, false, strict
Changes Take Effect: Immediately
Introduced: 8.5.109.05
Modified: 8.5.111.04

- If true, prevents the callback creation if the same customer number has already two callbacks in the queue.

- If strict, prevents the callback creation if the same customer number has already one callback in the queue.

- If false, does not check whether the customer is already in queue when creating the callback.

This option applies to both immediate and scheduled callbacks.

_throttle_request_parameters_limit

Section: Queue Management
Default Value: 8
Valid Values: integer
Changes Take Effect: Immediately
Introduced: 8.5.110.07
Modified: 8.5.111.08
Related Options: _throttle_request_parameters

Specifies the maximum number of immediate callback requests that a given customer number (or any other parameters configured in _throttle_request_parameters) can query per day. You can set the list of parameters to apply this throttling in the option _throttle_request_parameters.

In versions older than 8.5.111.08, the default value is 6.

_throttle_callbacks_per_service_1

Section: Queue Management
Default Value: 500
Valid Values: integer
Changes Take Effect: Immediately
Introduced: 8.5.109.05

Specifies the maximum number of callbacks per service for the duration set in _throttle_ttl_1. A second limitation can be introduced with the options _throttle_callbacks_per_service_2 and _throttle_ttl_2. Genesys recommends using these options to prevent a high rate of callback requests in a given period of time. For instance, to set a limit of 500 callbacks in 5 minutes or 1000 callbacks in 1 hour, use the following configuration:

```
_throttle_callbacks_per_service_1=500
_throttle_ttl_1=300
_throttle_callbacks_per_service_2=1000
_throttle_ttl_2=3600
```

_throttle_ttl_1

Section: Queue Management
Default Value: 300
Valid Values: integer
Changes Take Effect: Immediately

Specifies the duration in seconds used to limit the callbacks as specified in
_throttle_callbacks_per_service_1. A second limitation can be introduced with the options
_throttle_callbacks_per_service_2 and _throttle_ttl_2. Genesys recommends using these
options to prevent a high rate of callback requests in a given period of time. For instance, to set a
limit of 500 callbacks in 5 minutes (300 seconds) or 1000 callbacks in 1 hour (3600 seconds), use
the following configuration:

```
_throttle_callbacks_per_service_1=500
_throttle_ttl_1=300
_throttle_callbacks_per_service_2=1000
_throttle_ttl_2=3600
```

If these pairs of conditions xx_1 and xx_2 are satisfied, the callback request should be throttled.

_throttle_callbacks_per_service_2

Section: Queue Management
Default Value: 1000
Valid Values: integer
Changes Take Effect: Immediately

Specifies the maximum number of callbacks per service for the duration set in _throttle_ttl_2.
A second limitation can be introduced with the options _throttle_callbacks_per_service_1
and _throttle_ttl_1. Genesys recommends using these options to prevent a high rate of
callback requests in a given period of time. For instance, to set a limit of 500 callbacks in 5
minutes or 1000 callbacks in 1 hour, use the following configuration:

```
_throttle_callbacks_per_service_1=500
_throttle_ttl_1=300
_throttle_callbacks_per_service_2=1000
_throttle_ttl_2=3600
```

_throttle_ttl_2

Section: Queue Management
Default Value: 3600
Valid Values: integer

Changes Take Effect: Immediately

Specifies the duration in seconds used to limit the callbacks as specified in
_throttle_callbacks_per_service_2. A second limitation can be introduced with the options
_throttle_callbacks_per_service_1 and _throttle_ttl_1. Genesys recommends using these
options to prevent a high rate of callback requests in a given period of time. For instance, to set a
limit of 500 callbacks in 5 minutes (300 seconds) or 1000 callbacks in 1 hour (3600 seconds), use
the following configuration:

```
_throttle_callbacks_per_service_1=500
_throttle_ttl_1=300
_throttle_callbacks_per_service_2=1000
_throttle_ttl_2=3600
```

If these pairs of conditions xx_1 and xx_2 are satisfied, the callback request should be throttled.

_throttle_customer_number_limit

Section: Queue Management
Default Value: 6
Valid Values: Any integer
Changes Take Effect: Immediately
Dependencies: _throttle_request_parameters
Introduced: 8.5.109.05
Discontinued: 8.5.110.07

Specifies the maximum number of **immediate** callback requests that a given customer number
can query per day. You can set the list of parameters to apply this throttling in the option
_throttle_request_parameters.

Starting in 8.5.110.07, this option is discontinued. Use option
_throttle_request_parameters_limit instead.

_throttle_request_parameters

Section: Queue Management
Default Value: _customer_number
Valid Values: String
Changes Take Effect: Immediately
Introduced: 8.5.109.05

Specifies a comma-separated list of request parameters to which the
_throttle_customer_number_limit option applies. For example,
_throttle_request_parameters=_customer_number, _target

_enable_in_queue_checking

Section: Queue Management
Default Value: true
Valid Values: true, false, strict
Changes Take Effect: Immediately
Introduced: 8.5.109.05
Modified: 8.5.111.04

- If true, prevents the callback creation if the same customer number has already two callbacks in the queue.

- If strict, prevents the callback creation if the same customer number has already one callback in the queue.

- If false, does not check whether the customer is already in queue when creating the callback.

This option applies to both immediate and scheduled callbacks.

_max_queued_callbacks_per_service

Section: Queue Management
Default Value: 1000
Valid Values: integer
Changes Take Effect: Immediately
Introduced: 8.5.108.02

Maximum number of queued callbacks per service. By default, this limit is set to 1000.

_queue_poll_period

Section: Queue Management
Default Value: -120,120
Valid Values: string
Changes Take Effect: Immediately
Introduced: 8.5.107.19

Defines the time range in minutes to pick callbacks in queue from the current time. The default range means that GMS will poll from (now() - 120min) to (now() + 120min). For performance reasons, polling is done on a small period of data to avoid system overloading. As a rule of thumb, the fetched period should not contain more than 1000 callbacks to process/ping (overall, taking all virtual queues in account) in the 60-seconds default interval configurable using the queue-polling-rate option.

_queue_poll_period_recovery

Section: Queue Management
Default Value: -1440,-120
Valid Values: Range
Changes Take Effect: Immediately
Introduced: 8.5.107.19

Defines the time range in minutes to pick callbacks in queue from the current time for the recovery period. Callbacks in SCHEDULED status are picked for submission; other callbacks in active states are ping on ORS sessions and re-submitted if failed. The default range means that GMS will poll from (now() - 1440min) to (now() - 120min).

_queue_ping_ors_period

Section: Queue Management
Default Value: -120,-2
Valid Values: string
Changes Take Effect: Immediately

Defines the time range in minutes to ping ORS for resubmission. The default range means that GMS will poll from (now() - 120min) to (now() - 2min). Must be within the _queue_poll_period range.

_disallow_impossible_phone_numbers

Section: Queue Management
Default Value: true
Valid Values: boolean
Changes Take Effect: Immediately
Introduced: 8.5.107.19
Modified: 8.5.108.02

Disables callbacks for unreachable phone numbers that contain unexpected characters, cannot be parsed, or are too long for the specified country.

> ### Important
> If _disallow_impossible_phone_numbers is true, you must set the value of the _default_country option.

_disallow_premium_phone_numbers

Section: Queue Management
Default Value: true
Valid Values: boolean
Changes Take Effect: Immediately
Modified: 8.5.108.02

Disables callbacks for premium numbers, such as 900 numbers in the USA. Premium US numbers are listed in Wikipedia and are often called a 900 number or a 1-900 number ("one-nine-hundred"). The customer phone number is checked according to the country configured in the _default_country option.

> ## Important
>
> If _disallow_premium_phone_numbers is true, you must set the value of the _default_country option.

_default_country

Section: Queue Management
Default Value: US
Valid Values: string
Changes Take Effect: Immediately
Introduced: 8.5.107.19
Modified: 8.5.108.02

Sets the default country code as defined in ISO 3166. You must set this option if _disallow_impossible_phone_numbers or _disallow_premium_phone_numbers is enabled.

Reporting section

No results No results

_rep_userevent_enable

Section: Reporting
Default Value: false
Valid Values: boolean
Changes Take Effect: Immediately

If set to true, the callback data is reported via user events to the switch and the specified DN (_rep_userevent_switch and _rep_userevent_dn).

This option is mandatory.

_rep_userevent_switch

Section: Reporting
Default Value:
Valid Values: string
Changes Take Effect: Immediately

Switch to which the DN _rep_userevent_dn belongs. Required if _rep_userevent_enable is set to true.

_rep_userevent_dn

Section: Reporting
Default Value:
Valid Values: string
Changes Take Effect: Immediately

DN to which the reporting user event is sent. If _rep_userevent_enable is set to true, a value for this parameter is required.

_rep_userevent_mediatype

Section: Reporting
Default Value: 0
Valid Values: integer
Changes Take Effect: Immediately

Media type for the reporting user event generated from callback application. Accepts an integer value (ENUM) as per TLib specification for Media Types. Required when _rep_userevent_enable is set to true.

> **Tip**
>
> This is an advanced parameter. To modify the value of an advanced parameter, you must enable **Advanced Parameters** in the Service Management UI.

Scheduled Call section

_desired_connect_time_threshold

Section: Scheduled Call
Default Value: 180
Valid Values: integer
Changes Take Effect: Immediately

Number of seconds of delay before the desired connect time for service execution.

more...

_desired_time

Section: Scheduled Call
Default Value:
Valid Values: UTCDate
Changes Take Effect: Immediately

Callback desired time. Format is ISO 8601 (in UTC) 'yyyy-MM-ddTHH:mm:ss.SSSZ'. For example: '2013-05-28T15:30:00.000Z'

_request_time_bucket

Section: Scheduled Call
Default Value: 5
Valid Values: 5,10,15,20,30,60,120
Changes Take Effect: Immediately
Modified: 8.5.202.03

Period of time during which GMS attempts to schedule a specific amount of requests. By default, the time buckets are 5 minutes and the max_request_by_time_bucket option is set to 100; as a result, a maximum of 100 scheduled requests can be done in the given 5 minutes bucket.

Starting in 8.5.202.03, the following values are valid for this option: 5,10,15,20,30,60,120.

> ### Important
> Applicable if the _max_request_by_time_bucket option is set.

_request_execution_time_buffer

Section: Scheduled Call
Default Value: 120
Valid Values: Integer (Seconds)
Changes Take Effect: Immediately

Time within which a request should be submitted to the Callback Orchestration execution service.

_request_queue_time_stat

Section: Scheduled Call
Default Value:
Valid Values:
Changes Take Effect: Immediately

Statistic used to define when a request should be submitted to the Callback Orchestration execution service. The request_execution_time_buffer value and request_queue_time_stat statistic options define when a queued request should be submitted to the execution service. For example, you can use the ExpectedWaitTime statistic to set this option:
"ExpectedWaitTime;Queue;8999@SIP_Server;Environment"

_reject_future_desired_time

Section: Scheduled Call
Default Value: 1M

Valid Values: 1<letter> where letter can be: 'y' for year, 'M' for month, 'w' for week, 'd' for day, 'h' for hour, and 'm' for minute.
Changes Take Effect: Immediately

Time limit in future to schedule callbacks. If a callback is scheduled later than this limit, the callback request is rejected. You can specify a general option for all your callback services in your GMS application by setting the `callback._reject_future_desired_time` option. The default is one month (1M).

URS Queueing

_max_queue_wait

Section: URS Queueing
Default Value: 0
Valid Values: integer
Changes Take Effect: Immediately
Introduced: 8.5.112.05

Maximum time (in seconds) for the virtual interaction to wait in queue prior to contacting the customer if the virtual interaction has not been selected for routing. If set to 0 (default), this option is ignored. If you set this option, configure `_urs_ewt_estimation_method = ursdial`.

This option only applies to user-terminated delayed scenarios where `_userterminated_first_connect_party=CUSTOMER`. If max-queue-wait is configured, and if its value is exceeded, the system adds the `GMS_Max_Queue_Wait` user data to the outbound interaction.

_urs_vq_priority_boost_on_connect

Section: URS Queueing
Default Value: 500
Valid Values: integer
Changes Take Effect: Immediately
Introduced: 8.5.112.05

Priority to be added to the virtual interaction after connecting the customer call or chat session. This option only applies to scenarios where `_userterminated_first_connect_party=CUSTOMER`.

To use this option, import the GMS_URS_Strategy_85109_v2.58.zip strategy file available for download in the Callback Solution guide.

_eta_pos_threshold

Section: URS Queueing
Default Value: 0:10,10:5,20:2,30:1,40:0
Valid Values: string
Changes Take Effect: Immediately

> ### Important
>
> This legacy option applies only if `_urs_ewt_estimation_method` is **not** set to `ursdial`.

Threshold at which the service determines that an agent is available for an interaction with the customer. If an interaction is in first position then independent of this setting, the system assumes that the threshold is reached.

For example, the default value "0:10,10:5,20:2,30:1,40:0" sets different threshold positions depending on the interaction's Expected Time to Answer (ETA) as calculated by URS.

- If the service instance's ETA is between 0 and 10 and if the interaction position in queue is less than 10, the interaction is ready to be connected.

- If the service instance's ETA is between 10 and 20 and if the interaction position in queue is less than 5, the interaction is ready to be connected, and so on.

> ### Tip
>
> This is an advanced parameter. To modify the value of an advanced parameter, you must enable **Advanced Parameters** in the Service Management UI.

_urs_request_timeout

Section: URS Queueing
Default Value: 100
Valid Values: string
Changes Take Effect: Immediately

Duration (seconds) to wait for receiving HTTP responses from Universal Routing Server.

_urs_virtual_queue

Section: URS Queueing
Default Value: {Specify virtual queue to be used by strategy}
Valid Values: string
Changes Take Effect: Immediately

Virtual queue (alias) to which the service request will be added.

_urs_queued_ttl

Section: URS Queueing
Default Value: 14400
Valid Values: integer
Changes Take Effect: Immediately

The total maximum time (seconds) to wait for a target. After the specified duration has lapsed the virtual interaction will be removed from virtual queue and the callback service will exit.

This option is mandatory.

_max_time_to_reach_eta_pos_threshold

Section: URS Queueing
Default Value: 14160
Valid Values: integer
Changes Take Effect: Immediately

Maximum time (seconds) to wait until the service request reaches the queue position controlled by the ETA Position Threshold parameter (`_eta_pos_threshold`).

This option is mandatory.

_target

Section: URS Queueing
Default Value: {specify your target here}
Valid Values: String

Changes Take Effect: Immediately
Modified: 8.5.108.02, 8.5.109.05, 8.5.109.08, 8.5.114.09

Routing target that specifies the agent/queue resource that will process this request.

- Starting in 8.5.108.02, you can set multiple targets in this option, limited to 5.

- Starting in 8.5.114.09, the limit is increased to 15.

Single Target

For a **single** target, format the string according to the URS target specification: `<Target String>@<StatServer name>.<Target Type>` where `Target Type` is one of the following:

- A (Agent)

- AP (Agent Place)

- GA (Group of Agents)

- GP (Group of Places)

- GC (Campaign Group)

`<Target String>` can be a skill expression. In that case, `<Target String>` must start with '?:'. For example:

- `Billing@StatServer.GA`—Routes to Agent Group "Billing".

- `?:English=20&;Loans=2@StatServer.GA`—Routes to any agent matching the skill expression.

See the Universal Routing Server (URS) documentation for additional information about URS targets.

Multiple Targets

To set multiple targets, create a JSON-formatted string array of maximum 15 elements as follows:

```
[
    {
        "target": "<Target String>@<StatServer name>.<Target Type>",
        "timeout": "<integer>",
        "clear":<true/false>,
        "stat_to_check": "<stat name>",
        "stat_operator": "< or >",
        "stat_value": "1"
    }
]
```

- The `timeout` property specifies how long to wait in seconds before switching of targets.

- The `stat_to_check` property can be set to any of the values supported by the Statistics parameter passed to the IRD function `SData(Target, Statistics)`, unless `target` is a skill expression. If `target` is a skill expression, you must choose one of the following values:

- RStatAgentsReadyvoice—agents ready for voice media.

- RStatAgentsReady—agents ready for any media.

- RStatAgentsTotal—agents logged in.

- The stat_value property specifies the threshold for the statistic passed in stat_to_check. If the condition set by the combination of stat_to_check, stat_operator, and stat_value is met, the current target is skipped, except if it is the last target of the list.

- If clear=true, the target will be overridden when switching to the next target; if clear=false, the target will be expanded with the next target.

> ## Important
> If you set multiple targets in this option, then _urs_queued_ttl should be set to the total queue time across all targets.

more...

_urs_extension_data

Section: URS Queueing
Default Value: No default value
Valid Values: <data>=<value>
Changes Take Effect: Immediately
Introduced: 8.5.203.02
Discontinued: From 8.5.114.09 to 8.5.202.03

Provides the additional configuration extension data required for advanced scenarios and used for the customization of the URS prioritization strategy. Use this option to pass additional extension data to the URS strategy, for example, if you customize the WaitForTarget IRD strategy.

Multiple values can be combined to form a comma-separated list that the URS strategy will parse. For example, _urs_extension_data=targets=VAG1,VAG2,VAG3;

> ## Tip
> This is an advanced parameter. To modify the value of an advanced parameter, you must enable **Advanced Parameters** in the Service Management UI.

_min_urs_ewt_pos_polling_interval

Section: URS Queueing
Default Value: 2
Valid Values: integer
Changes Take Effect: Immediately

Lowest possible check frequency to avoid constant URS polling. The Callback service checks the EWT Position for the virtual interaction waiting in the virtual queue. Generally, the check frequency is half of the current EWT.

> ## Tip
> This is an advanced parameter. To modify the value of an advanced parameter, you must enable **Advanced Parameters** in the Service Management UI.

This option is mandatory.

_max_urs_ewt_pos_polling_interval

Section: URS Queueing
Default Value: 30
Valid Values: integer
Changes Take Effect: Immediately

Highest possible check frequency to avoid polling delays on fluctuating loads. The Callback service checks the EWT Position for the virtual interaction waiting in the virtual queue. Generally, the check frequency is half of the current EWT.

> ## Tip
> This is an advanced parameter. To modify the value of an advanced parameter, you must enable **Advanced Parameters** in the Service Management UI.

This option is mandatory.

_min_queue_wait

Section: URS Queueing
Default Value: 0
Valid Values: integer
Changes Take Effect: Immediately

Minimum time (in seconds) for the virtual interaction to wait in queue prior to getting selected for routing.

_agent_reserve_timeout

Section: URS Queueing
Default Value: 30
Valid Values: integer
Changes Take Effect: Immediately

Agent reservation time during which the agent is blocked for routing. After this timeout, URS automatically unblocks the agent.

Reservation is done using RDND blocking in URS. If you use multiple URS instances, you must add the following URS configuration for self-awareness:

- using=lds
- lds=map

This option is mandatory.

_urs_call_interaction_age

Section: URS Queueing
Default Value:
Valid Values: (\"t\"=\"seconds\"<space>\"milliseconds\")
Changes Take Effect: Immediately

If specified, the URS strategy will set the interaction age to the specified value; otherwise, by default, the interaction will be placed at the end of the queue. This is generally useful in the classic callback scenarios where the original interaction is already in the queue prior to the GMS callback being offered. The format is i.e: (\"t\"=\"seconds\"<space>\"milliseconds\"), for example, _urs_call_interaction_age="(\"t\"=\"1449157660 829\")".

_urs_ewt_estimation_method

Section: URS Queueing
Default Value: ursdial
Valid Values: string
Changes Take Effect: Immediately

For all new deployments, `ursdial` should be used. Other options (`stat` and `urs`) are deprecated.

- Selecting `ursdial` (highly recommended) enables URS-based dialing where the estimation of time to dial is determined by URS (version 8.1.400.07 or greater).

- If `urs` or `stat` is selected, the callback time to dial is determined by checking ewt and pos retrieved from URS or Stat Server respectively.

> **Tip**
>
> This is an advanced parameter. To modify the value of an advanced parameter, you must enable **Advanced Parameters** in the Service Management UI.

This option is mandatory.

_urs_vq_priority

Section: URS Queueing
Default Value:
Valid Values: integer
Changes Take Effect: Immediately

Priority to be set for the virtual interaction when submitting to `_urs_virtual_queue`. For example, you can set this value to 100. If you leave this option blank, no priority will be set. For further details, see the callback documentation.

> **Tip**
>
> This is an advanced parameter. To modify the value of an advanced parameter, you must enable **Advanced Parameters** in the Service Management UI.

_urs_vq_priority_increment

Section: URS Queueing
Default Value:
Valid Values: integer
Changes Take Effect: Immediately

Increment interval to use to increment the priority, every
`_urs_vq_priority_increment_interval`. for example, 10. For further details, see the callback
documentation.

> **Tip**
>
> This is an advanced parameter. To modify the value of an advanced parameter, you
> must enable **Advanced Parameters** in the Service Management UI.

_urs_vq_priority_increment_interval

Section: URS Queueing
Default Value:
Valid Values: integer
Changes Take Effect: Immediately

Number of seconds to wait to increment the priority of `_urs_vq_priority_increment`. For
example, 60 to increment the priority every 60 seconds. If you leave this option blank, the
incrementation of the priority is disabled.

> **Tip**
>
> This is an advanced parameter. To modify the value of an advanced parameter, you
> must enable **Advanced Parameters** in the Service Management UI.

_on_route_done_delay

Section: URS Queueing
Default Value: 0

Valid Values: Integer
Changes Take Effect: Immediately
Introduced: 8.5.106.14

The number of seconds, after the target selection, by which to allow delays when fetching user data from a URS `WaitForTarget` strategy after the target was selected. This option ensures that user data is consistently added to the interaction. It does not delay the callback execution.

_urs_prioritization_strategy

Section: URS Queueing
Default Value: WaitForTarget
Valid Values: String
Changes Take Effect: Immediately

Universal Routing Server strategy to be used for the service instance prioritization. The default value shown here matches the name of the URS strategy that you imported into IRD. If you changed the name of the strategy, update this value to reflect the correct name.

_urs_strategy_update_sub_routine

Section: URS Queueing
Default Value: SetRouteDelay
Valid Values: String
Changes Take Effect: Immediately

Subroutine in URS that changes the `RouteDelay` parameter of the virtual interaction representing the service instance. This option informs URS to reserve this interaction's target when it becomes available. The default value shown here matches the name of the URS subroutine that you imported into IRD. If you changed the name of the subroutine, update this value to reflect the correct name.

_urs_udata_xfer_keys

Section: URS Queueing
Default Value:
Valid Values: String
Changes Take Effect: Immediately

Comma-separated list of Callback KVPs to be retrieved from the routing strategy and added to preview interactions.

See also

_urs_url

Section: no category
Default Value: No default value
Valid Values: String URL
Changes Take Effect: Immediately

URS URL or comma-separated list of URS URLs if the enable_urs_loadbalancer option is `true` in the GMS application. For example:

`http://host#1:7311,http://host#2:7311,http://host#3:7311`

GMS uses this list to query URS statistics.

.

Voice - User Originated

_resource_group

Section: Voice - User Originated
Default Value: DNIS
Valid Values: resource_group
Changes Take Effect: Immediately

Resource group from which access number is to be allocated.

This option is mandatory.

_provide_code

Section: Voice - User Originated
Default Value: false
Valid Values: boolean
Changes Take Effect: Immediately

If true, returns a randomly generated code to be used for the authentication of the user originated (inbound) call.

This option is mandatory.

_booking_expiration_timeout

Section: Voice - User Originated
Default Value: 30
Valid Values: integer
Changes Take Effect: Immediately

Used to book the access number resource for a period of time. The customer needs to make the call within the specified timeout to ensure a successful match.

This option is mandatory.

_userorig_connect_limit

Section: Voice - User Originated
Default Value: 3
Valid Values: integer
Changes Take Effect: Immediately

Maximum number of times that the customer can request a connection or a reconnection.

_src_route_point

Section: Voice - User Originated
Default Value:

Valid Values: string
Changes Take Effect: Immediately

If set, this route point will be used as the source DN (focus device) when routing the user originated (inbound) call to the agent. In most cases, the source DN is automatically detected correctly and doesn't need to be set. Hence, it is recommended to leave it empty.

> **Tip**
>
> This is an advanced parameter. To modify the value of an advanced parameter, you must enable **Advanced Parameters** in the Service Management UI.

_offer_callback_vxml_app_url

Section: Voice - User Originated
Default Value:
Valid Values: string
Changes Take Effect: Immediately

URL of the custom VXML application to be used to offer a callback (treatment). If _offer_callback is set to true, a VXML app must be used to offer the callback. By default, this parameter is empty and the callback application uses the built-in VXML app.

> **Tip**
>
> This is an advanced parameter. To modify the value of an advanced parameter, you must enable **Advanced Parameters** in the Service Management UI.

_offer_callback

Section: Voice - User Originated
Default Value: false
Valid Values: boolean
Changes Take Effect: Immediately

When this option is set to true, the callback option is presented to the caller. The IVR application specified in the _offer_callback_vxml_app_url option is used to present the callback option. If the _offer_callback_vxml_app_url is blank (default), the built-in VXML application is executed.

This option is mandatory.

Voice - User Terminated

_ixn_createcall_hints

Section: Voice - User Terminated
Default Value:
Valid Values: JSON-formatted string
Changes Take Effect: Immediately
Modified: 8.5.108.02

Set the hints parameter of the scxml ixn:createcall request for the outbound callback.

- In versions prior to 8.5.108.02, you can use this parameter to override the default CPD behavior and, in this scenario, the _cpd_enable flag is ignored.

- Starting in 8.5.108.02, you can use this parameter to merge the default CPD behavior and, in this scenario, the _cpd_enable flag is ignored. If you do not set this option, Callback adds the default value to the _ixn_createcall_hints parameter.

> **Tip**
>
> This is an advanced parameter. To modify the value of an advanced parameter, you must enable **Advanced Parameters** in the Service Management UI.

more...

_agent_first_via_rp

Section: Voice - User Terminated
Default Value: false
Valid Values: boolean
Changes Take Effect: Immediately
Introduced: 8.5.108.02

Enables dialing of the call from the route point (set in the _route_point option) in a user-terminated scenario *connect to agent first* where the agent preview mode is disabled. Otherwise, the call will be dialed directly from the agent's DN.

This option is mandatory.

_dial_retry_timeout

Section: Voice - User Terminated
Default Value: 300
Valid Values: integer
Changes Take Effect: Immediately
Introduced: 8.5.106.16

Time to wait (seconds) between retries to reach the customer.

> Tip
>
> This is an advanced parameter. To modify the value of an advanced parameter, you must enable **Advanced Parameters** in the Service Management UI.

This option is mandatory.

_call_timeguard_timeout

Section: Voice - User Terminated
Default Value: 15000
Valid Values: integer
Changes Take Effect: Immediately

Time duration (ms) for CPD (Call Progress Detection) to wait for a response. Setting this value to a low value may result in incorrect call state being returned.

> Tip

This is an advanced parameter. To modify the value of an advanced parameter, you must enable **Advanced Parameters** in the Service Management UI.

_calling_party_number

Section: Voice - User Terminated
Default Value:
Valid Values: string
Changes Take Effect: Immediately

Number to be displayed (Caller ID) on the customer's device when the call arrives at the device

_calling_party_display_name

Section: Voice - User Terminated
Default Value:
Valid Values: string
Changes Take Effect: Immediately

Name to be displayed on the customer's device when the call arrives. This value will take effect only if you set up _calling_party_number.

_agent_preview_data

Section: Voice - User Terminated
Default Value: Value 1,Value 2,Value 3,Value 4,Value 5
Valid Values: string
Changes Take Effect: Immediately

Comma-separated ordered list of values to be displayed in the agent preview dialog.

_agent_preview_via_rp

Section: Voice - User Terminated
Default Value: false
Valid Values: boolean
Changes Take Effect: Immediately

If set to `true`, in an agent preview scenario, the call will be dialed from the route point specified by the `_route_point` option. Otherwise, the agent DN will make the call.

This option is mandatory.

_agent_disposition_timeout

Section: Voice - User Terminated
Default Value: 45
Valid Values: integer
Changes Take Effect: Immediately

Time duration allowed for the agent to take action on the Agent Disposition dialog.

_enable_disposition_dialog

Section: Voice - User Terminated
Default Value: false
Valid Values: boolean
Changes Take Effect: Immediately

If set to true, after a callback is completed (interaction deleted) agent is shown a dialog to specify the disposition of the call and schedule a retry as necessary.

This option is mandatory.

_agent_reject_retry_timeout

Section: Voice - User Terminated
Default Value: 0

Valid Values: integer
Changes Take Effect: Immediately

Time (in seconds) to wait after the agent rejects the service request prior to putting it back in the queue. While waiting, the callback status is set to QUEUED.

_vq_for_outbound_calls

Section: Voice - User Terminated
Default Value:
Valid Values: string
Changes Take Effect: Immediately

Virtual Queue (alias) to which the real call is momentarily queued. Setting this option allows reporting systems, which depend on Virtual Queue events, to gather all the required information related to outbound calls.

_plugin_on_dial_timeout

Section: Voice - User Terminated
Default Value: 120
Valid Values: integer
Changes Take Effect: Immediately

Maximum time limit (in seconds) for the plugin to send an asynchronous response to the GMS Callback Service.

_preview_userevent_mediatype

Section: Voice - User Terminated
Default Value: 0
Valid Values: integer
Changes Take Effect: Immediately

Media type for the agent preview user event generated by the callback application. This option supports an integer value (ENUM) as per TLib specification for Media Types. This option is required when _agent_preview_enable is set to true.

> **Tip**
>
> This is an advanced parameter. To modify the value of an advanced parameter, you must enable **Advanced Parameters** in the Service Management UI.

_disposition_userevent_mediatype

Section: Voice - User Terminated
Default Value: 0
Valid Values: integer
Changes Take Effect: Immediately

Media type for the agent disposition user event generated by the callback application. This option supports an integer value (ENUM) as per TLib specification for Media Types. This option is required when _enable_disposition_dialog is set to true.

> **Tip**
>
> This is an advanced parameter. To modify the value of an advanced parameter, you must enable **Advanced Parameters** in the Service Management UI.

_ixn_createcall_timeout

Section: Voice - User Terminated
Default Value: 32
Valid Values: Integer
Changes Take Effect: Immediately

Timeout (in seconds) to wait for the interaction creation.

> **Tip**
>
> This is an advanced parameter. To modify the value of an advanced parameter, you

must enable **Advanced Parameters** in the Service Management UI.

This option is mandatory.

_on_user_confirm_timeout

Section: Voice - User Terminated
Default Value: CONNECT-ANYWAY
Valid Values: CONNECT-ANYWAY, CANCEL
Changes Take Effect: Immediately

Selects the action to perform if the user does not submit his or her confirmation in response to the push notification.

- `CONNECT-ANYWAY` will continue with the call.
- `CANCEL` cancels the service request.

_prefix_dial_out

Section: Voice - User Terminated
Default Value: 9
Valid Values: Integer
Changes Take Effect: Immediately

Prefix required to perform a user-terminated (outbound) call from the system.

_route_point

Section: Voice - User Terminated
Default Value:
Valid Values: String
Changes Take Effect: Immediately

Route point from which the system can create a user-terminated (outbound) call.

This option is mandatory.

_userterminated_first_connect_party

Section: Voice - User Terminated
Default Value: CUSTOMER
Valid Values: CUSTOMER, AGENT
Changes Take Effect: Immediately

First party to connect when _call_direction is set to USERTERMINATED. Set this option to CUSTOMER to call the customer first; set this option to AGENT to call the agent first.

This option is mandatory.

_agent_preview

Section: Voice - User Terminated
Default Value: false
Valid Values: Boolean

Changes Take Effect: Immediately

Enables Agent Preview. If set to true, the Preview Dialog with caller information is displayed to the agent.

_agent_preview_timeout

Section: Voice - User Terminated
Default Value: 30
Valid Values: Integer (seconds)
Changes Take Effect: Immediately

Duration in seconds that the agent has to preview the callback information and submit a reply. The Preview dialog will automatically close after this timeout and submit a reject from the agent. In this scenario, the call will go back in the queue. During this period, the agent is reserved for the Callback interaction and is not an eligible target for other interactions; therefore, Genesys recommends to evaluate carefully when extending this timeout beyond 30 seconds (default).

_cpd_enable

Section: Voice - User Terminated
Default Value: true
Valid Values: Boolean
Changes Take Effect: Immediately

Enables CPD. If this option is set to true, CPD will be performed on a callback made to the customer.

- If CPD results in a human or silence detection, the call will be routed to the agent.

- If a fax is detected, the call will be disconnected and marked complete.

- If an answering machine is detected, the answering machine treatment is played.

This option is mandatory.

_cb_dim_channel

Section: Voice - User Terminated
Default Value: WEB
Valid Values:
Changes Take Effect: Immediately

Callback origination channel that will be reported as part of the callback reporting. This option is generally applicable when one service is used for several channels.

_plugin_on_dial_url

Section: Voice - User Terminated
Default Value:
Valid Values: String
Changes Take Effect: Immediately

URL of the SCXML custom plugin to execute various custom scenarios after a user terminated callback is dialed; these scenarios depend on the state of the call. By default, there is no value for this option and the plugin execution is disabled.

_plugin_on_dial_associate_ixn

Section: Voice - User Terminated
Default Value: true
Valid Values: Boolean
Changes Take Effect: Immediately

True to allow the ORS session executing the custom plugin to control the call (interaction). This enables the plugin to perform some advanced actions which require the session to be in control of the interaction.

> ### Important
>
> Applicable only if the _plugin_on_dial_url option is set.

_plugin_on_dial_invoke_on_call_failed

Section: Voice - User Terminated
Default Value: true
Valid Values: Boolean
Changes Take Effect: Immediately

True to allow the custom plugin to be invoked even when the dialed call to the customer has failed by not being answered by a human.

> ### Important
>
> Applicable only if the _plugin_on_dial_url option is set.

_max_dial_attempts

Section: Voice - User Terminated
Default Value: 3
Valid Values: Integer
Changes Take Effect: Immediately

Maximum number of attempts to call the customer.

Voice Treatment

No results No results No results No results

See also

_on_customer_connect_treatment_url

Section: no category
Default Value: " "
Valid Values: String URL
Changes Take Effect: Immediately

URL of a VXML application to execute after a user terminated callback is dialed and the call is answered by a human.

By default, this option is empty and the VXML execution is disabled.

.

No category

The following options are used for advanced customization in the `service.<service_name>` section of your GMS configuration or in your REST queries. ---

_ors

Section: no category
Default Value:
Valid Values: String
Changes Take Effect: Immediately

Comma-separated list of ORS URLs, for example:

`http://host1:port1,http://host2:port2`

Overrides any ORS connections, if they are present.

> **Important**
> Set this option in the `service.<service_name>` section of your GMS configuration, not in the Service Management UI.

_ors_lb_strategy

Section: no category
Default Value: circular
Valid Values: circular, linear
Changes Take Effect: Immediately

Strategy for the ORS load balancer. This option overrides the `_ors_lb_strategy` option of the GMS application. Supported values are: `circular` or `linear`.

Set this option in the `service.<service_name>` section of your GMS configuration, not in the Service Management UI.

_redirect

Section: no category
Default Value:
Valid Values: String
Changes Take Effect: Immediately
Introduced: 8.5.106.19

Defines a redirection of your POST queries to another callback service. For example, if you set this parameter for the `foo` service as follows....

```
[service.foo]
_redirect=foo_b
```

.... when you will invoke foo with a POST query....

```
POST /1/service/callback/foo
```

.... GMS internally redirects your query to foo_b:

```
POST /1/service/foo_b
```

- This option is restricted to callback services.

- Set this option in the GMS configuration, not in the Service Management UI. You can disable and overwrite this option in your REST queries if you set _overwritable_options=_redirect in your service configuration.

_overwritable_options

Section: no category
Default Value:
Valid Values: String
Changes Take Effect: Immediately
Introduced: 8.5.106.19

Defines the parameters (_ors, _redirect, and/or _target) that you wish to be able to overwrite in your REST queries.

This option enables you to pass non-request parameters in your REST queries. For example, if you set _overwritable_options=_ors, a customer application can schedule a callback and pass the _ors parameter of the REST query to submit this callback to a given ORS.

```
POST /1/service/callback/foo

{
 "_ors": "http://myors:4421"
}
```

Another scenario is to set _overwritable_options=_redirect in order to disable the redirection for some queries. For example, if you set the _redirect parameter for the foo service as follows....

```
[service.foo]
_redirect=foo_b
_overwritable_options=_redirect
```

.... when you invoke foo with a POST query, you can overwrite the value of the _redirect option and disable the redirection to foo_b:

```
POST /1/service/callback/foo?_redirect=""
```

_status_notification_debug

Section: no category
Default Value: false
Valid Values: true, false
Changes Take Effect: Immediately

If true, enables the debug mode for notification.

_status_notification_language

Section: no category
Default Value: false
Valid Values: String
Changes Take Effect: Immediately

Application language that matches one of the supported languages used for push notifications.

_on_customer_connect_treatment_url

Section: no category
Default Value: " "
Valid Values: String URL
Changes Take Effect: Immediately

URL of a VXML application to execute after a user terminated callback is dialed and the call is answered by a human.

By default, this option is empty and the VXML execution is disabled.

_urs_url

Section: no category
Default Value: No default value
Valid Values: String URL
Changes Take Effect: Immediately

URS URL or comma-separated list of URS URLs if the enable_urs_loadbalancer option is true in the GMS application. For example:

```
http://host#1:7311,http://host#2:7311,http://host#3:7311
```

GMS uses this list to query URS statistics.

Office Hours Options

_include_schedule<n>

Section: office-hours
Default Value:
Valid Values: service.<office hours ID>
Changes Take Effect: Immediately

The name of another office-hours service to include in the current service. If you need to combine the schedules of several teams, use this option to import schedules from other Office Hours Services to combine working hours of multiple teams and shifts in the same Office Hours service. In this scenario, you can create a main Office Hours service (All Offices in the example below) into which you can import all the Office Hours schedules (Regular Hours, Holidays, Exceptions, and timezones). This results in the union of the imported Office Hours schedules within the timezone of the main Office Hours.

For example, create and configure two office Hours services, one named Office SFO, the other Office Amsterdam. Then create the All Offices service and set _include_schedule parameter to combine schedules of both teams. A service referenced in _include_schedule parameter can include additional _include_rule and _include_schedule option(s) and in its configuration.

_include_rule<n>

Section: office-hours
Default Value:
Valid Values: service.<national_holidays>
Changes Take Effect: Immediately

ID of another office-hours service. All rules, such as _bh_regular<n>, _bh_add<n>, _holiday<n> from the specified service (and it's predecessors) are included into the current service. The timezone specified in the included service is ignored. Instead the timezone specified in the current service, applies to all of the rules configured in the current service.

_holiday<n>

Section: office-hours
Default Value:
Valid Values: "YYYY-MM-dd" or "MM-dd"
Changes Take Effect: Immediately

Holidays are specified either by full date (including the year), or only by month and day. In the latter case it is treated as a recurrent event, which is repeated annually. The name of the parameter should be unique within this service (section of options). For example, 2016-07-01 or 07-01.

The parameter name should be unique within this service. For instance, _holiday1, _holiday2, and so on.

_bh_regular<n>

Section: office-hours
Default Value:
Valid Values: DDD-DDD HH:mm-HH:mm
Changes Take Effect: Immediately

Days and hours of the week where regular office hours apply; for example, Mon-Fri 9:00-17:00. The range of days can overrun the end of the week, for example, you can specify Sat-Mon that includes 3 days: Saturday, Sunday, Monday. The hours are specified in 24-hour format. The range of hours can overrun the end of the day, for example 19:00-7:00 to model the shift that starts at 7 pm and ends at 7 am.

The parameter name should be unique within this service. For instance, _bh_regular2, _bh_regular2, and so on.

_bh_add<n>

Section: office-hours
Default Value: No default value
Valid Values: YYYY-MM-dd HH:mm-HH:mm or MM-dd HH:mm-HH:mm
Changes Take Effect: Immediately

Specifies a special date and time period when the office will be open.

Additional days can be specified by providing either full date (including the year) or only the month and the day. For example, 2014-11-30 10:00-14:00 or 11-30 10:00-14:00

If you do not specify the year, the event is considered as recurrent and will be repeated every year. The name of the parameter should be unique within this service (section of options). For instance, _bh_add1, _bh_add2, and so on.

_use_alternative<n>

Section: office-hours
Default Value:
Valid Values: String
Changes Take Effect: Immediately

A comma-separated list of one or more alternative `office-hours` services that should be used during specified periods instead of the current (master) `office-hours` service configuration, for example: `office-hours-summer`, holidays.
If you add an `office-hours` service to this list (for instance, `office-hours-summer`), you must configure its _effective_dates parameter. During the effective dates, the master service uses the configuration of the alternative services instead of its own configuration.
If you set more than one service in this option, the (master) `office-hours` service uses the _timezone parameter of this list's first service (`office-hours-summer` in our example) and will use the "include rule" approach to combine the effective hours.

> ### Important
> Not supported by the Service Management UI in 8.5.106.

_effective_dates<n>

Section: office-hours
Default Value:
Valid Values: YYYY-MM-dd
or YYYY-MM-dd to YYYY-MM-dd
Changes Take Effect:

Period while the `office-hours` service applies. If the service has at least one _effective_dates parameter configured, then the office is considered closed outside of the configured _effective_dates<n> ranges.
Use this parameter to restrict the service to a given period of time (a few weeks or months). For example, 2016-04-01 or 2016-04-01 to 2016-10-31.
This parameter is mandatory if you set the _use_alternative<n> parameter.
_effective_dates<n> parameters apply to all of the parameters configured in the current office-hours service and in the other included services (for example _include_rule, _include_schedule, _use_alternative) services.

> **Important**
> Not supported by the Service Management UI in 8.5.106

Callback Services API

APIs related to Callback Services, builtin services, and ORS scenarios are detailed in GMS API References.

- Callback Services API
- Calendar API
- Capacity API

For custom samples, see:

- ClassicCallbackSample illustrates how to implement an IVR (Genesys Voice Platform VoiceXML) application that communicates with GMS and performs classic Callback scenarios.

- Custom Callback Sample implements an On-Dial plugin to interface with the GMS Callback service. Developers should use this sample as a reference to build a Composer application that is invoked as a plugin from GMS Callback.

The Service Management UI also includes a Sample panel to test your Callback Services.

Modified in: 8.5.2

Getting Started

When you add a callback service, you define a **Service Name**, which is referred to as {callback-execution-name} in this API documentation. Each time that you perform a callback query, you must specify the {callback-execution-name} in the URI parameters.

Accessing your Callback Service

The URLs used by the Callback API are dependent on the execution name of the Callback service that you have just created. Callback services are available at the following URL:

`http://<host>:<port>/genesys/1/service/callback/{callback-execution-name}`

For instance, if you create a callback service named `callback-for-mobile`, then {callback-execution-name} is `callback-for-mobile`, its configuration in GMS is located in the `service.callback-for-mobile` section, and you can access the callback service at the following URL:

`http://<host>:<port>/genesys/1/service/callback/callback-for-mobile`

Overwriting Configuration in Queries

To overwrite service configuration parameters in your POST REST queries (Start-Callback), use the _overwritable_options option. This option lets you define a list of overwritable parameters that you will be able to pass in the Body of your REST request.

> ### Important
> This list can include the `_ors` and `_target` options only.

For example, if you set:

_overwritable_options = _ors,_target

Then, you can pass _ors and _target in your REST query:

```
POST /1/service/callback/callback-for-mobile
{
     "_ors": "http://myors:4421",
    "_target": Billing@Stat_Server1.GA
}
```

Passing Configuration Tokens in Queries

Added in: 8.5.104

In your service configuration, you can create token variables that can be used in other configuration parameters. Then, at runtime, you can pass values for these tokens in POST REST queries (Start-Callback) and these values will be used to modify your configuration.

CBCK terminated preview

Q ⌄ Search Table

Select All ✏

+ Add New 🗑 Delete ⬤ A

Name	⇕	Value	⇕	Descr
🔒 _customer_number				**Reque** match USEROI custome request pa
🔒 _service		🔒 **callback**		
☐ _type		**ors**		
☐ my_token_name		**my_token**		

① Create a token variable

⌄ Chat (1)

② Use the token in your configuration:

⌄ **URS Queueing (1)**

☑ _target   Billing@my_token.GA ✏

To create a token variable, create a new service parameter and configure its value with a string matching the following format: $<any-token-name>$

For instance, create:

my_token_name = my_token

Then, you can use the body parameter my_token=<anyvalue> in your REST queries. As a result, the occurrences of my_token in this service configuration will be replaced with the query's provided value.

For example, if you wish to create a callback request for the CLBCK-terminated-preview service using the Stat_Server1 server target, use the following query:

```
POST /genesys/1/service/callback/CLBCK-terminated-preview
HTTP/1.1
Host: 127.0.0.1:8080
Cache-Control: no-cache
Content-Type: application/x-www-form-urlencoded
_customer_number=01822256&my_token=Stat_Server1
```

When GMS receives my_token=Stat_Server1 in the query information, it replaces the my_token placeholder with Stat_Server1 everywhere that it is used in the configuration of CLBCK-terminated-preview. Using our example, the result would be:

_target = Billing@Stat_Server1.GA

Tip

Use this feature to avoid duplicating configuration for multiple services that handle the same functionality, but use different queues or servers.

Understanding Callback States

When the Callback request is submitted, it gets through several callback states and ORS handles some of these callback states while processing the associated callback interaction. You can access the callback status in the _callback_state parameter of the callback's JSON representation.

Important

The _callback_state parameter is incompatible with the _new_desired_time property.
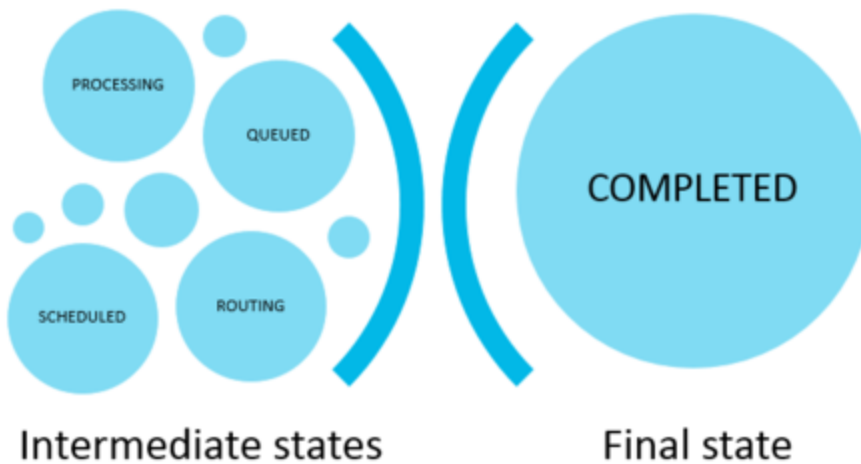
Callback states	While in ORS	Description
PROCESSING	✓	The customer is connected to an agent and talking with this agent.
QUEUED	✓	The callback is actively waiting for an agent in ORS/URS; the agent is not assigned yet.
SCHEDULED		The Callback service handles the callback (there are no sessions started in ORS). While in this state, the request is handled by the callback service running in GMS until the specified desired_time is approaching.
ROUTING	✓	Customer phone is reached and waiting for an agent.
COMPLETED		The call has ended and the Callback is completed with the reason specified in _callback_reason.
PAUSED	✓	The call is paused. See Pausing Callback for details.

Callback reasons in COMPLETED State

You can get the following reasons in the _callback_reason parameter when receiving the COMPLETED state.

ABANDONED_IN_QUEUE

The Callback interaction was deleted prior to routing the interaction to the agent because the customer abandoned.

AGENT_CONNECTED

Callback Service successfully routed the interaction to the agent.

AGENT_PREVIEW_CANCEL

The agent canceled the callback preview request. To get this state reason, create an Agent First Preview service and configure the following options with the following values, for example: _agent_preview=true, _agent_preview_allow_reject=3, _agent_preview_set_notready_reason='Coffee Break', _agent_preview_set_notready_reason_attribute=false, agent_preview_set_notready_reason_key='ReasonCode', _agent_preview_timeout_set_notready=true, _agent_preview_via_rp=false

AGENT_PREVIEW_CANCEL_AFTER_<n>REJECTS

The agent rejected the request '<n>' times.

AM_CONNECTED

Callback Service successfully routed the interaction to the answering machine.

CANCELLED

Callback Service received a cancel request for this callback.

CANCELLED_BY_ADMIN

Callback Service received a cancel request from the Service Management UI for this callback.

FAIL_AGENT_CONNECT

The Callback interaction could not be connected to the agent. This error may happen when the value of _max_time_to_wait_for_agent_on_the_call is too short.

FAIL_CALL_TO_CUSTOMER

Replaces FAIL_USER_UNREACHABLE since GMS 8.5.102.14. Callback Service could not connect the customer.

FAIL_ERROR

Callback Service failed due to an unknown error.

FAIL_FAX_REACHED

Callback Service could not connect the customer. The provided number was answered by a fax machine.

FAIL_INBOUND_TIMEOUT

The customer did not make the call within the expected _booking_expiration_timeout period defined for User-Originated scenarios.

FAIL_INCORRECT_CONFIG_MEDIA_TYPE

The _media_type option is set to an incorrect value. Callback Service only processes voice and chat interactions.

FAIL_INTERACTION_DELETED

The callback interaction was deleted prior to routing the interaction to the agent. This error may happen when _wait_for_agent=true and the agent hung up the call.

FAIL_IXN_UNKNOWN_MEDIA_TYPE

The media type of the interaction is not supported by Callback Service. Callback Service only processes voice and chat interactions.

FAIL_LOAD_MESSAGE_FILE

Callback Service cannot load the strings resource file specified in the _notification_message_file option.

FAIL_NO_CUSTOMER_NUMBER

Customer number is missing.

FAIL_QUEUEING

The Callback request could not be queued. This error may happen when an error occurs while requesting the route delay to URS.

FAIL_TARGET_NOT_FOUND

Callback Service cannot reserve the requested target to handle the request. This error may happen when the value of _urs_queued_ttl is too short.

FAIL_TIMEOUT_TTL

Callback Service did not manage to handle the request in the specified time (_ttl).

FAIL_USER_NO_CONFIRM

The user confirmation was not received although it was required; this issue can occur if _on_user_confirm_timeout is not set to CONNECT-ANYWAY.

FAIL_USER_UNREACHABLE

Reported as FAIL_CALL_TO_CUSTOMER prior to GMS 8.5.102.14.

NOT_AVAILABLE

Callback Service exited with no specified reason.

SUBMIT_ERROR

GMS did not manage to submit the Callback service request to Orchestration Server for processing.

List of API Queries

Start or Schedule Callback

Initiates a callback request. It validates the request by doing the following:

- Checks parameters, in general (in particular, if the target queue is valid).
- Checks the customer number against exceptions.
- Checks the time criteria of the request against the business.

- If invalid:

  - Returns the appropriate error.

  - Sends a reporting event to the GMS data manager indicating that the callback request has been rejected.

- If valid:

  - Creates a unique ID for the request.

  - Sends a reporting event to the GMS data manager indicating that the callback request has been accepted and started.
    This event also indicates the state of the request (immediate or scheduled).

  - If the request needs to be scheduled for a later date/time, the request and its associated data will be stored in the module persistent data storage.

  - If the request can be started now, an ORS session is initiated using the associated SCXML-based service with this particular callback request.
    Note: the provisioned data for the execution service to be started will be used as input along with the input parameters from the request itself.

  - Returns the ID generated for this request.

Starting in 8.5.2, you can redial a COMPLETED callback by submitting the callback ID to create a copy of this callback. The properties and user data of the copied callback are merged with the parameters of the new callback submitted in the POST query.

- The parameters specified in the POST query override the copied properties.

- Internal retry flags and properties such as _callback_state, _ors_session_id, _desired_time will be ignored when creating the callback copy.

> ### Tip
>
> You can include any of the **_xxx** callback option parameters in your start query if they are not configured in the service; for example **_target**, **_wait_for_agent**, **_paused_services_list**, **_paused_services_id**, or any other **_xxx** parameter listed in the *Callback Service Options Reference Guide*. If the option is already configured in the service, the query parameter's value is ignored and the service option value is used. See Overwriting Configuration in queries to learn about overwriting configuration in queries.

POST /genesys/1/service/callback/{callback-execution-name}

Initiates a callback request.

Header

Content-type	application/json

	multipart/form-data application/x-www-form-urlencoded	
URI Parameters		
Name	Type	Description
callback-execution-name *required	string *path*	Name of the callback execution service provisioned in GMS.
Body (JSON content)		
_customer_number*required	string	Number to call back. This parameter can also be replaced by any parameter specified in the option _mandatory_customer_lookup_keys (comma-separated list of attributes) that can identify a unique customer.
_copy_from_id Introduced in 8.5.2	string	ID of a Callback in COMPLETED state. The properties and user data of this completed callback are copied in the new callback and use for redial. • Properties specified in the POST request will override copied properties. • The following properties and internal retry flags will be excluded from the copy:
_desired_time	string	Desired time to have the callback. By default, the desired time is the current time. This option format is ISO 8601 "yyyy-MM-ddTHH:mm:ss.SSSZ" such as "2013-05-28T15:30:00.000Z" **Note:** The Callback is an *immediate* Callback based on the following rule: immediate = _desired_time < {current_time} + option(_request_execution_time_buffer) + computed(option(_request_queue_time_stat)) Note that the desired time is a time, not a duration. Additional examples: • _desired_time is in 1h, _request_execution_time_buffer=300 (5min), statistic set is "EstimatedWaitTime" returning, for example, 10min

		then the Callback is not immediate and will be submitted later for execution because `immediate = today 16h > today 15h + 5 min + 10 min = false` • _desired_time is in 5min, _request_execution_time_buffer=120 (2min), statistic set is "EstimatedWaitTime" returning, for example, 5min then the Callback is immediate and is submitted for execution because `immediate = today 16h05 > today 16h + 2 min + 5 min = true`
<property>	string	Any properties key/values to be attached. Key/Values may be used in Orchestration execution service. Keys without an underscore prefix are User Attached Data.
_callback_state	string	Forces creation of Callback in a specified state. **Important:** This is for advanced users that handle Callback life-cycle externally to GMS. By default, the `_callback_state` value is either QUEUED or SCHEDULED depending if the Callback is processed as immediate or scheduled (respectively).
_urs_virtual_queue	string	Queue to use for this callback if several virtual queues are used for callback with identical configuration.
_request_queue_time_stat	string	Queue statistics. For example, "ExpectedWaitTime;Queue;8999@SIP_Server;En Note: If the _request_queue_time_stat option is configured in the Callback service, the request parameter is ignored.

Responses

Name	Description
200 OK	
Response Body (JSON content)	
_id *required*	The service id for which a successful callback request was registered.
ID *only for immediate callback*	Dialog Event ID
Action *only for immediate callback*	Dialog Action.
Text *only for immediate callback*	Text to display
OkTitle *only for immediate callback*	Label for the OK button.

Name	Value
500 Internal Server Error	
Response body (JSON Content)	
code	50006
phrase	ORS_MAX_SUBMIT_RETRIES
message	"Callback {id} reached maximum attempts to submit to ORS reached ({max-attempts})"
exception	com.genesyslab.gsg.services.callback.CallbackExceptionMaxORSSubmitAttempts
properties	{ "id": "callback id", "max-attempts": <value for _max_ors_submit_attempts> }

Name	Value
429 Too Many Requests	
Response body (JSON Content)	

Name	Value
code	40001
phrase	NUMBER_ALREADY_BOOKED
message	"There is already {max_queued} or more Callbacks QUEUED for this number, please refer to _enable_in_queue_checking for detail."
exception	com.genesyslab.gsg.services.callback.CallbackExceptionAlreadyBooked
properties	{ "max_queued": <1 if _enable_in_queue_checking=strict or 2 if _enable_in_queue_checking=true>}

Name	Value
429 Too Many Requests	
Response body (JSON Content)	
code	40002
phrase	THROTTLE_SERVICE_LIMIT
message	"Limit of queued callbacks for {service} is reached."
exception	com.genesyslab.gsg.services.callback.CallbackExceptionThrottled

properties	{
	    "service": <service name>
	}

Name	Value
429 Too Many Requests	
Response body (JSON Content)	

Name	Value
code	40003
phrase	THROTTLE_SERVICE_INTERVAL_LIMIT
message	"Limit of queued callbacks for {service} is reached for interval {interval}s."
exception	com.genesyslab.gsg.services.callback.CallbackExceptionThrottled

properties	{
	
	"service": <service name>,
	
	"interval": <interval throttling limit>
	
	}

Name	Value

429 Too Many Requests

Response body (JSON Content)

code	40004
phrase	THROTTLE_SERVICE_PARAMETER_LIMIT
message	"Limit of queued callbacks for {service} is reached for parameter {parameter}. Reached {attempts} times today."
exception	com.genesyslab.gsg.services.callback.CallbackExceptionThrottled
properties	{ "service": <service name>, "parameter": <parameter triggering the throttling>, "attempts": <number of attempts reached> }

Name	Value
400 Bad Request	
Response body (JSON Content)	
code	40020
phrase	INVALID_OPERATION
message	• "Request cannot be processed because callback {id} to copy is not COMPLETED. Check parameter _copy_from_id"
exception	com.genesyslab.gsg.services.callback.CallbackExceptionInvalidOperation
properties	{"id": <callback id>, "service": <service name>, "time": <ISO UTC time>, "state": <callback state>, "message": <ORS server's message>, "filter": <filtering expression> }

Name	Value
400 Bad Request	
Response body (JSON Content)	
code	40030
phrase	CALLBACK_NOT_FOUND
message	"Callback {id} to copy from cannot be found"
exception	com.genesyslab.gsg.services.callback.CallbackExceptionNotFound
properties	{ "id": <callback service id>,

Name	Value
	"service": <service name>, "time": <ISO UTC time> }

Name	Value

400 Bad Request

Response body (JSON Content)	
code	40050, 40051
phrase	• SLOT_UNAVAILABLE (40050) • SLOT_UNAVAILABLE_PROPOSAL(40051)
message	• "No time slots available." • "Too many requests at desired time slot {slot}. Proposing time slots." • "Office is closed at desired time slot {slot}. Proposing time slots."
exception	com.genesyslab.gsg.services.callback.CallbackExceptionAvailability

properties	{
	
	"slot": <ISO UTC time range>,
	
	"service": <service name>
	
	}

Name	Value

500 Internal Server Error

Response body (JSON Content)	
code	50020

Name	Value
phrase	BAD_CONFIGURATION
message	• "Service option {service} / _default_country is not configured. But option _disallow_impossible_phone_numbers is set. We cannot validate phone numbers without knowing the country." • "Service option {service} / _default_country is not configured. But option _disallow_premium_phone_numbers is set. We cannot validate phone numbers without knowing the country." • "Unable to parse option: _request_queue_time_stat={statistic}" • "Missing default_chat_endpoint option in chat section because this service has parameter _media_type=chat" • "Missing default _client_timeout option in chat section because this service has parameter _media_type=chat" • "Option service.{service} / _business_hours_service not configured." • "Option _business_hours_service is invalid: {message}" • "Service undefined: {service}" • "Service {service} has unknown value for option _type" • "Service {service} has option _type != ors" • "Service {service} has option _service != callback"
exception	com.genesyslab.gsg.services.callback.CallbackExceptionConfiguration

properties	{
	
	   "service": <service name>
	
	}

Name	Value
500 Internal Server Error	
Response body (JSON Content)	
code	50005
phrase	CALENDAR_ERROR
message	message returned by Calendar service
exception	com.genesyslab.gsg.services.callback.CallbackExceptionCalendarError
properties	

Name	Value
500 Internal Server Error	
Response body (JSON Content)	
code	50004
phrase	CAPACITY_ERROR
message	message returned by Capacity service
exception	com.genesyslab.gsg.services.callback.CallbackExceptionCapacityError
properties	

Name	Value
500 Internal Server Error	
Response body (JSON Content)	
code	50030
phrase	ORS_ERROR

Name	Value
message	• "Invalid ORS response" • message returned by ORS strategy
exception	com.genesyslab.gsg.services.callback.CallbackExceptionFromORS
properties	

Name	Value
500 Internal Server Error	
Response body (JSON Content)	
code	50040
phrase	SERVICE_REDIRECT_FAILED
message	message from redirected service
exception	com.genesyslab.gsg.services.callback.CallbackExceptionServiceRedirect
properties	

Name	Value
400 Bad Request	
Response body (JSON Content)	
code	40040
phrase	NUMBER_REJECTED
message	• "Customer Number is not allowed, because it is invalid. Check option "Customer Number is not allowed, because it is invalid. Check option _disallow_impossible_phone_numbers" • "Customer Number is not allowed, because it's a

Name	Value
	premium number. Check option _disallow_premium_phone_numbers" • "Customer Number is not allowed, because it failed validating. Check option _disallow_impossible_phone_numbers" • "Customer Number is not allowed. Check option _exceptions"
exception	com.genesyslab.gsg.services.callback.CallbackExceptionNumber
properties	

Name	Value
500 Internal Server Error	
Response body (JSON Content)	
code	50050
phrase	UNKNOWN_ERROR
message	"Error processing callback {message} "
exception	com.genesyslab.gsg.services.callback.CallbackExceptionUnknown
properties	{ "message": <message catched> }

Example

```
POST http://localhost:8080/genesys/1/service/callback/request-callback
{
    "_customer_number": "5115",
    "usr_customer_name": "Bob Markel",
    "usr_reason": "billing question",
    "_device_notification_id":
"b16416334828b1d26ef14f329628b55b5a8c631d8928a371a5584722dd7fb673",
    "_device_os": "comet",
    "_desired_time":"2013-06-17T10:25:00.000Z"
}
```

Result

```
 200 OK
{
  "_id":"a550a12e-ca77-4146-98d0-58960e0939f7"
}
```

The result of this operation is different if the callback is immediate or schedule. If immediate, some information may be returned in response along with service_id.

```
200 OK
   {
     "ID": "0",
     "Action": "ConfirmationDialog",
     "Text": "You will receive the call shortly",
     "OkTitle": "Ok",
     "_id": "361-58ce803e-362c-477f-8ac8-5bbc93f9acc7"
   }
```

Cancel-Callback

The Cancel-Callback API cancels a Callback request, by doing the following:

- Validates that the request is still in the queue.

  - If not, returns the appropriate error.

  - If valid, removes the request from the scheduling queue.

- Checks the state of the Callback request:

  - If _callback_state=QUEUED, a callback cancel event is submitted to the execution service.

- Callback request is marked _callback_state=COMPLETED with _callback_reason=CANCELLED.

DELETE /genesys/1/service/callback/**{callback-execution-name}/{service_id}**

Cancels a Callback request

URI Parameters

Name	Type	Description
callback-execution-name *required	string *path*	Name of the callback execution service of 'ors' type provisioned in GMS.
service_id *required	string *path*	This is the service id returned from the initial start callback response.
discard_ors_failure	boolean	False by default. If true, GMS can bypass ORS failures and marks the cancellation of the callback.

		Set this option to `true` to manage troubleshoot cases that happen if the callback session is exited in ORS while the record is not marked as `COMPLETED` in GMS.

Responses

200 OK

No JSON Body

Name	Value

400 Bad Request

Response body (JSON Content)	
code	40010
phrase	BAD_PARAMETER
message	- Generic parser exception message: Typically, a bad date parsing may fall there as a bad parameter error with the appropriate statement. - Generic missing parameter exception message (case of controller level detection).
exception	com.genesyslab.gsg.services.callback.CallbackExceptionBadParameter

properties	{
	
	"id": \<callback id\>,
	
	"keys": \<missing lookup key\>,
	
	"day": \<specified day value\>,
	
	"properties": \<lookup properties\>,
	
	"option": \<invalid option key\>
	
	}

Name	Value
400 Bad Request	
Response body (JSON Content)	
code	40020
phrase	INVALID_OPERATION
message	• "Callback {id} does not contain _desired_time property." • "Callback {id} cannot be cancelled or completed - _callback_state={_callback_state}" • "Callback {id} cannot be cancelled - unable to process ORS cancel request : {message} " • "Callback {id} cannot be cancelled - No ORS session found. (_callback_state=QUEUED while _ors_session_id=null?)" • "Rejecting update : {service}=[{id} @ {time}] - reached state COMPLETED"
exception	com.genesyslab.gsg.services.callback.CallbackExceptionInvalidOperation
properties	{"id": <callback id>, "service": <service name>, "time": <ISO UTC time>, "state": <callback state>, "message": <ORS server's message>, "filter": <filtering expression> }

Name	Value
400 Bad Request	
Response body (JSON Content)	
code	40030

Name	Value
phrase	CALLBACK_NOT_FOUND
message	• "Callback {id} cannot be found" • "Callback {id} cannot be found - {service}= [{id} @ {time}]"
exception	com.genesyslab.gsg.services.callback.CallbackExceptionNotFound

properties	{
	  "id": <callback service id>,
	  "service": <service name>,
	  "time": <ISO UTC time>
	}

Name	Value
500 Internal Server Error	
Response body (JSON Content)	
code	50030
phrase	ORS_ERROR
message	• "Invalid ORS response" • message returned by ORS strategy
exception	com.genesyslab.gsg.services.callback.CallbackExceptionFromORS
properties	

Name	Value
500 Internal Server Error	

Name	Value
Response body (JSON Content)	
code	50050
phrase	UNKNOWN_ERROR
message	"Error processing callback {message} "
exception	com.genesyslab.gsg.services.callback.CallbackExceptionUnknown
properties	{ "message": <message catched> }

Examples

```
DELETE http://localhost:8080/genesys/1/service/callback/BasicCallback/a550a12e-
ca77-4146-98d0-58960e0939f7
Result 200 OK

DELETE http://localhost:8080/genesys/1/service/callback/BasicCallback/a550a12e-
ca77-4146-98d0-58960e0939f7
Result 400 Bad Request
{
    "message": "No such request to cancel : [a550a12e-ca77-4146-98d0-58960e0939f7]",
    "exception": "com.genesyslab.gsg.services.callback.CallbackException"
}

DELETE http://localhost:8080/genesys/1/service/callback/callback-test/361-cf088d4e-88ab-452c-
ac1f-39086cc96cbe
Result 400 Bad Request
{
    "message": "Request already cancelled or completed : [361-cf088d4e-88ab-452c-
ac1f-39086cc96cbe]",
    "exception":
"com.genesyslab.gsg.services.callback.exceptions.CallbackExceptionInvalidOperation"
}
```

If you set `discard_ors_failure=true`, the previous query will get a 200 OK response, though the error will be logged as an error in ORS.

```
DELETE http://localhost:8080/genesys/1/service/callback/callback-test/61-cf088d4e-88ab-452c-
ac1f-39086cc96cbe?discard_ors_failure=true

Result 200 OK
```

Reschedule-Callback

The Reschedule-Callback API changes various input parameters associated with a given callback service. This request will have the Callback request id that is to be updated. This API does the following:

- Validates that the request is still in the scheduling queue.
  - If not, returns the appropriate error.
  - If valid, updates the request in the scheduling queue.

Note: The Reschedule operation is available only for requests where _callback_state=SCHEDULED.

PUT /genesys/1/service/callback/{callback-execution-name}/{service_id}
Reschedules a Callback request

Header		
Content-type	application/json multipart/form-data application/x-www-form-urlencoded	

URI Parameters		
Name	Type	Description
callback-execution-name *required	string *path*	Name of the callback execution service of 'ors' type provisioned in GMS.
service_id *required	string *path*	This is the service id returned from the initial start callback response.

Body (JSON content)		
_new_desired_time	string	The new time for which to reschedule the callback. If provided and validated through office-hours, _callback_state will be automatically switched to "scheduled" or "immediate", discarding _callback_state property.
_callback_state	string	Possible values are SCHEDULED, QUEUED, ROUTING, PROCESSING, COMPLETED. Note: The _new_desired_time parameter triggers the re-schedule operation, discarding the _callback_state parameter.
<other properties>	any	Properties to be updated in request.

Responses

200 OK
No JSON Body

Name	Value
400 Bad Request	
Response body (JSON Content)	
code	40010
phrase	BAD_PARAMETER
message	• "Callback {id} does not contain the mandatory customer lookup keys {keys}" • "Callback {id} does not contain _desired_time property." • "Callback {id} contains _desired_time property in the past (-%ds < %ds < %ds) - epoch %ds" • "Callback request contains _desired_time property too far in future (-%ds < %ds < %ds) - epoch %ds" • "Cannot create service, missing mandatory callback option {option}" • "Cannot create service, empty mandatory callback option {option}" • Generic parser exception message: Typically, a bad date parsing may fall there as a bad parameter error with the appropriate statement. • Generic missing parameter exception message (case of controller level detection).
exception	com.genesyslab.gsg.services.callback.CallbackExceptionBadParameter
properties	{"id": <callback id>, "keys": <missing lookup key>, "day": <specified day value>, "properties": <lookup properties>, "option": <invalid option key> }

Name	Value
400 Bad Request	
Response body (JSON Content)	
code	40020
phrase	INVALID_OPERATION
message	• "Invalid service stored for callback {id}." • "Request cannot be processed because callback {id} to copy is not COMPLETED. Check parameter _copy_from_id" • "Callback {id} is no longer scheduled. State= {state}" • "Callback {id} has invalid desired time stored." • "Rejecting update : {service}=[{id} @ {time}] - reached state COMPLETED"
exception	com.genesyslab.gsg.services.callback.CallbackExceptionInvalidOperation
properties	{"id": <callback id>, "service": <service name>, "time": <ISO UTC time>, "state": <callback state>, "message": <ORS server's message>, "filter": <filtering expression> }

Name	Value
400 Bad Request	
Response body (JSON Content)	
code	40030
phrase	CALLBACK_NOT_FOUND

Name	Value
message	"Callback {id} cannot be found" "Callback {id} cannot be found - {service}= [{id} @ {time}]"
exception	com.genesyslab.gsg.services.callback.CallbackExceptionNotFound

properties	{
	
	"id": \<callback service id\>,
	
	"service": \<service name\>,
	
	"time": \<ISO UTC time\>
	
	}

Name	Value

400 Bad Request

Response body (JSON Content)	
code	40050, 40051
phrase	- SLOT_UNAVAILABLE (40050) - SLOT_UNAVAILABLE_PROPOSAL(40051)
message	"No time slots available." "Too many requests at desired time slot {slot}. Proposing time slots." "Office is closed at desired time slot {slot}. Proposing time slots."
exception	com.genesyslab.gsg.services.callback.CallbackExceptionAvailability

properties	{
	
	"slot": \<ISO UTC time range\>,
	
	"service": \<service name\>
	
	}

Name	Value
500 Internal Server Error	
Response body (JSON Content)	
code	50020
phrase	BAD_CONFIGURATION
message	• "Service option {service} / _default_country is not configured. But option _disallow_impossible_phone_numbers is set. We cannot validate phone numbers without knowing the country." • "Service option {service} / _default_country is not configured. But option _disallow_premium_phone_numbers is set. We cannot validate phone numbers without knowing the country." • "Unable to parse option: _request_queue_time_stat={statistic}" • "Missing default_chat_endpoint option in chat section because this service has parameter _media_type=chat" • "Missing default _client_timeout option in chat section because this service has parameter _media_type=chat" • "Option service.{service} / _business_hours_service not configured." • "Option _business_hours_service is invalid: {message}" • "Service undefined: {service}" • "Service {service} has unknown value for option _type" • "Service {service} has option _type != ors" • "Service {service} has option _service != callback"
exception	com.genesyslab.gsg.services.callback.CallbackExceptionConfiguration
properties	{

Name	Value
	"service": <service name> }

Name	Value
500 Internal Server Error	
Response body (JSON Content)	
code	50005
phrase	CALENDAR_ERROR
message	message returned by Calendar service
exception	com.genesyslab.gsg.services.callback.CallbackExceptionCalendarError
properties	

Name	Value
500 Internal Server Error	
Response body (JSON Content)	
code	50004
phrase	CAPACITY_ERROR
message	message returned by Capacity service
exception	com.genesyslab.gsg.services.callback.CallbackExceptionCapacityError
properties	

Name	Value
500 Internal Server Error	
Response body (JSON Content)	
code	50030
phrase	ORS_ERROR
message	• "Invalid ORS response" • message returned by ORS strategy
exception	com.genesyslab.gsg.services.callback.CallbackExceptionFromORS
properties	

Name	Value
400 Bad Request	
Response body (JSON Content)	
code	40040
phrase	NUMBER_REJECTED
message	• "Customer Number is not allowed, because it is invalid. Check option "Customer Number is not allowed, because it is invalid. Check option _disallow_impossible_phone_numbers" • "Customer Number is not allowed, because it's a premium number. Check option _disallow_premium_phone_numbers" • "Customer Number is not allowed, because it failed validating. Check option _disallow_impossible_phone_numbers" • "Customer Number is not allowed. Check option _exceptions"
exception	com.genesyslab.gsg.services.callback.CallbackExceptionNumber
properties	

Name	Value
500 Internal Server Error	
Response body (JSON Content)	
code	50050
phrase	UNKNOWN_ERROR
message	"Error processing callback {message} "
exception	com.genesyslab.gsg.services.callback.CallbackExceptionUnknown
properties	{ "message": <message catched> }

Examples

Successful Rescheduling

```
PUT http://localhost:8080/genesys/1/service/callback/
BasicCallback/a550a12e-ca77-4146-98d0-58960e0939f7
{
  "_new_desired_time":"2013-05-27T15:05:00.000Z"
}
Result
200 OK
```

Failed Rescheduling

```
PUT http://localhost:8080/genesys/1/service/callback
/callback-test/361-d61e636da-3109-436c-877e-8d7174277bb9
{
  "_new_desired_time":"2014-07-22T10:00:00.000Z"
}
Result
400 Bad Request
{
    "message": "Callback '361-738dadcb-9d20-4557-8e24-fddb82f9c1b8'
 is no longer scheduled. State=PROCESSING",
    "exception": "com.genesyslab.gsg.services.callback.exceptions
.CallbackExceptionInvalidOperation"
}
```

No availability

```
PUT http://localhost:8080/genesys/1/service
```

```
/callback/BasicCallback/a550a12e-ca77-4146-98d0-58960e0939f7
{
  "_new_desired_time":"2013-05-27T16:45:00.000Z"
}
Result
400 Bad Request
{
      "message": "Too many requests at desired time
[2013-05-27T16:45:00.000Z, 2013-05-27T16:50:00.000Z].
Proposing time slots.",
      "exception": "com.genesyslab.gsg.services.callback
.CallbackExceptionAvailability",
      "availability":
      {
          "2013-05-27T16:50:00.000Z": 5,
          "2013-05-27T16:35:00.000Z": 5,
          "2013-05-27T16:40:00.000Z": 5,
          "2013-05-27T16:55:00.000Z": 3,
          "2013-05-27T16:25:00.000Z": 5,
          "2013-05-27T16:30:00.000Z": 5
      }
}
```

Sample operation typically performed by ORS execution

```
PUT http://localhost:8080/genesys/1/service/callback
/callback-test/361-738dadcb-9d20-4557-8e24-fddb82f9c1b8
{
  "_callback_state":"PROCESSING",
  "_reason":""
}
Result
200 OK
{}
```

Delete Callback (Forget Me)

Introduced in **8.5.201**

Deletes one or more Callback Service instance(s) by passing service IDs or Customer Numbers. You can delete a Callback only if it is in SCHEDULED or COMPLETED status. This API enables you to support General Data Protection Regulation and enables you to "forget" customers.

To use this query, you need Basic Authentication. Therefore, you must provide the authentication credentials in the auth parameter of the operation. There are two ways to provide credentials in an auth object:

- In an open form containing the username and password fields of a user defined in the Configuration Server.

- In an encoded form using encoded fields, similar to the Basic Authentication header, which is a Base64-encoded composite string of "username:password".

POST /genesys/1/admin/callback/ops/delete

Deletes one or more callback request(s).		
Header		
Content-type	application/json	
Body (JSON content)		
_customer_number	String array	List of Customer Numbers or Service IDs that identify the callback service instances that must be deleted.
_id	String array	List of service IDs that identify the callback service instances that must be deleted.

Responses

Name	Description	
200 OK		
Response Body (JSON content)		

success
required
	Array	Array of service IDs and Customer Numbers that were deleted or were considered as successful with a reason.
		    [{ "_id": "68542134" },
		    { "reason": "no
		    callback(s) to delete",
		    "_customer_number":
		    "132456" }]

errors
required
	Array	Array of service IDs and Customer Numbers that were **not** deleted with the associated error codes.
		    [{ "non-existing-lookup-
		    key": "132456", "code":
		    40010, "phrase":
		    "BAD_PARAMETER",
		    "message": "No such
		    lookup possible for
		    {properties}" },
		
		{ "code": 40020, "phrase":
		"INVALID_OPERATION", "_id":
		"118-576b21b4-a235-4ba5-92d4-102cbbb54bca",
		"message": "Callback
		118-576b21b4-a235-4ba5-92d4-102cbbb54bca
		cannot be deleted -
		_callback_state=PROCESSING" }]

Name	Value
500 Internal Server Error	
Response body (JSON Content)	
code	50020
phrase	BAD_CONFIGURATION
message	• "Service option {service} / _default_country is not configured. But option _disallow_impossible_phone_numbers is set. We cannot validate phone numbers without knowing the country." • "Service option {service} / _default_country is not configured. But option _disallow_premium_phone_numbers is set. We cannot validate phone numbers without knowing the country." • "Unable to parse option: _request_queue_time_stat={statistic}" • "Missing default_chat_endpoint option in chat section because this service has parameter _media_type=chat" • "Missing default _client_timeout option in chat section because this service has parameter _media_type=chat" • "Option service.{service} / _business_hours_service not configured." • "Option _business_hours_service is invalid: {message}" • "Service undefined: {service}" • "Service {service} has unknown value for option _type" • "Service {service} has option _type != ors" • "Service {service} has option _service != callback"
exception	com.genesyslab.gsg.services.callback.CallbackExceptionConfiguration

properties	{
	
	  "service": <service name>
	
	}

Name	Value
400 Bad Request	
Response body (JSON Content)	
code	40010
phrase	BAD_PARAMETER
message	• "No such lookup possible for {properties}" • "No lookup possible. No properties to look for." • Generic parser exception message: Typically, a bad date parsing may result in a bad parameter error with the appropriate statement. • Generic missing parameter exception message (case of controller level detection).
exception	com.genesyslab.gsg.services.callback.CallbackExceptionBadParameter

properties	{
	
	"id": <callback id>,
	
	"keys": <missing lookup key>,
	
	"day": <specified day value>,
	
	"properties": <lookup properties>,
	
	"option": <invalid option key>
	
	}

Name	Value
400 Bad Request	
Response body (JSON Content)	
code	40020

Name	Value
phrase	INVALID_OPERATION
message	"Cannot process 'filter' parameter correctly : {filter}"
exception	com.genesyslab.gsg.services.callback.CallbackExceptionInvalidOperation
properties	{"id": <callback id>, "service": <service name>, "time": <ISO UTC time>, "state": <callback state>, "message": <ORS server's message>, "filter": <filtering expression> }

Name	Value
500 Internal Server Error	
Response body (JSON Content)	
code	50050
phrase	UNKNOWN_ERROR
message	"Error processing callback {message} "
exception	com.genesyslab.gsg.services.callback.CallbackExceptionUnknown
properties	{ "message": <message catched> }

Example

```
POST http://localhost:8080/genesys/1/admin/callback/ops/delete
{
        "_id": ["118-576b21b4-a235-4ba5-92d4-102cbbb54bca"],
        "_customer_number": [
                "132456",
```

```
            "1111",
            "3333"
        ]
}
```

Result

```
Response: 200 OK
{
        "success": [
                {
                        "reason": "no callback(s) to delete",
                        "_customer_number": "132456"
                },
                {
                        "_id": "118-27f3bed5-6e3a-4c89-903f-dae562b30481"
                },
                {
                        "_id": "118-c2ce7a84-d33a-4d8d-88a0-b76a563f2324"
                }
        ],
        "errors": [
                {
                        "code": 40020,
                        "phrase": "INVALID_OPERATION",
                        "_id": "118-576b21b4-a235-4ba5-92d4-102cbbb54bca",
                        "message": "Callback 118-576b21b4-a235-4ba5-92d4-102cbbb54bca cannot
be deleted - _callback_state=PROCESSING"
                }
        ]
}
```

Query Callback By ID

Introduced in 8.5.207

Retrieves a callback request by its ID.

GET /genesys/2/service/callback/{callback-execution-name}/{id}
Queries the outstanding callback associated with a given ID.

URI Parameters

Name	Type	Description
callback-execution-name *required	string *path*	Name of the callback execution service of 'ors' type provisioned in GMS.
id *required	string *path*	Callback ID.

Responses

Name	Description
200 OK	
Response Body (JSON content)	

<none>	- If accepted, the currently outstanding callback request.
	`[`
	
	    `{`
	      `"_id": <callback id>,`
	      `"desired_time": <ISO UTC time>,`
	      `"url": <service URL>,`
	      `"_expiration_time": <ISO UTC time>,`
	      `"_service_name": <service-name>,`
	      `"_customer_number": <customer`
	`number>,`
	      `"_callback_state": <callback state>,`
	      `"_time_scheduled": <ISO UTC time>`
	    `}`
	 `]`
	
	   - If not, an error code indicating the reason.

Name	Value
400 Bad Request	
Response body (JSON Content)	
code	40010
phrase	BAD_PARAMETER
message	"No such lookup possible for {properties}" "No lookup possible. No properties to look for." - Generic parser exception message: Typically, a bad date parsing may result in a bad parameter error with the appropriate statement. - Generic missing parameter exception message (case of controller level detection).
exception	com.genesyslab.gsg.services.callback.CallbackExceptionBadParameter
properties	`{`

Name	Value
	"id": <callback id>, "keys": <missing lookup key>, "day": <specified day value>, "properties": <lookup properties>, "option": <invalid option key> }

Name	Value

400 Bad Request

Response body (JSON Content)	
code	40020
phrase	INVALID_OPERATION
message	"Cannot process 'filter' parameter correctly : {filter}"
exception	com.genesyslab.gsg.services.callback.CallbackExceptionInvalidOperation
properties	{"id": <callback id>, "service": <service name>, "time": <ISO UTC time>, "state": <callback state>, "message": <ORS server's message>, "filter": <filtering expression> }

Name	Value

500 Internal Server Error

Name	Value
Response body (JSON Content)	
code	50050
phrase	UNKNOWN_ERROR
message	"Error processing callback {message} "
exception	com.genesyslab.gsg.services.callback.CallbackExceptionUnknown
properties	{ "message": <message catched> }

Example

```
GET http://localhost:8080/genesys/1/service/callback/BasicCallback/120-07f85068-650d-4cce-
a5e7-396dfa22455b
```

Result

```
200 OK

{
  "_callback_state": "SCHEDULED",
  "_expiration_time": "2020-05-11T11:59:59.000Z",
  "_service_name": "BasicCallback",
  "_id": "124-07f85068-650d-4cce-a5e7-396dfa22455f",
  "_customer_number": "12345",
  "_url": "/genesys/1/service/callback/BasicCallback/120-07f85068-650d-4cce-
a5e7-396dfa22455b",
  "_time_scheduled": "2020-04-16T12:52:31.521Z",
  "_desired_time": "2020-04-27T12:00:00.000Z"
}
```

Query-Callback by Lookup Properties

Modified in 8.5.111

The Query-Callback API queries the current set of outstanding Callback services associated with a given property.

Notes:

- Outstanding Callback services are requests where _callback_state is one of the following values: SCHEDULED, QUEUED, ROUTING, PROCESSING, COMPLETED.

- Properties allowing the Callback request trackback are defined as comma-separated keys with the service option _customer_lookup_keys.

- The API returns each callback for which the looked-up property is or was equal to the value specified in the requested property.

- Starting in 8.5.111, you can configure the list of values to be retrieved when calling this query by setting the returned-keys option at the GMS application level.

- To use the _customer_number lookup property regardless of whether you specify a callback service name or not in the API URL, the _fix_plus_on_int_phone_numbers option must be identical in the callback section and in each service-specific section.

  - This is the expected behavior if you stick to defaults.

    - If a callback service has a distinct value for _fix_plus_on_int_phone_numbers, you can only use the _customer_number lookup property by specifying the service name in the API URL.

GET /genesys/1/service/callback/{callback-execution-name}?{property=value}

GET /genesys/1/service/callback?{property=value}

Queries the current set of outstanding Callback services associated with a given property.

URI Parameters

Name	Type	Description
callback-execution-name	string *path*	Name of the callback execution service of 'ors' type provisioned in GMS.
property=value *required*	string *path*	This is a property name used to query the callback. Properties allowing the Callback request trackback are defined as comma-separated keys with the service option _customer_lookup_keys.

		If you specify several properties, you may need to use the operand property.
operand	string	Operand to use for the properties defined in the service option _customer_lookup_keys. Possible values are AND or OR. Default is AND. When multiple **property=value** are provided in the query, the operand specifies which operation to perform on matched Callback requests: • AND means that all **property=value** must match; • OR means any **property=value** can match.
_callback_state Since 8.5.101.03	string	Specifies a unique state to filter onto. For example: • `_callback_state='COMPLETED'` filters callbacks and returns only callbacks in COMPLETED state. • `_callback_state='!COMPLETED'` filter callbacks and only return the ones that are not COMPLETED. > **Important** > The character "!" is used to negate a case. You can query the following callback states: `SCHEDULED, QUEUED, ROUTING, PROCESSING, COMPLETED`.
_desired_time_from Since 8.5.101.03	string	Specifies ISO timestamps. All callback matching lookup properties that were scheduled before this time will be filtered out.

_desired_time_to Since 8.5.101.03	string	Specifies ISO timestamps. All callback matching lookup properties that were scheduled after this time will be filtered out.

Responses

Name	Type	Description
200 OK		
Response Body (JSON content)		

<none>	• If accepted, JSON array of service IDs of the currently outstanding callback requests.
	
	`[`
	
	`    {`
	`        "_id": <callback id>,`
	`        "desired_time": <ISO UTC time>,`
	`        "_callback_state": <callback state>,`
	`        "_expiration_time":<ISO UTC time>,`
	`        "_customer_number": <customer number>,`
	`        "url": <service URL>`
	`    },`
	
	`…`
	
	` ]`
	
	• If not, an error code indicating the reason.	

Name	Value
400 Bad Request	
Response body (JSON Content)	
code	40010
phrase	BAD_PARAMETER
message	"No such lookup possible for {properties}" "No lookup possible. No properties to look for." - Generic parser exception message: Typically, a bad date parsing may result in a bad parameter error with the appropriate statement. - Generic missing parameter exception message (case of controller level detection).
exception	com.genesyslab.gsg.services.callback.CallbackExceptionBadParameter

properties	{
	
	"id": <callback id>,
	
	"keys": <missing lookup key>,
	
	"day": <specified day value>,
	
	"properties": <lookup properties>,
	
	"option": <invalid option key>
	
	}

Name	Value
400 Bad Request	
Response body (JSON Content)	
code	40020
phrase	INVALID_OPERATION
message	"Cannot process 'filter' parameter correctly : {filter}"
exception	com.genesyslab.gsg.services.callback.CallbackExceptionInvalidOperation
properties	{"id": <callback id>, "service": <service name>, "time": <ISO UTC time>, "state": <callback state>, "message": <ORS server's message>, "filter": <filtering expression> }

Name	Value
500 Internal Server Error	
Response body (JSON Content)	
code	50050
phrase	UNKNOWN_ERROR
message	"Error processing callback {message} "
exception	com.genesyslab.gsg.services.callback.CallbackExceptionUnknown
properties	{ "message": <message catched> }

Example

```
GET http://localhost:8080/genesys/1/service/callback
/BasicCallback?_customer_number=555-5461206
```

Result

```
200 OK
[
   {
      "_id": "a550a12e-ca77-4146-98d0-58960e0939f7",
      "desired_time": "2013-05-27T15:05:00.000Z",
      "_callback_state": "QUEUED",
      "_expiration_time": "2014-11-03T18:36:45.000Z",
      "_customer_number": "555-5461206",
      "url": "/1/service/callback/BasicCallback/a550a12e-ca77-4146-98d0-58960e0939f7"
```

```
  },
  {
    "_id": "4a1ea889-1ef7-432d-a543-cff96b4a2daf",
    "desired_time": "2013-05-27T15:10:00.000Z",
    "_callback_state": "SCHEDULED",
    "_expiration_time": "2014-11-03T18:36:45.000Z",
    "_customer_number": "555-5461206",
    "url": "/1/service/callback/BasicCallback/4a1ea889-1ef7-432d-a543-cff96b4a2daf"
  }
]
```

Query-Availability

v1

Query-Availability v1

This query returns a simple map of slots in which the office capacity is not full.

GET /genesys/1/service/callback/{callback-execution-name}/availability

Returns a simple map of slots in which the office capacity is not full.

URI Parameters		
Name	Type	Description
callback-execution-name *required	string *path*	Name of the callback execution service of 'ors' type provisioned in GMS.
JSON Body		
start	date	Start date is specified in ISO 8601 format, using UTC as the timezone: yyyy-MM-ddTHH:mm:ss.SSSZ. If not specified, it is assumed to be now.
timestamp	date	Alias to start parameter; kept for compatibility reasons.
number-of-days	integer	Used as an alternative to the end date. If neither end nor number-of-days is specified, the end date is assumed to be the same as the start date.
end	date	End date is specified in ISO 8601 format, using UTC as timezone: yyyy-MM-ddTHH:mm:ss.SSSZ. If neither end nor number-of-days is specified, the end date is assumed to be the same as the start date.
max-time-slots	integer	Maximum number of time slots to be included in the response when the office is open and capacity is above zero. It can be used to improve the performance of the query over a long period of time.

> **Important**
>
> If neither of the parameters `number-of-days` and end parameters are specified, the default time range matches 1 bucket only (as configured in the _request_time_bucket service option).

Request example:

```
GET http://localhost:8080/genesys/1/service/callback/Callback_VQ/
availability?start=2014-12-03T15:00:00.000Z
```

Response

The Callback controller provides a facet to the availability service, which uses the calendar service underneath. Just as the calendar service takes three non-mandatory input parameters (start, number-of-days, end), the availability service should accept the same parameters and pass them on to the calendar service.

- The response contains a map of time slots and capacity counters.

- The slots are ordered in ascending order.

- Any time slots where the capacity is full (for example, zero) are not provided in the response. Similarly, if the office is closed, those time slots are not provided in the response.

```
200 OK
{
    // All periods are ordered in ascending time order
    "2014-10-17T13:00:00.000Z":"5",
    "2014-10-17T13:10:00.000Z":"4",
    // there were no agents available between 13:20 and 13:30 UTC
    //hence the time slot is not reported
    "2014-10-17T13:30:00.000Z":"5"
}
```

v2

Query-Availability v2

This query includes more query options than v1 and returns an array of ordered slots that include detailed capacity information and timezone information.

GET /genesys/2/service/callback/{callback-execution-name}/availability
> Returns an array of ordered slots that include detailed capacity information and timezone information.

URI Parameters

Name	Type	Description
callback-execution-name *required*	string 　　*path*	Name of the callback execution service of 'ors' type provisioned in GMS.
start	date	Start date in the "ISO 8601" format, using the UTC timezone: "yyyy-MM-ddTHH:mm:ss.SSSZ". If not specified, the default start date is the date on which the query was submitted. • If you set the `start` parameter, do not set the `start-ms` or `timestamp` parameters. • You must also set the end or `number-of-days` parameter; otherwise, the end date is assumed to be the `start` date.
start-ms	long	Start date in epoch time, that is, the number of milliseconds since `00:00:00, Thursday, 1 January 1970 (UTC)`. • You must also set the `end-ms` or `number-of-days` parameter; otherwise, the end date is assumed to be the `start-ms` date. • If you set the `start-ms` parameter, do not set the `start` or `timestamp` parameters.
number-of-days	integer	Number of days used to define the availability period starting at the `start` or `start-ms` date. You can use this parameter instead of the end or of the `end-ms` parameter.
end	date	End date, in "ISO 8601" format, using the UTC timezone: `yyyy-MM-ddTHH:mm:ss.SSSZ`. By default, if neither the "end" nor the "number-of-days" parameter is specified, then the end date is assumed to be the start date.
end-ms	long	End date in epoch time, that is the number of milliseconds since `00:00:00, Thursday, 1 January 1970 (UTC)`.

		Set only one of the end, end-ms, or number-of-days parameters.
max-time-slots	integer	Maximum number of time slots to include in the response if the office is open and the capacity greater than zero. You can use this parameter to improve query performance over a lengthy period of time.
timezone	string	Timezone for the start and end date parameters. Additionally, the response object will return the localTime fields formatted in this timezone.
report-busy	boolean	If true, the response includes the slots where the office is open and where callbacks are booked to full capacity. By default, report-busy is false.
JSON body: **None**.		

> ## Important
>
> If neither of the parameters number-of-days, end, and end-ms parameters are specified, the default time range matches 1 bucket only (as configured in the _request_time_bucket service option).

Responses

If successful, the response returns multiple values that describe the slots, availability, and capacity for a given slot.

Name	Type	Description
200 OK		
Response Body (JSON content)		
slots *required*	String array of slots	Array of ordered slots and each slot includes the minute duration (durMinutes), and the timezone. • The array of slots includes detailed information about each slot. • Slots are sorted in ascending

Name	Type	Description

		order by their time.
		
		• Slots are all the same duration, specified in the durMinutes value.
		
		• The "timezone" value specifies the timezone used for the "localTime" fields in slots' information.
		
		``
		{
		  "slots": [
		    {
		      "utcTime": <UTC time>,
		      "localTime": <UTC time>,
		      "capacity": <capacity>,
		      "total": <total>
		    },
		    (...)]
		  "durationMin": <duration in minutes>,
		  "timezone": <timezone>
		}
		``
		Each slot includes:
		
		• "utcTime" specifies when this slot begins in UTC time.
		
		• "localTime" reports the same time as "utctime", but formatted using the "timezone" set in the request.
		
		• "capacity" value is the number of available callbacks that can be scheduled within this timeslot.
		
		• "total" is the total capacity that is configured for this timeslot.

Name	Value
400 Bad Request	
Response body (JSON Content)	
code	40010

Name	Value
phrase	BAD_PARAMETER
message	• "day parameter must be between 1 and 7, inclusively. Actual value is: {day}" • "No time slots available. The requested time period is in the past." • Generic parser exception message: Typically, a bad date parsing may fall there as a bad parameter error with the appropriate statement. • Generic missing parameter exception message (case of controller level detection).
exception	com.genesyslab.gsg.services.callback.CallbackExceptionBadParameter

properties	{
	"id": <callback id>,
	"keys": <missing lookup key>,
	"day": <specified day value>,
	"properties": <lookup properties>,
	"option": <invalid option key>
	}

Name	Value
400 Bad Request	
Response body (JSON Content)	
code	40050, 40051
phrase	• SLOT_UNAVAILABLE (40050) • SLOT_UNAVAILABLE_PROPOSAL(40051)
message	• "No time slots available." • "Too many requests at desired time slot {slot}.

Name	Value
	Proposing time slots." • "Office is closed at desired time slot {slot}. Proposing time slots."
exception	com.genesyslab.gsg.services.callback.CallbackExceptionAvailability

properties	{
	
	 "slot": <ISO UTC time range>,
	
	 "service": <service name>
	
	 }

Name	Value

500 Internal Server Error

Response body (JSON Content)	
code	50020
phrase	BAD_CONFIGURATION
message	• "Option service.{service} / _business_hours_service not configured." • "Option _business_hours_service is invalid: {message}" • "Service undefined: {service}" • "Service {service} has unknown value for option _type" • "Service {service} has option _type != ors" • "Service {service} has option _service != callback"
exception	com.genesyslab.gsg.services.callback.CallbackExceptionConfiguration
properties	{ "service": <service name>

Name	Value
	}

Name	Value
500 Internal Server Error	
Response body (JSON Content)	
code	50005
phrase	CALENDAR_ERROR
message	message returned by Calendar service
exception	com.genesyslab.gsg.services.callback.CallbackExceptionCalendarError
properties	

Name	Value
500 Internal Server Error	
Response body (JSON Content)	
code	50004
phrase	CAPACITY_ERROR
message	message returned by Capacity service
exception	com.genesyslab.gsg.services.callback.CallbackExceptionCapacityError
properties	

Name	Value
500 Internal Server Error	
Response body (JSON Content)	
code	50050
phrase	UNKNOWN_ERROR
message	"Error processing callback {message} "
exception	com.genesyslab.gsg.services.callback.CallbackExceptionUnknown
properties	{ "message": <message catched> }

Examples

Request example:

```
http://localhost:8010/genesys/2/service/callback/callback-PST
/availability?start=2016-04-13T09:00:00.000&end=2016-04-13T16:00:00.000
&timezone=America/Toronto
```

```
{
  "slots": [
    {
      "utcTime": "2016-04-13T13:00:00.000Z",
      "localTime": "2016-04-13T09:00:00.000",
      "capacity": 42,
      "total": 100
    },
    {
      "utcTime": "2016-04-13T13:05:00.000Z",
      "localTime": "2016-04-13T09:05:00.000",
      "capacity": 67,
      "total": 100
    },
    {
      "utcTime": "2016-04-13T13:10:00.000Z",
      "localTime": "2016-04-13T09:10:00.000",
      "capacity": 91,
      "total": 100
    }
    ...
  ],
  "durationMin": 5,
  "timezone": "Eastern Standard Time"
}
```

> **Important**
>
> Existing calendar configurations must be updated for the time zone definition. Instead of EST or PST time zones that were configured using Configuration Manager, you must use time zones as allowed in Java (http://en.wikipedia.org/wiki/List_of_tz_database_time_zones), such as `America/Toronto`, or `Europe/Paris`. You must also change the service option _type from `ors` to `builtin`.

Query-Callback by Queue(s)

Modified in 8.5.111

The Query-Callback API queries the current set of outstanding Callback services in the given queue(s).

Starting in 8.5.111, you can filter and configure the list of values to be passed and retrieved when calling this query through the following options at the GMS application level: returned-keys and filter-keys.

> **Important**
>
> Outstanding Callback services are requested if their `_callback_state` is one of the following values: `SCHEDULED, QUEUED, ROUTING, PROCESSING, COMPLETED`.

To use this query, you need Basic Authentication. Therefore, you must provide the authentication credentials in the `auth` parameter of the operation. There are two ways to provide credentials in an `auth` object:

- In an open form containing the username and password fields of a user defined in the Configuration Server.

- In an encoded form using encoded fields, similar to the Basic Authentication header, which is a Base64-encoded composite string of "`username:password`".

GET /genesys/1/admin/callback/queues?target={callback-execution-name}☆t_time={iso_start_time}&end_time={iso_end_time}

Queries the current set of outstanding Callback services in given queue(s).

URI Parameters

Name	Type	Description
{iso_start_time}	string	This is the minimum time for

		which to query callback requests. The format is ISO 8601 "yyyy-MM-ddTHH:mm:ss.SSSZ". For example: "2013-05-27T15:30:00.000Z"
{iso_end_time}	string	This is the maximum time for which to query callback requests. If not specified, requests that are due in the next 24 hours are returned. The format is ISO 8601 "yyyy-MM-ddTHH:mm:ss.SSSZ". For example: "2013-05-28T15:30:00.000Z"
{states}	string	Comma-separated list of callback states used to filter the returned results. For example, if `states=SCHEDULED,QUEUED`, only scheduled and queued callbacks are returned. If not specified, all the callbacks of the given queue are returned.
{max}	integer	This is the maximum number of requests to return for each queue. If not specified, 500 maximum requests per queue are returned.
callback-execution-name	string	Name of the callback execution service provisioned in GMS. For example, BasicCallback. If not specified, the queues for all services are returned.
{max}	integer	This is the maximum number of requests to return for each queue. If not specified, 500 maximum requests per queue are returned.

Responses

Name	Mandatory	Description
200 OK		
Response Body (JSON content)		

Name	Mandatory	Description

List of target queues
required
	string	If accepted, a tree list of target queues and the following properties:
		
		<Queue name>: {
		
		"_customer_number": <customer number>,
		
		"_callback_state": <callback state>,
		
		"_desired_time": <callback UTC desired time>,
		
		"_id": <callback service id>,
		
		"url": <request>
		
		}

Name	Value
400 Bad Request	
Response body (JSON Content)	
code	40020
phrase	INVALID_OPERATION
message	"Query range spans too wide time range (%d / %d). Adjust query parameters for time range."
exception	com.genesyslab.gsg.services.callback.CallbackExceptionInvalidOperation
properties	{"id": <callback id>, "service": <service name>, "time": <ISO UTC time>, "state": <callback state>, "message": <ORS server's message>, "filter": <filtering expression> }

Name	Value
500 Internal Server Error	
Response body (JSON Content)	
code	50050
phrase	UNKNOWN_ERROR
message	"Error processing callback {message} "
exception	com.genesyslab.gsg.services.callback.CallbackExceptionUnknown
properties	{ "message": <message catched> }

Example

```
GET http://localhost:8080/genesys/1/admin/callback/queues
```

Result

```
200 OK

    {
      "BasicCallback":
      [
          {
              "_customer_number": "654321",
              "_callback_state": "PROCESSING",
              "_desired_time": "2013-06-07T16:25:00.000Z",
              "_id": "fd30abb97bd04885b544893276fb534b",
              "url": "/1/service/callback/BasicCallback/fd30abb97bd04885b544893276fb534b"
          }
      ],
      "AdvancedCallback":
      [
          {
              "_customer_number": "654321",
              "_callback_state": "QUEUED",
              "_desired_time": "2013-06-07T16:35:00.000Z",
              "_id": "07d2ddd506f04b4ba91aba59c4fa8871",
              "url": "/1/service/callback/AdvancedCallback/07d2ddd506f04b4ba91aba59c4fa8871"
          },
          {
              "_customer_number": "654321",
              "_callback_state": "SCHEDULED",
              "_desired_time": "2013-06-07T16:45:00.000Z",
              "_id": "8f68d4969d904d039ccf0101fac39283",
              "url": "/1/service/callback/AdvancedCallback/8f68d4969d904d039ccf0101fac39283"
```

```
        }
    ]
}
```

Query Counter Watermarks

This query counts the current set of executed callback instances per queues or for a given queue. Executed callback instances are:

- Callbacks that are in execution within ORS

- Callbacks do not have their _callback_state property set to SCHEDULED

- Callbacks do not have their _callback_state property set to COMPLETED in GMS storage. Callbacks in such a state for more than 3 hours are discarded.

To use this query, you need Basic Authentication. Therefore, you must provide the authentication credentials in the auth parameter of the operation. There are two ways to provide credentials in an auth object:

- In an open form containing the username and password fields of a user defined in the Configuration Server.

- In an encoded form using encoded fields, similar to the Basic Authentication header, which is a Base64-encoded composite string of "username:password".

Important

You can use this API to ensure that you do not book more Callbacks than you have licenses for.

GET /genesys/1/admin/callback/watermarks?service_name={callback-execution-name}

GET /genesys/1/admin/callback/watermarks

Counts the current set of executed callback instances per queues or for a given queue.

URI Parameters

Name	Type	Description
{callback-execution-name}	string	Name of a callback execution service. If you set this parameter, the response will return the watermarks for the specified service only. If the service name is not set, the response returns

		the total count of executed callback instances in queues and the count per service.
		
		You can query watermarks for several services in a single query. To do so, add as many `service_name` values as you need to your query:
		
		`GET /genesys/1/admin/`
		`callback/`
		`watermarks?service_name=service1&service_nam`

Responses

HTTP code	200
HTTP message	OK

Response Body (JSON content)	If accepted, a list of target queues and the count of callbacks that are in execution within ORS or that do not have their `_callback_state` property set to `SCHEDULED` or `COMPLETED`) in GMS storage.
	
	`{`
	
	`    "total": <total of callbacks in progress>,`
	`    "services": {`
	`      <service-1>: <number of callbacks in progress for service-1 >,`
	`        ...`
	`      <service-n>: <number of callbacks in progress for service-n>,`
	`    }`
	`  }`

Name	Value
500 Internal Server Error	
Response body (JSON Content)	
code	50050
phrase	UNKNOWN_ERROR
message	"Error processing callback {message} "
exception	com.genesyslab.gsg.services.callback.CallbackExceptionUnknown
properties	{ "message": <message catched> }

Example

Operation

GET `http://localhost:8080/genesys/1/admin/callback/watermarks`

Result

200 OK

```
{
  "total": 1,
  "services": {
    "callback-immediate": 0,
    "callback-test": 1
  }
}
```

GET `http://localhost:8080/genesys/1/admin/callback/watermarks?service_name=callback-immediate`

Result

200 OK
```
{
  "total": 0,
  "services": {
    "callback-immediate": 0
  }
}
```

Check in Queue Position

This query enables your application to query for the position and Estimated Wait Time while the GMS Service request is in QUEUED status. This query is used to provide additional details in the Callback UI.

Name	Type	Description
POST /genesys/1/service/{callback-service-id}/check-queue-position		
BODY Parameters		
{callback-service-id} **required**	string	ID of the callback execution service. For example, `445-f4fa53ec-8e93-4836-ba35-f0bd74a025a8.`

> ### Important
> The GET method is also supported for this feature.

Response

HTTP code	200
HTTP message	OK
Response Body (JSON content)	JSON-formatted information for the given service ID: • `app_version`: Callback strategy version. • `wt`: The time that the call has waited in queue. • `connid`: Interaction ID in the Virtual Queue. • `ewt`: The estimated time that customer will wait for the callback. • `positioninqueue`: The callback's current position in the queue. • `_position`: position of the interaction in the virtual queue (top position = 1). • `_eta`: estimated wait time to the agent availability. • `_total_waiting`: total number of requests waiting in queue. • `priority`: The callback priority in the Virtual Queue. • `agents_logged_in`: The number of agents that have logged in. • `ors_session_id`: ORS session ID of the callback. • `ewt_at_offer`: The estimated wait time when the callback is offered. • `pos_at_offer`: The callback's position in the queue when the callback is offered. • `callback_type`: The type of callback. • `time_callback_accepted`: The time when the callback is accepted. • `channel`: The callback channel. • `skill_expression`: The callback's target or skill expression. • `ewt_at_first_dial`: The estimated wait time when the first outbound call happened. • `pos_at_first_dial`: The callback's position in the queue when the first outbound call happened. • `time_at_first_dial`: The time when the first outbound call happened.

	• dial_attempt: The number of dials that agent has attempted.
	• is_snoozed: If true, shows that the callback is snoozed.
	• dial_result: The result of callback dial.
	• time_customer_connected: The time when the customer connected.

Errors

Name	Value
500 Internal Server Error	
Response body (JSON Content)	
code	50050
phrase	UNKNOWN_ERROR
message	"Error processing callback {message} "
exception	com.genesyslab.gsg.services.callback.CallbackExceptionUnknown
properties	{ "message": <message catched> }

Example

Operation

```
POST /genesys/1/service/445-f4fa53ec-8e93-4836-ba35-f0bd74a025a8/check-queue-position HTTP/1.1
Connection: close
Content-Length: 0
Content-Type: application/x-www-form-urlencoded; charset=UTF-8
```

Response:

```
200 OK
{
    "app_version":"v2.41",
    "wt":26,
    "connid":"006e02aea54bc008",
    "ewt":0,
    "positioninqueue":0,
    "_position":1,
    "_eta":0,
    "_total_waiting":1,
    "priority":500,
```

```
    "agents_logged_in":3,
    "ors_session_id":"00ACLU5N00CV19601K015B5AES000003",
    "ewt_at_offer":0,
    "pos_at_offer":0,
    "callback_type":"WAIT_FOR_AGENT",
    "time_callback_accepted":1508959666,
    "channel":"WEB",
    "skill_expression":"GMSCallbackAgents@stat.GA",
    "ewt_at_first_dial":"100.0",
    "pos_at_first_dial":"1",
    "time_at_first_dial":1508959684,
    "dial_attempt":1,
    "is_snoozed":false,
    "dial_result":"PERSON",
    "time_customer_connected":1508959690
}
```

Export Cancelled Callback Records

Added in: 8.5.110
This query exports the callbacks that were cancelled by the Service Management UI only (Bulk Cancel).

- The data will be exported in CSV format.

- The request will export the records cancelled from the last 30 days to the next 15 days.

- You can export additional fields with the retrieved callback records.

By default, the CSV report includes the following default properties: `_desired_time`, `_service_name`, `_customer_number`, `urs_virtual_queue`, `_vq_for_outbound_calls`, and `target`.

To use this query, you need Basic Authentication. Therefore, you must provide the authentication credentials in the auth parameter of the operation. There are two ways to provide credentials in an auth object:

- In an open form containing the username and password fields of a user defined in the Configuration Server.

- In an encoded form using encoded fields, similar to the Basic Authentication header, which is a Base64-encoded composite string of "`username:password`".

Name	Type	Description
POST /genesys/1/admin/callback/reportcancelled		
BODY Parameters		
callback_reason *required*'	string	The reason for the cancellation. For example, `CANCELLED_BY_ADMIN`.
exported_properties	string	List of properties to export for

Name	Type	Description
		each selected record. For example: ["_service_id,_desired_time"]. If this parameter is empty or missing, the following properties will be exported by default: _desired_time, _service_name, _customer_number, urs_virtual_queue, _vq_for_outbound_calls, and target.

Response

HTTP code	200
HTTP message	OK
Response Body (JSON content)	CSV-formatted results for exported records: <property-1>,<property-2>,...,<property-n> <record-1-property1>,<record-1-property2>,...,<record-1-pro ... <record-n-property1>,<record-n-property2>,...,<record-n-propertyn>

Errors

HTTP code	400
HTTP message	Callback reason is missing.

HTTP code	204
HTTP message	No record found.

Name	Value
500 Internal Server Error	
Response body (JSON Content)	
code	50050
phrase	UNKNOWN_ERROR
message	"Error processing callback {message} "
exception	com.genesyslab.gsg.services.callback.CallbackExceptionUnknown

Name	Value
properties	{ "message": <message catched> }

Example

Operation

```
POST /genesys/1/admin/callback/reportcancelled
{
"callback_reason": "CANCELLED_BY_ADMIN",
"exported_properties": []
}'
```

Response:

```
Access-Control-Allow-Credentials →true
Access-Control-Allow-Origin →chrome-extension://aicmkgpgakddgnaphhhpliifpcfhicfo
Access-Control-Expose-Headers →
Content-Disposition →attachment; filename="report.csv"
_desired_time,_service_name,_customer_number,_target,_vq_for_outbound_calls,_urs_virtual_queue
2017-07-04T22:00:00.000Z,callback-
gms,5115,Billing@Stat_Server.GA,GMS_Callback_VQ_OUT,GMS_Callback_VQ
```

Operation

```
POST /genesys/1/admin/callback/reportcancelled

HTTP/1.1
Connection: keep-alive
Content-Type: application/json
User-Agent: Mozilla/5.0 (Windows NT 6.1; WOW64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/
50.0.2661.102 Safari/537.36
Cookie: JSESSIONID=1ff4o2zwehsbx6fzdfwb66jsa
Authentication: Basic=....

{
    "callback_reason": "CANCELLED_BY_ADMIN",
    "exported_properties": ["_service_id,_desired_time"]
}
```

Response

```
desired_time,customer_number,exported_properties1,exported_properties2
2017-05-11T12:22:00+00:00,3329284556,exported_value1,exported_value2
2017-05-11T12:21:00+00:00,3329284576,exported_value1,exported_value2
2017-05-10T07:21:00+00:00,3329284577,exported_value1,exported_value2
```

Implement Preview and Disposition Scenarios

If you implement a custom agent desktop and wish to integrate the Preview and Disposition scenarios

to your Callback application, you need to configure Preview and Disposition options in your Callback service. After you do this, your Custom Agent Application will receive the following `UserEvent` events from Orchestration Server:

- `CallbackInvitationEvent`—The Callback invitation that contains the attached data for the preview. The invitation includes the list of actions that the agent can perform–accept, reject, or cancel. Your Agent application displays the actions and the attached data for the preview to the agent, then submits a Preview Response to your Callback service.

- `CallbackDispositionEvent`—The Callback disposition event that provides the URL to which you submit the disposition selected by the agent. Your Agent application then submits a Disposition Response to your Callback service through this URL.

Troubleshooting

This troubleshooting page lists basic information to help you fix some common issues on Genesys Callback. If you do not find the answer to your issue, please refer to the latest known issues from GMS Release Note.

A callback that was answered is blocked in queued or processing status

In this scenario, the update query for this callback has failed. This means that you should update the DFM files to ensure correct callback processing.

URS replies with errorresponse 404

URS could not find the subroutine that was started by the original `WaitForTarget` strategy with the callback start request.

```
13:25:56.750_R_I_ [19:10] routing interface request received:
urs/call/011b02924f228001/invoke, client=34(ORS_RTP_Node1_PR), ref=773
13:25:56.750_R_I_ [19:12] routing interface errorresponse '404' sent to
client=34(ORS_RTP_Node1_PR), ref=773
```

To solve this issue, compile both the strategy and subroutine as described here.