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

Genesys Engage cloud Administrator's Guide

Genesys Engage cloud Public

12/29/2021

Table of Contents

Administrator's Guide

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

Inbound

- Platform Administration
- Agent Setup
- Designer
- Genesys Softphone
- Genesys Widgets
- Gplus Adapter for Microsoft Dynamics

Digital

- Platform Administration
- Agent Setup
- Designer
- eServices Manager
- Genesys Widgets
- Genesys Co-browse

Outbound

- Platform Administration
- Genesys Softphone
- Outbound
- Gplus Adapter for Microsoft Dynamics 365

Self-service

- Platform Administration
- Agent Setup
- Designer
- IVR Administration
- Genesys Widgets

Workforce Optimization	Analytics
Workforce Management Recording	Historical Reporting Real-Time Reporting GVP Reporting

What is the role of an Administrator?

Administrators play several parts.

Architect

The Genesys Engage cloud Administrator is the architect and overseer of the Genesys Engage platform (the physical and non-physical resources that you use to run your contact center, including people, skills, and data). The platform is managed from the Configuration Database.

Agent Setup

Administrators create agents, configure accounts, and assign skills to agents. Administrators set up switches, voicemail, and interaction routing, and manage call recording. Administrators set up outbound calling campaigns.

Monitoring

Administrators also monitor the performance of the contact center using real-time and historical reporting and Workforce Management.

Solution notes

Are you looking for some quick answers or notes about these solutions? Here are answers to a few. If you have other questions, please send us an email at Techpubs.webadmin@genesys.com.

> **Important**
> Please note that Genesys does *not* provide customers with the ability to reach 911 or

other emergency services. Customers shall inform any individuals who may be present where the Genesys Engage cloud Services are used, or who use the Genesys Engage cloud Services, of the non-availability of 911 or other emergency dialing.

What are the PC requirements?

- **RAM:** 4 GB RAM (800 MB available for the Screen Recording Service, 400 MB available for WWE)
- **Processor:** Dual-core, 2GHz CPU
- **Hard drive:** 5 GB of available space (in total) for the Screen Recording Service installation and working space
  Note: Genesys Interaction Recording disk space requirements vary greatly based on the retention period of media assets on the desktop prior to download to the API. As a general guideline in assessing hard disk space, estimate the retention period of media assets and combine that configuration with sizing of files based on % of screen recording.
- **Supported Operating Systems:** Windows 7, 8, and 10, 32-bit or 64-bit
- **Citrix version:** Citrix Receiver 3.3
- **Supported browsers:**
  - Chrome 75+
  - Firefox 68+
  - Microsoft Internet Explorer 11 (exception: Callback is not supported in IE)
  - Microsoft Edge (Callback is supported in Microsoft EdgeHTML version 16.0 and up and on Chromium-based Edge)
    Note: With Microsoft Edge, Genesys recommends to add all sites involved in smart-failover and Identity Provider in trusted sites. For example: gwa-usw1.genesyscloud.com; api-usw1.genesyscloud.com.

There are no drivers, applets or any other downloads that need to be made to the PC, with the exception of the Genesys SIP Endpoint and Screen Recording Client for users who are processing interactions (calls, emails).

What are the bandwidth requirements?

Traffic	Bandwidth	Transport Via	When
Voice (SIP/RTP)	100 kbps G.711, (40 kbps G.729)	MPLS	Per call
LDAP	Negligible	MPLS	Per use logon
Web Service	Typically low	MPLS or Internet (HTTPS)	Varies based on the application
Desktop/CTI	16 kbps	MPLS or Internet (HTTPS)	Per call

Traffic	Bandwidth	Transport Via	When
Screen Recording	350 kbps two screens	MPLS or Internet (HTTPS)	Per recorded call/screen
Call playback	50 kbps	Internet (HTTPS)	User initiated
Report download	Varies	MPLS or Internet (HTTPS)	User initiated or can be scheduled (sFTP)

Where can I find an overview of Genesys Engage cloud services?

The Genesys Engage cloud User Guide provides information on how Genesys Engage cloud can work for, and support, your enterprise.

What hard and soft phones does Genesys Engage cloud support?

- Genesys: 420HD
- AudioCodes: 420HD, 430HD, 440HD
- Polycom: SPIP_321, SPIP_331, SPIP_335, SPIP_450, SPIP_550, SPIP_560, SPIP_650, SPIP_670, VVX_300, VVX_310, VVX_400, VVX_410, VVX_500, VVX_600, VVX_1500
- Yealink: SIP-T19P, SIP-T20P, SIP-T21P, SIP-T22P, SIP-T26P, SIP-T28P

Will I be able to test changes I make to my environment?

Genesys knows that contact centers require tricky routing of interactions. As such, you will have a chance to test changes you might make to aspects of your environment. Check out the details here.

About Genesys Engage cloud

Genesys Engage cloud architecture leverages the world-class SIP-based GVP/ Media Services products at the core of Genesys Engage while providing the full Genesys Engage suite of services in the cloud. This offer is customizable to meet the needs of enterprise customers, whether they need a large number of seats, heavy-duty premise system integration, or bespoke routing and reporting. Built on market-leading universal queuing and routing features, the full Genesys Engage suite will add the ability to support blended agents of any type, including blending inbound with outbound, voice with e-Services, or any other combination.

Top tasks for Genesys Engage cloud

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

Whether you're new to Genesys or would just like to know how to carry out one of our most-requested tasks, we hope this page has the information you need.

Can't find it? Ask us.

Get started

New to Genesys? Need a refresh? Learn the basics about using your all-in-one Genesys contact center platform.

- Getting started as an Administrator

People & Permissions

Manage the people in your organization.

- Set up a new agent
- Set up your teams and permissions
- Create and manage skills
- Manage users (Persons)

Business & Operations

Manage resources and settings for your business and operations.

- Make changes to a routing application (Designer content requires login)
- Manage your resources
- Make changes to Business Attribute values

Single Sign-On

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

Most Genesys Engage cloud applications allow a logged-in user to navigate across those applications without prompting for credentials again. It can also be configured to use SAML 2.0 for integrations with third-party identity providers such as Okta or Google. There are many advantages to enabling single sing-on (SSO) in Genesys Engage cloud—for example:

- Users need to remember only one password.
- User credentials are managed by a third-party identity provider.
- Users must have multi-factor authentication by a third-party identity provider for additional security.
- Users only need to log in once to gain access to Genesys Engage cloud applications that have SSO enabled and non-Genesys applications that use the same identity provider.

SSO support by application

View which Genesys Engage cloud applications support SSO.

Applications	Single Sign On Support	Notes
Agent Desktop	Yes	
Agent Setup	Yes	
Callback	Yes	
Cloud Data Download Service	Yes	
CX Contact	Yes	
Designer	Yes	
Developer Console	Yes	
Genesys CX Insights	Yes	Supported in version 9.0.013.0+. Contact your Genesys representative to enable.
Genesys Softphone	Yes	
Screen Recording	Yes	Supported in Agent Desktop version 9, but not with custom

Applications	Single Sign On Support	Notes
		desktops.
Real-Time Reporting (Pulse)	Yes	Supported in Real-Time Reporting version 9.
Gplus Adapter Salesforce	Yes	
Recording, Quality Management and Speech Analytics	Future Roadmap	
Workforce Management	Yes	Not supported for supervisor accounts for administrative activities.
Agent Scripting Administration	No	
Interactive Insights	No	
Outbound	No	
Platform Administration (GAX) *Includes plug-ins like eServices Manager and IVR Administration*	No	
WebRTC	Yes	

SSO Configuration - Genesys Engage cloud

To enable Single Sign-On for your environments, see the configuration help in Agent Setup.

> **Important**
> SSO can be configured for different groups and you can have multiple identity providers, as long as there is only one per region.

If you're planning to enable SSO, consider the following conventions for creating users:

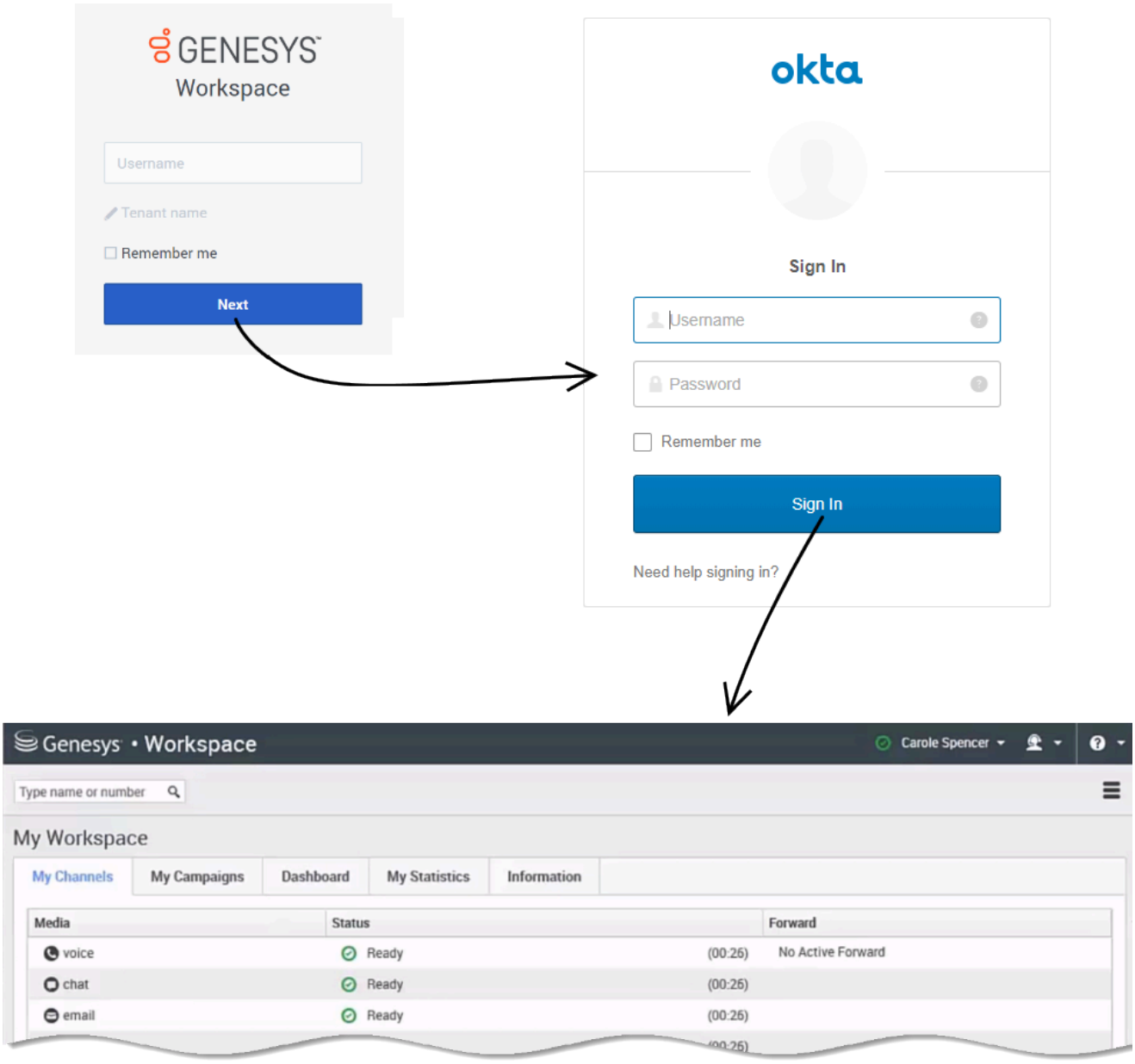
- The domain declared in the identity provider metadata should be part of the user name stored within Genesys, to create the most seamless experience. (Example: `john@mycompany.com`) Otherwise, users would need to enter a Tenant or enter the domain before their username. (Example: `mycompany\john`)
- The username provisioned within Genesys Engage cloud should match the username in the external identity provider.

SSO Configuration - Identity Provider

Genesys Engage cloud must be defined as an application within the identity provider to support the SSO integration.

Genesys Engage cloud supports the SAML 2.0 protocol as a standard interface to identity providers, and has successfully validated with popular IdPs, including Okta and Ping. Other identity providers can be supported provided they comply with SAML 2.0 and you validate the integration before using in production.

How does SSO work for users?



Let's look at the login process for Agent Desktop with SSO enabled and Okta configured as the third-party identity provider. **Note:** The login flow is the same for all supported identity providers.

First, click the Agent Desktop icon in Genesys Portal and enter your username. You must log in to the application even though you're already logged in to your workstation.

Click **Next**. Genesys redirects you to Okta where you're prompted to enter your username and password. Once you log in with Okta, you're redirected back to Agent Desktop and automatically logged in. Alternatively, if you are already logged in with Okta when you click **Next**, Genesys skips the Okta login and automatically logs you in to Agent Desktop.

Now that you're authenticated with the identity provider, you can choose any SSO-enabled application from Genesys Portal and you'll be automatically logged in without entering your credentials.

If you happen to close all browser tabs without logging out of the applications, you will remain logged in for five minutes. If a second window or browser is opened after five minutes, to either the same application or any other SSO-enabled application, you will once again be prompted for your credentials.

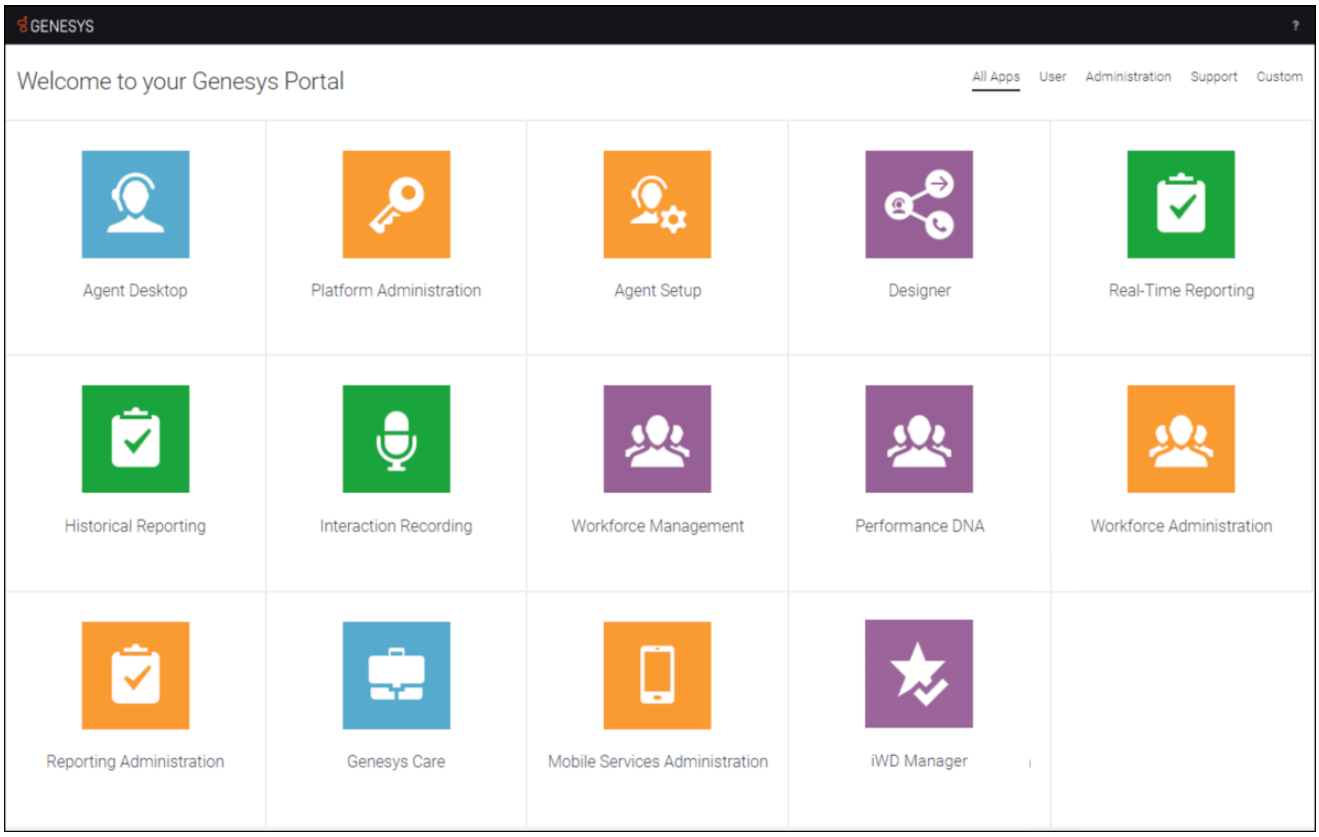
Cloud basics for administrators

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

Welcome to Genesys Engage cloud. This Genesys offering enables you to get started today with an all-in-one Genesys contact-center platform.

Access your applications

Genesys Portal is your access point to all Genesys Engage cloud applications.



All of your Genesys applications can be found on the **All Apps** screen. These applications are also divided into three categories, which you can select by clicking the corresponding category links at the top right of the Portal.

To customize categories for your contact center, contact Genesys Customer Care.

The out-of-the-box categories are:

User – User applications can be used by contact-center agents and supervisors, such as Agent Desktop, Workforce Management, and Reporting.

Administration – Administration applications can be used by contact-center administrators and supervisors to configure users and applications based on their roles.

Support – Support applications are used by authorized contact-center administrators to submit requests with Genesys Customer Care, and to view the status and schedule of service for Genesys applications.

Documentation and **eLearning** are accessed via the Help (?) menu, located in the top-right corner of Genesys Portal.

Application	Application Description
Agent Desktop	Agent Desktop provides agents and knowledge workers with non-intrusive access to the information, processes, and applications they need to perform their jobs more efficiently.
Agent Scripting Administration	Agent Scripting Administration is a scripting tool that is used to prompt agents throughout the call-handling process with customers.
Cloud iWD	Cloud intelligent Workload Distribution (iWD) allows you to dynamically prioritize the distribution of workitems from external source systems to the employee best suited to handle them.
Designer	Genesys Designer helps you to design assisted service or Routing applications, as well as self-service or Interactive Voice Response (IVR) applications.
Genesys Customer Care	Genesys Customer Care is a portal for Genesys customers to locate resources, tools and information that will aid in resolving your issues, including the ability to create and track tickets based on Salesforce.
IVR Administration	IVR Administration assists your business with providing cost-effective customer interactions 24/7 for voice, video, and web-based interactions.
Outbound	Outbound Contact is an automated system for creating, modifying, running, and reporting on outbound campaigns for proactive customer contact. Outbound Contact Server (OCS) provides automated dialing and call-progress detection, so that an agent is required only when a customer is connected. OCS also intelligently uses customer data to ensure that campaigns are contacting the right customers, not just a large number of customers.
Platform Administration	Platform Administration helps you to manage your user accounts and configure settings to maintain your contact center. It also provides a dashboard and widgets to see contact-center activities in real-time.
Recording	Recording provides recording control via a host of integrations across the suite. The recorded segments are linked to the contact's account history, and include information about date and time, duration, and the parties involved for each segment.
Reporting	Reporting assists contact-center managers in: • Assessing the day-to-day operations of their contact center resources for the routing and handling of interactions. • Better tuning resources to increase utilization

Application	Application Description
	and efficiency. • Benchmarking key performance indicators of quality and service. • Identifying corrective actions to help reduce costs and increase service
Reporting Administration	Reporting Administration is used to provision new users for Reporting.
Workforce Management	Workforce Management (WFM) provides tools to enable contact-center managers to better manage their workforce. WFM enables managers to create accurate staffing plans that take into account projected contact volumes and average handle times, as well as the various skills and skill levels of the agent population. Contact center managers can achieve these goals, by using WFM's advanced forecasting, scheduling, and real-time adherence capabilities.

Role Access

Access to Genesys Portal applications is provided based on the following user roles:

Application/Role	Agent	Supervisor	CC Manager	Administrator
Agent Desktop		*	*	*
Agent Scripting Administration				
Agent Setup		++	+++	+
Callback		**	**	
Cloud iWD				
Designer		***	***	
Genesys Customer Care				
IVR Administration				
Outbound				
Platform Administration	****			
Recording				
Reporting				
Reporting Administration				
Training				
Workforce	*****	*****	*****	*****

Application/Role	Agent	Supervisor	CC Manager	Administrator
Management				

Important

* - Separate Agent accounts must be created for Supervisors, Managers, and Administrators. See How to create an Agent account for Supervisors, Managers, and Administrators with dual role in Platform Administration.

** - Supervisors and Managers have read-only access to callbacks and callback configuration.

*** - Supervisors and Managers only have access to modify Designer parameters.

**** - Agents only have the ability to change their own password in Platform Administration.

***** - Agents work on a different UI from Supervisors, Managers, and Administrators.

+ - Read and write access.

++ - Limited read and write access.

+++ - This role is not defined within Agent Setup.

Account Types

The following types of accounts are provisioned in Genesys Engage cloud:

Account Type	Provisioned By	Provisioning Tool	LDAP enabled*	Note
Agent Scripting Administration Account	Genesys	Agent Scripting	No	
Customer Care Account	Genesys	Genesys Salesforce	No	
IVR Administration Account	Customer Administrator	IVR Administration	Yes	One per IVR Administration region
Outbound Account	Genesys	Genesys Engage	No	
Platform Account	Customer Administrator	Platform Administration	Yes	
Reporting Account	Customer Administrator	Reporting Administration	No	

Important

- Certain types of accounts – as specified in the table – can be configured to use authentication through customer LDAP-compatible directory service.

- It is recommended that all accounts for a particular user be created with the same user name and password. The table above outlines the different authentication and authorization systems where separate user accounts need to be created.
- The recommended naming convention for user accounts is the following: <login_name/first_last_name>@<customer_company>, for example: john.bull@acme.com. It is important to include <customer_company> in the user name as unique name domain identifier.
- For supervisor/administrator users who need access to the Agent Desktop or Agent Workforce Management interface, the suggested naming convention is the following: <login_name/first_last_name>@<customer_company>.agent

Password Rules

User accounts are required to meet the following password rules:

- **Password Length** - Must include a minimum of 8 characters.
- **Password Complexity** - Must include lowercase, uppercase and a number.
- **Password Lockout** - After 5 failed login attempts, the user account is locked for 30 minutes.
- **Password Changes** - Must be changed every 90 days.
- **Password Re-use** - Cannot re-use any of their last 5 passwords.

Managing User Passwords

A user may be configured to set a new password the first they log in, or after a system administrator has reset their password. In this case, the **Change Password** dialog box appears.

To change the password, the user must complete the following steps:

1. Enter a new password in the **New Password** field.
2. Enter the same password in the **Confirm Password** field.
3. Click **Ok**.

Changing a Password

If configured, a user can change their password at any time, using the following procedure:

1. Log in to the user account in Platform Administration. The user’s first name appears in the top header bar of the window.
2. Select the first name.
3. Click **Change Password**.
4. In the **Change Password** dialog box, enter the current password, enter a new password and confirm the new password.

5. Click **Save**.

Using Accounts in Applications

The table below lists correlation between account types and Genesys Engage cloud applications.

Application/Account	Account	Notes on Provisioning
Agent Desktop	Platform Account	
Agent Scripting Administration	Agent Scripting Account	Account should be enabled by Genesys for use in this interface
Cloud iWD	Platform Account	
Designer	Platform Account	
Genesys Customer Care	Customer Care Account	
IVR Administration	IVR Administration Account	
Outbound	Genesys Engage Account	Account should be enabled by Genesys for use in this interface
Platform Administration	Platform Account	
Recording	Platform Account	
Reporting	Reporting Account	
Reporting Administration	Reporting Account	
Training	No account needed	
Workforce Management	Platform Account	Account should be assigned to security group in WFM by customer administrator

Call control in Agent Desktop

In Genesys Engage cloud, agents control calls using the Agent Desktop web interface. Agent Desktop supports call control in the majority of call scenarios.

Hold/retrieve, conference, call transfer, call termination are fully supported for all types of agent phones.

Engaging another agent, as well as transferring a call to an agent, agent group, or queue should rely on Favorites in the desktop communicator. Only a few use cases require an agent to call an external party by means of dialing a phone number.

In the scenario where an agent needs to interact with an IVR via a telephone keypad, the agent can use Agent Desktop to send dual-tone multi-frequency (DTMF) signals to the IVR.

The Call Answer (Call Accept) desktop feature is disabled in environments where telephony infrastructure between Genesys and the agent's phone, or the phone itself, does not support the functionality.

This feature is not supported if:

1. There is a PSTN between Genesys and the phone.

2. The phone is behind an Avaya switch.

In these environments, agents need to answer calls manually or auto-answer needs to be setup on their phones.

Begin provisioning your contact center

Click here to begin provisioning your contact center.

Provisioning your Contact Center

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

The Genesys Engage cloud platform enables you to create basic virtual contact-center infrastructure and begin using it within minutes. Upon completion of the steps below, you will be able to call your virtual contact center, accept and control calls in Desktop as an agent, monitor results of the calls in Real-Time and Historical Reporting as a supervisor, and also play back recordings of the calls in the Recording interface.

Provision your Contact-Center Resources

Before you start provisioning your contact-center resources, Genesys will supply you with a test IVR application and a test public phone number that can be called to reach the application. The application will have an initial greeting and a simple IVR menu with options resulting in the routing of calls to available agents with specific Skills associated with specific Virtual Queues. The Skills and the Virtual Queues may be pre-created for you by Genesys, but you can also create your own, as described below, and ask Genesys to update the IVR application logic.

Create a Skill in Platform Administration

In the top menu, select **Configuration** to access the **Configuration Manager** page. While hovering over the **Accounts** area, select **Skills**.

1. Navigate to **Skills** > **Company Folder**.
2. Click **New (+)**.
3. Enter a Skill name in the **Name** field.
4. Click **Save**.

Create a Virtual Queue in Platform Administration

In the top menu, select **Configuration** to access the **Configuration Manager** page. While hovering over the **Switching** area, select **DNs**.

1. Navigate to **Switches** > **Target Switch** > **DNs** > **Company Folder** > **Virtual Queues**.

2. Click **New (+)**.
3. Enter a Virtual Queue name in the **Number** field.
4. Select **Virtual Queue** in the **Type** field.
5. Repeat the Virtual Queue name in the **Alias** field.
6. Click **Save**.

Provision your Contact-Center Agents

The next step is to provision a contact-center agent and the associated resources in Platform Administration.

Creating an agent consists of the following actions:

1. Create an agent account.
2. Create an agent's Place and associate it with an extension.
3. Associate an agent with Skills.
4. Create an agent's Login.

Create an Agent Account

In the top menu, select **Configuration** to access the **Configuration Manager** page. While hovering over the **Accounts** area, select **Persons**.

1. Click **New (+)**.
2. Enter the agent's user name in the **User Name** field – normally this is the agent's email address or *<login_name>@<company_domain>*.
3. Make sure that the **Agent** checkbox is checked.
4. Enter the agent's first and last names in the corresponding fields.
5. Optionally, enter the agent's email address in the **E-mail Address** field.
6. Enter the agent's corporate ID in the **Employee ID** field. The Employee ID may be the user name or another unique corporate ID.
7. Enter the agent's password and confirm it.
8. Click **Apply**.

Create an Agent's Place and Associate it with an Extension

> **Important**
> Steps 6 through 15 are not required for Chat and/or E-mail only customers.

In the same **Persons** window from the previous step, perform the following actions:

1. Click the **Browse** button located in the **Default Place** field.
2. Click the **Show Hierarchy** link.
3. Navigate to the folder in which you would like to create the **Place**.
4. Click **New (+)**.
5. Enter the Agent's user name attribute value in the **Name** field.
6. Click on the **DNs** tab while still remaining on this **Place** screen.
7. Click **Add**.
8. Click the **Show Hierarchy** link.
9. Navigate to **Switches** > **Target Switch** > **DNs** > **Company Folder** > **Extensions**.
10. Click **New (+)**.
11. Enter the Agent's phone number attribute value in the **Number** field. The phone number in the simplest case can be a public phone number at which an agent can be reached (starting with "+" and the country code) or it can be an extension in the corporate switching system.
12. Select **Extension** in the **Type** field.
13. Enter the Agent's user name in the **Alias** field.
14. Click **Save**.
15. Select the newly created DN extension in the list.
16. Click **Save**.
17. Select the newly created Place from the list.
18. Click **Apply**.

Associate an Agent with Skills

In the same **Persons** window from the previous steps, perform the following actions:

1. Click the **Skills** tab on the left navigation pane.
2. Click **Add**.
3. Click the **Browse** button in the **Skill** field.
4. Navigate to your company folder.

5. Select a skill from the list.
6. Assign the appropriate skill level.
7. Click **OK**.
8. Click **Apply**.

Create an Agent's Login

> **Important**
> Agent Login is not required for Chat and/or E-mail only customers.

In the same **Persons** window from the previous steps, perform the following actions:

1. Click the **Agent Logins** tab on the left navigation pane.
2. Click **Add**.
3. Click the **Browse** button in the **Agent Login** field.
4. Navigate to **Switches** > **Target Switch** > **Agent Logins** > **Company Folder**.
5. Click **New (+)**.
6. Enter the Agent's phone number attribute value in the **Code** field.
7. Click **Save**.
8. Select the newly created Agent Login from the list.
9. Click **OK**.
10. Click **Apply** or **Save**.

Provisioning Agent Groups and Non-Agent Accounts

While it is not strictly required for basic contact-center provisioning, you may also provision additional types of resources as described below.

Create a Virtual Agent Group in Platform Administration

Virtual Agent Groups allow you to group agents dynamically based on their skills assignment. You can conveniently select and use them in Historical and Real-Time Reporting.

In the top menu, select **Configuration** to access the **Configuration Manager** page. While hovering over the **Accounts** area, select **Agent Groups**.

1. Click **New**.

2. Enter the Agent Group name in the **Name** field.
3. Click the **Options** tab on the left navigation pane.
4. Click **Add**.
5. In the pop-up dialog box, enter the following:
    a. **Section**: `Virtual`
    b. **Key**: `Script`
    c. **Value**: `SkillExists(“skill_name”)`
    d. Click **OK**.
5. Click **Save**.

Create Administrator/Manager/Supervisor Accounts

Your Administrators, Managers, and Supervisors may need more than one account in the system, depending on their role or needs. Please refer to the reference tables in the Role Access section.

The topics below describe how to add different types of accounts.

Create a Platform Account

Log in to the **Platform Administration** interface.

In the top menu, select **Configuration** to access the **Configuration Manager** page. While hovering over the **Accounts** area, select **Persons**.

1. Click **New (+)**.
2. Enter a user name in the **User Name** field – normally this is the user's email address or *<login_name>@<company_domain>*.
3. Make sure that the **Agent** checkbox is NOT checked.
4. Enter the user’s first and last names in the corresponding fields.
5. Enter the user’s email address in the **E-mail Address** field.
6. Enter the user’s corporate ID in the **Employee ID** field.
7. Enter a password and confirm it.
8. Click **Apply**.
9. Click on the **Member Of** tab on the left navigation pane.
10. Click **Add**.
11. Click the **Show Hierarchy** link.
12. Navigate to your company folder.
13. Select an appropriate Access Group. Normally, it will be *“<company> Administrators”*, *“<company> Supervisors”* or *“<company> Managers”*.
14. Click **Save**.

Create an IVR Administration Account

Log in to the **IVR Administration** interface (in each geographical region where your contact center is provisioned).

Repeat all of the steps described in the previous topic (Create a Platform Account).

Create a Reporting Account

> **Important**
> Use either Mozilla FireFox or Microsoft Internet Explorer to create a Reporting account.

Log in to the **Reporting Administration** interface. Leave the System and Authentication fields with the default values, and enter your Reporting user name and password.

1. Navigate to **Group Hierarchy** > **Interactive Insights Report Users** > **<company> Users**.
2. Click the **Create New User** icon in the toolbar.
3. Enter the user name (same as in Platform Account) in the **Account Name** field.
4. Enter the user's first and last names in the corresponding fields.
5. Enter a password in the **Password** field (it cannot equal the user name).
6. Click **Save** and **Close**.

Create an Agent Account for Administrator/Manager/Supervisor in Platform Administration

If your Administrator, Manager, or Supervisor also needs access to agent interfaces (such as **Agent Desktop** or the agent part of **Workforce Management**), you will need to create a separate agent account.

Separate Agent accounts need to be created for an Administrator, Manager, or Supervisor who plays a dual role.

1. Follow the steps described in How to Create an Agent in Platform Administration, but make the following modifications:
    a. Wherever a person’s email is used for a field value (**User Name**, **Employee ID**, **Place Name**, **DN Alias**), add the `.agent` suffix to it, such as *john.smith@company.com.agent*.

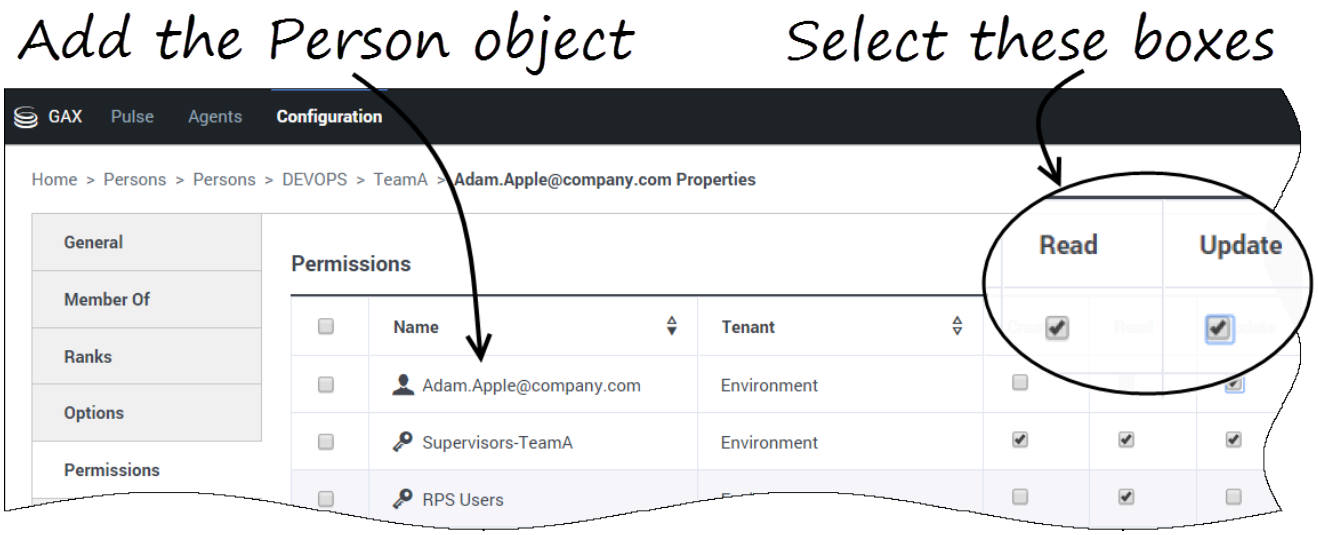
If your Administrator, Manager, or Supervisor needs to monitor Agents, follow these steps:

1. Click the **Options** tab on the left navigation pane.
2. Click **Add**.
3. In the dialog box, enter the following:

a. **Section**: htcc
b. **Key**: roles
c. **Value**: Supervisor,Agent
d. Click **Save**.

5. Click **Add**.
6. In the dialog box, enter the following:
   a. **Section**: interaction-workspace
   b. **Key**: privilege.teamlead.can-use
   c. **Value**: true
4. Click **Save**.

Allowing users to customize Pulse dashboards



If your users are working with Pulse for real-time reports, they might want to save any changes they make to their dashboards. You can enable this by granting them the proper permissions.

On the **Configuration Manager** page, under **Accounts**, go to **Persons** and find the User that you want to edit. On the **Permissions** tab, **Add** the Person object for this user. Once the Person object appears in the Permissions list, add both **Read** and **Update** permissions for this user.

Don't forget to **Save** the permission before closing the window.

Do this for each user you want to enable.

Begin Using your Contact Center

Having provisioned your contact center, click here to begin using your virtual contact center.

Using your contact center

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

Once you have configured the necessary contact-center resources, you can start using applications to make or control calls, and view the results of your calls in reports.

Using Agent Desktop to Accept, Control, and Make Calls

1. Log in to **Agent Desktop** with an agent account that you created while Provisioning your Contact Center. Make sure the agent has been assigned the Skills to be targeted by your IVR application. Also, make sure the agent is in possession of a phone set (corporate SIP Phone or PSTN phone, or even a personal cell phone) and that phone number associated with this set is configured as this agent's extension.
2. Make this agent ready in **Agent Desktop** for Voice (this is a default state).
3. Call the number associated with the test application (provided by Genesys), and navigate to the IVR Menu option intended to initiate routing to a skill target possessed by your agent.
4. Your agent phone will ring.
5. Accept the call on your phone set.
    a. Click **Accept** in the pop-up window if you are using a SIP Phone.
    b. Answer the call through your device if you are using a PSTN phone or a personal cell phone.
3. Use the navigation buttons to put the call on hold, retrieve the call, transfer the call to another phone number, or simply terminate the call.
4. You can also initiate an outbound call by entering the phone number in the phone input field.
5. Once the call is complete, the call statistics are updated on the Dashboard.

Use Real-Time Reporting to View Intra-Day Statistics and Reports

Real-Time Reporting can be viewed in **Platform Administration** by navigating to **Pulse** in the top menu.

Each user is presented with an individual dashboard based on a default dashboard provisioned by Genesys Administrator. At any moment it is possible to reset the individual dashboard to the default view.

Within a dashboard, users are able to add new Widgets.

Add a new Widget

1. Click **New (+)** within the dashboard.
2. Choose a Widget Template in the new pop-up window and click **Next**.
3. Choose a target configuration object to monitor (Agent, Group, Queue) and click **Next**.
4. Choose a list of statistics to view and click **Next**.
5. Customize the display options.
6. Click **Finish**.

All saved widgets are saved in the individual dashboards.

To obtain more details about Real-Time Reporting, refer to the Pulse documentation page.

Use Historical Reporting to view Contact-Center Analytics

To access **Historical Reporting**, log in to **Reporting** from Genesys Portal with an Administrator, Manager, or Supervisor account.

1. Click **Document List** in the top menu.
2. Navigate to **Public Folders** > **Interactive Insights** > **8.1.1**.
3. Select the desired Reports Group (such as Agents or Queues).
4. Select the reports you want to view.
5. When the Report opens, click **Refresh All** to invoke the Report Parameters.
6. Select one of the pre-defined date ranges (such as *Today*).
7. Apply any additional filter criteria.
8. Click **Run Query**.
9. Review the results in either the **Summary** or **Main** pages (navigation is available in the left pane).

For more information about Historical Reporting, refer to the Historical Reporting documentation page.

Or you can read the documentation available in the application by following these steps:

1. Click **Document List** from the top menu.
2. Navigate to **Public Folders** > **Interactive Insights** > **8.1.1** > **Documentation**.
3. View the *Universe Guide* or the *User's Guide*.

How to Set Up Your Teams

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

You can organize the personnel in your contact center into groups, called *Teams*, to better reflect your organizational structure and to meet your particular needs. You then set up Access Groups within the Teams to control data access and role privileges within your organization.

[+] Why use Access Groups to control what information users can access?

If you only have a few users in your contact center, you might want to give each of those users specific permissions and role privileges to access data and other call center resources (called *objects*) that they need to work with. This protects data and resources from possible misuse (malicious or otherwise) by users who do not need access. However, as your contact center grows and you hire more users, keeping track of who has what permissions can become an administrative nightmare.

Instead, Genesys recommends that you use Access Groups. Access Groups are virtual groups of users that have the same access permissions and role privileges. But the permissions and privileges are assigned to the Access Group, not to the users themselves. Users can belong to more than one Access Group, so if you need to add or change the permissions and privileges assigned to a user, you only have to add or move the User to or between appropriate Access Groups. Instead of keeping track of who has what permissions, you only need to know the Access Groups of which the user is a member.

The minimum Access Groups that Genesys recommends in a Team are a Managers Access Group and a Supervisors Access Group. By default, these groups should have full access to agent-related objects (Places, Agent Logins, and DN Extensions), and Read access to most other objects, including users. You can customize other permissions and privileges of each of these Access Groups as required. For example, you may want to enable only one or both of these Groups to view Real Time Reporting

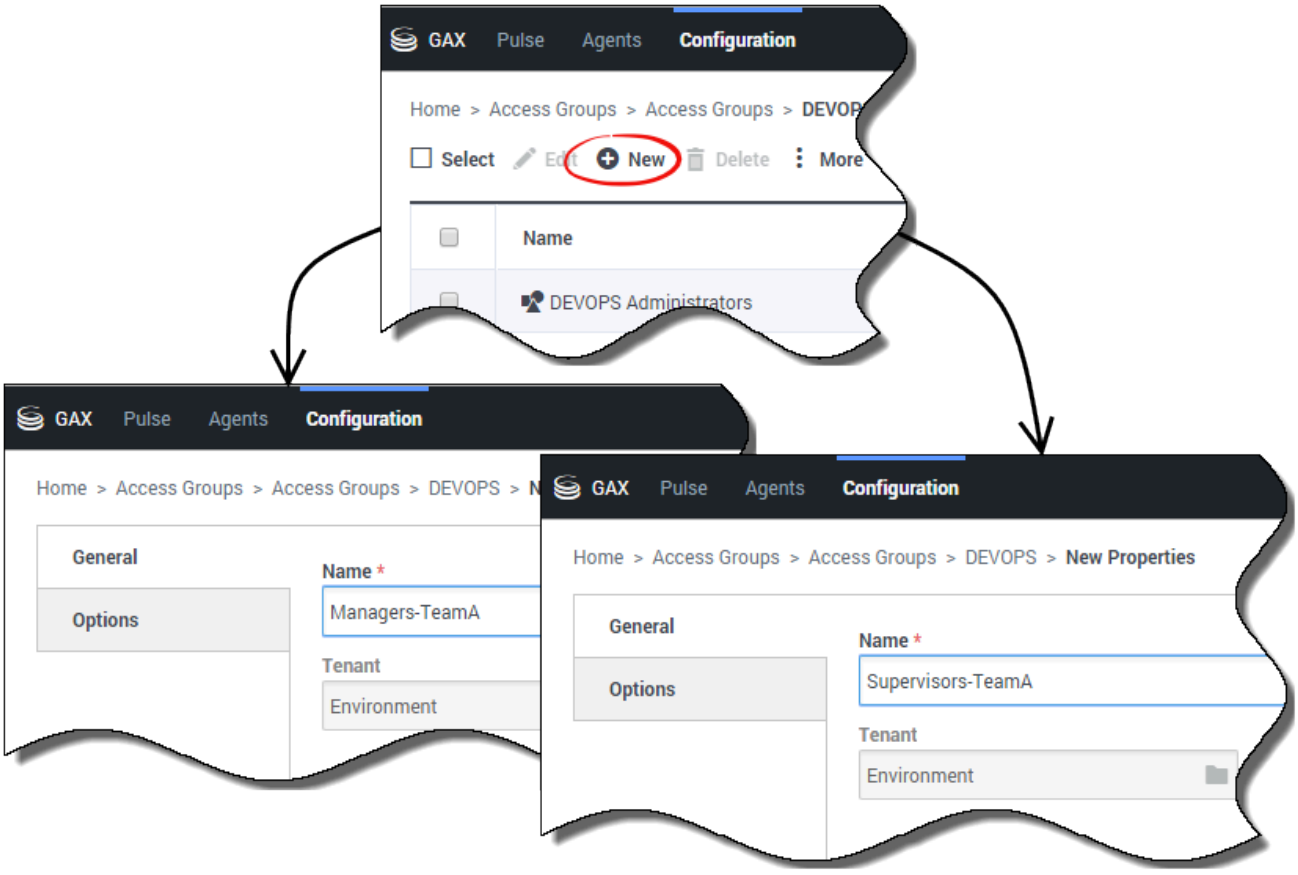
information.

The Basics

From the Access Groups and Users pages in Platform Administration you are going to:

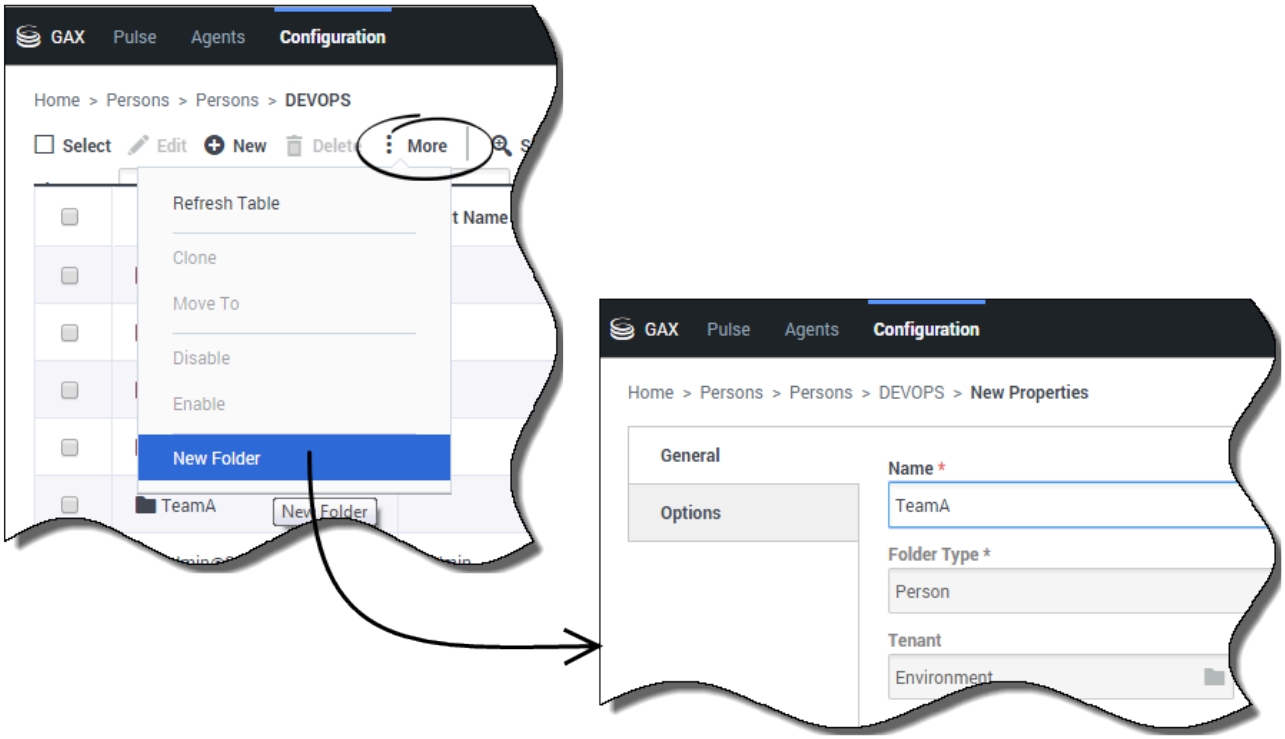
1. Create two Access Groups, one for Managers and one for Supervisors.
2. Create the new Team.
3. Add Managers and Supervisors to the Team, and to the Access Groups as appropriate.
4. Set up permissions to the Team folder for each Access Group.

Create Access Groups for Managers and Supervisors



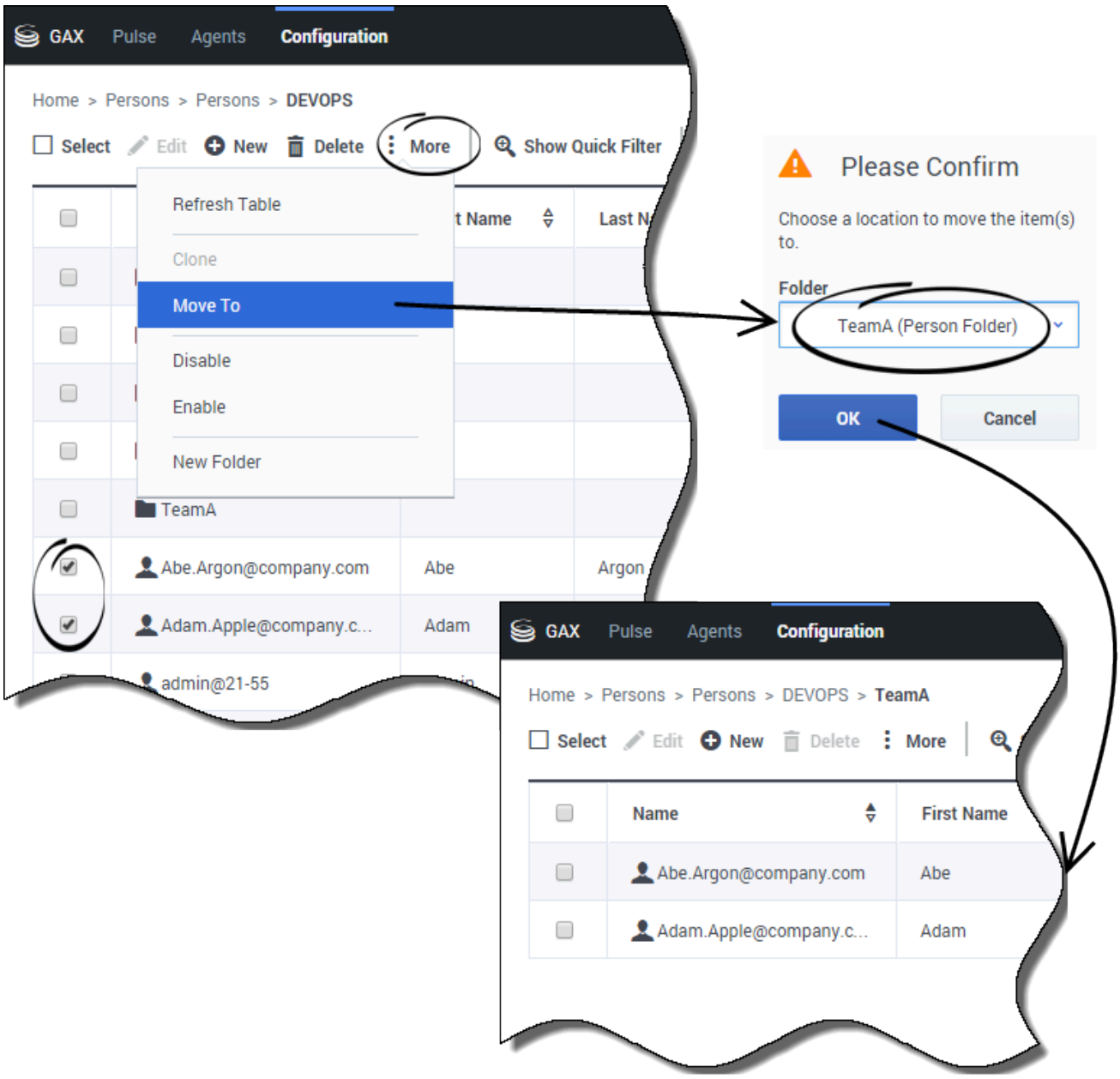
From the list of Access Groups in Configuration Manager, navigate to the folder where you are going to create the Managers Access Group and click **New**. Then do the same to create the Supervisors Access Group.

Create the Team



From the list of Persons in Configuration Manager, navigate to the folder in which you want to create the Team and use the **More** menu to create a new folder. This folder represents the Team, and will contain Team members, both individual users and groups of users.

Add Managers and Supervisors to the Team ...

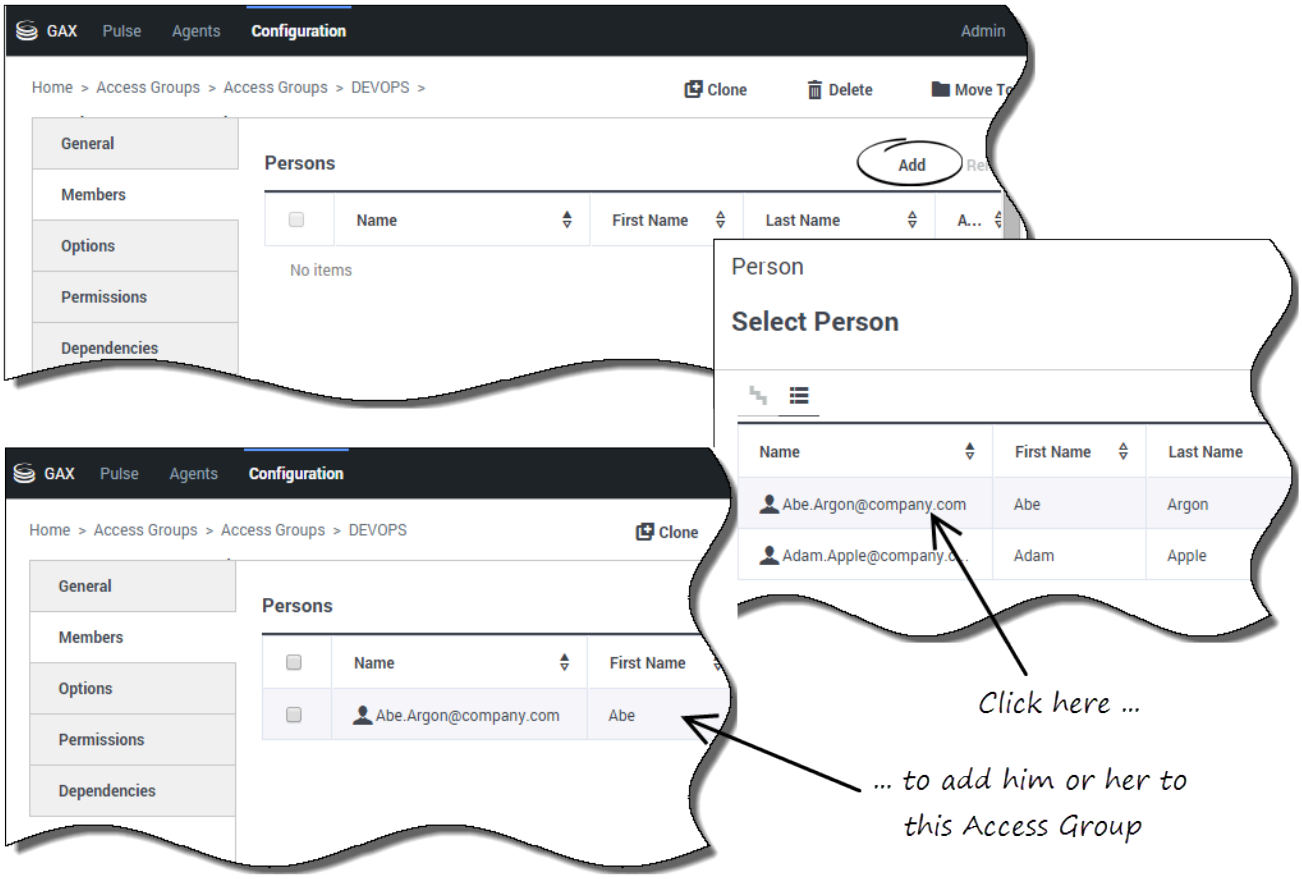


Important

Before you start this step, make sure that you have created user objects that represent your Managers and Supervisors. See Provisioning your Contact Center for more information.

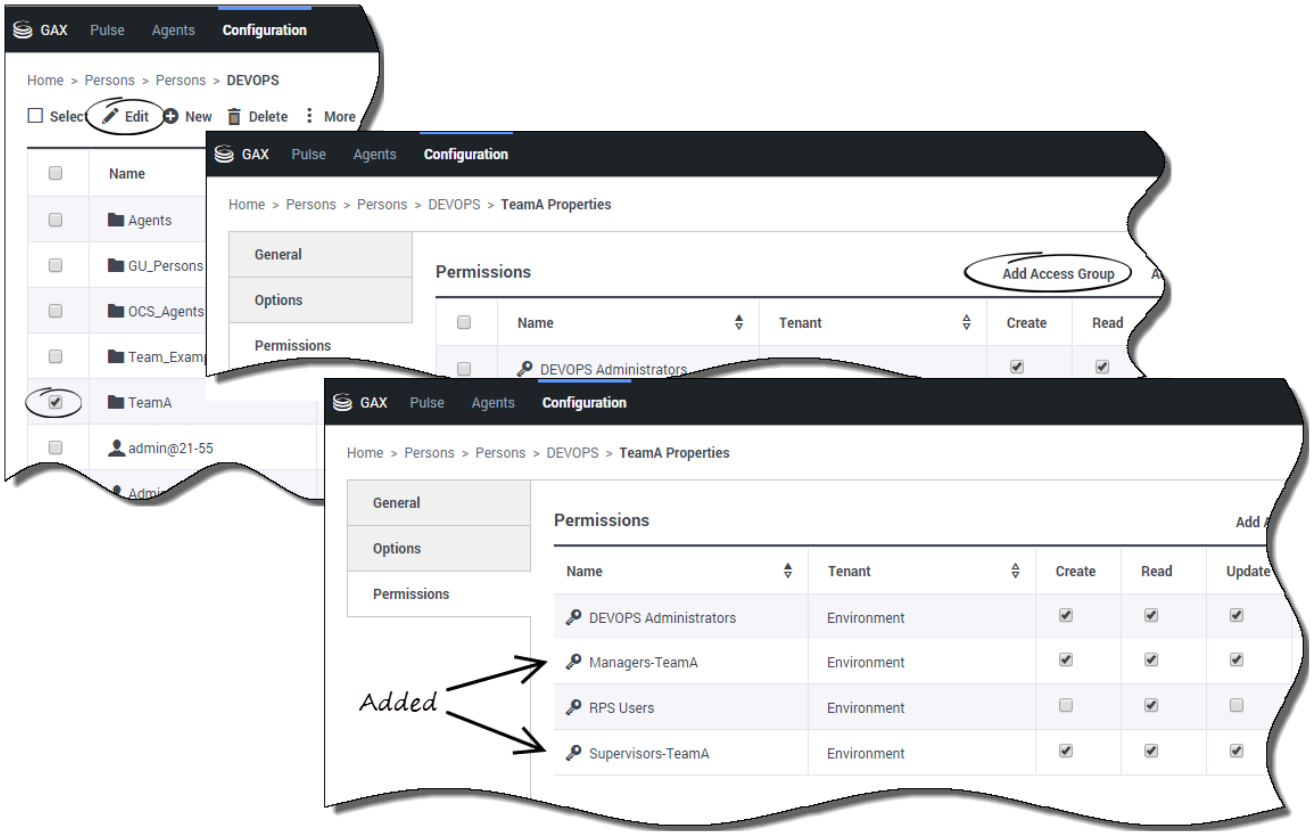
In the Persons list of the folder that contains the Managers and/or Supervisors that you want to add to this Team, use the **More** menu to move the selected Manager to the Team folder that you just created. If the Managers and Supervisors are currently in different folders, simply repeat this action in each folder. Within a given folder, you can select and move the Managers and/or Supervisors individually, or select them all and then move them to the Team.

… and to the Access Groups



From the **Members** tab in the properties of the Managers Access Group, click **Add** and navigate to the Team folder, and select the Managers to be added to the Group. A reverse entry will be made automatically to add the Manager Access Group to the **Members Of** tab of each Manager. Do these steps again to add Supervisors to the Supervisors Access Group.

Set permissions for Members of the Access Group



For your Team, you want the Managers and Supervisors Access Groups to be able to make changes to the Team, and you allow this on the **Permissions tab** of the Team folder. If the Managers and Supervisors Access Groups are not listed on this tab, click **Add Access Group** to add them. Then set the required permissions.

How To Work with Skills

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

You can assign skills to your agents so that different interactions get routed to the right person for the right kinds of tasks. For example, you can set up skills for a particular language – so that calls requiring an *English* agent are routed to an agent assigned with that skill. You can assign skills based on different business processes or special knowledge that your agents may have, whatever makes sense for your company.

How Genesys skill-based routing improves customer experience and contact center flow

When a customer contacts your organization, they want to connect with the person who is best able to help them. Genesys skills-based routing will greatly improve the efficiency and quality of your contact center.

Genesys skills-based routing will connect the customer to the agent who can best help them

By assigning skills to your agents, an administrator can ensure that customers will reach those agents who have the knowledge and experience related to their issue. Agents can have multiple skills, and the routing system can find the best available agent for the task.

How it works

Administrators assign skills to their agents, along with a level (or ranking) for each skill. The customer provides information related to their call (for example, by using keypad menu selections, submitting an online form, or describing their issue in a chat window). The routing strategy then searches for an available agent who is the best match for the skills required and connects them to the customer.

The benefits of using skills-based routing

- For customers, this vastly improves their customer service experience. They will connect with someone who can best help them with their issue.
- For agents, they get to utilize their skills and qualifications in a way that best serves the customers.

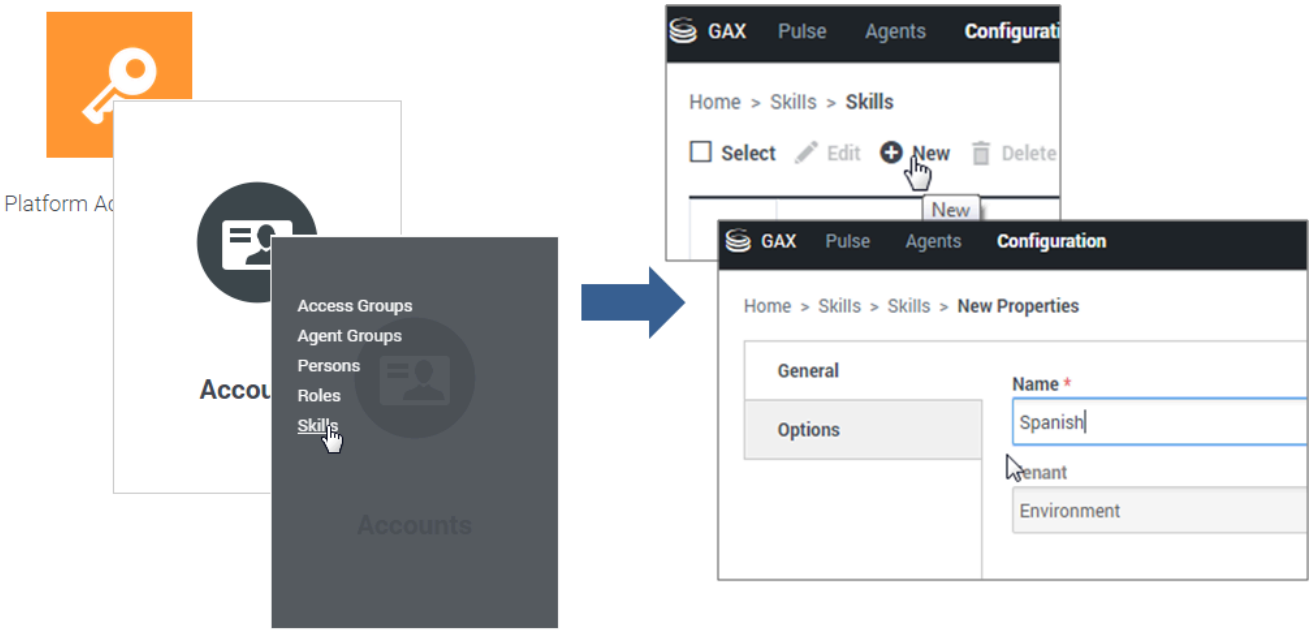
- For administrators, they can leverage agent skills to improve reporting statistics and manage agents according to the needs of the organization.

The Basics

Let's say you want to create a new skill based on *Spanish* language ability. The typical flow for setting that up will go like this:

1. Create the skill.
2. Assign the skill to the right agents.
3. Apply the skill to your routing strategy.

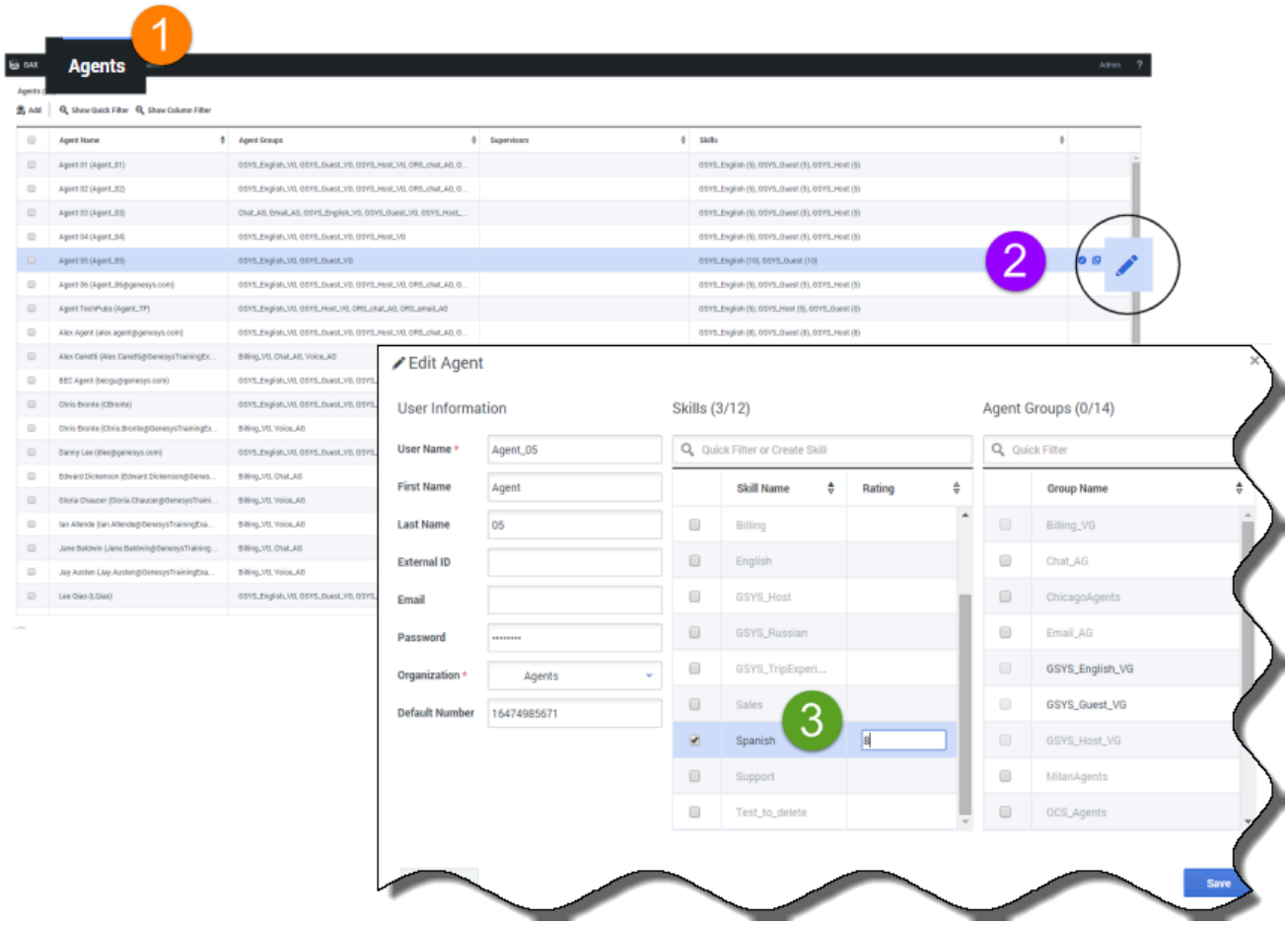
Create a new skill



From **Genesys Portal**, click the **Platform Administration** tile and go to **Skills**.

- Click **New +** and give the skill a name.
- In this example, we have created a new skill called *Spanish*.

Assign the skill to an Agent



Go to the **Agents** tab and highlight the agent you want to assign a skill to.

- Click  to edit the agent's settings.
- Select the desired skill from the **Skills** list. In this example, we've selected the *Spanish* skill.
- Add a numeric rating to indicate how proficient the agent is with that particular skill (this helps the routing strategy determine which agent should receive the call). We've given our agent a rating of 8 to indicate that they are relatively strong in Spanish.
- You can continue to apply more skills to the agent. When you are done, save your changes.

Use the skill in your routing strategy

You can now design your routing strategies to make use of these defined skills.

Using the Genesys Designer palette, you can combine various *blocks* to create a routing strategy. Some blocks that relate to agent skills routing are:

- Route Call
- Menu

To show how it works, we will edit an existing routing application to direct certain calls to our Spanish-speaking agent.

> **Important**
>
> Agent routing blocks are only used in the **Assisted Service** phase of the application structure.

From the Genesys Portal page, log on to **Designer** and select the application to modify.

To add a routing block, simply drag and drop the **Route Call** block from the Palette to the **Assisted Service** block under the Application Flow.

This opens the **Properties** section for the Route Call block, where you can select skills and other routing properties.

Add the *Spanish* skill by selecting it from the list. You can then add other skills, or select any of the other options for routing calls.

How to set up your agents

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

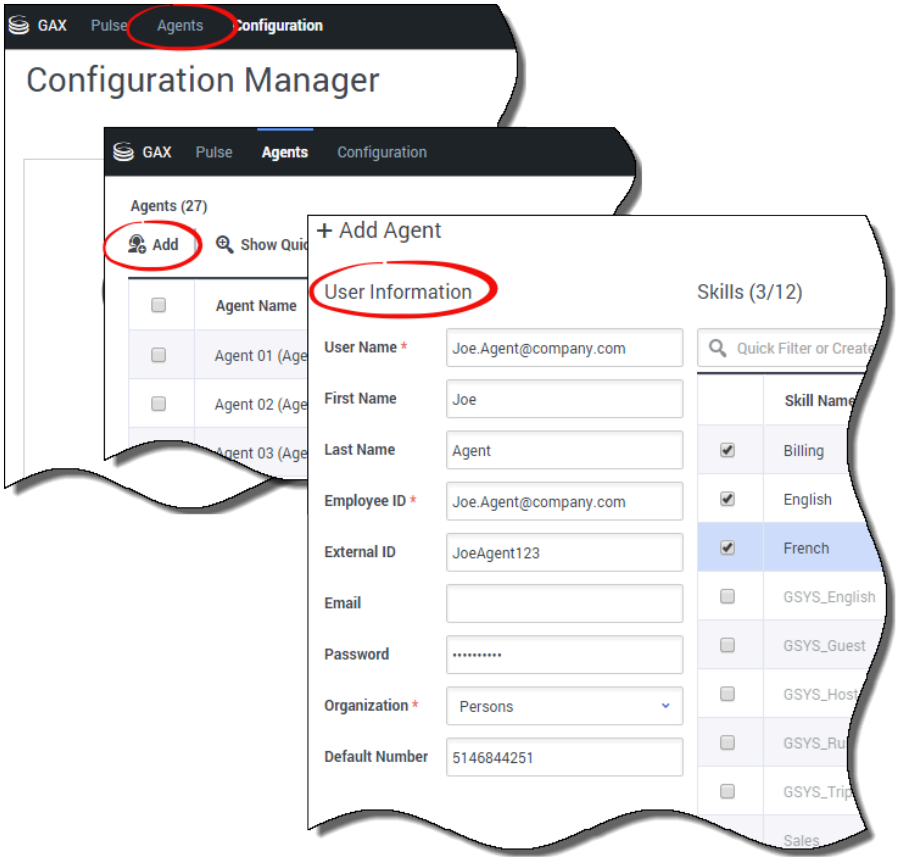
Before your contact-center agents can begin interacting with customers, you must set up their account.

The Basics

From the Agents screen in Platform Administration you are going to:

1. Create the Agent account.
2. Assign one or more Skills to the Agent.
3. Assign the new Agent to an Agent Group.

Create an Agent Account

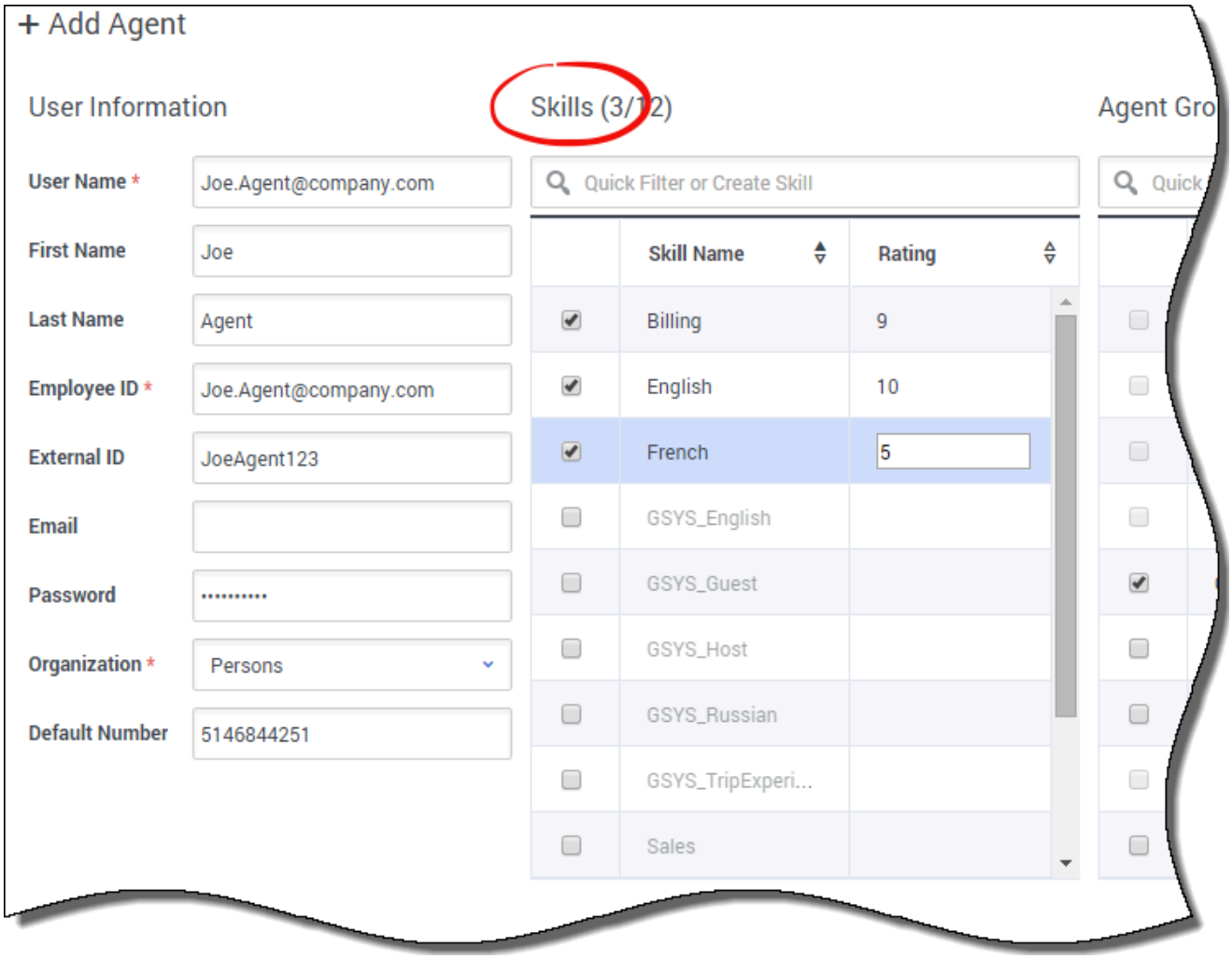


Select **Add** and enter the agent's personal information. Mandatory fields are indicated by a red star (*).

To help you fill in these fields, remember:

- **User Name** is normally the agent's email address or *login_name@company_domain*.
- **Employee ID** is a unique corporate ID.
- **Password** might be required (even though it is not indicated as mandatory) depending on the corporate password policy and rules.
- **Default Number** is the agent's telephone number, either a regular directory number or an international number (preceded by **+** and the country code).

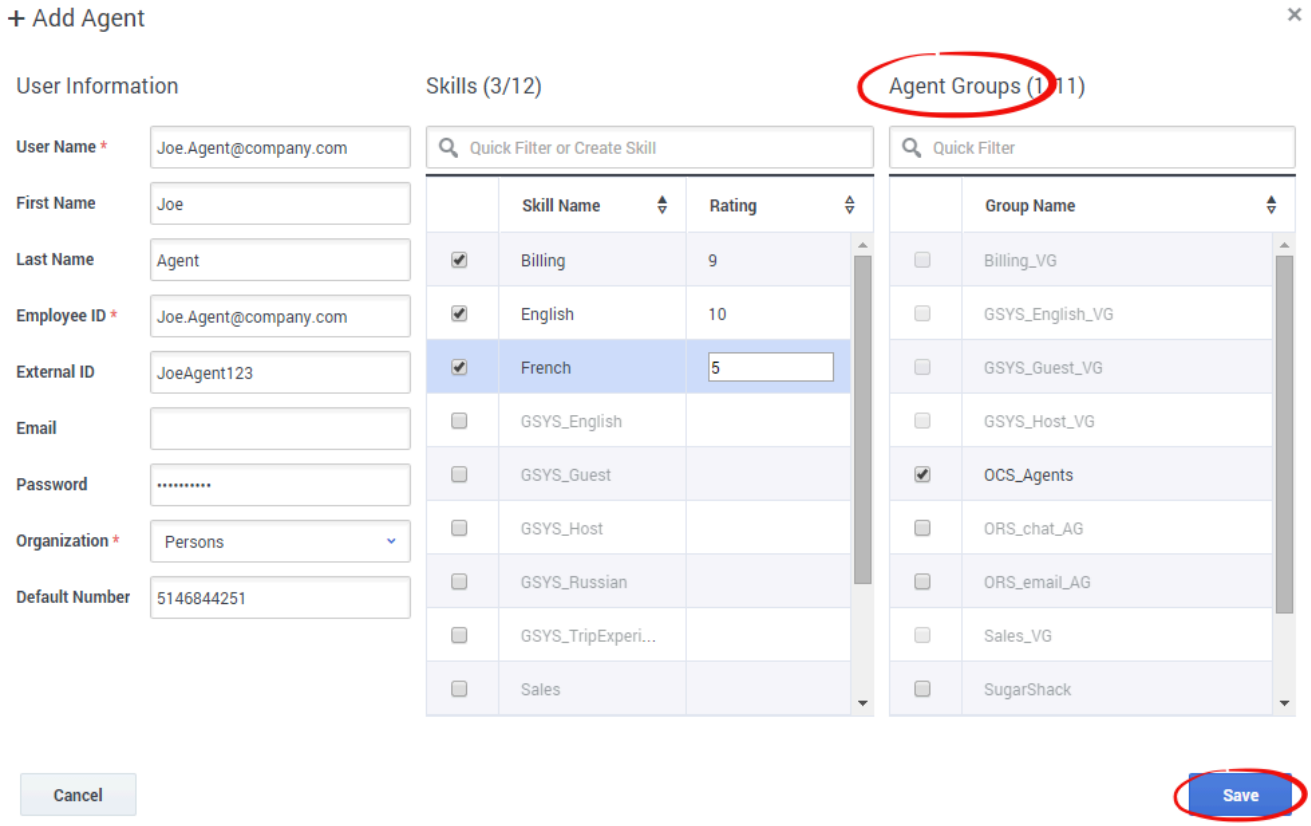
Assign Skills to Agents



Select the Skills that the Agent possesses, and then assign a Rating, or proficiency level, for each assigned skill. Inactive skills are grayed out and unavailable.

A Rating is a numerical representation of the proficiency of the Agent in a skill. Values are 1 to 10, with 10 being the most proficient.

Assign Agents to Agent Groups



Select the Agent Groups to which this Agent belongs.

Platform Administration

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

Welcome to the Platform Administration Help. This document introduces you to the GUI of Platform Administration and describes concepts and procedures relevant to using Platform Administration in your contact center.

About Platform Administration

Platform Administration introduces the next-generation user interface for Genesys that reduces both the overall operating costs and the time to deployment, by providing user-friendly interfaces that perform complex operations while at the same time preventing user error.

Configuration Manager

Within Platform Administration, Configuration Manager enables you to create and manage system-level configuration objects.

The **Configuration Manager** page is a central location for viewing and managing the configuration objects used by your system.

Configuration objects, also known as Configuration Database objects, contain the data that Genesys applications and solutions require to operate in a particular environment. These objects are all contained in the Configuration Database.

Configuration Manager lists these objects by type. For example, configuration objects related to users and their accounts are listed under the **Accounts** heading.

Hover over a configuration object type to view a list of the related configuration objects on your system.

[+] Naming Conventions

Because most objects in the Configuration Database mirror physical objects

in your contact center (for example, switches, agents, and installed applications), this document uses an initial capital letter for Configuration Database objects. For example, the word switch appears in lowercase when it refers to a physical switch in your contact center, but it is capitalized when it refers to the configuration object that mirrors your physical switch. Similarly, the word application appears in lowercase when it refers to a physical installation in your contact center, but it is capitalized when it refers to the configuration object that mirrors the installed program.

Common Object Properties

All objects have the following configuration properties and elements:

- **Name**—Names uniquely identify objects within a certain range. Therefore, the name, which can be up to 255 characters, is a required parameter for most types of objects. The exceptions are:
  - DNs and Agent Logins—Their unique numbers and codes identify them within a Switch.

  The way you name objects in your environment is important. Consistent and sensible naming conventions make your configuration environment easier to understand and faster to browse, leading to a more maintainable and usable configuration.

  > **Important**
  > Although Platform Administration supports the full character set in object names, the use of certain characters can cause problems in the behavior of other Genesys applications. Therefore, avoid spaces, dashes, periods, or special characters in object names. Consider using underscores where you might normally use spaces or dashes.

  The names you set for some types of objects must match the names of the entities that those objects represent elsewhere in an environment. For example, the names of Hosts must match the names given to the computers they represent in the data network environment.
- **State Enabled**—If checked, indicates that the entity represented by an object is in regular operating condition and can be used without any restrictions. If not checked, indicates that the entity represented by an object is being used in a non-production environment. Customer interactions cannot be directed to this target, even if operating information indicates that this object is available.

  Disabling a folder or an object that is a parent to other objects also disables all objects within the folder or all child objects of that parent object.

  For example:
  - Disabling an Agent Group folder disables all Agent Groups configured within this folder.

  However, if you disable a group of objects (for example, an Agent Group), the individual members of this group (in this example, Agents) remain enabled.

Accounts

> **Important**
>
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

The Accounts section of Configuration Manager enables you to configure the following objects:

- Access Groups
- Agent Groups
- Agent Skills
- Roles
- Users (Persons)

Access Groups

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

Access Groups are groups of Users who need to have the same set of permissions for Configuration Database objects.

In many cases, users fall into a small number of categories with similar access needs. A team of agents all performing the same tasks often has identical access needs. Two or three people responsible for maintaining a specific site of the contact center may also have identical access needs. You can greatly simplify access control by adding individuals to Access Groups and then setting permissions for those groups.

> **Important**
> - The **default** user account is not related to Access Groups and, therefore, does not appear as a member of any Access Group.

Overview

Display Options

The **Access Groups** list shows the Access Groups that are in your environment.

> **Important**
> Access Groups that are disabled will appear grayed out in the list.

To select or de-select multiple objects at once, click **Select**.

Click the name of an Access Group to view additional information about the object.

Procedures

Possible Procedures from this Panel

To create a new Access Group object, click **New**. To view or edit details of an existing object, click on the name of the object, or click the check box beside an object and click the **Edit**.

To delete one or more objects, click the check box beside the object(s) in the list and click **Delete**. You can also delete individual objects by clicking on the object and then clicking **Delete**.

> **Important**
> When you delete an Access Group, only the Access Group object itself is removed from the Configuration Database. Its member objects—Access Group and User objects—are not deleted.

Otherwise, select the check box beside one or more objects and click **More** to perform the following tasks:

- **Clone**—Copy an Access Group.
- **Move To**—Move an Access Group to another folder.
- Enable or disable Access Groups
- Create a folder.

Click on the name of an Access Group to view additional information about the object.

Creating Access Group Objects

To create an Access Group, perform the following actions:

1. In the header, click **Configuration**.
2. Under the **Accounts** heading, click **Access Groups**.
3. Click **New**.
4. Enter the following information. For some fields, you can either enter the name of a value or click **Browse** to select a value from a list:
   - **Name**—The name of the Access Group. You must specify a unique value for this property.
   - **Tenant**—N/A
   - **State Enabled**—If selected, indicates that the object is in regular operating condition and can be used without any restrictions.
5. Click **Apply** to save the access group. The **Members** tab appears.

6. In the **Members** tab, click **Add** to add a User. In the pop-up window, you can create a new User object by clicking **New**.
7. Perform one of the following actions after you have added a User to the Access Group:
   - Click **Save** to accept the changes and return to the object list.
   - Click **Apply** to accept the changes and remain in the tab.
   - Click **Cancel** to discard the changes.

Agent Groups

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

An Agent Group is a logical grouping of Agents. Agent Groups are typically set up to provide particular sets of contact-center services.

Overview

Display Options

The **Agent Groups** list shows the Agent Groups that are in your environment.

> **Important**
> Agent Groups that are disabled appear grayed out in the list.

To select or de-select multiple objects at once, click **Select**.

Procedures

Possible Procedures from this Panel

To create a new Agent Group object, click **New**. To view or edit details of an existing object, click on the name of the object, or click the check box beside an object and click **Edit**.

To delete one or more objects, click the check box beside the object(s) in the list and click **Delete**. You can also delete individual objects by clicking on the object and then clicking **Delete**.

> **Important**
> When you delete an Agent Group, only the Agent Group object itself is removed from the Configuration Database. Its member Agent objects are not deleted.

Otherwise, select the check box beside one or more objects and click **More** to perform the following tasks:

- **Clone**—Copy an Agent Group.
- **Move To**—Move an Agent Group to another folder.
- Enable or disable Agent Groups
- Create a folder.

Click on the name of an Agent Group to view additional information about the object.

Creating Agent Group Objects

To create an Agent Group object, perform the following actions:

1. In the header, click **Configuration**.
2. Under the **Accounts** heading, click **Agent Groups**.
3. Click **New**.
4. Enter the following information. For some fields, you can either enter the name of a value or click **Browse** to select a value from a list:
   - **Name**—The name of the Agent Group. You must specify a unique value for this property. You cannot change this value as long as this Agent Group contains at least one User.
   - **Capacity Table**—N/A
   - **Quota Table**—N/A
   - **Cost Contract**—N/A
   - **Site**—N/A
   - **Script**—Enter a valid expression on the **Script** tab to define the group as a Virtual Agent Group. The expression must be in Virtual Group Script Language (VGSL) and must define at least one skill (with optionally, a skill level) in the following format:
     `Skill("SkillName")>SkillLevel`
     Example
     `Skill("Spanish")>5`
   - **Tenant**—N/A
   - **State Enabled**—If selected, indicates that the object is in regular operating condition and can be used without any restrictions.
5. In the **Supervisors** tab, click **Add** to add a User. In the pop-up window, you can create a new User by clicking **New**.
6. In the **Agents** tab, click **Add** to add a User. In the pop-up window, you can create a new User by

clicking **New**.

7. Perform one of the following actions:
   - Click **Save** to accept the changes and return to the object list.
   - Click **Apply** to accept the changes and remain in the tab.
   - Click **Cancel** to discard the changes.

Virtual Agent Groups

Virtual Agent Groups

A Virtual Agent Group is similar to an Agent Group except that a Virtual Agent Group has no permanent members. Instead, an Agent becomes a member of a Virtual Agent Group if that Agent meets the criteria specified by the script. Agent membership in a Virtual Agent Group can change dynamically based on changes in the Virtual Agent Group criteria or changes in the object properties of the Agent.

When you click a Virtual Agent Group in Platform Administration, you see its current member Agents.

Important

Although you can create and configure them using Platform Administration, Virtual Agent Groups are used primarily by Reporting applications.

Warning

If Platform Administration finds Virtual Agent Groups (converted from an earlier installation) that contain illegal script expressions or include permanent members, Platform Administration will display an error message. To preserve correct functionality of the Virtual Agent Groups, you must address the problem manually by either correcting the error or converting the Virtual Agent Group to a non-virtual Agent Group by removing the expression from the configuration option script.

Agent Skills

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

Skills are qualities or abilities that Agents possess and that affect the placement of each Agent in a contact center hierarchy. Common Skills include abilities in different languages, particular categories of product knowledge, or ability in particular types of sales.

Related Topics

- How to Work With Skills
- Route Call Block
- Menu Block

Overview

Display Options

The **Skills** list shows the Skills that are in your environment.

> **Important**
> Skills that are disabled will appear grayed out in the list.

You can sort the items in the list by clicking a column head. Clicking a column head a second time reverses the sort order.

To select or de-select multiple objects at once, click **Select**.

Procedures

Possible Procedures from this Panel

To create a new Skill object, click **New**. To view or edit details of an existing object, click on the name of the object, or click the check box beside an object and click **Edit**.

Otherwise, click **More** to perform the following tasks:

- **Clone**—Copy a Skill.
- **Move To**—Move a Skill to another folder.
- Enable or disable Skills.
- Create a folder.

Click on the name of a Skill to view additional information about the object.

Creating Skill Objects

To create a Skill object, perform the following actions:

1. In the header, click **Configuration**.
2. Under the **Accounts** heading, select **Skills**. The **Skills** list displays.
3. Click **New**.
4. Enter the following information. For some fields, you can either enter the name of a value or click **Browse** to select a value from a list:
   - **Name**—The name of this Skill. You must specify a value for this property, and that value must be unique within the Configuration Database.
   - **Tenant**—N/A
   - **State Enabled**—If selected, indicates that the object is in regular operating condition and can be used without any restrictions.
5. Click **Save**.

Roles

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

Roles define what you can do in a given application. In Platform Administration, roles and their privileges are controlled by the use of Role objects, which are assigned to Users (including Agents) and Access Groups. Roles are application-specific, and must be defined for each application that supports them.

Overview

Display Options

You can sort the items in the list by clicking a column head. Clicking a column head a second time reverses the sort order.

To select or de-select multiple objects at once, click **Select**.

Persons (Users)

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

Persons (or users) are the contact-center personnel, including agents, who need access to Genesys applications. Agents are users who handle customer interactions directly. Non-agent users include managers and supervisors.

User management starts in the Persons window, a central location for creating, provisioning, and managing user accounts. Select **Persons** in the **Accounts** tile of the Configuration Manager toolbar to open it.

[+] What if I want to manage just agents, not other users?

You can create agents in either the Persons window (as described here) or in the Agents window. The result is the same—an agent of type Person is created and stored in the Configuration Database, and you can manage that agent in the **Persons** window or **Agents** window.

If at any time you want to work only with agents, you can select **Agents** in the Configuration Manager menu bar. This shows you a list of only the agents in your contact center, filtering out all non-agent users and giving you a shorter list to work with.

Creating an agent in the **Persons** window might take a little more time, because you must specify all agent properties. In the **Agents** window, some properties, such as **DNs** and **Agent Logins**, are created and assigned automatically.

In addition, the **Persons** window does not offer you additional time- and effort-saving features that are not available in the **Agents** window:

- You can create and modify multiple agents at once (see Create or modify more than one agent at a time).

- You can manage the Skills for selected agents at the same time (see Assign skills to more than one agent at a time.

So unless want to (or have to) define all agent properties yourself and on an agent-by-agent basis, you should consider creating and modifying the agent accounts from the **Agents** window of Configuration Manager.

What do you want to do?

- View users
- Create users
- Modify users
- Remove users
- Enable or disable users
- Create a new users folder
- Move a user between folders
- Reset a user's password
- Set or change a user's password

How do I view users?

The **Persons** window lists the users, both agents and non-agents, in your environment. You can quickly distinguish between agents and non-agents by looking at the icon to the left of the object name, as follows:

 Agent

 Non-agent

See How do I view resources for the different ways to use the **Persons** window to see the agents in your call center.

How do I create a new user?

> **Important**
>
> If you are creating a new agent, the method of creating one and setting it up to handle customer interactions, as described in this topic, requires that you assign all of the properties yourself. If you prefer to use a simpler method, in which some properties, such as **DN**s and **Agent Login**s, are created and assigned automatically, consider creating the agent account from the **Agents** window of Configuration Manager.

You can create a new user from scratch.

> **Important**
>
> Creating a new user by cloning is not supported.

Creating a new user from scratch

To create a new user from scratch:

1. In the **Persons** window, navigate to the folder in which you want to create the user, and click **New..**
2. On the **General** tab, enter the required information. For some fields, you can either enter the name of a value or click **Browse** to select a value from a list. **[+] Show field descriptions**
   .

   The following fields apply to all users, agents and non-agents:

   - **User Name**—Enter the email address of the user.
   - **Agent**—This checkbox indicates in this new user is an agent. It is selected by default, so if your new user is *not* an agent, clear the box.
   - **First Name**—The first name of the user.
   - **Last Name**—The last name of the user.
   - **E-mail Address**—The e-mail address of the user.
   - **Employee ID**—A code of up to 64 characters that identifies this user within the contact-center staff. You must specify a value for this property and that value must be unique.
   - **Password**—A password no longer than 64 characters that this user must use to log in to the environment. You cannot view the existing password. When creating a new user, this field is always enabled. When editing an existing user, this field is enabled only if **Reset Password** is not selected.

> **Important**
> Passwords can be subject to format rules.

- **Confirm Password**—The same password entered in the **Password** field, for confirmation purposes. When creating a new user, this field is always enabled. When editing an existing user, this field is enabled only if **Force Password Reset on Next Login** is not selected.
- **Force Password Reset on Next Login**—A check box that, if selected, will prompt the User to enter a new password upon the next login.
- **External ID**—This setting applies only if your environment is using external authentication, either LDAP or RADIUS. This may be the user name in the external authentication system. For LDAP, it might be a whole, or partial, LDAP URL corresponding to RFC 2255.
- **Tenant**—N/A
- **State Enabled**—If selected, indicates that the object is in regular operating condition and can be used without any restrictions.

If the new user is an agent (**Is Agent** is selected), the following fields also appear:

- **Default Place**—The Place assigned to this agent, by default. This optional parameter applies to a limited number of configurations, in particular those where a real-time association between a place and an agent cannot be established through a telephony login procedure.
- **Capacity Rule**—N/A
- **Cost Contract**—N/A
- **Site**—N/A

3. (Optional) In the **Member Of** tab, click **Add** to add the new user to an Access Group as follows:
   a. In the **Select Access Group** window, either navigate to the folder in which the Access Group exists, or enter the name of the Access Group in the Search box.
   b. Click on the Access Group to which you want the new user to be a member.
   c. Click **Save**.
4. In the **Ranks** tab, click **Add** to add a Rank. Enter the following information in the **New** window:
   - **Application Type**—The type of Application to which a user needs access. Consult the manuals for a GUI application to see if this application is using Ranks to enable or block certain functions. If a manual does not contain any references to Ranks, this application does not use Ranks, and you do not need to specify the Person's Ranks with respect to this application.
   - **Application rank**—The Rank with respect to Applications of the specified type.
5. If the new user is an agent, click **Apply** to display the following tabs.
   - In the Skills tab, click the **Add** button to add a Skill. Enter the following information in the **New** window:
     - **Skill**—The Skill assigned to this agent.
     - **Level**—The relative measure of the agent's proficiency in this Skill.
   - In the Agent Logins tab, click the **Add** button to add an Agent Login. Enter the following information in the **New** window:

- **Agent Login**—The Agent Login code assigned to this agent. You cannot assign an Agent Login to more than one agent.
- **Wrap-up Time**—N/A

6. Click **Save**.

How do I modify a user?

> **Tip**
> If you are modifying only multiple agents, you might consider saving time and effort by importing the changes. See the Can I create or modify more than one agent at a time? section.

To modify the properties for a user:

1. Either click on the user's name, or select a user in the **Persons** window and click **Edit**.
2. Click **Edit** and make the necessary changes.

> **Tip**
> You can also convert an existing agent to a non-agent or an existing non-agent to an agent by clearing or selecting, respectively, the **Agent** checkbox in the user's properties.

How do I remove (delete) a user?

To remove (delete) an agent, refer to Common Resource Operations.

> **Important**
> When you remove a user:
>
> - Any objects assigned to that user are not deleted from the Database.

- All active browser sessions associated with this user are closed. When the user next tries to access the system, he or she will be redirected to the login page.

How do I disable or enable a user?

To enable or disable an agent, refer to Common Resource Operations.

Important

- Disabled user are not removed from your contact center; they are only removed from operations. If you want to remove the agent from your contact center completely, you must delete the agent.
- When a user is disabled, all sessions to which the user is logged in are closed. If the disabled user tries to do any action in one of those sessions, he or she is redirected to the login page.

How do I create a new folder for users?

To create a folder:

1. In the **Persons** window, navigate to the folder in which you want to create the folder, and select **New Folder** from the **More** menu.
2. Enter a name for the folder, and click **Save**.

How do I move a user between folders?

To move a user between folders:

1. In the **Persons** window, select the user to be moved.
2. In the drop-down list in the **Confirmation** window, select the folder to which the Person is to be moved.

How do I reset a user's password?

If a user has lost or forgotten their password, or their password has been jeopardized in some way, you can force them to change their password when they next log in. Use the following steps:

1. In the **Persons** window, select the user whose password is to be reset.
2. On the **General** tab, select **Force Password Reset on Next Login**.
3. Click **Save**.

> ### Important
> **Force Password Reset** is disabled if the password has already been flagged to be reset by the user. It is enabled only when the user logs in and clears the flag by resetting the password.

How do I set or change a user's password?

If your password policy is to not allow a user to change his or her own password, use these steps to set or change the password for a user:

1. In the **Persons** window, select the user whose password is to be set or changed.
2. On the **General** tab, type a new password in the **Password** field and then type the same password (case-sensitive) in the **Confirm Password** field to confirm it. Your organization might have some rules governing what makes up a valid password, so make sure that the password you enter meets these rules.
3. Click **Save**.

Back to top

Customizing your Agent Desktop

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

As an administrator, you can customize what a given agent sees when they login to their Agent Desktop. You also have flexibility in deciding who gets to see what — for example, you can apply a given customization to a single agent at a time (set this on the **Person**) or to a whole group of agents at once (set this on the **Agent Group**).

You have three options for customizing what your agents see when they open their Agent Desktop:

- Web content
- Caller ID
- Corporate favorites

You also have the option to configure inactivity timeouts for your agents:

- Security

Since these customizations are unique to you and your business, they are not available out of the box. To get started on any of these customizations, you need to contact Genesys first. We will do the initial set up for you — create the phone numbers, URLs, or caller IDs that you tell us you need for your business.

> **Important**
> If any of the custom configurations need to be set up once for all Agents, contact Genesys.

Once things are set up in the system, you decide who can access the customization — by setting options in **Person** or **Agent Group** objects in Platform Administrator (see Configuration Notes below).

Let’s take a closer look at the options available to you.

Setting up custom web content

You can integrate external web sites into Agent Desktop, displayed in dedicated tabs.

These tabs give your agents quick access to any web sites that you decide they need, without their having to leave Agent Desktop. For example, your company's knowledge base or corporate search page.

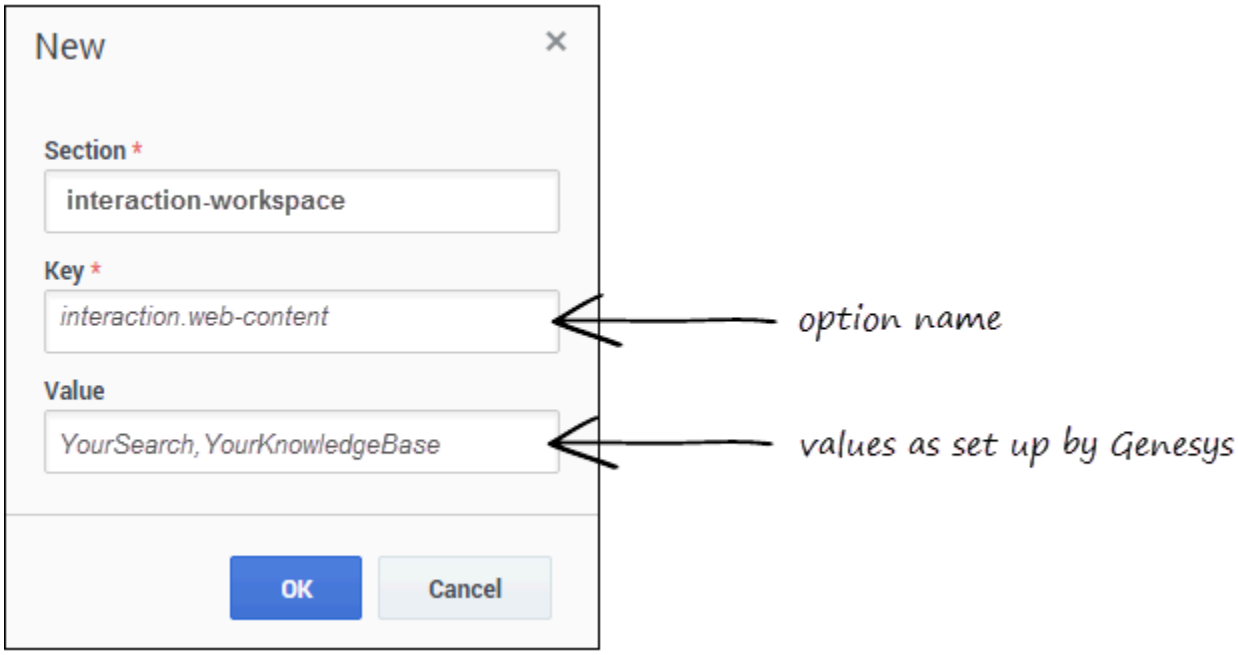
You can set up these web sites to appear in either the **Main View** or the **Interaction View**.

Setup

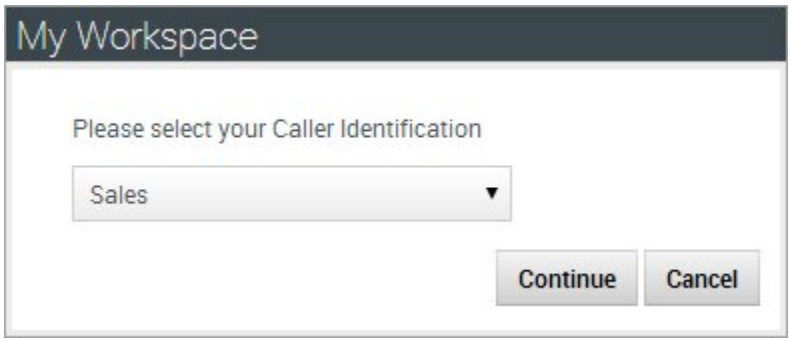
First, contact Genesys to get codes set up for each URL that you want to add to the Agent Desktop.

- **Main View** — You can set the Main View so that multiple tabs appear for different URLs. After you get your codes from Genesys enter them as a comma-separated list in **workspace.web-content**.
  For example,
  `YourSearch,YourKnowledgeBase,YourHelp`
- **Interaction View** — For Interaction View, you can only set up a single web site to appear when the interaction opens. When you get your code from Genesys, enter it in **interaction.web-content**.
  For example,
  `YourSearch`

[+] See sample configuration



Setting up Caller ID selection



> **Important**
> To get the initial setup for this customization added to your system, submit a service request to Genesys.

When the Caller ID feature is enabled, a "Please select your Caller Identification" popup box will appear to your agents when making a call.

Agents can then select the appropriate caller ID from your predefined list to display to the person receiving the call.

Setup

To setup the Caller ID list, contact Genesys. You will receive a code that maps to the list that you want displayed in the Caller ID popup.

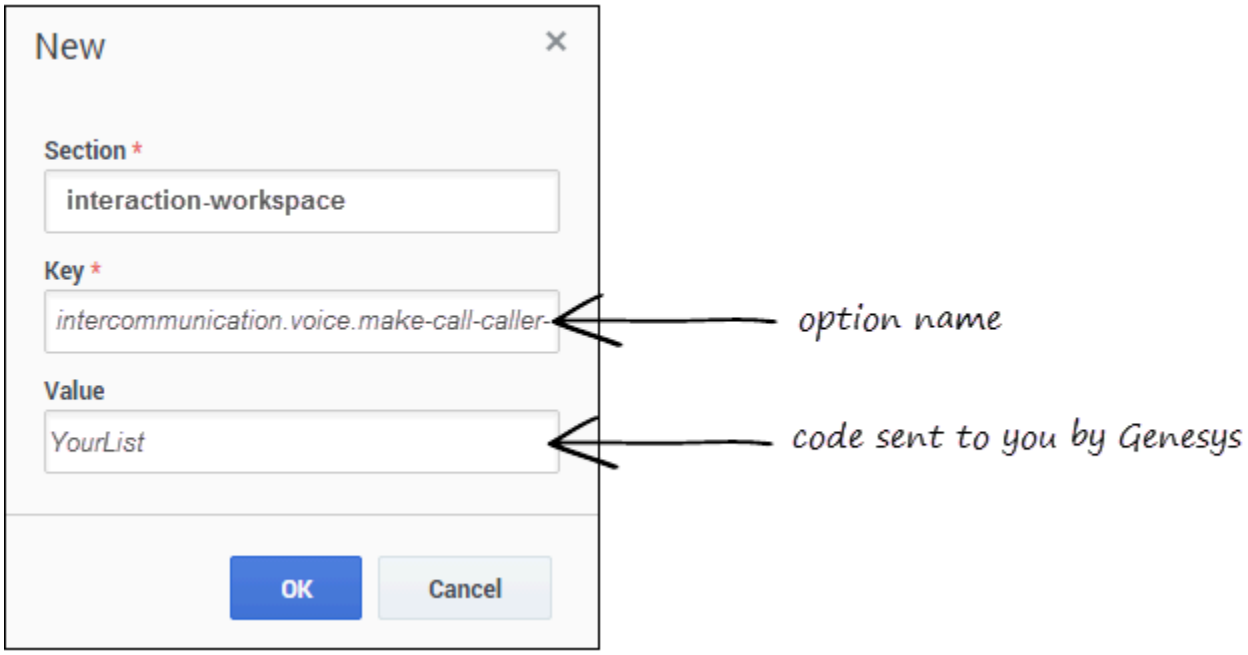
Once you get this code, enter it in the **intercommunication.voice.make-call-caller-id-business-attribute** option.

For example,

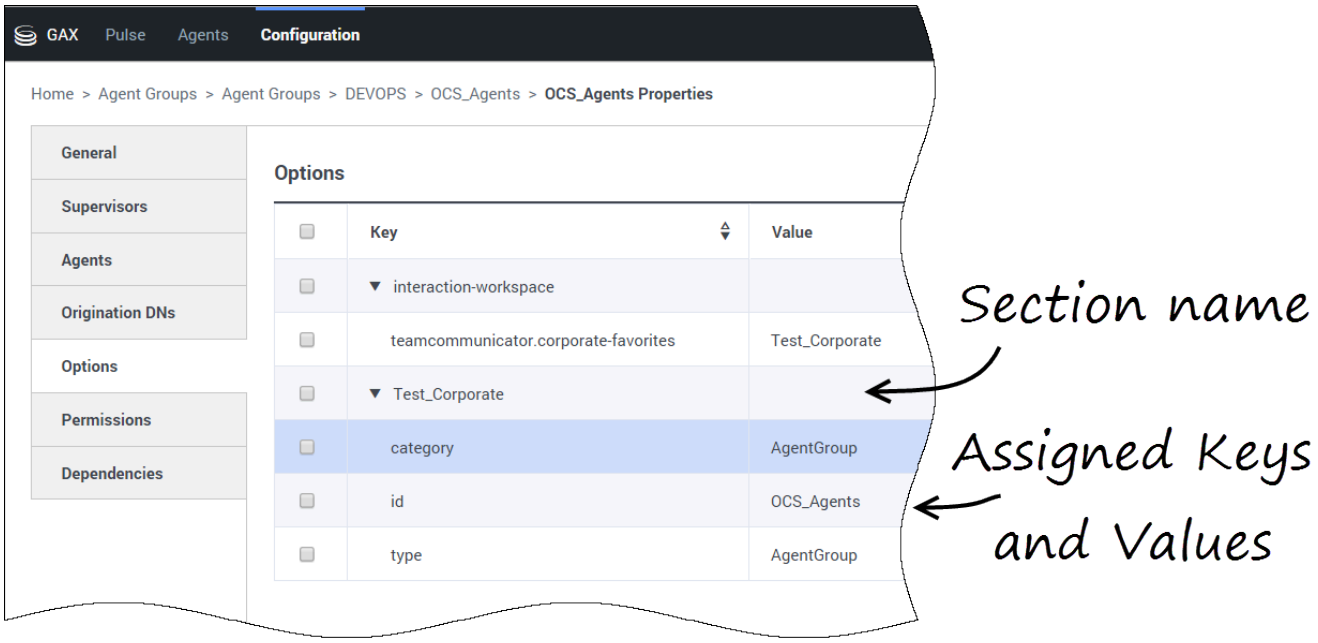
`YourList`

The change will show up for the next time your agent logs in or starts a new session.

[+] See sample configuration



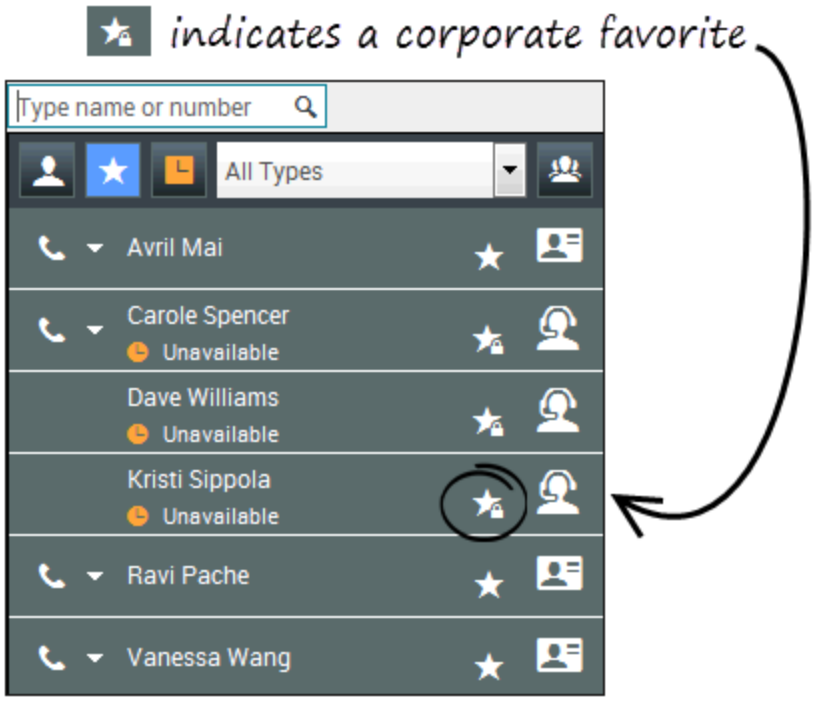
Setting up corporate favorites



A corporate favorite is an internal target or contact that people in your company need to call frequently, or who you want your agents to find quickly.

In your agents' **Team Communicator**, corporate favorites are marked by a locked gold star ( ).

[+] See a screenshot.



You can designate corporate favorites as context-sensitive, determined by a user role, or by the contact you are currently interacting with.

Setup

Basically, you need to create new options for a Person or an Agent Group objects to:

- Define corporate favorites
- Display corporate favorites for select Person(s) or Agent Group(s) in Agent Desktop Team Communicator

Define corporate favorites

To do this, in Platform Administrator, open the properties for the Agent or Agent Group you want to modify and go to **Options**. Add a new entry and enter the details as follows:

- For the **Section**, enter any name of your choosing.
- For the **Key**, enter one of the options available for the section type (see Corporate Favorite Options by Type, below). For example, you could enter *category*, *id*, *type*, or *display-name*.

> **Important**
>
> You can only enter one key at a time, so you will need to repeat this process for each option. Make sure to use the same **Section** name so that the keys are grouped together.

- For the **Value**, enter an option that applies to the key. For example, an *id* key would have a value of the Agent's user ID. A *type* key would have a value such as *AgentGroup*, *Agent*, *Skill*, or *RoutingPoint*. If you are entering more than one value, separate them with semi-colons.

Display corporate favorites for select Person(s) or Agent Group(s) in Agent Desktop Team Communicator

After you have defined the section and applicable key/value pairs, create a new section to display the Corporate Favorites in Agent Desktop Team Communicator:

- For the **Section**, enter `interaction-workspace`.
- For the **Key**, enter `teamcommunicator.corporate-favorites`.
- For the **Value**, enter the name of the section(s) you want the favorites to apply to. If you are entering more than one value, separate them with semi-colons.

[+] See sample configuration

Here is how you could do it for an Agent Group:

A) Create a section called *Test_Corporate* (make sure to use the same section name for all key/value pairs you will apply to this section):

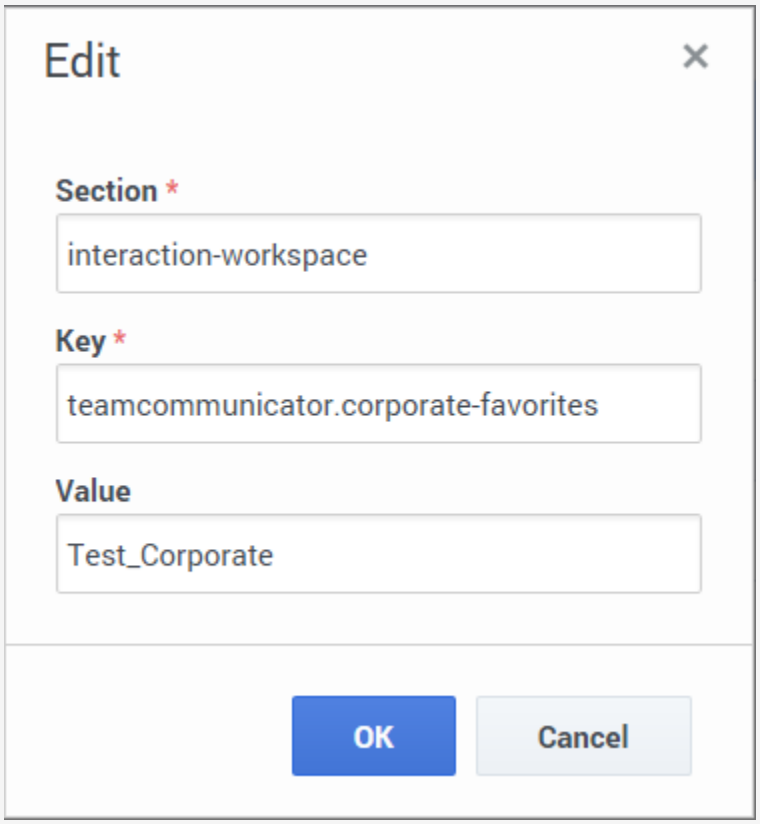
- **Key** = `type` and **Value** = `AgentGroup`.

Repeat this for the other mandatory keys:

- For *id*, **Key** = `id` and **Value** = `OCS_Agents` (as this is the name of the Agent Group we are modifying in this example).
- For *category*, **Key** = `category` and **Value** = `AgentGroup` (as this is an Agent Group category we are modifying — note there are no spaces).

B) Create a section called *interaction-workspace*:

- **Key** = `teamcommunicator.corporate-favorites` and **Value** = `Test_Corporate`.



C) Agents belonging to the modified Agent Group will now see the corporate favorites listed in their Team Communicator interface.

Corporate Favorite Options by Type

Expand each section to see the available **key** and **value** options for each Corporate Favorite type:

Agent

[+] Show options.

Key	Mandatory (Y/N)	Valid Values	Example
type	Y	Agent	Agent
id	Y	<user name of agent>	user123
category	Y	<semicolon-separated list of categories>	CorporateCategory1;FavoriteAgents
display-name	N	<display name of agent>	Jim Brown

Agent Group

[+] Show options.

Key	Mandatory (Y/N)	Valid Values	Example
type	Y	AgentGroup	AgentGroup
id	Y	<name of Agent Group>	AgentGroupMeridian
category	Y	<semicolon-separated list of categories>	CorporateCategory1;FavoriteAgents
display-name	N	<display name of Agent Group>	Meridian

Skill

[+] Show options.

Key	Mandatory (Y/N)	Valid Values	Example
type	Y	Skill	Skill
id	Y	<name of skill>	French
category	Y	<semicolon-separated list of categories>	French-speaking Agents;Mandarin-speaking Agents
display-name	N	<display name of skill>	French

Custom Contact

[+] Show options.

Key	Mandatory (Y/N)	Valid Values	Example
type	Y	CustomContact	CustomContact
category	Y	<semicolon-separated list of categories>	ExternalResources
firstname	N	<any string>	First
lastname	N	<any string>	External
phonenumber	Y (one or both)	<a semicolon-separated list of phone numbers>	+1555234567890;+555123454321
emailaddress		<a semicolon-separated list of email addresses>	external1@mail.dom;external2@mail.d
display-name	N	<display name of custom contact>	Angie Davis

Interaction Queue

[+] Show options.

Key	Mandatory (Y/N)	Valid Values	Example
type	Y	InteractionQueue	InteractionQueue

Key	Mandatory (Y/N)	Valid Values	Example
id	Y	<script name of interaction queue>	123
category	Y	<semicolon-separated list of categories>	CorporateCategory1;FavoriteAgents
display-name	N	<display name of interaction queue>	Quality Assurance

Security

You can configure Agent Desktop to log out an agent from the application if the agent has not used the keyboard or mouse for a period of time. Agent Desktop saves the agent's work, ends the session, logs out the agent on all media channels, and returns to the login page.

Setup

To set this up, configure the following options in the **[interaction-workspace]** section:

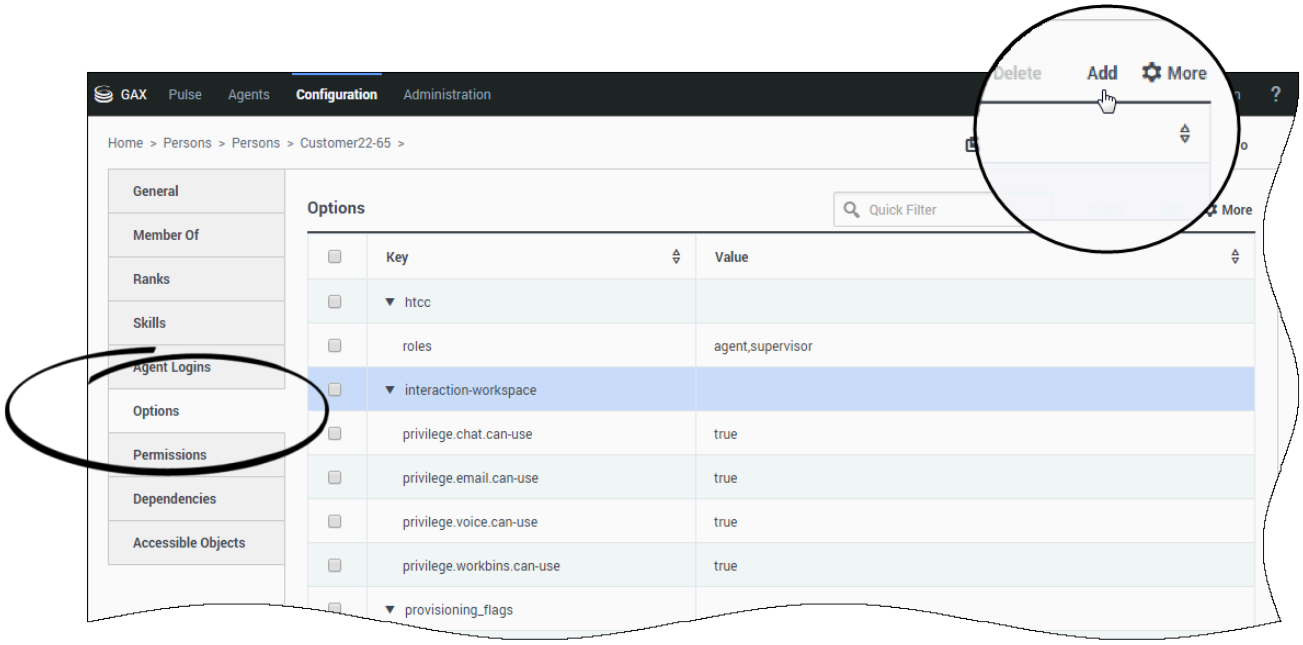
- **security.inactivity-timeout**
  This option specifies how long (in minutes) an agent can be inactive (no mouse or keyboard activity) before Agent Desktop ends their session. The default value for this option is `0`, which disables the inactivity timeout.

- **security.inactivity-alert-dialog-before-timeout**
  This option specifies how much advanced warning (in seconds) Agent Desktop will give to an agent who is about to have their session ended due to inactivity. The default setting is 30 seconds. (You must set this option to a value that is less than the value of `security.inactivity-timeout`.)

The changes for both of these options take effect as soon as the session is started or restarted.

> **Warning**
> After the timeout occurs, the Agent Desktop login dialog should be blank; however, if the agent has allowed the browser to remember his or her password, then the login dialog will still be populated with the agent's credentials. For security, agents should not allow web browsers to remember their login credentials.

Configuration Notes



- For each option, you **must** set the **section** name to `interaction-workspace`.
- You can set these customizations individually, on the Person object for each agent, or globally for each Agent group. Note that the group settings takes precedence over the individual setting.
- Whenever you add multiple values to an option, separate each value with a comma.

Update

To **update** an existing option, just click on it in Platform Administrator. The Edit window opens with all existing values filled out, and you can edit as you need to. For example, to add more favorites to your Corporate Favorites quick dial list.

Delete

To **delete** an existing option, you need to select the check box next to the option. This enables the Delete button in the toolbar. Once the button is enabled, you can click to remove the option. Note that there is no prompt — as soon as you click Delete, it's gone.

Finalize

When adding or changing an option, you always need to close out the change by taking one of these actions:

- Click **Save** to accept the changes and return to the object list.
- Click **Apply** to accept the changes and remain in the **Options** tab.
- Click **Cancel** to discard the changes.

> **Important**
> Any changes made to your Agent or Agent Group configurations will only take effect when a session of Agent Desktop if started or restarted.

Routing/eServices

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

The Routing/eServices section of Configuration Manager enables you to configure the following objects:

- Business Attributes
- Business Attribute Values

Business Attributes

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

Business Attributes are objects that serve as containers of a particular type for Attribute Value objects. Instances of this object are available enumerations (classifications), such as Media Type, Service Type, and others.

> **Important**
> Business Attribute names and values may be stored using UTF-8. This enables the attributes and their values to be entered and displayed in any language, including non-ASCII languages such as Japanese. If required, two or more languages can be combined for a single Business Attribute.

Overview

Business Attributes Overview

The **Business Attributes** list shows the Business Attributes that are in your environment.

> **Important**
> Business Attributes that are disabled appear grayed out in the list.

You can sort the items in the list by clicking a column head. Clicking a column head a second time reverses the sort order. You can add or remove columns by clicking **Select Columns**.

To select or de-select multiple objects at once, click **Select**.

Business Attribute Values

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

A Business Attribute Value is an object which represents a single value within the range of values for a given Business Attribute object.

Overview

Business Attribute Values Overview

The **Business Attribute Values** list shows the Business Attribute Values that are in your environment.

> **Important**
> - The **Business Attributes** list displays when you select **Business Attribute Values** in Configuration Manager. To access the **Business Attribute Values** list, you must first select a Business Attribute object and then a Business Attribute Value folder.
> - Business Attribute Values that are disabled appear grayed out in the list.

You can sort the items in the list by clicking a column head. Clicking a column head a second time reverses the sort order. You can add or remove columns by clicking **Select Columns**.

To select or de-select multiple objects at once, click **Select**.

Scripts

> **Important**
>
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

Scripts identify processing scenarios or treatments that can be applied to customer interactions. For example, an Alarm Reaction Script specifies how to react when an Alarm Condition is triggered. Capacity Rules scripts govern the number and kind of interactions each agent can process.

You can view script metadata (name, type, and whether the script is enabled) but you cannot view the actual script or modify its metadata. If you believe you need a script modified, you must contact Customer Care.

To view a list of your scripts, go to **Configuration > Environment > Scripts**.

Switching

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

The Switching section of Configuration Manager enables you to configure the following objects:

- Agent Logins
- DNs
- DN Groups
- Places

Agent Logins

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

Agent Logins are unique codes defined within a Switch and assigned to agents. They identify which agent is working at which Place during a particular working session.

The configuration of Agent Logins in the Configuration Database must match exactly the configuration of those Agent Logins in the switching system. Before adding or deleting a particular Agent Login, make sure that the same change was made in the database of the switching system.

When you specify Agent Logins as objects in a Switch, they are not associated with any particular agents. For information about how to assign Agent Logins to agents, refer to Creating Agents.

Overview

Agent Logins Overview

The **Agent Logins** list shows the Agent Logins that are in your environment.

> **Important**
> - The **Switches** list displays when you select **Agent Logins** in Configuration Manager. To access the **Agent Logins** list, you must first select a Switch object and then an Agent Logins folder.
> - Agent Logins that are disabled appear grayed out in the list.

To select or de-select multiple objects at once, click **Select**.

Procedures

Possible Procedures from this Panel

To create a new Agent Login object, click **New**. To view or edit details of an existing object, click on the name of the object, or click the check box beside an object and click **Edit**.

To delete one or more objects, click the check box beside the object(s) in the list and click **Delete**. You can also delete individual objects by clicking on the object and then clicking **Delete**.

> ### Important
> When you delete an Agent Login, the Agent Login is removed from the Configuration Database and from any Agent to which it is assigned. This might affect the configuration of a particular Agent in the contact center. Before deleting an Agent Login, consider viewing the dependencies tab to identify the Agent to which this Login is assigned. If you want to remove only the Agent Login from the Agent to which it is assigned, but leave it still existing in the Configuration Database and available for assignment to another Agent, open the User Account for the Agent, remove the Agent Login from the Agent, and click **Save**.

Otherwise, select the check box beside one or more objects and click **More** to perform the following tasks:

- **Clone**—Copy an Agent Login.
- **Move To**—Move an Agent Login to another folder.
- Enable or disable Agent Logins
- Create a folder.

Click on the name of an Agent Login to view additional information about the object.

Creating Agent Login Objects

To create an Agent Login object, perform the following actions:

1. In the header, click **Configuration**.
2. Under the **Switching** heading, click **Agent Logins**.
3. The **Switches** list displays.
4. Click the Switch object in which you wish to create an Agent Login.
5. Click the Agent Logins folder in which you wish to create an Agent Login.
6. Click **New**.

7. Enter the following information. For some fields, you can either enter the name of a value or click **Browse** to select a value from a list:
   - **Code**—The Agent Login code. You must specify a value for this property, and that value must be unique within the Switch. Once you set the value, you cannot change it.
   - **Switch**—The Switch to which this Agent Login belongs. This value is automatically set, based on the Switch being viewed in the **Agent Logins** list.
   - **Override**—N/A
   - **Switch-specific Type**—N/A
   - **Password**—N/A
   - **Confirm Password**—N/A
   - **Tenant**—N/A
   - **State Enabled**—If selected, indicates that the object is in regular operating condition and can be used without any restrictions.
8. Click **Save**.

DNs

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

DNs are communication devices, uniquely identified by their directory numbers (DNs), where customer voice interactions reside and are handled.

Because most types of DNs represent the actual devices of the telephone system, their specification in the Configuration Database must always correspond to their Switch settings. Remember that Platform Administration has no way of verifying this correspondence.

Related Topics

As a general rule, changes made to DN configurations in the Configuration Database must always follow the changes made to DNs within the telephone system, and never the other way around.

Overview

Display Options

The **DNs** list shows the DNs that are in your environment.

> **Important**
> - The **Switches** list displays when you select **DNs** in Configuration Manager. To access the **DNs** list, you must first select a Switch object and then a DN folder.
> - DNs that are disabled appear grayed out in the list.

You can sort the items in the list by clicking a column head. Clicking a column head a second time reverses the sort order. You can add or remove columns by clicking **Select Columns**.

Click **Group By** to group objects by various criteria.

To select or de-select multiple objects at once, click **Select**.

Procedures

Possible Procedures from this Panel

To create a new DN object, click **New**. To view or edit details of an existing object, click on the name of the object, or click the check box beside an object and click **Edit**.

To delete one or more objects, click the check box beside the object(s) in the list and click **Delete**. You can also delete individual objects by clicking on the object and then clicking **Delete**.

> **Important**
> When you delete a DN, it is removed from the Configuration Database and from any DN Group of which it is a member. If you want to remove only the DN from a DN Group of which it is a member, but leave it still existing in the Configuration Database and available for assignment to another DN Group, you must remove it from the DNs tab of the DN Group.

Otherwise, click **More** to perform the following tasks:

- **Clone**—Copy a DN.
- **Move To**—Move a DN to another folder.
- Enable or disable DNs.
- Create a folder.

Click on the name of a DN to view additional information about the object.

Creating DN Objects

To create a DN object, perform the following actions:

1. In the header, click **Configuration**.
2. Under the **Switching** heading, click **DNs**. The **Switches** list displays.
3. Click the Switch object in which you wish to create a DN.
4. Click the DN folder in which you wish to create a DN.

5. Click **New**.
6. Enter the following information. For some fields, you can either enter the name of a value or click **Browse** to select a value from a list:
   - **Number**—A directory number assigned to this DN within the Switch. You must specify a value for this property, and that value must be unique within the Switch for all DN types. Once you set the value, you cannot change it.
   - **Type**—The type of the DN. This value must be set to **Extension**.
   - **Switch**—The Switch to which this DN belongs. This value is automatically set, based on the Switch being viewed in the **DNs** list.
   - **Association**—N/A
   - **Register**—N/A
   - **Alias**—N/A
   - **Route Type**—N/A
   - **DN Group**—The DN Group to which this DN belongs.
   - **Override**—N/A
   - **Login ID**—N/A
   - **Switch-specific Type**—N/A
   - **Number of Trunks**—N/A
   - **Tenant**—N/A
   - **State Enabled**—If selected, indicates that the object is in regular operating condition and can be used without any restrictions.
7. Click **Save**.

DN Groups

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

DN Groups are logical groupings of DNs. You can use DN Groups in network-level routing algorithms and in some types of statistics.

Although a DN Group can contain DNs that belong to a number of different Switches, you can receive correct statistical information about this DN Group only if the reporting applications connect to the servers associated with those Switches.

Overview

Display Options

The **DN Groups** list shows the DN Groups that are in your environment.

> **Important**
> DN Groups that are disabled appear grayed out in the list.

You can sort the items in the list by clicking a column head. Clicking a column head a second time reverses the sort order. You can add or remove columns by clicking **Select Columns**.

To select or de-select multiple objects at once, click **Select**.

Places

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

A Place is a location that has one or more DNs operated by a single agent.

You configure Places and assign individual DNs to them in order to monitor performance and availability, and to provide this information to call-processing applications.

Overview

Display Options

The **Places** list shows the Places that are in your environment.

> **Important**
> Places that are disabled appear grayed out in the list.

You can sort the items in the list by clicking a column head. Clicking a column head a second time reverses the sort order.

To select or de-select multiple objects at once, click **Select**.

Procedures

Possible Procedures from this Panel

To create a new Place object, click **New**. To view or edit details of an existing object, click on the

name of the object, or click the check-box beside an object and click **Edit**.

To delete one or more objects, click the check-box beside the object(s) in the list and click **Delete**. You can also delete individual objects by clicking on the object and then clicking **Delete**.

> **Important**
>
> - You should only reconfigure a Place object when there are no Agents logged into the Place. Doing so can have an impact on both the status of the Place in Pulse reports and call routing to the associated Agents.
> - When you delete a Place, it is removed from the Configuration Database and from any Place Group it belongs to. If you want to remove a Place, remove it from the Place Group. This keeps it in the Configuration Database and allows it to be added to another Place Group.

Otherwise, click **More** to perform the following tasks:

- **Clone**—Copy a Place.
- **Move To**—Move a Place to another folder.
- Enable or disable Places.
- Create a folder.

Click on the name of a Place to view additional information about the object.

Creating Place Objects

To create a Place object, perform the following actions:

1. In the header, click **Configuration**.
2. Under the **Switching** heading, click **Places**.
3. In the **Places** window, click **New**.
4. Enter the following information. For some fields, you can either enter the name of a value or click **Browse** to select a value from a list:
   - **Name**—The name of this Place. You must specify a unique value for this property.
   - Capacity Rule—N/A
   - **Cost Contract**—N/A
   - **Site**—N/A
   - **Tenant**—N/A
   - **State Enabled**—If selected, indicates that the object is in regular operating condition and can be used without any restrictions.

5. In the **DNs** tab, click **Add** to add a DN. In the pop-up window, you can create a new object by clicking **New**.
6. Perform one of the following actions after you have added a DN to the Place:
   - Click **Save** to accept the changes and return to the object list.
   - Click **Apply** to accept the changes and remain in the tab.
   - Click **Cancel** to discard the changes.
7. Click **Save**.

Agents

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

Agents are Users who handle customer interactions directly. The **Agents** window lists all Agents in your environment (or Tenant, if you are in a multi-tenant environment). You can only see those objects for which you have access.

You can filter the contents of this list in several ways:

- Click **Show Quick Filter** and type a search term in the **Quick Filter** field. The list updates dynamically to show items that match the text in the **Quick Filter** field.
- Click **Show Column Filter** to show search fields for each column header. Enter a search term in one of these fields to quickly search the column for the search term.
- Click the cube icon to open the **Tenant Directory** window. In this window, click the Tenant that you want to select. Use the **Quick Filter** field to filter the Tenant list.

You can sort the items in the list by clicking a column head. Clicking a column head a second time reverses the sort order.

Creating an Agent

To create a new Agent, click **Add**.

[+] Show Procedure

Procedure: Creating an Agent

Purpose: To create an Agent in the **Agents** window.

Steps

1. Click **Add**. The **Add Agent** window appears.
2. Enter the following information. For some fields, you can either enter the name of a value or click **Browse** to select a value from a list:
   - User Information
     - **User Name**—The name that this User should use to log into the environment. You must specify a value for this property, and that value must be unique within the Configuration Database.
     - **First Name**—The first name of this User.
     - **Last Name**—The last name of this User.
     - **External ID**—This setting applies only if your environment is using external authentication, either LDAP or RADIUS. This may be the user name in the external authentication system. For LDAP, it might be a whole, or partial, LDAP URL corresponding to RFC 2255.
     - **Email**—The e-mail address of this User.
     - **Employee ID**—A code of up to 64-characters that identifies this User within the contact-center staff. You must specify a unique value for this property.
     - **Password**—A password no longer than 64-characters that this User must use to log in to the environment. You cannot view the existing password.

       > **Important**
       > Passwords can be subject to format rules.

     - **Organization**—The folder in which the Person object for this User is stored.
     - **Default Number**—The phone number of this User. This field is used to create the associated DN, Place, and Agent Login objects for this User. Platform Administration uses existing objects, if available.
   - Skills
     - You can add Skills to the Agent by:
       - Creating a Skill—Type the name of the new Skill in the **Quick Filter** field and click **+** to create the Skill.
       - Selecting an existing Skill—In the **Skills** section, select existing Skills from a list to add to the Agent. To select a Skill, click the check box beside the Skill and enter a numerical value in the **Rating** field.

> **Important**
> Since Agent Groups might be defined automatically based on a Skill (Virtual Agent Groups), the list of Agent Groups refreshes if you create a Skill.

- Agent Groups
  - You can add the Agent to an Agent Group by clicking the check box beside an item in the list.

3. Click **Save**.

> **Important**
> When adding an Agent in **Cloud** mode, Platform Administration also makes the following configuration changes:
>
> - Create Agent Logins in each Switch, using information that was entered in the **Default Number** field.
> - Create the User object and associate Agent Logins to the User object. **Wrap-up Time** for each Agent Login is set to `0`.

Other Actions

Once you select an Agent, you can:

- Edit Agent Information—Select an Agent and click **Edit** to edit Agent information. If you edit the **Default Number** field:
  - If this field is emptied, the Agent is disassociated from the current Place and its Agent Login objects are emptied.
  - If a new value is given, the agent is associated with a new Place, DN, and Agent Login that match the value. New objects are created, if they do not already exist. Existing objects are emptied and not deleted.
- Copy an Agent—Select an Agent and click **Clone** to make a copy of the Agent object.
- Change State—Select an Agent and click either **Enable** or **Disable** to change the state of an Agent.

Important

- Agents that are disabled appear grayed out in the list.
- When a User is either disabled or removed, Platform Administration invalidates all sessions associated with this User. Upon the User's next action, he or she will be redirected to the login page.

To delete one or more Agents, click the check box beside the Agent(s) in the list and click **Delete**.

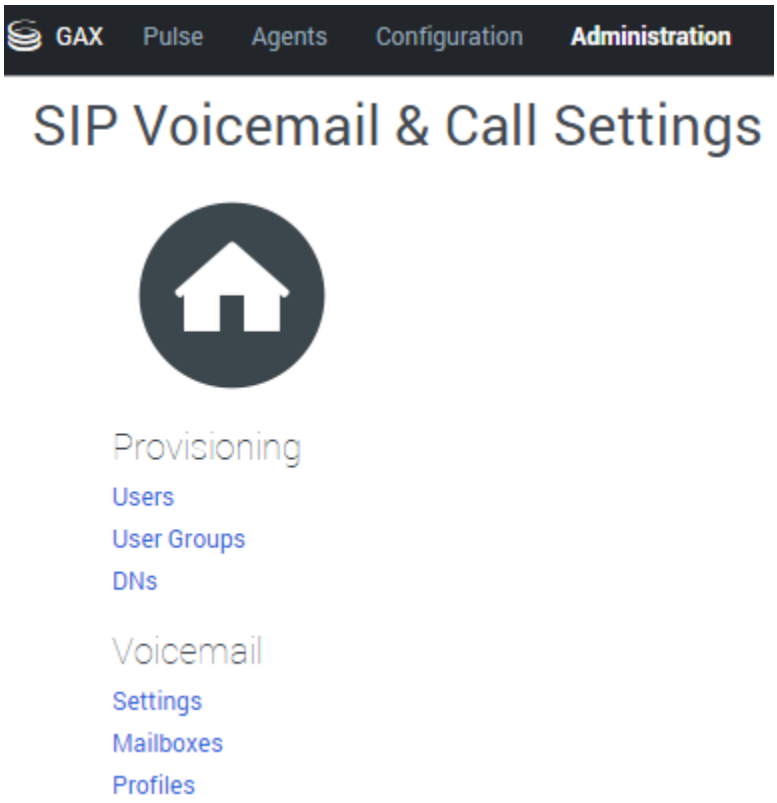
Important

- When deleting an Agent, Platform Administration does not delete the DN, Place, or Agent Login objects assigned to the Agent.
- You should only reconfigure a Place object when there are no Agents logged into the Place. Doing so can have an impact on both the status of the Place in Pulse reports and call routing to the associated Agents.

Voicemail Management

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

Voicemail management gives you control over Users and User Groups, DNs, Settings, Mailboxes, Greetings, and Voicemail Profiles.



To Access Voicemail Management...

1. Start Genesys Portal and select login to **Platform Administration**.
2. Select **SIP Voicemail and Call Settings** from the Administration menu.
3. Choose a category to manage:

- Users
- User Groups
- DNs
- Settings
- Mailboxes
- Greetings
- Voicemail Profiles

Managing Voicemail Users

Here, you can manage user roles, mailbox access, voicemail profile, time zone, DNs, voicemail notification and web service notification preferences:

General tab

1. Select **Users**, then search for and select a specific user. You can enter the first letter to see a list.
    - Specify Roles, Mailbox Access, Voicemail profile, and Time Zone from drop-down menus.
    - Click the values in the Agent Logins, DNs, and Mailboxes tables to view details.

Feature Server Roles field:

- Select one:
    - `User` (default) grants the user access to voicemail.
    - `Administrator` grants the user the ability to log in as administrator and perform all the tasks available in Agent Desktop.
    - `Group Mailbox Administrator` grants the user the ability to log in as administrator and manage group mailboxes. Only users who are Group Mailbox Administrators can change greetings and passwords for group mailboxes.

User Mailbox Access drop-down menu:

- Select one:
    - `Phone + Web View + Web Playback` enables the user to access voicemail over the phone and to view and play voicemail.
    - `Phone + Web View` enables the user to access voicemail over the phone and to view, but **not** play, voicemail.
    - `Phone Only` enables the user to access voicemail over the phone, without web access.

Voicemail Profile drop-down menu:

- To activate voicemail access, select a profile other than `Not Set`.

Time Zone drop-down menu:

- Select a time zone for message playback.

Email Notifications tab

- Set Notifications On or Off in the drop-down menu.
- Specify a destination email in the **Email To** field. Use the standard address format: *name*@*domain*.

Web Service Notifications tab

- Set Notifications On or Off in the drop-down menu.
- Specify a destination phone in the **Phone Number** field. Use only digits.
  Note: If your notification message does not include the user phone number, this field does not appear.

Managing Voicemail User Groups

You create user groups and perform most provisioning in Genesys Administrator, not here.

Here, you can manage user-group voicemail profiles:

General tab

1. Select **User Groups**, then search for and select a specific user. You can enter the first letter to see a list.
2. Specify a Voicemail profile from the corresponding drop-down menu.
   Voicemail Profile: To activate voicemail access, select a profile other than `Not Set`. If the menu is empty, you can create a voicemail profile or let the default calling profile apply to the user.

Email Notifications tab

- Set Notifications On or Off in the drop-down menu.
- Specify a destination email in the **Email To** field. Use the standard address format: *name*@*domain*.

Web Service Notifications tab

- Set Notifications On or Off in the drop-down menu.
- Specify a destination phone in the **Phone Number** field. Use only digits.
  Note: If your notification message does not include the user phone number, this field does not

appear.

Managing Voicemail DNs

To create DNs, see the topic DNs.

1. Select **DNs**, then search for and select a specific DN. You can enter the first number to see a list.
    - You can optionally set a password for the DN. This password controls device authentication.

Managing Voicemail Settings

General tab

- Set **Voicemail Enabled** to Yes or No in the drop-down menu.
- Click the value for **Voicemail Profile** to edit it.

Email Notifications tab

The system uses notification defaults if the voicemail profile doesn't contain a value for a given field.

- Turn **Notifications** On or Off by selecting from this drop-down menu.
- Enter the email address to receive notifications in the **Email to** field.

Web Service Notifications tab

The system uses notification defaults if the voicemail profile doesn't contain a value for a given field.

- Turn **Notifications** On or Off by selecting from this drop-down menu.
- Enter the phone number to receive notifications in the **Phone number** field.

Managing Voicemail Mailboxes

- Select **Mailboxes**, then search for and select a specific mailbox. You can enter the first number to see a list.
- Before you select a mailbox, you can enable or disable the Disclaimer. Click on that word to see the

status and instructions.

You can configure these settings for the selected mailbox:

Setting	Values (default value in bold)	Description
Status	**Active**, Locked	When Locked, four unsuccessfully times, locks you out for 10 minutes. Select Active to override.
Voicemail Deposit Enabled	**Yes**, No	**Yes** forwards unanswered calls to voicemail under various conditions, depending on the options set. **No** plays a "voicemail is disabled" message to the caller.
Mailbox Password	**System**, *user-selected*	Press **Reset** to reset the password to the system (default) value.
Max Messages	**10**, 1 to *n*	Select the second radio button and type a value to set a new maximum number of messages. Select **System** to restore the system (default) value.
Optout Phone	**System (Not Set)**, *any phone number or routing point*	When set, enables a caller to transfer out of voicemail to the specified destination at any time during a call. Select the second radio button and type a value to set a new optout phone. Select **System** to restore the value to the number in parentheses, which is the value set at the application or switch level for the configuration option `voicemail-optout-destination`.
Time Zone	**System**, *time zone from menu*	Select a time zone from the menu to set a new time zone for all mailboxes that use the system (default) time zone. Select **System** to restore the system value.
Language	**System (English(United States))**, *language from menu*	Select a language from the menu to set a new language. Select **System** to restore the system value.
Assigned	n/a	The Assigned table lists the objects (directory numbers, agents, users, user groups) to which this mailbox has been

Setting	Values (default value in bold)	Description
		assigned. For some objects you can click the object name to view the object.
Messages	Unread/Read(Unread high-priority messages/Read high-priority messages)	Press **Delete All** to delete all normal and high-priority messages.

Managing Greetings

Users can manage greetings for their personal mailboxes.
Users who are Group Mailbox Administrators can change greetings and passwords for group mailboxes, by accessing the group mailbox remotely, as described here.

To manage your greetings from a local or remote telephone, press 4 and use these commands:

- Press **1** to record an `extended absence greeting`.
- Press **2** to record a `personal greeting`.
- Press **5** to activate the `standard greeting` that inserts your recorded name into a standard message. For example: You have reached the mailbox of *play_recorded_name*.
- Press **6** to activate your `personal greeting`. Not available or played unless this greeting exists.
- Press **7** to activate your `extended absence greeting`. Not available or played unless this greeting exists.

After initiating each action, follow the audio prompts.

- Press * to exit the Greetings menu and return to main menu.

Managing Voicemail Profiles

About Voicemail Profiles

Voicemail profiles determine how long voicemails are kept for a user or user group before deletion. Voicemail profiles use a Class of Service model to enable the quick assignment of voicemail notifications and retention limits to user groups or specific collections of users.

You can create profiles that set retention limits of 1 to 10,000 days, or use **`No Limits`** to set voicemails not to expire. You can effectively disable voicemail profiles by keeping the default Retention Limit value of the System Profile, **`No Limits`**, and assigning no other profile to your users.

The System Profile applies only when none of the users or user groups assigned to a mailbox has an

assigned profile.

- To create a new profile, select **New**. To edit an existing profile, select it from the list.

General tab

- **Name** is an editable field.
  - Empty if you clicked **New**. Enter a name.
  - Populated if you selected an existing profile. You can change its name if you want.
- **Retention Limit** specifies the limit for each profile, in number of days.
  - Use the radio buttons to select `No Limits` or the editable field.
- **Notifications** are described in Notifications, below.

Notifications tab

- Set each of the first four values to `Yes` or `No` using its drop-down menu.
  - **Email Notification Enabled** Yes enables email (SMTP) notifications.
  - **Email Notification Allow User Setup** Yes allows users and user group administrators to turn email notifications on or off for themselves, and to specify the recipient email address. (`No` allows only administrators tocontrol user and user group settings.)
  - **Web Service Notification Enabled** Yes enables web (HTTP) notifications.
  - **Web Service Notification Allow User Setup** Yes allows users and user group administrators to turn web notifications on or off for themselves, and to specify the recipient phone number. `No` allows only administrators can control user and user group settings.
- All other Notifications settings are described in Notification Defaults.

Agent Scripting Administration

Agent Scripting Administration (Echoscripting) is a scripting tool used to prompt agents through the call-handling process with customers. It can be used for either inbound customer service or outbound telemarketing calls. Agent Scripting is a powerful tool that supports branching to guide agents through different paths of the script, depending on the customer's answers to scripted questions. Additionally, agents can capture information about customers and update the customer database, process orders, or track ongoing customer service calls.

Agent Scripting (Echoscripting) User Guide

Agent Setup

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

Agent Setup is a cloud-based web application that allows you to manage your cloud-based contact center and your agent accounts for products such as Agent Desktop and Gplus Adapters.

Link to video

Agent Setup now has a new and enhanced user interface, making it easier to create users, set up agent groups, and to configure desktop options.

See an overview of the changes in this video.

> **Tip**
> Be sure to check out other Agent Setup videos below!

Documentation

Click any of the topics below for details.

Contact Center Configuration

Configuring desktop settings

Managing desktop statistics

Agent Provisioning

Adding agents

Importing multiple agents

Managing desktop options
Managing CRM Adapter options
Managing Agent Groups
Managing Transactions
Managing Templates

Exporting agents

Agent Setup Videos

Below are some video tutorials on working with Agent Setup to configure and provision your contract center.

Note:

- While the Agent Setup interface in these videos has changed slightly in each release, the steps are basically the same for each release.

How to add a single agent Link to video	**How to update multiple agents** Link to video
How to import multiple agents Link to video	

Getting started

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

Before you start, review the following web browser requirements and access information to gain a full understanding of the Agent Setup tool. The intended audience for this guide are administrators that are responsible for creating and managing user accounts, configuring the call center settings, and assigning global settings to agent groups.

Web browser requirements

You access the Agent Setup interface through one of the following supported web browsers:

- Microsoft Internet Explorer 11
- Google Chrome 54 or later
- Mozilla Firefox 45 or later
- Microsoft Edge

Agent Setup User Roles

When setting up user accounts, it's important to understand how the Agent Setup user and administrator roles are distinguished and what Genesys product to use to add the accounts. Essentially, two main roles exist:

- **Agent Setup users**: Users are assigned to the **Administrator** access group, which grants the user the correct role to perform basic administration of users. To add Agent Setup users accounts to your contact center, import the administrator accounts or use Platform Administration.
- **Agent Setup administrators**: These users are assigned to both the **Administrator** access group and the **Agent Setup Administrator** access group. The Agent Setup Administrator access group is associated with the correct role and permissions to perform administrative tasks in Agent Setup. To add administrator accounts to your contact center, you can either import the administrator accounts or use Platform Administration.

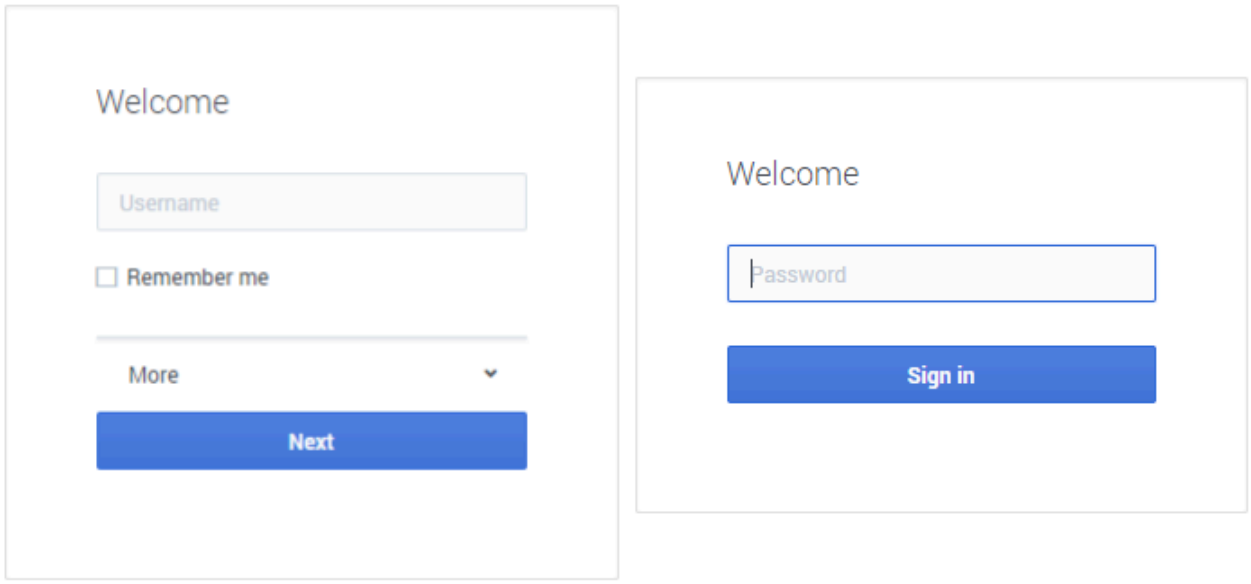
Note: In your access group folder, each of these access groups are prepended with your company name. For example, Genesys Administrator.

How do I access the Agent Setup interface?

Access to Agent Setup requires a web browser. If you haven't already, read Before you set up an agent.

You can access the Agent Setup interface by logging into your Genesys Portal and clicking the **Agent Setup** icon.

Logging in

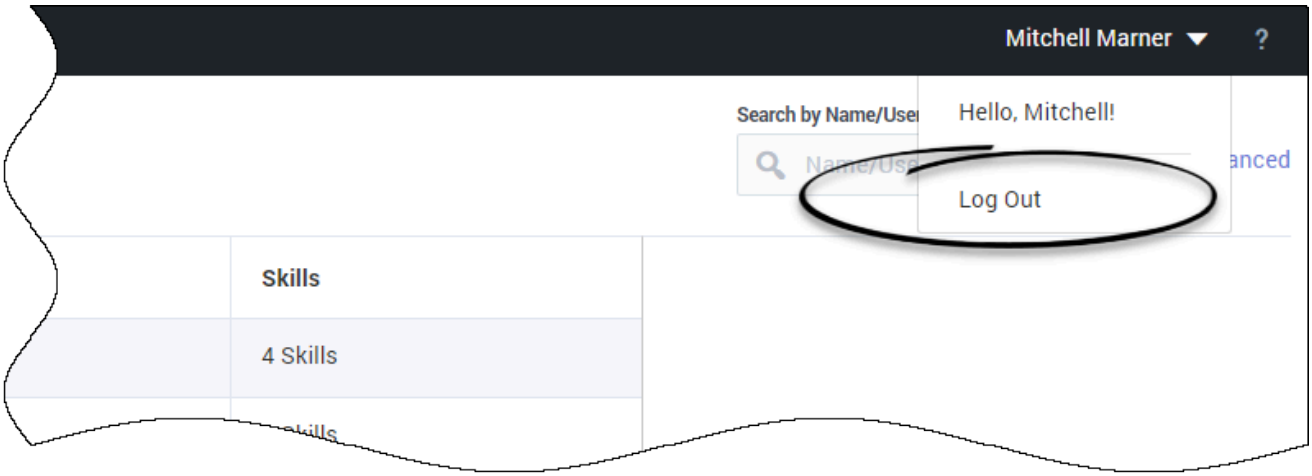


After you log into the Genesys Portal, the Agent Setup displays another login window and you must log in again.

Type your username, click **Next**, type your password, and click **Sign In**.

Note: You are not required to enter a Tenant name.

Logging out



When you have completed your administration tasks, click the menu in the upper-right corner, and select **Log Out**.

Bulk Import

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

As you set up and maintain your call center, you may sometimes need to input a large amount of data into the system. Using Agent Setup, you can import agent accounts and customer contact records in bulk, making the process smoother and more reliable. When you have a large number of objects to import, use the **Bulk Import** page under **Data Tools** in the Agent Setup application menu.

How do I import a file?

Link to video

Before you can import agent accounts or customer contact records in bulk, you must prepare a CSV file that contains all the required data.

The values from the imported file overwrite corresponding values that are specified in the template. For example, the value for the **Skill** parameter in the imported file takes precedence over the value for the **Skill** parameter that is specified in the template.

When you are ready to import the Source File, select the file from your local device, then click **Import**. You can validate the file to ensure that it is structured correctly.

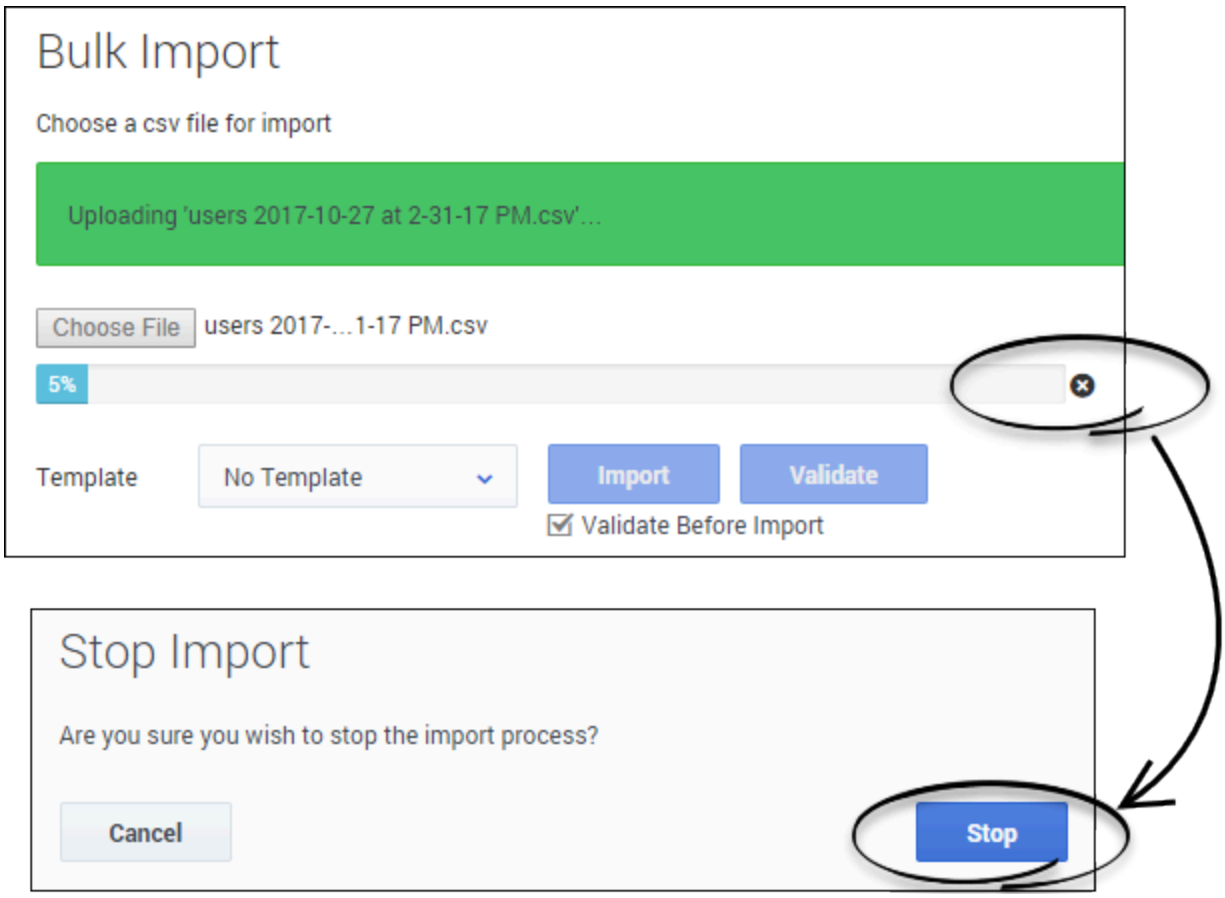
About CSV files

The CSV file must be a text file in a comma-separated format. In the source file each line represents a single user or agent.

When you import a CSV file, each parameter value is updated according to the following rules:

- If a value is present, the new value replaces the previous value.
- If an empty value is encountered, no change is made.
- If the value includes `/d` , the attribute is deleted.
- If no corresponding record is found, an error message is logged and the update continues to process.

How do I stop an import?



When you stop an import, a window asks you to confirm that you really want to stop the import. After you click **Stop**, agent records that were imported before you clicked ⊗ are retained and the import process does not proceed. By clicking **Stop**, you are not cancelling the entire import.

Importing agent accounts

By importing a large number of agent accounts at one time, you won't need to add the accounts one at a time and you can apply a template to configure options, such as DN, place, person, and log in options, to all accounts.

CSV file parameters

Property	Type	Mandatory?	Action	Description
Username	String	Yes	ADD, UPDATE, DELETE	The unique username of the agent or user.

Property	Type	Mandatory?	Action	Description
First Name	String	Yes	ADD, * needed for UPDATE if First Name change	The first name of the agent or user.
Last Name	String	Yes	ADD, * needed for UPDATE if Last Name change	The last name of the agent or user.
Email	String	No		The agent or user's email address.
Extension	String	No	/d to unassign and delete /r to unassign only	The extension #1 of the agent. A place is also created if the extension is specified.
Extension2	String	No	/d to unassign and delete /r to unassign only	The extension #2 of the agent. A place is also created if the extension is specified.
Extension3	String	No	/d to unassign and delete /r to unassign only	The extension #3 of the agent. A place is also created if the extension is specified.
Folder	String	No		The folder path to root folder (Persons) where new agent or user is created. For example, Company/ Department/Team.
Place Name	String	No	/d to unassign and delete /r to unassign only	The place name #1 of the agent. The place name "Extension" is used if not specified.
Place Name2	String	No	/d to unassign and delete /r to unassign only	The place name #2 of the agent. The place name "Extension2" is used if not specified.
Place Name3	String	No	/d to unassign and delete /r to unassign only	The place name #3 of the agent. The place name "Extension3" is

Property	Type	Mandatory?	Action	Description
				used if not specified.
Default Place	String	No		The name of the Default Place.
Employee ID	String	Yes	ADD	The employee identifier of the agent.
Change Password	Boolean	No		Specifies that the user will be prompted to change the password on their next log in.
Enabled	Boolean	No		Y specifies that the agent account is enabled. N specifies that the agent account is disabled.
Is Agent	Boolean	Yes	ADD	Y specifies that this user is an agent.
External ID	String	No		Any external ID used to identify the user.
Login ID	String	No		The user's login ID.
Is Supervisor	Boolean	No		Y specifies that this user is a supervisor.
Is Admin	Boolean	No		Y specifies that this user is an administrator.
Wrap Up Time	Numeric	No		The number of seconds an agent remains in ACW after call is disconnected. This field applies only to voice calls.
VoiceMail.MailBox	String	No		If specified, populate the following option to the Annex of all associated Agent log ins: AgentLoginAnnex:TServer/gvm_mailbox. Valid values: numerical (12345)

Property	Type	Mandatory?	Action	Description
Recording Hierarchy	String	No		If specified, populate the following option to the Agent's Annex: [recording] agent_hierarchy.
SIP Phone Type	String	No		The type of SIP phone the agent is using as a softphone.
Skill:<skill_name>	Numeric	No		The number that identifies the agent's skill level for the specified skill.
Switch:<switch_name>	Boolean	No		**Y** means that a DN and Agent Login are created for the agent on the specified switch.
AgentG:<agent_group_name>	Boolean	No		**Y** means that the agent is added to the specified agent group.
Annex	String	No		Any custom Annex value. For example, column name **Annex:htcc/roles** with a value of **Agent**.
Action	String	No		ADD adds a new agent. UPDATE updates an existing agent. DELETE removes an existing agent.

CSV file example

```
First Name,Last Name,Employee Id,Username,Is Agent,Is Supervisor,Is Admin,External
Id,Login ID,Extension,Skill:English,Switch: San Fran,Switch:
London,AgentG:Good,AccessG: Super,Annex,Action,Folder,Wrap Up Time,Enabled
Sarah,Lee,223344,sarahl@acme.com,N,Y,Y,ADD,Acme/Branch/Team 1,60
John,Doyle,223465,johnd@acme.com,Y,N,N,2233,1,Y,N,ADD,Acme/Branch/Team 2,0
Robert,Cook,244456,bobc@acme.com,Y,Y,Y,johnd@acme.com,5,Y,Y,ADD,Acme/Branch/Team
3,60,Y
```

> **Important**
>
> Bulk imports do **not** include the **Password** field in the imported file. A default password is associated to each user account in the imported file. The format of the default password is: *<username>*@*<company-name>*.com-*<Month>*-*<year>*, where *<Month>* is the three-digit abbreviation for the month. For example, `hjackson@demo.com-Feb-2017`. The **Reset Password** option is automatically enabled when adding or importing new users, so all users are prompted to reset their password the next time that they log in.

Importing multiple contacts

By importing contact records with Bulk Import, you can avoid having agents add contacts manually.

> **Important**
>
> After importing customer contact records, you must log into Agent Desktop and use Contact Directory to view the imported records.

CSV file parameters

Property	Type	Mandatory?	Action	Description
First Name	String	Yes	ADD, * needed for UPDATE if First Name change	The first name of the contact.
Last Name	String	Yes	ADD, * needed for UPDATE if Last Name change	The last name of the contact.
Phone Number	String	Yes		The contact's phone number.
Action	String	Yes		ADD adds a new contact. UPDATE updates an existing contact. DELETE removes an existing contact.

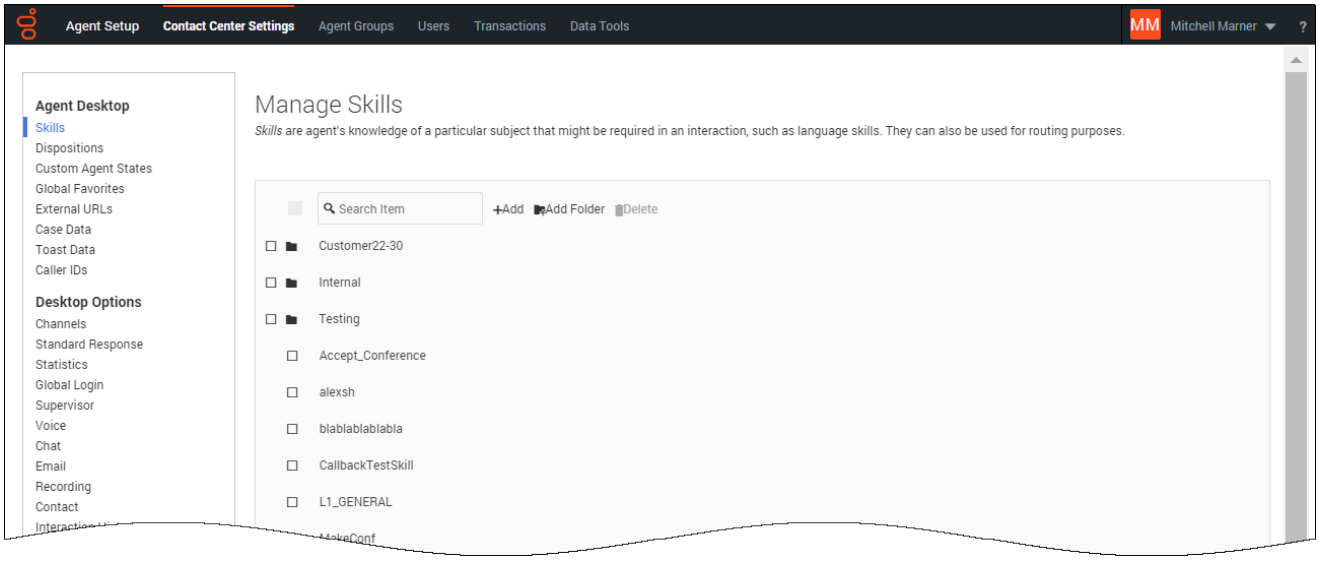
Contact Center Settings

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

A key factor in a well-configured contact center is how well the global contact centers are configured. This important task ensures your customers are routed to the correct agents and that the agents have the correct tools available to them as they handle customer interactions.

Configuring your contact center settings

The settings can be configured in any order, however, for initial setup, we recommend that you configure your contact center in the order displayed in the left navigation menu. The recommended order is also described in the sections below.



> **Tip**
> On the various screens, you select the check boxes next to the options that you want

to enable. Each check box has three states:

-  The option is inactive. To activate the option, click the check box once. The option is then set to the default setting as specified in the description in the tool tip.
  -  A check mark indicates that the option is enabled. You can click to disable the option.
  -  No check mark indicates that the option is disabled. You can click to enable the option.

Agent Desktop settings

Create desktop objects to customize the Agent Desktop for your agents. Objects include items that are specific to your business needs, such as disposition codes, Caller IDs, global favorites, and more. For more information, see Agent Desktop settings.

Desktop Options

Agent Setup includes a list of configuration options that you can customize for your contact center. Use these settings to enable, disable, or configure values for the various interaction channels and Agent Setup features that are available for your agents to use. For more information, see Desktop Options.

Agent Groups

Agent Groups are logical groupings of agent accounts, grouped by skill sets. See Agent Groups.

Transactions

Sometimes it might be necessary to override the desktop settings defined in Agent Desktop settings. Overrides allow you to manage business attributes that are applied at a transaction level, such as a call or a chat. To create business attribute overrides, see Transactions.

Desktop Statistics

You will need to add, edit, import and export statistics for the My Reports and Contact Center sections of Agent Desktop. To configure statistics, see Desktop Statistics.

Templates

Now that your contact center is configured, create templates that you can apply when you provision your contact center with agents. Templates define sets of configuration options such as login, place, DNs, and more, so that when you add agents, you don't need to build each agent account's settings

individually. For more information, see Templates.

CRM Adapter settings

Configure options for the Gplus Adapter. For more information, see CRM Adapter settings.

Next steps

After you configure your contact center settings, you can easily manage user accounts, because all the correct options will be available to you.

Agent Desktop settings

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

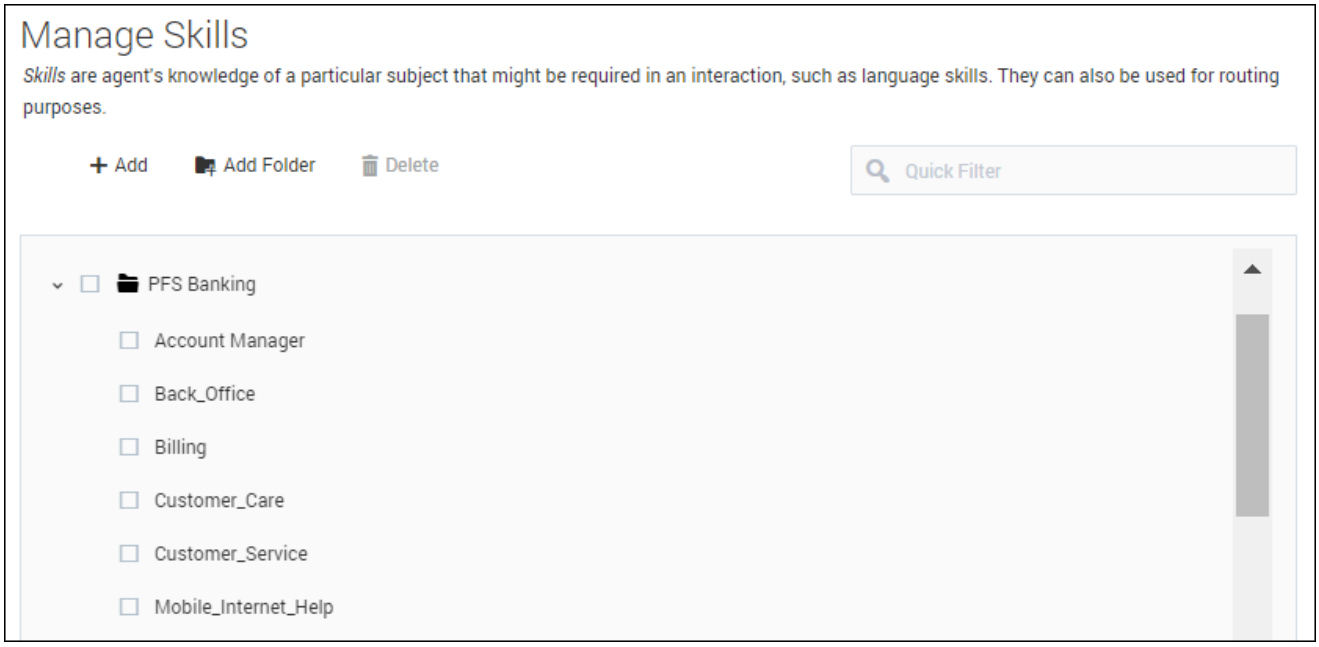
Configuring your Agent Desktop settings involves customizing the available options to meet the specific needs of your contact center.

To access the **Contact Center Settings** tab, click **Contact Center Settings** on the application menu across the top of Agent Setup. On the **Contact Center Settings** tab, you will see a navigation menu on the left hand side. You can see only the objects that permissions enable you to view. If you need to edit an object that you don't see on the interface, contact your administrator.

You can customize the following objects:

Skills

On the **Skills** page, you can add and delete skills, as well as organize the skills into folders.



Skills are an agent's knowledge of a particular subject that might be required in an interaction, such as language skills. They can also be used for routing purposes.

To add a skill:

1. Click **Add**.
2. Type a name in the **Name** field.

You can also organize skills into folders. If needed, select **Add Folder** to create a new folder.

Dispositions

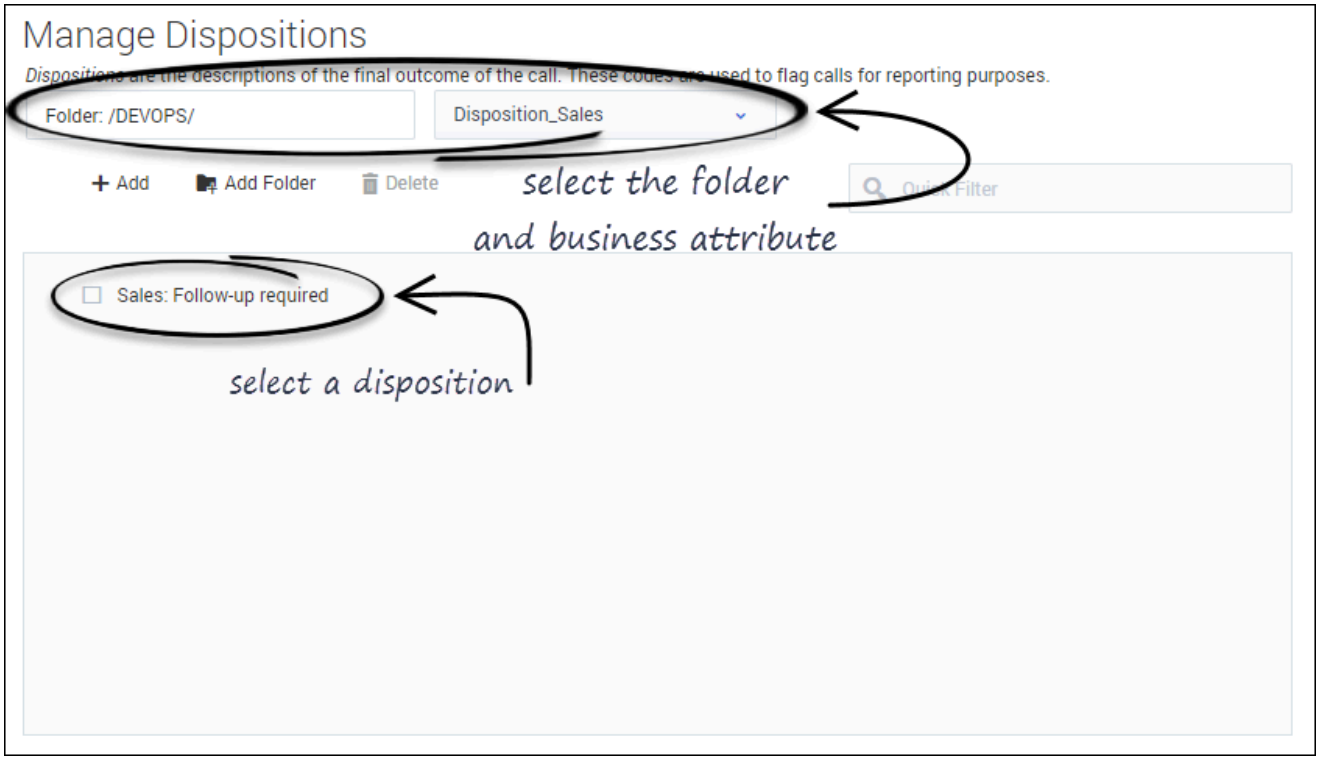
On the **Dispositions** page, you can create a list of Disposition options to be displayed when an agent makes an outbound call or transfer. *Dispositions* are descriptions of the final outcome of a call. These codes are used to flag calls for reporting purposes.

Creating a sharable Disposition business attribute

Dispositions can be used as sharable business attributes for agents, agent groups, and the global contact center. To view an article about how to create sharable business attributes, see Sharing Business Attributes in Agent Setup.

> **Tip**
>
> - To view an example of Disposition options in Agent Desktop, see What do I do at the end of a call? in *Agent Desktop Help*.
> - In the table of configured Dispositions, in addition to selecting and adding Dispositions, you can also filter items using the **Quick Filter** field and delete a Disposition.

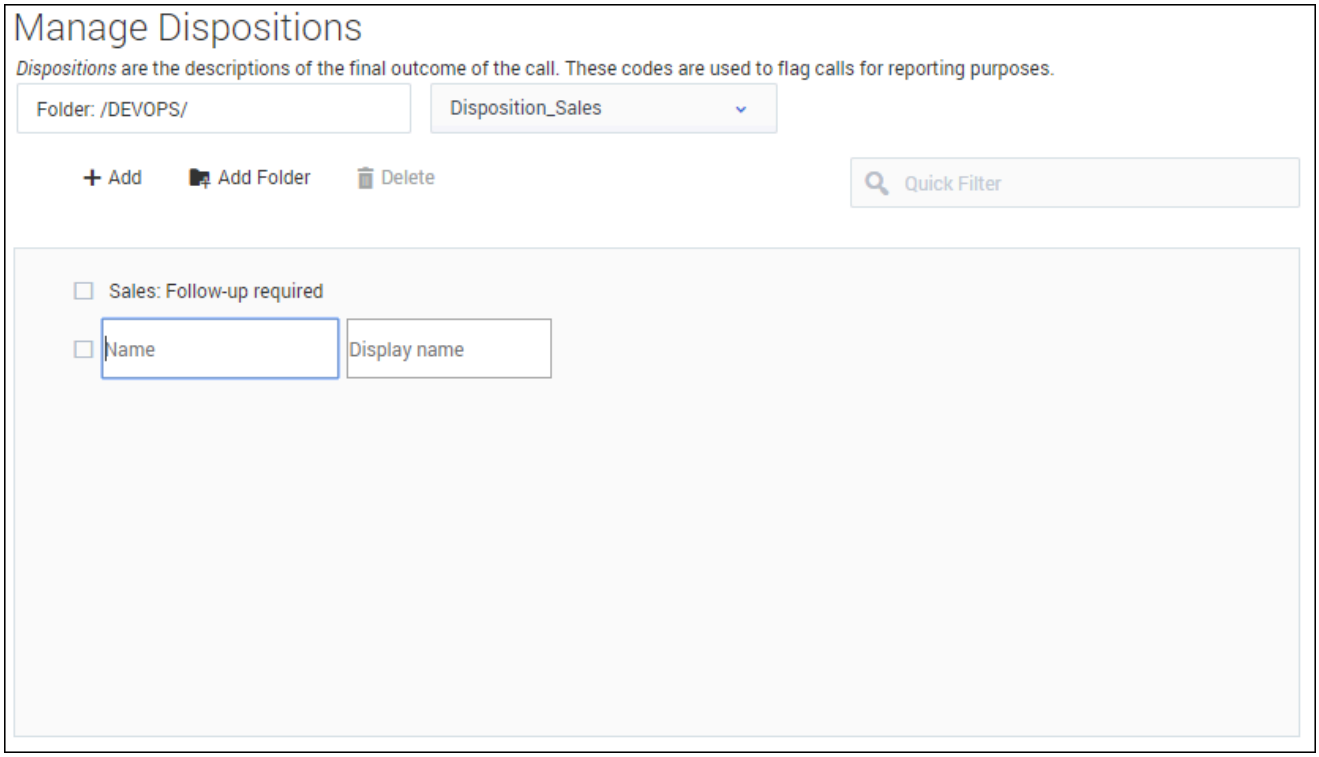
Selecting a disposition



1. Select a business attribute:
   a. Select the folder you want to view from the **Folder** drop-down list box.
   b. Click the second drop-down list box and select the business attribute for the Disposition.

      You can consider a business attribute to be the container that holds a list of Dispositions that have a common theme.
3. Select a Disposition from the list.

Adding a disposition



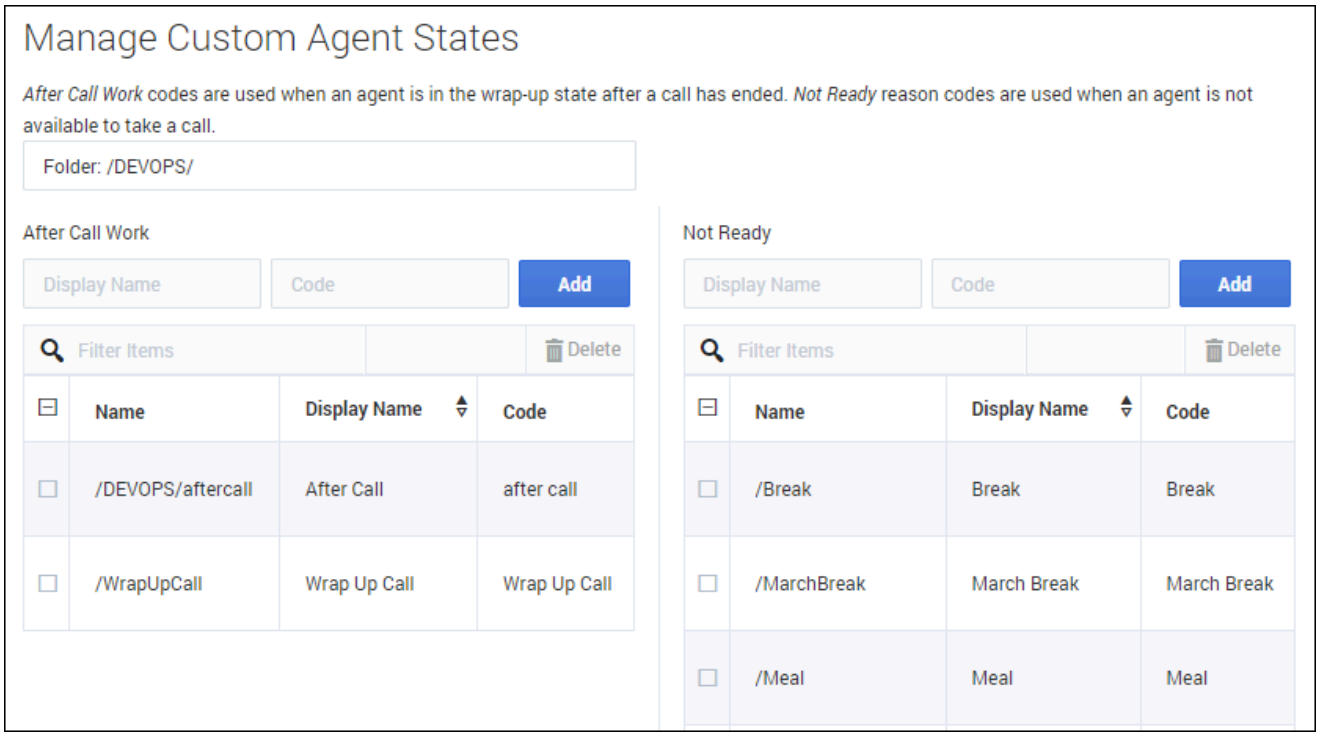
1. Select the folder you want to view from the **Folder** drop-down list box.
2. Click the second drop-down list box and select the business attribute into which you want to add the Disposition.
3. Click **Add**.
4. Type the **Name** and **Display Name** for the new Disposition.

Tip

To add a new folder to the list, you can do one of the following:

- Click **Add Folder**; or
- To add a subfolder, click the **Folder** drop-down list box, select a folder, and then click the **Add folder** icon.

Custom agent states

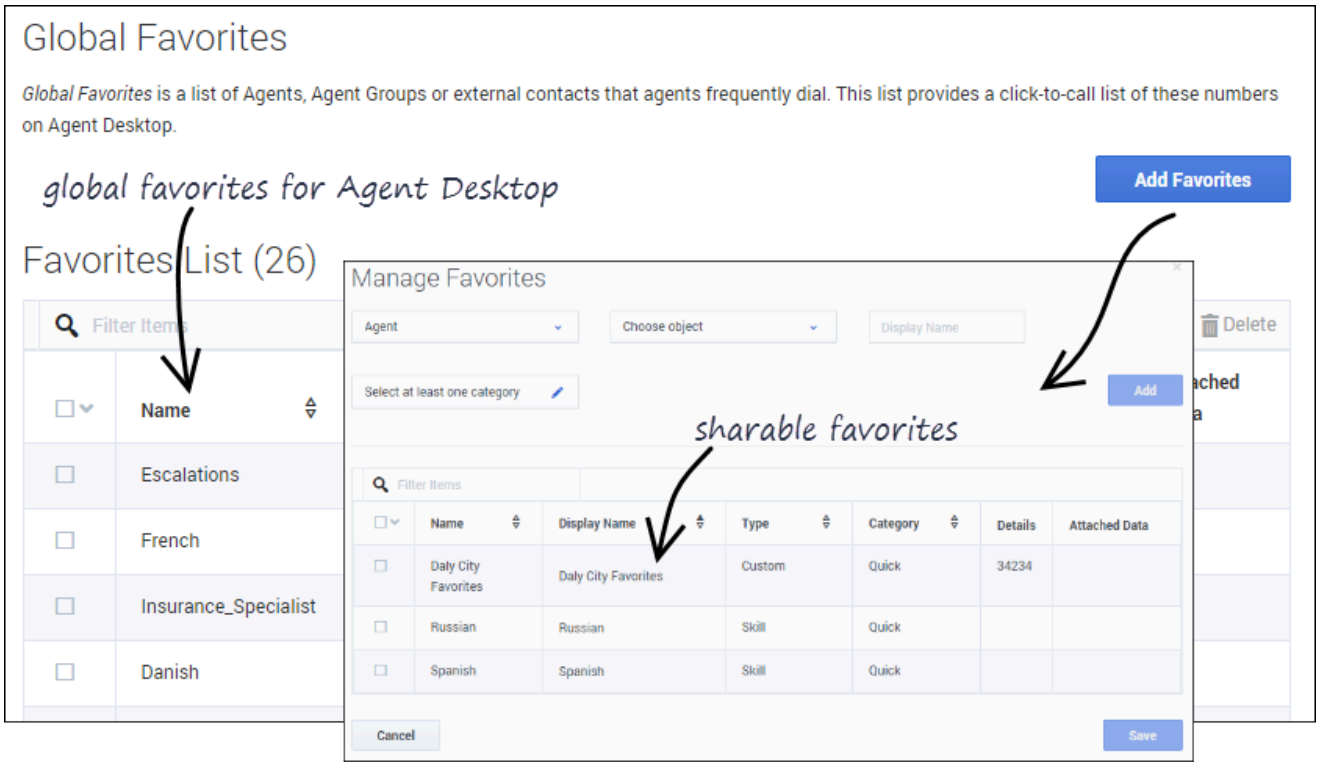


On the **Custom Agent States** tab, you can add custom agent states that an agent can select for After Call Work and Not Ready reasons.

You can organize custom agent states into folders. When you select the **Folder** option, a drop-down list box is displayed where you can select a folder or click the add folder icon to create a new folder.

Once you have selected the folder, enter an appropriate **Display Name** (example: *Lunch Break*) and a relevant **Code** that is unique.

Global Favorites



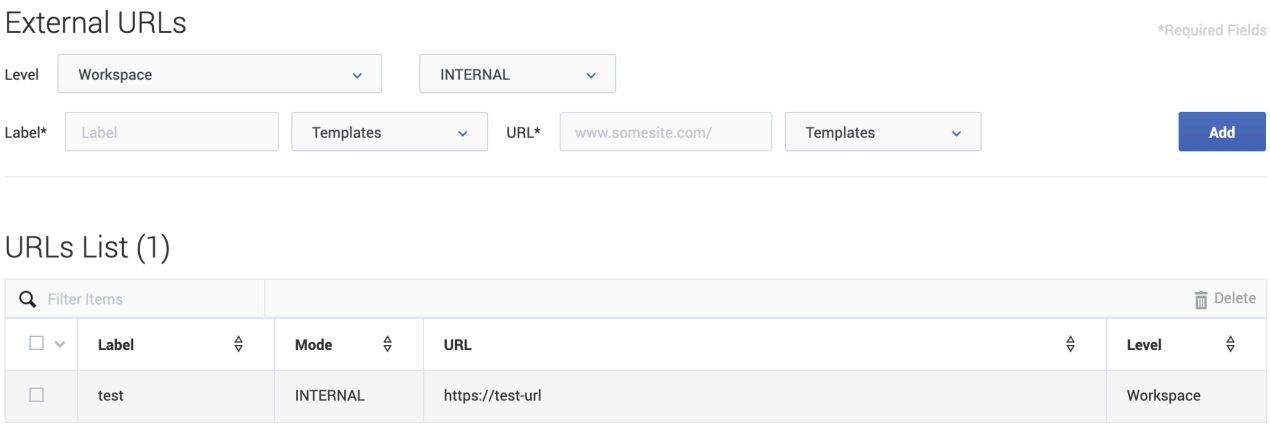
As a contact center administrator, you can create and maintain a list of Favorites, which are contacts and internal targets that your agents can access quickly in the Team Communicator. To see how agents use Global Favorites in Agent Desktop, see How do I set up my favorites? in the *Agent Dektop Help* guide.

In addition to managing favorites from the **Desktop Settings** page, you can also manage favorites on the **Agent Groups** page and the **Add/Manage User** window. No matter from what screen you add a favorite, you can include the favorite in the list of shareable favorites to be reused in all places where Global Favorites (or Personal Favorites for user accounts) are available for configuration in Agent Setup.

To learn more about how to work with Global Favorites, including a scenario-based demonstration, see Working with Global Favorites.

External URLs

You can integrate internal and external websites into your agent's main view. At the Agent Desktop level, the external website is displayed in dedicated tabs. These tabs allow you to access the website without leaving Agent Desktop.



At the interaction level, there are two ways to display websites depending on the main interaction type.

- For voice, the external website is displayed in the background when an interaction is selected. When the interaction is unselected or closed, the external website is replaced with the previous panel displayed in the background.
- For multimedia, the external website is displayed in a dedicated view in the case data.

To add a URL to the list, select **Workspace** or **Interaction** from the **Level** menu, type a label for the URL, type or paste the URL in the URL field, and click **Add**.

Depending on the Level you choose - Workspace or Interaction - the following options populate in the Templates menus for both Label and URL:

Workspace:

- Agent.FullName
- Agent.UserName
- Agent.LastName
- Agent.FirstName
- Agent.EmployeeId

Interaction:

- Agent.FullName
- Agent.UserName
- Agent.LastName
- Agent.FirstName
- Agent.EmployeeId
- AttachedData.{{attached-data-key-name}}

- Interaction.id
- AttachData.KVPName

An external web site URL can be defined using the `$AttachedData.Y$` parameter to have flexible text in the URL where Y represents an attached data name.

> ### Warning
>
> - If the `$AttachedData.Y$` parameter is specified for the external website URL, and Y is not part of the attached data associated with an interaction, the external website is not displayed.
> - If an agent manually adds the attached data to the interaction (or if it's added by any other means), then the external website tab is displayed with the content of the external website.

For example, the external website specified as follows:

```
label = Customer Location
url = https://www.bing.com/search?q=$AttachedData.CustomerLocation$
```

With this URL, if an interaction is received with a `CustomerLocation` attached data with the value of `Paris`, then the external website will be displayed in a dedicated tab called **Customer Location**, and the external web site will be `https://www.bing.com/search?q=Paris`

- `mode`: Specifies the mode Agent Desktop uses to display the web application. The default value is `INTERNAL`. The possible values are:
  - `INTERNAL`: The web application is displayed in a dedicated tab in Agent Desktop. It can be viewed and used when the agent clicks the tab to display it. This is the default behavior.
  - `EXTERNAL`: The web application is displayed in a popup window.
  - `BACKGROUND`: The web application is displayed in the background when the interaction is selected. When the interaction is deselected or closed, the third-party website is replaced with the previous panel that was displayed in the background. This mode is only applicable at the interaction level.
  - `HIDDEN`:The web application is loaded but not displayed in Agent Desktop. You can use this mode to integrate Agent Desktop with a non-web application.

Case Data

On the **Case Data** page, you can specify the information to be displayed to agents when a call, chat, or other interaction is transferred to them.

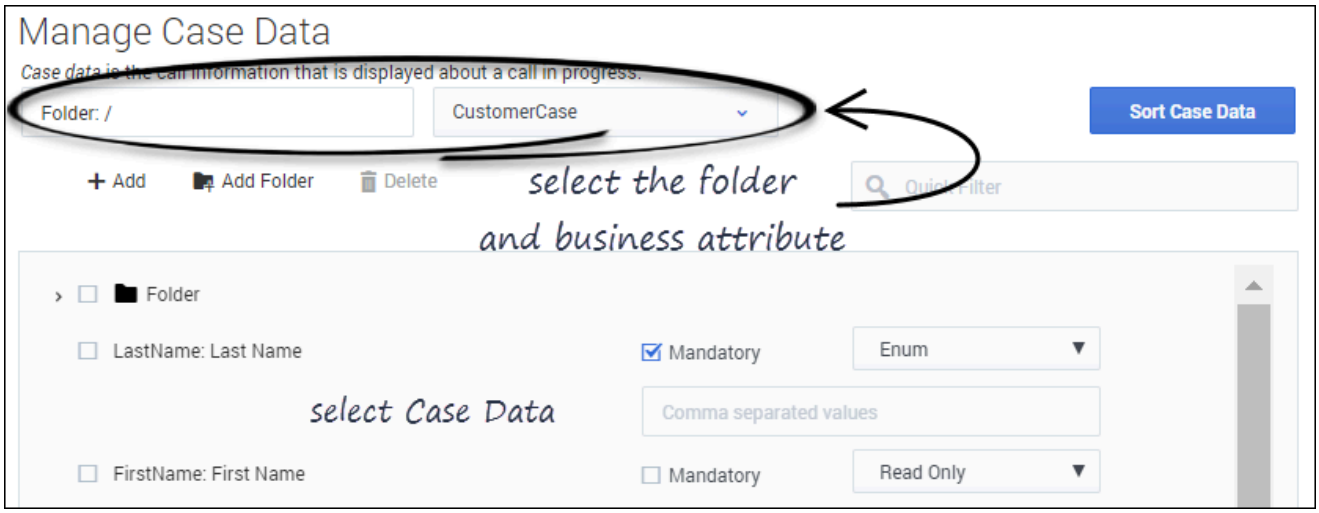
Creating a sharable Case Data business attribute

Case data can be used as sharable business attributes for agents, agent groups, and the global contact center. To view an article about how to create sharable business attributes, see Sharing Business Attributes in Agent Setup.

Tip

- To view an example of Case Data in Agent Desktop, see Contact and interaction history in *Agent Desktop Help*.
- In the table of configured Case Data, in addition to selecting and adding Case Data, you can also filter items using the **Quick Filter** field and delete a Case Data.

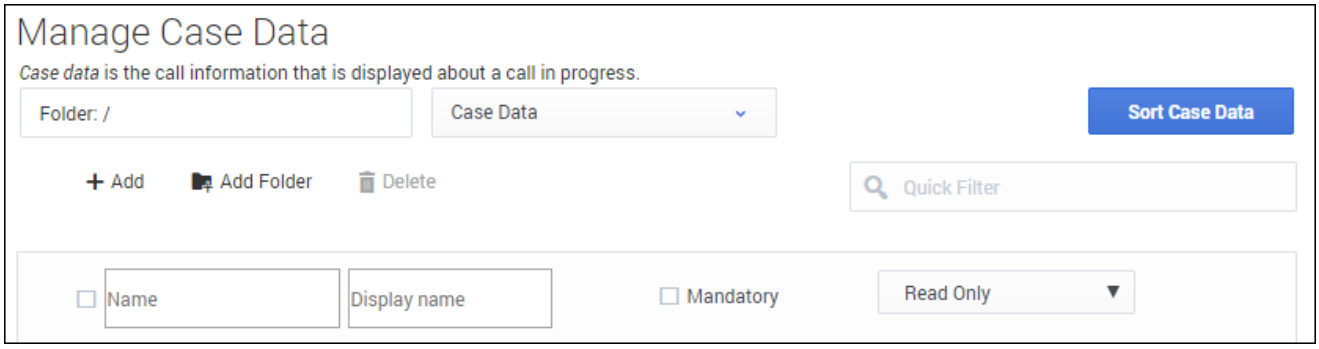
Selecting Case Data



1. Select a business attribute:
   a. Select the folder you want to view from the **Folder** drop-down list box.
   b. Click the second drop-down list box and select the business attribute for the Disposition.

      You can consider a business attribute to be the container that holds a list of Dispositions that have a common theme.
2. Select a Case Data from the list.

Adding Case Data



Select the folder you want to view from the **Folder** drop-down list box.

1. Click the second drop-down list box and select the business attribute into which you want to add the Case Data.
2. Click **Add**.
3. Specify a name and display name.
4. Select **Mandatory** to specify that the case data is mandatory.
5. Enable the **Read Only** option if you do not want the agent to be able to update the data. Or, select a data validation type, such as **Boolean**, **String**, **Integer**, or **Enum** which allows the agent to edit the data. If the case data is mandatory, the **Read Only** option is disabled.

Tip

To add a new folder to the list, you can do one of the following:

- Click **Add Folder**; or
- To add a subfolder, click the **Folder** drop-down list box, select a folder, and then click the **Add folder** icon.

To configure the order in which the case is displayed in the UI, click **Sort Case Data** and follow the instructions in the dialog box.

Toast Data

On the **Toast Data** page, you can specify the information to be displayed to the agent when they receive a call, chat, or other interaction.

Creating a sharable Toast business attribute

Toast Data can be used as sharable business attributes for agents, agents groups, and the global contact center. To view an article about how to create sharable business attributes, see Sharing Business Attributes in Agent Setup.

Tip

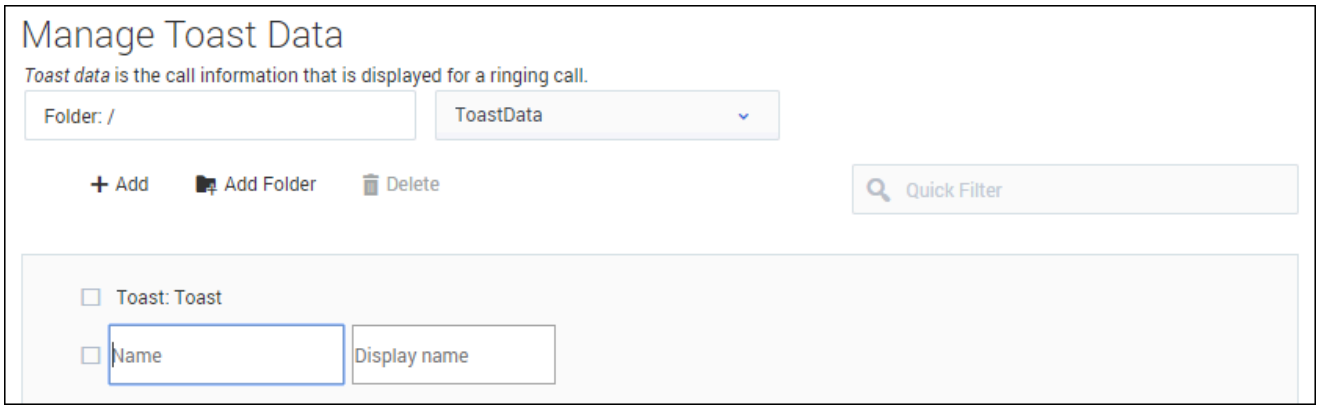
- To view an example of Toast Data in Agent Desktop, see Contact and interaction history in *Agent Desktop Help*.
- In the table of configured Toast Data, in addition to selecting and adding Toast Data, you can also filter items using the **Quick Filter** field and delete a Toast Data.

Selecting Toast Data

1. Select a business attribute:
   a. Select the folder you want to view from the **Folder** drop-down list box.
   b. Click the second drop-down list box and select the business attribute for the Disposition.

      You can consider a business attribute to be the container that holds a list of Dispositions that have a common theme.
2. Select a Toast Data from the list.

Adding Toast Data



Select the folder you want to view from the **Folder** drop-down list box.

1. Click the second drop-down list box and select the business attribute into which you want to add the Toast Data.
2. Click **Add**.
3. Specify a name and display name.

> **Tip**
>
> To add a new folder to the list, you can do one of the following:
>
> - Click **Add Folder**; or
> - To add a subfolder, click the **Folder** drop-down list box, select a folder, and then click the **Add folder** icon.

Caller ID

On the **Caller ID** page, you can create a list of Caller Identification (ID) options to be displayed when an agent makes an outbound call or transfer. A Caller ID is typically the phone number that you want displayed on the phone of the person who receives the call when the agent consults, conferences, or transfers a call.

You can also specify a display name, such as your company name, to be included with the Caller ID. For example, SalesForMore 1-555-555-5555. In this case, Agent Desktop shows **SalesForMore** on the list of Caller IDs that an agent can select.

Creating a sharable Caller ID business attribute

Caller IDs can be used as sharable business attributes for agents, agents groups, and the global contact center. To view an article about how to create sharable business attributes, see Sharing Business Attributes in Agent Setup.

Tip

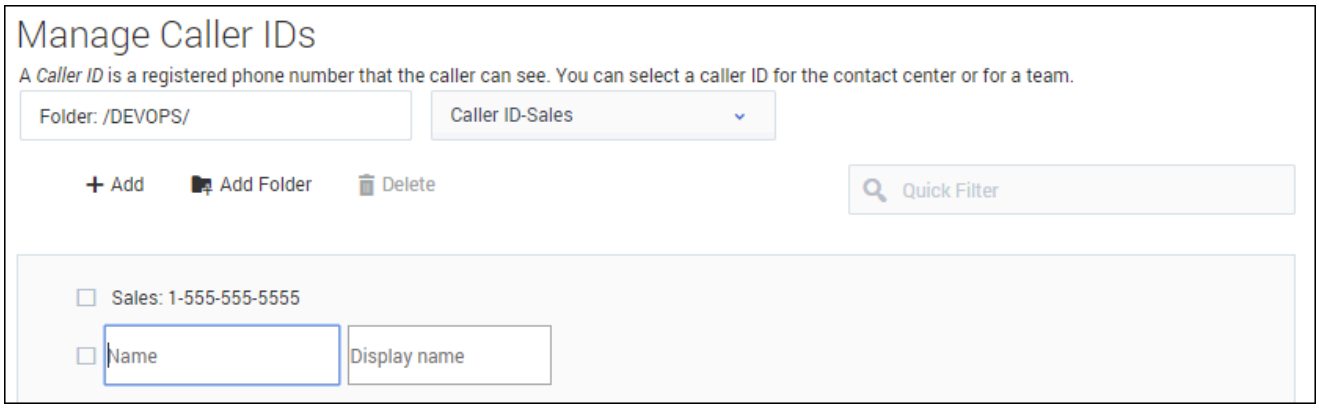
- To view an example of Caller ID options in Agent Desktop, see How do I manually select a Caller Identification? in *Agent Desktop Help*.
- In the table of configured Caller IDs, in addition to selecting and adding Caller IDs, you can also filter items using the **Quick Filter** field and delete a Caller ID.

Selecting a Caller ID

1. Select a business attribute:
   a. Select the folder you want to view from the **Folder** drop-down list box.
   b. Click the second drop-down list box and select the business attribute for the Caller ID.

      You can consider a business attribute to be the container that holds a list of Caller IDs that have a common theme.
2. Select a Caller ID from the list.

Adding a new Caller ID



1. Select the folder you want to view from the **Folder** drop-down list box.
2. Click the second drop-down list box and select the business attribute into which you want to add the Caller ID.
3. Click **Add**.
4. Type the **Name** and **Display Name** for the new Caller ID.

Tip

To add a new folder to the list, you can do one of the following:

- Click **Add Folder**; or
- To add a subfolder, click the **Folder** drop-down list box, select a folder, and then click the **Add folder** icon.

Next Steps

Now that you have configured your Agent Desktop settings, you can proceed to configure Desktop Options for your contact center. See Desktop Options.

Enable Single Sign-On

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

Single Sign-On (SSO) identity authentication enables your users to securely access multiple Genesys applications with a single credential.

After entering their username in the application login screen, users are taken to your company's authentication provider where they will enter their username and password. After that, they will not have to log in again until your authentication expires which is typically every eight hours. For a list of Genesys Engage cloud applications supporting SSO, see Single Sign-On.

You can enable Single Sign-On for your environments in the **SAML** section of Agent Setup under **Single Sign-On**. Security Assertion Markup Language (SAML) is an open standard for exchanging authentication and authorization data between parties, in particular, between an identity provider (IdP) and a service provider (SP).

> **Important**
> You do not need to enter any IdP-metadata in the "Region Name" field in order to enable SAML.

SAML fields and actions

Region Name	Base URL	Status	Actions
APS2	[redacted]	ON	
USW1	[redacted]	ON	

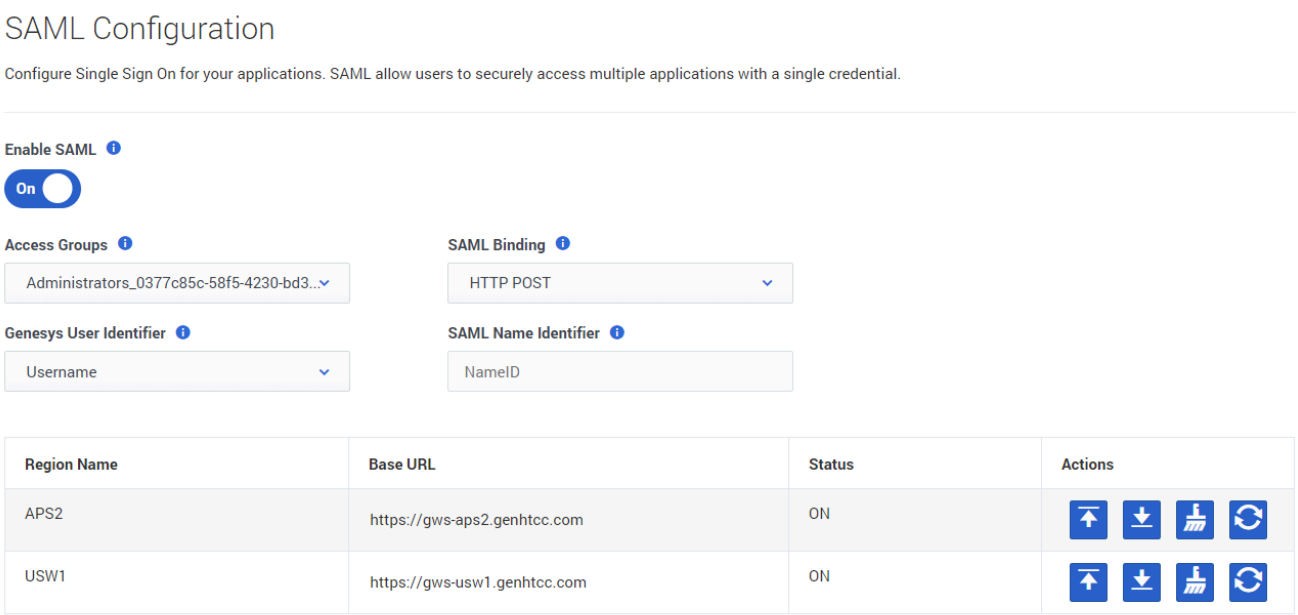
On the SAML Configuration screen, a table displays the following information:

- **Region Name** - the name of the region in which your contact center is located.
- **Base URL** - the base URL associated with the region. This field is editable - simple double click

anywhere within the text box to edit it.

- **Status** - indicates the status of configuration:
  - **ON** - the configuration is complete.
  - **OFF** - there is no configuration.
  - **PENDING** - configuration is in progress.
- **Actions** - you take any of the following actions for a particular region:
  - **Upload SAML metadata** enables you to upload your metadata;
  - **Download SAML metadata** enables you to download SP metadata for your use. This is available after your IdP metadata has been uploaded;
  - **Clear SAML metadata** enables you to clear previously uploaded metadata;
  - **Reload SAML configuration** refreshes the configuration for a specific region.

Configure SAML



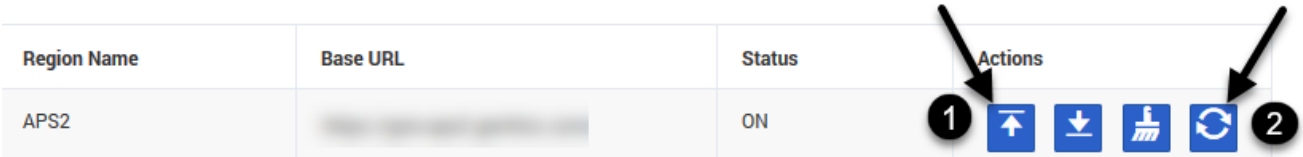
To configure SSO:

1. From the **Access Groups** list, select one or more access groups. These groups contain users who will use SSO.
2. Optional: From the **SAML Binding** list, select the SAML Binding type (HTTP POST or HTTP Redirect).
3. The next 2 fields specify how to match the user defined in your IdP with its corresponding Genesys user at the time of login. In the **Genesys User Identifier** field, select the field you wish to use as the user identifier on the Genesys side - either the Username or the External ID.

4. In the **SAML Name Identifier** field, enter the name of the attribute of your SAML assertion that contains the user identifier. This attribute is matched with the Genesys Username (or External ID). If you leave this field empty, the "NameID" attribute is used by default.
5. Set the Base URL to the region(s).
6. Upload the idP metadata to the region(s).
7. Turn the **Enable SAML** to the **On** position.
8. Click **Save**.

When SAML configuration completes, the status changes from PENDING to ON and the **Download SAML metadata** button is enabled. Note: for secondary regions, SAML configuration can take about 15 minutes.

Reconfigure SAML



If SAML is already enabled and you need to reconfigure it with new IdP metadata, do the following:

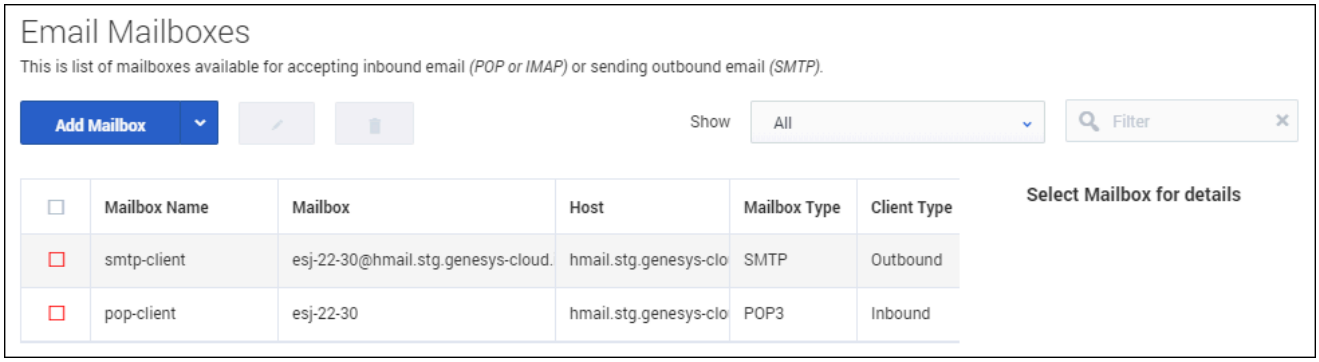
1. Upload the new IdP metadata (remember: for secondary regions, SAML configuration can take up to 15 minutes).
2. Next, you must click the **Reload SAML configuration** button.

Email Mailboxes

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

Genesys Email enables you to create and configure inbound and outbound mailboxes for your contact center through **Digital Management** in the **Contact Center Settings** section.

Mailboxes

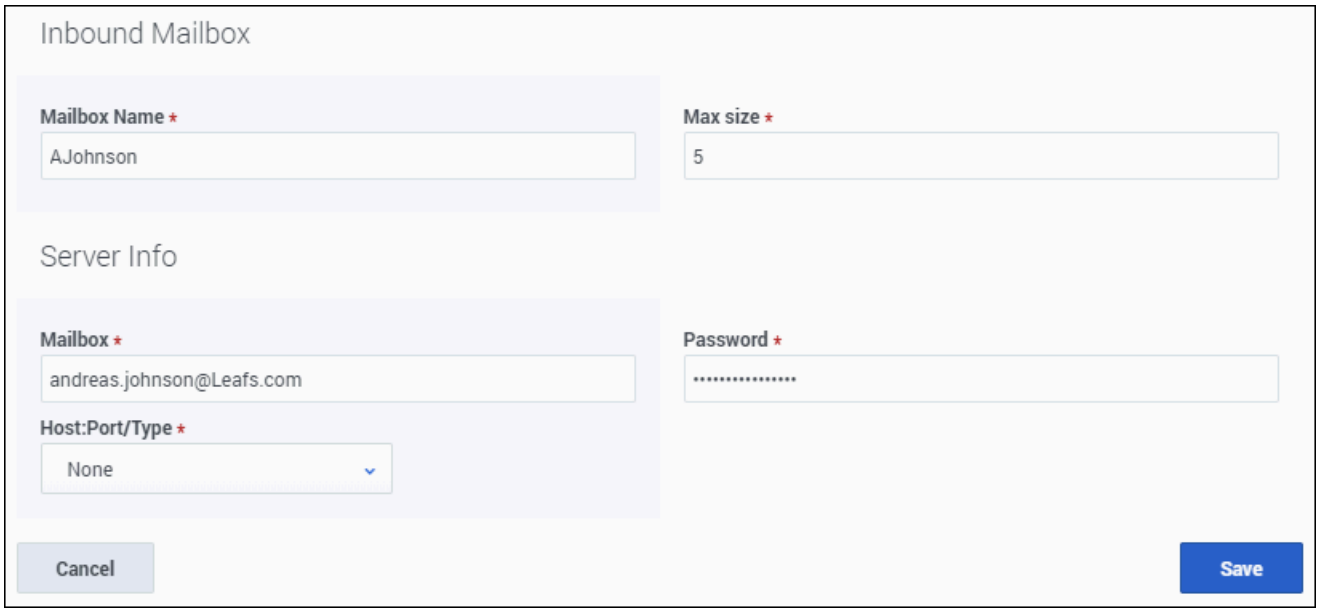


The **Email Mailboxes** sub-section displays all configured mailboxes for your contact center. The mailbox table will be empty if there are no mailboxes configured.

Select a mailbox in this view to see its details displayed on the right side of the view.

You can also filter this view by Inbound and Outbound mailboxes, as well as search for mailboxes.

Add or edit a mailbox



To create a new mailbox, select either **Inbound** or **Outbound** from the **Add Mailbox** menu. A mailbox is created in a disabled state.

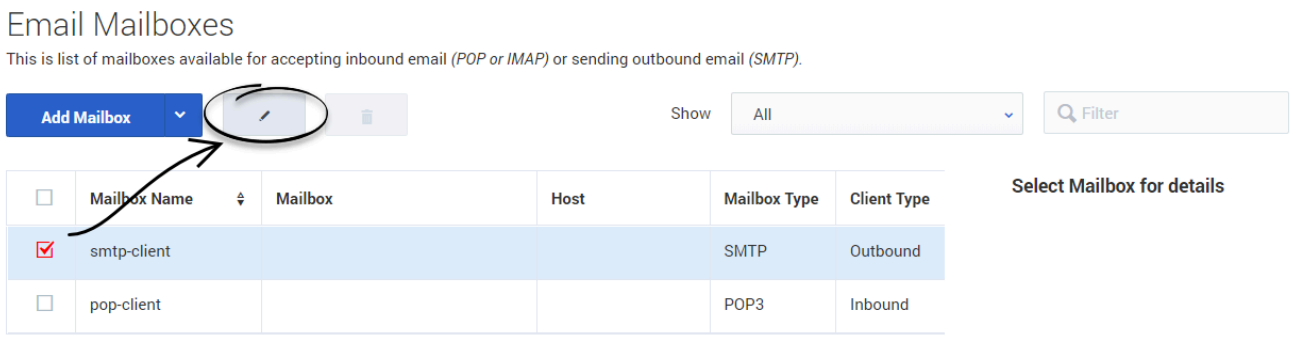
Mailboxes are enabled when they are assigned to a Digital application in Designer. If the application stream is disabled, so are the mailboxes assigned to it.

Once enabled, the selection box beside the mailbox turns red. The mailbox cannot be deleted until it is disabled.

Note: Only one outbound mailbox is permitted at this time.

To edit a mailbox, select the mailbox you wish to edit and click the **Edit** button at the top of the page.

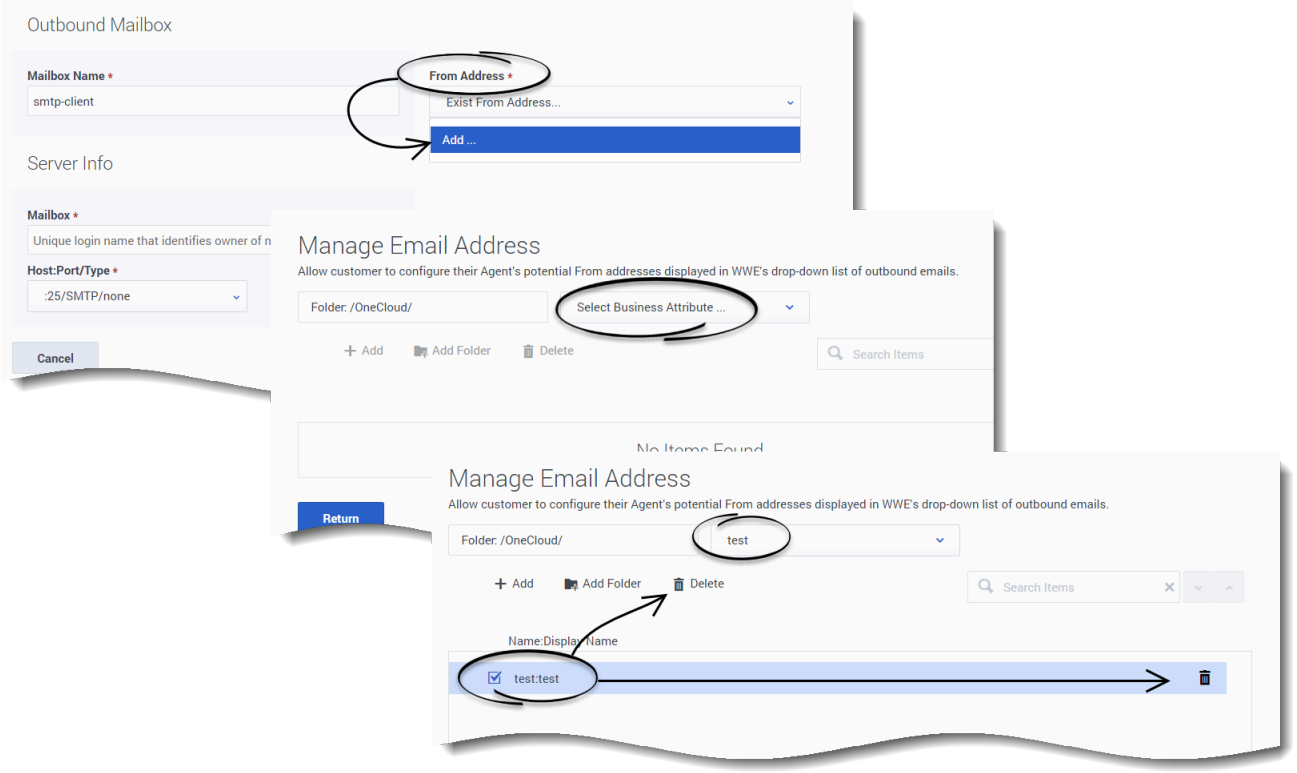
Add or delete Business Attributes



Business Attributes can be added and deleted within Agent Setup through the **From Address** field in the **Outbound Mailbox** window.

From the main **Email Mailboxes** window, select the outbound email mailbox name and click the **Edit** ✎ button.

From the drop-down list under **From Address**, select **Add...**. The **Manage Email Addresses** window opens. Here you can add a new Business Attribute, as well as delete an existing Business Attribute. *(click to expand the image)*



Use the following table for advice on how to fill in the mailbox creation/edit modal fields:

Outbound mailbox create and edit form

Label	Field type	Default values	Possible values	Action	Description
Mailbox Name	Strings: alphanumeric, no special characters, no more than 40 characters	smtp-client	N/A	A read-only field to be displayed	Outbound mailbox names are set as smtp-client
From Address	Strings: alphanumeric	N/A	N/A	Create and Edit	The email address

Label	Field type	Default values	Possible values	Action	Description
					associated with this account
Mailbox	Strings: alphanumeric	N/A	N/A	Create and Edit	The name used to log in to the corporate email server
Password	Strings: alphanumeric	N/A	N/A	Create and Edit	The password associated with this account
Host/Port/Type	Drop-down list	N/A	SMTP	Create and Edit	The name of the corporate SMTP server

Inbound mailbox create and edit form

Label	Field type	Default values	Possible values	Action	Description
Mailbox Name	Strings: alphanumeric, no special characters, no more than 40 characters	N/A	N/A	Create only, Edit field is read-only	Any valid login name associated with a POP/IMAP account
Max size	Numeric: minimum = 5, maximum = 20	5	Range between 5 and 20	Create and Edit	The maximum size (in MB) of an incoming message
Mailbox	Strings: alphanumeric	N/A	N/A	Create and Edit	The login name associated with the POP/IMAP account
Password	Strings: alphanumeric	N/A	N/A	Create and Edit **Note:** The user must type in a password.	The password associated with this account
Host/Port/Type	Drop-down list	N/A	POP3, IMAP	Select only	Parameters of the corporate email server where the account resides

Desktop Statistics

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

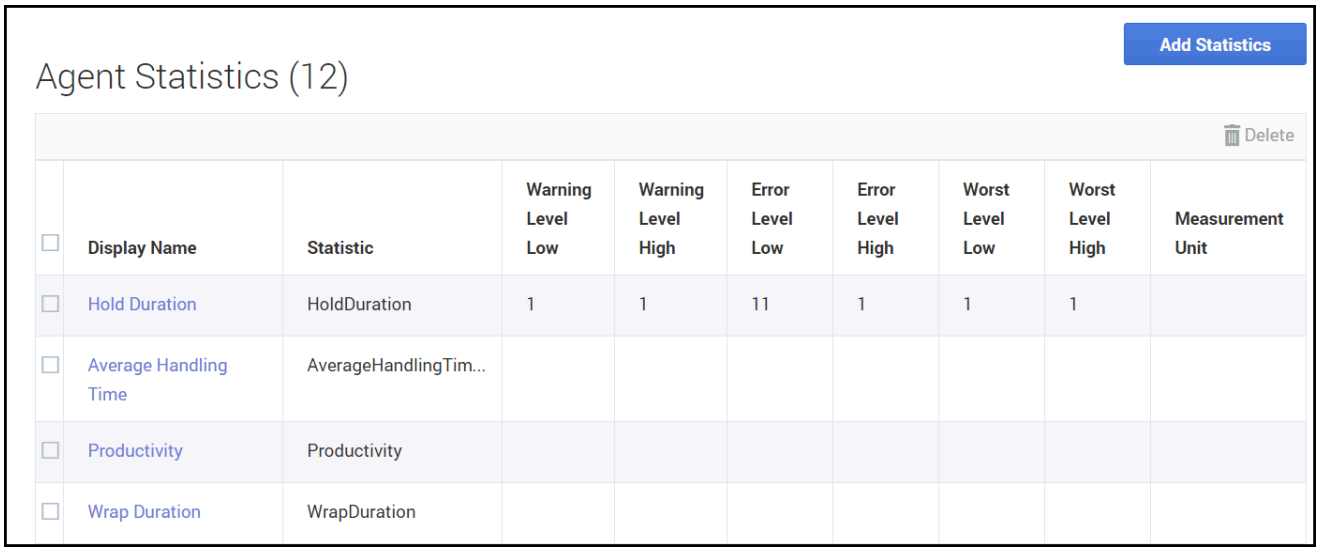
The **Desktop Statistics** section allows you to add, edit, import and export statistics to be displayed under **My Reports** and **Contact Center** panes of the Agent Desktop dashboard or the **Reports** view in Gplus Adapter. The **Statistics** tab includes three sub-tabs to help you manage your statistics:

- Agent Statistics
- Contact Center Statistics
- Statistic Definitions

Essentially, you add statistics to a global statistics list. Then from that list, you can select the statistics that you want to apply. Statistics can be re-used, meaning that if you create a statistic for a specific agent group, for example, you can then apply that favorite to another agent group.

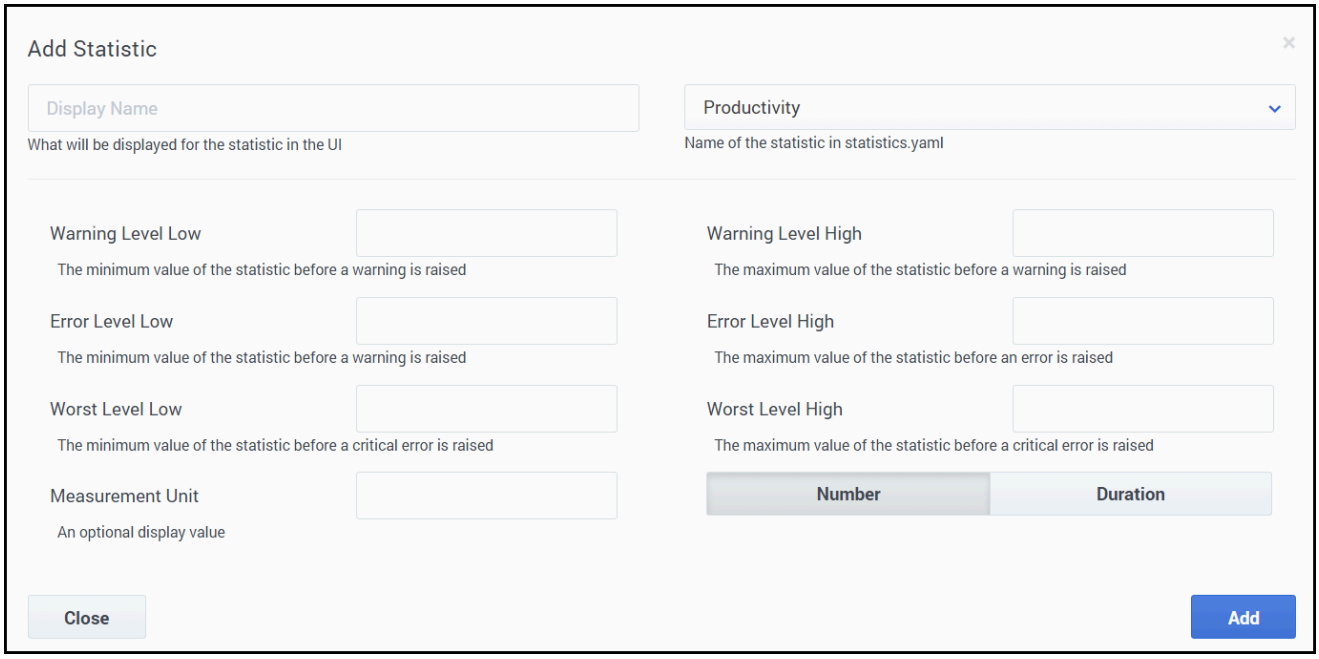
Agent Statistics

On **Agent Statistics** tab, manage the statistics that are displayed under **My Reports** to the agent. The **My Reports** page gives agents a view of the current state of their own activity.



To access the **Agent Statistics** tab, click **Contact Center Settings > Agent Statistics**. The list on the page includes all agent statistics that are already configured, either by you or another administrator. For a new implementation, the list might be blank.

To add a shared statistic to the list, click **Add Statistics** to access a list of shared statistics. In the window that opens, select an option from the list of statistics, and then click **Add**.



If you don't see the statistic you want, you can create a new statistic for the global list and then add that statistic for the **Agent Statistics** list.

1. Click **Add Statistics**.
2. In the window that opens, click **Add** to create a new statistic.
3. In the **Add Statistic** window, configure the attributes.
4. Click **Add**.
   The statistic is now listed in the list on the **Add Statistic** window. This means that the statistic is now available in the global list, but is not yet assigned.
5. To assign the new statistic to **Agent Statistics** list, select the statistic and click **Add**.
   The window closes and the new statistic is assigned to the **Agent Statistics** list.

[+] Show a list of Agent Statistic attributes

You can set the following attributes for each statistic that you specify to be displayed to agents.

- **Display Name**: The name of the statistic to be displayed.
- **Name of the statistic in the statistics.yaml file**: This must correspond to the value of the `name` attribute in the **statistics.yaml** file.
- **Warning Level Low**: The minimum value of the statistic before a warning is raised. No warnings above this value.
- **Warning Level High**: The maximum value of the statistic before a warning is raised. No warnings below this value.
- **Error Level Low**: The minimum value of the statistic before an error is raised. No errors above this value.
- **Error Level High**: The maximum value of the statistic before an error is raised. No errors below this

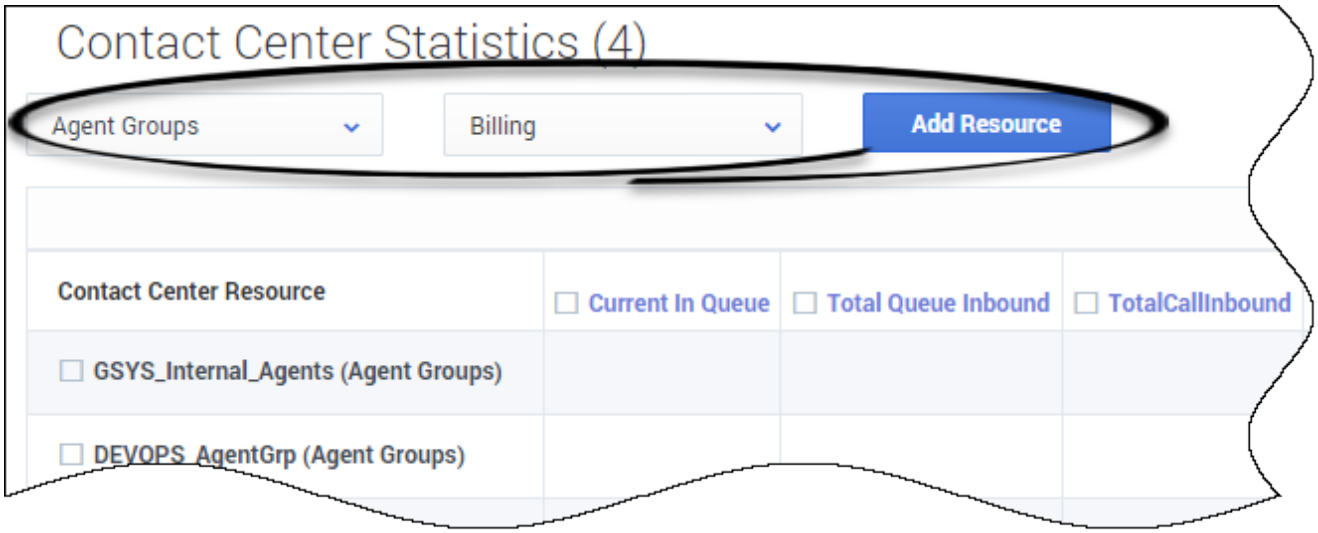
value.

- **Worst Value Low**: The minimum value of the statistic before a critical error is raised.
- **Worst Value High**: The maximum value of the statistic before a critical error is raised.
- **Measurement Unit**: an optional display value.
- **Number**: Indicates the statistic type is numeric.
- **Duration**: Indicates the statistic type is duration (in HH:MM:SS format).

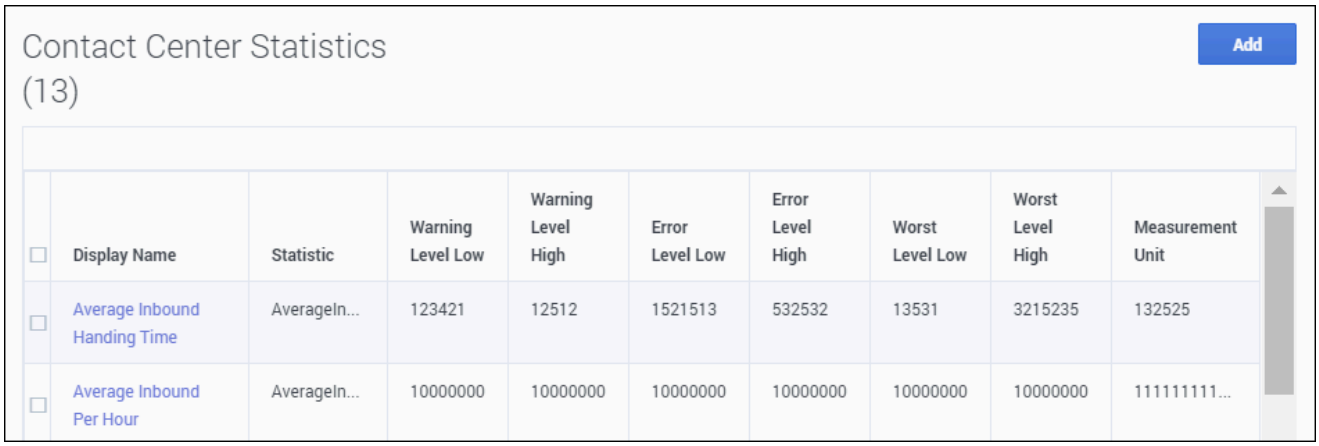
Contact Center Statistics

On the **Contact Center Statistics** tab, manage the statistics that are displayed under **Contact Center**. The **Contact Center** tab gives agents a view of the current state of activity for the call center.

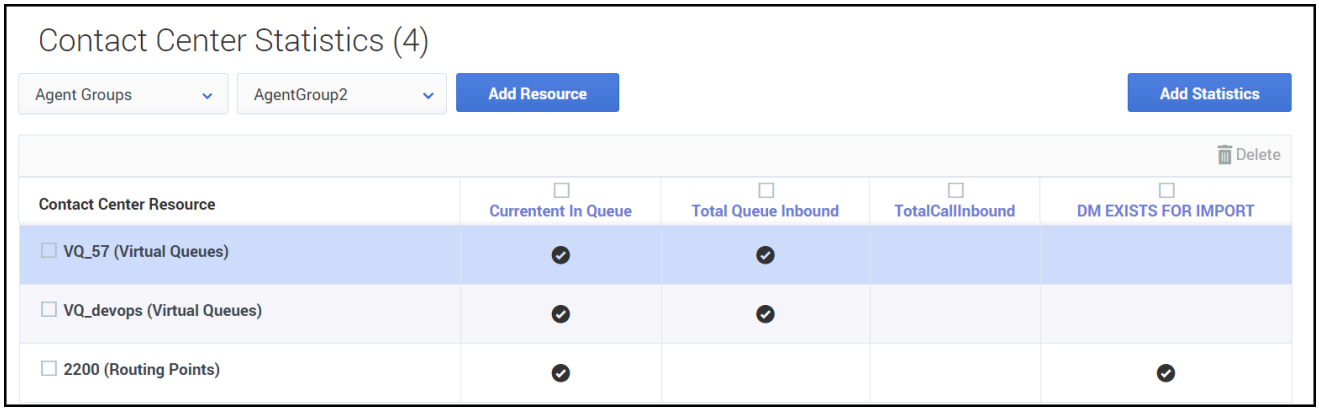
To access the **Contact Center Statistics** tab, click **Contact Center > Contact Center Statistics**.



To configure Contact Center Statistics, you must first add a resource. Resources can include agent groups, queues, DN groups, and routing points. In the first drop-down list box, select the type of resource you want to add, and then select the resource from the second drop-down list box. Click **Add Resource**. This action adds a row for the resource to the **Contact Center Statistics** table.



Then you can add statistics by clicking **Add Statistics**. The **Add Statistics** window opens, where you can either select a preconfigured statistic or create a new one. For details on how to select or add statistics, see Agent Statistics. This action adds a column to the **Contact Center Statistics** table. Continue adding the statistics that you want to add to the table.



When you are done, notice the black checkmarks in the **Contact Center Statistics** table. The black checkmarks indicate which statistics apply to each resource. For example, the **current number in queue** statistic does not apply to agent groups. Agent desktops shows only statistics with check marks; other desktops show only statistics with hyphens (-).

Important

Agents can only see Route Points on switches they are given permission to view. For example: Your Contact Center Statistics are configured to monitor the following routing points: RP1, RP2, and RP3. RP1 belongs to one switch while RP2 and RP3 belong to another switch. An agent who is only configured to log into the switch that contains RP1 can only see RP1 in the Contact Center Statistics. RP2 and RP3 are not visible to the agent because they belong to a switch that agent doesn't have

> permission to access.

Statistic definitions

Using the **Statistic Definitions** tab, you can import, export, and edit statistic definitions. The supported formats for the statistic definition files are: .CSV and .YAML.

About CSV files

The .CSV file must be a text file in a comma-separated format. In the source file each line represents a single statistic.

About YAML files

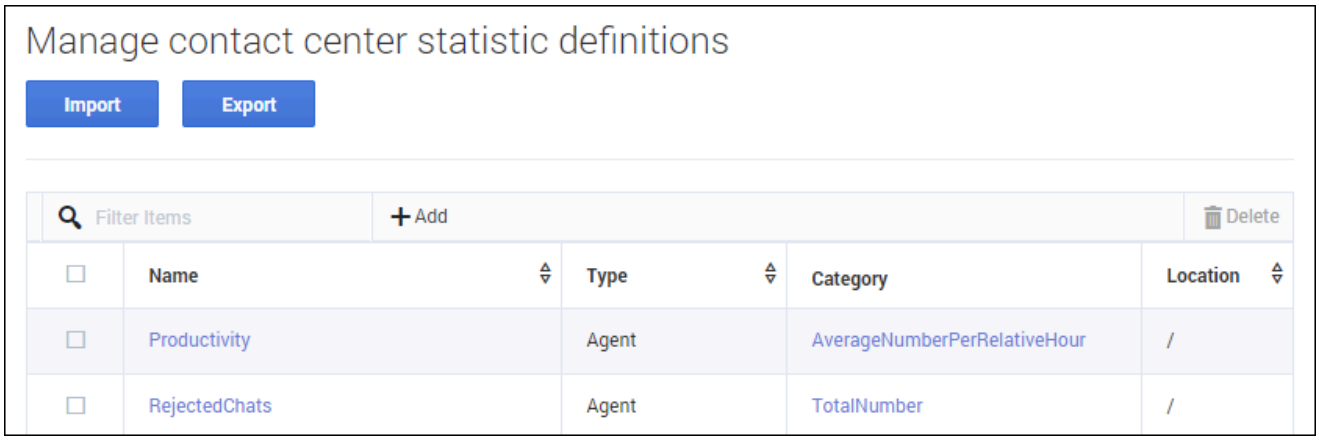
Use a YAML file to import custom statistics from a valid .YAML file. The following example shows one custom statistic from a sample .YAML file:

[+] Show an example

```
name: ReadyDuration

 objectType: AGENT
 notificationMode: PERIODICAL
 notificationFrequency: 10
 statisticDefinitionEx:
   category: TotalAdjustedTime
   intervalType: GrowingWindow
   mainMask: WaitForNextCall
   subject: DNStatus
   dynamicTimeProfile: '8:00'
   dynamicFilter: MediaType=voice
 saveHistory: false
 insensitivity: 1
 location: /
```

How do I import custom statistic definitions?



To import a .CSV or .YAML file that contains your custom statistic definitions, click **Import**, select your file, and then click **Open**. User Setup validates the file and indicates which, if any, values need to be corrected. If you receive such a message, make the recommended corrections and then try again.

How do I export statistic definitions?

You can export your statistic definitions to an external .CSV or .YAML file. You can select your custom statistics, global statistics, or both. Then you can choose which file type to export. The exported file downloads to your local computer.

Agent Groups

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

An *Agent Group* is a logical grouping of agents. Agent Groups are typically set up to provide particular sets of contact-center services.

Prerequisites

Before you can configure Agent Groups:

- You need to configure skills that will assign to the Agent Groups. If you haven't done so already, go to the Skills tab and set up the skills you want to use.
- If you plan on creating Agent Groups by selecting specific agent accounts to the group, you first need to add the agent accounts. If you want to set up Virtual Agent Groups, it is not necessary to add the agent accounts first.

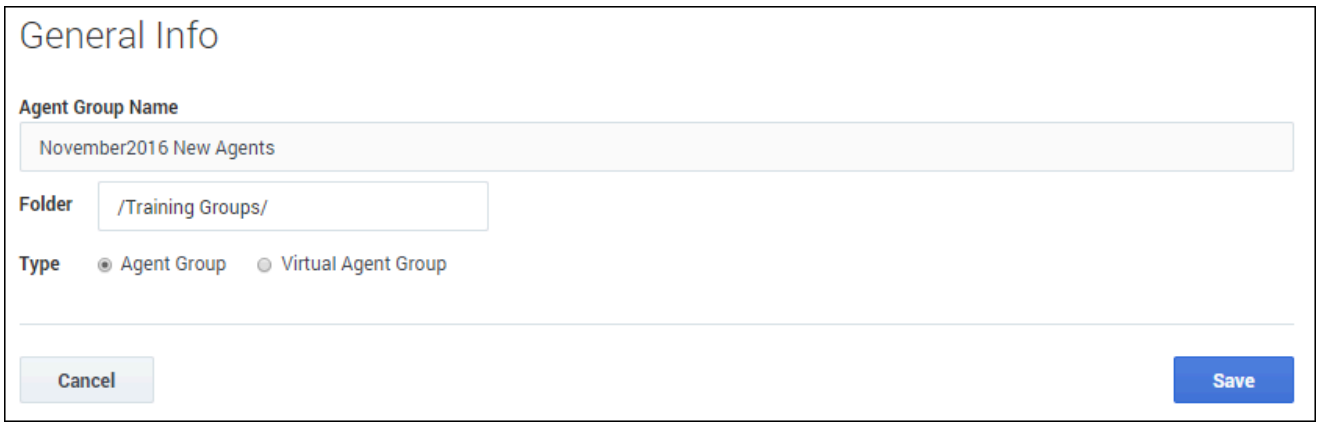
Configuring Agent Groups

You have two options for creating Agent Groups:

- Create an **Agent Group** and then manually add agents to the group. For example, if you want to create an Agent Group for a group of new hires, you create the Agent Group, and then manually add each of the new hire's agent accounts to the group.
- Create a **Virtual Agent Group** to automatically group agents, according to their skills. You create an Agent Group object and define a script to identify the skills that you want the agents in the group to share. For example, you can create a virtual Agent Group for all agents that can speak Russian.

Both types of Agent Groups are configurable on the **Agent Groups** page. Use the following procedures to create and configure your Agent Groups:

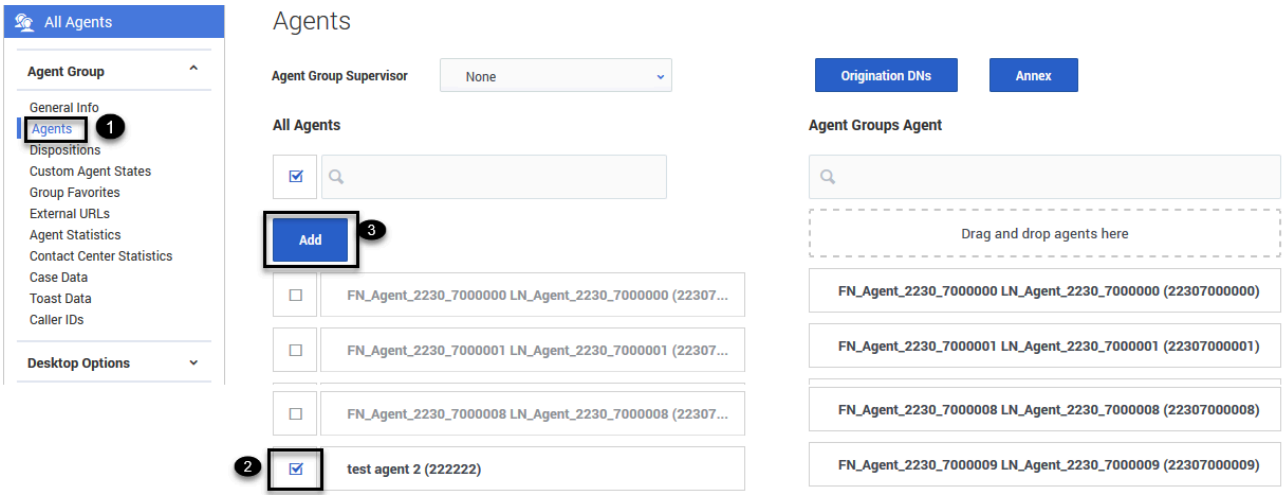
Creating an Agent Group



In the **New Agent Group** window, type the name you want to assign to the Agent Group and click **Save**. In our example, the new Agent Group is a group of new hires, so let's name the Agent Group `November2016 New Agents`.

You can put the new Agent Group into a folder. This is useful for organizing your Agent Groups to make them easy to locate in the future. For example, you can place our new group into the **Training Groups** folder. When you select the **Folder** option, a drop-down list box is displayed where you can select a folder or click the add folder icon to create a new folder.

Adding agents to an Agent Group

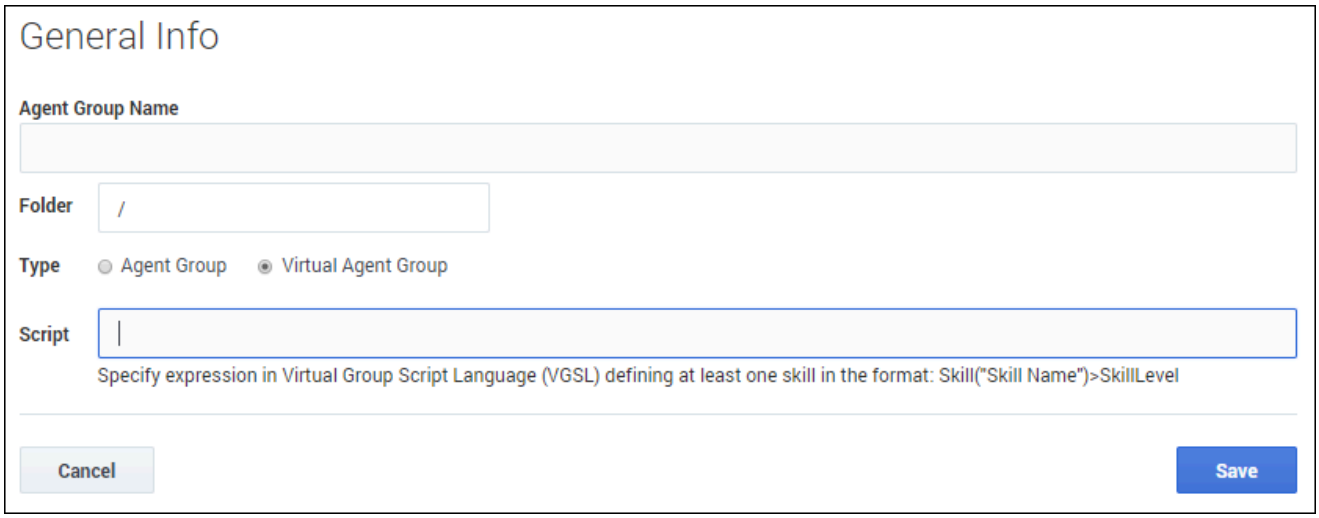


When you create an Agent Group, you can go to the Agents tab and add agents to the group.

You can either drag the agent over to the Agent Groups Agent column, or you can click the box next to the agent's name in the All Agents list and then click Add. This automatically drops the agent into the Agent Groups Agent list.

> **Tip**
> On your Agent Groups list, you might see Agent Groups that have an **Add script** link in the script column. If you see this, it means that you or someone else created an Agent Group and no agents have been assigned to it. In other words, it's an empty group. You have the option to edit the group to manually add agents or click **Add script** to create a virtual Agent Group.

Creating a virtual Agent Group

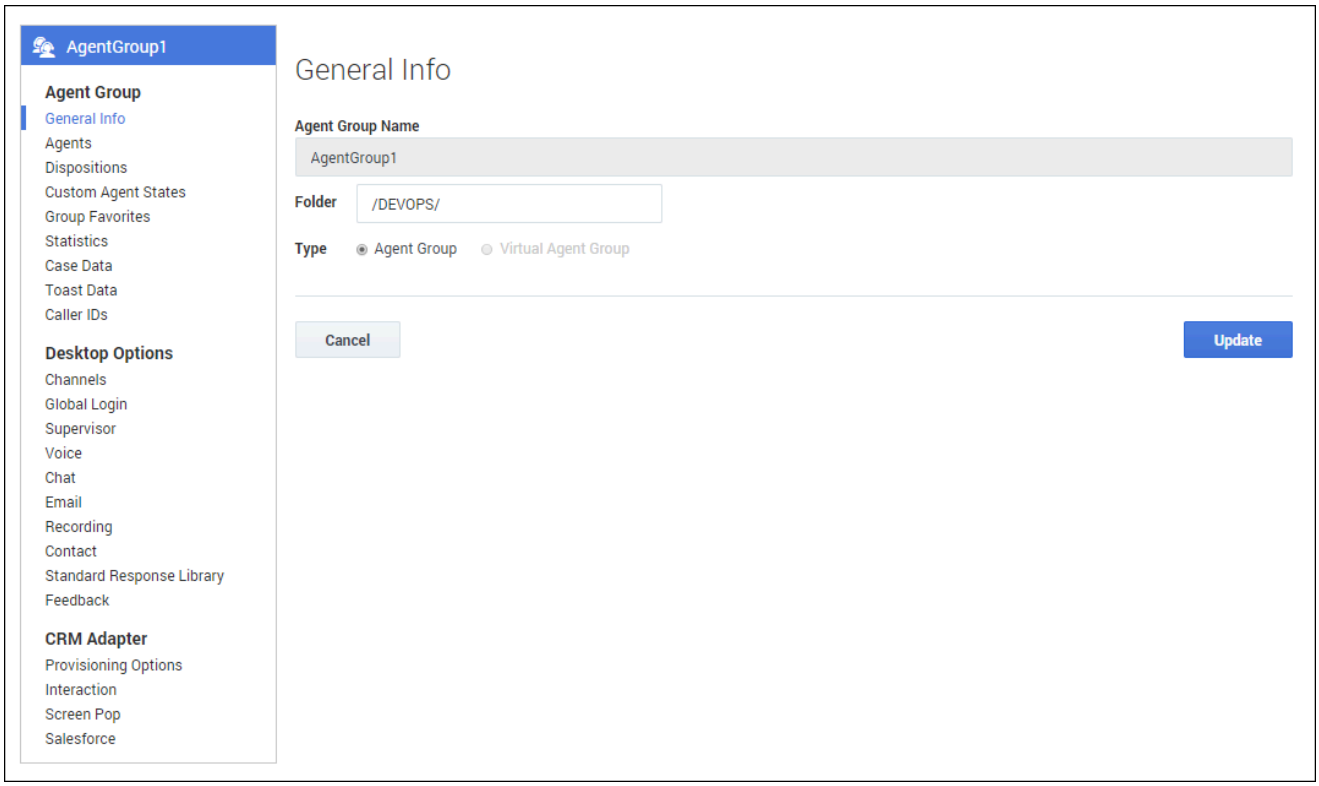


You can create a virtual Agent Group by specifying a script when you add an Agent Group.

As opposed to a regular Agent Group, you can't manually add agents to a virtual Agent Group. Agents are automatically included in the group if the agent is assigned the skill that is specified in the script.

After you add the script, you will see a link in the **Script** column on the **Agent Groups** page. The script identifies the common skills shared by all agents that belong to the group. You can edit the script by double-clicking the script text and then updating the script in the dialog box that is displayed.

Configuring the desktop for an Agent Group



After you populate your Agent Group with agents, you can assign objects to the group. Under each of the following tabs, make your selections:

- **Dispositions**: See Manage Dispositions.
- **Custom Agent States**: See Manage Agent States.
- **Group Favorites**: Group favorites are configured the same way as Global Favorites. See Global Favorites.
- **External URLs**: See External URLs.
- **Agent Statistics**: See Statistics.
- **Case Data**: See Manage Case and Toast Data
- **Toast Data**: See Manage Case and Toast Data.
- **Caller IDs**: See Manage Caller ID.
- **Desktop Options**: Use the **Desktop Options** tabs to configure the desktop settings for the Agent Group. For information, see Desktop Options.
- **CRM Adapter**: Use the **CRM Adapter** tabs to configure the Gplus Adapter settings for the Agent Group. For information, see CRM Adapter options.

Next Steps

After you have configured Agent Groups, you can proceed to configure:

- Business attribute overrides (Transactions)
- Templates

Transactions

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

Use the **Transactions** page to manage business attribute overrides that are applied at a transactional level, such as a call or a chat. You can configure the following business attributes at the transaction level:

- Dispositions
- Favorites
- Case Data
- Toast Data
- Caller IDs

Select **Transactions** in the navigation menu to access the **Transactions** page. The **Transactions** main view lists the transactions only by name. Click the link for the transaction to view the details.

Use the following procedures to create and configure transactions:

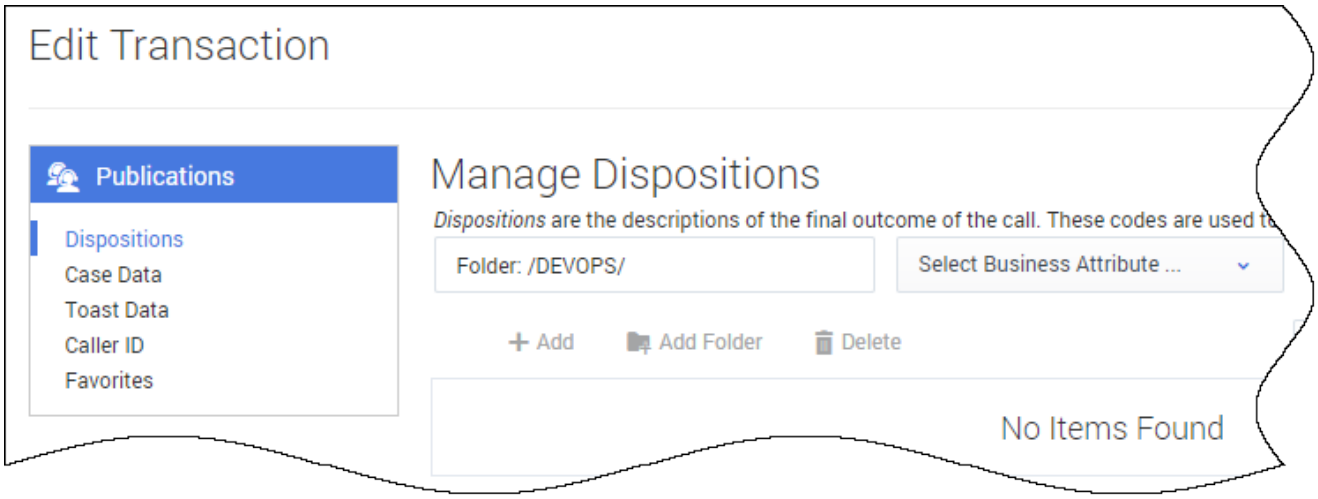
Adding a new transaction

Click the **New Transaction** button, type a unique name in the **Transaction name** field, select a folder to store the transaction configuration, and then click **Add**. If you do not select a folder, the transaction is stored in the `/default` directory.

The name of the transaction is prefixed with the name of the folder. For example, `UserPreference/<TransactionName>`. At this point, configuration is not complete. You need to add business attributes to the transaction.

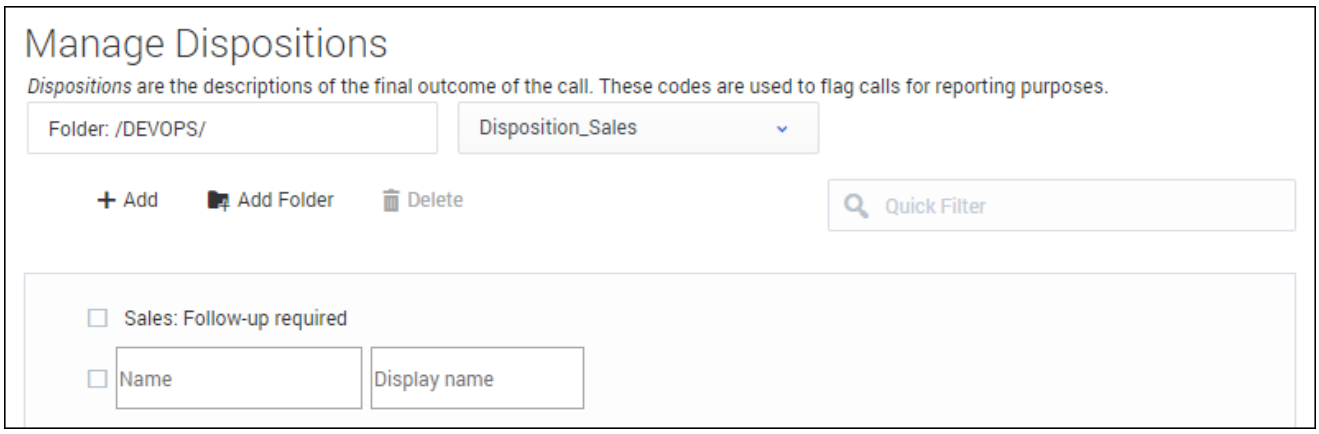
> **Tip**
>
> You can manage folders from this page. When you select the **Folder** option, a drop-down list box is displayed where you can select a folder or click the add folder icon to create a new folder.

Adding business attributes to a transaction



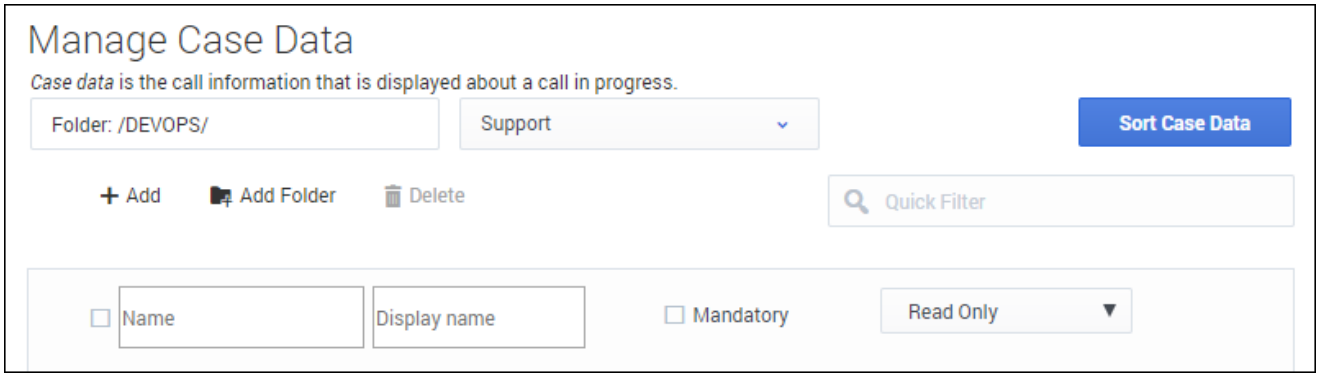
To edit a transaction, click the link in the **Name** column. If the transaction has no configured business attributes, as is the case when you add a new transaction, you are prompted to complete the configuration. Follow the prompts to add **Dispositions**, **Case Data**, **Toast Data**, **Caller ID** and **Favorites** business attributes. After you add the business attributes, you can define them.

Managing dispositions



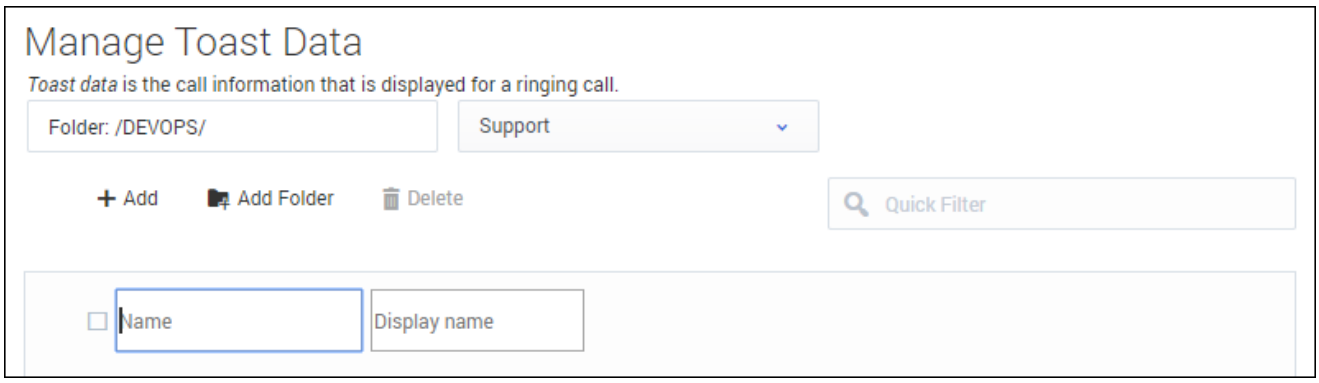
Dispositions are the descriptions of the final outcome of the call. These codes are used to flag calls for reporting purposes. To learn more about how to manage dispositions, see Managing dispositions.

Managing case data



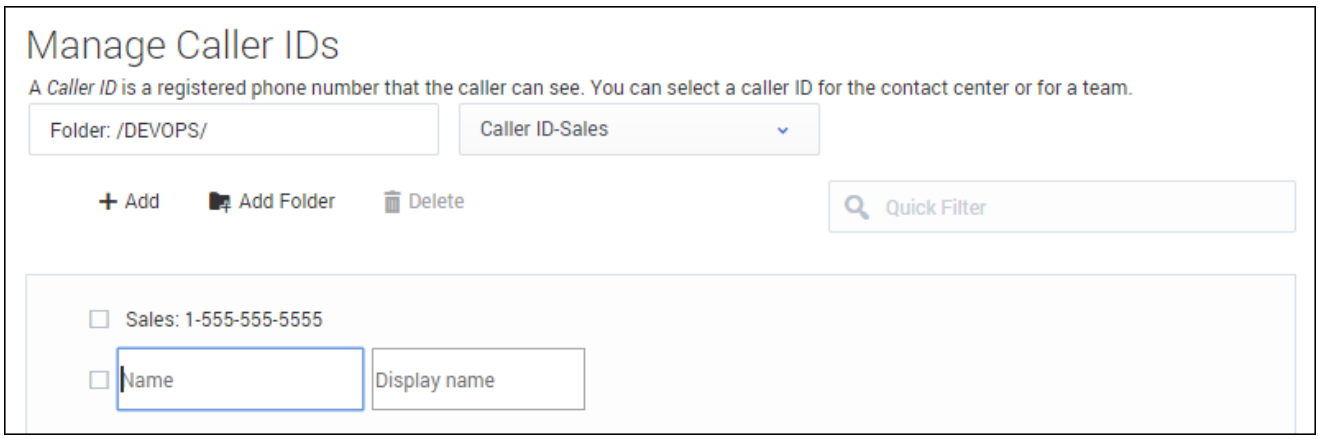
On the Manage Case Data screen, you can specify the information to be displayed to the agent when an interaction is transferred to them. To learn more about how to manage case data, see Managing case data.

Managing toast data



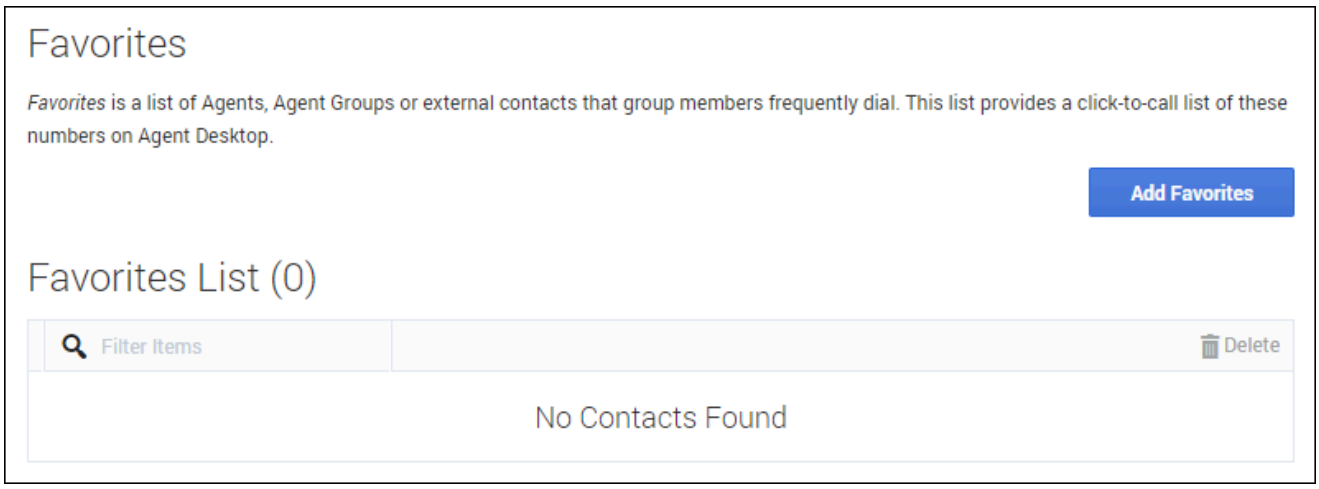
On the Manage Case and Toast Data screen, you can specify the information to be displayed to the agent when they receive a call, chat, or other interaction. To learn more about how to manage toast data, see Managing toast data.

Managing caller IDs



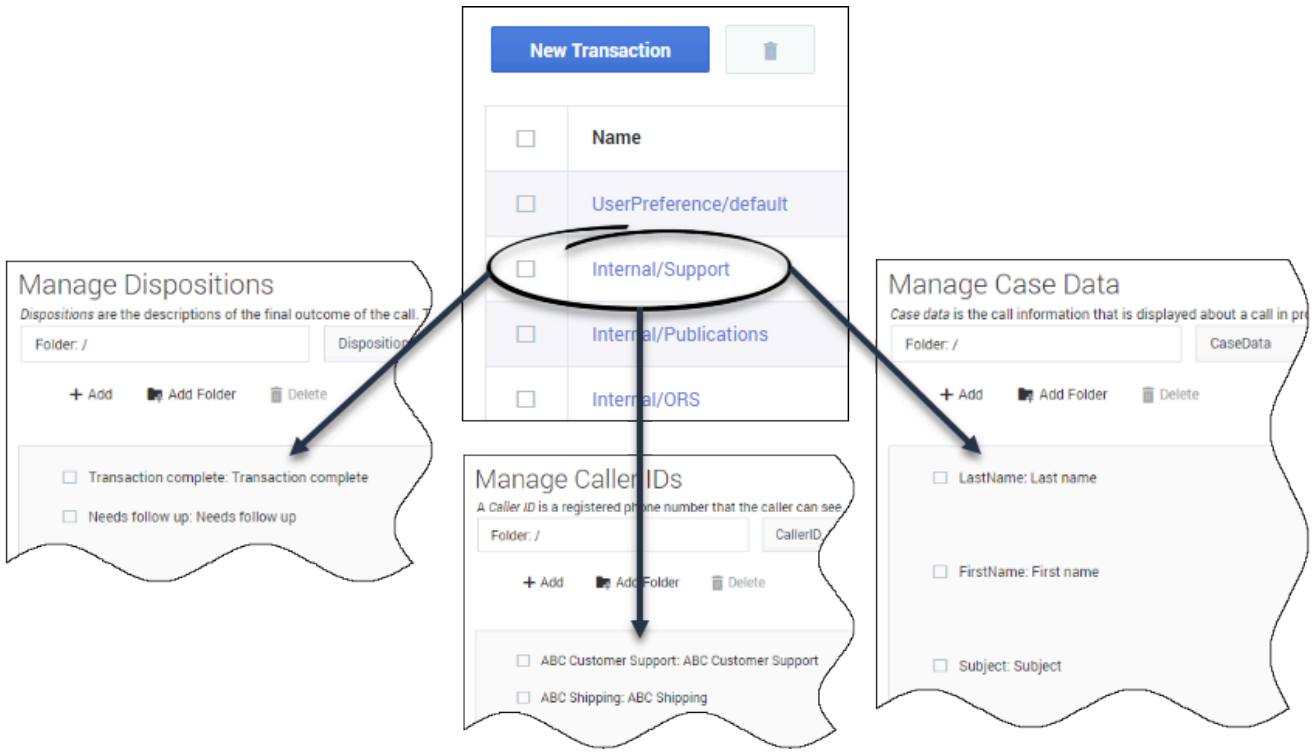
You can allow agents to select from a predefined list of caller IDs. The selected caller ID is then displayed to the person receiving the call when the agent consults, conferences, or transfers a call. To learn more about how to manage caller IDs, see Managing caller IDs

Managing favorites



On the **Favorites** screen, you can create a list of agents, agent groups or external contacts that agent group members frequently dial. This list provides a click-to-call list of these numbers on Agent Desktop. To learn more about how to manage favorites, see Global Favorites.

What should a configured transaction look like?



In this example, the transaction includes the following business attributes:

Dispositions:

- Transaction complete
- Needs follow-up

Case Data

- First name
- Last name
- Subject

Caller IDs

- ABC Customer Support
- ABC Shipping

Use Case

In Agent Desktop, you have the ability to override desktop options by setting the value of a KVP to the name of a transaction list object. The desktop option is `interaction.override-option-`key that is configured in Channels and Options. In the inbound or outbound routing application, you set the value of the identified key to the name of the transaction list object and the desktop options in the transaction list object are used instead of the Agent Desktop application or agent group objects.

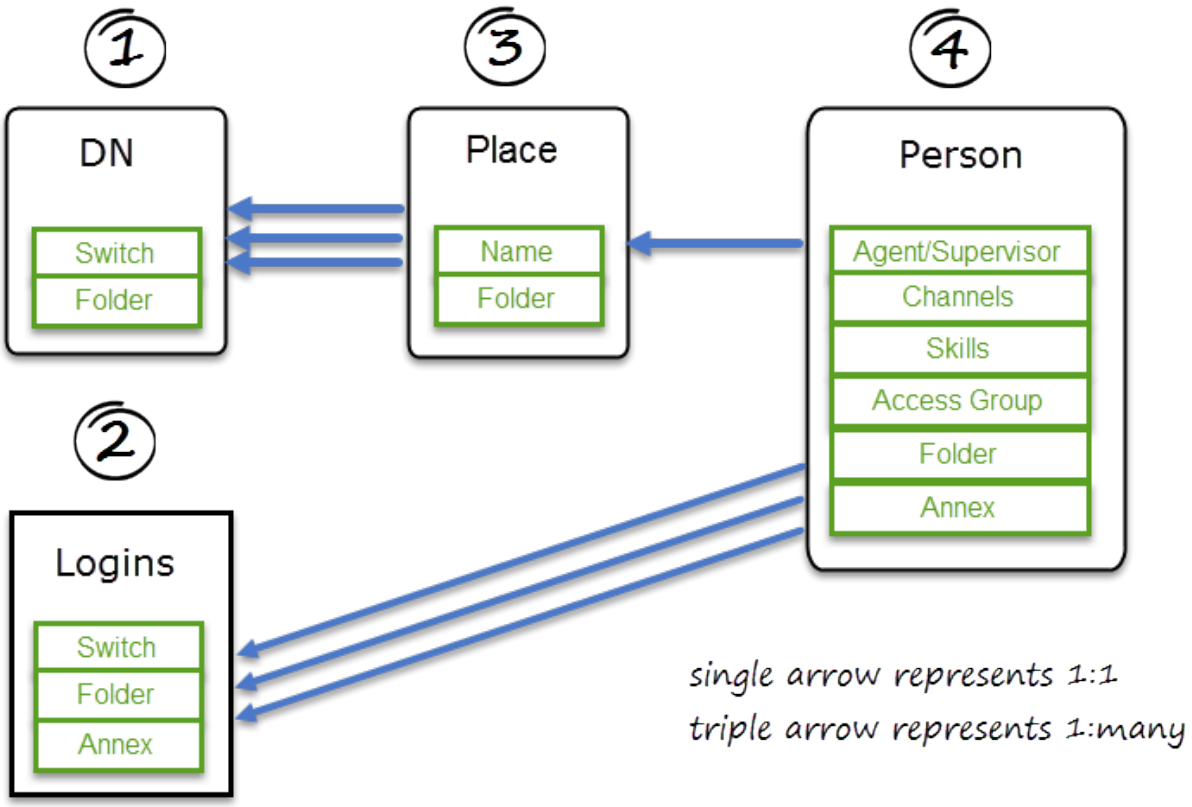
Next Steps

Now that you have configuration business attribute overrides, you can proceed to configure templates. See Templates.

Templates

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

Templates allow you to create a set of configuration options that you can apply to agent accounts when you add an agent. The following diagram shows the relationship between the templates and the sequence in which the templates must be created:

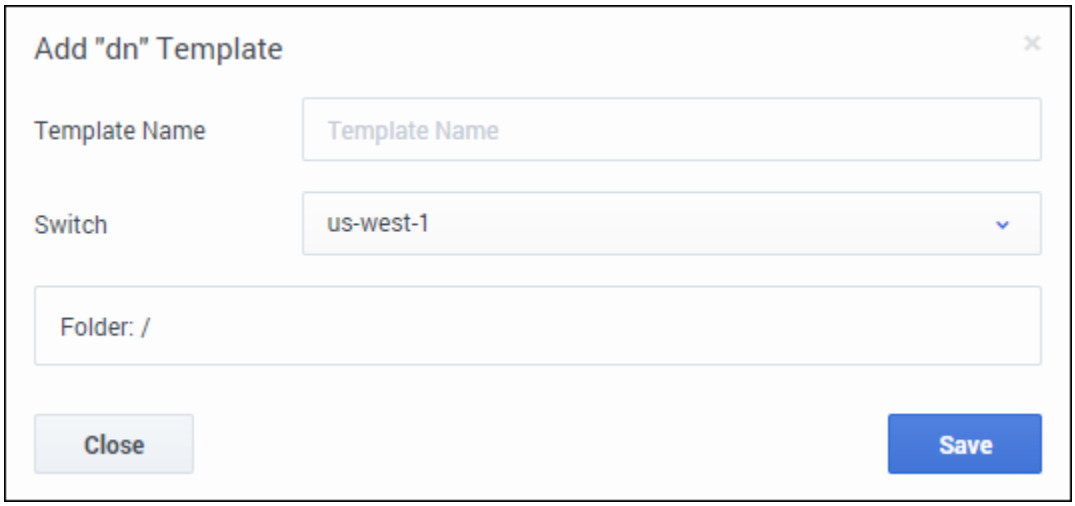


Access the **Templates** tabs on the **Contact Center Settings** page.

To add a template, select a template type, and then click **Add**. For each template you create, ensure that you give the template a unique name.

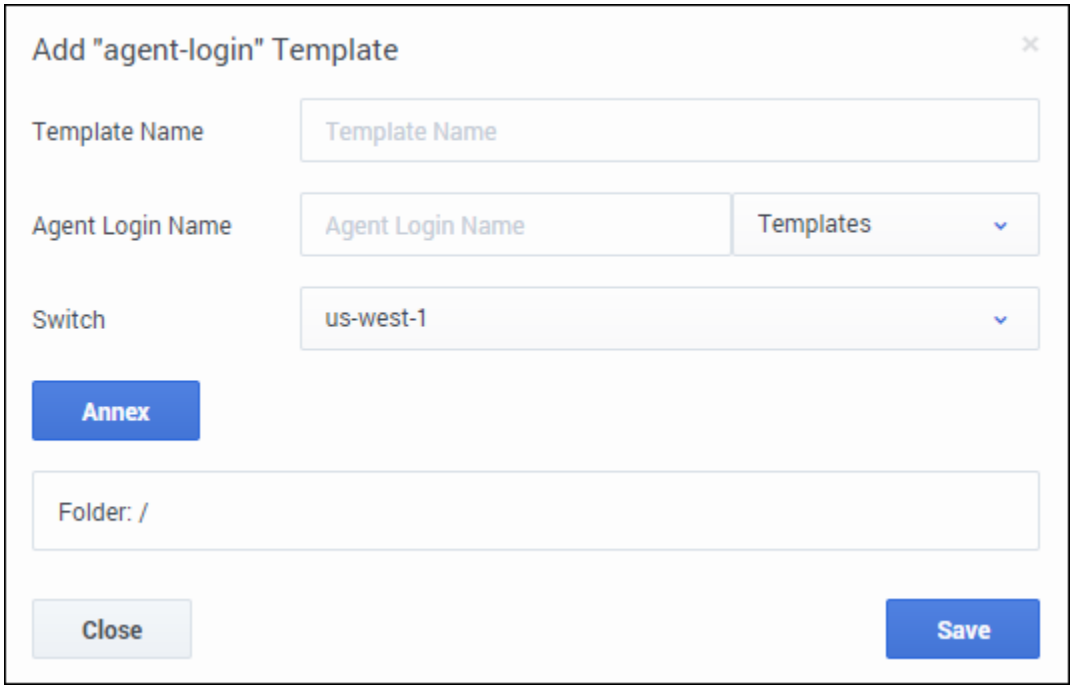
To configure a specific template, click on the **Template Name** and make your intended changes.

DN template



DN Templates specify the switch and folder where the DN needs to be created.

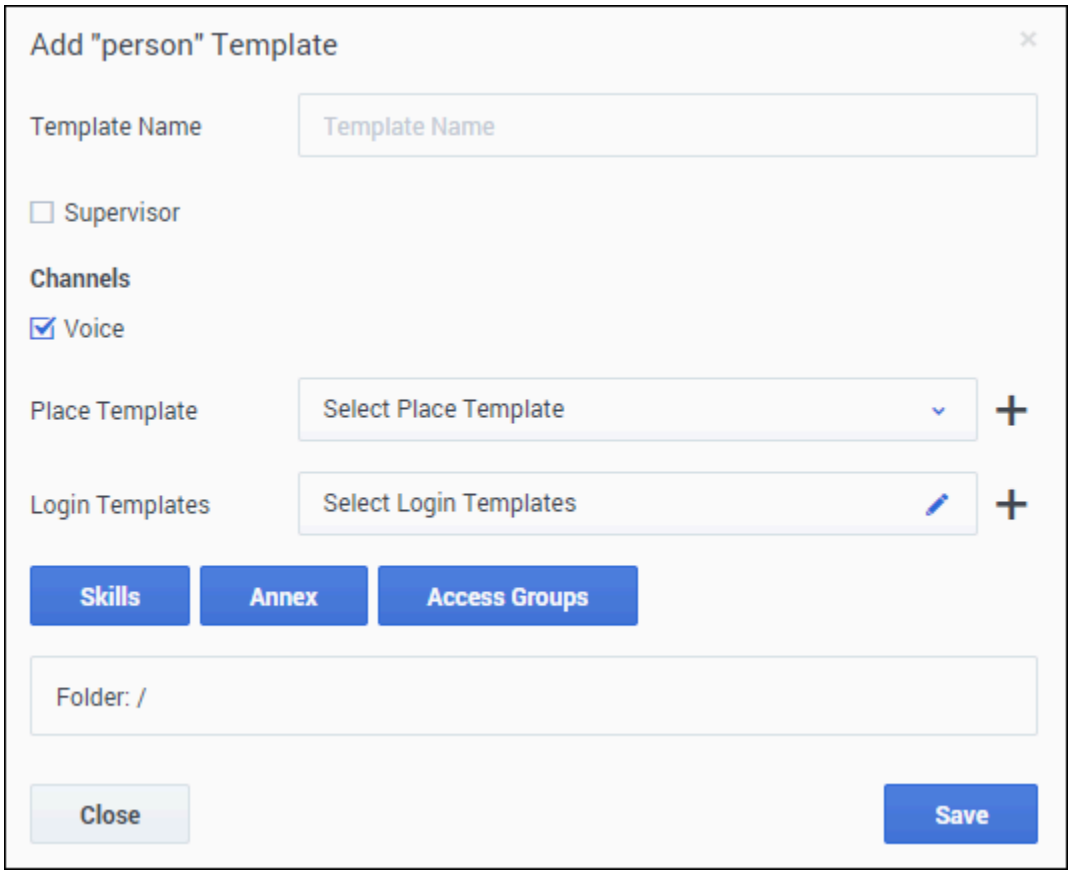
Login template



Agent Login Templates specify the switch, folder, and annex options.

In the **Agent Login Name** field, you can use a variable to specify the contents of the field, for example, {{person.userName}}.

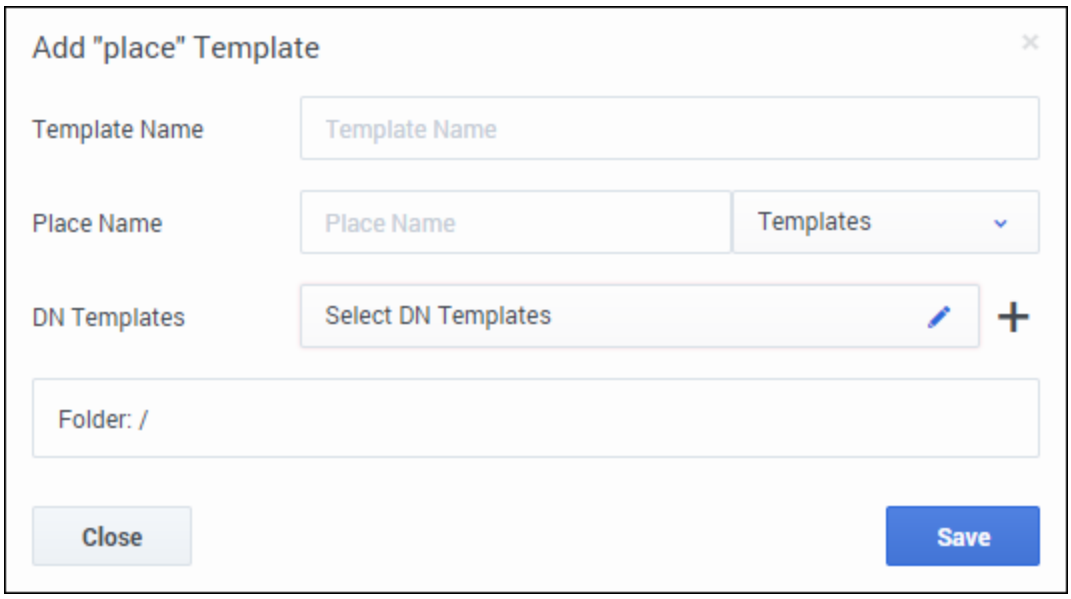
Person Templates



Person Templates define properties such as Channels, Skills, Access Groups, Annex options, folder, and Supervisor role for newly created agents.

You also need to select a place template and one or more login templates. Then you can select the channels, skills, annex, and access groups that you want to apply to this template. Before you save the **Person** template, select the folder where the user is created.

Place template



Place Templates must contain one or more DN Templates.

If your contact center has a requirement to specify unique place names for different business groups or regions, type a place name variable. For example, the place name might be the same as the agent's phone number or as the agent's user name. Place names can contain the following variables:

- {{person.userName}}
- {{person.emailAddress}}
- {{phone.number}}
- {{phone.description}}
- {{phone.index}}

You can also combine variables. For example: {{person.userName}}-{{phone.index}} creates **JohnAppleseed-1** and **JohnAppleseed-2** if JohnAppleseed has two phone numbers.

Next Steps

Now that you have set up templates, you can proceed to provision your contact center with agents. See Agent Accounts.

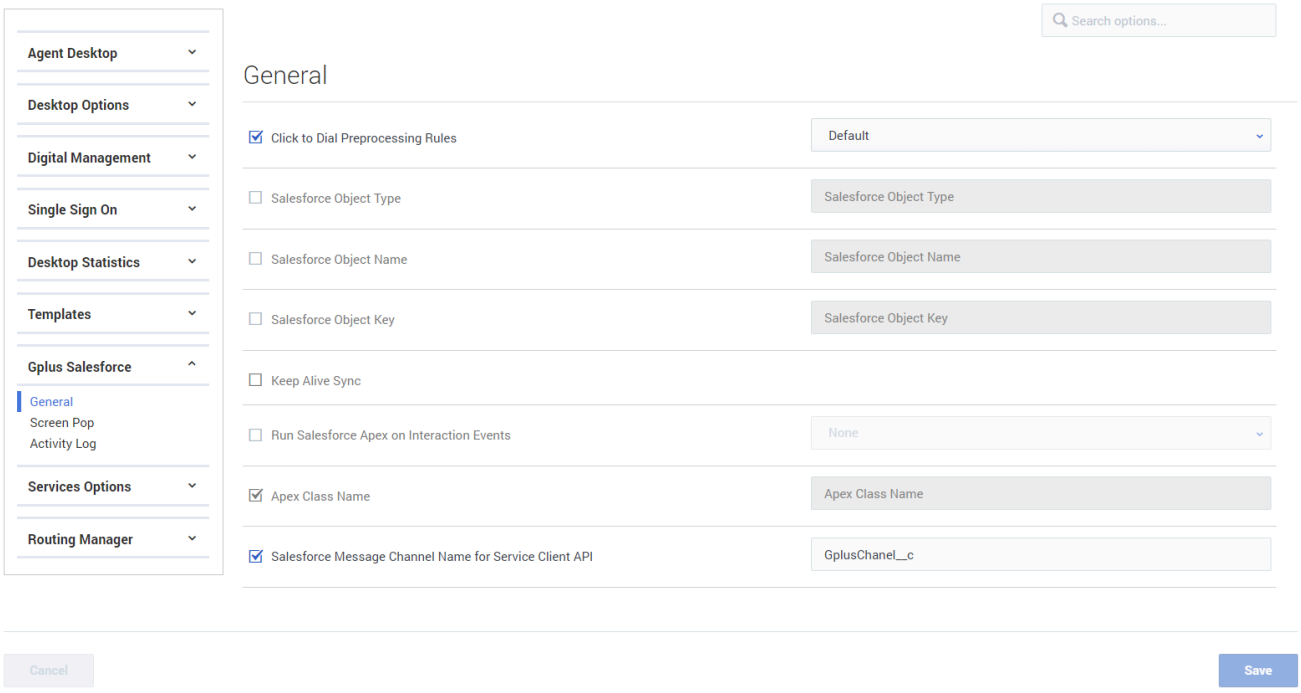
Gplus Salesforce

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

Configure options for the Gplus Adapter on the **Contact Center Settings** page.

The **Gplus Salesforce** options are organized into the following sections:

General options

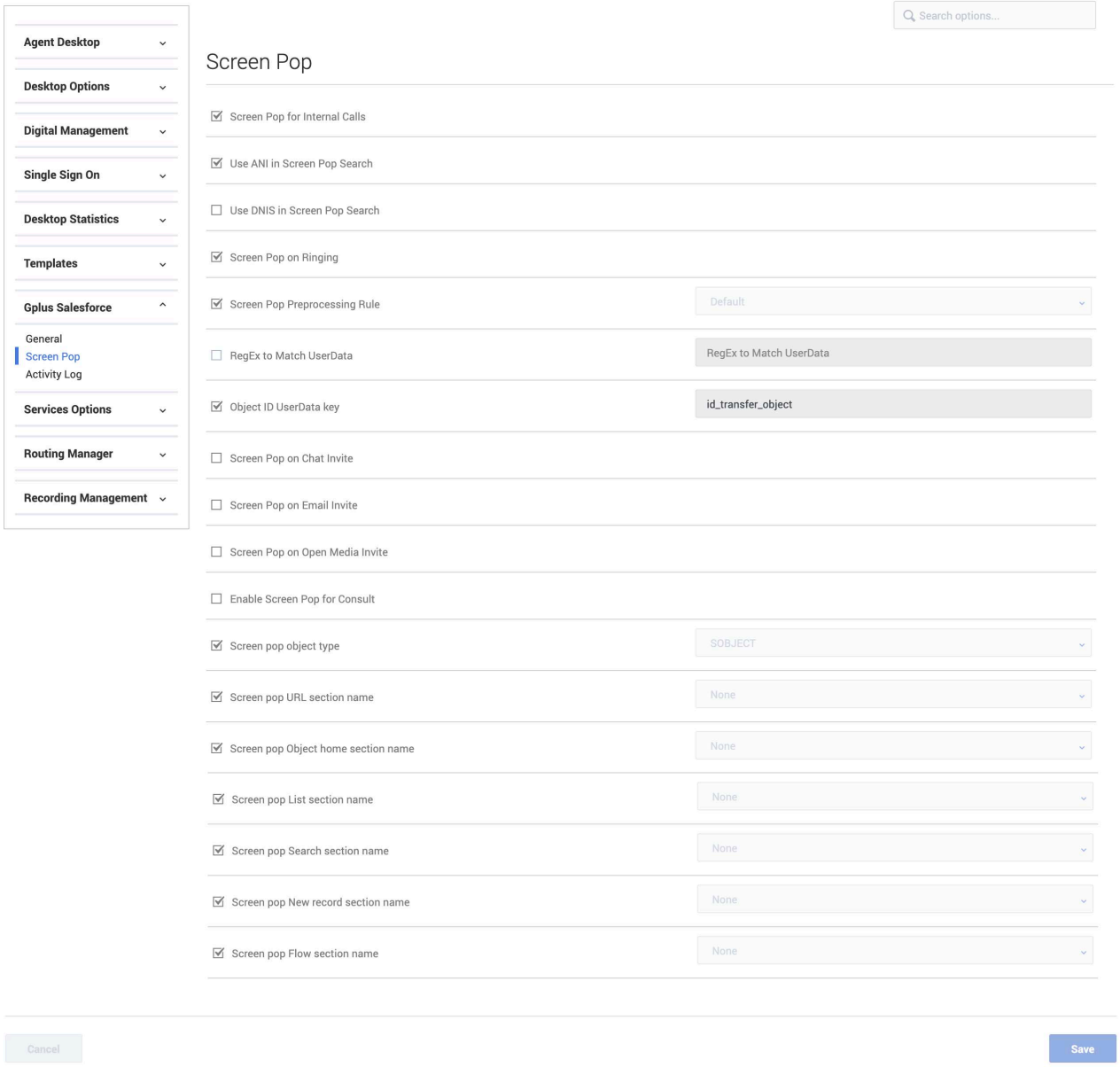


In the **General** section, configure the following options:

- **Click to Dial Preprocessing Rules** specifies the rules to apply to a phone number before making a dialing request.

- **Salesforce Object Type** specifies the key that Adapter uses when saving the type of the focused Salesforce object to attached data.
- **Salesforce Object Name** specifies the key that Adapter uses when saving the name of the focused Salesforce object to attached data.
- **Salesforce Object Key** specifies the key that Adapter uses when saving the ID of the focused Salesforce object to attached data.
- **Keep Alive Sync** specifies whether to prevent Adapter from logging out of Salesforce when there is no activity in Salesforce beyond the configured time period. If enabled, Adapter refreshes the Salesforce session at short intervals and keeps the Adapter-Salesforce session alive.
- **Run Salesforce Apex on Interaction Events** specifies the interaction event which triggers Salesforce Apex to run.
- **Apex Class Name** configures the name for a Salesforce custom apex class.
- **Salesforce Message Channel Name for Service Client API** specifies the name of the message channel in Salesforce for Service Client API.

Screen Pop settings

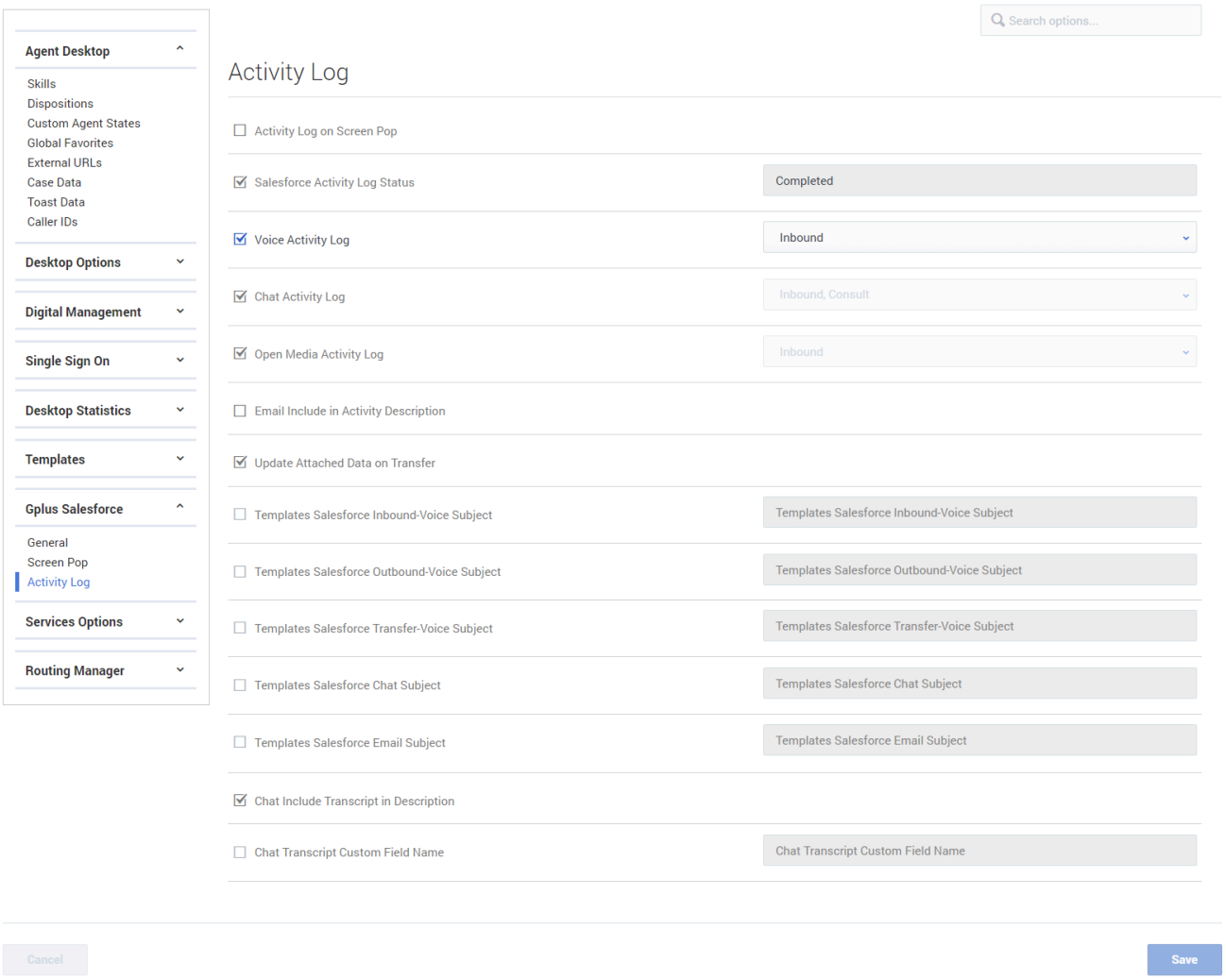


In the **Screen Pop** section, configure the following options:

- **Screen Pop for Internal Calls** specifies whether the adapter initiates a screen pop for internal calls.
- **Use ANI in Screen Pop Search** specifies whether ANI is to be used in the screen pop search.
- **Use DNIS in Screen Pop Search** specifies whether DNIS is to be used in the screen pop search.
- **Screen Pop on Ringing** specifies whether the adapter initiates a screen pop when the call is ringing. If not enabled, or set to **False**, the adapter initiates a screen pop when the call is established.

- **Screen Pop Preprocessing Rule** specifies the preprocessing rule to apply to the ANI or DNIS before showing the screen pop. When you enable this option, you can choose the default preprocessing rules or add a rule. If you add a rule, you will need to define a **Display Name** for the rule and define the **RegEx**, **Prefix**, and **Description**.
- **RegEx to Match UserData** specifies a regular expression pattern used to match one or more UserData keys. The values for these keys are used to build the search and screen pop expression. If this option is not defined, the adapter uses the existing `cti_` prefix logic.
- **Object ID UserData key** specifies the custom UserData key that the adapter should use for the object ID of the focused page when performing a transfer.
- **Screen Pop on Chat Invite** specifies whether the adapter initiates a screen pop in Salesforce immediately after displaying a chat invite notification. If the option is not set or is set to false, the adapter initiates a screen pop only when an agent accepts the chat invite.
- **Screen Pop on Email Invite** specifies whether the adapter initiates a screen pop in Salesforce immediately after displaying an email invite notification. If the option is not set or is set to false, the adapter initiates a screen pop only when an agent accepts the email invite.
- **Screen Pop on Open Media Invite** specifies whether the adapter initiates a screen pop in Salesforce immediately after displaying the open media invite notification. If the option is not set or is set to false, the adapter initiates a screen pop only when an agent accepts the open media invite. This option supports the items configured in the openmedia.workitem-channels.
- **Enable Screen Pop for Consult** specifies whether the adapter initiates a screen pop in Salesforce when using voice consult. This option works with option **Update Attached Data on Transfer**
- **Screen pop object type** specifies the the type of object (for instance "URL" or "OBJECTHOME") to screen pop in Salesforce.
- **screen pop URL section name** specifies the URL for the screen pop object type.
- **Screen pop Object home section name** specifies the object home (for instance, an account or contact) for the screenpop object type.
- **Screen pop List section name** specifies a list view for the screenpop object type.
- **Screen pop Search section name** specifies the top results section of the search page for the screenpop object type.
- **Screen pop New record section name** specifies the new record name (for instance, new account or new contact name) for the screenpop object type.
- **Screen pop Flow section name** specifies the target UI screen flow for the screenpop object type.

Activity Log options



In the **Activity Log** section, configure the following options:

- **Activity Log on Screen Pop** specifies whether Adapter creates the activity in Salesforce on screen pop and modifies logged activity when interaction is marked done. Activity on screen pop is created for inbound and internal calls. This option depends on **Screen Pop for Internal Calls** being selected and the value of **Voice Activity Log** containing **Inbound** and **Internal**.
- **Salesforce Activity Log Status** specifies the custom task status to be applied when an activity log is created by Adapter. Prerequisite: The status must exist in the Salesforce Task Status Picklist.
- **Voice Activity Log** specifies the call types for the adapter to create in the activity history for Salesforce.
- **Chat Activity Log** specifies the chat types for the adapter to create in the activity history for Salesforce.
- **Open Media Activity Log** specifies the open media types for the adapter to create in the activity

history for Salesforce. To disable the open media activity log creation, set the option to None.

- **Email Include in Activity Description** specifies to include the text version of the email body in the activity description in Salesforce. The email body is saved for only inbound emails and outbound replies.
- **Update Attached Data on Transfer** specifies whether to update the call data with the ID of the Saleforce object the agent is viewing when performing a transfer or conference. When set to **True**, the agent that receives the voice or chat interaction gets a screen pop for the most relevant object.
- **Chat Include Transcript in Description** specifies whether the adapter saves the chat transcript, as part of the activity description in Salesforce.
- **Chat Transcript Custom Field Name** specifies the name of a custom activity field in Salesforce.

Custom templates

You can specify what the adapter saves for the subject field in the Salesforce activity by using custom templates with the following options:

- Templates Salesforce Inbound-Voice Subject
- Templates Salesforce Outbound-Voice Subject
- Templates Salesforce Transfer-Voice Subject
- Templates Salesforce Chat Subject
- Templates Salesforce Email Subject

The value specified for each of these fields can be a combination of text and parameters to achieve the desired result for each type of interaction.

For example, you might want the activity for all inbound voice calls to have details about the customer's name and phone number. In this case, here's how you would configure the option:

```
Templates Salesforce Inbound-Voice Subject = Customer: {interaction.contact}, Phone Number: {interaction.ani}
```

If a call comes in from Willard Clinton at 123-456-7890, when the call ends the adapter creates the related activity in Salesforce with following subject field: `Customer: Willard Clinton, Phone Number: 123-456-7890`

Another common use case might be to include information from UserData for all inbound calls. This could come from your own UserData keys or from UserData provided by Agent Desktop (see the table above). For example, you want the activity history's subject to include information from your own UserData key, PurposeOfCall, and the contact's name. Here's how you should configure the option:

```
Templates Salesforce Inbound-Voice Subject = Call from {interaction.contact} about {userData.PurposeOfCall}
```

In this scenario, if Willard Clinton calls into the contact center and chooses the Technical Support option in the IVR, that information is saved in the PurposeOfCall UserData key. When the call ends, the adapter creates the related activity in Salesforce with the following subject field: `Call from Willard Clinton about Technical Support`

The adapter supports the following templating variables:

Variable	Description
interaction.ani	The number that originated the call. This variable identifies the caller for inbound calls and is best used in inbound templates.
interaction.callType	The type of call (inbound/outbound).
interaction.caseId	The unique ID of the related case.
interaction.contact	The first and last name of contact.
interaction.dnis	The last call dialed (useful for call transfer). This variable identifies the outbound location for outbound calls and is best used in outbound templates.
interaction.endDate	The date and time when interaction ended.
interaction.isConsultation	This is true if the interaction is a consultation.
interaction.startDate	The date and time when the interaction started.
userData.<key>	This can be any UserData key available for the interaction. You might also find some of the following UserData keys useful (they're included by default by Agent Desktop on transfers): • userData.GCS_TransferringAgentName — The name of the transferring agent. • userData.GCS_TransferringDate — The date and time of transfer. • userData.GCS_TransferringEmployeeId — The ID of the transferring employee. • userData.GCS_TransferringReason — The reason for the transfer. This is an empty string if no reason exists.
contact.EmailAddresses	A list of email addresses associated with the contact.
contact.PhoneNumbers	A list of phone numbers associated with the contact.

Next Steps

After you have configured the Gplus Adapter options, you can proceed to set up your Agent Groups. See Agent Groups.

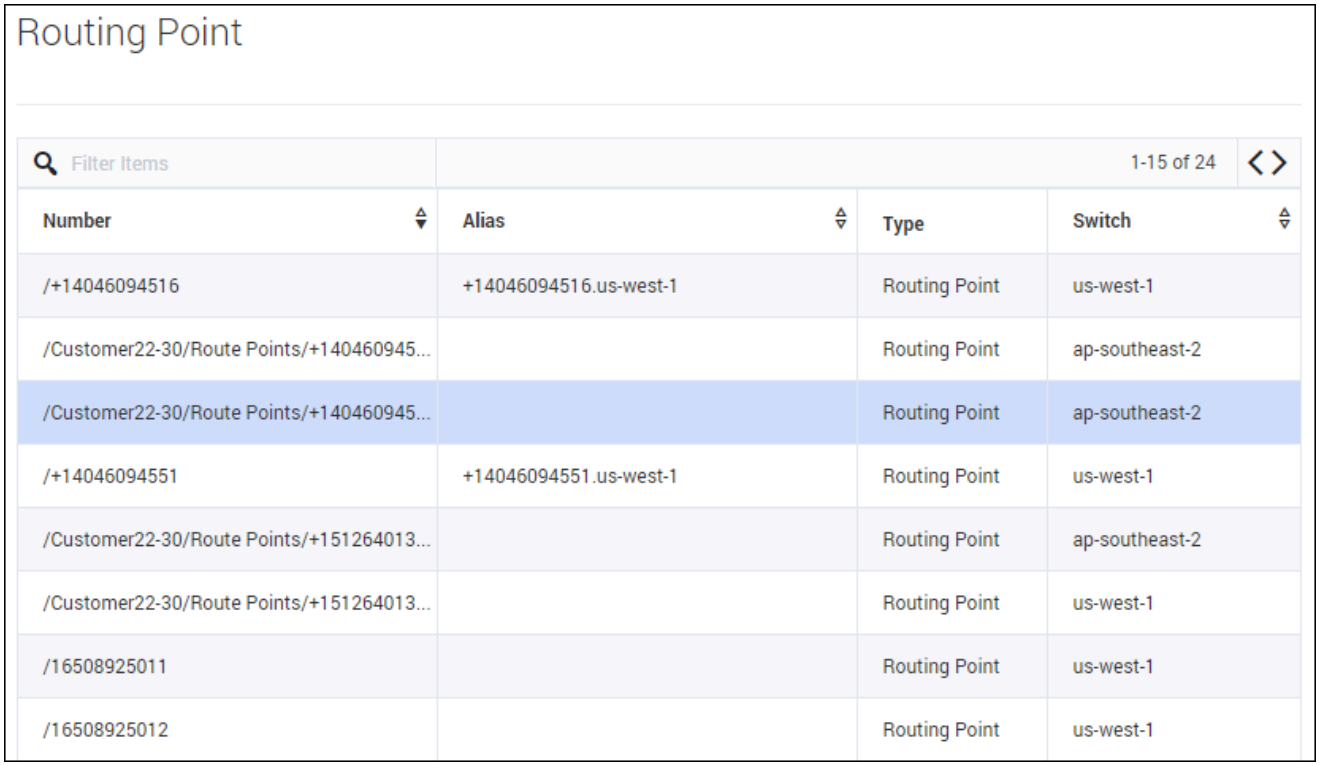
Routing Manager

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

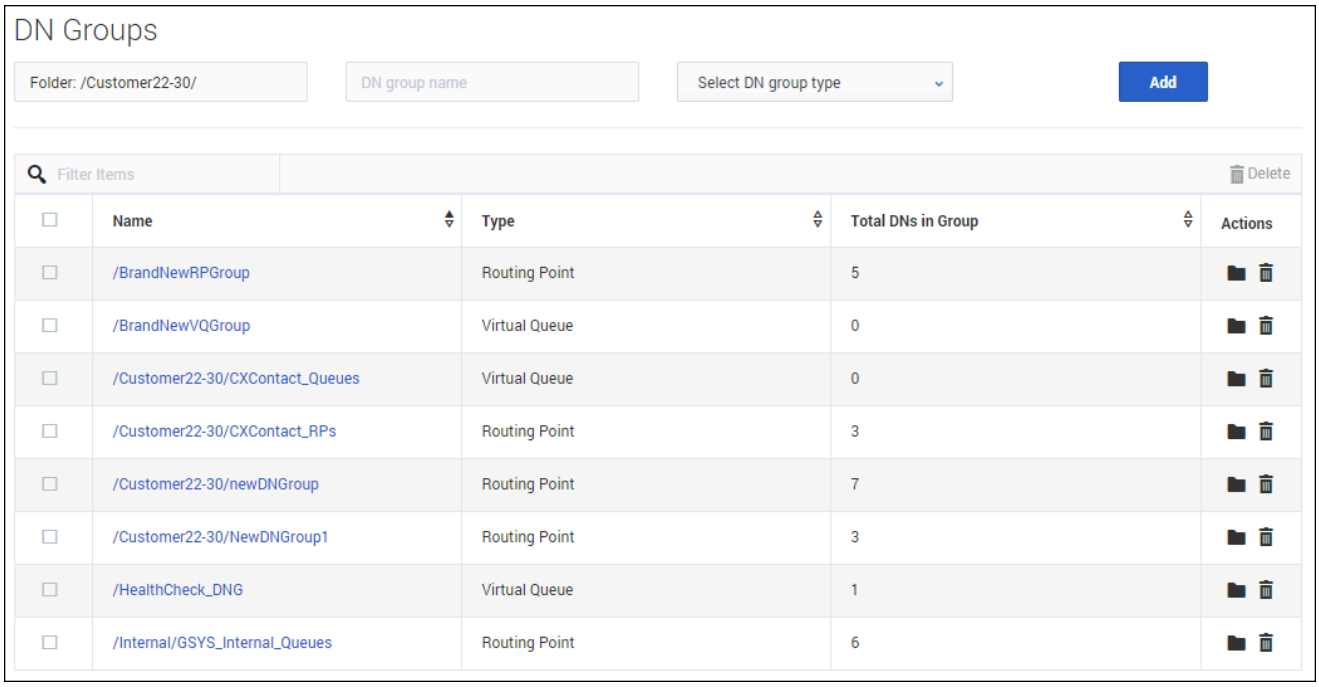
The **Routing Manager** section on the **Contact Center Settings** allows you to view your **Routing Points** and your **Virtual Queues**, as well as create **DN Groups** for both.

In the **Routing Point** and **Virtual Queue** sections, the **Alias** name for each line is editable; however all other fields are read only.

You can also filter both lists by **Number**, **Alias**, or **Switch**.



DN Groups



In the **DN Groups** section, you can group your Routing Points and Virtual Queues for display in Real-time Reporting with Pulse and to simplify the on-boarding of new agent and business groups.

To create a new DN group, select an existing Folder or create a new one, enter the DN group name, and select the group type before clicking **Add**.

Once you have created a new DN group, select it from the list and begin adding DNs (directory numbers) to the group. You can add DNs by dragging and dropping, selecting one or more and clicking **Add**, or by clicking on the arrow that pops up when you hover over an available DN. You can remove a DN from a group by clicking the **X** that pops up when you hover over that DN.

Desktop Options

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

To configure desktop options, click **Contact Center Settings > Desktop Options**. In the **Desktop Options** section, you can select the channels and other options that are available to all agents and users in your contact center. Click the links below to view an image and option descriptions for each section.

- Channel options
- Standard Response options
- Statistics options
- Global Login options
- Supervisor options
- Voice options
- Chat options
- Email options
- Outbound options
- Workitem options
- Facebook options
- Twitter options
- Recording options
- Contact options
- Interaction History Advanced Search options
- Standard Response Library options
- Genesys Softphone options
- Feedback options
- My Channels options
- Service Client options

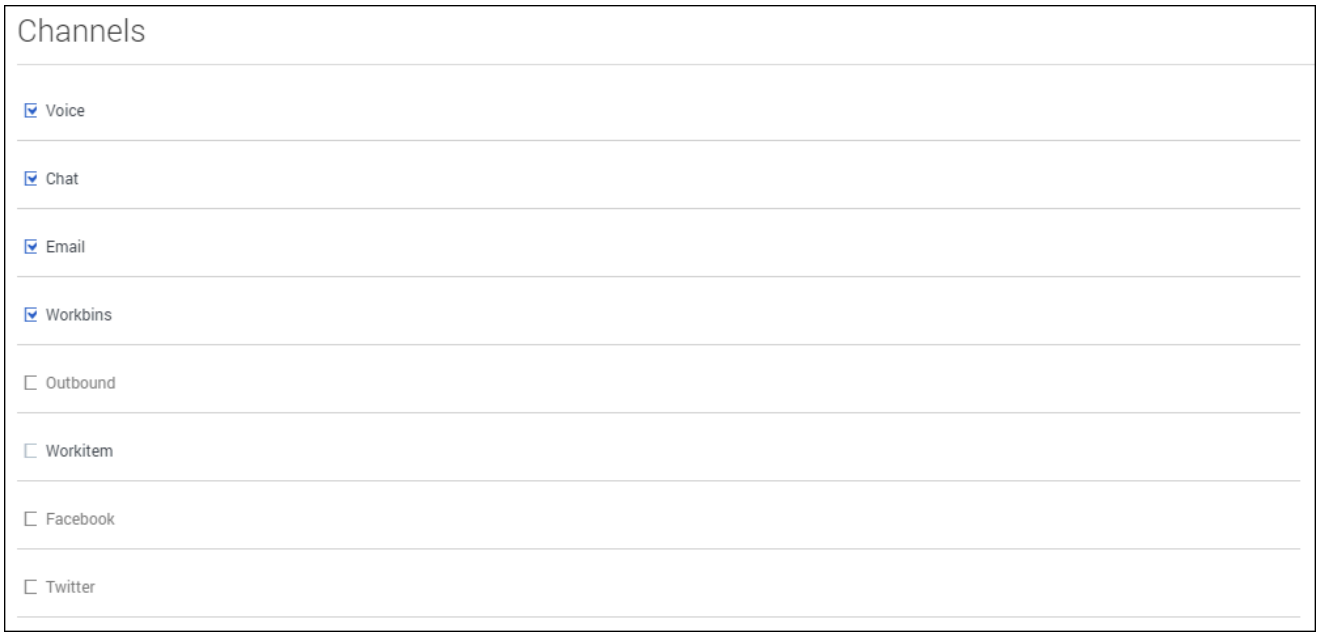
Next Steps

After you have configured your Desktop Options, configure options for the Gplus Adapter. See CRM Adapter. You can also proceed to configure your Agent Groups. See Agent Groups.

Channel options

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

In the **Channels** section under **Desktop Options**, select the channels you want to enable in your contact center.



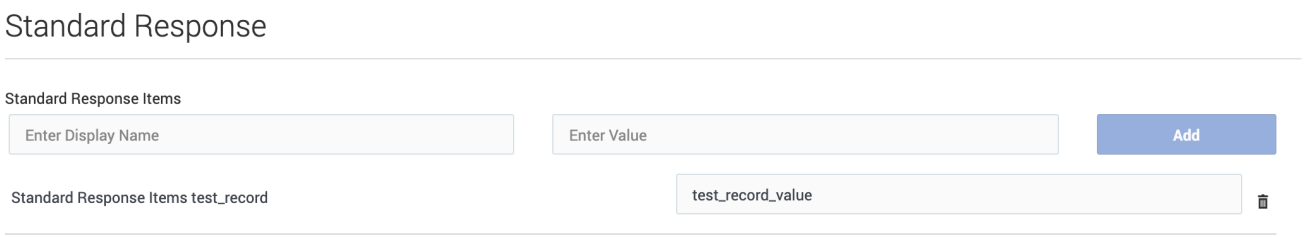
The choices that you make in the **Channels** section are not automatically applied to all agents and users. Your selection provides a set of channels that you choose from when you add an agent. For example, you can select **Voice**, **Chat**, **Email**, **Workbins**, **Outbound**, **Workitem**, **Facebook**, or **Twitter** for the contact center. Then, when you add an agent or edit an agent, you can configure the individual agent to only have access to certain channels.

You can also enable **Genesys Altocloud** for **Chat** channel users working with the new Agent Desktop. When editing or creating a user in Agent Setup, you can select **Altocloud** under **Desktop Options** in the **Users** section. This option is not available in the **Contact Center Settings** or **Agent Groups** sections. For more information, see Altocloud Chat Interaction.

Standard Response options

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

In the **Standard Response** section, you can specify custom field codes, such as agent nicknames, roles, departments, and email signatures.



You can format custom field codes to represent agent attributes using `Agent.<CustomAgentAttr>`, or any custom attribute by dropping the agent modifier and using `<CustomFieldCode>`. Enter a **Display Name** and the corresponding **Value** for each field code.

Statistics options

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

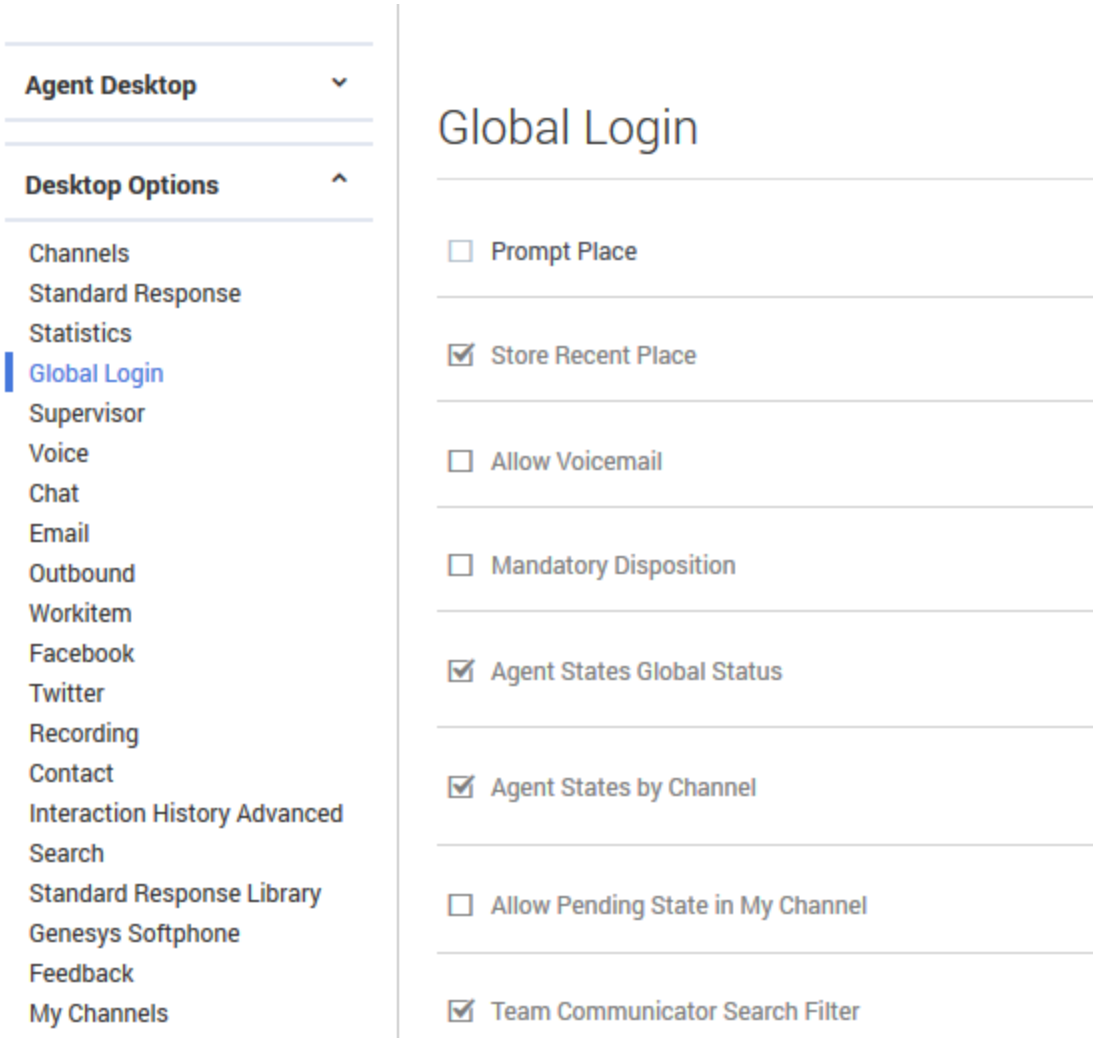
In the **Statistics** section of the **Contact Center Settings** page, you can configure the following options:

- **Default Statistics displayed** specifies the statistics displayed in the Agent Desktop menu bar. The statistic specified by this option is the name of a section containing the statistic definition or the statistic object.
- **Refresh Time** defines the frequency of notification (in seconds) for statistics.
- **Time for Statistics Reset** specifies the time that you want agent and call center statistics to be reset in Agent Desktop. Indicate the time using UTC time in 24-hour format. For example, PDT midnight is 7AM in UTC and should be specified as `7:00`.
- **Maximum quantity of Gadget Statistics displayed** specifies the quantity of statistics displayed.

Global Login options

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

In the **Global Login** section under **Desktop Options** on the **Contact Center Settings** page, configure the following options:



- **Prompt Place** requires the agent to specify a place when they log in.
- **Store Recent Place** specifies whether the most recently used Place is stored. This option is applicable when Prompt Place option is set to true.
- **Voicemail** enables the Voicemail menu.
- **Mandatory Disposition** specifies whether it is mandatory for the agent to select a disposition before marking an interaction as done.
- **Agent States Global Status** defines the available agent states in the global Status menu.
- **Agent States by Channel** defines the available agent state actions in the My Channels contextual menu.
- **Allow Pending State in My Channel** allows an agent to have a pending state in the My Channel tab.
- **Team Communicator Search Filter** specifies the list of filters that an agent can use to search for contacts and internal targets in the Team Communicator. When you select this option, all filters are enabled by default. To customize the filters, click the drop-down list box and deselect the filters that

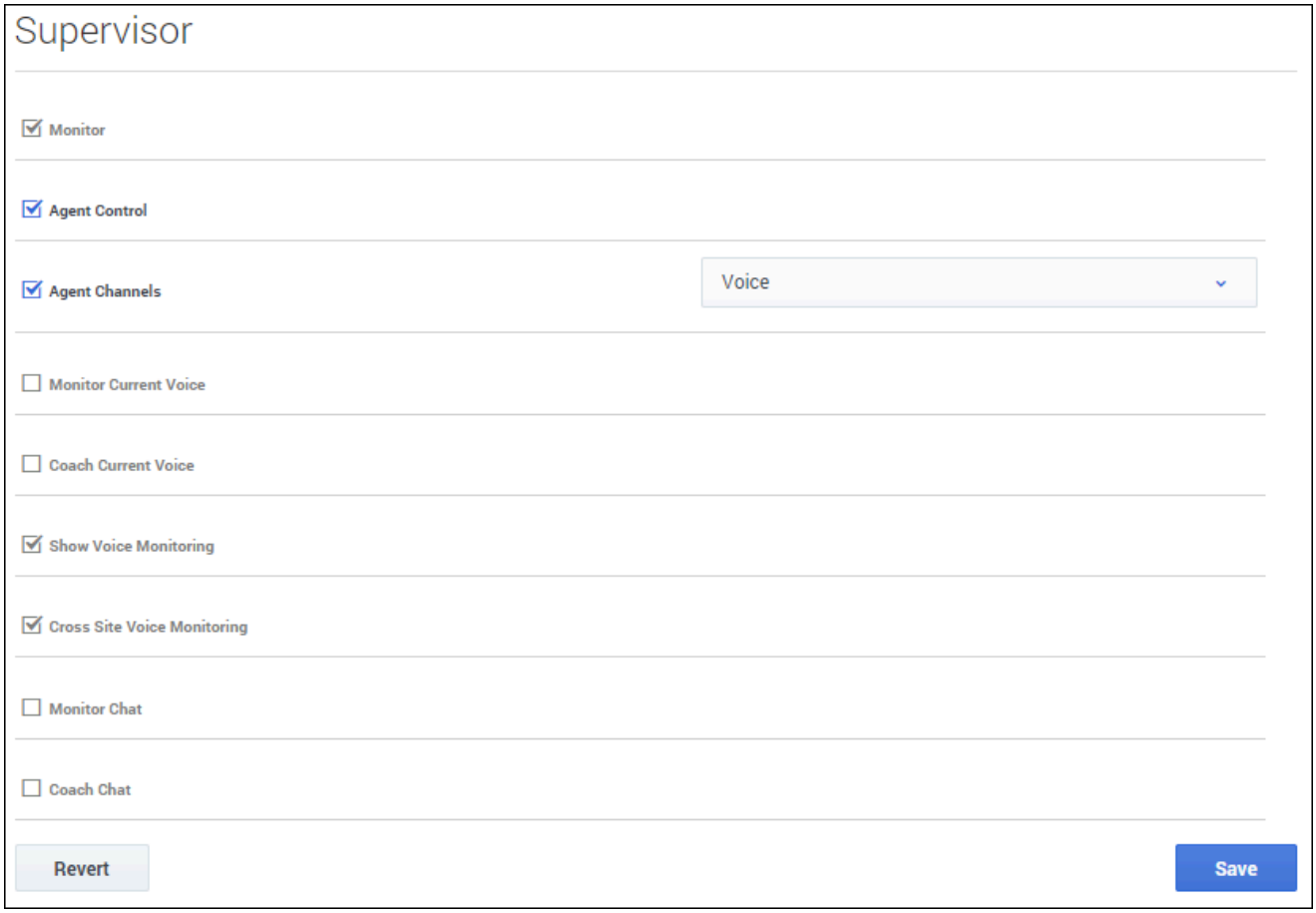
you don't want to include.

- **Team Communicator Exclude from Groups** - When you select this option, the agents from excluded groups are not visible in Team Communicator, Favorites, and Statistics.
- **Team Communicator Exclude Groups** - Specifies the list of agent groups to be excluded from searches and statistics. Virtual agent groups are not supported. This option is overridden by the Team Communicator Restrict to Groups option.
- **Team Communicator Include Groups** - Specifies the list of agent groups that are returned for searches and statistics. Overrides the Team Communicator Exclude Groups option. Virtual agent groups are not supported.
- **Team Communicator Restrict to Groups** - When you select this option, only agents from restricted groups are visible in Team Communicator, Favorites, and Statistics.
- **Team Communicator Max Favorites** specifies the maximum size of the favorites list displayed in the Team Communicator.
- **Transaction Override Attached Data Key** allows you to override desktop options in Agent Desktop by setting a key value to the name of a transaction list object. When you set the value of the identified key to the name of the transaction list object, desktop options in the transaction list object are used instead of the Agent Desktop application or agent group objects.
- **Inactivity Timeout (min)** specifies the number of minutes of agent inactivity (no mouse or keyboard usage) that triggers application locking. If the agent has been inactive longer than the number of minutes this option, the agent session is ended.
- **Alert before timeout (sec)** specifies the number of seconds that pass before the inactivity alert is sent to the agent.
- **Sidebar Region** specifies the order of the Region to be displayed in the sidebar.
- **Use Performance Tracker** allows the use of the Performance Tracker.
- **Use Dashboard** allows the use of the Dashboard.
- **Auto-focus on inbound interactions** enables inbound interactions to automatically be in focus when accepted.
- **Channel to be auto-focused** specifies whether the inbound interaction of a particular media type is automatically in focus when accepted.
- **Invalidate Auth SSO session on Workspace logout** specifies whether the Single sign-on (SSO) authentication session is invalidated when an agent logs out of the Workspace. When this option selected, both Workspace and user authentication sessions get terminated when the agent exits the Workspace and the agent is redirected to the authentication login page. When this option is cleared, only the Workspace session is removed when the agent exits the Workspace and the agent is redirected to the Workspace session ended page.
- **Show Change Account Link** specifies whether or not to display the Change Login Account link on the login error page and the second step login page. This option is not used if the **Invalidate Auth SSO session on Workspace logout** option is selected.

Supervisor options

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

In the **Supervisor** section under **Desktop Options**, configure the following options:



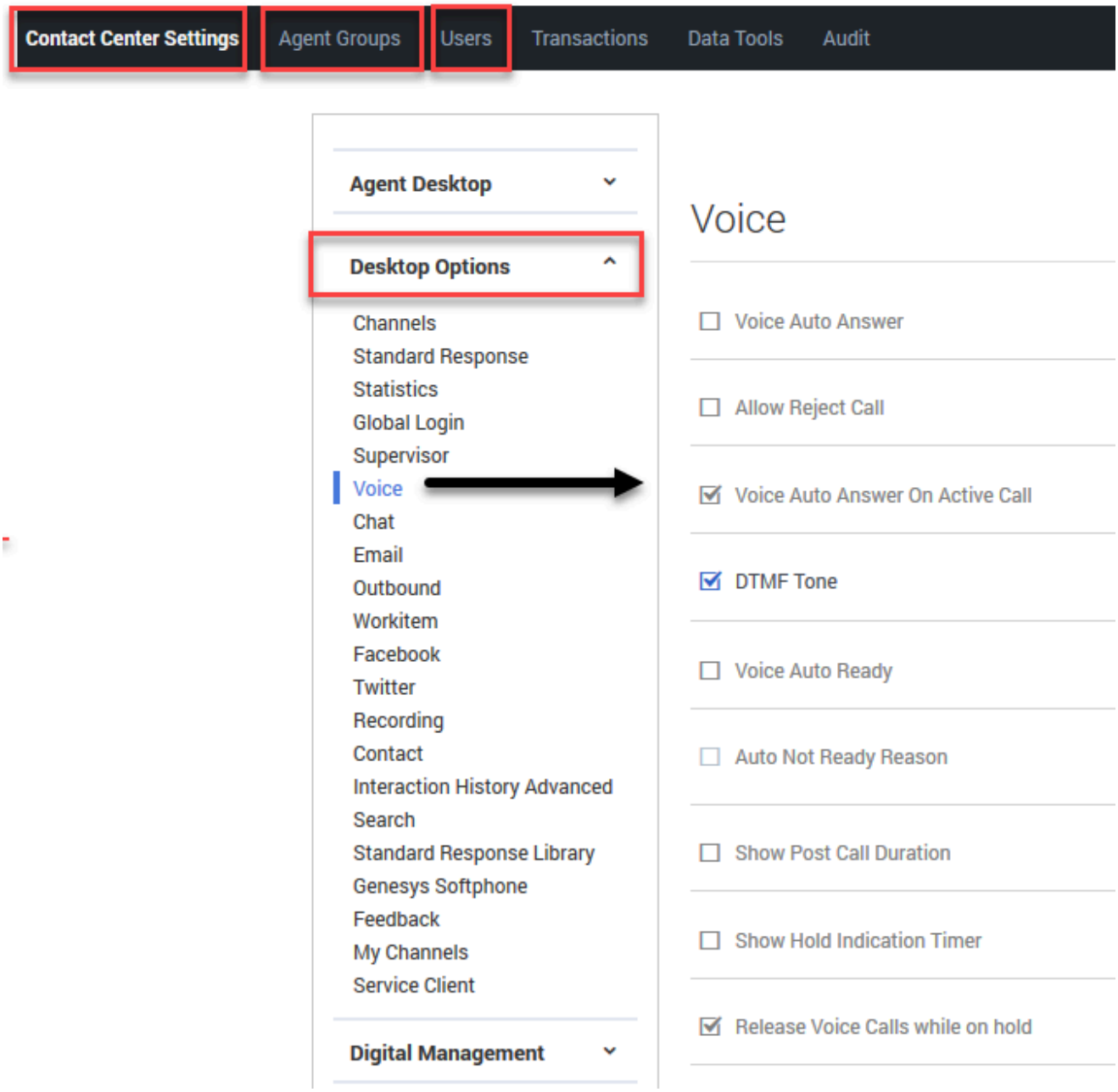
- **Monitor** enables supervisor and team lead monitoring on agent interactions.
- **Agent Control** allows the supervisor or team lead to access the **My Agents** view.

- **Agent Channels** defines the channels to be displayed in the **My Agents** view.
- **Refresh Agent Control** enables the manual refresh of agent states.
- **Agent Page Limit** specifies the maximum number of rows displayed per page in the My Agents tab.
- **Agent Page Refresh** specifies the frequency, in seconds, to refresh the list of users. Auto-refresh is turned off when this is set to '0' (zero).
- **Monitor Current Voice** allows the supervisor or team lead to monitor voice interactions.
- **Coach Current Voice** allows the supervisor or team lead to coach voice interactions.
- **Show Voice Monitoring** enables agents to be notified when the current call is being monitored by a supervisor or team lead.
- **Cross Site Voice Monitoring** allows the supervisor or team lead to perform supervision of an agent that is located at a different location.
- **Monitor Chat** allows the supervisor or team lead to monitor chat interactions.
- **Coach Chat** allows the supervisor or team lead to coach chat interactions.
- **My Team Workbins** enables supervisors to see the workbins of the agents in their Agent Group.
- **Access to Interaction Management** enables supervisors to see Interaction Management.
- **Allow Moving Interactions to Queue** enables supervisors to move interactions from displayed workbins to available queues.
- **Allow Moving Interactions to Workbin** enables supervisors to move interactions from displayed workbins to other workbins.

Voice options

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

In the **Voice** section of the **Contact Center Settings** page, configure the following options:



- **Voice Auto Answer** specifies that voice interactions are automatically answered when an agent receives the invitation.
- **Voice Auto Answer On Active Call** specifies that a voice interaction is not automatically answered if the agent has another active call.
- **DTMF Tone** specifies to play a tone when the agent select a DTMF key.
- **Voice Auto Ready** places the agent in the Ready state as soon as they log in.
- **Auto Not Ready Reason** places the agent in Not Ready state as soon as they log in. When you select this option, define the default Not Ready reason code.

- **Show Post Call Duration** allows the post-call duration for a voice call to be shown in the interaction bar's tool tip area.
- **Show Hold Indication Timer** allows the hold duration to be shown to the agent in the interaction bar.
- **Release Voice Calls while on Hold** enables agents to manually end voice calls that are on hold.
- **Conference** enables one-step voice conferencing.
- **Two-Step Voice Conference** enables two-step voice conferencing.
- **Clear Conference on Release** clears a conference call of all parties when the last agent leaves the conference call.

- **Cancel ACW on Done** specifies that the voice channel is to be returned to the agent's former availability status when the agent marks the voice interaction as Done while in After Call Work status.
- **Cancel ACW on Change** specifies that After Call Work status should be cancelled when an agent switches from After Call Work to Ready or Not Ready during a call.
- **Lookup Contact for Voice** activates Agent Desktop for contact lookup when an interaction is presented to an agent.
- **Create Contact for Voice** will create a contact if the initial contact lookup fails to find an existing contact.
- **Caller ID Anonymous** specifies whether the anonymous Caller ID is enabled. This option applies to only outbound calls.
- **Caller ID for Consultations** enables agents to use a Caller ID for consultations.
- **Caller ID for Conferences** enables agents to use a Caller ID for single-step conferences.
- **Caller ID for Transfers** enables agents to use a Caller ID for single-step transfers.
- **Routing Point for Skill Based Transfers** specifies the call number used by the Routing Point feature.
- **Routing Based Actions** defines the list of routing-based actions that an agent can perform. Options include:
  - **Make Call**
  - **One Step Conference**
  - **Initiate Conference**
  - **One step Transfer**
  - **Initiate Transfer**
- **Routing Based Targets** defines the list of contact types to which an agent can route calls through the actions defined in the **Routing Based Actions** option. Options include Agent, Routing Point, Type Destination, Outbound Record, and Contact. Note that Agent Groups and Skills targets are not listed here, because they are configured through a routing strategy.
- Options to exclude a range of numbers from the Team Communicator. For each option, if you enable it, you must type a regular expression (RegEx) to match against the following items:
  - Entered phone number
  - Selected agent or routing point

    For example, to exclude extensions that are 7 digits and start with the numbers 7 or 8, type `^(7|8)\d{6}$`

  - **Make Call Exclude Number** specifies the format for a phone number to exclude from Team Communicator when making a call.
  - **Single Step Conference Exclude Number** specifies the format for a phone number to exclude from Team Communicator on a single step conference.
  - **Single Step Transfer Exclude Number** specifies the format for a phone number to exclude from Team Communicator on a single-step transfer.
  - **Consultations Exclude Number** specifies the format for a phone number to exclude from Team Communicator on consultation.

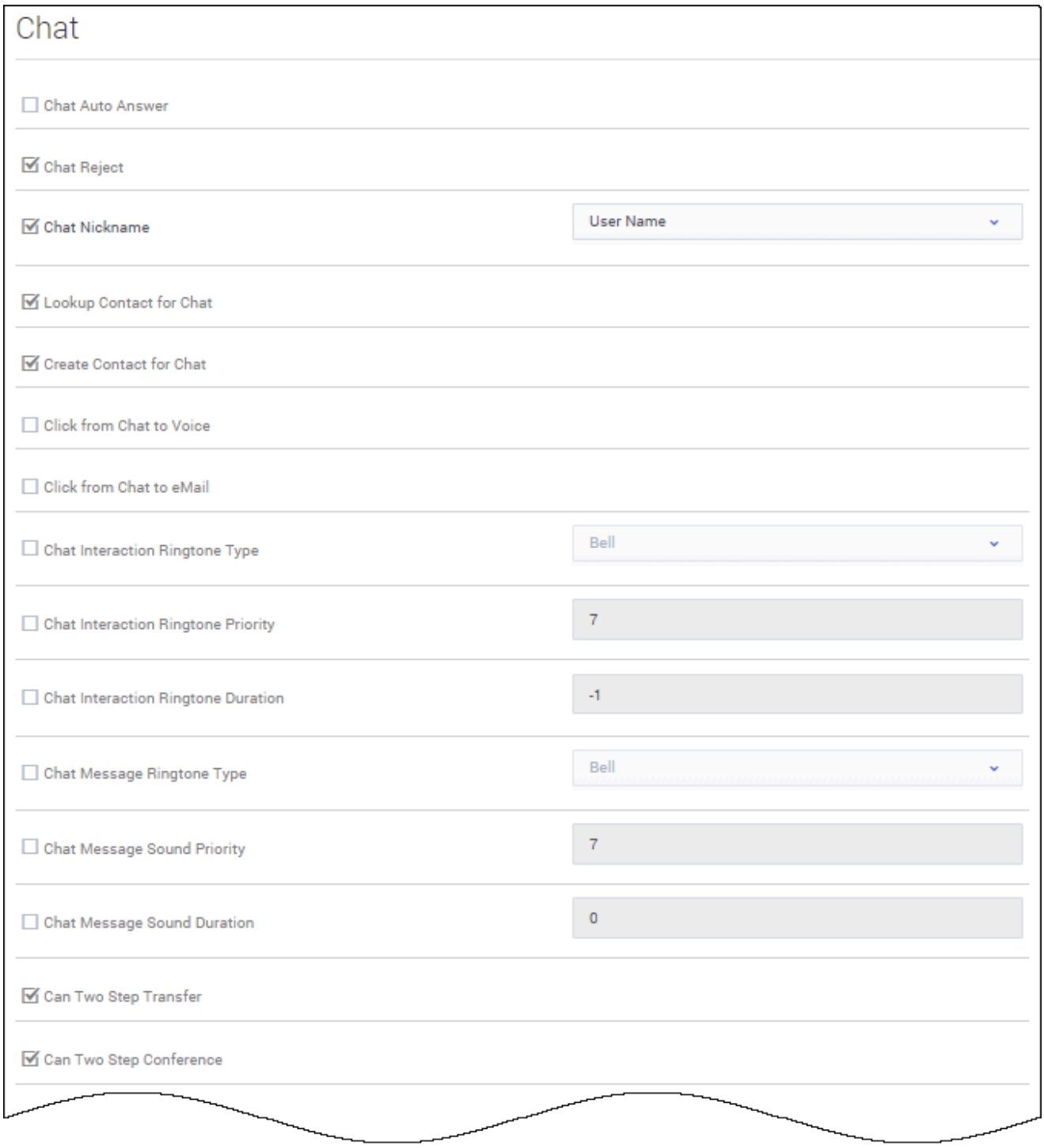
    These options are not applicable to global or personal favorites.

- **Voice Ringtone Type** specifies the voice ringtone type. Enable the option and then select the ringtone type from the drop-down list box.
- **Voice Ringtone Priority** specifies the priority level of the voice ringtone within the channels. Enable the option and then type the number for the priority level.
- **Voice Ringtone Duration** specifies how long the ringtone rings. Use the following values:
  - `-1` specifies to repeat the ringtone until the call is answered.
  - `0` specifies to play the ringtone only one time.
  - 1 or more specifies to play the ringtone for the specified duration (in milliseconds) and then repeat until the call is answered.
- **Can One Step Transfer** enables instant conferencing of a voice call.
- **Can Two Step Transfer** enables two-step transfer of a voice call.
- **Can Extend After Call Work** enables the extension of After Call Work indefinitely, until the user's status is changed manually.
- **Can Use WebRTC** enables WebRTC in Agent Desktop. Note: The **Can Use Voice Channel** option must be enabled.
- **Can change speaker volume** allows an agent to change speaker volume. Note: The **Can Use Voice Channel** option must be enabled.
- **Can mute microphone** allows an agent to mute and unmute the microphone. Note: The **Can Use Voice Channel** option must be enabled.
- **Can mute speaker** allows an agent to mute and unmute the speaker. Note: The **Can Use Voice Channel** option must be enabled.
- **Can send DTMF** specifies if DTMF is available for WebRTC calls. Note: The **Can Use Voice Channel** option must be enabled.
- **WebRTC Ringtone Type** specifies the voice channel ringing sound -
- **WebRTC Ringtone Priority** is the priority of ringtone. The higher the integer, the higher the priority.
- **WebRTC Ringtone Duration:**
  - -1 plays means to play and repeat until an explicit message stops it with an established event.
  - 0 means to play the entire sound one time.
  - An integer greater than 0 means a time in milliseconds to play and repeat the sound.

Chat options

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

In the **Chat** section of the **Contact Center Settings** page, configure the following options:



- **Chat Auto Answer** specifies that chat interactions are automatically answered when an agent receives the invitation.
- **Chat Reject** enables agents to reject incoming chats.

- **Chat Nickname** specifies the nickname that is displayed on the chat window in interactions. Type one the following syntax options:
  - `$Agent.UserName$` to display full username.
  - `$Agent.LastName$` to display only last name.
  - `$Agent.FirstName$` to display only first name.
  - `$Agent.FullName$` to display full name.
  - `$Agent.EmployeeId$` to display employee ID.
- **Lookup Contact for Chat** activates Agent Desktop for contact lookup when an interaction is presented to an agent.
- **Create Contact for Chat** will create a contact if the initial contact lookup fails to find an existing contact.
- **Click from Chat to Voice** enables users to initiate a voice interaction by clicking a phone number in a chat transcript.
- **Click from Chat to eMail** enables users to initiate an email by clicking an email address in a chat transcript.
- **Chat Interaction Ringtone Type** specifies the chat ringtone type. Enable the option and then select the ringtone type from the drop-down list box.
- **Chat Interaction Ringtone Priority** specifies the priority level of the chat ringtone within the channels. Enable the option and then type the number for the priority level.
- **Chat Interaction Ringtone Duration** specifies how long the ringtone rings. Use the following values:
  - `-1` specifies to repeat the ringtone until the chat is answered.
  - `0` specifies to play the ringtone only one time.
  - 1 or more specifies to play the ringtone for the specified duration (in milliseconds) and then repeat until the chat is answered.
- **Chat Message Ringtone Type** specifies the chat message ringtone type. Enable the option and then select the ringtone type from the drop-down list box.
- **Chat Message Sound Priority** specifies the priority level of the chat message sound within the channels. Enable the option and then type the number for the priority level.
- **Chat Message Sound Duration** specifies the length of the sound. Use the following values:
  - `-1` specifies to repeat the sound until the chat message is read.
  - `0` specifies to play the sound only one time.
  - 1 or more specifies to play the sound for the specified duration (in milliseconds) and then repeat until the chat message is read.
- **Can Two Step Transfer** enables two-step transfer of a chat, beginning with a chat consultation.
- **Can Two Step Conference** enables two-step conference of a chat, beginning with a chat consultation.
- **Can One Step Transfer** enables instant chat transfer.
- **Auto-disconnect** automatically disconnects a chat session when the agent is the last remaining party.
- **On-hold Queue** specifies the Interaction Queue where a chat interaction is placed when an agent

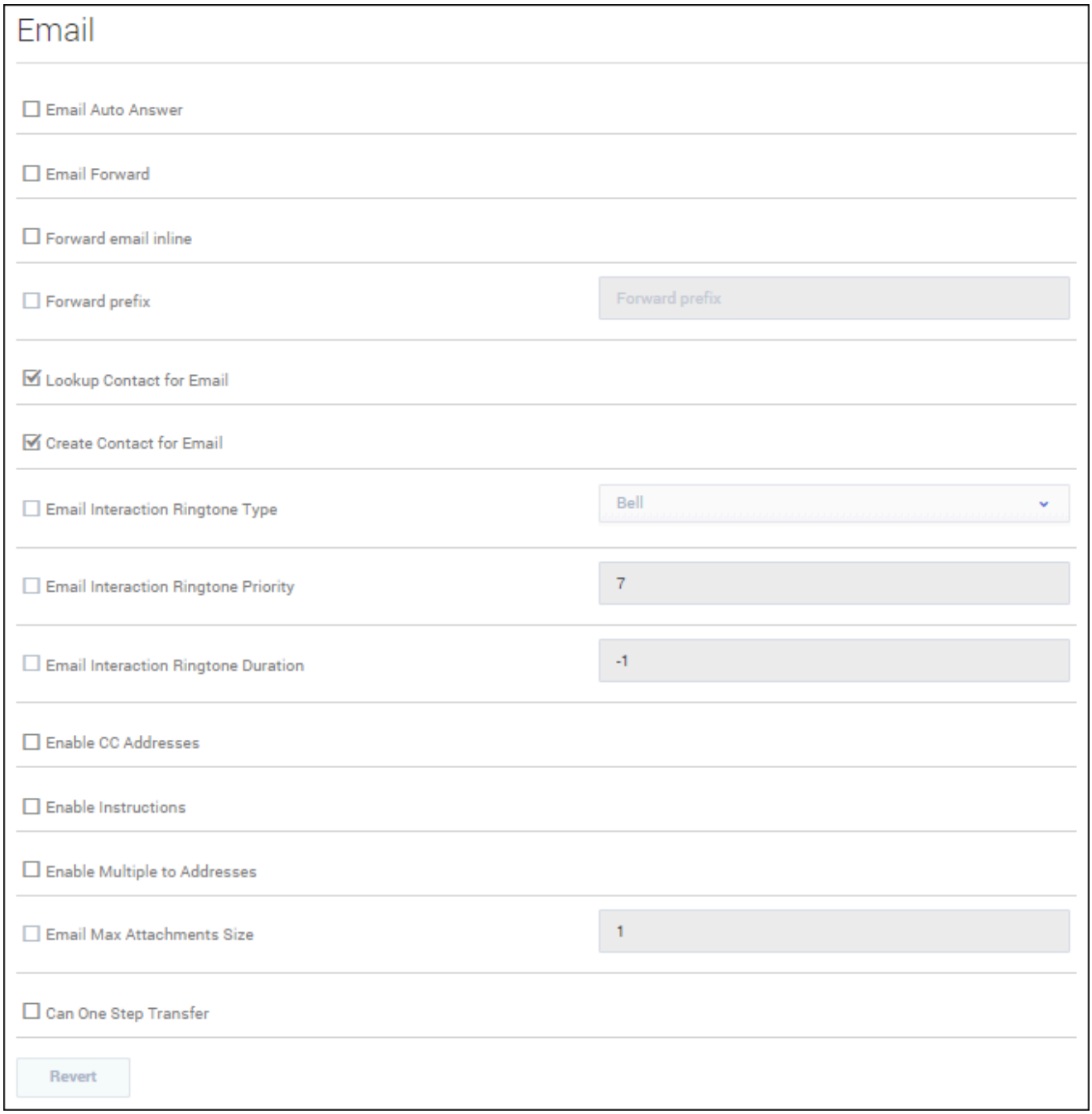
places it on hold. This option can be overridden by a routing strategy.

- **Warning time** specifies the time in seconds that a warning alarm alerts an agent of a customer awaiting a chat response.
- **Maximum time** specifies the time in seconds
- **Prompt for end** displays a confirmation prompt to an agent when they select End. This option can be overridden by a routing strategy.
- **URL push max records** specifies the maximum size of the pushed URL list.
- **Timestamp** specifies whether the timestamp is displayed in the Chat transcript area.
- **Typing** specifies whether a typing notification is sent to customers during a chat interaction.
- **Typing timeout** specifies the duration in seconds of when the typing notification is displayed after the last keystroke.
- **Decline** allows an agent to decline incoming chat interactions.
- **URL push** allows an agent to push URLs to customers during chat sessions.
- **Interaction Disposition** allows an agent to set a disposition code for chat interactions.

Email options

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

In the **Email** section under **Desktop Options**, configure the following options:



- **Email Auto Answer** specifies that email interactions are automatically answered when an agent receives the invitation.
- **Email Can Decline** enables agents to decline an incoming email.
- **Enable Email Save** enables agents to save in-progress outgoing email.
- **Enable Email Send** enables agents to send outgoing email.
- **Enable Add Attachment** enables agents to add attachments to outgoing email.

- **Allow Email Deletion** enables agents to delete an email.
- **Enable Email Reply** enables agents to reply to an email.
- **Enable Email Reply All** enables agents to reply all to an email.
- **Email Address Regular Expression** specifies the regular expression used to recognize an email address of a target entered in Team Communicator.
- **Email Forward** enables agents to forward emails to another agent.
- **Forward email inline** enables agents to forward an email as part of the body of an email, instead of as an attachment.
- **Forward prefix** specifies the prefix added to an email that includes an inline email. For example, "Fwd:".
- **Lookup Contact for Email** activates Agent Desktop for contact lookup when an interaction is presented to an agent.
- **Create Contact for Email** will create a contact if the initial contact lookup fails to find an existing contact.
- **Email Interaction Ringtone Type** specifies the email ringtone type. Enable the option and then select the ringtone type from the drop-down list box.
- **Email Interaction Ringtone Priority** specifies the priority level of the email message ringtone within the channels. Enable the option and then type the number for the priority level.
- **Email Interaction Ringtone Duration** specifies how long the ringtone rings. Use the following values:
  - `-1` specifies to repeat the ringtone until the email message is read.
  - `0` specifies to play the ringtone only one time.
  - 1 or more specifies to play the ringtone for the specified duration (in milliseconds) and then repeat until the email message is read.
- **Enable CC Addresses** enables the ability to select CC addresses.
- **Enable Instructions** enables an instruction field in the email message being forwarded.
- **Enable Multiple to Addresses** enables the ability to select more than one address in the To line.
- **From Addresses**
- **Email Max Attachments Size** specifies the maximum size in megabytes allowed for files attached to an external email interaction. Files larger than this maximum size will be refused by the system and an error message will be displayed to the user.
- **Can One Step Transfer** enables one-step transfer for emails.
- **Enable Interim Send** enables agents to send an interim email before a final reply.
- **Enable Click to Dial from Email** enables agents to dial a phone number while handling an email.
- **Allow Moving Email to Workbin** enables agents to move an email to Workbin.
- **Allow Email Format Toggle** enables agents to toggle between HTML and text formats.
- **Allow Email HTML Format** enables agents to use HTML formatting.

Outbound options

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

In the **Outbound** section under **Desktop Options**, configure the following options:

Outbound

- [x] Allow Reject Preview Record
- [x] Allow Cancel Record
- [x] Allow Marking Record as Do Not Call
- [ ] Allow Reschedule Outbound Record
- [x] Allow Reschedule Before Calling Contact
- [x] Allow Reschedule on New Number
- [x] Allow Setting Call Result
- [ ] Treatment Mode
- [x] Allow Request Next Preview Record
- [x] Allow Dialing an Alternative Chained Record
- [x] Allow Push Preview
- [x] Max Simultaneous Preview Record
- [x] Timed Preview Auto Dial

- **Allow Reject Preview Record** enables agents to reject a preview record. If an agent declines a preview record, it can be processed by another agent in the campaign.
- **Allow Cancel Record** enables agents to cancel a preview record. If an agent cancels a preview record, it is not processed again during the current campaign.
- **Allow Marking Record as Do Not Call** enables agents to mark a contact as *Do Not Call*.
- **Allow Reschedule Outbound Record** enables agents to reschedule an outbound record.
- **Allow Reschedule Before Calling Contact** enables agents to reschedule a call before calling the contact on the record.
- **Allow Reschedule on New Number** enables agents to add a new record to the chain by rescheduling with a new number.
- **Allow Setting Call Result** enables agents to set call results for outbound interactions.
- **Treatment Mode** specifies the type of treatment to be applied for the outbound record after it is marked as processed. If not set or set to none, no treatment is applied.
- **Allow Request Next Preview Record** enables agents to request a new preview record while processing of the previous preview record terminates.
- **Allow Dialing an Alternative Chained Record** enables agents to dial a number from the preview record chain that is different than the number selected by the system.
- **Allow Push Preview** enables agents to use outbound push preview.
- **Max Simultaneous Preview Record** specifies the maximum number of simultaneous Outbound Preview records an agent can view in the interaction window.
- **Timed Preview Auto Dial** specifies the method in which calls are dialed for agents in Preview, Push Preview, or Reschedule campaigns.The options are as follows:
  - The agent can manually dial a record.
  - The record is auto-dialed as soon as the record is accepted.
  - The record is auto-dialed after a specified number of seconds.

Facebook options

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

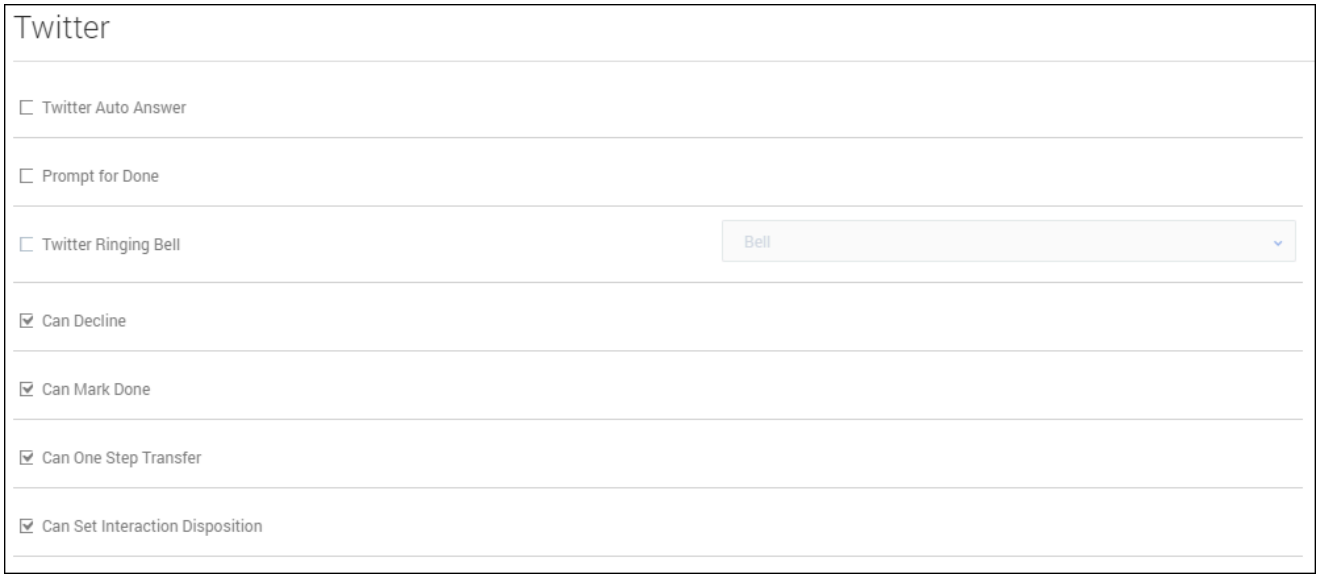
In the **Facebook** section under **Desktop Options**, configure the following options:

- **Facebook Auto Answer** automatically accepts a Facebook interaction when an Invite event if received.
- **Prompt for Done** prompts a confirmation message when the agent clicks **Done**.
- **Facebook Ringing Bell** specifies the sound played when a Facebook interaction is ringing.
- **Decline** allows the agent to decline an incoming interaction. This option depends on the Channels option for Facebook.
- **Mark Done** allows the agent to mark an interaction as **Done** with further processing. This option depends on the Channels option for Facebook.
- **One-step transfer** allows the agent to use one-step transfer.
- **Set Interaction Disposition** allows the agent to set a disposition code before marking an interaction as **Done**. This option depends on the Channels option for Facebook.

Twitter options

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

In the **Twitter** section under **Desktop Options**, configure the following options:

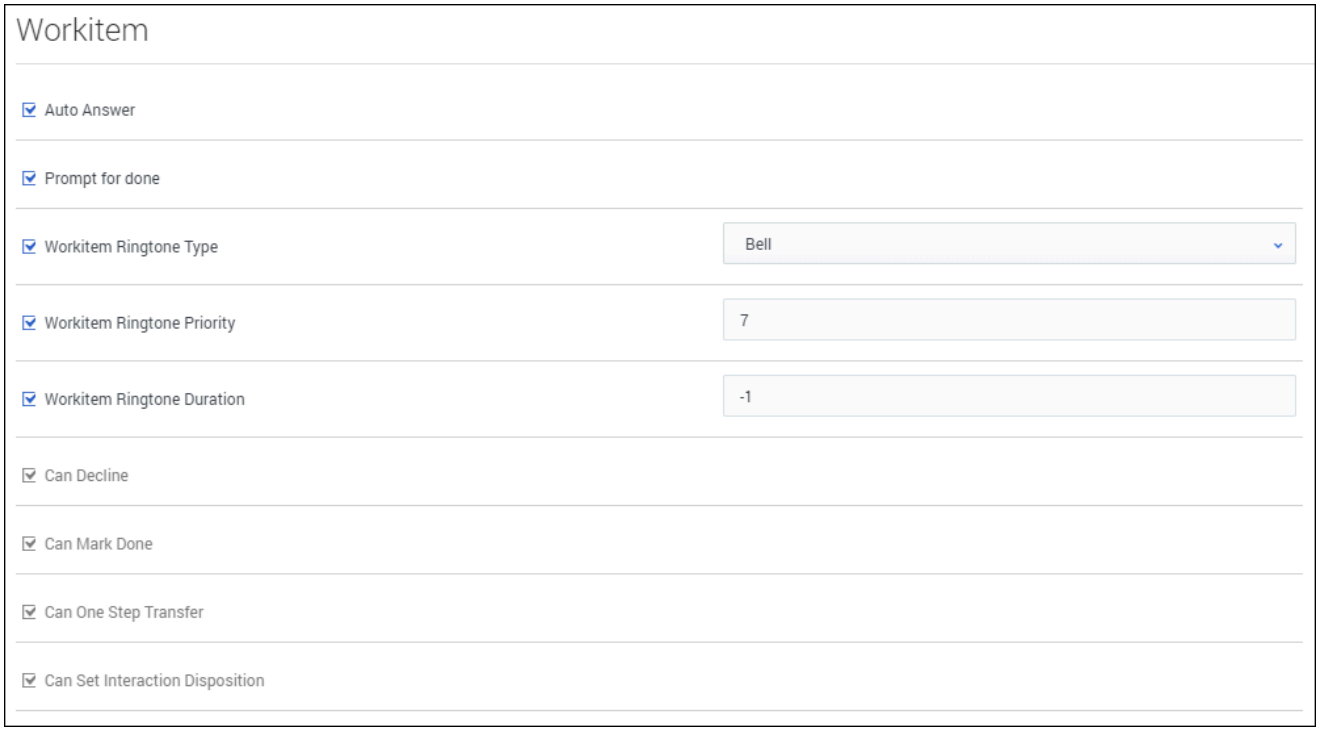


- **Twitter Auto Answer** automatically accepts a Twitter interaction when an Invite event if received.
- **Prompt for Done** prompts a confirmation message when the agent clicks **Done**.
- **Twitter Ringing Bell** specifies the sound played when a Twitter interaction is ringing.
- **Decline** allows the agent to decline an incoming interaction. This option depends on the Channels option for Twitter.
- **Mark Done** allows the agent to mark an interaction as **Done** with further processing. This option depends on the Channels option for Twitter.
- **One-step transfer** allows the agent to use one-step transfer.
- **Set Interaction Disposition** allows the agent to set a disposition code before marking an interaction as **Done**. This option depends on the Channels option for Twitter.

Workitem options

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

In the **Workitem** section under **Desktop Options**, configure the following options:



- **Auto answer** automatically accept an interaction when an invite event is received.
- **Prompt for done** presents a confirmation message to the user when they press the **Done** button. This option is only available for Open Media interactions.
- **Workitem ringtone type** specifies the sound that is played when a Workitem interaction is ringing.
- **Workitem ringtone priority** specifies the priority of a Workitem ringtone. The higher the integer, the higher the priority.
- **Workitem ringtone duration** specifies the duration of a Workitem ringtone (-1 plays and repeats the

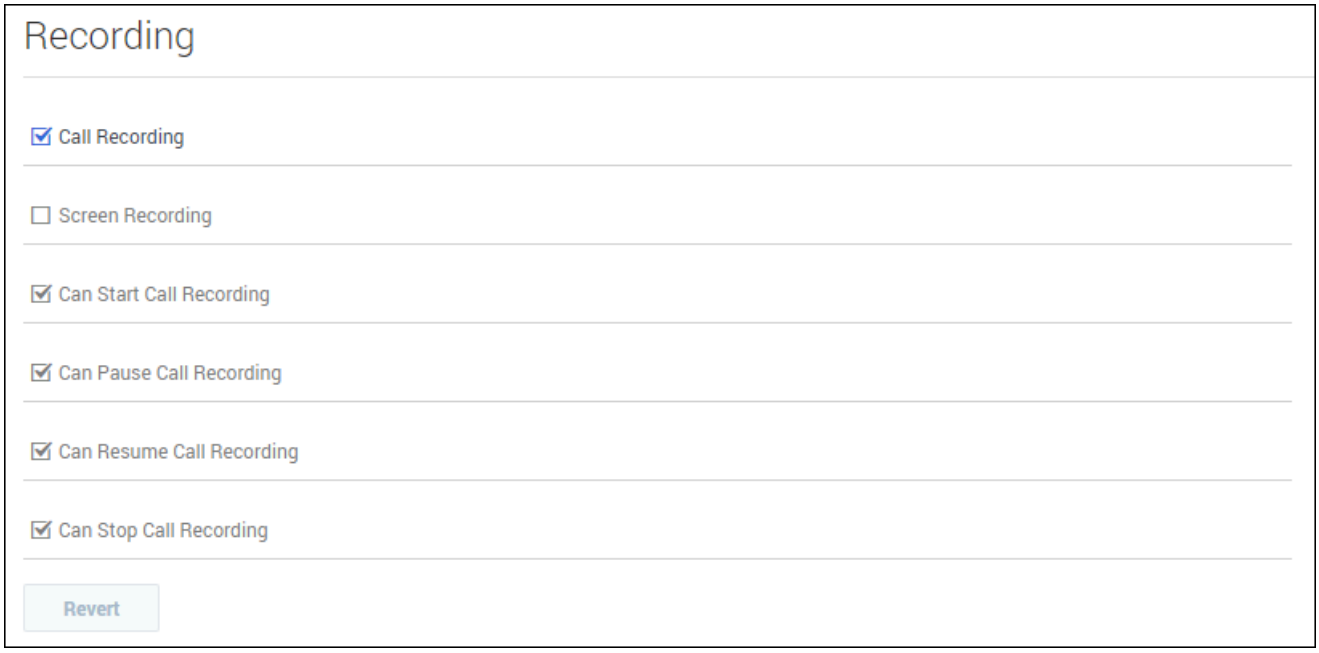
sound until an action is taken, 0 play the whole sound one time, and an integer > 0 sets a time in milliseconds to play and repeat the sound).

- **Decline** enables the agent to decline incoming Workitem interactions.
- **Mark Done** enables the agents to mark done a Workitem interaction without further processing.
- **One-step transfer** enables the agent to use instant Workitem transfer.
- **Interaction Disposition** enables the agent to set the disposition codes for Workitem interactions.

Recording options

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

In the **Recording** section, you can configure the following options:

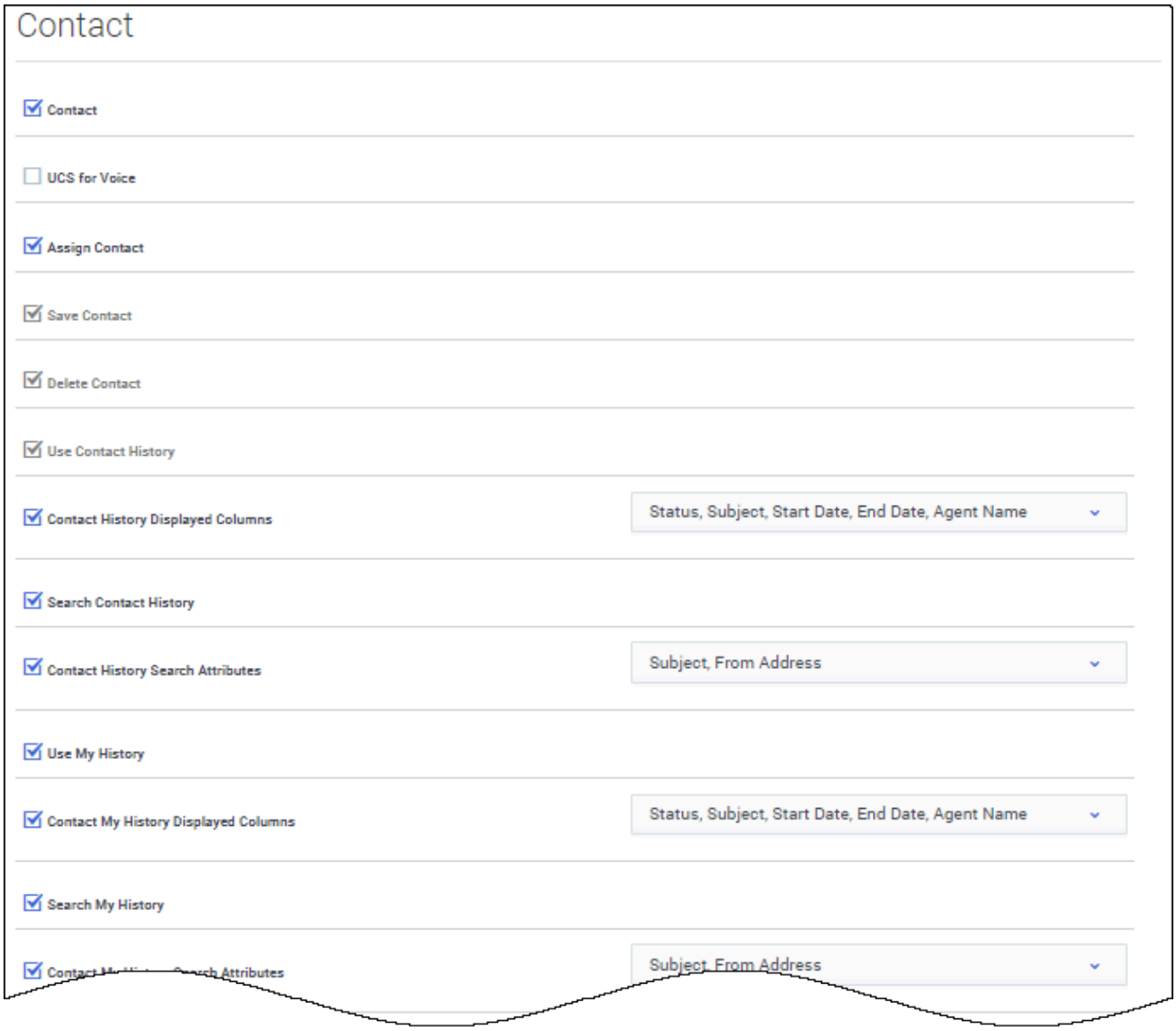


- **Call Recording** sets the agent's desktop to record all voice interactions.
- **Screen Recording** sets the agent's desktop to record all screen interactions.
- **Can Start Call Recording** allows the agent to start an active recording.
- **Can Pause Call Recording** allows the agent to pause an active recording.
- **Can Resume Call Recording** allows the agent to resume an active recording.
- **Can Stop Call Recording** allows the agent to stop an active recording.

Contact options

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

In the **Contact** section of the **Contact Center Settings** tab, you can enable the following options:



- **Contact** enables agents to access contact features.
- **UCS for Voice** enables the Universal Contact Server to generate voice interaction history in Agent Desktop. This option is enabled when the **Contact** and **Voice** check boxes are selected.
- **Assign Contact** enables agents to assign contacts to an interaction. This option is enabled when the **Contact** check box is selected.
- **Save Contact** enables agents to modify contacts.
- **Delete Contact** enables agents to delete contacts.
- **Use Contact History** enables agents to access contact history. This option is enabled when the **Contact** check box is selected.
- **Contact History Display Columns** allows you to manage the interaction attributes you want to display in the **Contact History** view in Agent Desktop. This option is enabled when the **Contact** and

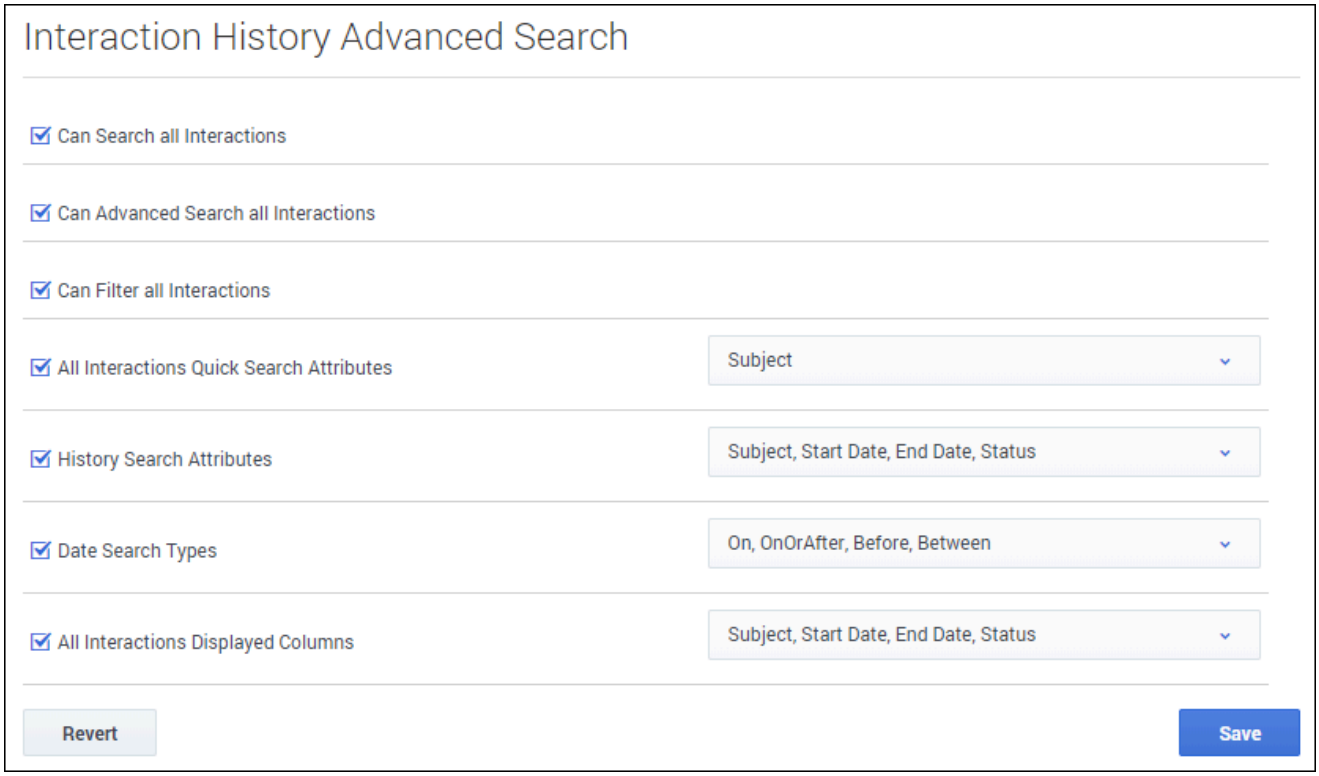
Use Contact History check boxes are selected.

- **Search Contact History** enables agents to search contact history. This option is enabled when the **Contact** and **Use Contact History** check boxes are selected.
- **Contact History Search Attributes** allows you to manage the interaction attributes you want to display in the **Contact History Quick Search** view in Agent Desktop. This option is enabled when the **Contact**, **Use Contact History**, and **Search Contact History** check boxes are selected.
- **Use My History** enables agents to access their own contact history. This option is enabled when the **Contact** check box is selected.
- **Contact My History Displayed Columns** allows you to manage the interaction attributes you want to display in the **My History** view in Agent Desktop. This option is enabled when the **Contact** and **Use My History** check boxes are selected.
- **Search My History** enables agents to search their own contact history. This option is enabled when the **Contact** check box is selected.
- **Contact My History Search Attributes** allows you to manage the interaction attributes you want to display in the **My History Quick Search** view in Agent Desktop. This option is enabled when the **Contact**, **Search My History**, and **Search My History** check boxes are selected.
- **Open from History** allows agents to open in-progress emails from the interaction history. This option is enabled when the **Contact** and **Email** check boxes are selected.
- **Use Directory** enables agents to view and search the contact directory. This option is enabled when the **Contact** check box is selected.
- **Contact Directory View** specifies the format for the Contact History View. Options include **List View** or **Grid View**. This option is enabled when the **Contact** check box is selected.
- **Contact Notepad** allows an agent to display and edit notes while handling an interaction. The notepad is a feature that enables agents to enter comments about the current interaction or about a selected interaction. Agents can view the note in most channels as well as contact history, My History, and My Workbin.
- **Use Contact History Notepad** allows an agent to see notes about interactions in Contact History.

Interaction History Advanced Search options

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

In the **Interaction History Advanced Search** section, you can configure the following options:



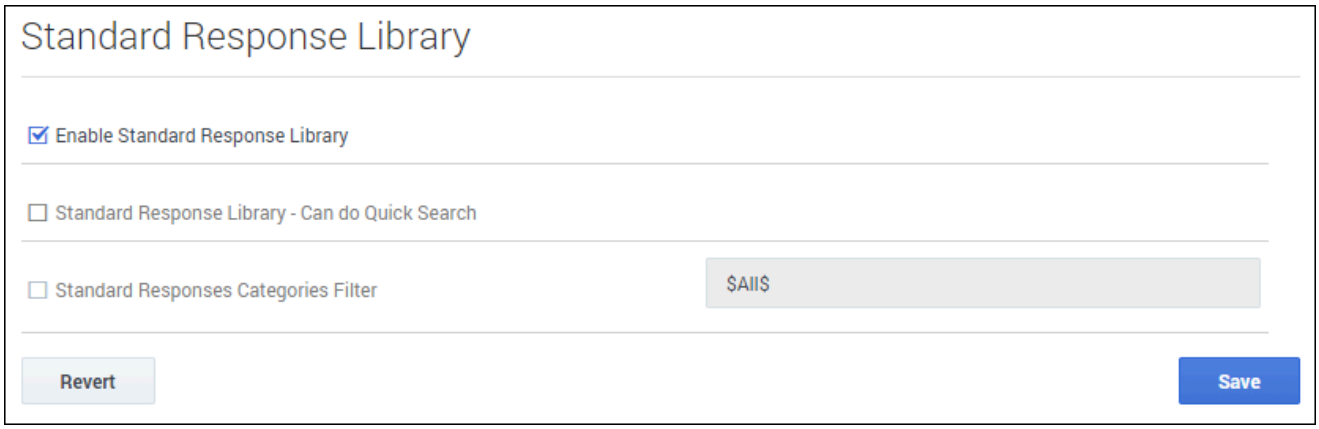
- **Can Search all Interactions** allows an agent to search among all interactions.
- **Can Advanced Search all Interactions** allows an agent to conduct an advanced search among all interactions.
- **Can Filter all Interactions** allows an agent to filter all interactions.
- **All Interactions Quick Search Attributes** specifies the interaction attributes used to search

interactions in quick search mode of the Interaction Search view.

- **History Search Attributes** specifies the list of interaction attributes an agent can use in the Advanced Search mode of the Contact History, My History and Interaction Search views.
  Search attributes include Subject, Start/End Date, Status, Contact, From Address, Phone Number, Released Date, Sent Date, Notes, and Interaction Type.
- **Date Search Types** specifies the search types available to search the contact database by date. Types include On, On or After, Before, and Between.
- **All Interactions Displayed Columns** specifies the interaction attributes displayed in the result list of the Interaction Search view.

Standard Response Library options

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

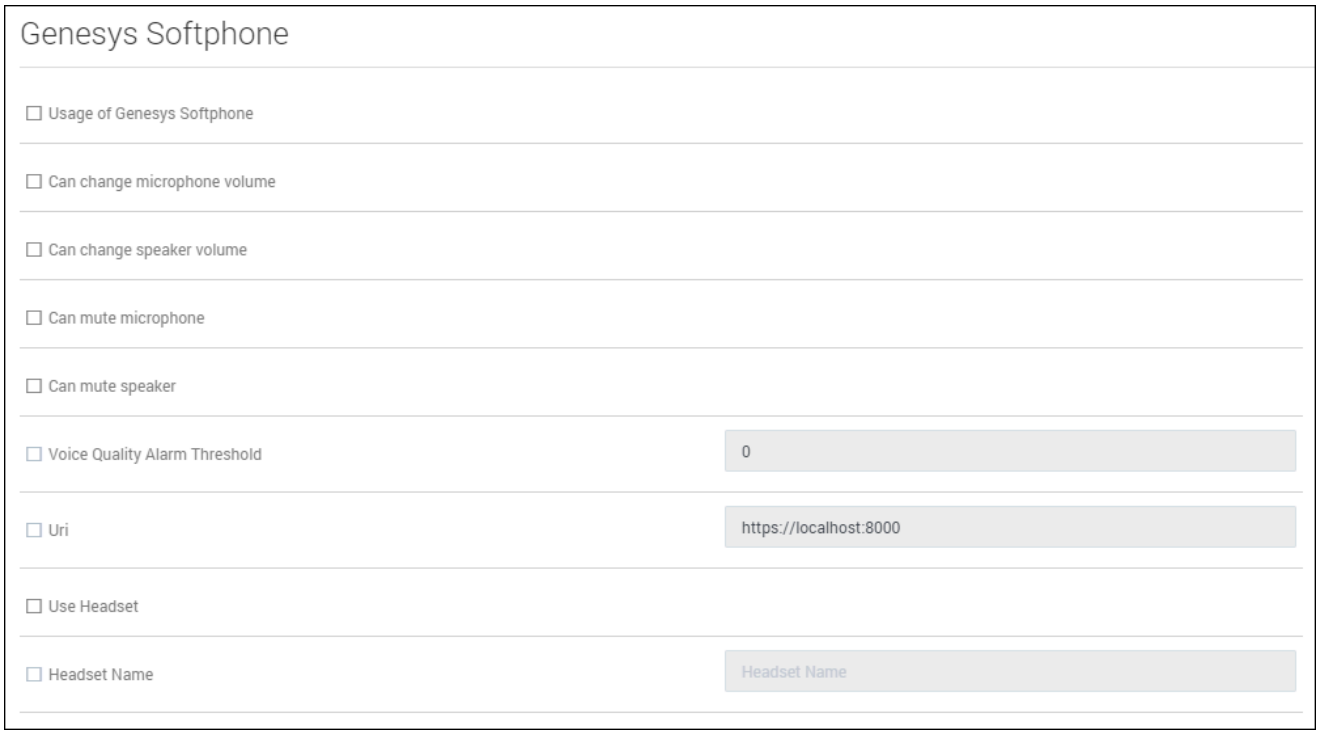


In the **Standard Response Library** section, you can enable or disable the Standard Response Library and the ability to Quick Search within the library. You can also apply a filter to specify what response categories are available to the agent in Agent Desktop.

Genesys Softphone options

> **Important**
>
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

In the **Genesys Softphone** section of the **Contact Center Settings** page, you can enable the following options:



- **Usage of Genesys Softphone** enables the use of Genesys Softphone.
- **Can change microphone volume** allows the agent to control the volume of their microphone.
- **Can change speaker volume** allows the agent to control the volume of their speaker.
- **Can mute microphone** allows the agent to mute and unmute their microphone.
- **Can mute speaker** allows the agent to mute and unmute their speaker.
- **Voice Quality Alarm Threshold** specifies the mean opinion score (MOS — a measure of reported network quality ratings) threshold for generating Voice Quality Alarms. The value `0` disables the alarms.

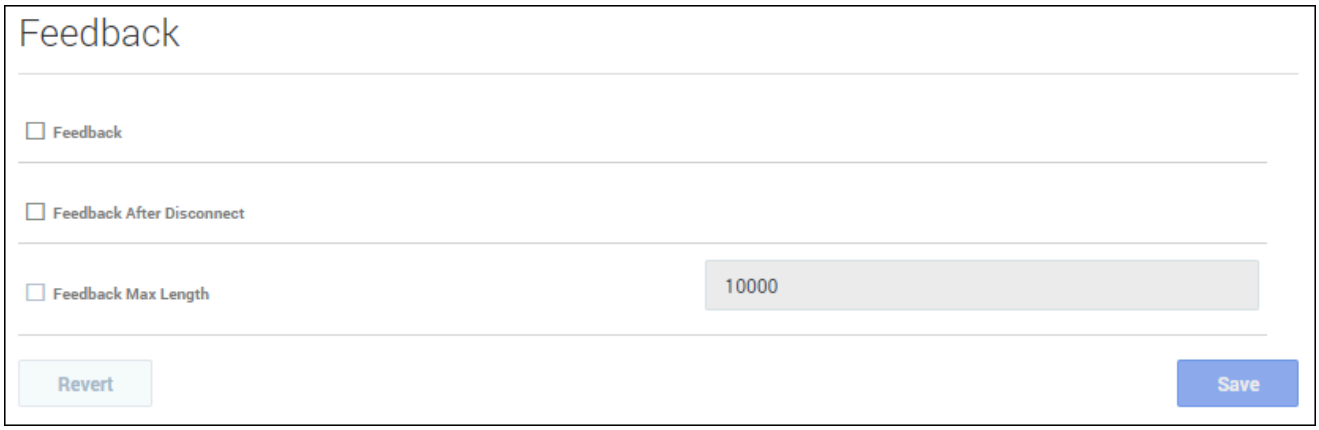
The recommended threshold value is `3.5`.

- **URI** specifies the URI of the SIP endpoint.
- **Use Headset** enables the use of a headset for voice calls.
- **Headset name** specifies the name of the headset model if **Use Headset** is enabled.
- **Can Use WebRTC** - Enables WebRTC in Workspace. Depends on 'Voice - Can Use Voice Channel'.
- **Can change speaker volume** - Allows an agent to change speaker volume. Depends on 'Voice - Can Use Voice Channel' and 'WebRTC - Can Use WebRTC'.
- **Can mute microphone** - Allows an agent to mute and unmute the microphone. Depends on 'Voice - Can Use Voice Channel' and 'WebRTC - Can Use WebRTC'.
- **Can mute speaker** - Specifies if DTMF is available for WebRTC calls. Depends on 'Voice - Can Use Voice Channel' and 'WebRTC - Can Use WebRTC'.
- **Can send DTMF** - Specifies if DTMF is available for WebRTC calls. Depends on 'Voice - Can Use Voice Channel' and 'WebRTC - Can Use WebRTC'.
- **Quality request interval** - Specifies the voice channel ringing sound-configuration string. It consist of three options:
  - **WebRTC Ringtone Type** - Set WebRTC call ringing sound. Depends on 'Voice - Can Use Voice Channel' and 'WebRTC - Can Use WebRTC'.
  - **WebRTC Ringtone Priority** - Set WebRTC call ringing priority between different channels. Depends on 'Voice - Can Use Voice Channel' and 'WebRTC - Can Use WebRTC'.
  - **WebRTC Ringtone Duration** - Set WebRTC call ringing duration. Depends on 'Voice - Can Use Voice Channel' and 'WebRTC - Can Use WebRTC'.

Feedback options

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

In the **Feedback** section, you can enable the following options:



- **Feedback** enables the **Feedback** icon to be displayed on the **About** dialog box.
- **Feedback after disconnect** enables the **Feedback** dialog box to be displayed when a user's browser unexpectedly disconnects. The **Feedback** dialog box is displayed when the user logs back in.
- **Feedback Max Length** specifies how many characters are accepted in the Feedback dialog box.

When Agent Desktop or Gplus Adapter doesn't work as expected, Genesys recommends that agents report the incident in the following ways:

- Use the **Feedback** options to capture logs and details about what the agent was doing in their most recent Agent Desktop or Gplus Adapter session. These logs are sent directly to Genesys and are available to Customer Care for later reference. To provide feedback reports to Genesys, depending on what you enable in the **Feedback** section of the **Desktop Options** page, agents can use either the:
  - **Feedback** option on the **Help** menu
  - **Feedback** dialog box that is displayed after the web browser unexpectedly disconnects and reconnects
- Afterwards, report the incident to their direct supervisor or administrator in person, following your organization's reporting policies and procedures.

When an agent reports an issue to a supervisor or administrator, that person should contact Genesys Customer Care as soon as possible to provide the reporting user's name and DN at the time of the incident, as well as the approximate time the incident occurred. At this point, the Customer Care agent should be able to retrieve the relevant logs that were sent through the **Feedback** options to determine what the issue might have been.

My Channels options

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

In the **My Channels** section, you can enable the following options:

- **My Channels** allows an agent to access the My Channels tab.
- **My Channels Pending State** allows an agent to have a pending state in My Channels.

Service Client options

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

Before using the Service Client API to customize how your web application or website integrates with Agent Desktop, first set the Service Client options in Agent Setup.

In the **Service Client** section of the **Contact Center Settings** page, configure the following options:

- **Block Mark Done Timeout** specifies the duration of timeout in milliseconds. This timeout duration is started by the function `interaction.blockMarkdone()`.
- **List of User Data Read Allowed** specifies the list of keys that can be read in the user data. This applies to the UserData property of the interaction object returned by a function or an event.
- **List of User Data Write Allowed** specifies the list of keys that can be written to when using functions `interaction.setUserData()` and `interaction.deleteUserData()`.
- **Auto Login** returns an agent to their previous state if they are unexpectedly logged out of Agent Desktop.

Agent accounts

> **Important**
>
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

On the **Users** tab, you can manage agent accounts, including adding new accounts, importing agent accounts from an external file, and exporting agents.

> **Important**
>
> - To perform most tasks on user accounts, your account must have Administrator privileges. If you are unable to perform certain actions using the Agent Setup interface, check that your account privileges are correct.
> - To add administrator accounts to your contact center, you can either import the administrator accounts or use Platform Administration.

After you log in to the Agent Setup interface, the first screen that you will see is the **Users** tab. If your contact center already has agents assigned to it, you see a list of configured agents. In a fresh installation, you might see an empty list. Regardless, as part of your regular maintenance on your contact center, you need to add agents.

How do I add a single agent?

Link to video

Click the **Add User** link. You can select a template from the **Template** drop-down list box if you have configured one or more templates. A template automatically selects some of the options on the **Add User** screen.

Complete all the required (*) fields on the screen. If you make an error, a message is displayed to tell you the correct format for a field. Most fields are self-explanatory, such as the name and email fields.

This video demonstrates how to add a single agent.

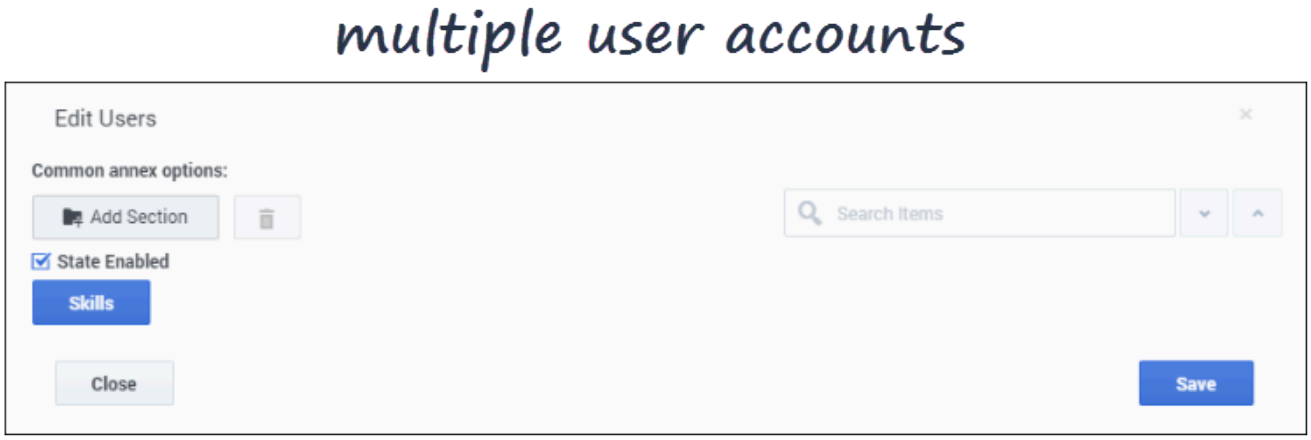
Use the following table for advice on how to fill in the fields:

Option	Description
Folder	From the drop-down list box, select the business group to which you want to assign the user.
First Name	Type the first name of the user.
Last Name	Type the last name of the user.
Username	Type the username for the user.
Login Code	Optional. Type a login code for the user. An example of a login code is the user's employee ID. Important The **Login Code** option is hidden by default, therefore, it might not be displayed on your screen. If you require the **Login Code** option, contact Genesys Professional Services to request that they enable the Agent Login Code option.
Email	Optional. Type the user's email address.
Reset Password	When selected, this option specifies that when the next time the user logs in, he or she will be prompted to change the password. This option only applies to agent user accounts.
Password	Type a default password for the user.
Password confirm	Type the password again.
Employee ID	Optional. Type the user's employee ID.
External ID	Optional. Type the user's external ID.
Capacity Rule	Optional. Choose an existing Capacity Rule from the drop down.
State enabled	Enables or disables the user account.
Multimedia Agent	Specifies that this agent is only enabled for multimedia (non-voice).
Supervisor	Select this option if this agent has a supervisory role.
Phone Number(s)	In addition to the default phone number, you can assign additional phone numbers to the agent. The agent has the opportunity to select the number when they log in to accept calls. For example, if you type `hjackson` in the **Username** field, when you type a phone number in the **Number** field of the first row, the **Place Name** field displays `hjackson`. If you type a number in the second row, the associated `Place Name` field displays `hjackson2` and the place name in the first row changes to `hjackson1`. If your contact center has a requirement to specify unique place names for different business groups or regions, you can overwrite the default place name.
Place Name	The value that you type in the **Username** field automatically populates the **Place Name** column. If you don't want to use the username as a place name, clear the **Use** check box.
Use	The **Use** option specifies that the username is to

Option	Description
	be used for the associated phone number. This option is selected by default. When you clear the **Use** check box, the phone number is used as a place name.
Wrap Up Time	You can configure wrap up time for an agent. Wrap up time is the amount of time the agent is allotted to complete after-call tasks, such as typing notes or selecting a disposition. The wrap-up time is measured in seconds. To disable wrap up time, type 0.
Agent Voice Mail	Type the voice mail box extension number for the agent to use.
Softphone	Select this check box if this user will use a softphone.
SIP Phone Type	When you enable Softphone, you must select a softphone type.
Skills	Select the skills that you want to assign to the agent, assign a skill level, and click **Add**. With many skills to choose from, you can filter the list of skills by using the search box.
Caller IDs	Click Caller ID and add at least one identifier to be displayed on a receiver's call display. If you add more than one, the agent has the option to choose which caller ID they want to display. You can also enable the agent to use an anonymous caller ID.
Agent Groups	Assign this agent to an agent group, which is a logical grouping of agents that provide particular sets of contact center services. For more information about agent groups, see Managing Agent Groups.
Annex	Assign this agent to an access group, which is a logical grouping of agents that are provided specific access permissions.
Favorites	Personal Favorites is a list of Agents, Agent Groups or external contacts that group members frequently dial. This list provides a click-to-call list of these numbers on Agent Desktop. Personal favorites are configured the same way as Global Favorites. See Global Favorites.
Switches	Select the switches where DNs and Agent Logins should be provisioned.
External URLs	Integrate internal and external websites into your agent's main view. See External URLs.
Filter	For each agent group, you can control what information that group can see from other agent groups in the Team Communicator, Favorites, and Statistics. After you select an agent group on the Agent Groups window, select the Filter option on the menu bar. In the list of agents groups, you can choose to restrict or exclude agent groups by

Option	Description
	selecting the appropriate check boxes. You can also manage filters on the **Desktop Options** page. For more information, see Desktop Options.
Desktop Options	Assign channels and other options to the agent by clicking **Desktop Options** and then selecting from the listed options. You can select more than one option. For a detailed look at what options you can configure for an agent, see the Desktop Options.
CRM Adapter	**CRM Adapter**: Use the **CRM Adapter** tab to configure the Gplus Adapter option settings for the agent group. For information, see CRM Adapter settings.
Recording Hierarchy	The recording hierarchy for an agent specifies the access control applied to recordings of calls handled by that agent. Select the Recording Hierarchy check box, and then type the hierarchy string. See Access control for Recording users for more information about Recording agent hierarchy, access groups, and partitioning.

Editing agents



You can edit a single agent or a group of agents. When you edit a single agent, you can edit any of the agent's details. For a group edit, the two options available for you to edit are Annex and Skills.

When you edit one agent, click the agent's name in the Name column. In the **Manage User** window, you can make your updates.

To edit multiple users at once, select the agents that you want to edit, click **Edit Users**, and then make your changes in the **Edit Users** window. When you edit multiple users, consider the following:

- You can add Annex options or modify existing options. You can only delete Annex options that are identical for all selected agents.
- Only values that are common for the selected users are shown for Skills and Annex. For example, if:
  - Agent 1 has Russian skill level of 10 and English skill level of 5

- Agent 2 has Russian skill level of 10 and English skill level of 7

  You will see the Russian skill level of 10 and English skill level is blank on the **Edit Users** screen under **Skills**.

Link to video

This video gives you a demonstration on how to update multiple agents.

Importing agent accounts

When you need to add a large number of agent accounts, you can use the **Bulk Import** tab in Agent Setup. By importing a large number of agent accounts at one time, you won't need to add the accounts one at a time and you can apply a template to configure options, such as DN, place, person, and login options, to all accounts.

How do I import a file that contains agent accounts?

Link to video

Before you can import agent accounts , you must have a .CSV file that has all the required data.

The values from the imported file overwrite corresponding values that are specified in the template. For example, the value for the **Skill** parameter in the imported file takes precedence over the value for the **Skill** parameter that is specified in the template.

When you are ready to import the file, click **Choose File**, select the file from your local device, and then click **Import**. If you want, you can validate the file to ensure that it is structured correctly.

About CSV files

The .CSV file must be a text file in a comma-separated format. In the source file each line represents a single user or agent.

When you import a .CSV file, each parameter value is updated according to the following rules:

- If value is present, the new value replaces the previous value.
- If an empty value is encountered, no change is made.
- If the value includes `/d` , the attribute is unassigned and deleted. If the value includes `/r` , the attribute is unassigned only.
- If no corresponding record is found, an error message is logged and the updated continues to process.

[+] Show .CSV file parameters

Property	Type	Mandatory?	Action	Description
Username	String	Yes	ADD, UPDATE, DELETE	The unique username of the agent or user.
First Name	String	Yes	ADD, * needed for UPDATE if First Name change	The first name of the agent or user.
Last Name	String	Yes	ADD, * needed for UPDATE if Last Name change	The last name of the agent or user.
Email	String	No		The agent or user's email address.
Extension	String	No	/d to unassign and delete /r to unassign only	The extension #1 of the agent. A place is also created if the extension is specified.
Extension2	String	No	/d to unassign and delete /r to unassign only	The extension #2 of the agent. A place is also created if the extension is specified.
Extension3	String	No	/d to unassign and delete /r to unassign only	The extension #3 of the agent. A place is also created if the extension is specified.
Folder	String	No		The folder path to root folder (Persons) where new agent or user is created. For example, Company/ Department/Team.
Place Name	String	No	/d to unassign and delete /r to unassign only	The place name #1 of the agent. The place name "Extension" is used if not specified.
Place Name2	String	No	/d to unassign and delete /r to unassign only	The place name #2 of the agent. The place name "Extension2" is used if not specified.

Property	Type	Mandatory?	Action	Description
Place Name3	String	No	/d to unassign and delete /r to unassign only	The place name #3 of the agent. The place name "Extension3" is used if not specified.
Default Place	String	No		The name of the Default Place.
Employee ID	String	Yes	ADD	The employee identifier of the agent.
Change Password	Boolean	No		Specifies that the user will be prompted to change the password on their next log in.
Enabled	Boolean	No		Y specifies that the agent account is enabled. N specifies that the agent account is disabled.
Is Agent	Boolean	Yes	ADD	• Specifies that this user is an agent. If this field is set to a Y but the Is Supervisor field is not, only the Agent Login, DN, and Places for the agent are created. If the Is Supervisor field is set to a Y, additional supervisor features and permissions are enabled. • If the user is an administrator, this field is not required.
External ID	String	No		Any external ID used to identify the user.
Login ID	String	No		The user's login ID.

Property	Type	Mandatory?	Action	Description
Is Supervisor	Boolean	No		Y specifies that this user is a supervisor. If this is the case, the Is Agent field must be set to Y. At that point, all required desktop logins and features are enabled, along with the required supervisor features and permissions.
Is Admin	Boolean	Yes		Y specifies that this user is an administrator. If this is the case, the Is Agent field (and Is Supervisor field) is not required.
Wrap Up Time	Numeric	No		The number of seconds an agent remains in ACW after call is disconnected. This field applies only to voice calls.
VoiceMail.MailBox	String	No		If specified, populate the following option to the Annex of all associated Agent log ins: AgentLoginAnnex:TServer/gvm_mailbox. Valid values: numerical (12345)
Recording Hierarchy	String	No		If specified, populate the following option to the Agent's Annex: [recording] agent_hierarchy.
SIP Phone Type	String	No		The type of SIP phone the agent is using as a softphone.
Skill:<skill_name>	Numeric	No		The number that identifies the agent's skill level

Property	Type	Mandatory?	Action	Description
				for the specified skill.
Switch:<switch_name>	Boolean	No		**Y** means that a DN and Agent Login are created for the agent on the specified switch.
AgentG:<agent_group_name>	Boolean	No		**Y** means that the agent is added to the specified agent group.
Annex	String	No		Any custom Annex value. For example, column name **Annex:htcc/roles** with a value of **Agent**.
Action	String	No		ADD adds a new agent. UPDATE updates an existing agent. DELETE removes an existing agent.

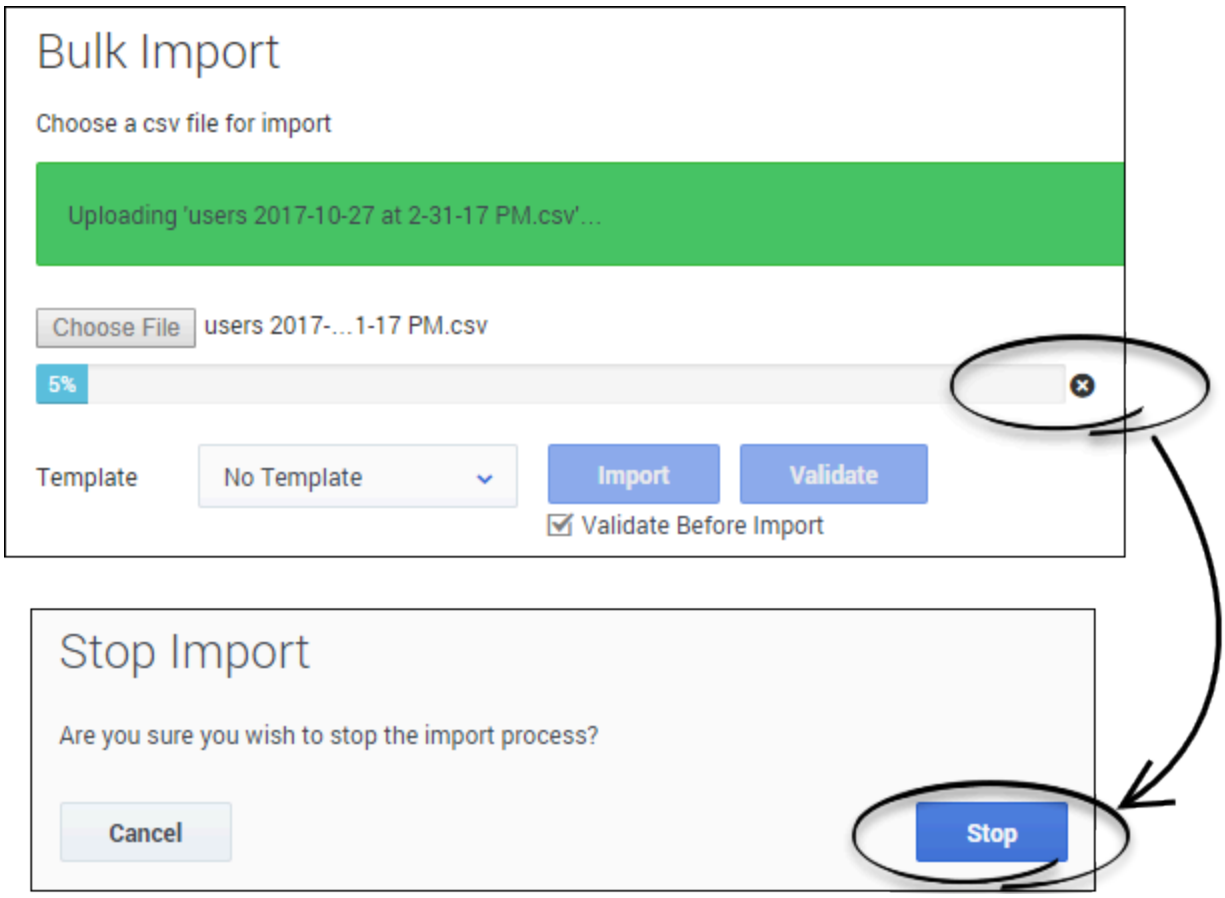
[+] Show an example

```
First Name,Last Name,Employee Id,Username,Is Agent,Is Supervisor,Is Admin,External
Id,Login ID,Extension,Skill:English,Switch: San Fran,Switch:
London,AgentG:Good,AccessG: Super,Annex,Action,Folder,Wrap Up Time,Enabled
Sarah,Lee,223344,sarahl@acme.com,N,Y,Y,ADD,Acme/Branch/Team 1,60
John,Doyle,223465,johnd@acme.com,Y,N,N,2233,1,Y,N,ADD,Acme/Branch/Team 2,0
Robert,Cook,244456,bobc@acme.com,Y,Y,Y,johnd@acme.com,5,Y,Y,ADD,Acme/Branch/Team
3,60,Y
```

Important

When performing a bulk import, the **Password** field is not included in the imported file. A default password is associated to each user account in the imported file. The format of the default password is: *<username>*@*<company-name>*.com-*<Month>*-*<year>*, where *<Month>* is the three-digit abbreviation for the month. For example, `hjackson@demo.com-Feb-2017`. The **Reset Password** option is automatically enabled when adding or importing new users, therefore, all users are prompted to reset their password the next time that they log in.

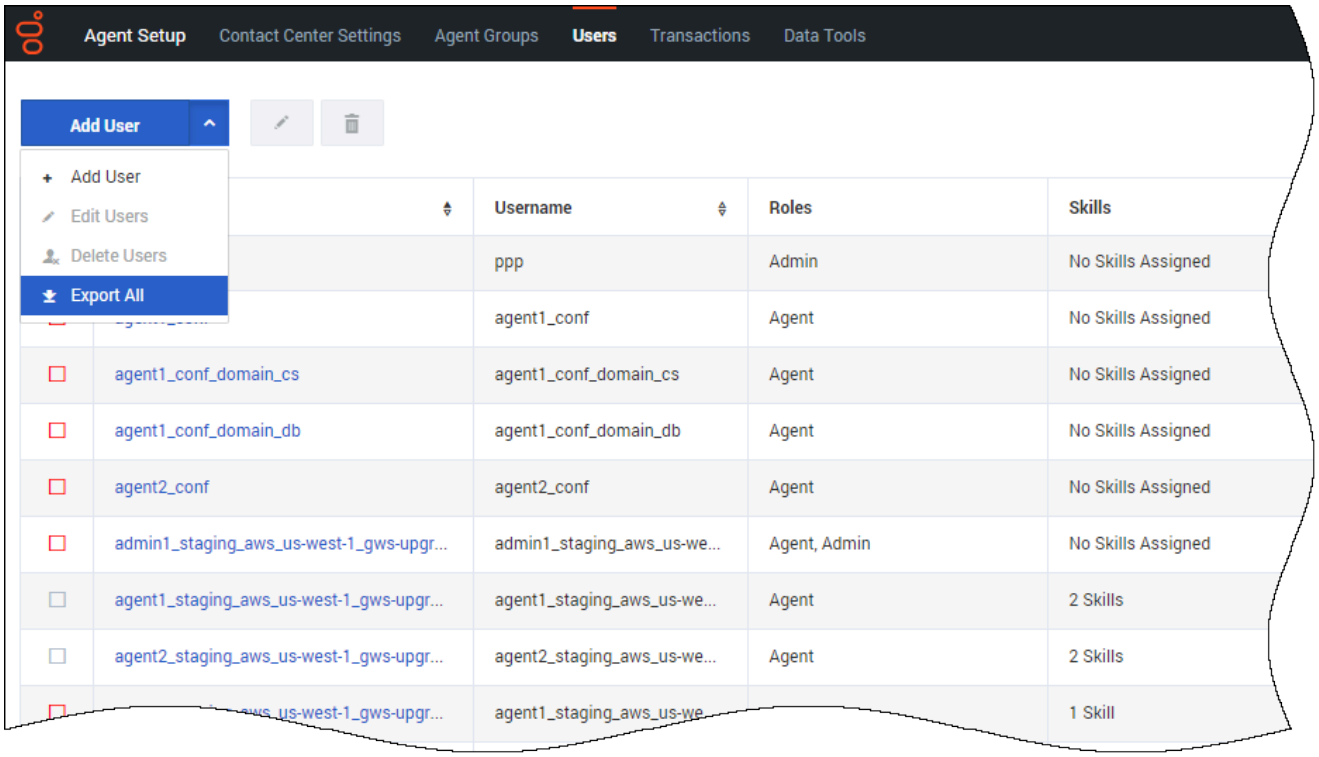
How do I stop an import?



When you stop an import, a window is displayed in which you can confirm that you really want to stop the import. After you click **Stop**, agents records that were imported before you clicked ⊗ are retained and the import process does not proceed.

Exporting agent accounts

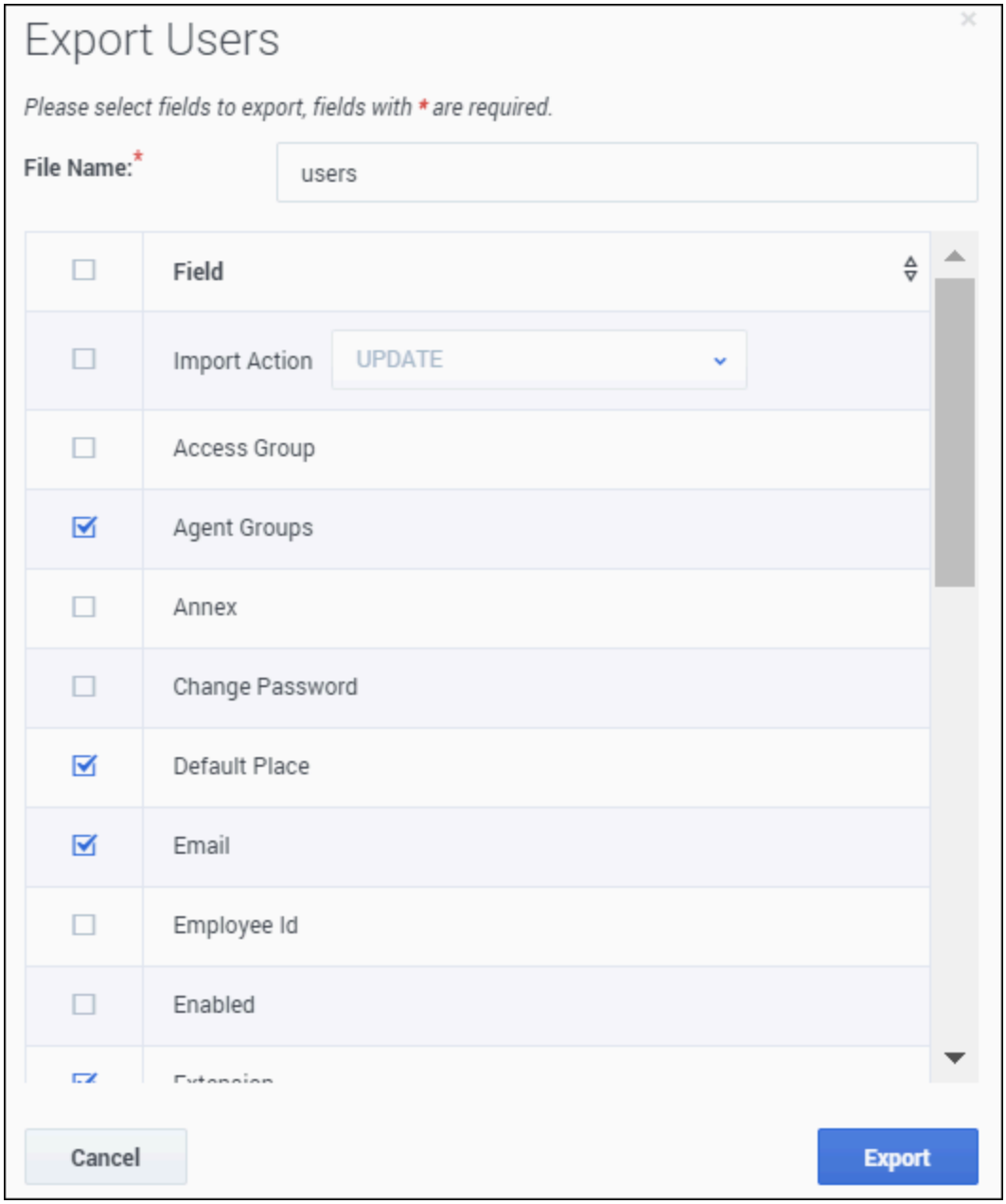
There may be circumstances where you need to create a backup file of all your agent accounts or you might want to export agent accounts to update them using another application and then import them back in. When you export agent records, a spreadsheet document is downloaded to your local computer.



On the **Users** page, you can export all users or a subset of users.

- **Export All**: Click the down arrow on the **User Actions** menu and select **Export All**.
- **Export Selected**: To select a subset of users, select the users and then select **Export Selected** from the **User Actions** menu. You can select the users by:
  - Click the check boxes next to the users you want to export.
  - Filter the list of users by using the search boxes at the top of the **Users** list and then click **Select All** or individually select the users you want to export.

How do I select the fields that I want to export?

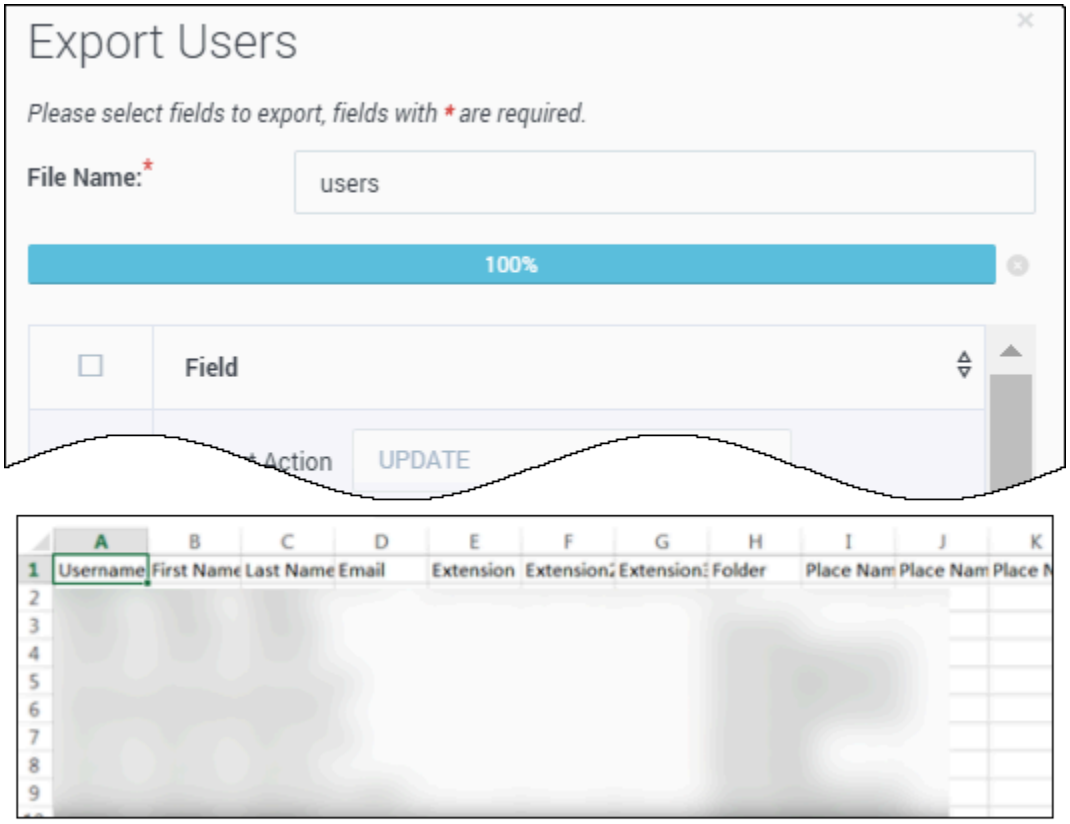


On the **Export Users** window, you can give your exported file a custom name. This makes it easier for you to locate on your local system after you export. You can either accept the default selections or customize the list. If you want to export all fields, select the check box at the top of the list, next to the **Field** label.

Important

- If you are exporting your user accounts as a means to modify the records and then import them back in, select the **Import Action** field. If you don't select this field, the .CSV file will not import correctly.
- If you are including Skills in your export, the skill level must be zero or greater for it to be listed in the export files.

Where did my export go?



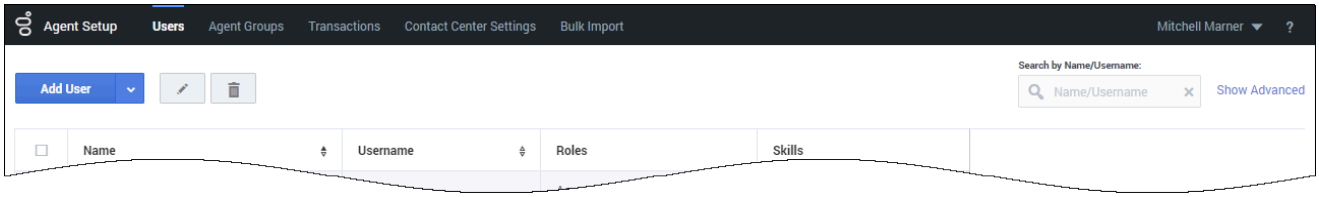
When the progress bar at the top of the **Export Users** window reaches 100 per cent, the document is completely downloaded to your local computer. How you access your downloaded files varies by operating system. In general, you can open your **Downloads** folder, and look for the **.csv** file that corresponds to the file name you specified.

You can then double-click the file to open it and view the exported data.

> **Important**
>
> If you use Microsoft Excel, the program automatically truncates all leading zeros from numbers in CSV files when you open the file by double-clicking on its name. You need to change at least the columns where the leading zeros occur to text format. For more information on how to resolve this issue, see this FAQ.

Locating agent accounts

What you see in the Agent Setup interface is determined by your permission settings.



> **Important**
>
> If you see a red check box next to an agent account, this means that the agent's account is not valid due to some missing information. Hover your mouse over the check box to see what information is missing. For example, if a voice agent does not have a configured default **Place Name** field and your contact center is not configured to prompt the agent to enter place when he or she logs in, the agent's account is considered invalid until you update the agent account to supply the missing information.

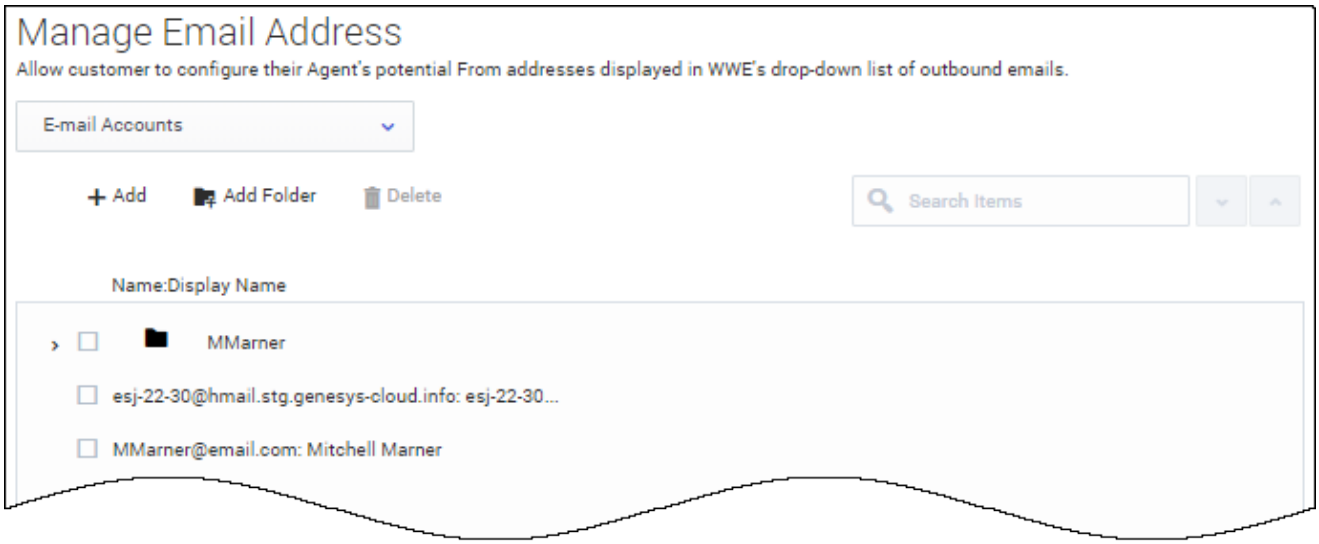
Note the following options that you can use to locate specific agents on the list:

- **Search by Name/Username**: Use this field to type an agent name or username, or part of an agent's name, to filter the list to display only agents that match your search words.
- **Agent Groups**: Use this field to type the name of an agent group. All agents that are assigned the specified agent group will be listed.
- **Roles**: Use this drop-down list box to filter the list to display only agents that are assigned a specific role.
- **Skills**: Use this drop-down list box to filter the list to display only agents that are assigned a specific skill.
- **State**: Use this drop-down list box to filter the list to display either All, Enabled, or Disabled.
- **Page navigation**: To scroll through the list pages, use the back and next arrow next to the page number at the bottom of the page.

Manage Email Addresses

> **Important**
>
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

Email Address Management in the **Digital Management** sub-section of the **Users** section, allows you to set up additional From email addresses for your agents to use in Agent Desktop.



You can configure your agent's potential From addresses displayed in Agent Desktop drop-down list of outbound emails.

When setting up additional addresses, you must first select or create the Business Attribute folder that will contain the Attribute Values that are used as available addresses. These come from the addresses of email interactions.

Once your intended Business Attributes folder is selected, you can begin adding new From email addresses by entering the address **Name** and an associated **Display name**. Once you have updated the users profile, the new From email addresses will be available for the agent to use when they start a new session or restart their current session.

Capacity rules

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

Capacity rules allow you to define an agent's ability to handle multiple interactions concurrently for different channels. When the limits for a channel are reached, the agent stops receiving calls/ interactions for that channel.

When you create a capacity rule, you define rules for every enabled channel, except for workbins and outbound channels. For each channel, you first need to set the maximum number of interactions an agent can handle, and then you can optionally define rules that also set limits for that channel based on the number of interactions being handled on other channels.

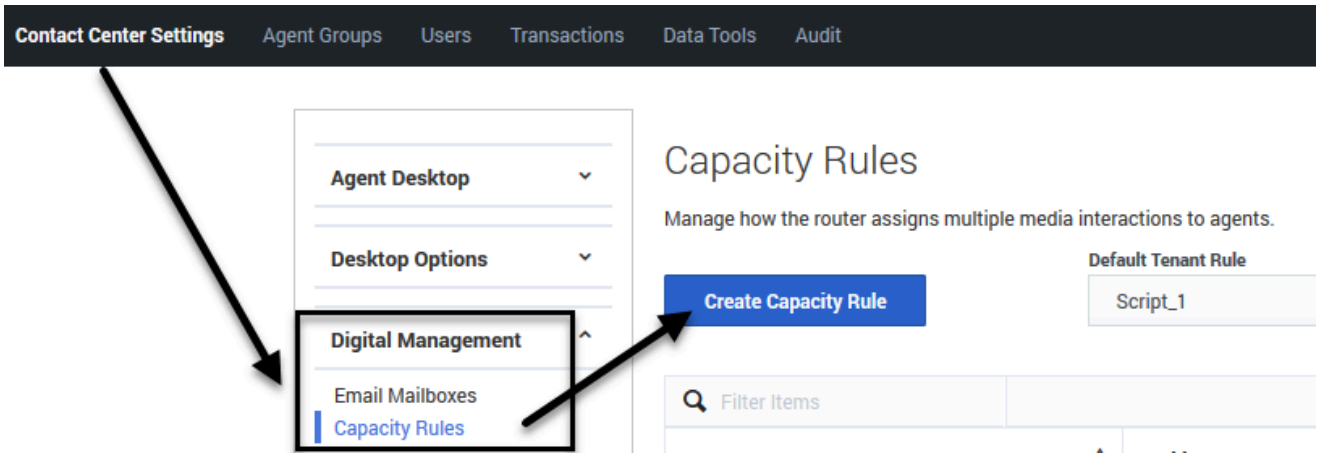
Example	Result
You could define the following for an agent using a chat channel: • Allow a maximum of 3 chat interactions.	This rule means that the system will not route new chat interactions to an agent under the following conditions: • If the agent is handling 3 or more chat interactions.

You could define the following for an agent using a chat channel:
• Allow a maximum of 3 chat interactions.
• Do not send new chat interactions when the agent is busy with either:
  • 1 email
  • 1 voice
	This rule means that the system will not route new chat interactions to an agent under the following conditions:
	• If the agent is handling 3 or more chat interactions.
	• If the agent is handling 1 or more email interactions.
	• If the agent is handling 1 or more voice calls.

You could define the following for an agent using a chat channel:
• Allow a maximum of 3 chat interactions.
• Do not send new chat interactions when the agent is busy with either:
  • 1 emailand 1 chat concurrently
  • 1 voice
	This rule means that the system will not route new chat interactions to an agent under the following conditions:
	• If the agent is handling 3 or more chat interactions.
	• If the agent is already handling *at the same time* 1 or more email interactions *and* 1 or more chat interactions.
	• If the agent is handling 1 or more voice calls.

Example	Result
(**Note:** This is considered advanced functionality and is used in rare circumstances)	

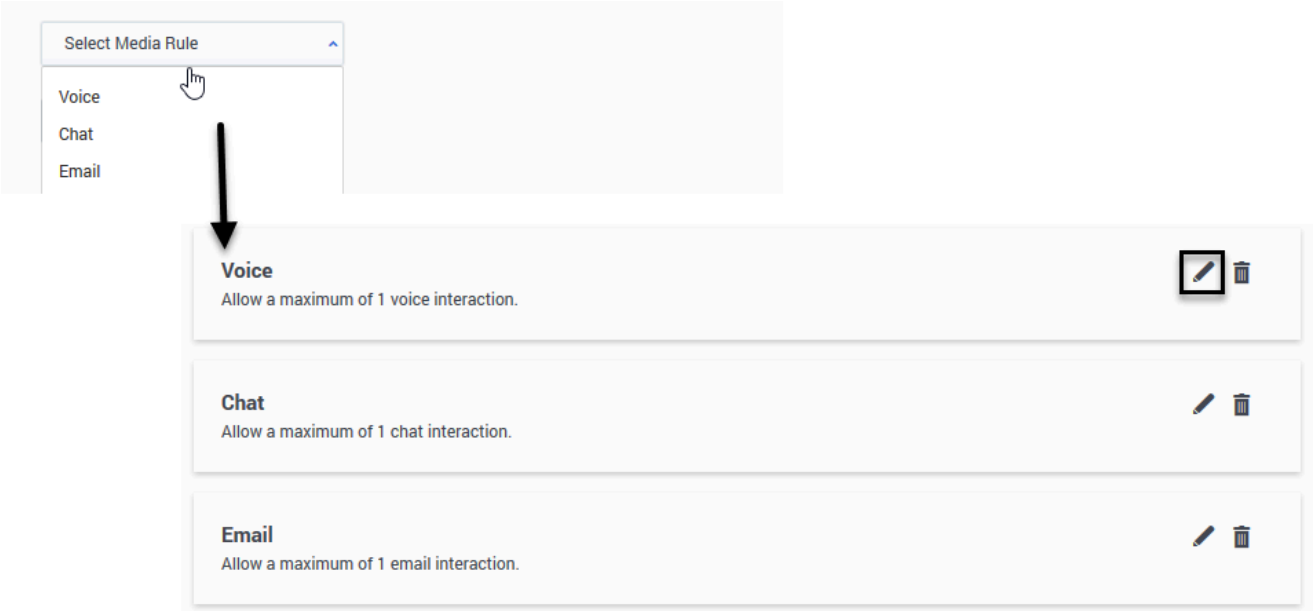
Create a Capacity Rule



To create a capacity rule:

1. Go to **Contact Center Settings** -> **Digital Management** -> **Capacity Rules** screen in Agent Setup.
2. Click **Create Capacity Rule**.

Select media channels



On the **Create a Capacity Rule** screen, enter a name for the capacity rule and specify the rule folder.

Next, from the **Select Media Rule** menu, select all channels that apply to your contact center.

When you select a channel, a default media rule for that channel appears onscreen. To open and edit the rule, click the pencil icon.

Now, you'll create the media rules by specifying the following:

- the maximum allowable interactions an agent can handle for a media channel.
- (optional) add limit conditions for that channel, based on interactions being handled on other channels.

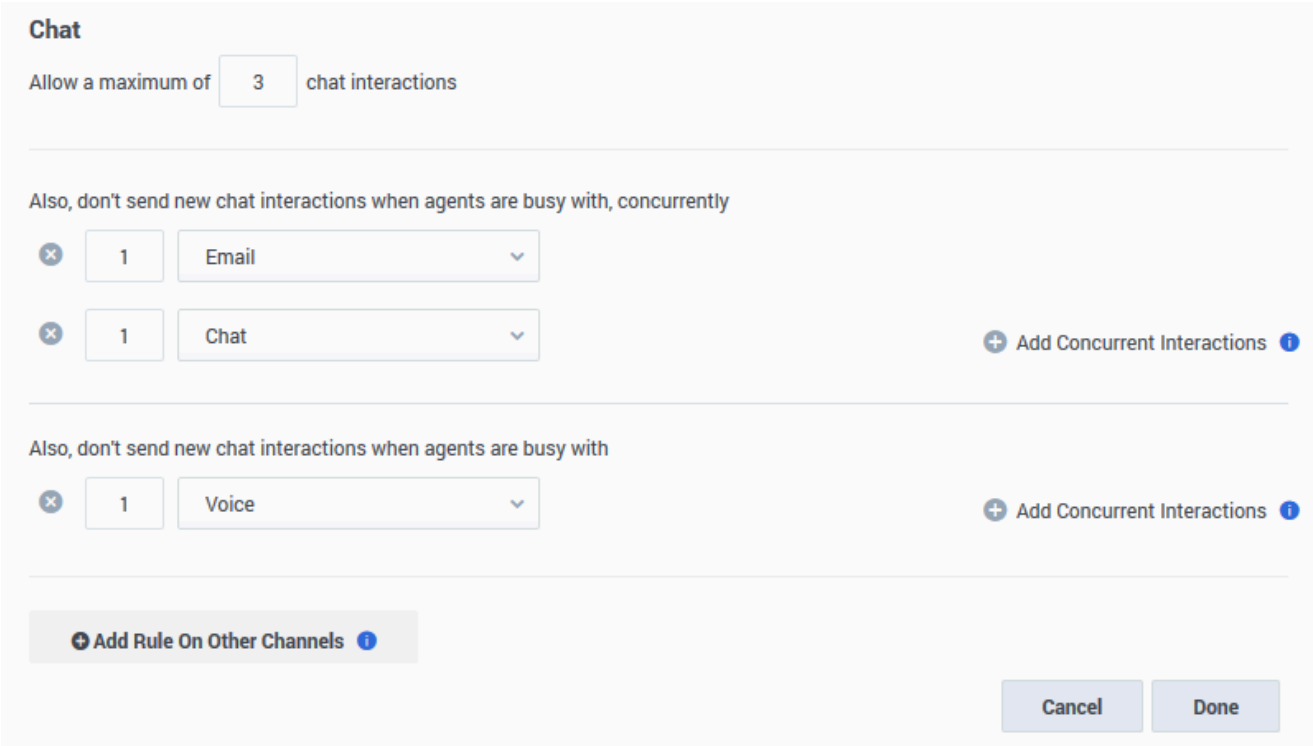
Define maximum number of interactions

To define the maximum allowable number of interactions an agent can handle for a media channel, click the pencil icon to open and edit the media rule for the respective channel.

Next, in the **Allow a maximum of <channel> interactions** section, simply enter an integer in the text box to define the maximum allowable number of agent can handle for that media channel.

You can click **Done** or continue on to define additional rules that depend on other channels.

Specify limits that depend on other channels



To specify rules that depend on the number of interactions of other channels being handled: Click **Add Rule on Other Channels**. Next, enter the number of interactions and select the channel that you want to depend on.

You can repeat and click **Add Rule on Other Channels** button to add more limit conditions on other channels.

Advanced, rarely used: Sometimes, you may need to define a more complex rule, that sets a limit based on interactions of several channels being handled at the same time by the agent. For that case, click **Add concurrent Interactions** to build composite rules.

To illustrate how this works, let's look at the last example from above:

Example	Instructions

• Allow a maximum of 3 chat interactions.
• Also, don't send new chat interactions when the agent is busy with:
  • 1 email and 1 chat concurrently
  • 1 voice
	To enable:
	1. Because the rule will contain chat, email, and voice interactions, you must have already selected all three channels from the **Select Media Rule** menu. That means you should see three separate sections on your screen - one for each channel.
	2. Click the pencil icon in the chat section to define the chat media rules.
	3. In the **Allow a maximum of chat interactions** field, enter **3** in the text box.
	4. In the next section, for the **Also, don't send new chat interactions when agents are busy with:** option, specify an integer of **1** in the text box and then select **Email** from the **Select Channel** menu.
	5. Click **Add Concurrent interactions**.
	6. In the **Also, don't send new chat interactions when agents are busy with, concurrently:** section, specify an integer of **1** in the text box and then select **Chat** from the **Select Channel** menu.
	7. Click **Add Rule on Other Channels**.
	8. In the next section, for the **Also, don't send new chat interactions when agents are busy with:** option, specify an integer of **1** in the text box and then select **Voice** from the **Select Channel** menu.

Edit a capacity rule

To edit a capacity rule that has already been created, go to **Contact Center Settings** -> **Digital Management** -> **Capacity Rules** screen in Agent Setup. Locate the capacity rule from the menu and click the corresponding pencil icon. The Edit Capacity Rule window opens. Edit any or all of its properties.

Assign a default capacity rule

When you create a new capacity rule, it appears in the **Default Tenant Rule** menu on the main **Contact Center Settings** -> **Digital Management** -> **Capacity Rules** screen. Select any Capacity Rule from this menu to make it the default Capacity Rule for a tenant.

Assign a capacity rule to an agent

You can assign a capacity rule to a user in Agent Setup, as follows:

1. From the Users tab, locate the user/agent or add a new one.
2. From the Capacity Rules menu, find the capacity rule you want to apply.

You can also assign a capacity rule to a user in a CSV for XLS file that will be imported via Bulk Import:

- For CSV files - In the CapacityRule column, enter the capacity rule name.
- For XLS files - In the CapacityRule column, specify the capacity rule DBID.

Sharing Business Attributes in Agent Setup

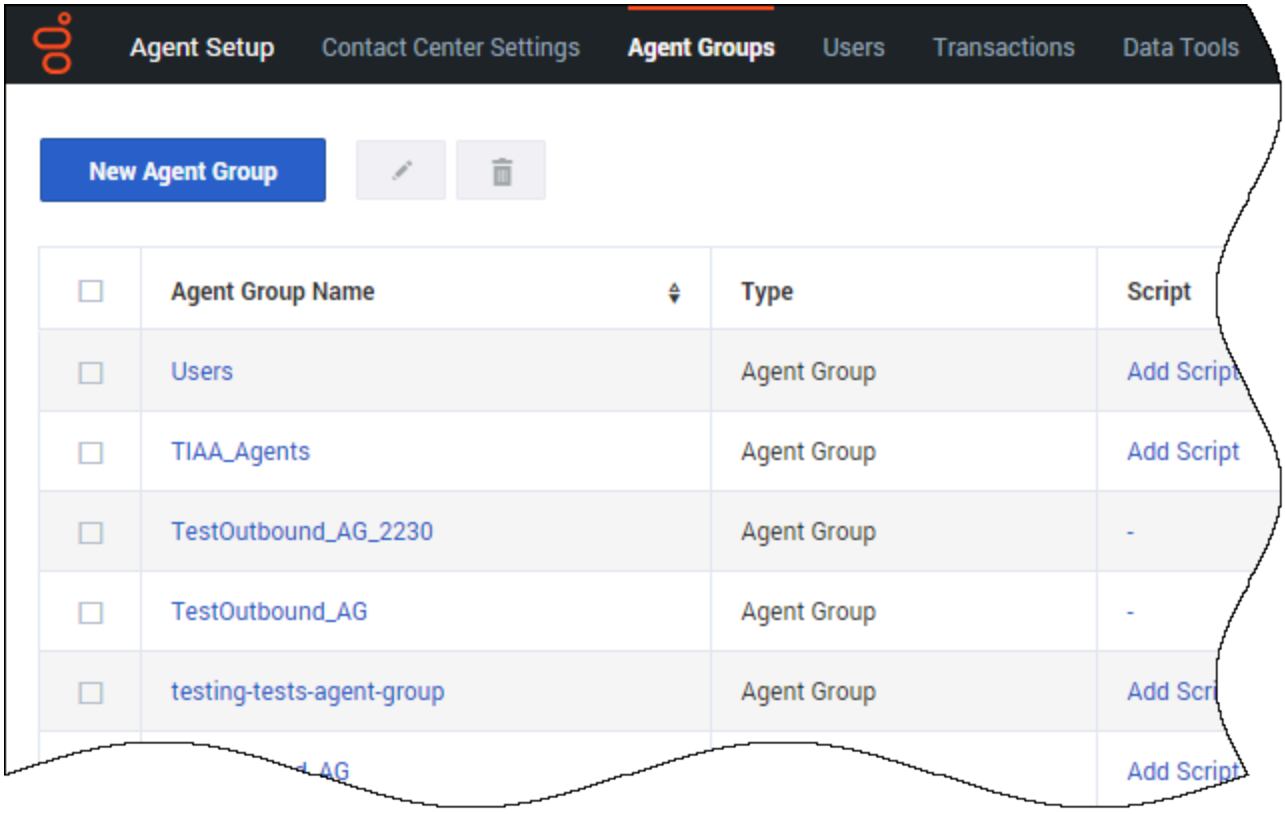
> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

In Agent Setup, you can define a business attribute to be shared at an agent, agent group, and global level. This means that you only need to create a business attribute once and it automatically populates the Business Attribute lists on the **Agent Group**, **Manage User**, or **Desktop Settings** pages.

You can consider a business attribute to be the container that holds a list of objects that have a common theme. You can define shared business attributes for the following objects:

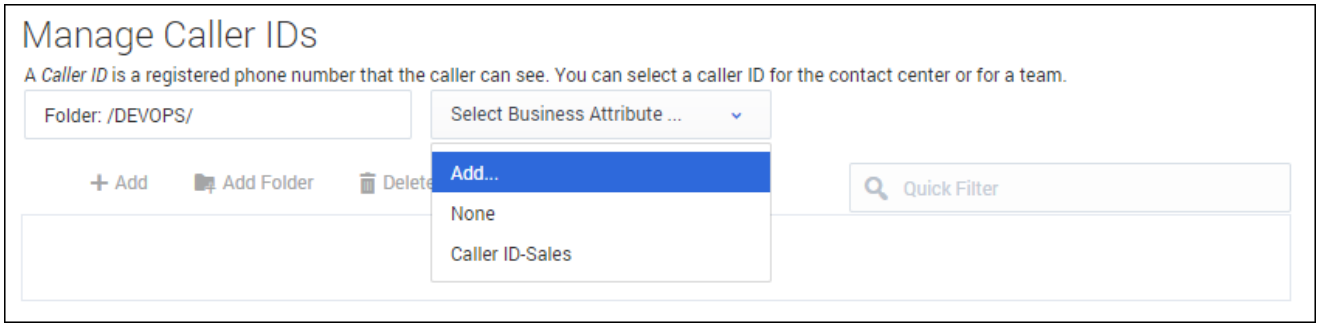
- Caller ID
- Dispositions
- Case Data
- Toast Data

Let's take a look at how Henry, a contact center administrator, uses sharable business attributes.



He wants to create a list of Caller IDs for a specific Agent Group that can be shared to other Agent Groups at a later date.

Henry first needs to create a Caller ID business attribute. He needs to go to the right page in the Agent Setup user interface, so he clicks **Agent Groups**, selects the agent group he wants to edit, and then clicks **Caller IDs**.



Henry clicks the **Folder** drop-down list box to choose a folder into which the business attribute should go. By default, the tenant home folder is automatically selected.

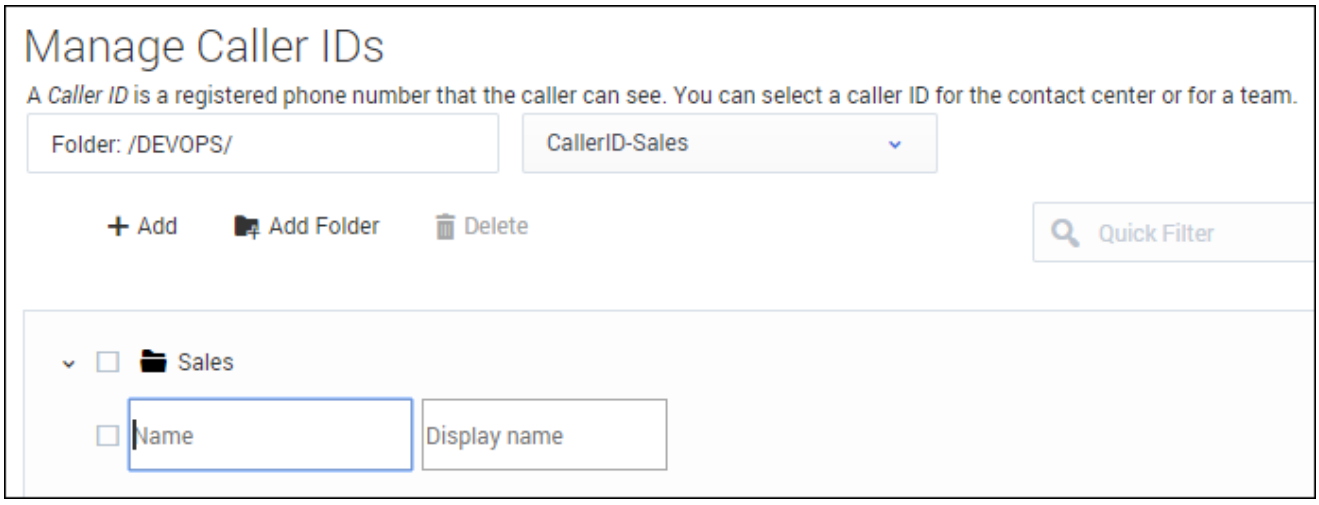
Then Henry clicks the **Add Business Attribute** drop-down list box and selects **Add**.

The other options in the list are:

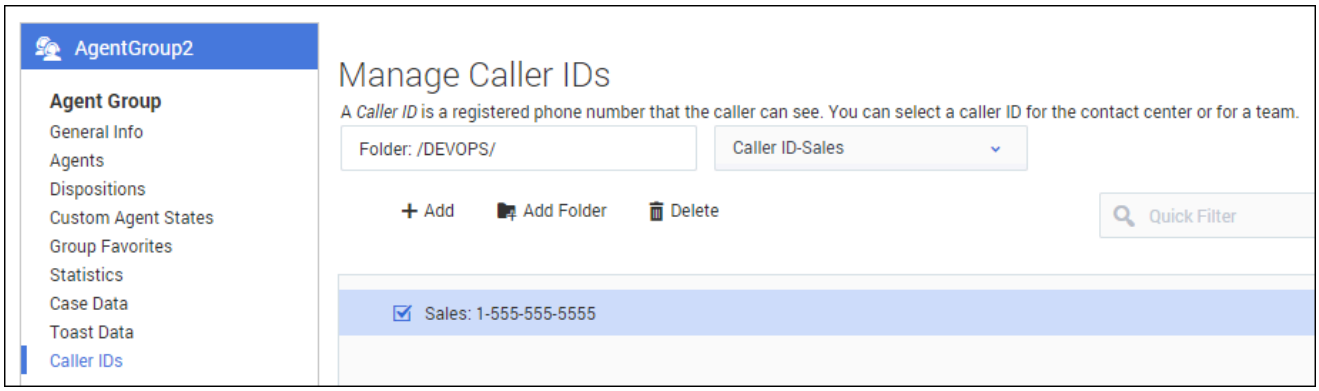
- **None** - This tells the system to not define any business attribute.
- Already configured Caller ID business attributes - These attributes were previously created on this page, or at the agent group level. This is the benefit of shared business attributes. When a business attribute is created once, it populates where needed in other locations.

The **Add Business Attribute** window opens and Henry can now type the name of the business attribute. To be able to easily detect the purpose of the attribute, he prefixes the name with `CallerID`, for example, `CallerID-Sales`.

After he clicks **OK**, the new Caller ID business attribute is displayed not only in the list on the Agent Group page, but also in the Caller ID list on the **Desktop Settings** and **Manage Agent** pages. However, at this point the business attribute is empty; it doesn't contain any actual caller IDs.



Now that the business attribute is defined, Henry can populate the business attribute with a list of caller IDs. To do this, for each caller ID he wants to add, he clicks the **Add** icon to define the **Name** and the **Display Name**. He can add more than one caller ID to the business attribute and he can organize the caller IDs into folders.



Two weeks after he created the caller ID list for one agent group, Henry needs to configure another agent group with the same list of caller IDs. All he needs to do is access the agent group (**Agents Groups > Agent Group2**), and then select the sharable business attribute and caller ID. In this scenario, he selects **/DEVOPS/ > Caller ID-Sales > Sales: 1-555-555-5555**.

The benefit of sharable business attributes is that Henry didn't need to create a new list of caller IDs for another Agent Group.

Remember that this scenario demonstrates how to create a Caller ID business attribute, but the steps also apply to Dispositions, Case Data, and Toast Data.

Working with Global Favorites

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

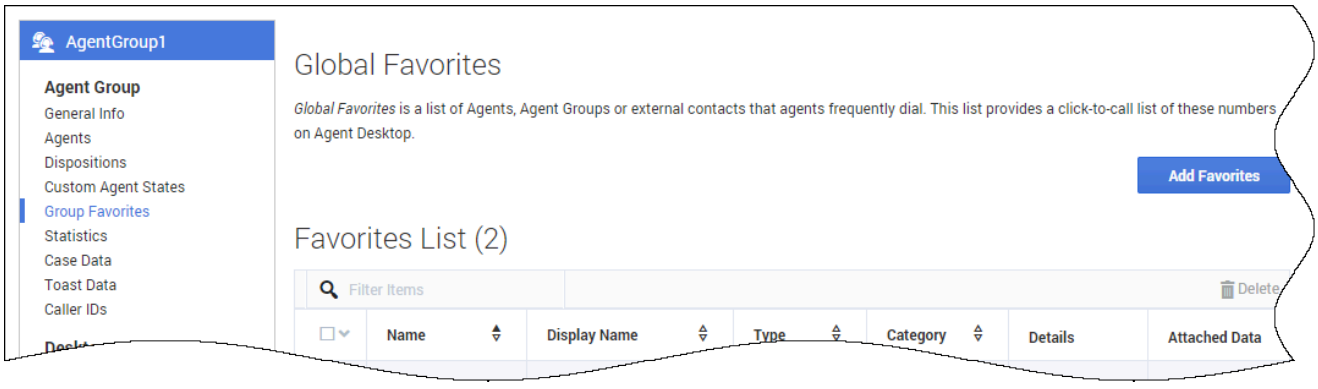
As a contact center administrator, you can create and maintain *Global Favorites*, which are contacts and internal targets that your agents can access quickly in the Team Communicator in Agent Desktop or the Gplus Adapter.

You can create a list of favorites for each of the following levels:

- Desktop settings (everyone)
- Agent Groups (only members of the group)
- Users (personal)

Essentially, you add favorite contacts and targets to the global Favorites List. Then from that list, you can select the favorites that you want to apply to agent groups, individual users, or the whole contact center. Favorites can be re-used, meaning that if you create a favorite for a specific agent or agent group, for example, you can also apply that favorite to another agent group or individual.

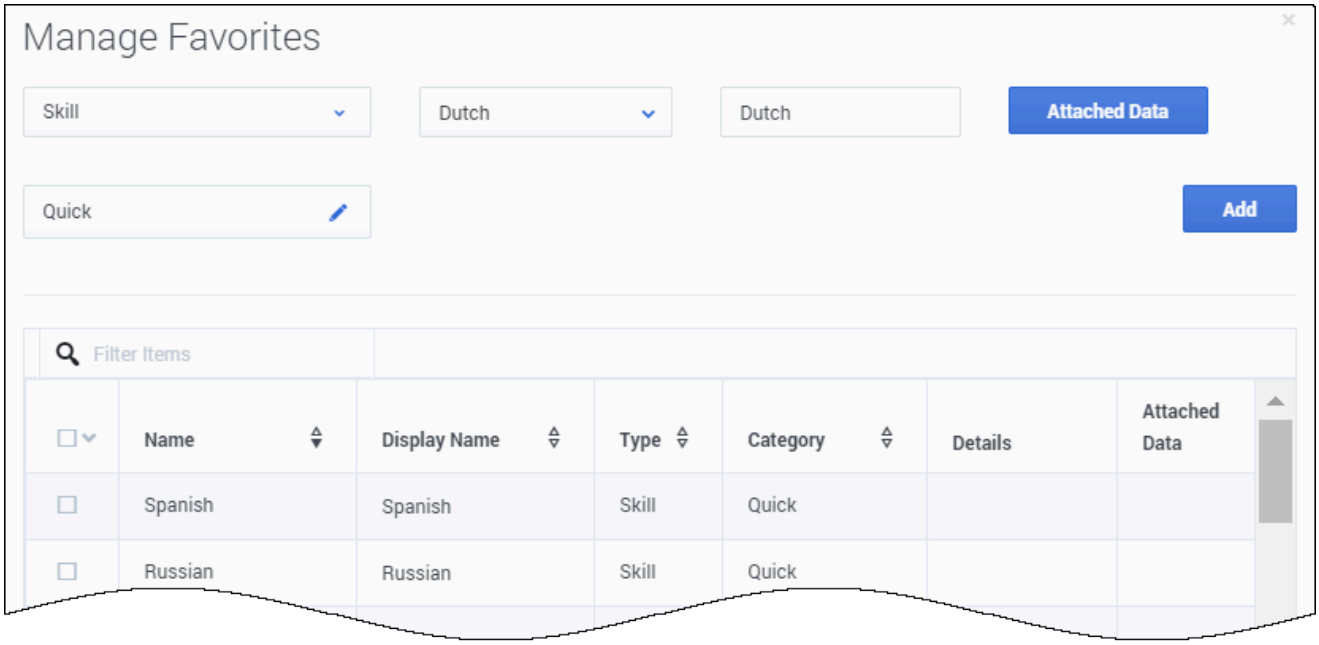
To help demonstrate how this all works, let's observe how Henry, a contact center administrator, adds a favorite to one agent group and then applies the same favorite to another agent group.



Henry is configuring Agent Desktop settings for AgentGroup1 and AgentGroup2. As part of the setup, he needs to create a list of contacts to be displayed in the Team Communicator for each agent that belongs to the groups. For the purposes of this demonstration, he'll add only one favorite. In reality,

he would set up as many favorites as required to meet the needs of his contact center. The goal is to include a quick way for agents to transfer an interaction to an agent who can speak Dutch. These are the steps he takes:

1. Access the Agent Group (**Agent Groups> Agent Group1**)
2. Click **Group Favorites**.



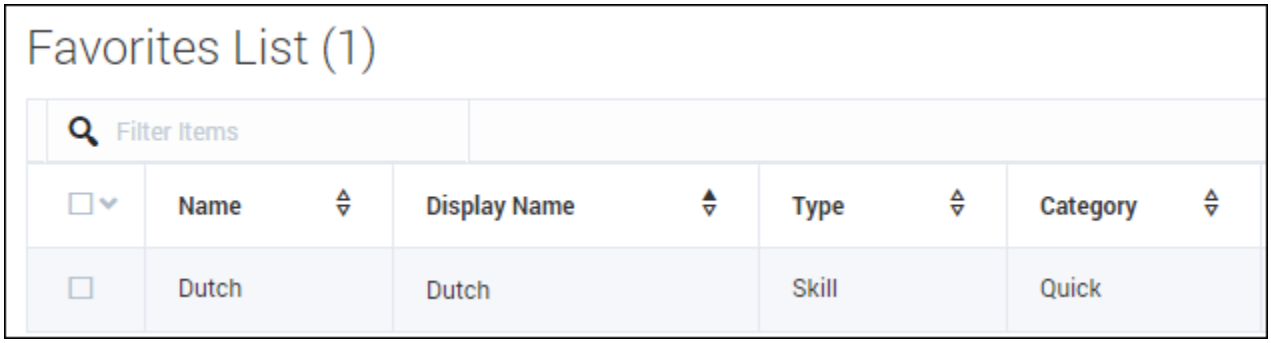
3. Add the Dutch skill as a new favorite in the global list:
   a. Click **Add Favorites**.
   b. On the **Manage Favorites** window, select **Skill**.
   c. Select **Dutch** from the second drop-down list box.
   d. Select a category. Categories organize how the favorites are displayed in Team Communicator. He selects **Quick**.
   e. Click **Add**.

The new favorite (Dutch) is now added to the global favorites list on the **Manage Favorites** window. However, at this point, the new favorite is not assigned to AgentGroup1 yet. It's simply available in the global list. He now needs to add the new favorite to AgentGroup1. Notice that **Dutch** is automatically selected. If it wasn't, Henry would need to select the check box beside **Dutch**.

4. Click **Save**.

Result

When Henry clicks **Save**, the **Manage Favorites** window closes and the new favorite is added to the **Group Favorites** list for AgentGroup1, as shown here.



Now it's time for Henry to add **Dutch** as a group favorite for AgentGroup2. To do this, all he needs to do is:

1. Access Agent Groups (**Agent Groups > AgentGroup2**).
2. Click **Group Favorites** to open the global list of favorites.
3. Click **Add Favorites**
4. Select **Dutch** from the **Manage Favorites** list.
5. Click **Save**.

All members of AgentGroup1 and AgentGroup2 now have **Dutch** listed in the **Quick** contact list in the Team Communicator.

Now that Dutch is listed in the global favorites list, Henry has the option in the future to easily add **Dutch** as a personal favorite for other agent groups, individual users, or all users in the contact

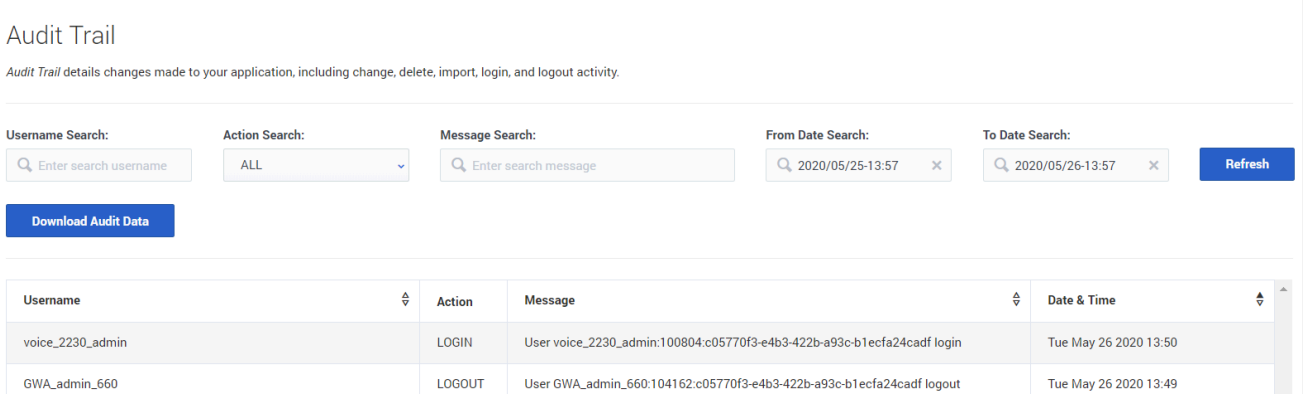
center.

Audit

> **Important**
>
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

The Audit Trail, located in the **Audit** section, details the actions taken in your Agent Setup application, including update, delete, import, login, and logout activities.

> **Important**
>
> There are currently no limits placed on the number of audit logs or how long they are kept in the Audit Trail.



The summary table on the Audit tab contains an entry for every action taken in the application. The table lists the following details:

- Username - Of the user who made the change.
- Action - The type of action made. See a description of each action in the table below.
- Message - Specific details about the action. This could be the exact file name of an imported file or the name of a skill that was updated or created.
- Date & Time - The date and time that the action took place.

- Refresh - Updates/refreshes the audit search results.
- Download Audit Data - Exports the audit logs to .xlsx files.

If you click on any entry within this summary table, a new **Audit complete information** window appears. It lists the details outlined above and includes the **Audit Record** ID.

Each action type is documented in the table below. Note that an 'object' can mean an agent, an agent group, a skill, or a transaction.

Action	Description
LOGIN	A user (identified by a username) has logged in. The Message field specifies the following information about the user who logged in: • Username • User ID • Contact Center ID • The action (login), including any applicable reason or error.
LOGOUT	A user (identified by a username) has logged out. The Message field specifies the following information about the user who logged out: • Username • User ID • Contact Center ID • The action (login), including any applicable reason or error.
CREATED	An object was created (for example, a new agent).
DELETED	An object was deleted (for example, an agent).
UPDATED	An object was updated. If, for example, an agent was added to an agent group, this entry identifies the user who made the update and the name of the user added to the group.
IMPORT	A file was imported. The Message field specifies the name of the file that was imported.

Genesys Predictive Engagement (Altocloud)

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Using Predictive Engagement in Genesys Engage cloud.

Genesys Predictive Engagement (previously called Altocloud), a real-time journey analytics platform, can observe and analyze visitors on your digital properties, such as websites. Altocloud can predict real-time probability for visitors to achieve a desirable business outcome, and then acts to offer the most appropriate and effective channel to assist them in completing their journey. For an in-depth look at Altocloud, see the full documenatation set here.

To set up Altocloud in your Genesys Engage cloud environment, complete these initial tasks.

Prepare your tenant

As a customer, evaluate your interest in Altocloud service for predictive engagement. Reach out to your Genesys account team to discuss terms and conditions prior to making a purchase decision.

With your Genesys account team, evaluate whether your environment has the necessary prerequisites.

With your Genesys project team, enable Genesys Altocloud functionality on your tenant.

You receive:

- A reference to the region where Altocloud is accessible. For more information on using regions to consume API information, see the Region Authentication table.
- An email containing instructions and credentials on how to connect to the Genesys Cloud Admin UI to define business logic, configure integrations, and query analytical dashboards.

Configure permissions and routing logic

Work with your Genesys project team to:

1. Configure necessary Genesys Cloud permissions for administrators. For more information, see Altocloud permissions overview.
2. Configure necessary Genesys Cloud permissions for agents to access the Altocloud journey dashboards within Agent Desktop. Permissions can be set globally for the entire agent population, for an agent group, or for individual agents. For more information see Altocloud permissions overview.
3. Define the appropriate routing logic for chats that Altocloud proactively triggers.
4. If this is the first time you have configured Web Chat with Genesys Widgets, request the corresponding API key to use the Chat API (Advanced Chat v3). You will need this later for the initial provisioning of Widgets on your website.

> **Important**
>
> After the initial service roll-out, if you need additional agent provisioning or routing logic, you may need to raise a Service Request with Genesys Care.

Log in to Altocloud

In your Genesys Portal, click the **Altocloud** tile.

Configure journey shaping settings

On the **Admin** tab, navigate to the Altocloud section.

In the **Web Tracking** section, enable domain whitelisting. This step authorizes your website domain to be a valid originator of web journey events.

Configure your Segments. Segments categorize your website visitors into audiences that can be used in real-time to trigger engagement logic.

Configure your Outcomes.

Outcomes represent business objectives and Altocloud uses them to predict whether a visitor's behavior is leading them to achieve a desirable result on their own, or whether an agent needs to intervene.

Configure your web pages

> **Important**
>
> See Prepare widgets for Altocloud to deploy Genesys Widgets for Genesys Cloud. Widgets v.2 is the Genesys Cloud name used for Genesys Widgets. For Genesys Engage, see the Deployment guide for instructions.

Once you deploy Genesys Widgets, obtain the tracking snippet and add it to all web pages that Altocloud should observe.

Configure your web pages with Genesys Widgets:

- If you are already running Genesys Widgets, ensure that you have the proper version.
- If needed, fetch the latest version from the public CDN (instructions in deployment guide).

Configure your widgets to enable Web Chat on every page. You must use Genesys Engage v3 API transport. For more information, see Configure widgets and services.

Obtain the API key from your Genesys customer service representative (x-api-key).

Use Live Now

To confirm that the Altocloud is properly provisioned and functional, navigate to Live Now and observe web visits in real-time.

Create engagement logic

Configure action maps to engage with your website visitors.

Receive chats in Agent Desktop

Log in to Agent Desktop.

To receive a live chat interaction, set your status to **Available**.

Expand the **Journey** tab and view a customer's journey.

Additional integrations

Take notice of the following integrations.

Architect workflows

Administrators and contact center managers can now create workflows in Architect. This feature allows customers to streamline business processes by using Predictive Engagement to build in-house and third-party integrations.

For more information, see About Architect and Work with workflows.

Trigger architect flows

Predictive Engagement administrators can now use a new Architect flow action to configure action maps that trigger Architect flows that integrate with in-house and third-party systems. For more information, see About Architect flow actions.

Create leads in Salesforce

Predictive Engagement facilitates an integration with Salesforce by providing predefined Genesys Cloud data actions and an example Architect workflow. An administrator can build an action map that uses the Architect flow action along with these data actions and workflow to create leads and assign campaign attributions in Salesforce when qualified users arrive on your website. For more

information, see Salesforce lead creation and campaign attribution.

CX Contact

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

CX Contact is a web-based application used to create and manage outbound campaigns. More specifically, you can use the application to do the following:

- Campaigns - Set up and manage outbound campaigns, and monitor the status of running campaigns.
- Contact Lists - Import contact lists, apply upload rules to a list, create list automation jobs, and define user field labels.
- Compliance - Restrict contact attempts to records within a contact list, and import or manage contact suppression lists.

Introduction

Link to video

- Get to know campaign structure and terminology
- Learn about dialing modes and IVR modes
- Learn about pacing and optimization
- Learn about time zone assignment options

Campaigns

Link to video

- Create a dialing profile
- Create a campaign template
- Create a campaign group
- Learn about the metrics displayed on the campaign dashboard

Contact Lists

Link to video

- Learn about list fields, file formats, and database tables
- Import or manage a contact list or contacts within a list
- Set weights for contact lists
- Define list rules
- Create data mapping schemas
- Create list automation tasks
- Define or edit user field labels

Compliance

Link to video

- Suppress contact attempts based on a set of prescribed rules
- Assign a time zone to phone numbers containing a specific country code or area code
- Suppress contact attempts based on the location of contacts
- Define allowable calling windows for each day of the week for a given region
- Suppress contact attempts by date
- Add or manage a suppression list

CX Contact Solutions Overview

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

CX Contact is part of a fully customizable cloud solution that enables you to set up, run, and fine-tune your outbound campaigns.

Through a series of Genesys Engage cloud products, you can define your routing application, import contact lists, set up a campaign, apply compliance rules, and monitor and assess your campaign through real-time and historical reports.

Here are the four stages you'll go through to make the most out of CX Contact in Genesys Engage cloud:

Stage 1—Configuring outbound routing
Stage 2—Setting up a campaign
Stage 3—Dialing and call handling
Stage 4—Monitoring a campaign

Stage 1—Configuring outbound routing

Genesys handles the initial configuration of outbound routing.

Using Platform Administration, we configure the routing points, agent groups, and virtual queues. We use Designer to configure voice scripting and call flow.

After that, you can log into these applications at any time to tweak the settings. Here are a few instructions:

- How to set up Outbound Routing
- How to create and modify DNs
- How to set up agents
- How to add the Route Call Block in Designer

Stage 2—Setting up a campaign

Link to video

Before you sign into the CX Contact application to set up your campaigns, you should visit the Campaign Structure and Terminology page in the CX Contact Help manual to learn about the following five components that make up a campaign:

- Dialing profile
- Session profile
- Campaign template
- Campaign group
- Contact list

After that, you can log into CX Contact to set up and manage your campaigns.

Many of the key features and tasks available to you in CX Contact are listed in the table below. **Note**: Clicking any of the links in this table will take you to the CX Contact Help manual.

Category	Key Tasks
Campaigns	• Create a dialing profile • Create a campaign template • Create a campaign group • Apply call treatments • Learn about the metrics displayed on the campaign dashboard
Compliance	• Suppress contact attempts based on the results of previous attempts • Assign a time zone to phone numbers containing a specific country code or area code • Suppress contact attempts based on the location of contacts • Define allowable calling windows for each day of the week for a given region • Suppress contact attempts by date • Import or manage a suppression list
Contact Lists	• Learn about contact list fields, formats, and database tables

Category	Key Tasks
	• Import or manage a contact list or contacts within a list • Set weights for contact lists • Define list rules • Create List Automation tasks • Create data mapping schemas • Define or edit user field labels • Import specification files
Settings	• Define or edit General settings • Define or edit Security settings • Define or edit Pacing settings

Stage 3—Dialing and bridging to an agent

Dialing

When you set up a campaign in CX Contact, you'll need to choose a dialing mode or IVR mode that best suits your campaign. Your choice will depend on the type of campaign you're running, the number of agents (if any) assigned to the campaign, and compliance regulations.

Depending on the dialing mode or IVR mode selected, you can also apply pacing and optimization parameters to influence dialing behavior. Refer to the Dialing modes and IVR modes page for a complete description of each dialing mode and IVR mode as well as the pacing options and optimization parameters that apply to each.

Before starting a dialing event, the system refers to all selected pacing options, optimization parameters, call treatments, and compliance rules in place. It places the call and then, for agent-assisted campaigns, hands the call off to an agent once it detects a voice on the line.

Call handling

All interactions take place via the Agent Desktop application. The way in which an agent handles a call depends on the dialing mode or IVR mode used in a campaign.

Link to video

For Predictive and Progressive dialing and IVR modes, the outbound calls are directed to the agents'

workstation and dialed automatically. The video to the left shows you how agents handle these calls.

In a Preview dialing mode, agents preview the customer case information and then manually dial the customer's phone number.

For a complete description of how agents use Agent Desktop to handle each type of outbound call, refer to the Outbound campaigns page in the Agent Desktop Help manual.

Stage 4—Monitoring a campaign

Once you've set up your campaigns, you'll want to monitor the status of those that are still running and look at the results of those that have ended. You can use a series of Genesys reporting tools to accomplish these tasks.

Real-time reporting

To monitor the status of an ongoing campaign in real-time, you have two options: the CX Contact campaigns dashboard and Genesys Pulse.

CX Contact campaigns dashboard

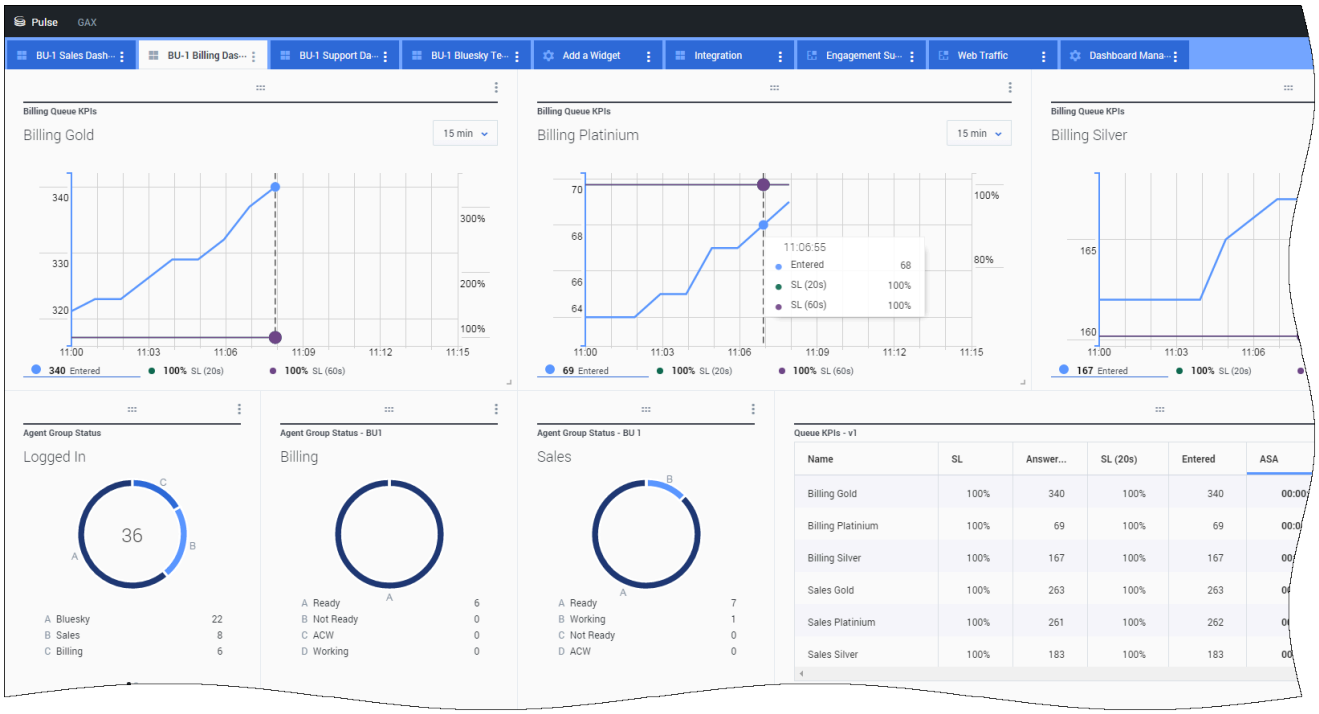
The CX Contact campaigns dashboard provides a statistical overview of call activity for each campaign group that is currently running. If a campaign group contains multiple contact lists, the data is broken down by contact list.

	Lists							Time	Attempts			Agents		Hit Ratio	Abandoned	
Campaigns/Groups				%		%		Remain	Done	Remain	PerMin	Logged In	Busy Factor	Success	Count	%
Welcome Notifications																
Welcome Notifications-bb75fb88@TestOutbound_AG_2...	1460	172	26	1.78%	48	3.29%	1214	00:08:22	74	1460	6	5	80.00%	30.00%	25	4.50%
canada4	1200	100	10	0.83%	30	2.50%	1060	00:02:22	40	1200	2			33.33%	8	3.00%
canada3	90	32	0	0.00%	10	11.11%	48	00:02:00	10	90	2			0.00%	10	100.00%
canada2	170	40	16	9.41%	8	4.71%	106	00:04:00	24	170	2			32.00%	7	3.00%

For a complete description of fields and metrics displayed on the campaigns dashboard, refer to the Campaigns dashboard page in the CX Contact Help menu.

Genesys Pulse

Use Genesys Pulse to generate in-depth reports on agent activity and campaign activity.



In the Genesys Pulse application, you can add a report widget to your report dashboard, choose a template or define your own, select objects and statistics to include in your report, and specify default settings – like the name, refresh rate, and type of widget. And then you can save and download your report as a CSV file.

For a complete list of available agent statistics through Pulse, refer to the Agent Statistics page in the Pulse Help manual. For a complete list of available campaign statistics through Pulse, refer to the Campaign Statistics page in the Pulse Help manual.

Historical reports

Now you want to retrieve statistics of a campaign that has ended. OK, you have two options:

- CX Contact List Export
- Genesys CX Insights

CX Contact List Export

Using the List Automation feature in CX Contact, you can schedule an automatic list export for when a campaign ends. This list export will contain call result fields. For more information about this feature, refer to the List Automation page in CX Contact.

Genesys CX Insights

Genesys CX Insights uses data stored in our database and presents it in readable reports. For a list of Outbound reports, refer to the Outbound Contact reports page and the Agent Outbound Campaign Report in the *Genesys CX Insights* section of this manual.

Callback

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Getting Started with Callback.

Businesses sometimes cannot offer on-demand, low-wait agent help because of resource limitations or increased service usage. In these situations, the best option is to offer some form of deferred service that can connect consumers and agents later, at a mutually-beneficial time.

Genesys Callback, which takes its name from the most typical form of deferred agent voice connection, also provides support for a user-originated call-in feature, in addition to a range of scenario and sourcing options. In short, Genesys Callback is much more than callback functionality. Consider the name to be shorthand for the rich set of scenarios that are described in this Callback documentation.

> **Important**
> Genesys Callback is enabled by Genesys Engagement Services (GES) behind-the-scenes and you might see references to GES or "Engagement Services" throughout Callback pages or within resource names.

What is a Callback?

A Callback is a data record that contains data for calling a customer at some time in the future. Callbacks generally consist of attached data and configuration data. Attached data is specific to the end customer and might be used in screen pops, routing, reporting, and other applications. Configuration data describes how a callback is fulfilled; this includes target expressions, timeout values, and virtual queue names.

Using Callback, you can associate a callback service with a phone number, which enables a caller to request a callback. Depending on which scenario you choose to implement, the callback is then routed to an agent who processes the call.

Supported Scenarios

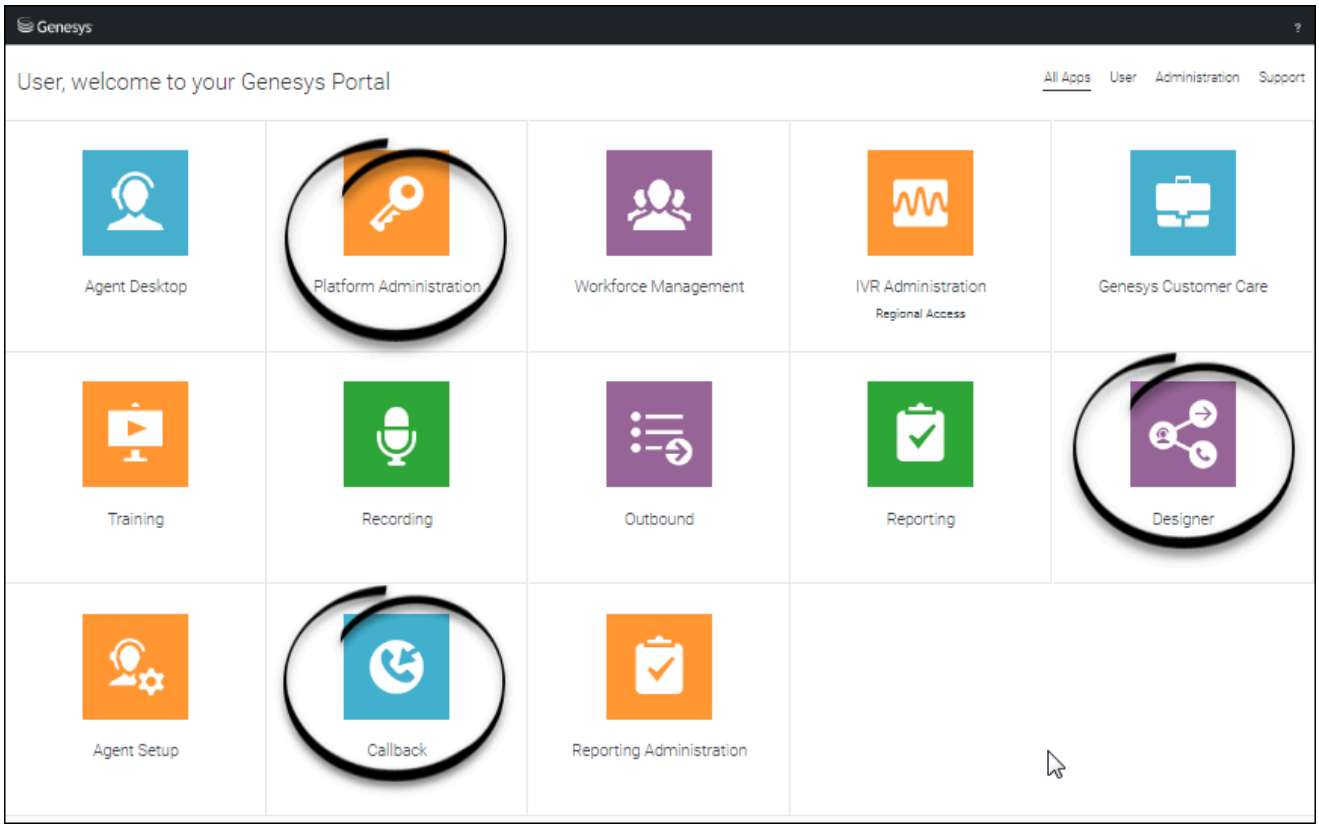
For information about the supported scenarios for Genesys Callback, see Callback Scenarios.

Callback Reports

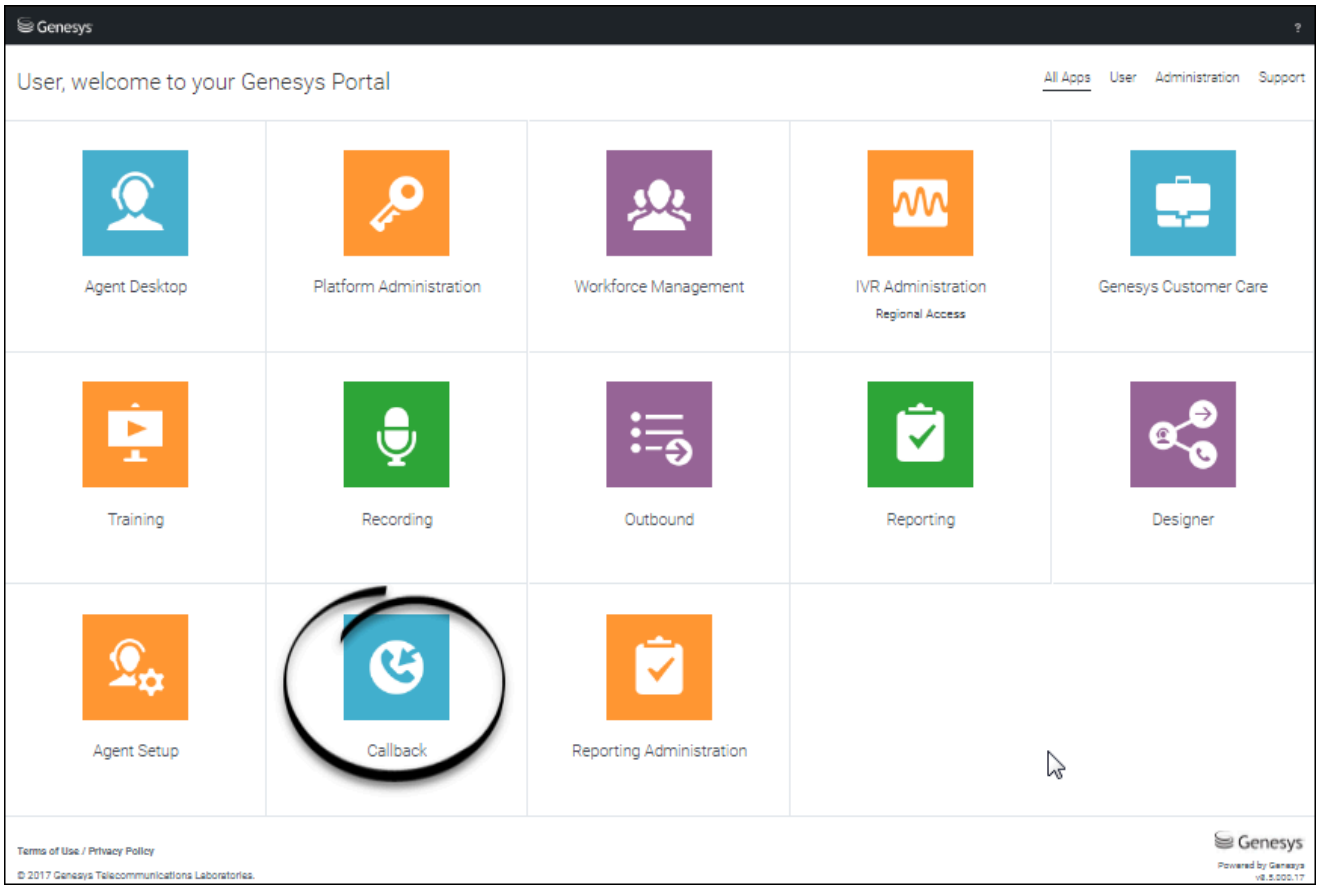
For information about real-time reporting for callbacks, see Real-Time Reporting with Pulse in the *Reporting in the cloud* documentation. Callback activity is tracked as part of the Queue reports. You can add a report widget that is based on the Callback Activity template to your Pulse dashboard.

For information about Callback historical reporting using Genesys Customer Experience Insights (GCXI), see Callback Reports in the *Reporting in the cloud* documentation.

About the Callback Application

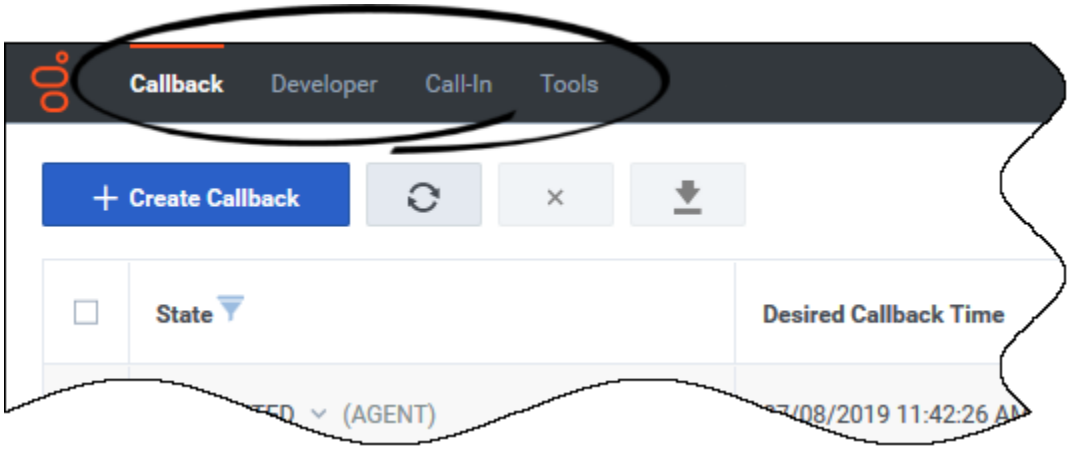


The Callback application is – as its name suggests – specific to callback functionality and management. If you are a Callback Administrator, there are two additional applications that you might use to configure and provision the Callback application. On the Genesys Portal, they are Platform Administration and Designer. Platform Administration is used to manage contact center resources and user access to Callback features. Designer is used to provision Callback-related applications.



Click the **Callback** icon on the Genesys Portal to open the application.

In the Callback application, tabs, dialogs, and panels display based on your role permissions. If you cannot view a feature, use the Platform Administration application to check your access group.



The Callback application has multiple tabs. You might see all available tabs or only some of the tabs in your Callback UI. Administrators will make sure that you have access to the tab or tabs that you need to do your job. Access is based on the role(s) to which you have been assigned.

The following tabs are available in the Callback UI:

- **Callback**: Displays the list of callbacks. In some scenarios, you might need to create or monitor a callback request, or even cancel a set of callbacks. Callback Monitor is the minimum role required to access the **Callback** tab. To modify callback records, you require the Callback Administrator or Callback Supervisor role.
- **Developer**: You must be a member of the Callback Administrator or Callback Developer role to access the **Developer** tab. From the **Developer** tab, you can access developers' tools such as the list of errors related to your Callback API queries as well as pages on which you can validate your API keys or provision Push Notifications.
- **Call-In**: You must be a member of the Callback Administrator or Callback Developer role to access the **Call-In** tab. If you have the Click-To-Call-In scenario provisioned, use the **Call-In** tab to view the Click-To-Call-In records.
- **Tools**: You must be a member of the Callback Administrator, Callback Supervisor, or Callback Developer role to access the **Tools** tab. The **Tools** tab offers additional views and tools to assist with callback management, configuration, and troubleshooting. For example, you can search for a specific interaction within the callback records or force a refresh of tenant configuration data.

Getting Started with Genesys Callback

The Callback application is supported in the following browsers:

- Firefox
- Google Chrome
- Microsoft EdgeHTML version 16.0 and up and on Chromium-based Edge

Before you start working with the Callback application, you need to configure your callback scenario. You can review the supported callback scenarios on the Callback Scenarios page. In addition, you must provision Designer applications for Callback and configure the callback service. This section provides information about where to start.

1. To get started, go to the Provisioning Callback page. The page tells you what you need to configure in Platform Administration and then guides you through the Designer application provisioning process, including how to determine which type of Designer applications you require for your callback scenario.
2. Genesys provides predefined roles for the Callback application to ensure that your users only have access that is appropriate for your business needs. If you are new to Genesys Callback, you might have to assign resources (users) to appropriate access groups. In addition, you have the option to restrict access to queues (and lists of callbacks) in the Callback application based on your lines of business. You can create your own custom access groups and enable or disable Read permissions as required.
   Read the following pages to understand which roles are required for access to each tab in the Callback UI:
   - Managing Callbacks

- Using the Developer Tab
- Using Callback Tools

  For detailed information about Callback-specific roles and access groups, see Controlling User Access.

3. After you have completed Callback provisioning and testing to ensure that calls are routed correctly, and Callback users have been assigned to the correct roles and access groups, you can begin to use the Callback UI. The **Callback** tab displays the list of callback records. Users with sufficient permissions use the **Callback** tab to manage the callback records, including creating, editing, or cancelling callbacks.
4. Callback Administrators and Developers have access to a **Developer** tab in the Callback interface. Use the **Developer** tab to manage callback activity and features at a more technical level. For example, you can check for errors in Callback API queries or validate API keys. To learn more about the **Developer** tab, see Using the **Developer** Tab.
5. If you work with Genesys Engagement Services/Callback REST APIs, you can find links to tutorials and additional information on the Genesys Engage REST APIs and Tutorials for Callback page in the Callback documentation.

Callback Scenarios

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Callback scenarios.

It does not matter how a callback originates, the voice interaction is always converted to a virtual call and added to the queue where it is monitored so the system can provide information – such as the Estimated Wait Time (EWT) for the queue – to future callers.

This page describes the callback scenarios that Genesys supports with Callback.

Immediate Callbacks

An Immediate callback is a callback that is set in motion when your customer (a "consumer") makes a request to be called back as soon as an agent who can provide assisted service becomes available. A consumer can request an Immediate callback in the following ways:

- While the consumer is in-queue on an IVR:
  - A consumer's call arrives and the caller is offered Immediate callback. If the caller accepts, he or she confirms the phone number for the callback.
- Through an API call; in other words, the consumer makes the request from a mobile app or website:
  - A consumer is using your company's or organization's mobile app or website and encounters a situation where he or she requires assisted service by voice. The consumer taps or clicks a button to request a callback, confirms or provides the number at which he or she would like to receive the call, and receives a confirmation message that the request was received.

No matter how the Immediate callback is requested, when an agent who satisfies the required skill expression is ready, then the consumer is called and the call is routed to the agent.

Because user/context data might be attached to the API request for a callback, key components of the consumer's app or web journey can be preserved for agent or reporting use.

Scheduled Callbacks

A Scheduled callback is a callback that is set in motion when your customer (a "consumer") makes a request to be called back in the future, at an approximate time that works for the consumer's schedule. A consumer can request a Scheduled callback in the following ways:

- While in-queue on an IVR:
  - A consumer's call arrives and the caller is offered the option to schedule a callback for some time in the future. If the caller accepts, he or she confirms the phone number for the callback and is prompted to input the time at which he or she would like to receive it. Using the caller's requested time, the system searches for the closest-matching available time to connect with an agent.
- Through an API call (from a mobile app or website):
  - A consumer is using your company's or organization's mobile app or website and encounters a situation where he or she requires assisted service by voice. The consumer taps or clicks a button to request a callback, confirms or provides the number at which he or she would like to receive the call, interacts with a date/time picker to search for availability, and receives a confirmation message that the request was received.

No matter how the Scheduled callback is requested, if an agent who has the required skill set is ready at the specified time, then the consumer is called and the call is routed to the agent.

Because user/context data might be attached to the API request for a callback, key components of the consumer's app or web journey can be preserved for agent or reporting use.

Click-To-Call-In (Immediate)

> **Important**
>
> To implement this scenario, you need to use the corresponding Call-In API to initiate the Click-To-Call-In request.

A Click-To-Call-In (Immediate) interaction is set in motion when your customer (a "consumer") taps a button in a mobile app that is designed to trigger a Call-In API request:

- A consumer is using your company's or organization's mobile app and encounters a situation where he or she requires assisted service by voice. The consumer taps a button that you have provisioned in your app to connect consumers to your contact center.
- The system responds with call-in details immediately. Using that information, the app triggers a call to your contact center. The system attempts to match the caller to existing information. For more information, see Provisioning the Click-To-Call-In (Immediate) Scenario.
- If the attempt to match the caller to a Call-In request is successful and your Designer application is configured to route the call when a match is made, then the call is queued on hold like any other call to the contact center. If the consumer is placed in a queue where the EWT is above the configured threshold, then the consumer might be offered a callback option.

Because user/context data might be attached to the API request, key components of the consumer's app journey can be preserved for agent or reporting use.

Click-To-Call-In (Delayed)

> **Important**
>
> To implement this scenario, you need to use the corresponding Call-In API to initiate the Click-To-Call-In request.

A Click-To-Call-In (Delayed) interaction is set in motion when your customer (a "consumer") taps a button in a mobile app that is designed to trigger a Callback API request. The following description of what happens next is a brief summary. See How Click-To-Call-In (Delayed) Works for additional information. To provision the Click-To-Call-In scenario, see Provisioning the Click-to-Call-In Scenario.

- A consumer is using your company's or organization's mobile app and encounters a situation where he or she requires assisted service by voice. The consumer taps a button to contact your center.
- The mobile app sends a callback request, which includes Push parameters that the system uses to contact the consumer to provide information about the callback.
- The call is queued in the Click-To-Call-In (Delayed) virtual queue.
- When the consumer reaches the top of the queue, the system sends a Push Notification to the mobile app to notify the consumer that the callback is ready.
- When the consumer accepts the callback, the system immediately replies with a Push Notification that provides a phone number to call and, if configured, a unique access code that the consumer will be asked to enter before the call is initiated.
- When the consumer calls in and enters the access code, if required, the system attempts to match the caller to an existing callback request. When a match is made, the caller is queued and routed to the next available agent who satisfies the required skill expression.

Because user/context data might be attached to the API request, key components of the consumer's app journey can be preserved for agent or reporting use.

How Click-To-Call-In (Delayed) works

Your mobile app sends a Callback request when the consumer taps the button or link that you have provisioned for the Click-To-Call-In Delayed feature. The call is then queued in the Click-To-Call-In (Delayed) virtual queue. The Callback request includes Push parameters. To use Push Notifications with Callback, see Provisioning Push Notifications for information.

A Click-To-Call-In (Delayed) request is routed to an agent when the following criteria are met:

1. The Call-In request can be matched to an existing Callback request in the system.

2. Your Click-to-Call-In Match Designer application is configured to route the call.
3. An agent (with the correct skills if skills are configured) is ready to accept the call.

There are, however, a number of things that can happen during an active Click-To-Call-In (Delayed) session that might impact the session's flow. For example, the system might fail to match a Click-To-Call-In (Delayed) request on the first attempt. As long as the Click-To-Call-In and Callback requests remain valid and outstanding, though, the consumer can call and try again.

There are also Designer settings that can purge the callback from the system or remove the callback from its queue:

1. The **Callback Purge Time (minutes)** for the virtual queue is reached before the consumer responds to Push Notifications or before an agent is available to assist the caller. In this case, the callback is purged from the system.
2. The end of the business day, based on the configured Business Hours, occurs before the **Callback Purge Time (minutes)** is reached, before the consumer responds to Push Notifications, or before an agent is available to assist the caller. In this case, the callback is purged from the system.
3. The **Push Callback Expiry Time (minutes)** setting in Designer causes the callback session to terminate when that time interval expires. When this happens, the callback is removed from the queue, but not purged from the system until one of the previous two events occurs.

A Call-In API request that is associated with a Callback request that was purged or terminated will fail because no match can be made between the Call-In and Callback requests.

Provisioning Callback in Designer

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Provisioning Callback in Designer.

Callback is provisioned through Designer. To create and configure Callback services, Designer includes a set of blocks dedicated to Callback.

Provisioning Callback is slightly different than provisioning applications for route points and chat endpoints. For information about the supported Callback scenarios, see Callback Scenarios. This page provides information about how to provision your callback scenario in the Designer application.

Before You Start

Before you provision Callback in Designer, make sure that the following objects are created in Platform Administration and ready for use:

- A Callback Administrator.
- Route Points to be used for Inbound strategies that offer Callback.
- Route Points to be used for Outbound strategies, if Callback calls to consumers will be handled by separate Outbound strategies.
- Virtual queues to store Callbacks. Genesys recommends that you use three queues for callbacks. For more information about the virtual queues, see Provision the Callback Virtual Queues.
- At least one agent who will process Callbacks.

Provision the Callback Virtual Queues

In addition to an Inbound virtual queue, Genesys recommends that you have two additional virtual queues for Callback reporting purposes. That means that you will have the following three queues for Callback:

- Inbound virtual queue
- Callback virtual queue
- Outbound virtual queue

Genesys recommends that you use the following naming conventions for the queues:

- Inbound virtual queue: <VQ_name>_VQ
  For example, `Sales_VQ`
- Callback virtual queue: <VQ_ name>_VQ_CB
  For example, `Sales_VQ_CB`
- Callback outbound virtual queue: <VQ_name>_VQ_CB_OUT
  For example, `Sales_VQ_CB_OUT`

Having all three Callback-related virtual queues provides the following functionality:

- For each call type, the system can keep track of and compare Estimated Wait Time (EWT) and other important queue statistics separately.
- You can configure both historical and real-time reporting. While the Inbound and Callback virtual queues collect statistics such as EWT and which calls accept the callback offer and which calls reject it, the Outbound virtual queue collects data for outbound interactions such as how long the customer had to wait for an agent to connect during the callback attempt.

After you create the virtual queues that will be used in your callback scenario, you must provision applications and Callback services in Designer. The virtual queues that you have created for callback functionality will be required to complete the Designer application and services provisioning.

Provisioning your First Callback Scenario in Designer

The following Callback provisioning workflow assumes that you have already completed the configuration described in the Before you Start section, above.

- Provision Callback for the inbound strategy. For information, see Create your Designer Applications and Provision a Designer Application to Offer Callback Through the IVR.
- Provision Callback for the outbound strategy. For information, see Create your Designer Applications and Provision the Designer Callback Application.
- Provision Business Hours for Callback.
- Provision the Callback services in Designer.
- Test your Callback scenario configuration.

Create your Designer Applications

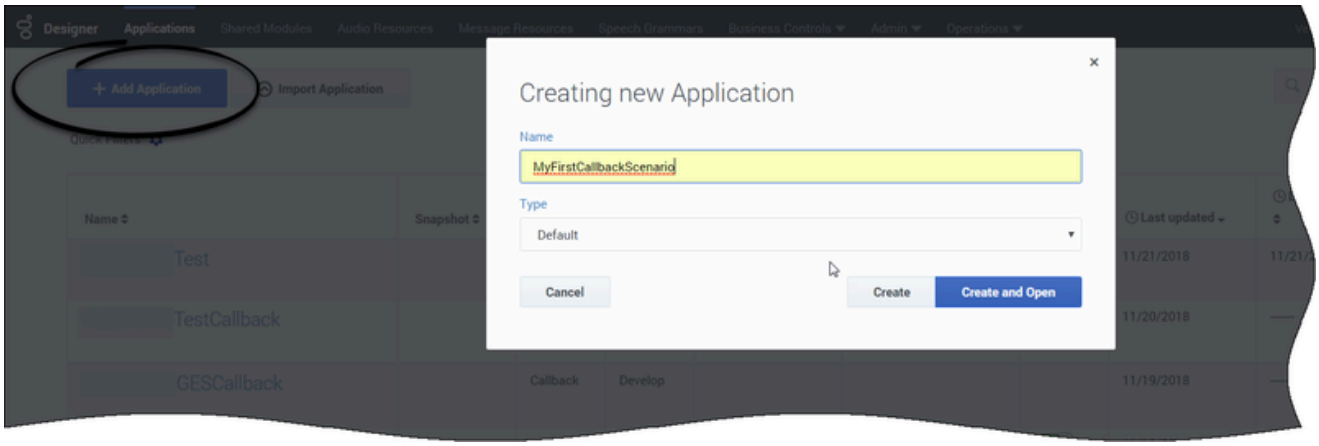
The following table provides information about the types of Designer applications that you require for each supported callback scenario. Create the Designer applications after you have created the Inbound, Callback, and Outbound virtual queues and before you provision the Callback services.

Callback Scenario	Designer Application Type
In-Queue Callback	Default + Callback

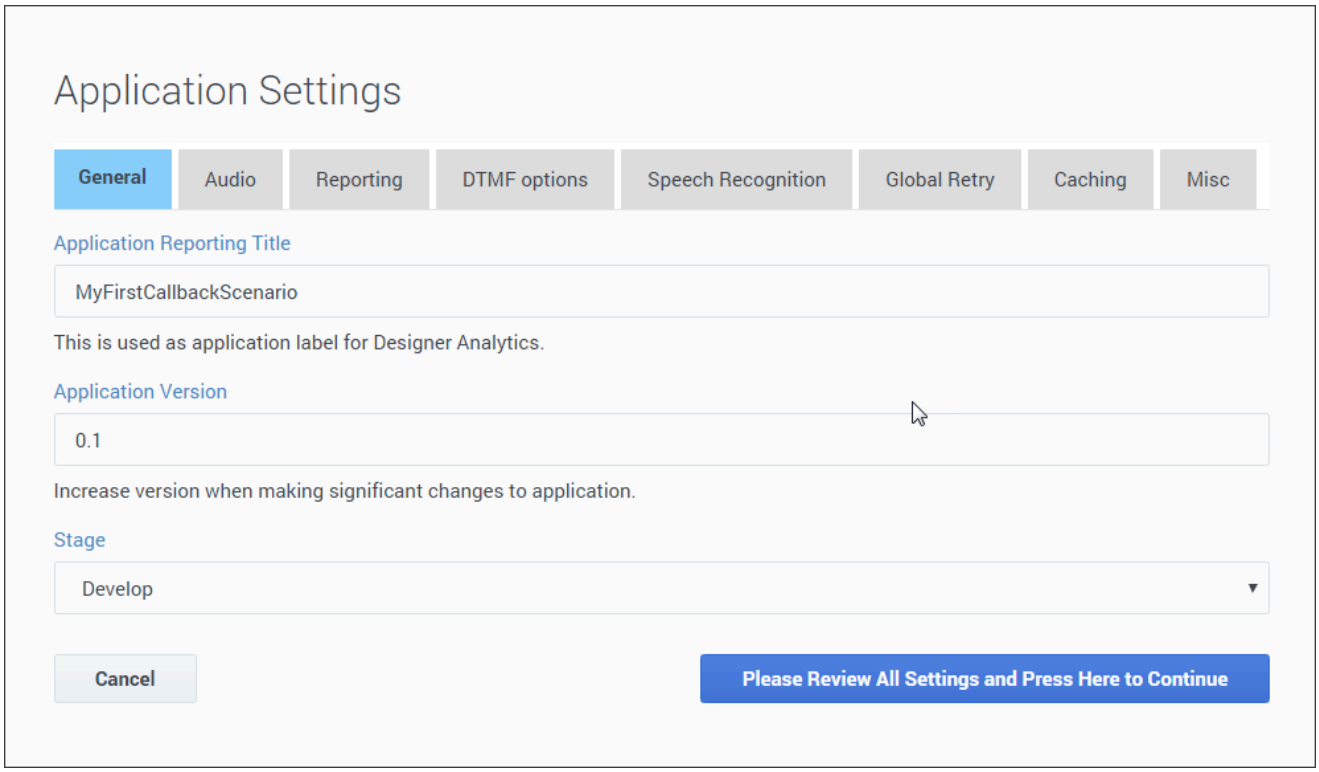
Callback Scenario	Designer Application Type
Scheduled Callback	Default + Callback
Web Callback	Callback

The following procedures show you how to provision Designer applications for Callback. The Default-type Designer application provides the Callback offer to a customer waiting in a queue; for example, in a scenario where the customer is connected to an IVR. The Callback-type Designer application provides the Callback attempt for scheduled and web callback scenarios.

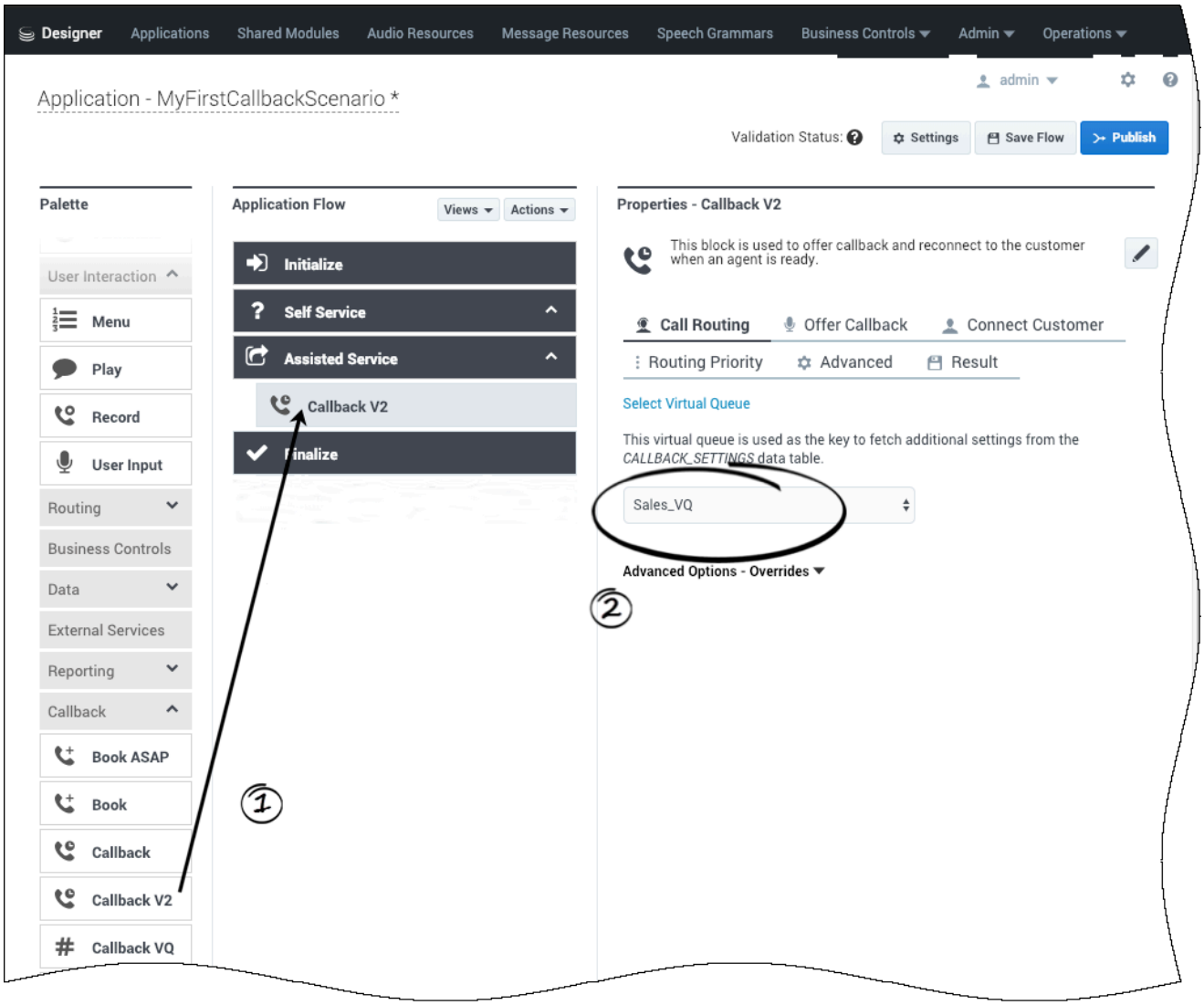
Provision a Designer Application to Offer Callback Through the IVR



In Designer, add a new application. The application type must be `Default`. This application is used to offer callback through the IVR.



There are no mandatory settings changes for the Default application, however, if there are any specific settings that you typically use for Designer applications, consider if those settings are required for your Callback Default-type application and make any necessary updates.

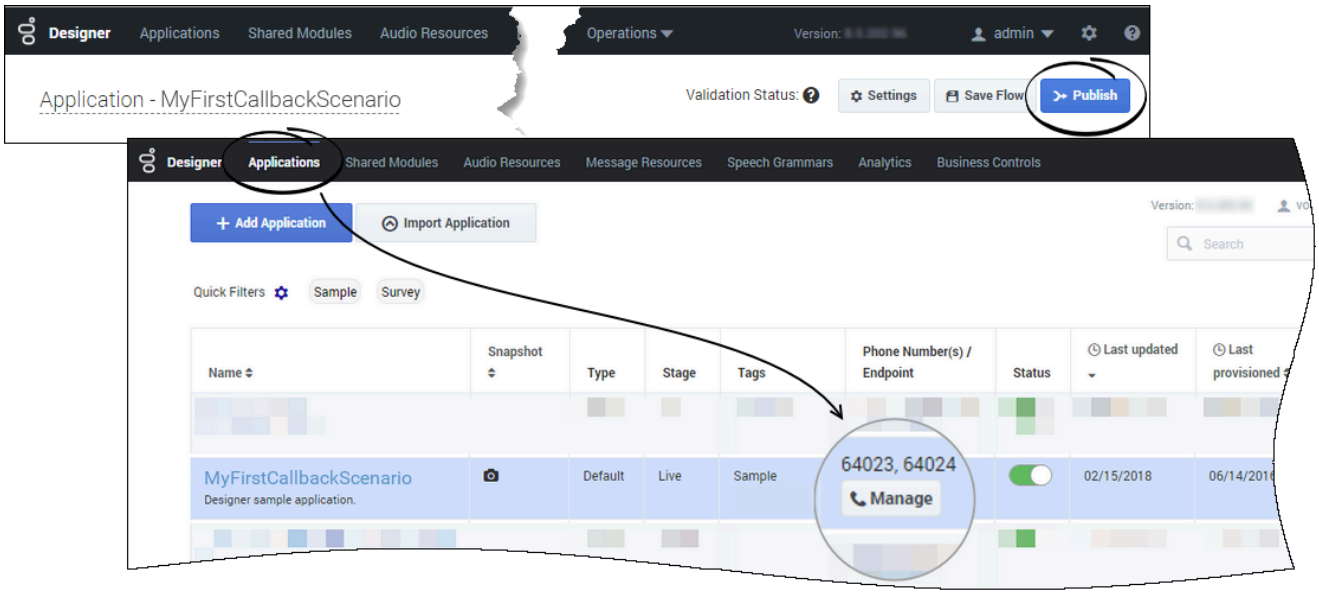


In Designer, select your Default-type application.

Scroll the **Palette** list to reach the **Callback** items. Drag and drop a **Callback V2** block into the **Assisted Service** phase of your application.

In the properties panel, under **Call Routing**, select the Inbound virtual queue that you configured for Callback.

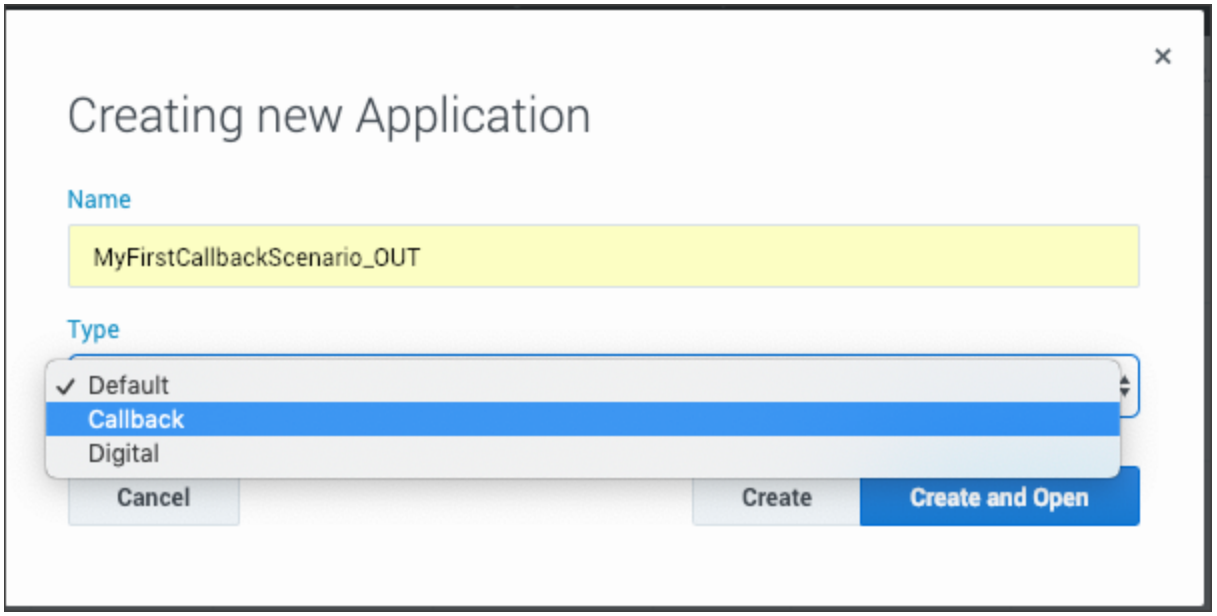
For additional information about the Callback V2 block properties, see the Designer documentation.



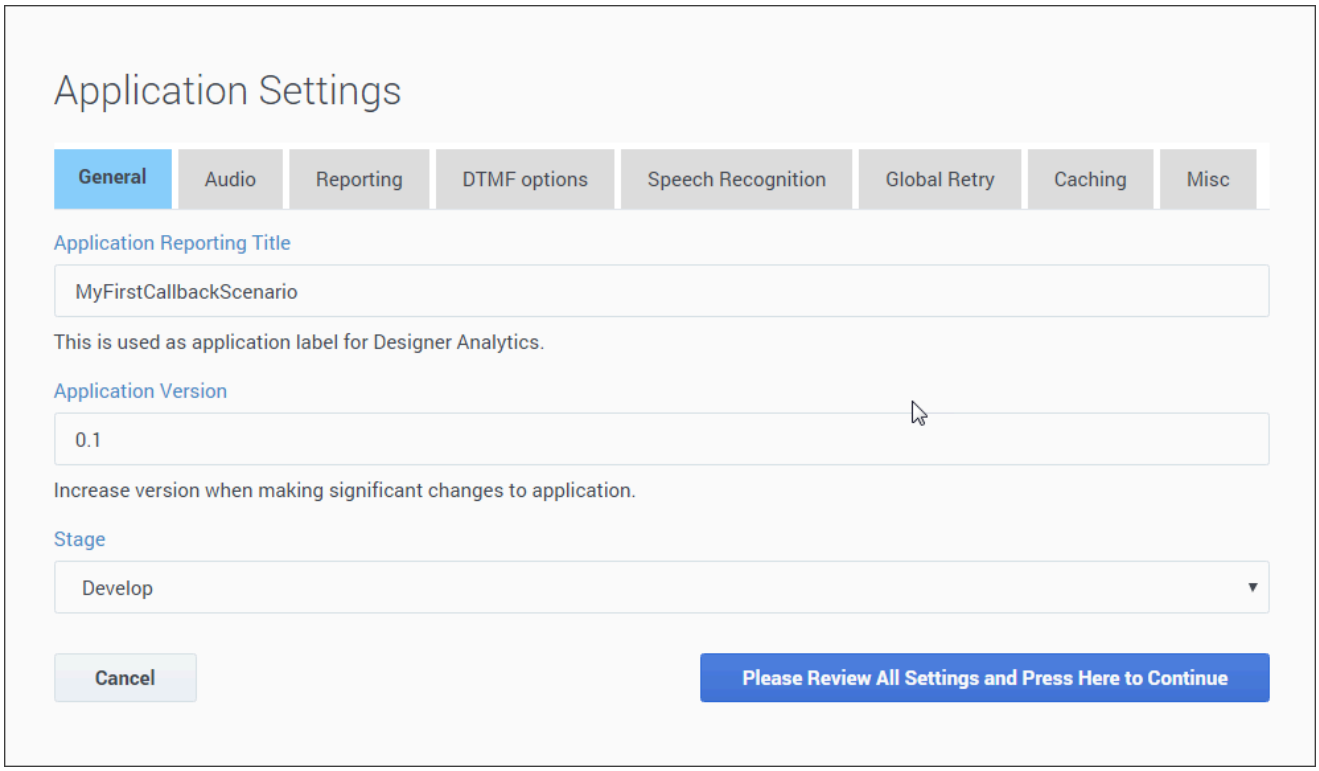
Publish your application and remember to assign a phone number to it.

Provision the Designer Callback Application

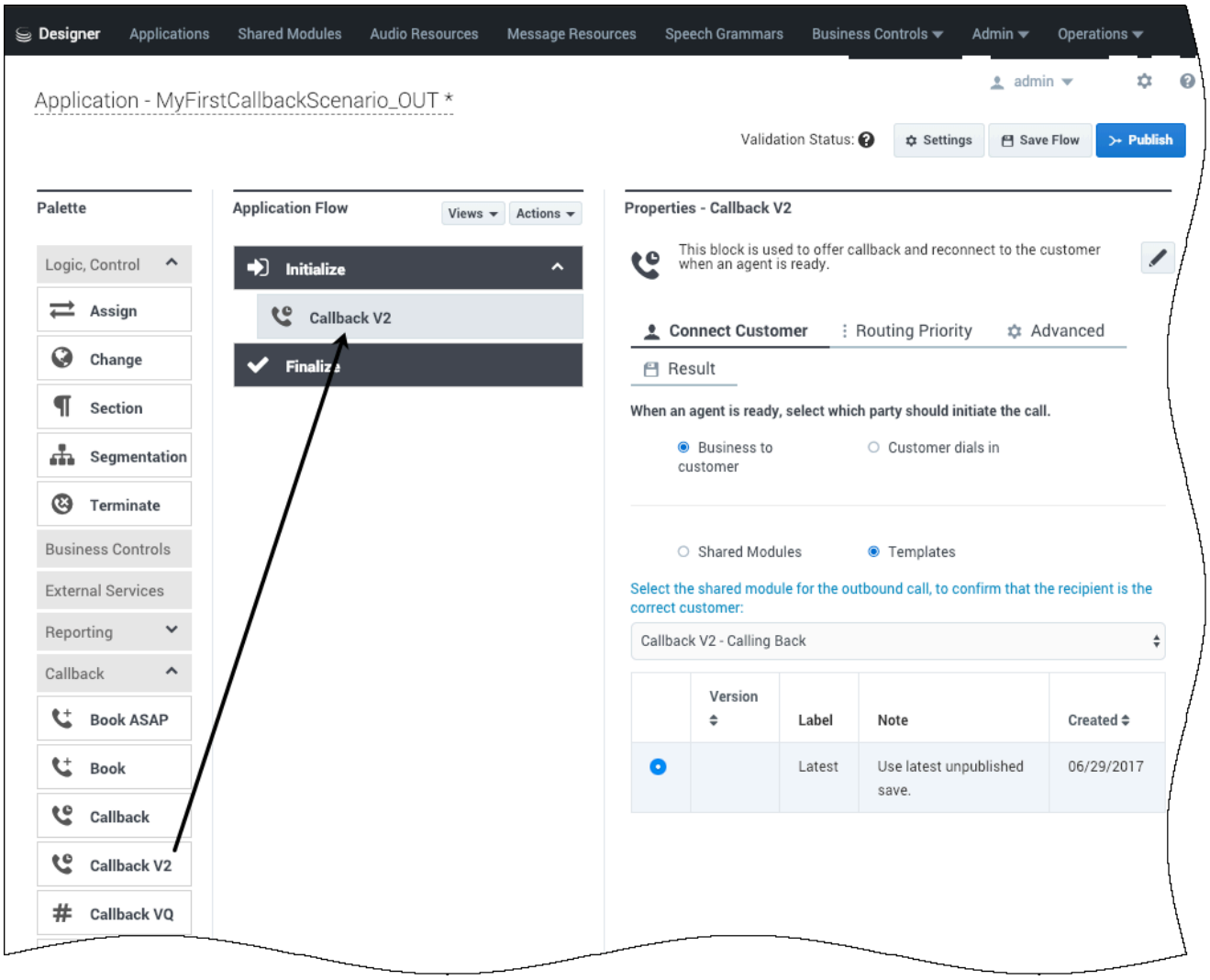
> **Important**
>
> You must create the Callback-type Designer application before you can edit the CALLBACK_SETTINGS data table.



In Designer, add a new application. The application type must be `Callback`. This application is used to re-connect with customers who requested a callback.



There are no mandatory settings changes for the Callback-type application, however, if there are any specific settings that you typically use for Designer applications, consider if those settings are required for your Callback application and make any necessary updates.



In Designer, select your Callback-type application.

Scroll the **Palette** list to reach the **Callback** items. Then drag and drop a **Callback V2** item into the **Initialize** section of your application. For information about the Callback V2 block properties, see the Designer documentation.

> **Important**

> In the Callback V2 properties panel, under **Connect Customer**, only the `Business to customer` option is currently supported.

Click **Publish** to make your application available. For information about publishing a Designer application, see Saving and Publishing Your Application.

Provision Business Hours for Callback

In Designer, you must configure the Business Hours object, including the timezone, before you configure the CALLBACK_SETTINGS data table. The time zone that you configure is used for scheduled callbacks. You cannot save the CALLBACK_SETTINGS data table before the business hours are configured.

For information about configuring your business hours in Designer, see Business Hours in the Designer documentation.

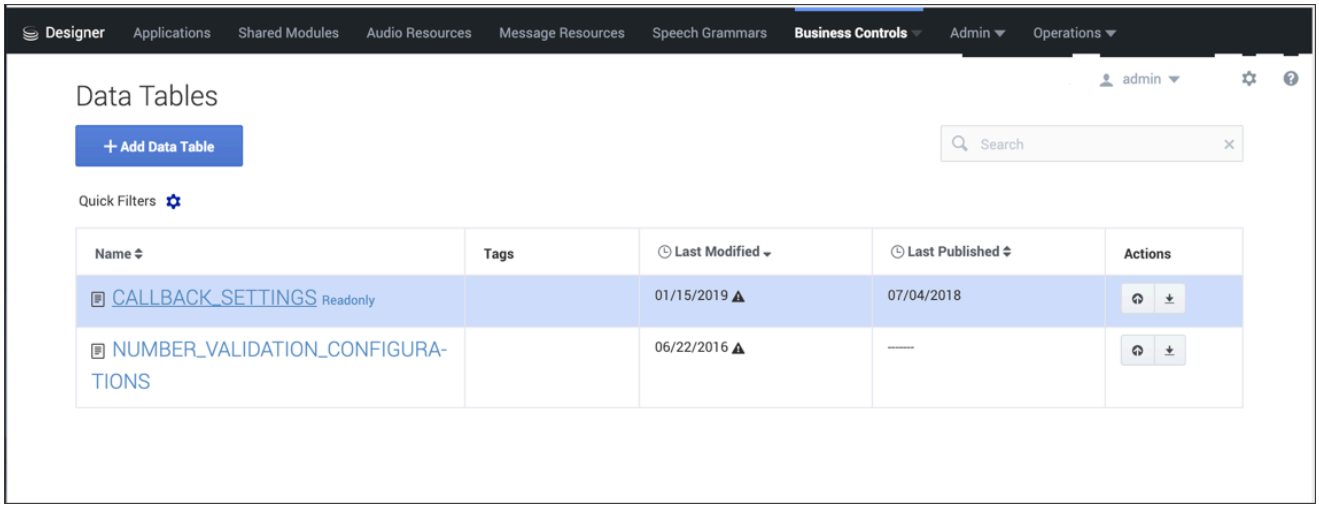
Provision the Callback Services

The configuration parameters for Callback services are stored in the Designer CALLBACK_SETTINGS data table. For detailed information about the CALLBACK_SETTINGS table, see Callback Settings Data Table in the Designer documentation.

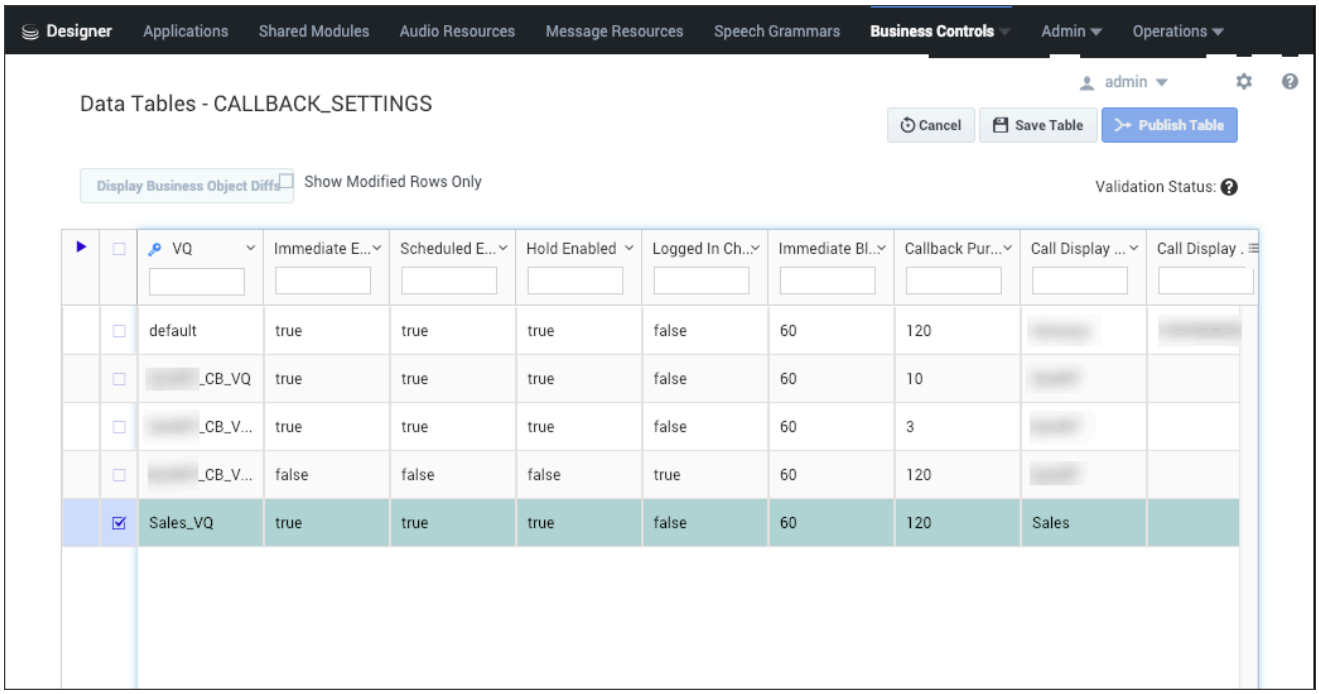
You must make sure that the following prerequisites are completed before you add your queue to the CALLBACK_SETTINGS table:

- You must create the Callback-type Designer application before you can edit the CALLBACK_SETTINGS data table.
- You must configure the business hours, including the time zone, before making the following updates to the CALLBACK_SETTINGS data table. You cannot save the data table before the business hours are configured.
- The virtual queues that you will use for callback functionality must be created and saved in Platform Administration.

Adding a Callback Virtual Queue to the CALLBACK_SETTINGS Data Table



In Designer, navigate to **Business Controls** > **Data Tables**. Click **CALLBACK_SETTINGS** and add an entry for your Callback Inbound queue as described here.



The `default` VQ row in the CALLBACK_SETTINGS table must include a valid Designer Callback-type application, a valid Business Hours object, and a semantically-correct skill expression.

In the row in which you are configuring the Inbound virtual queue, select your Designer Callback-type

application in the **Callback Application** column of the table. In the figure, this is the `Sales_VQ` row of the table.

If you defined Callback Skills for Callback agents, you can use this as a condition for the Inbound virtual queue.

For Callback-type applications, you must specify the routing point in the **Routing Point** column. The routing point for Callback-type applications is required for Callback to be fully functional. Outbound calls can sometimes fail if this is not configured.

Testing Your Callback Scenario

Now you can test the in-queue Callback scenario. Call the phone number that is assigned to the Designer Default-type application and accept an in-queue callback for an external phone number on which you can receive calls. If an agent with a matching skill is already logged in to the voice channel in Agent Desktop, you will receive a phone call on the external phone number. Once you have accepted the callback on the external phone, the call will be connected to the logged-in agent.

To test scheduled callback, call the phone number that is assigned to the Default-type application and choose `Scheduled Callback`. Listen carefully to the prompt. If it is using default settings, then it will ask you to specify a day and time for the callback based on the Pacific time zone. Also pay close attention to the actual date and time that you booked before accepting, especially if you have entered a time intended to be on the current day; the system might have offered you a time slot that is a week later. The outbound call experience is identical to the in-queue callback scenario.

To test web callback, see Managing Callbacks for information about creating a callback in the Callback application, or Genesys Engage REST APIs and Tutorials for Callback for information about creating a callback using the REST API.

Controlling User Access

> **Important**
>
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Controlling user access.

You can limit user access and activities in the Callback application to what is appropriate for each user's role in your enterprise. For example, to perform their duties, it is sufficient for some users to only view the list of callback records in the Callback application, without the ability to modify a callback in any way. Other users might require additional permissions; for example, some users might require access permissions that allow them to view errors associated with callbacks so they can troubleshoot problems.

You configure the restrictions on user access and activities for Callback in the Platform Administration application. If you do not configure access and activity restrictions for a user, then that user has full access to everything in the Callback application.

Genesys provides predefined Roles for the Callback application to ensure that your users have access that is appropriate for your business needs – such as the ability to view and modify Callbacks, use the **Developer** tab, and view Callbacks based on lines of business. For detailed information about Roles, see the Roles section in the Platform Administration documentation.

Migration of Roles

If you have not yet moved to the Roles and Access Group settings described on this page, the original method for configuring Callback access still applies to your setup. In the original configuration, Callback access was granted when the **ges** or **gms** section in the Person object's annex included a `role` option (for example, Administrator).

To move to the new configuration method for granting access, add your user to the correct Callback or custom Access Group and remove the **ges** or **gms** section. If you don't remove the **ges** or **gms** section, the old configuration applies and the Access Group is not taken into consideration.

Callback Roles

Genesys provides the following default Callback Roles:

- **Callback Administrator**—Callback Administrators have full access to the Callback application, which includes the ability to create, cancel, and reschedule callbacks, and to export reports. Users with this Role can also access all of the **Developer** tab features.

- **Callback Supervisor**—Supervisor users have full access to the Callback panel, which includes the ability to create, cancel, and reschedule callbacks, and export reports. They cannot access the **Developer** tab in the Callback application.
- **Callback Monitor**—Monitor users can only view callbacks.
- **Callback Developer**—Developer users can view callbacks on the **Callback** tab and have full access to the **Developer** tab, which includes the ability to view recent errors, test API keys, and to provision Callback.

If a user is a member of more than one Role, the Role that allows the most access to Callback features takes precedence.

Predefined Access Groups Supporting Callback

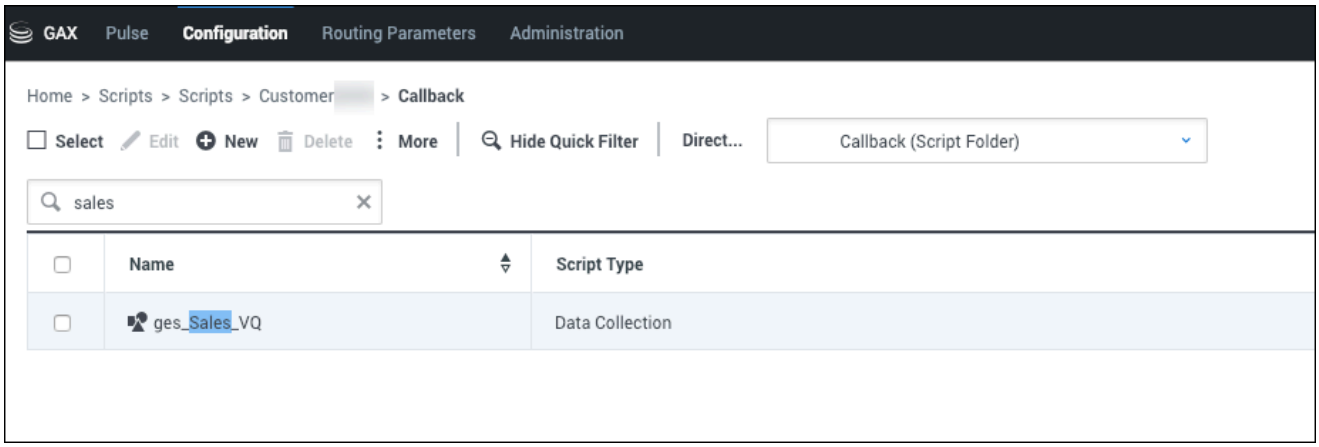
By default, Genesys defines a list of Access Groups and adds Callback Roles to some of these groups, as described below. Users who are already in these Access Groups are given Callback permissions by default. For example, any user in your Administrators Access Group is automatically granted the Callback Administrator role.

> **Important**
>
> The Access Group name is prefaced with your company's business name if the Access Group is not Callback-specific. For example, if your business name is ACME, then the Access Group for Administrators is called "ACME Administrators".

Access Group	Callback Administrator Role	Callback Supervisor Role	Callback Monitor Role	Callback Developer Role
Administrators	✓			
Supervisors		✓		
Managers			✓	
Callback Developers				✓

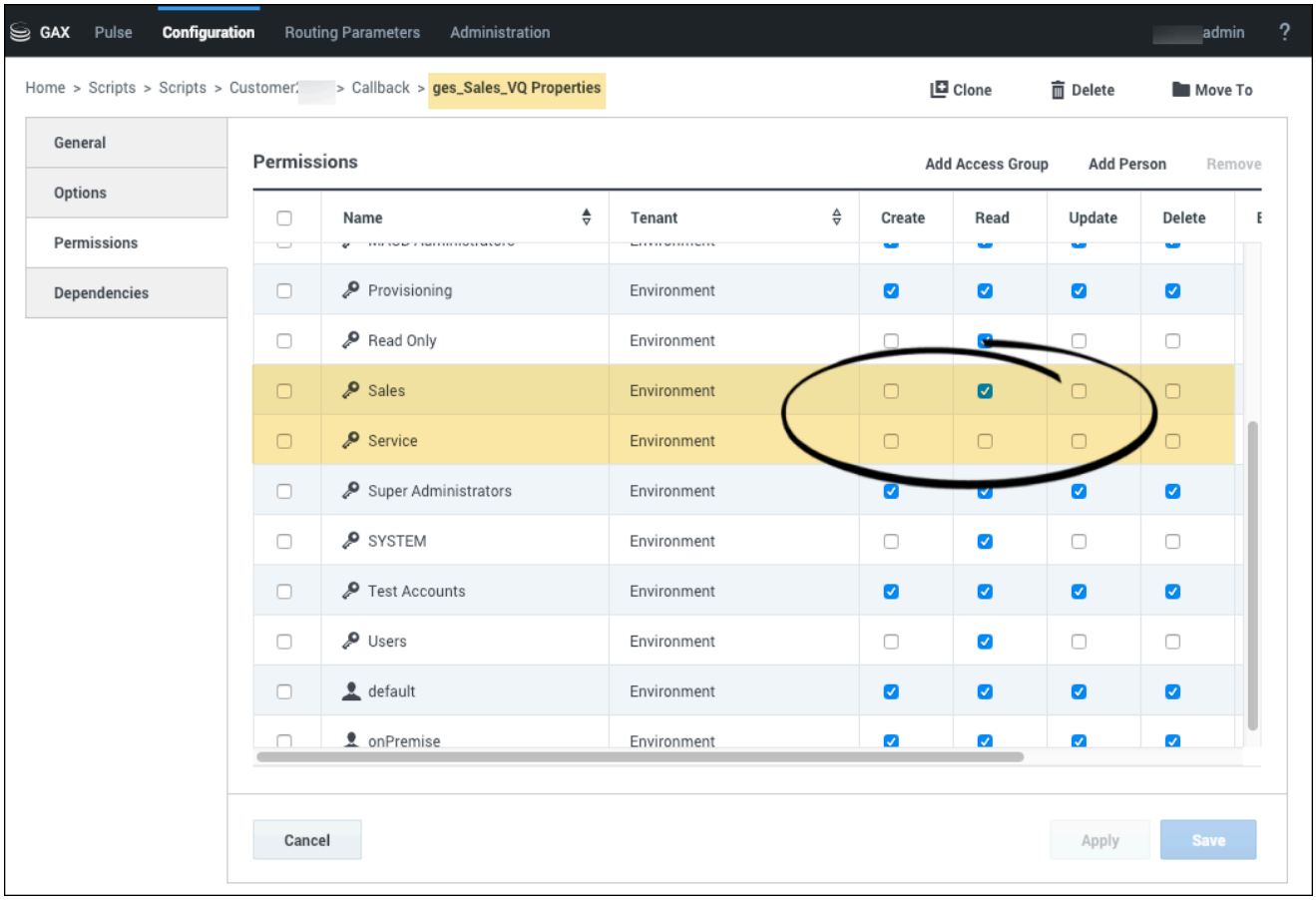
Line of Business Segmentation

By default, all users who are part of standard Access Groups and can access the Callback application will have Read permission for all the Virtual Queues. To restrict access to queues based on your lines of business, you can create custom Access Groups and enable or disable Read permissions as required.



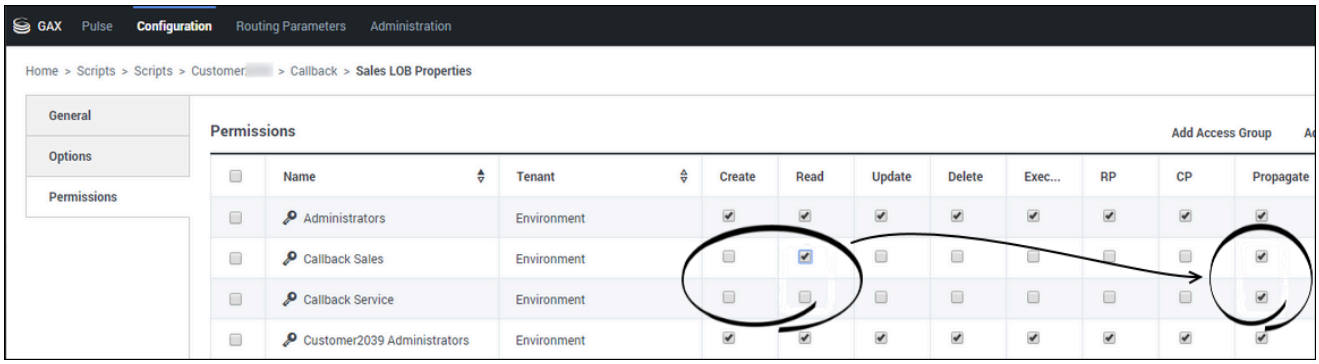
Any time you provision a Virtual Queue in Designer's CALLBACK_SETTINGS data table, the Callback service automatically creates a Script object in **Platform Administration** > **Scripts** > **Callback**. The created Script object has the same name as the Virtual Queue and is prefixed with the ges_ label. For example, if you create a Virtual Queue called `Sales_VQ`, there will be a Script object called `ges_Sales_VQ` in the Callback directory.

To control access to queues based on your lines of business, you must create Access Groups for your various lines of business and then enable or disable access to the script objects that represent the virtual queues for each group. For a user to access a specific queue, the Access Group to which the user belongs must have the Read permission for the script object that represents that queue. The Read permission is assigned by default to all Access Groups, which means that all Access Groups can access all Virtual Queues until you change the permissions. To deny access to a Virtual Queue, navigate to the Script object associated with the queue and remove the Read permission from Access Groups that do not require access to that queue.



For example, if your Tenant has two lines of business called **Sales** and **Service**, you could create two Access Groups for Callback: **Sales** and **Service**. Then, navigate to the script object representing the VQ and add that Access Group with read permission:

- In the **ges_Sales_VQ** Script object, retain the Read access for the **Sales** team and disable the Read permission for the **Service** team.
- In the **ges_Service_VQ** Script object, retain the Read access for the **Service** team and remove the Read permission from the **Sales** team.



To set permissions on groups of Virtual Queues (instead of one at a time), create subfolders under the **Scripts** > **Callback** folder and apply appropriate permissions to the subfolder. Then, move the Script objects representing the various Virtual Queues into the corresponding subfolder. Any Script object that is in a subfolder will inherit the permissions of that subfolder. Check the **Propagate** box to apply the permissions to any object that is in the folder. The permissions apply to any Virtual Queue that is in the subfolder now and will apply to any new Virtual Queue that you add to the subfolder in the future.

Managing Callbacks

> **Important**
>
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Managing callbacks.

You might see *Service* or *Service Name* mentioned in the Callback UI. A service, in this context, is a virtual queue. The service name, therefore, refers to the name of the virtual queue. On the **Callback** page in the UI, the **Service Name** column identifies, by name, the virtual queue associated with each callback.

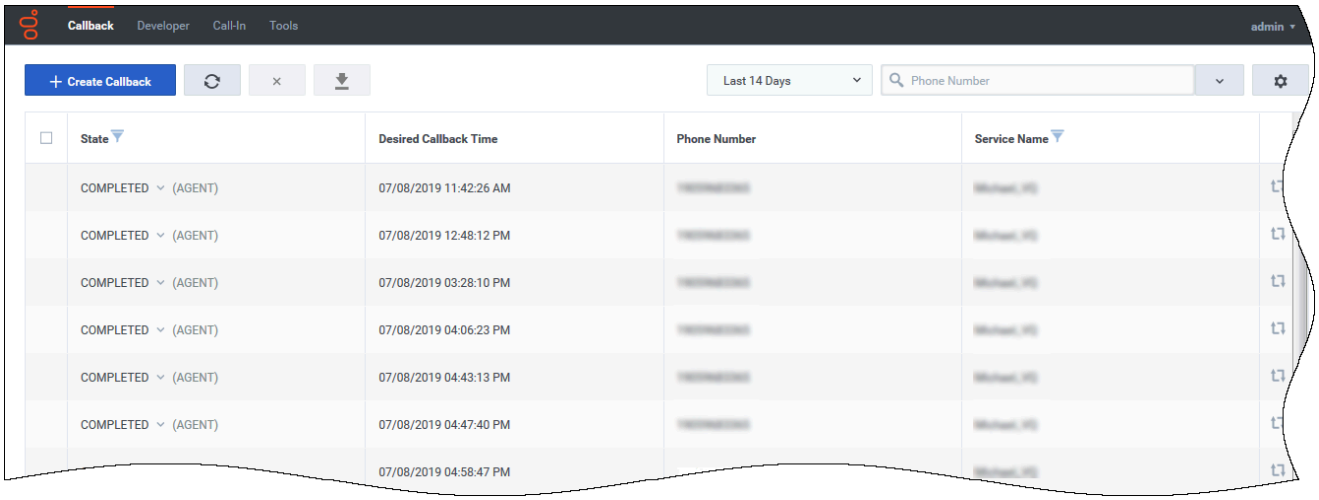
Callback Monitor is the minimum role required to access the **Callback** tab.

The **Callback** page in the Callback application displays the list of callback records. Callbacks are displayed in pages of 250 items. On this tab, you can create, edit, or cancel callbacks if you have sufficient user privileges. Your ability to access tabs and dialogs on the **Callback** page is based on your Role permissions.

A callback record will continue to display on the **Callback** page for a fixed number of days after the **Desired Callback Time** has passed; you might need to select a different time filter in order to see it.

> **Important**
>
> Callback records are stored for 14 days. The 14-day TTL setting starts at the **Desired Callback Time**. The `Callback TTL (seconds)` setting in the CALLBACK_SETTINGS data table has no effect on callback record storage duration; 14 days is a fixed value for all callback records.

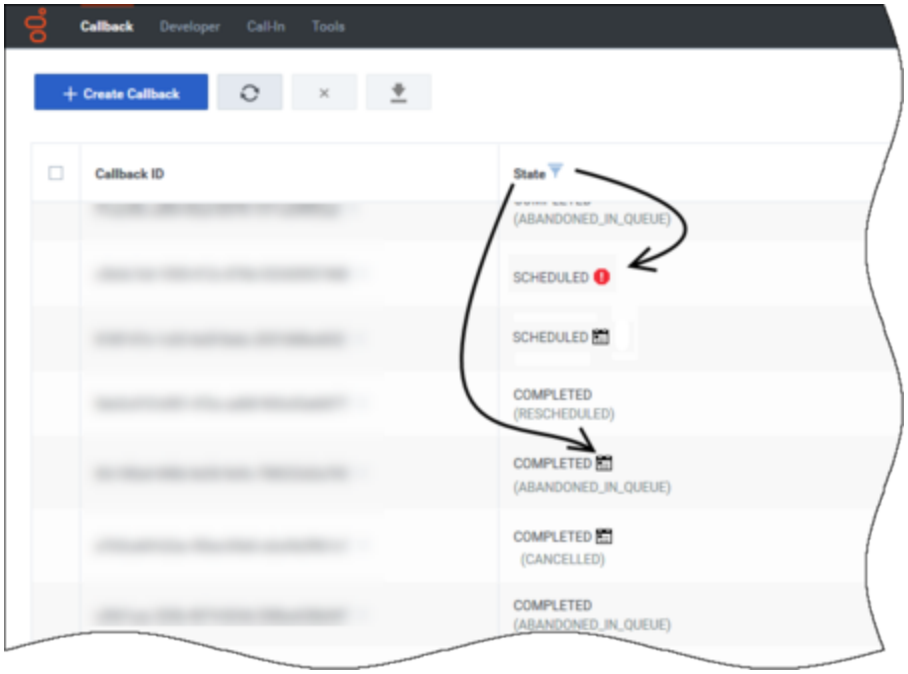
Callback states



Callback records can have the following states:

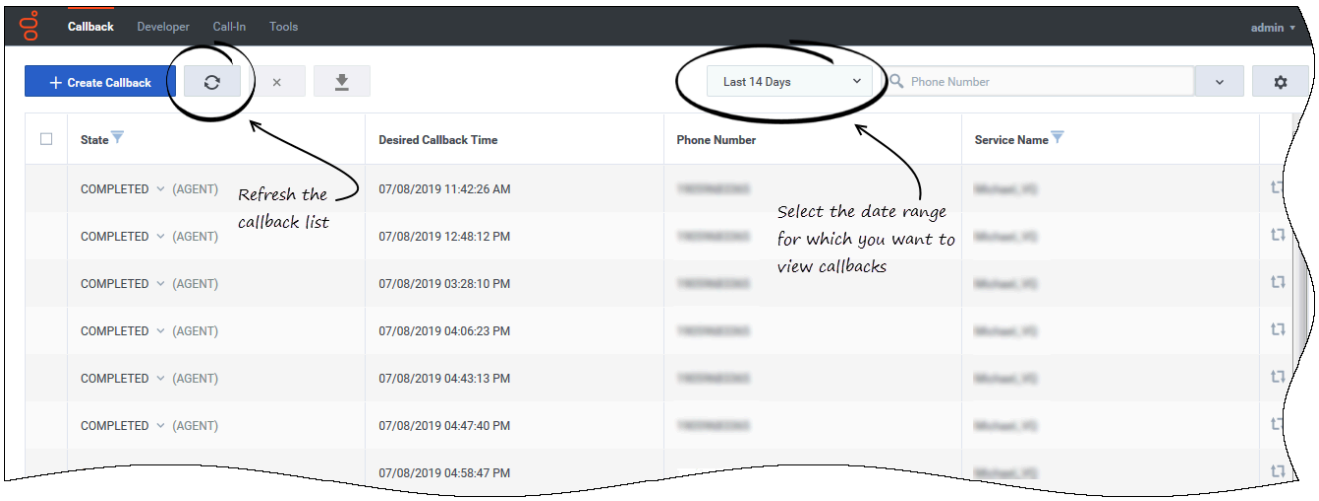
- SCHEDULED—Callback Management is handling the request (there are no sessions started in the system). While in this state, the request is available in the **Callback** tab when the specified **Desired Callback Time** is upcoming.
- QUEUED—Callbacks actively waiting for an agent in ORS/URS; the agent is not assigned yet.
- ROUTING—Agent is reserved but the call is not yet routed to the agent.
- PROCESSING—Callback being handled by assigned agents. You cannot cancel a callback when it is in the PROCESSING state.
- COMPLETED—Callback was completed with a result; for example, timed-out, cancelled, and so on. See the **Result** tab of your Callback block in Designer. The completed reason appears beside COMPLETED in the state column of the table.

When a callback in the SCHEDULED state remains in that state past the **Desired Callback Time**, a warning icon displays to alert you. Hovering your cursor over the icon displays the warning message.



A calendar icon displays beside callbacks that originated as scheduled callbacks. If there is no icon associated with a callback's state, then the callback was initiated as an immediate callback.

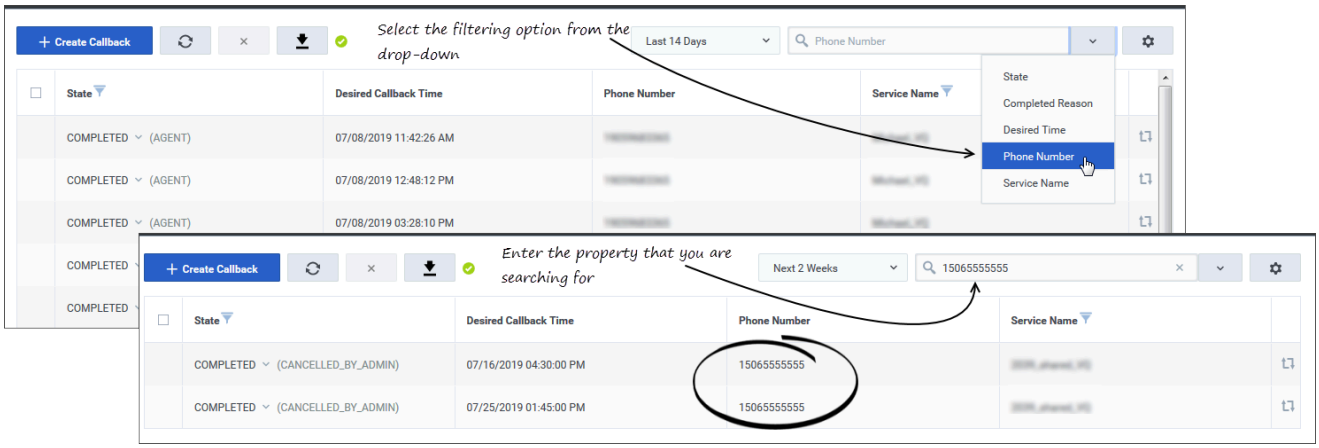
Callback Administration Overview



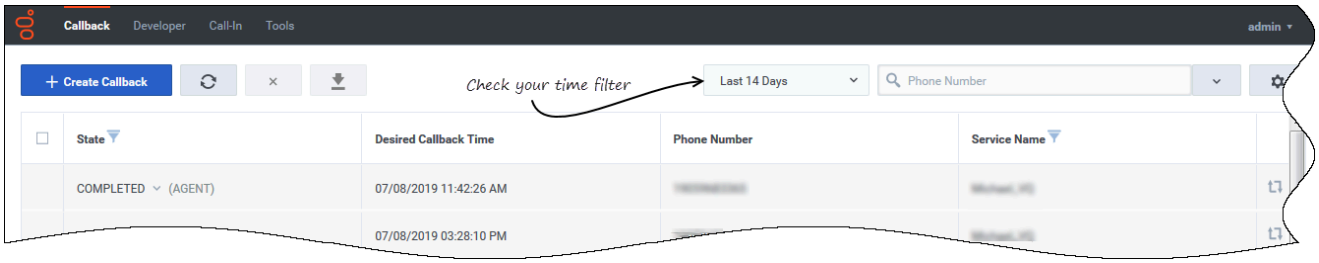
The **Callback** tab includes the following features:

- Refresh—You can force the interface to refresh the list of callbacks.
- Date range selector—The date range drop-down menu lets you filter the list of callback records based

on a specific (pre-defined) period of time. In the Advanced Options window, you can add your own custom date range to the drop-down menu.

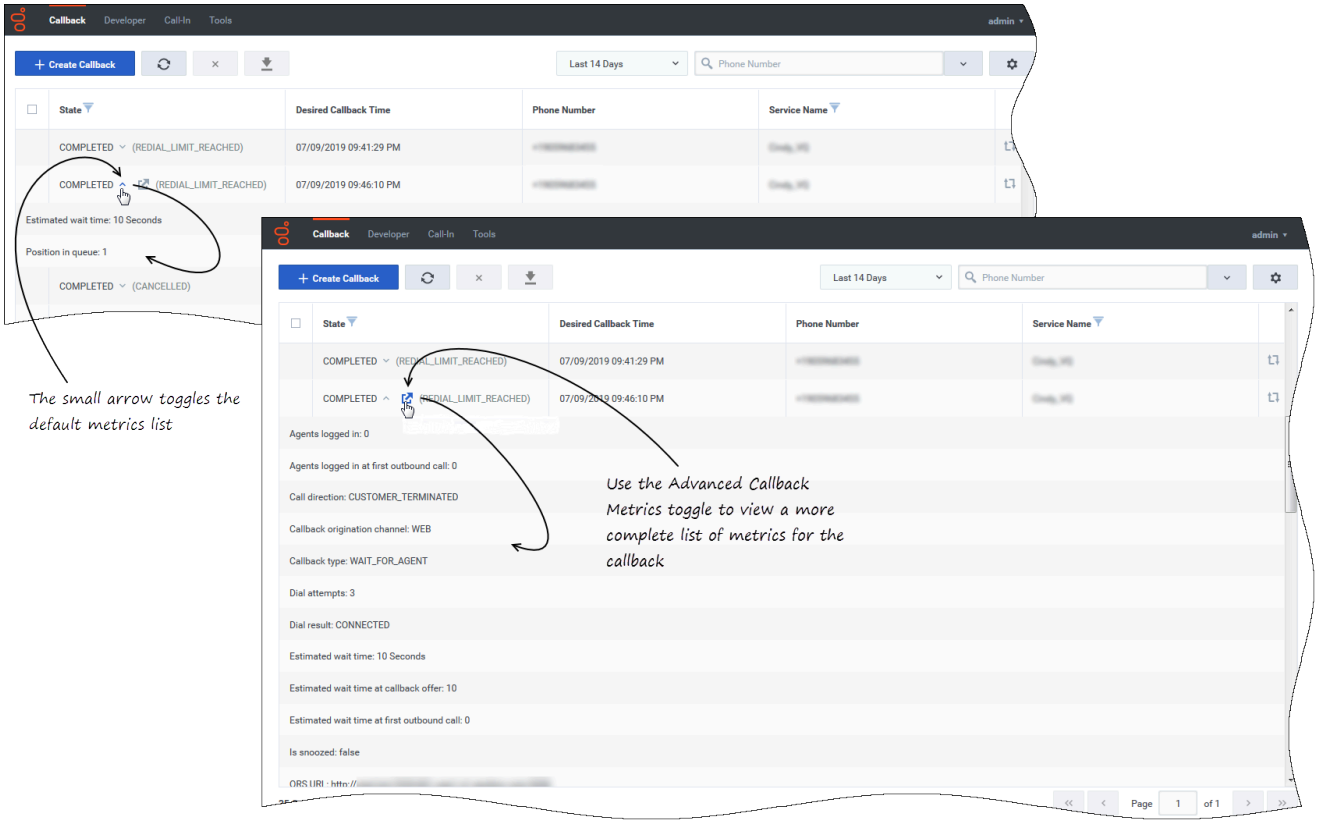


- Search box—You can select a column field in the drop-down at the right of the Search box. The interface then filters the results dynamically as you type.



If you do not see your callback after you successfully create it, make sure that your time filters are correct. For example, if you scheduled a callback to occur in two days, it does not show up if the **Last 30 Days** label is selected; switch to **Next 7 Days** for example.

Metrics



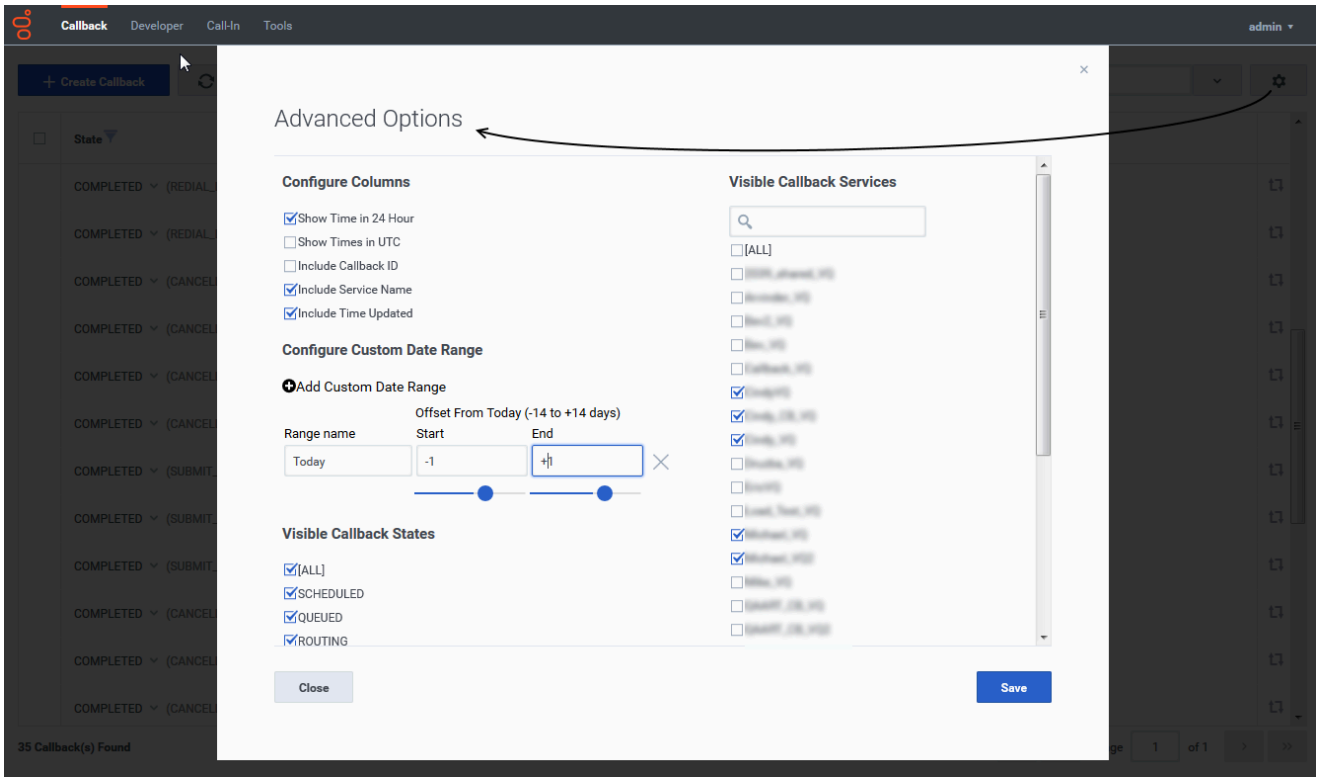
On the **Callback** tab, you have access to diagnostic data or *metrics* for each callback record. The metrics that are available for a callback depend on the callback's state. As the state of the callback progresses, more data becomes available, which means that additional metrics become available as well.

To view the metrics for a callback, click the arrow beside the record. Callback provides some basic information below the callback record (`Position in Queue` and `Estimated Wait Time`).

To view all metrics for the callback, click the `Advanced Callback Metrics` toggle. For example, you might want to know who cancelled a particular callback. That information is available in the list of advanced metrics.

> ### Tip
> The username of the person who created, cancelled, rescheduled, or recreated a callback is captured only when the action is performed in the Callback UI.

Advanced options

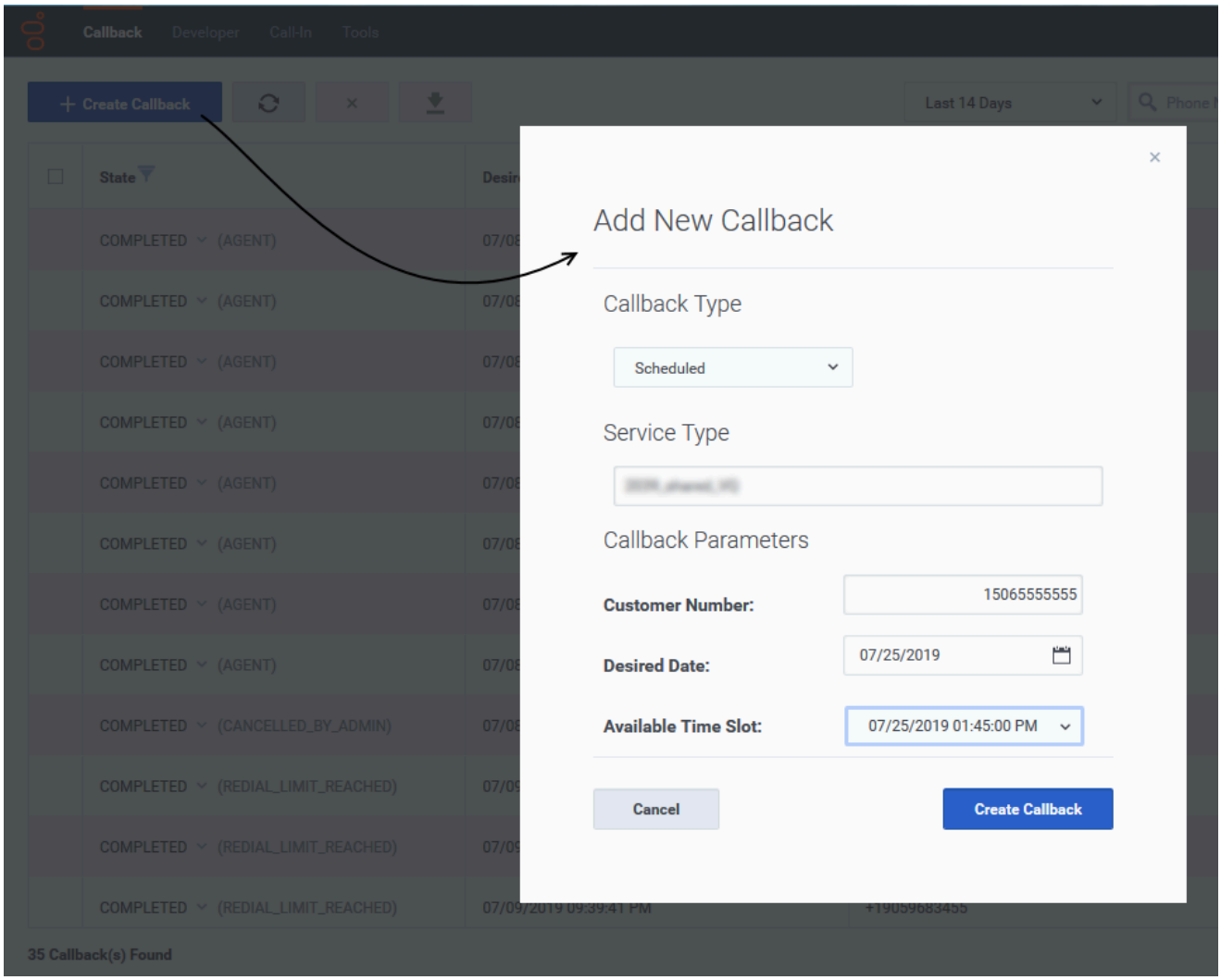


Advanced Options—Opens a new window where you can customize how the **Callback** tab displays data:

- **Configure Columns**—Select display properties for columns.
  - **Show Time in 24 Hour**—Show all times in 24-hour time rather than AM/PM format.
  - **Show Times in UTC**—Show all times in the table in UTC time rather than local time.
- **Configure Custom Date Range**—Add a time range filter.
- Filter the display of callbacks by **States** and **Services**—To filter the list of callbacks that display on the **Callback** tab, you can select specific callback states and virtual queues (services). To view callbacks in any state, select ALL in the **Visible Callback States** list. Similarly, to view callbacks from every virtual queue, select ALL in the **Visible Callback Services** list.

Create or Schedule a Callback

> **Important**
>
> This feature requires the **Callback Administrator** or **Callback Supervisor** role.



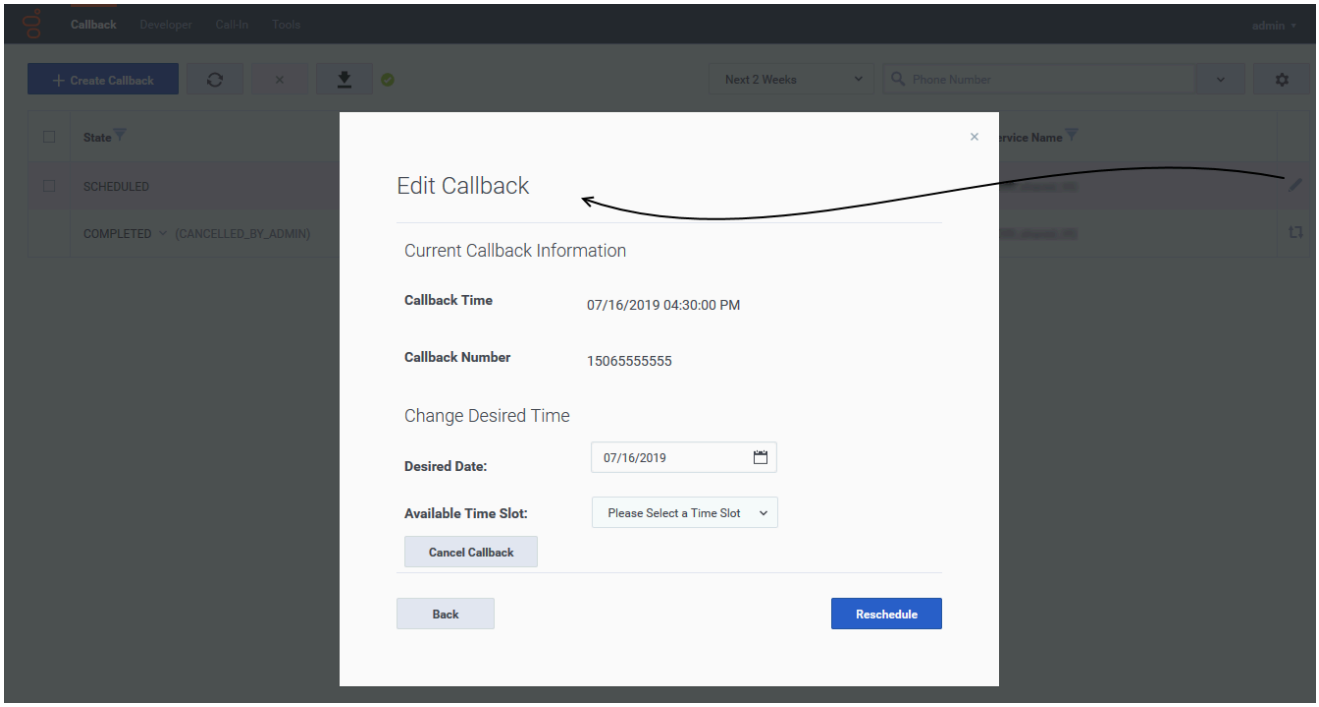
You can submit or schedule a callback request by clicking **Create Callback**. A new dialog opens. Select the **Callback Type** (`Immediate` or `Scheduled`), then a **Service Type** (the name of the virtual queue provided in Designer), and enter the number to call in the **Customer Number** field. For scheduled callbacks, select a date and time for the call.

Tip

If you provision a virtual queue for callbacks in Designer while you are logged in to the Callback UI, be aware that the new queue will not be available for use in the Callback UI until you log out and log in again because Callback fetches virtual queue permissions data only at login. In other words, Callback determines which virtual queues you have access to when you are logging in, and only then.

Edit a Callback

> **Important**
> This feature requires the **Callback Administrator** or **Callback Supervisor** role.

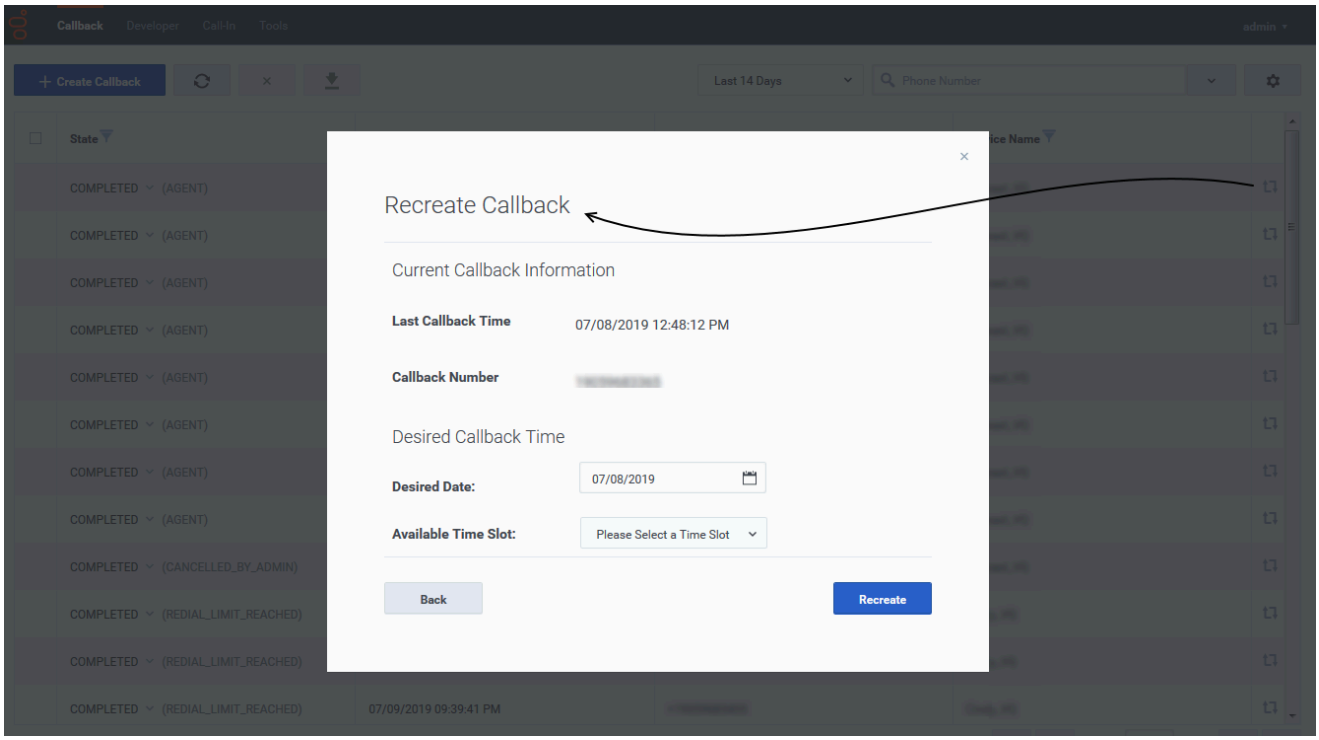


If a callback can be edited, a blue pencil displays in the last column of the table. Click the pencil icon to display the edit options for that callback:

- You can choose to reschedule the callback (only for SCHEDULED callbacks).
- You can choose to cancel the callback.

Recreate a Callback

> **Important**
> This feature requires the **Callback Administrator** or **Callback Supervisor** role.

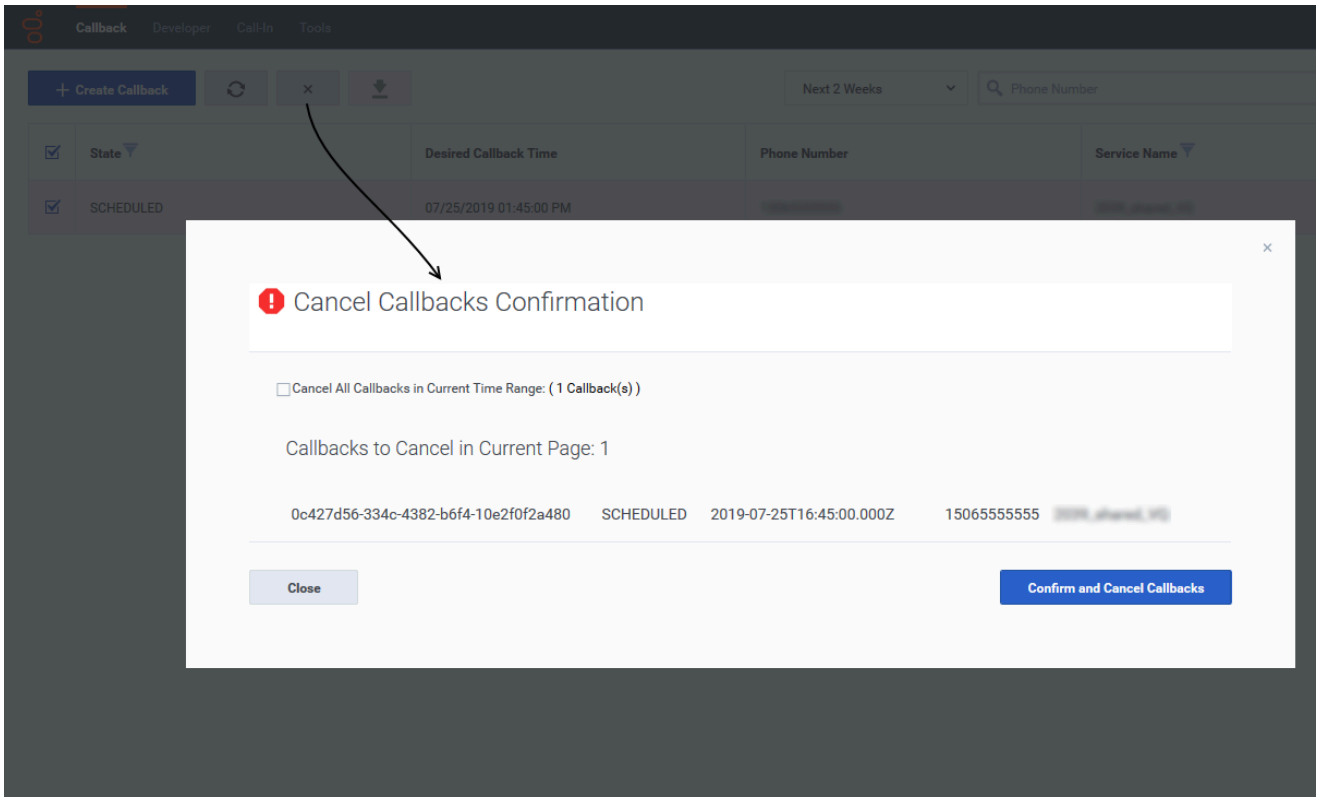


For completed callbacks that can be recreated, the  icon displays in the last column of the table. Click the icon to open the **Recreate Callback** dialog for that callback.

> ## Important
>
> Name recordings will be preserved if a callback is rescheduled within four days of the completion of the original callback.

Bulk Cancel and Export of Callback Records

> ## Important
>
> This feature requires the **Callback Administrator** or **Callback Supervisor** role.

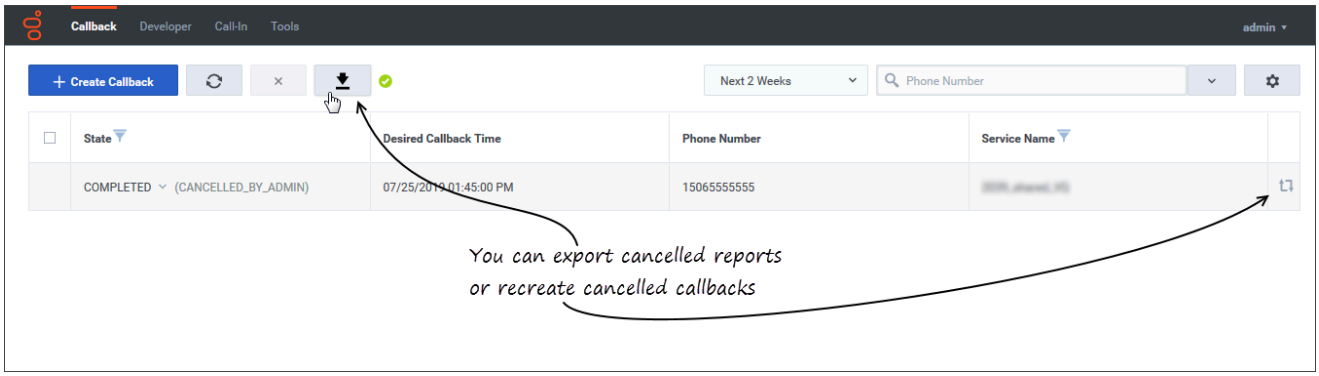


Make sure to select an appropriate time range to filter callbacks, then make a callback selection to activate the **Cancel Callbacks** button. When you click this button, the **Cancel Callbacks Confirmation** dialog opens, displaying the selected callback cancellations.

You can either:

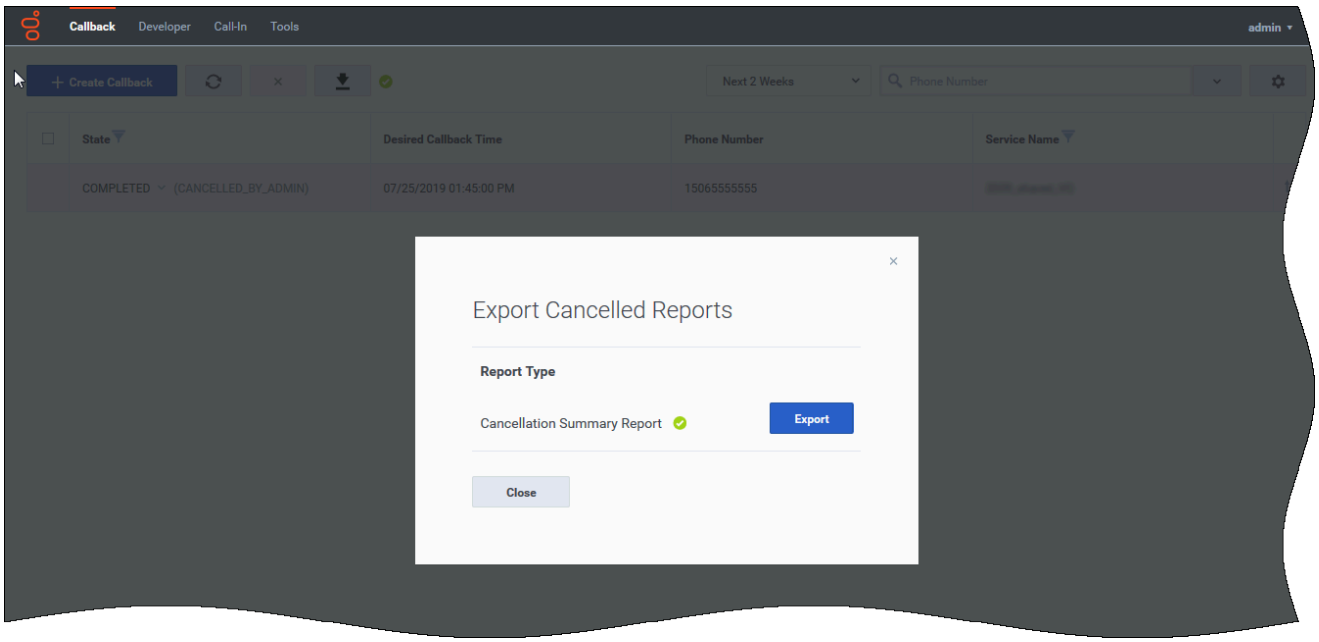
- Continue with the selections you have made and click **Confirm** to cancel them.
- Check **Cancel All Callbacks In Current Time Range** to override your previous callback selections and delete all callbacks in the current time range. In that case, the list of displayed callbacks disappears from the dialog. Callbacks that are in the PROCESSING state cannot be cancelled. If there are any callbacks in this state within the selected time range, they are excluded in the **Cancel Callbacks Confirmation** dialog box and are not cancelled.

Click **Confirm** to cancel the records.



The resulting display shows all the callbacks that you have cancelled and that are now in the `COMPLETED (CANCELLED_BY_ADMIN)` state.

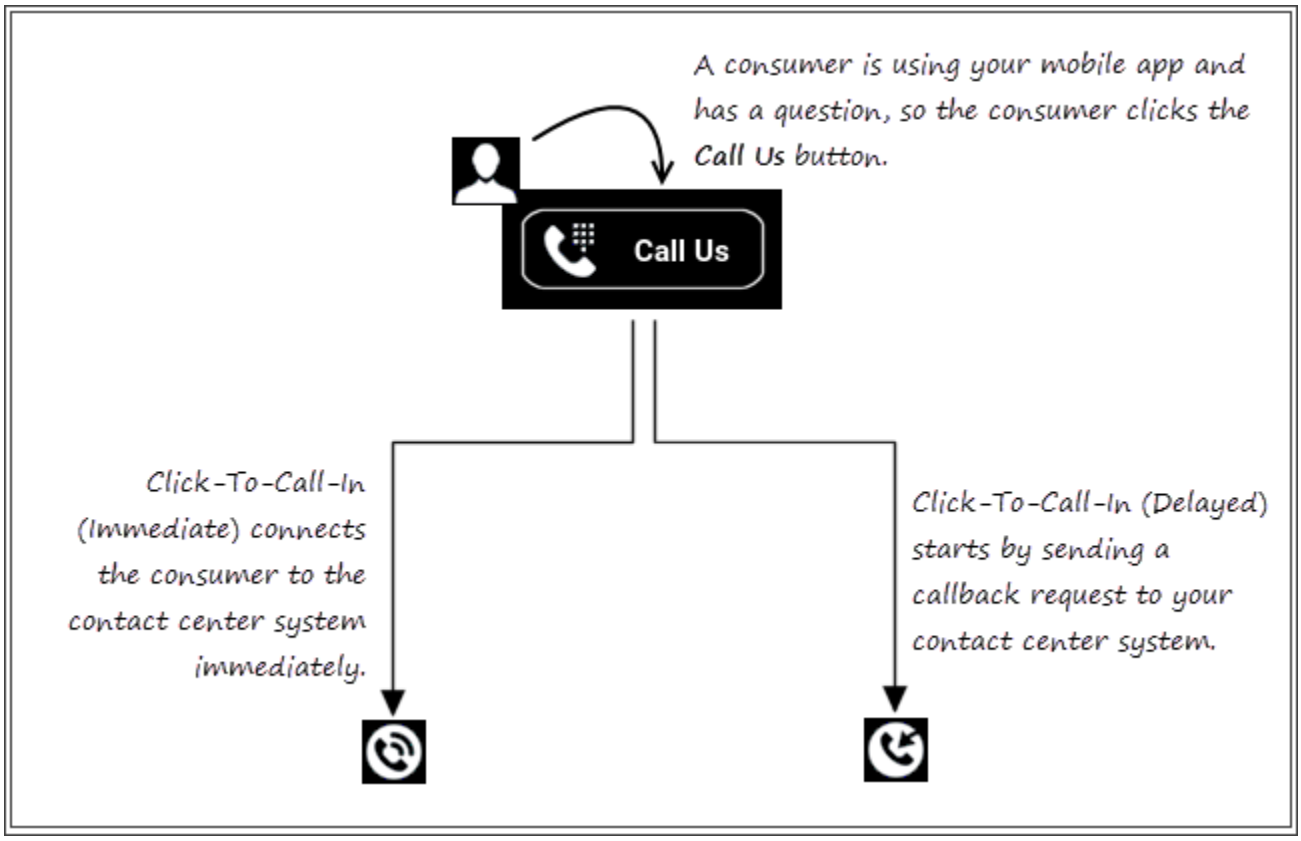
Download Reports



The **Export Cancelled Reports** dialog provides the **Cancellation Summary Report** file only if you recently cancelled some callbacks. This report shows the cancellation status of the recently-cancelled callbacks.

Provisioning the Click-to-Call-In Scenario

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Provisioning the Click-to-Call-In scenario.



You can use Callback's Click-To-Call-In feature to let consumers initiate a call to your contact center directly from your mobile app. Click-To-Call-In is advantageous because it can provide the context of the call to the agent. In addition, the data can be used for routing and reporting purposes.

You can implement the Click-To-Call-In functionality so consumers are immediately connected to the contact center; tapping the **Call Us** button in your app initiates the call. Depending on the Estimated Wait Time (EWT) on the queue, the consumer might be given the option to take a callback after they are connected to your contact center. The other option is to configure the feature to delay the call

until the system is sure that there is an agent available who has the correct skills to assist the consumer. In both scenarios – immediate or delayed connection to your contact center – you must configure Click-to-Call-In groups in the Callback UI. If you configure the delayed scenario, then you must also provision Push Notifications.

Click-To-Call-In (Delayed) Scenario Push Notifications

Push Notifications are not used with the Click-To-Call-In (Immediate) scenario.

If you use the Click-To-Call-In (Delayed) scenario with your mobile app, then you must provision Push Notifications. Your mobile app sends a callback request when the consumer taps the button or link that you have provisioned for the Click-To-Call-In (Delayed) feature. The callback request includes the Push parameters. For information about the Push Notifications supported with Callback, see Callback Push Notifications. For a description of the Click-To-Call-In (Delayed) scenario, see How Click-To-Call-In (Delayed) works.

Provisioning the Click-To-Call-In (Immediate) Scenario

To provision the Click-to-Call-In (Immediate) scenario, you will work in both Callback and in Designer. You can do the work in the Callback UI before you do the work in Designer or after. Either way, the configuration in both must be completed before you can test your Click-To-Call-In (Immediate) scenario.

The Click-To-Call-In (Immediate) scenario uses the Genesys Engagement Services (GES) Call-In API.

For information about the callback-related APIs, see Genesys Engage REST APIs and Tutorials for Callback.

Overview of the Provisioning Steps

You do not require a dedicated virtual queue for the Click-To-Call-In (Immediate) scenario. The consumer is simply calling into your contact center.

You must provision the following for the Click-To-Call-In (Immediate) scenario:

- In the Callback UI, provision the following:
  - Click-to-Call-In groups: Required by the system to determine which telephone number is appropriate for a Click-To-Call-In request. The system also uses the group configuration to determine whether or not an access code is required when the consumer dials in.
- In the Designer UI, provision the following:
  - A Click-to-Call-In Match application: This application must include the Click-to-Call-In Match Block. This Designer application must be of type `default`. During this part of the process, the system attempts to match a caller to a Call-In request based on data received with the request and data that the system provides in the response, such as the number that the caller dialed (the DNIS number), the access code (if required), and the caller's telephone number. If the attempt to match a caller to a Call-In request is successful and your Designer application is configured to route the call

when a match is made, then the call is queued on hold like any other call to the contact center.

Provisioning the Click-To-Call-In (Delayed) Scenario

To provision the Click-to-Call-In (Delayed) scenario, you will work in both Callback and in Designer. You can do the work in the Callback UI before you do the work in Designer or after. Either way, the configuration in both must be completed before you can test your Click-To-Call-In (Delayed) scenario.

The Click-To-Call-In (Delayed) scenario uses the following Genesys Engagement Services (GES) APIs:

- Callback
- Call-In

For information about callback-related APIs, see Genesys Engage REST APIs and Tutorials for Callback.

Before You Start

Before you start provisioning Designer applications for Click-To-Call-In (Delayed), create a virtual queue in Platform Administration for this purpose. You need to add the queue to the CALLBACK_SETTINGS data table and then configure it, so be sure to give it a name that helps to identify it; for example, ClickToCallInDelayed_VQ.

Overview of the Provisioning Steps

You must provision the following for the Click-To-Call-In (Delayed) scenario:

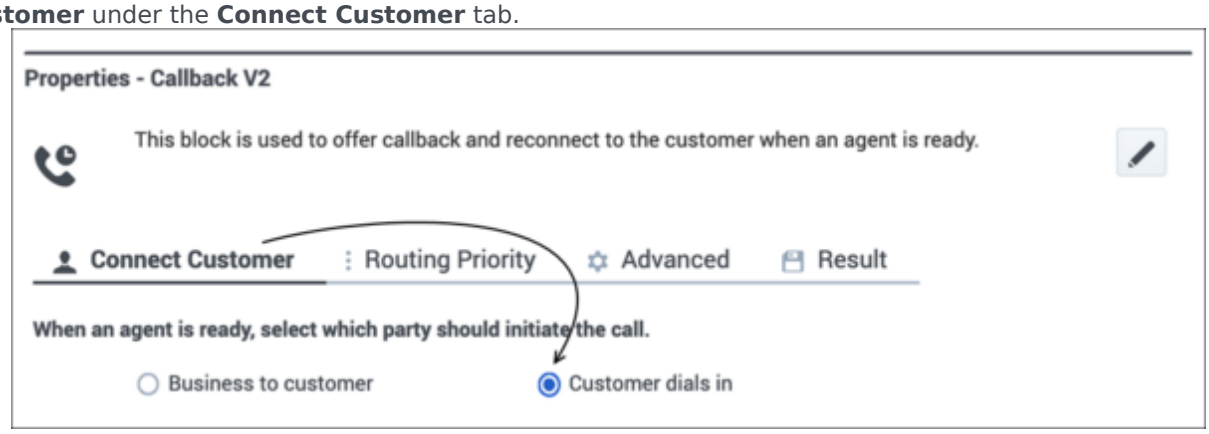
- In the Callback UI, provision the following:
  - Push Notifications: Required for communicating information about the call-in to the consumer such as when the callback is ready, what number to call and – if required – what access code to enter.
  - Click-to-Call-In groups: Required by the system to determine which telephone number is appropriate for a Click-To-Call-In request. The system also uses the group configuration to determine whether or not an access code is required when the consumer dials in.
- In the Designer UI, provision the following:
  - A Callback application specifically for Click-To-Call-In (Delayed): Used to start the callback process. This Designer application must be of type `Callback`. During this part of the process, the call is queued in the Click-To-Call-In Delayed virtual queue.
  - The Click-To-Call-In Delayed virtual queue in the CALLBACK_SETTINGS data table.
  - A Click-to-Call-In Match application: Used when the customer calls the phone number that the system provided in a Push Notification. This Designer application must be of type `default`. During this part of the process, the system attempts to match the caller to an existing callback record and, if successful, routes the call to an available agent. For more information about the Click-to-Call-In Match block, see Click-to-Call-In Match Block.

Every environment is different and requires its own specific routing logic, error handling, and so on. That means that, in addition to the configuration described in this section, you must consider what customized configuration you require in the Designer applications and in the virtual queue used with

the Click-To-Call-In (Delayed) scenario. This section describes only the basic configuration that is required in all environments to provision the Click-To-Call-In (Delayed) scenario in Designer.

Provision the Click-To-Call-In Delayed Callback Application

1. In Designer, create a new application. Select `Callback` for the type.
2. Add a **Callback V2** block to the **Initialize** phase.
3. In the properties of the **Callback V2** block, select **Customer dials in**, rather than **Business to customer** under the **Connect Customer** tab.



4. In all other regards, configure the application as required for your contact center.
5. Save and publish your Click-To-Call-In (Delayed) Callback application.

Configure Settings for the Click-To-Call-In Delayed Virtual Queue in the Data Table

1. In Designer, navigate to **Business Controls** > **Data Tables** and open the CALLBACK_SETTINGS data table.
2. If you have not yet done so, add the Click-To-Call-In Delayed virtual queue to the table; otherwise, find the queue in the table.
3. Make sure that you configure all of the following settings in the data table for this queue, in addition to any other parameters that you choose to customize for your environment:
   - **Pushed Callback Expiry Time (minutes)**. This setting specifies how long the callback session stays alive after the system sends the initial Push Notification to the consumer. If the callback expires, any subsequent Call-In request related to that callback session will fail.
   - **Push Notification Threshold (minutes)**. This setting specifies when to send the CALLBACK_UPCOMING Push Notification to the consumer. For example, if you specify a value of 1, then when the Estimated Wait Time (EWT) on the queue drops below 1 minute, the system sends a CALLBACK_UPCOMING Push Notification to the consumer if conditions are favorable for the callback to proceed. For detailed information about the CALLBACK_UPCOMING Push Notification, including information about cases in which the system does not send this notice, see Callback Push Notifications.
   - **Call Direction.** This must be set to `USERORIGINATED`.

- **Callback Application**. Select the `Callback`-type application that you created for the Click-To-Call-In (Delayed) scenario.

  The **Callback Purge Time** and the **Business Hours** that you specify in the CALLBACK_SETTINGS data table for the Click-To-Call-In Delayed virtual queue are applied to this queue just as they are applied to any other queue. Those two settings are relevant when the system must determine if the callback can be purged or not. In addition, Genesys recommends that you configure the **Skill Expression** for the virtual queue to ensure that the calls are routed to the agents who can best serve the consumers who are calling in.

  > **Tip**
  > The callback will be purged at the end of the day based on the specified business hours, regardless of the **Callback Purge Time** status.

4. Save and then publish your updated data table.

Provision the Click-to-Call-In Match Application in Designer

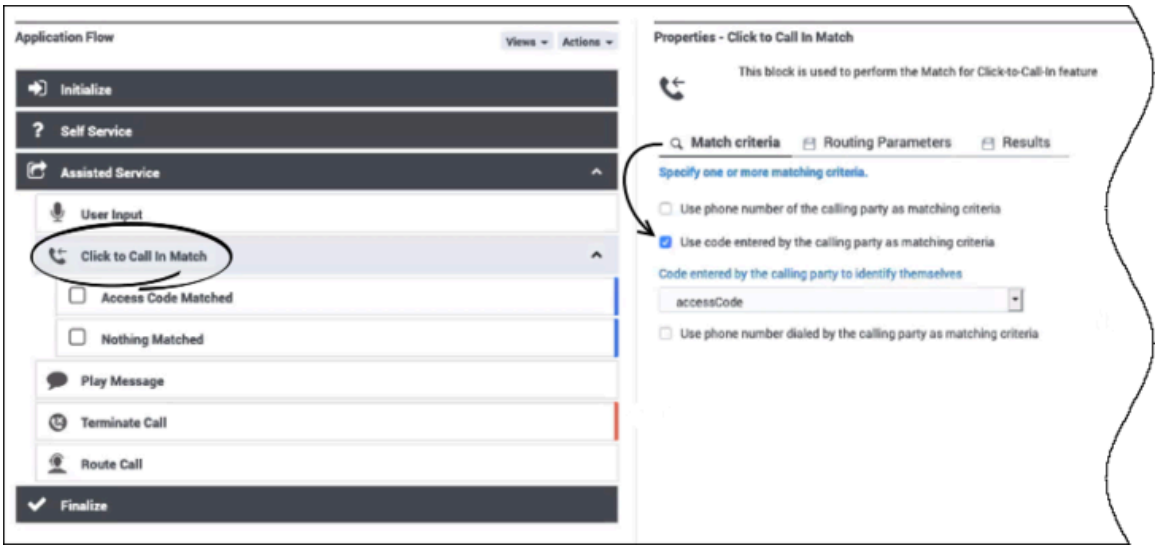
In addition to the `Callback`-type Designer application, you also require a Click-To-Call-In Match application for the Click-To-Call-In (Delayed) scenario. When a Push Notification sends the number and optional access code to the consumer, then the consumer calls in. The Match application matches the caller to the original callback request. Based on the results of the matching attempt, the application either routes the call to an agent (successful match) or the call moves to the next block (no match). The routing is performed by a **Route Call** block.

The Match application must be a `Default`-type application. It can be used with multiple Click-to-Call-In Delayed virtual queues and multiple Click-to-Call-In Groups.

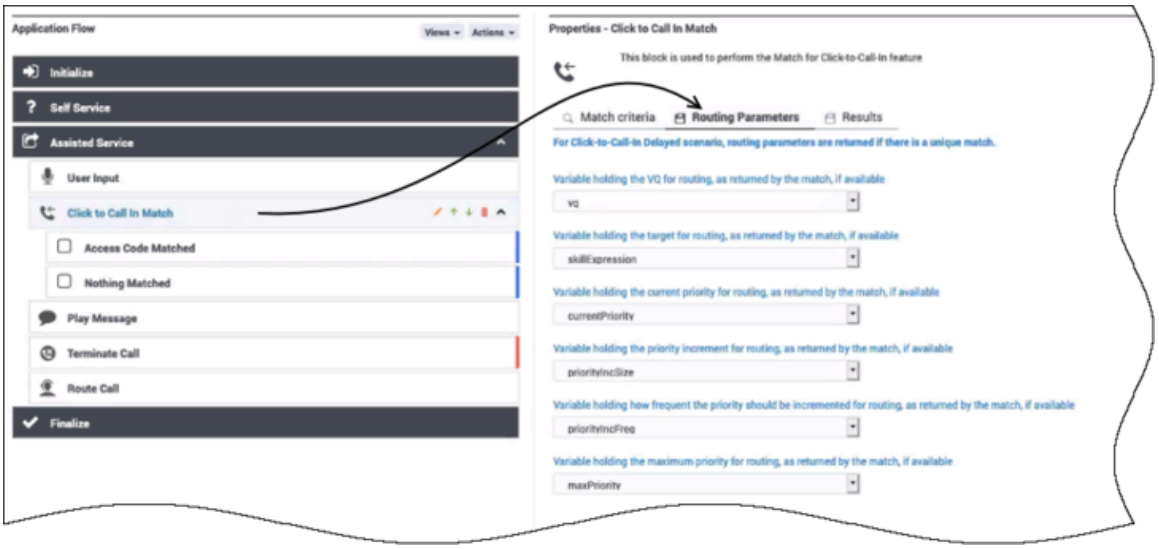
Sample Configuration

For the purposes of the following procedure, let's assume that consumers who are using the Click-To-Call-In feature are prompted to enter an access code. We'll have the application perform the match only on the access code.

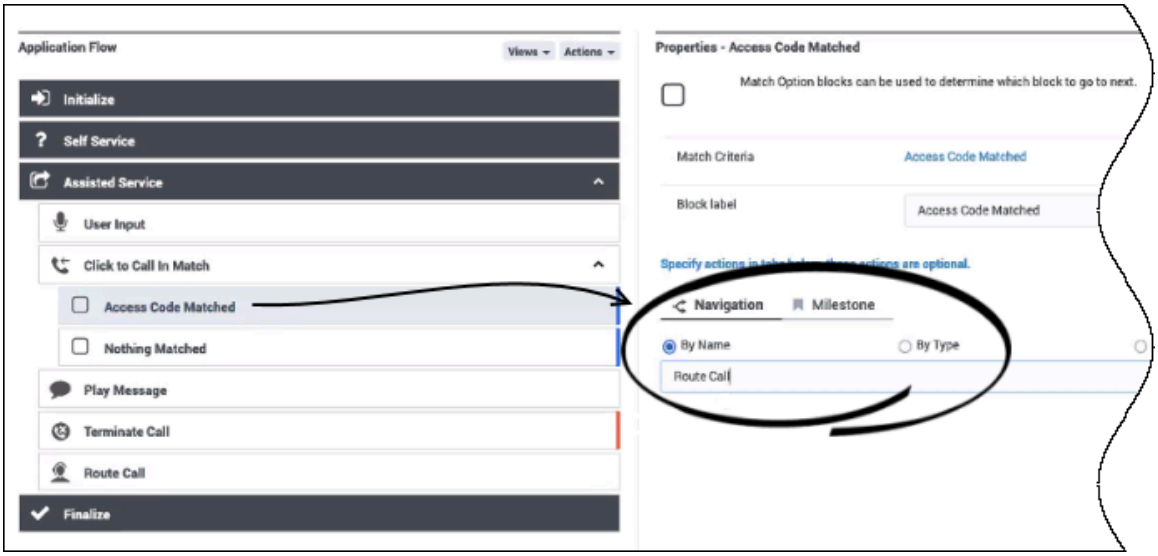
1. In Designer, create an application of type `Default`.
2. Click the **Initialize** phase and add the variables that you will use to store block outputs.
3. We will add the following blocks to the **Assisted Service** phase to handle the call flow:
   - User Input: This is required when you are using access codes with the Click-To-Call-In (Delayed) scenario. You configure this block to prompt the user to enter the access code.
   - Click to Call In Match: In the Click To Call In Match block, on the **Match criteria** tab, select the option to use the access code that the consumer enters as the matching criterion.



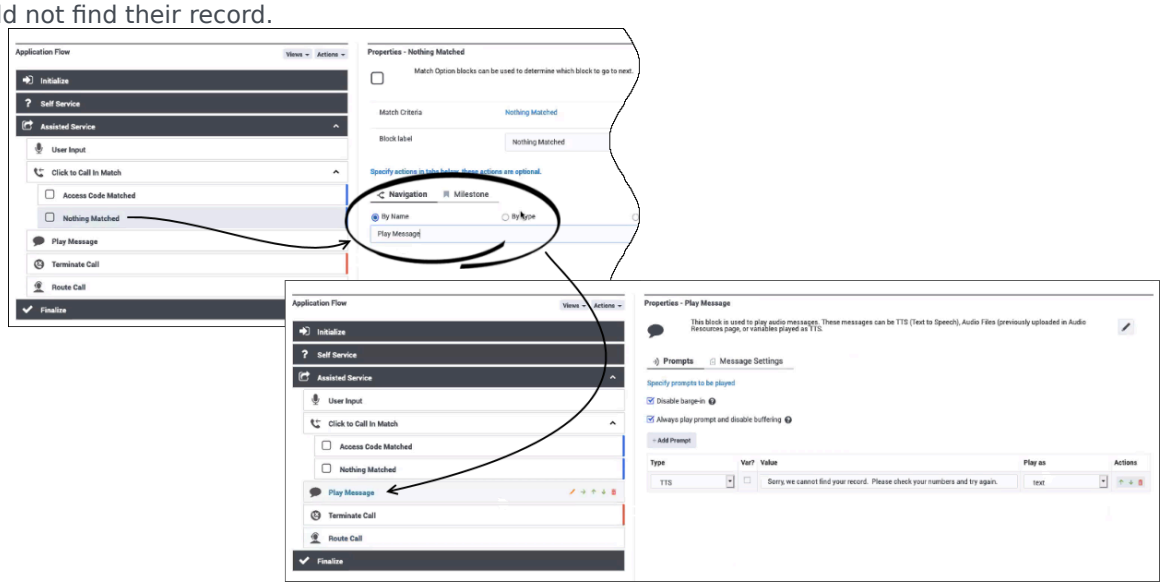
Make sure that you configure the parameters on the **Routing Parameters** tab. As with any application, consider the logic that needs to be configured to handle all outcomes appropriately for your contact center.



If a match is made between the caller and an existing callback record, then the call needs to move to the Route Call block.



- Play Message: When the system can find no successful match for the caller or the caller is dropped from the Match block for any reason, then the Play Message block informs the caller that the system could not find their record.



- Terminate Call: When no match can be found between a caller and existing callback records, and the message in the Play Message block has finished playing, then the call drops to the Terminate Call block and the call ends.
- Route Call: When a caller matches a callback request that exists in the system, then the system routes the call based on the information in this block. The routing priorities, virtual queue, skill expressions, and so on are included in the **Routing Parameters** tab of the Click To Call In Match block. You must assign the variables used in the **Routing Parameters** tab to the corresponding fields in the Route Call block in order for the call to retain its priority from the original callback request.

Viewing Click-To-Call-In records

> **Important**
>
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Viewing Click-to-Call-In records.

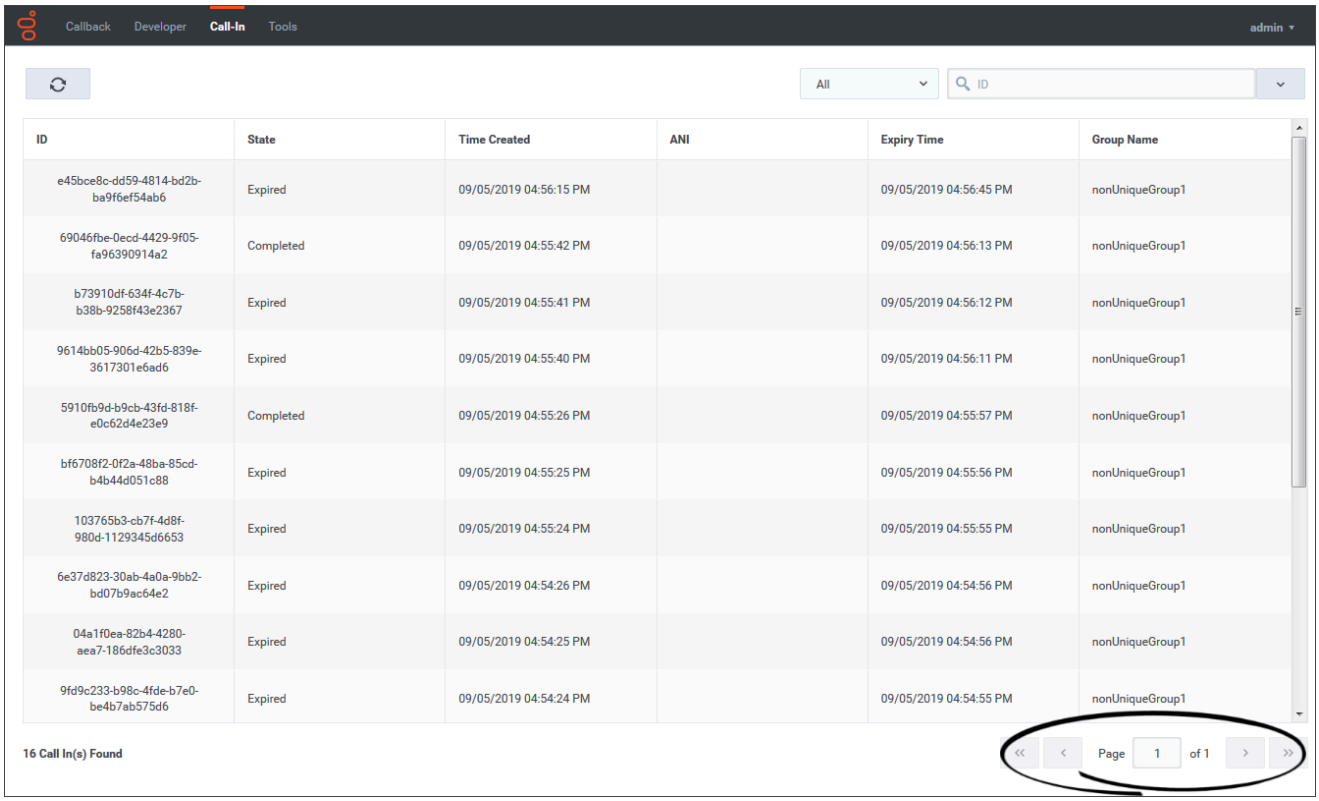
Callback supports the Click-To-Call-In scenario. Use the **Call-In** tab in the Callback UI to view the Click-To-Call-In records.

> **Important**
>
> To view the **Call-In** tab, you must be a member of the **Callback Administrator** or **Callback Developer** role.

For information about provisioning the Click-To-Call-In groups that are required for Callback Click-To-Call-In functionality, see Configuring Click-To-Call-In Groups.

You can find information about using the Genesys Engagement Service (GES) APIs, including the Click-To-Call-In API, on the Genesys Engage REST APIs and Tutorials for Callback page in this Guide.

Using the Call-In tab



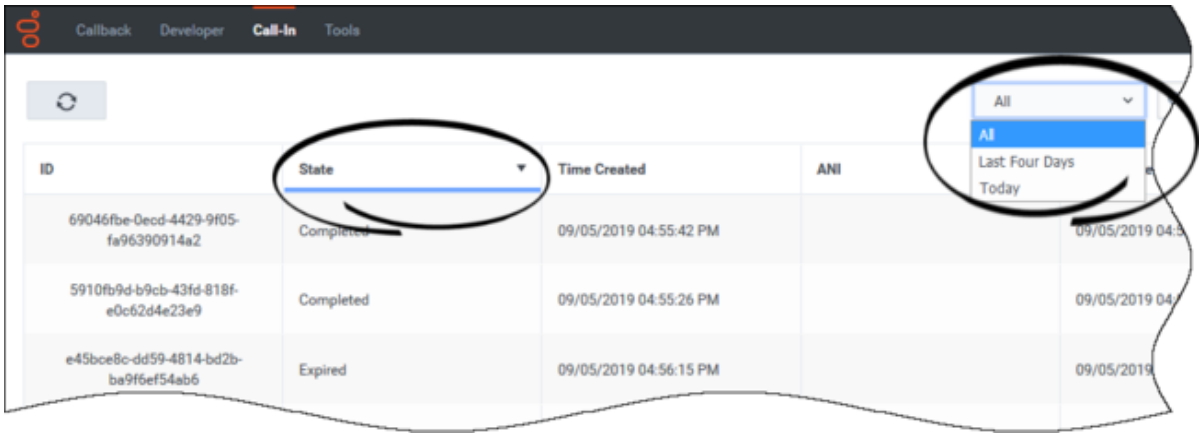
The **Call-In** tab lists up to 5000 Click-To-Call-In records. Once you have more than 5000 Click-To-Call-In records, the oldest records are dropped from the list in the UI.

One page on the **Call-In** tab displays up to 500 records. Once you have more than 500 records, you can use the pagination tools at the bottom of the page to navigate through the records.

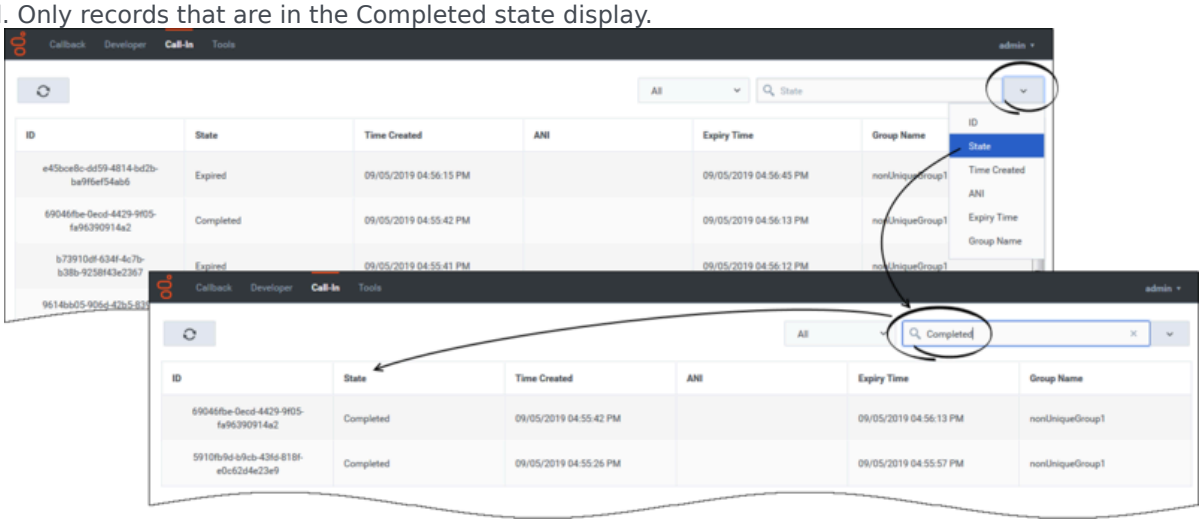
Filtering and sorting the records list

To help you find specific records more easily, the **Call-In** tab offers the following filtering tools:

- Filter the list using the predefined time-range filters available at the top of the page.
- Click a column heading to sort the records in ascending or descending order.



- Search based on a specific record attribute. In this example, we want to find all of the records that are in the Completed state. We select `State` from the drop-down menu and enter `Completed` in the **Search** field. Only records that are in the Completed state display.

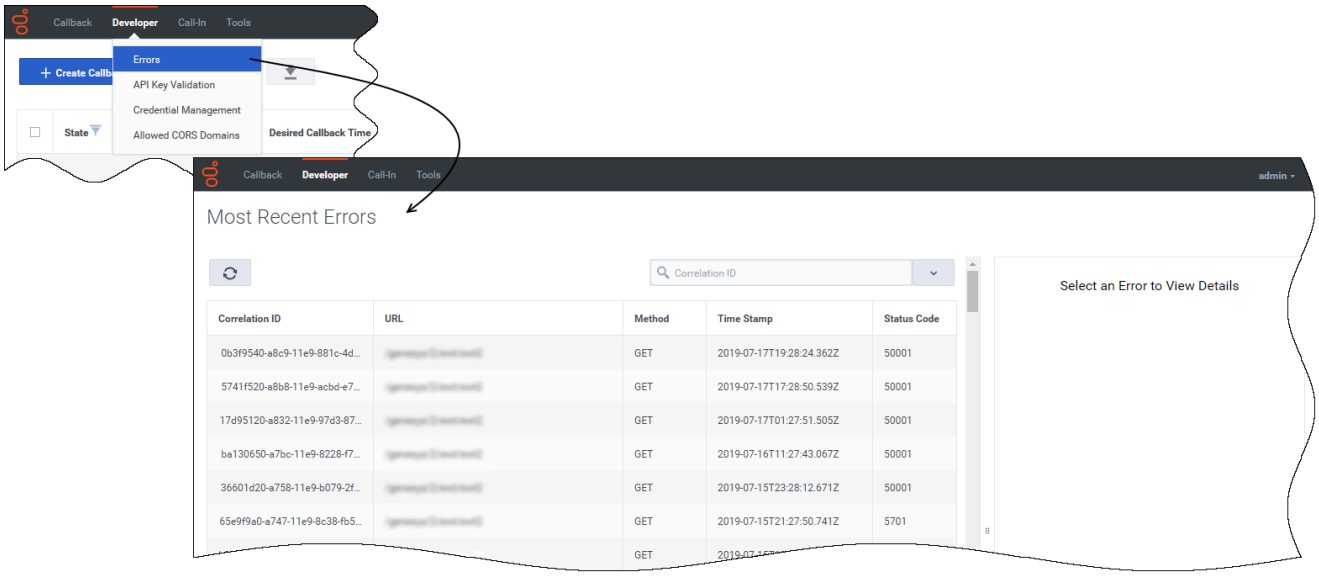


Using the Developer Tab

> **Important**
>
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Troubleshooting and validating functionality.

> **Important**
>
> The **Developer** tab requires the **Callback Administrator** or **Callback Developer** role.

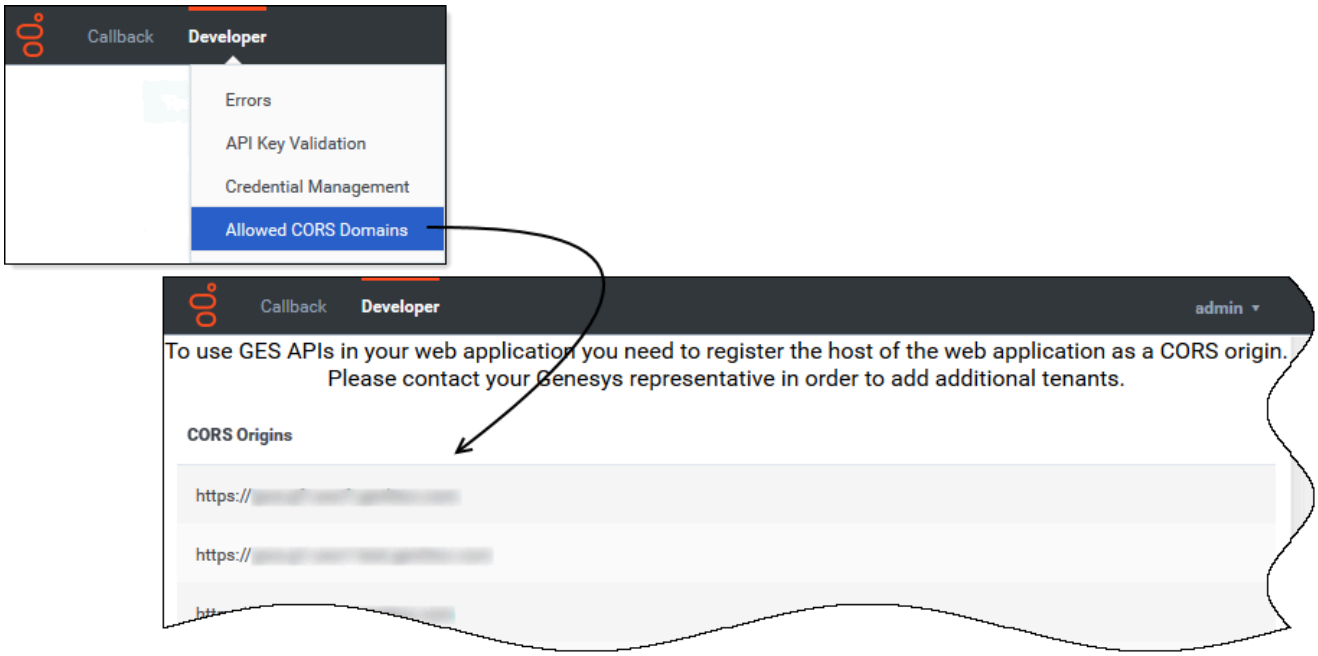
Displaying Errors



The **Developer** > **Errors** page displays the list of errors related to your Callback API queries. If you select one of the listed errors, the tab displays more information in the right-hand panel. When an API call returns an error, the response includes a correlation ID. Use this correlation ID as input into the Search bar to find more information about a specific API call.

This should help you to solve API issues when developing an application using the REST APIs.

Viewing CORS Domains

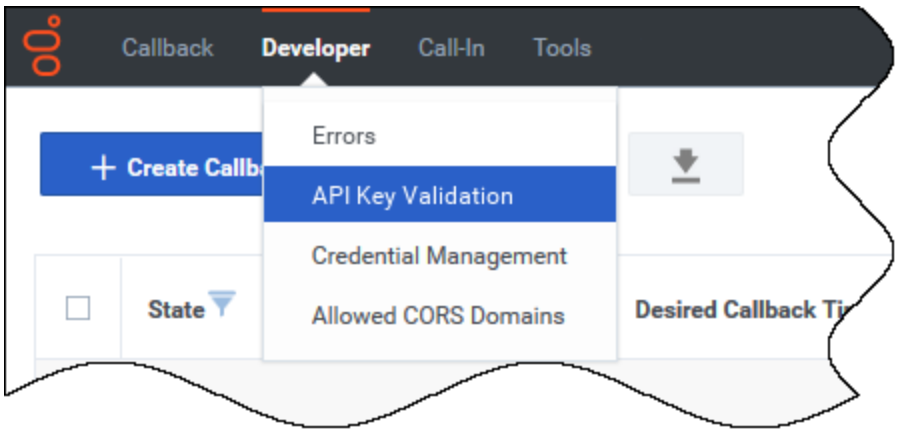


To access Genesys Engagement Services (GES) APIs in a Web application, your Web application's host must be registered as a "safe" domain or origin. The **Developer** tab includes an **Allowed CORS Domains** page, which you can access from the **Developer** tab menu. On the **Allowed CORS Domains** page, you can view the list of domains or origins that have been provisioned for your use. Cross-origin resource sharing (CORS) requests to the origins in the list are allowed.

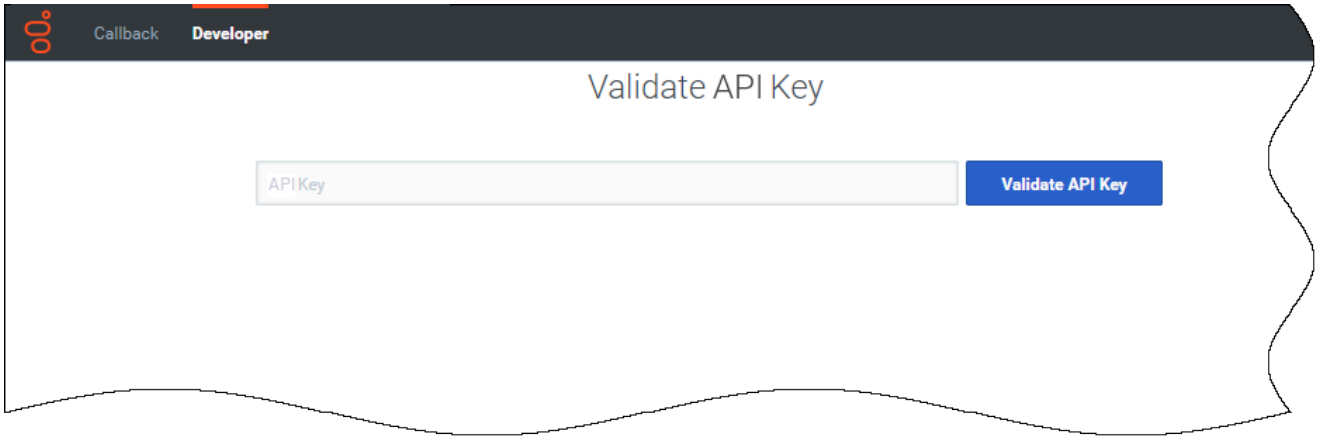
The list of origins on the **Allowed CORS Domains** page is read-only. To add cross-origin resources to the list, you must contact your Genesys representative.

> ### Tip
> If you do not currently have the **Allowed CORS Domains** page in your Callback UI, contact Genesys Customer Care to request it.

Validating your API Key



If your permissions allow it, the **Developer** tab menu includes **API Key Validation**.



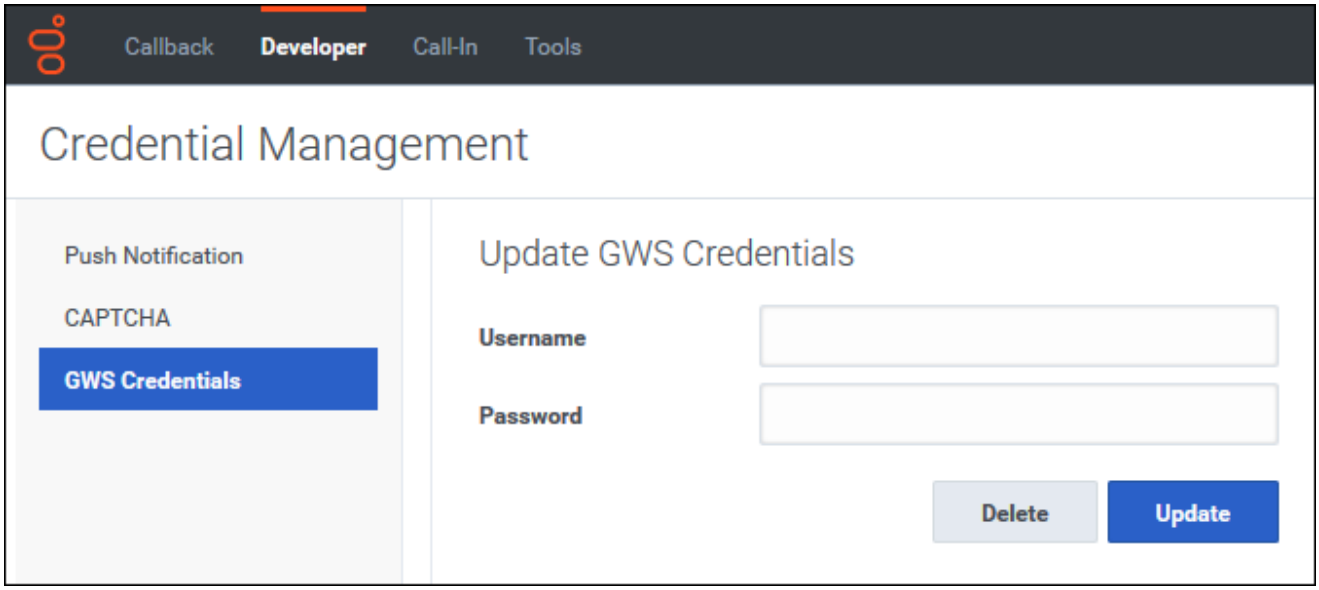
Copy/paste your API Key into the textbox and click the **Validate API Key** button.

- If your API Key is incorrect, the UI displays **Invalid API Key**.
- If your API Key is valid, the UI displays **Valid API Key**.

Using Credential Management

The **Developer** tab menu includes **Credential Management** if you have sufficient permissions to view this part of the interface. Use **Credential Management** to register, manage, and test credentials for features within your Callback environment and features such as push notifications that

work within other supported delivery networks.



Callback Administrators and Developers have access to GWS statistics information through Callback's Statistics API, but you must register your credentials on the **Credential Management** > **GWS Credentials** tab before you can retrieve the statistics.

For additional information about the Callback APIs, see Genesys Engage REST APIs and Tutorials for Callback.

On the **Credential Management** page, Callback Administrators and Developers also manage credentials for the following features:

- Push notification
- CAPTCHA

Provisioning Push Notifications

> **Important**
>
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Provisioning Push Notifications.

Push Notifications are messages that are proactively sent from an app to a known, permitted client (either a mobile device or a web browser). Modern smartphone users will be well acquainted with push notification behavior within their favorite apps.

The **Push Notification** page in Callback helps you to integrate Push Notification support into your apps. In the Callback UI, you implement Push Notifications on the **Developer** > **Credential Management** > **Push Notification** tab. Only users who are members of the Callback Developer and/or Administrator Roles can access this tab. See Controlling User Access for more information about controlling user access to Callback tabs and features. As long as you have sufficient permissions to view the **Push Notification** page, then you have full access to use all of its features.

In terms of Callback features, you might want to include a Click To Call feature in your mobile app so customers can easily connect with you when they need an agent's assistance. As part of that feature's implementation, the system can communicate with the customer through Push Notifications; for example, in cases where the system needs to let the customer know that an agent is available to take their call or to offer follow-up actions such as the ability to confirm or cancel the call. For information about provisioning the Click-To-Call-In feature in Callback, see Provisioning the Click-to-Call-In Scenario.

Callback Push Notifications

If Push parameters are provided with a Callback Create API request on a standard callback virtual queue, the system can send Push Notifications for the following events:

- CALLBACK_UPCOMING—When the callback's Estimated Wait Time (EWT) is below the configured threshold for the queue (configured under the **Push Notification Threshold (minutes)** setting in the CALLBACK_SETTINGS data table), the CALLBACK_UPCOMING event sends a Push Notification to let the consumer know that they can expect a phone call soon. This event is sent only once. In other words, if the EWT increases above the threshold after the notification is sent and then drops below the threshold again, the CALLBACK_UPCOMING Push Notification is not sent again.
  The system might not send a Push Notification for the CALLBACK_UPCOMING event in the following situations:
  - When the EWT is already low (below the configured threshold) when the Callback Create API request arrives.
  - When the system determines that there is not enough time to complete the callback because the callback will be purged or the end of the business will occur before the callback is at the top of the queue.

- CALLBACK_READY—Used only with the Click-To-Call-In (Delayed) scenario. When a consumer's callback request reaches the front of the queue, the CALLBACK_READY event sends a Push Notification to alert the consumer. If the consumer accepts the callback, the mobile app sends a Call-In Create API request. The system immediately sends another Push Notification with the information that the user needs to call in to the contact center.
  Note that the callback "acceptance" is a function of your mobile app's configuration. For example, your mobile app developer might configure your mobile app to automatically make the call-in request as soon as it receives the CALLBACK_READY notification or the app could require the consumer to manually make the call-in request by clicking a button to acknowledge receipt of the notification.
- CALLBACK_PURGED—This Push Notification is sent if the callback has been purged from the system and the callback was not rescheduled.
- CALLBACK_ATTEMPT_FAILED—This event occurs when the system makes the outbound call for a callback, but the call fails to connect to the consumer for some reason. Another outbound call can be attempted later. In the CALLBACK_SETTINGS data table, you can configure the number of times that the system will make outbound calls for each Callback request before declaring failure. This Push Notification is not used with the Click-To-Call-In (Delayed) scenario because there are no outbound calls associated with that scenario.
- CALLBACK_FINAL_ATTEMPT_FAILED—This event occurs when the system makes the final outbound call associated with a Callback request and the call fails to connect to the consumer. In the CALLBACK_SETTINGS data table, you can configure the number of times that the system will make outbound calls for each Callback request before declaring failure.This Push Notification is not used with the Click-To-Call-In (Delayed) scenario because there are no outbound calls associated with that scenario.

Firebase Cloud Messaging Push Notifications

Genesys Callback currently supports Google Firebase Cloud Messaging (FCM) Push Notifications. Using the Callback UI, you can integrate Firebase Cloud Messaging into your web or mobile applications. For detailed information about FCM, see the Google Firebase Cloud Messaging documentation. You use the **Push Notification** page in the **Developer** tab to register your Firebase credentials with Callback and to test that the credentials are valid.

Before You Start

Before you start working with the **Push Notification** page on the Callback **Developer** tab, ensure that you have a Google Firebase account. You require the following Firebase objects to register with Callback and to implement and test the Push Notification functionality:

- The Firebase project name that is associated with the app that you are registering with Callback.
- The client email address that is associated with the project. See Client Email for more information.
- The private key.
- The sender ID that is associated with the project.

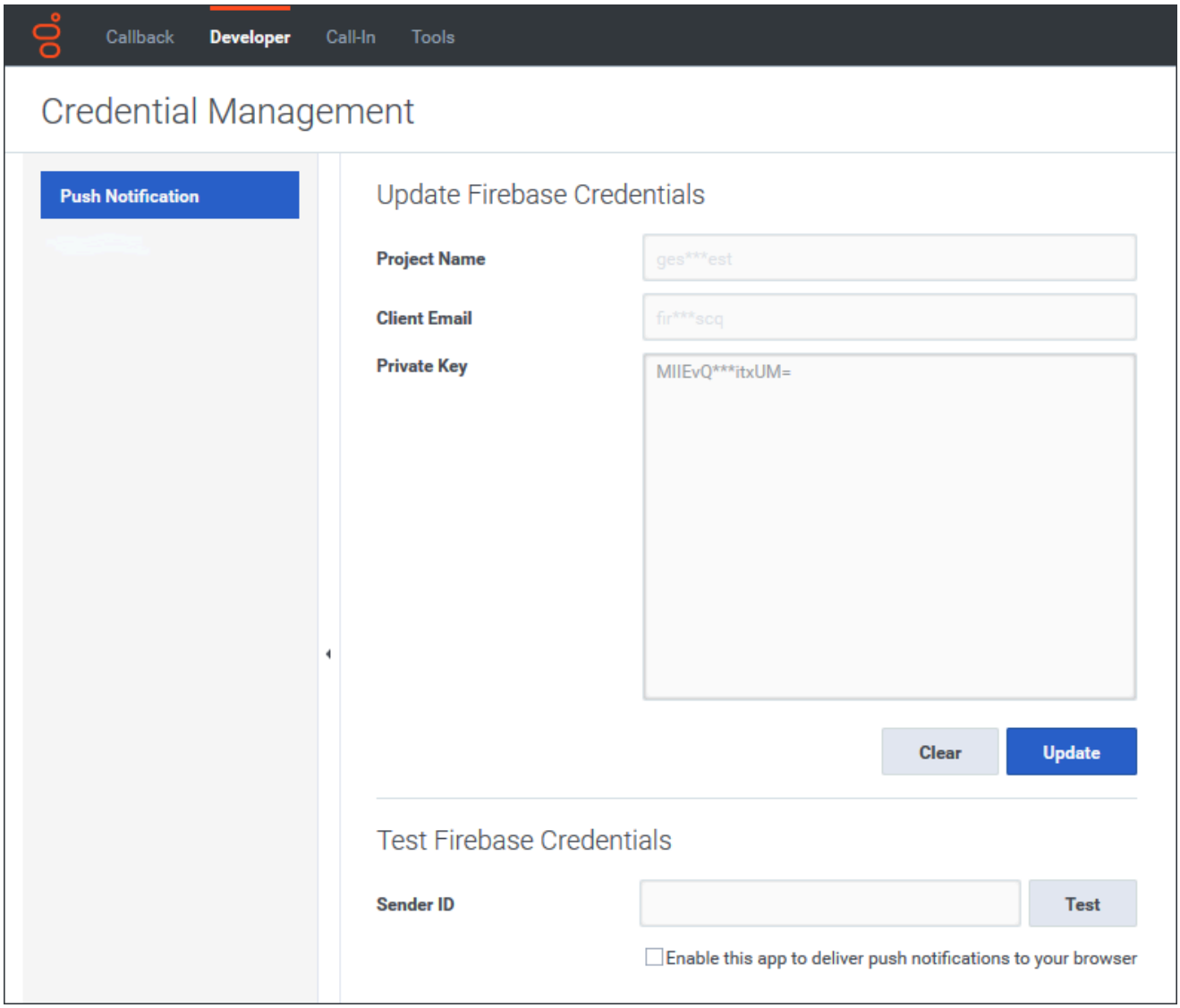
Using the Push Notification Page

You can perform the following activities on the **Push Notification** page:

- Provision and update your Firebase credentials in Callback.
- Determine if your Firebase credentials for an app are already configured in Callback.
- Perform a test to verify that the Firebase credentials that you entered are correct and configured in Callback and, by extension, that the Push Notifications are working as intended.

To configure the Firebase credentials in Callback, you enter and update those credentials on the **Update Firebase Credentials** pane of the **Push Notification** page. You use the **Test Firebase Credentials** pane to verify that web Push Notifications are working in your environment. If the Firebase credentials that you provisioned on the **Update Firebase Credentials** pane are correct and valid, then – when you run a test using your Firebase Sender ID – the **Push Notification** page returns a message that your Firebase credentials were successfully configured.

Register your Firebase Credentials with Callback



To implement the Firebase Cloud Messaging functionality, you must provide the following information in the **Update Firebase Credentials** pane of the **Push Notification** page:

- Project name
- Client email address
- Private key

After you click the **Update** button, a message appears in the bottom right-hand corner of the page to let you know that your credentials were successfully updated or that there was an issue.

Genesys recommends that you also test your Firebase credentials to make sure that the credentials have been successfully configured in Callback. To run the test, you require the Firebase Sender ID

associated with the project that is identified on the **Update Firebase Credentials** pane.

Project Name

This is a user-friendly name that you configure for the Firebase project. For example, `fcm-client-11959`. Once configured, the Firebase project name (or Project ID) can be found in the **Project settings** > **General** tab on the Firebase console. On the **Push Notification** page, you must enter the Firebase project name *exactly* as it is configured in Firebase.

Client Email

This is an email address that is associated with the Firebase project. On the Firebase console, this is the service account, however, in the file that contains the private key, the account is called the client email address. For example, a client email address might be `firebase-adminsdk-rkbys@fcm-client-11959.iam.gserviceaccount.com`. Once configured, you can find the service or client email account on the **Project settings** > **Service accounts** tab on the Firebase console or within the file that contains your private key.

Private Key

The private key is used by an application to access the Firebase API. You generate a file that contains the private key for your app on the **Project settings** > **Service accounts** tab of the Firebase console. Generating the file downloads it to your local machine. The private key is a very long character string and is only available in the downloaded file.

If, for any reason, you need to generate a new private key for an application that was previously configured in Callback, then you can do so on the Firebase console, but then you must update the credentials in Callback using the **Update Firebase Credentials** pane on the **Push Notification** page. Remember to test the new credentials as well.

Test Your Credentials Configuration

If you are registering new Firebase credentials with Callback or updating credentials, then test the credentials to verify that Firebase and Callback can validate the configuration. You require your Firebase Sender ID to test your credentials. You can find the Firebase Sender ID in the **Project settings** > **Cloud Messaging** tab on the Firebase console. You must check the **Enable this app to deliver push notifications to your browser** box to run the test. The **Push Notification** page returns either a "successful configuration" message or a "failure" message.

If you receive a failure message, use the following process to recover and try again:

1. Check that the Sender ID that you have entered on the **Update Firebase Credentials** pane matches the Sender ID on the Firebase console for the app that you are registering with Callback. Run the test again.
2. If the Sender ID matches and you do not receive a Push Notification, then refresh the browser and run the test again.
3. If you don't receive a Push Notification, then navigate to your browser's Settings and ensure that the browser allows notifications from your Genesys Engagement Service (GES)/Callback host. If you need

help to find the setting, check the documentation for your browser. If your browser is blocking notifications, then alter the setting to enable notifications. Run the test again.

4. If you don't receive a Push Notification, then refresh your browser again, check all of your FCM credentials to ensure that they are entered correctly, and run the test again.

Displaying the Firebase Credentials on the Push Notification Page

After you have successfully configured the Firebase credentials in the Callback UI, and you refresh the **Push Notification** page or leave the page and return to it later, you will see only a few characters for each entry (project name, client email address, and the private key). This is for security purposes. Once configured, it is impossible to see the complete entries again on the **Push Notification** page. The Project name, Client email address, and Private key sections on this page explain where you can find the full credentials again, if required.

Provisioning CAPTCHA

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Provisioning CAPTCHA.

A *CAPTCHA* is used as a security measure. Its purpose is to determine if an interaction on a website is made by a human or a bot. If you have a website that includes a callback feature, then you can provision and use a CAPTCHA widget to verify that a human is requesting a callback. This protects against being flooded with callback requests from a fraudulent source.

Currently, Callback supports integration with only one CAPTCHA widget: Google reCAPTCHA. This page describes how to enable the use of reCAPTCHA with Callback.

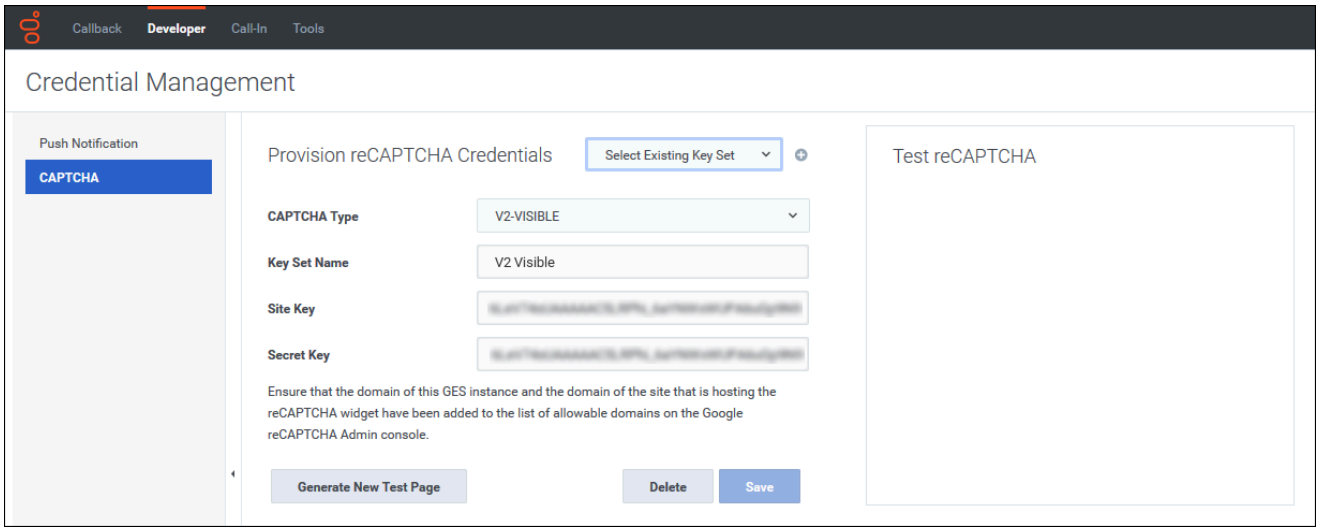
Important Information About Enabling reCAPTCHA on Callback Requests

Integrating reCAPTCHA into your Callback API operations means that reCAPTCHA validation will be required with *all* of the Callback API operations. For example, if you have both a website and a mobile app, you cannot implement reCAPTCHA on callback requests for only your website – it must be built into your mobile app as well. Also, this implementation of reCAPTCHA is not integrated with the Genesys Widget for Callback. If you are using the Genesys Widget for Callback, then Genesys recommends that you avoid enabling this reCAPTCHA capability.

Provisioning Google reCAPTCHA for Callback

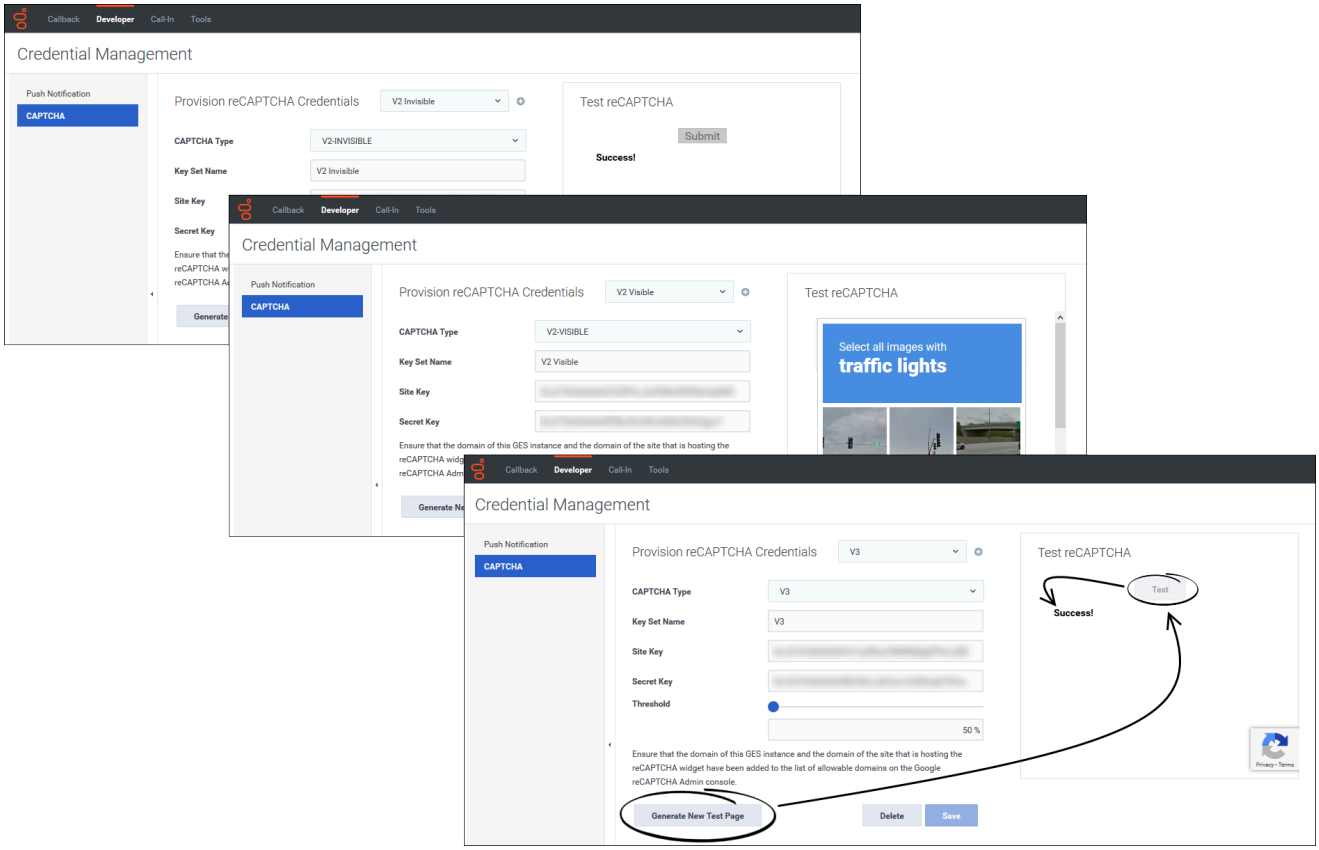
Callback currently supports Google reCAPTCHA. To provision reCAPTCHA for Callback, you must have a Google account. If you have not yet done so, go to the reCAPTCHA Admin console to register your site and to specify the reCAPTCHA version that you want to use. Callback supports reCAPTCHA v3, reCAPTCHA v2 Checkbox (Visible), and reCAPTCHA v2 (Invisible). Review the Google reCAPTCHA documentation; it describes the three versions of reCAPTCHA and contains information to help you to register your site on the Admin console.

On the Google reCAPTCHA Admin console, it is important that you add the domain of your Callback instance and the domain of the site that is hosting the reCAPTCHA widget to the list of allowable domains.



After you have registered your site on the Google reCAPTCHA Admin console, you must register your reCAPTCHA credentials with Callback on the **Developer** > **Credential Management** > **CAPTCHA** tab. The CAPTCHA type, site key, and secret key that you specify must match the settings for the reCAPTCHA widget that you provisioned in the Google Admin console.

Specify a name (the **Key Set Name**) to identify the reCAPTCHA widget and its associated credentials. It is helpful to use a descriptive name. For example, you might enter a name that reflects the type of reCAPTCHA widget that you're using (`V2 Visible`). The **Key Set Name** is used only in Callback; it is not specified or used in the Google reCAPTCHA Admin console.



After you save your reCAPTCHA credentials on the **CAPTCHA** page in Callback, be sure to test them. When you click **Generate New Test Page**, a testing widget opens on the page. The widget will vary slightly depending on the type of reCAPTCHA that you're provisioning. You might have to check a box or otherwise interact with the widget, but you will be presented with a **Submit** or **Test** button at some point. If your credentials were correctly configured and you pass the reCAPTCHA validation, then the widget returns a `Success!` message. If it fails, check your credentials or – if you are using the V3-type reCAPTCHA – try adjusting the threshold, and then try again.

Using Callback Tools

> **Important**
>
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Monitor callback queues.

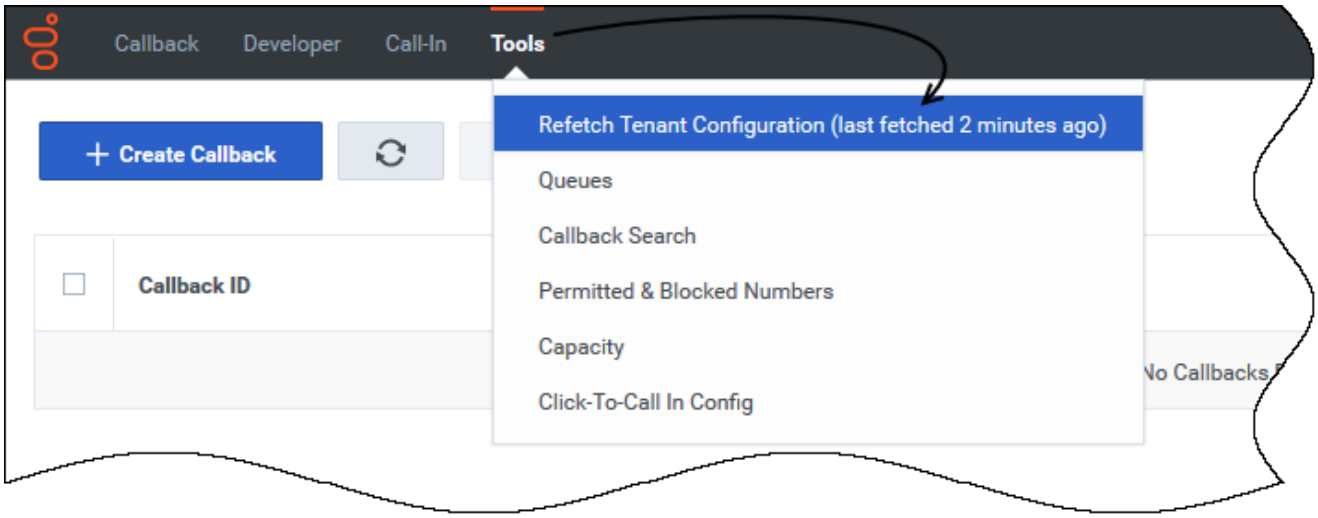
The Callback application includes a **Tools** tab, which offers additional views and tools to assist with callback management, configuration, and troubleshooting. You must be a member of the Administrator, Supervisor, or Developer Role to access the **Tools** tab.

The **Tools** tab includes the following menu options:

- You can look up when your tenant's configuration data was last refreshed. You can also force a data refresh.
- You can view the complete list of configured queues. The list includes detailed information about each queue.
- You can search for a specific interaction within the callback records.
- You can configure rules that will be assigned to queues. For example, you might want to limit the countries to which you allow callbacks or limit the number of scheduled callbacks that you allow within a certain time interval.

This page gives you an overview of the **Tools** tab menu options. Some of the work that you can perform on pages that are accessible from the **Tools** tab requires more explanation than an overview can provide. In those cases, links to pages that contain detailed information are included in the descriptions.

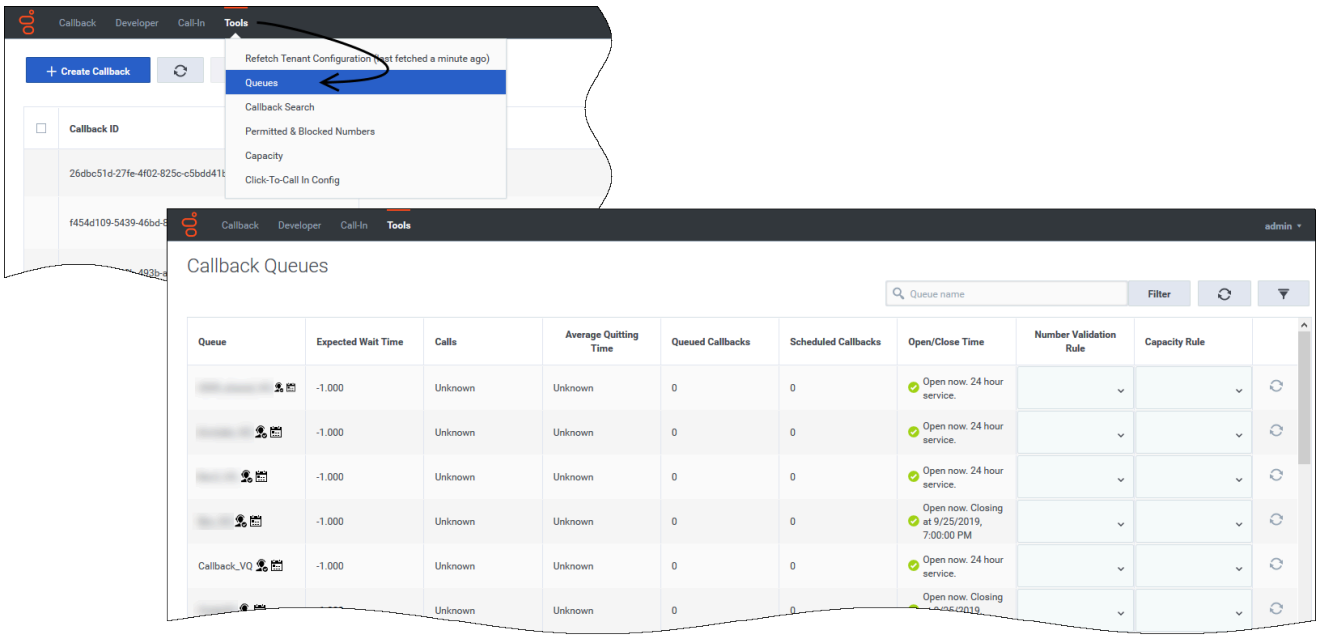
Refreshing your Tenant Configuration data



By default, the system refreshes the Tenant configuration every 30 minutes. The **Tools** > **Refetch Tenant Configuration** menu option shows you how long it has been since the data was last updated.

When you click **Refetch Tenant Configuration**, the system refreshes tenant configuration data throughout the Callback application. For example, you might want to force a refresh of data to immediately reflect configuration changes in features that are affected by such settings.

Viewing the list of configured queues

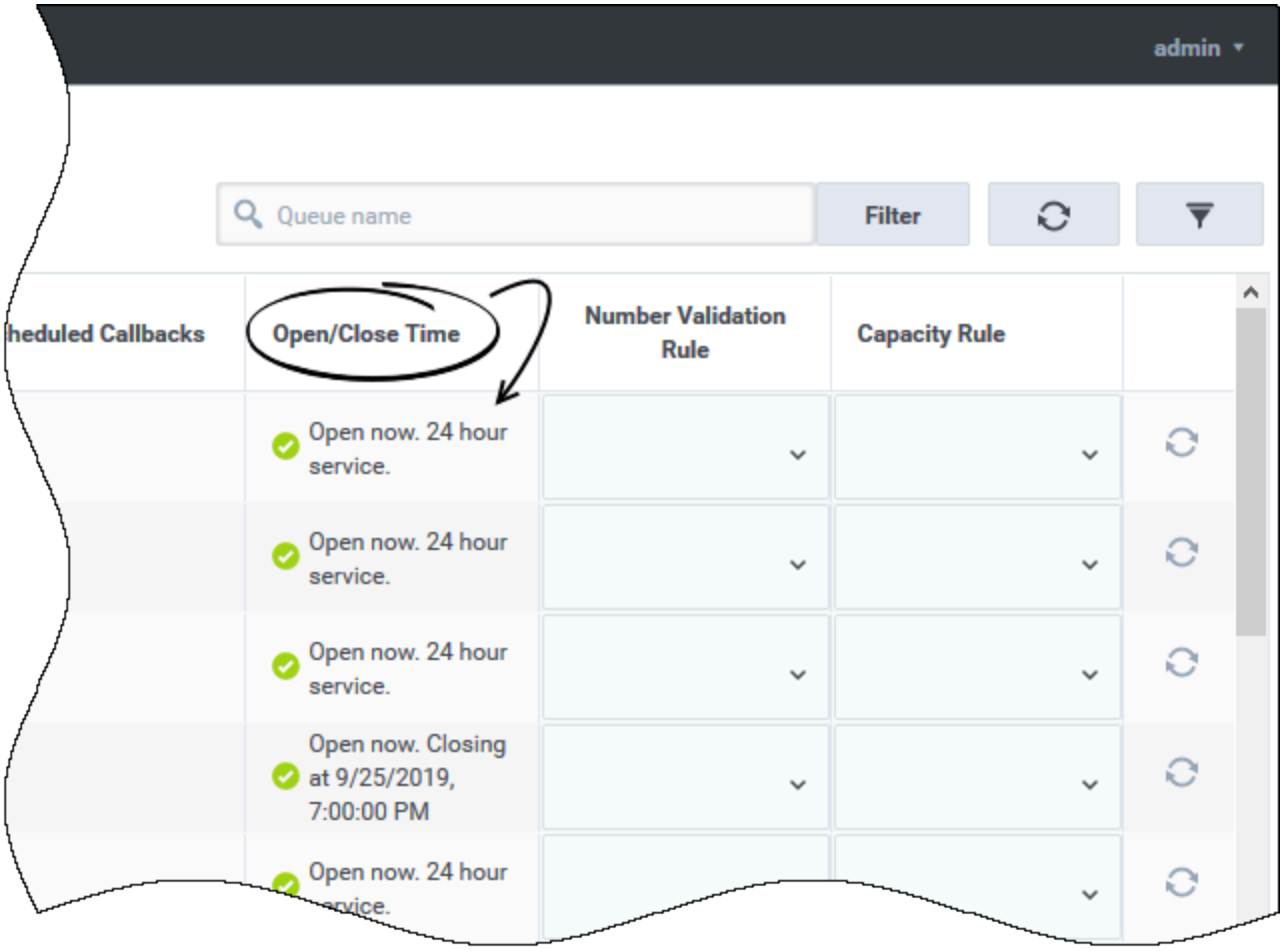


The **Tools** > **Queues** menu option opens a page that shows you the list of configured queues. The page includes information for each queue such as Estimated Wait Time and the number of callbacks in queue. Icons beside each queue's name indicate whether the queue accepts Immediate ( ) or Scheduled ( ) callbacks or both.

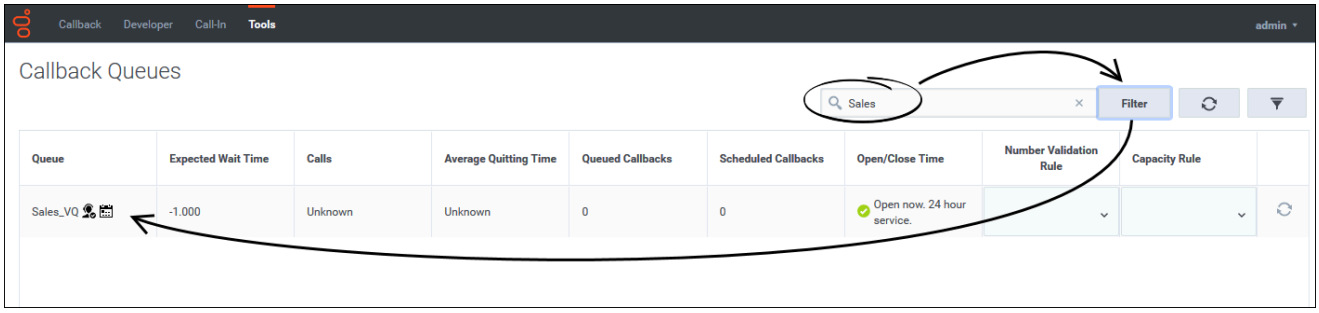
Queues are displayed in pages of 50 items. If Unknown values are displayed for a queue, it means that the queue cannot return data for some reason; for example, the queue might be inactive.

> ## Tip
>
> If you provision a virtual queue for callbacks in Designer while you are logged in to the Callback UI, be aware that the new queue might not display on the **Callback Queues** page until you log out and log in again because Callback fetches virtual queue permissions data only at login. In other words, Callback determines which virtual queues you have access to when you are logging in, and only then.

The **Callback Queues** page pulls data when you first open the page, but does not continually check for new data. To update the data for a specific queue, click **Refresh** for that queue. When you click **Refresh** at the top of the page, it refreshes the data for every queue.



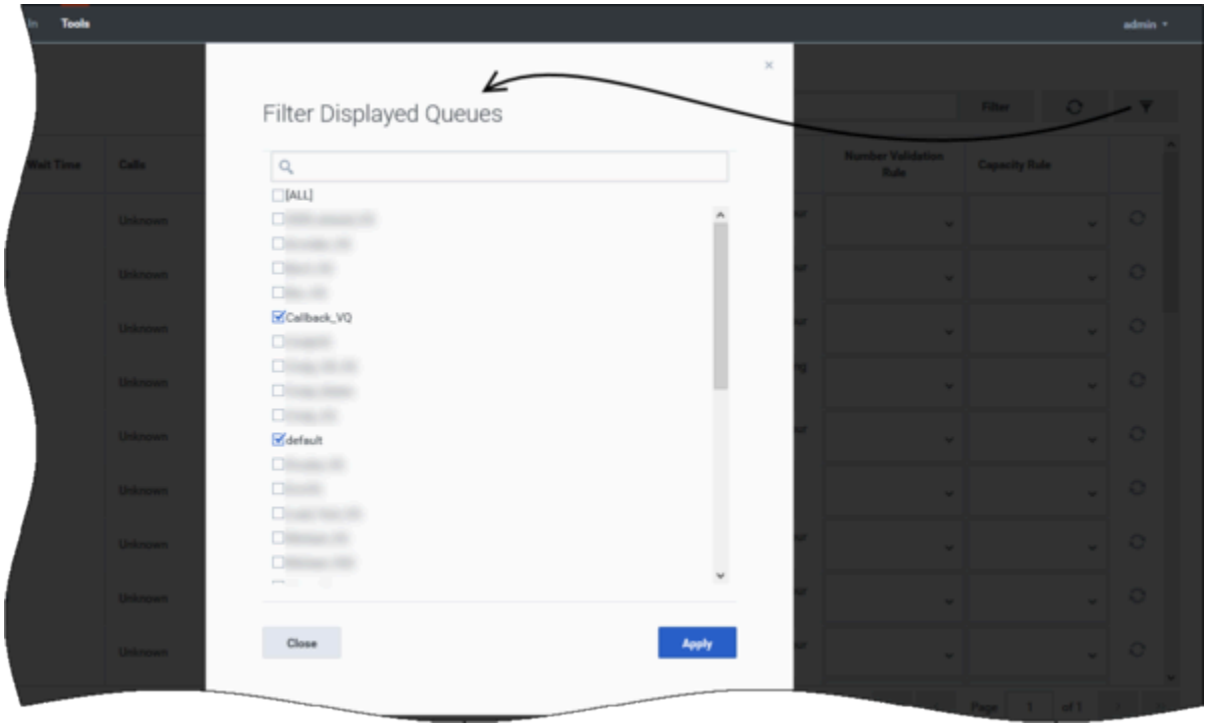
The **Callback Queues** page provides information about Open and Close times for each queue. The time displayed is based on the time zone configured for the device that you are using to access the Callback application.



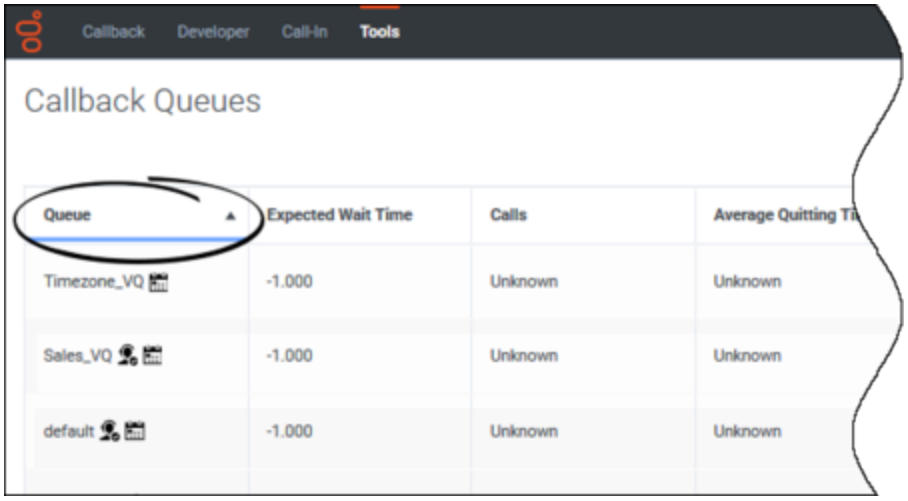
On the **Callback Queues** page, it is sometimes helpful to reduce the list to only one queue or a few queues for which you want to find information.

To filter the list, enter a queue name or part of a queue name in the field at the top of the page and click **Filter**. Using this method, you can filter the list of queues using only one criterion at a time. For example, let's say that you enter `Sales` in the field and click **Filter**. Any queue that includes `Sales` in its name will display. If you clear `Sales` from the field and filter on the word `Service`, then all queues that have `Service` in their name display, replacing the list of `Sales`-related queues.

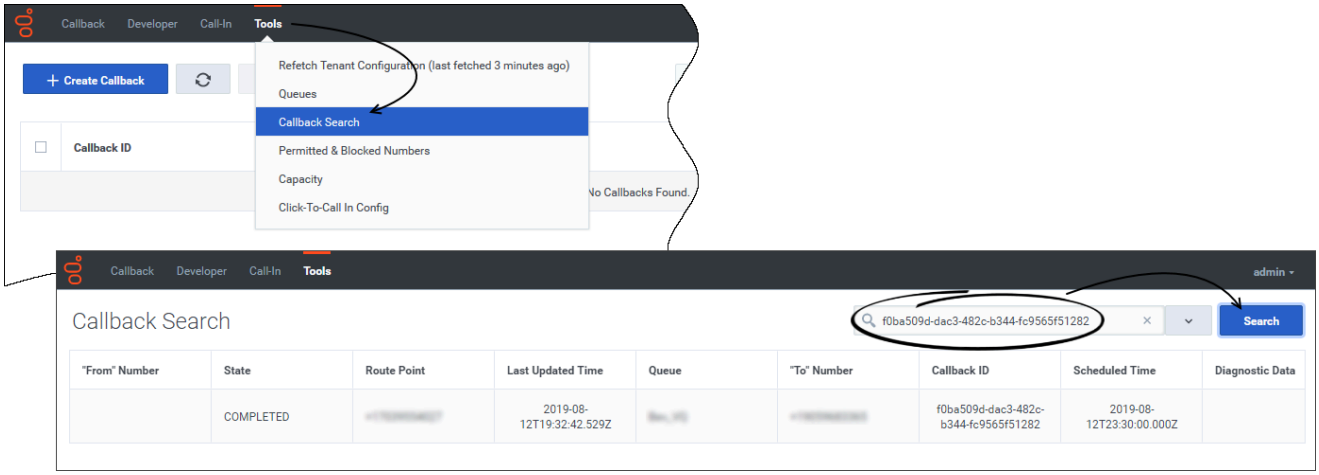
Alternatively, when the specific queues that you want to view on the **Callback Queues** page have dissimilar names, use the **Filter Displayed Queues** dialog. From the list of virtual queues in the dialog, select those that you want to display. If you have a lot of queues, use the filter at the top of the dialog to help you find the ones that you are looking for.



In addition to filtering, you can also click the **Queue** column heading to sort data based on that column.



Searching for a specific interaction



The **Tools** > **Callback Search** menu option lets you search for a specific callback interaction. To search for an interaction, enter either the callback ID or the customer's phone number and click **Search**.

Configuring rules for queues

The **Tools** tab includes menu options to configure the following types of rules, which you can then assign to queues:

- Patterns: Country and number validation rules. These rules specify the countries to which you allow

callbacks and can include a list of "blacklist" numbers (blocked number patterns) within those countries.

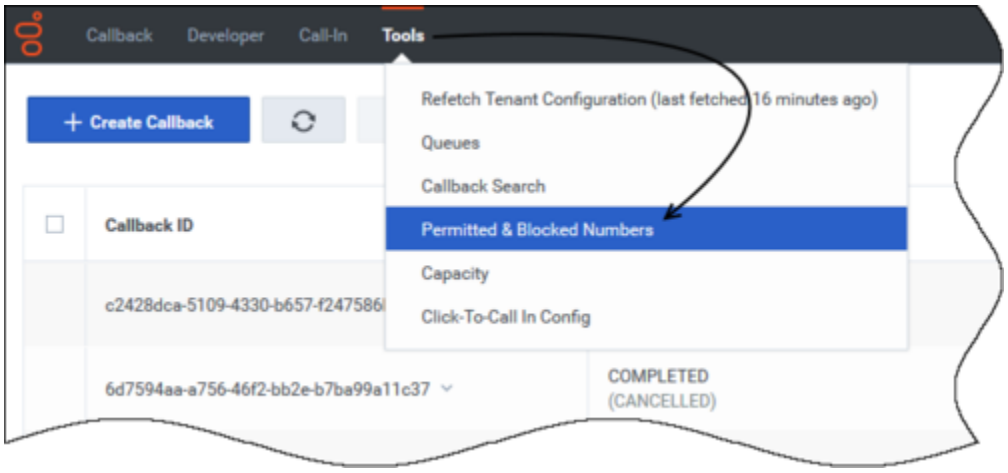
- Capacity: Callback capacity rules. These rules specify the maximum number of scheduled callbacks that are allowed within each time slot for a week.

Configuring Country and Number Validation Rules

> **Important**
>
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Configuring Country and Number Validation rules.

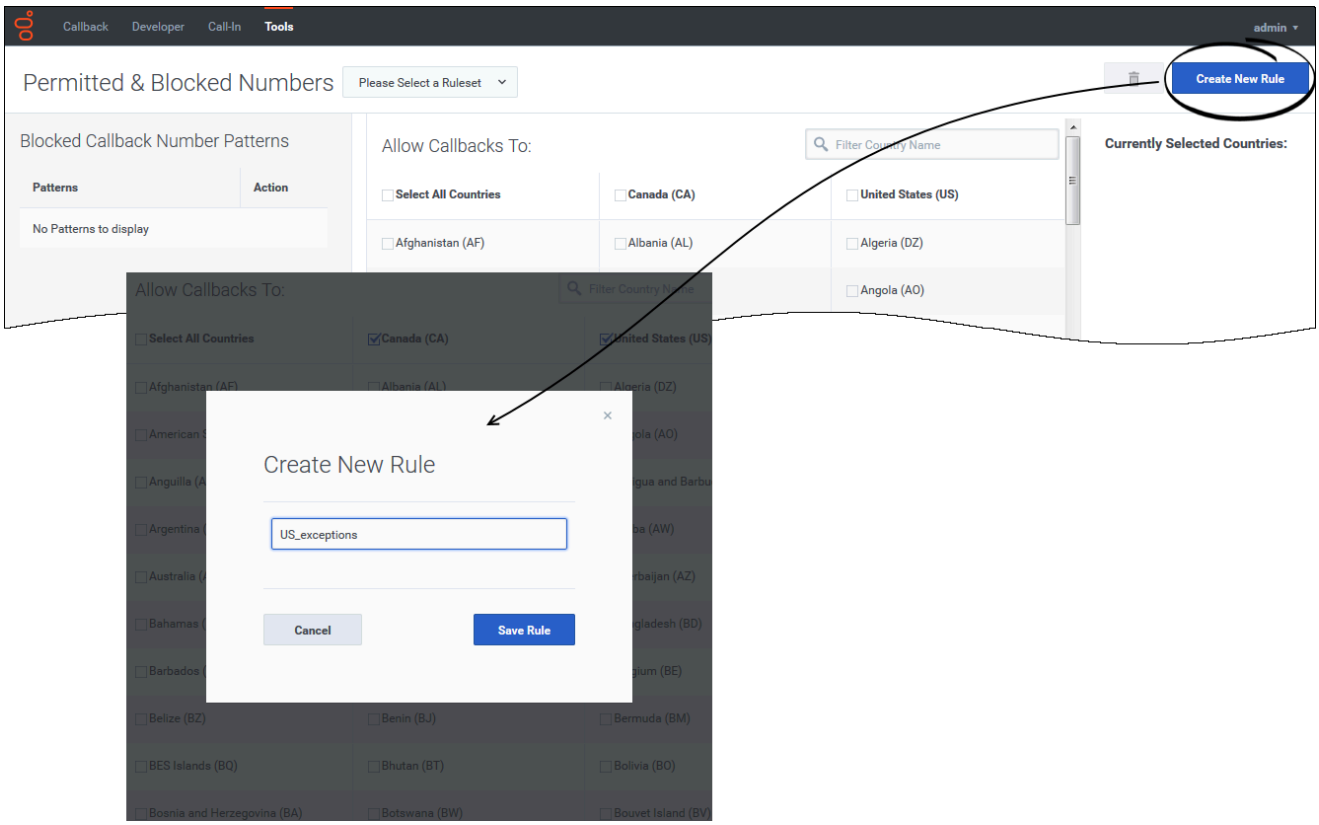
> **Important**
>
> To create, edit, or delete country and number validation rules, you must be a member of the **Callback Administrator**, **Callback Developer**, or **Callback Supervisor** role.

To avoid initiating callbacks to certain countries or to numbers that you know to be fraudulent, you can now configure country and number validation rules on the **Tools** > **Permitted & Blocked Numbers** page, and then assign those rules to queues on the **Callback Queues** page. You can create a rule to specify the list of countries to which you permit callbacks for specific queues. You can also optionally enhance a rule by defining simple or complex (regular expression) numeric patterns that will block callback attempts to matching numbers.

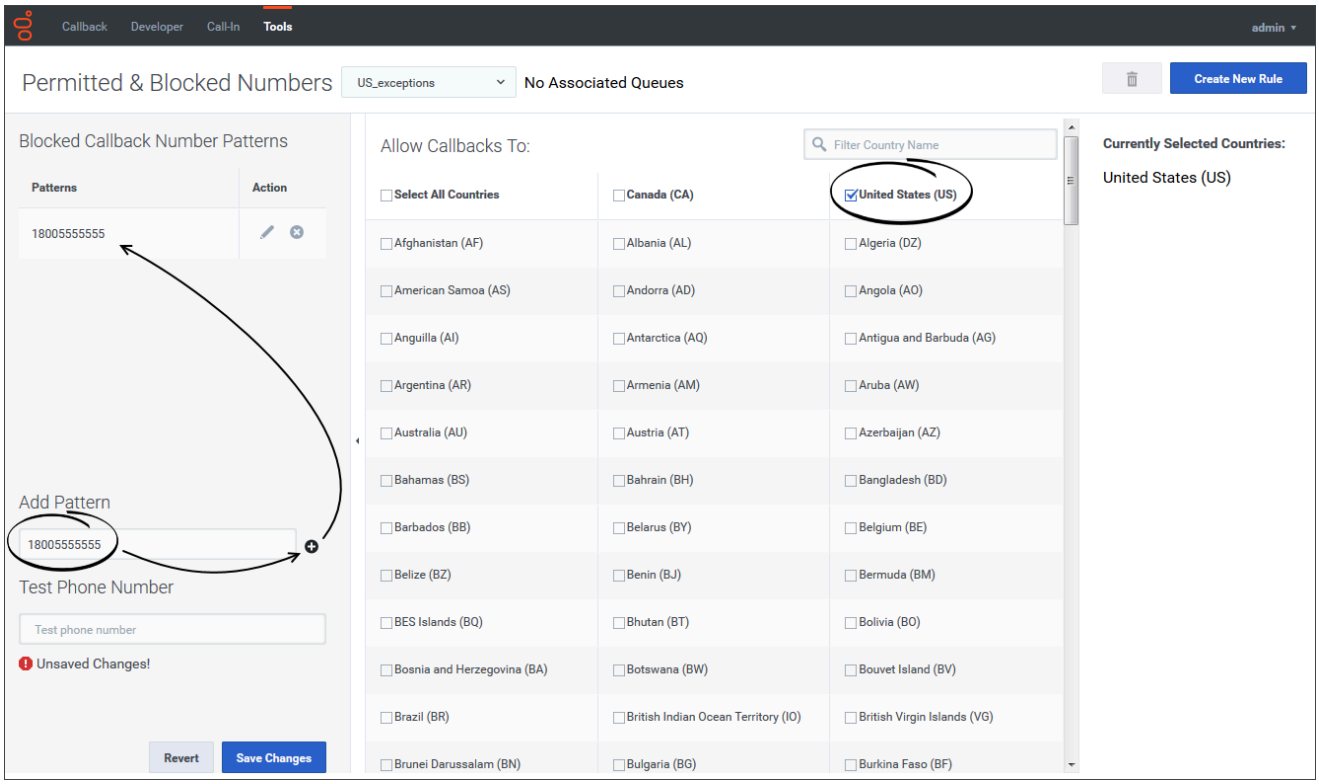


Once a rule is configured, you can assign it to as many queues as necessary. It is important to note, though, that you can assign only one rule to each queue. Before creating a rule, consider to which queue or queues you will assign it and make sure that the rule incorporates the countries to which you permit callbacks for that queue as well as any numeric patterns in any of those countries that you want to block. Rules are checked before the system initiates a callback.

Configuring Country and Number Validation Rules



The first step in creating a new country and number validation rule is to give the rule a name. Later, you will need to pick the rule from a list when you assign it to a queue, so make sure the name is descriptive enough for you to understand what it allows and denies.

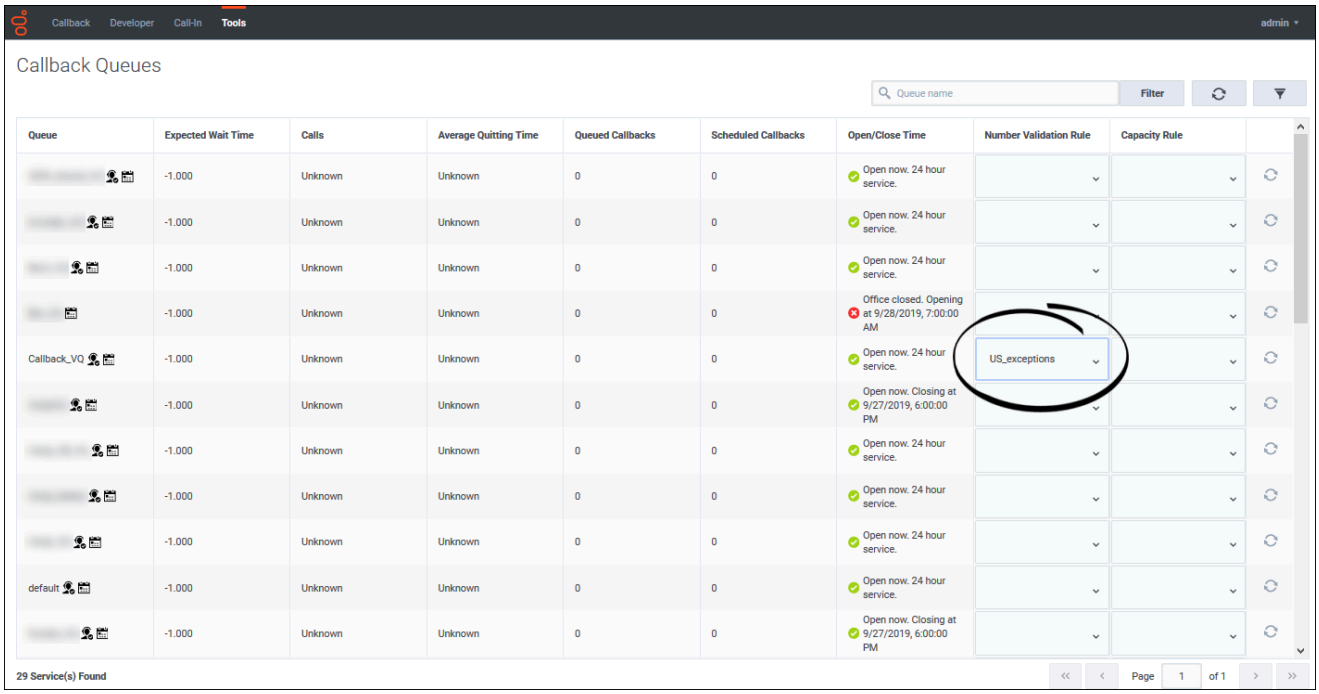


When creating a rule, you must select at least one country on the **Permitted & Blocked Numbers** page. You can enter blocked number patterns only after you make country selections. Adding a country to the rule means that you allow callbacks to that country. For countries that are not added to a rule, no callback attempt will be initiated on the queue or queues to which the rule is applied.

After you have selected the countries to which you allow callbacks, enter any patterns that you want to block. Patterns must be entered as regular expressions. For example, if you have a list of suspicious phone numbers for the selected countries and you do not want the system to make callback attempts to those numbers, then add those to the rule. The figures on this page show the addition of specific numbers, but you could enter `^1234[0-9]+`, which would match any phone number that starts with 1234. While a rule must contain at least one country, the addition of blocked number patterns is optional.

To add a blocked number pattern, simply enter it in the field and click "plus" (⊕). You must enter patterns individually; that is, enter a pattern and add it to the rule before entering another pattern.

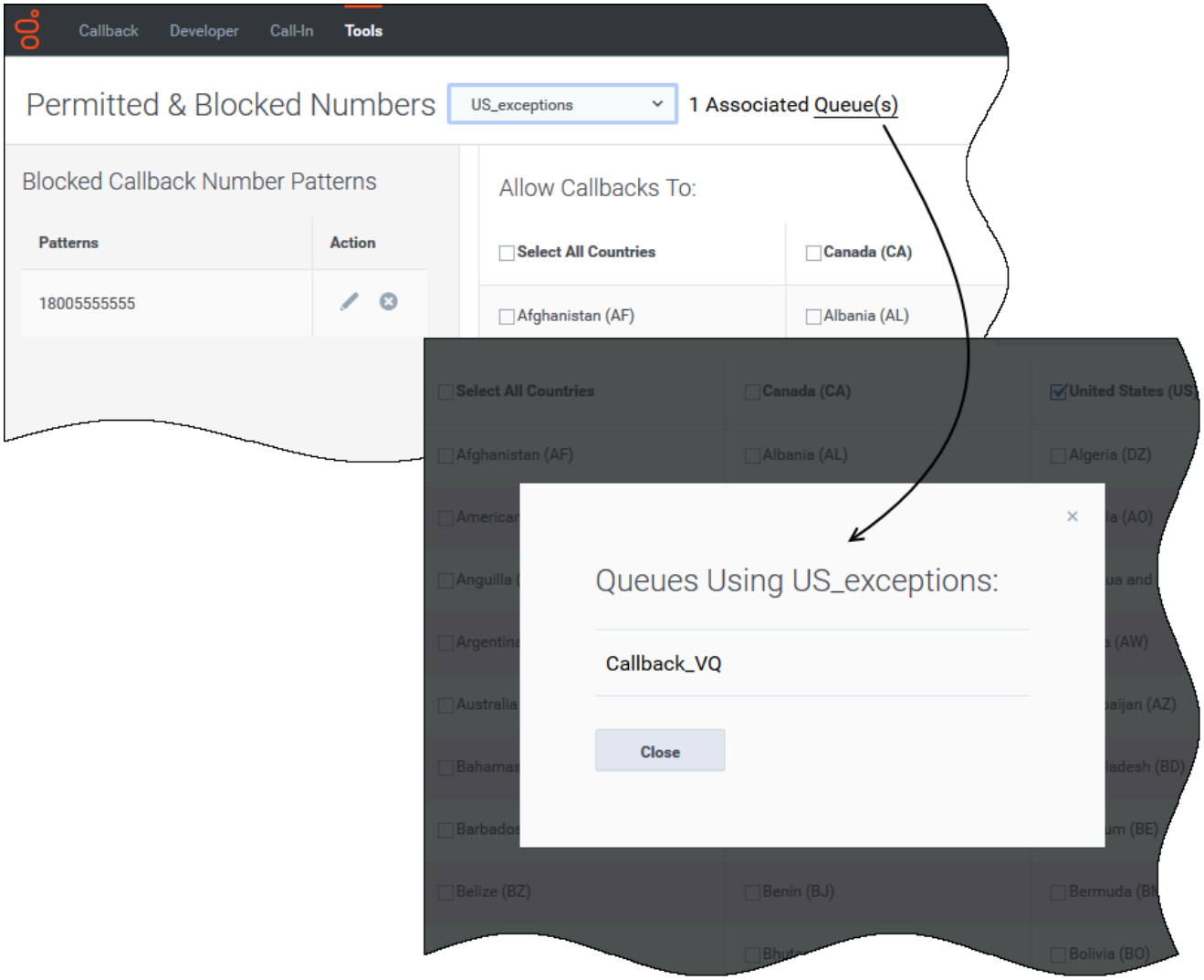
Assigning Rules to Queues



After you have configured at least one rule, you can assign the rules to your queues. You can assign a rule to multiple queues, but you can assign only one rule to each queue.

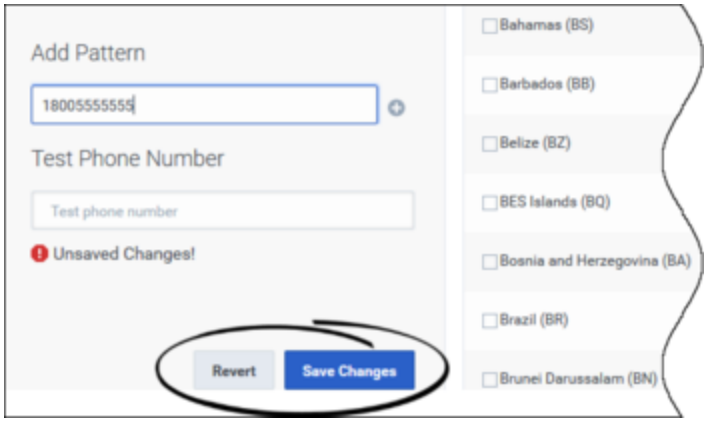
To assign a rule to a queue, navigate to the **Tools** > **Queues** page in your Callback UI. Locate a queue to which you want to assign a rule. You might need to filter the list of queues to locate what you're looking for. Once you find the queue, open the drop-down menu in the **Number Validation Rule** column. Select the rule that you want to assign to the queue. The rule is applied to the queue immediately and that change to the queue is saved automatically.

Editing and Deleting Rules



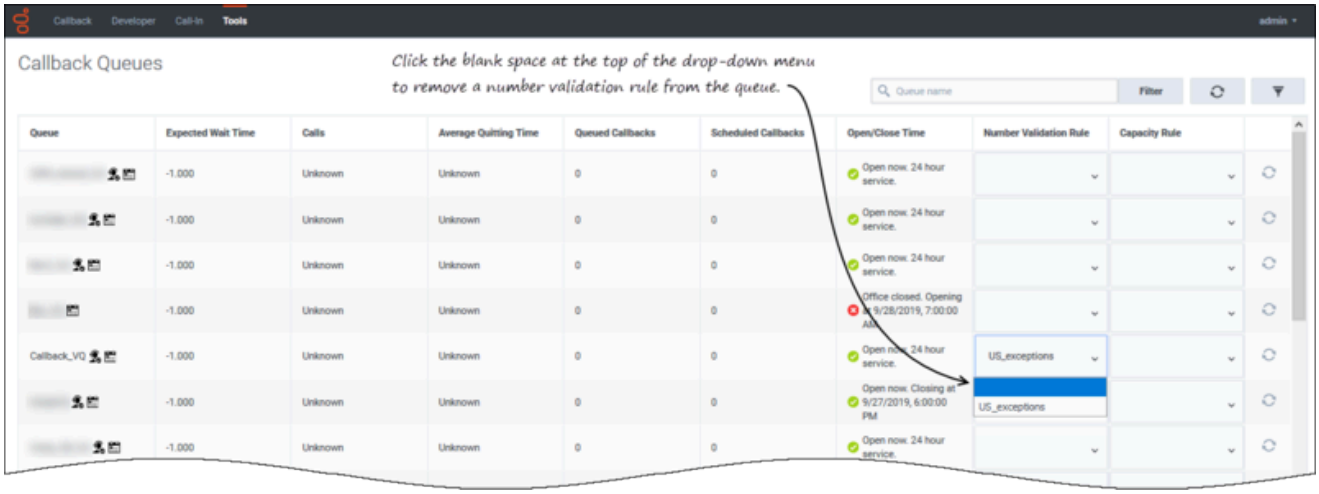
For your reference, each rule displays the list of queues to which the rule is assigned (or indicates that the rule is not yet assigned) so you will know which queues, if any, you are impacting when you edit a rule.

Changes that you make to a rule's definition are applied to any queue to which the rule is assigned as soon as you click **Save Changes**. If, while editing a rule, you decide that you want to discard your changes – perhaps you have made some mistakes or you discover that you are editing the wrong rule – click **Revert** to return the rule to its last-saved configuration.

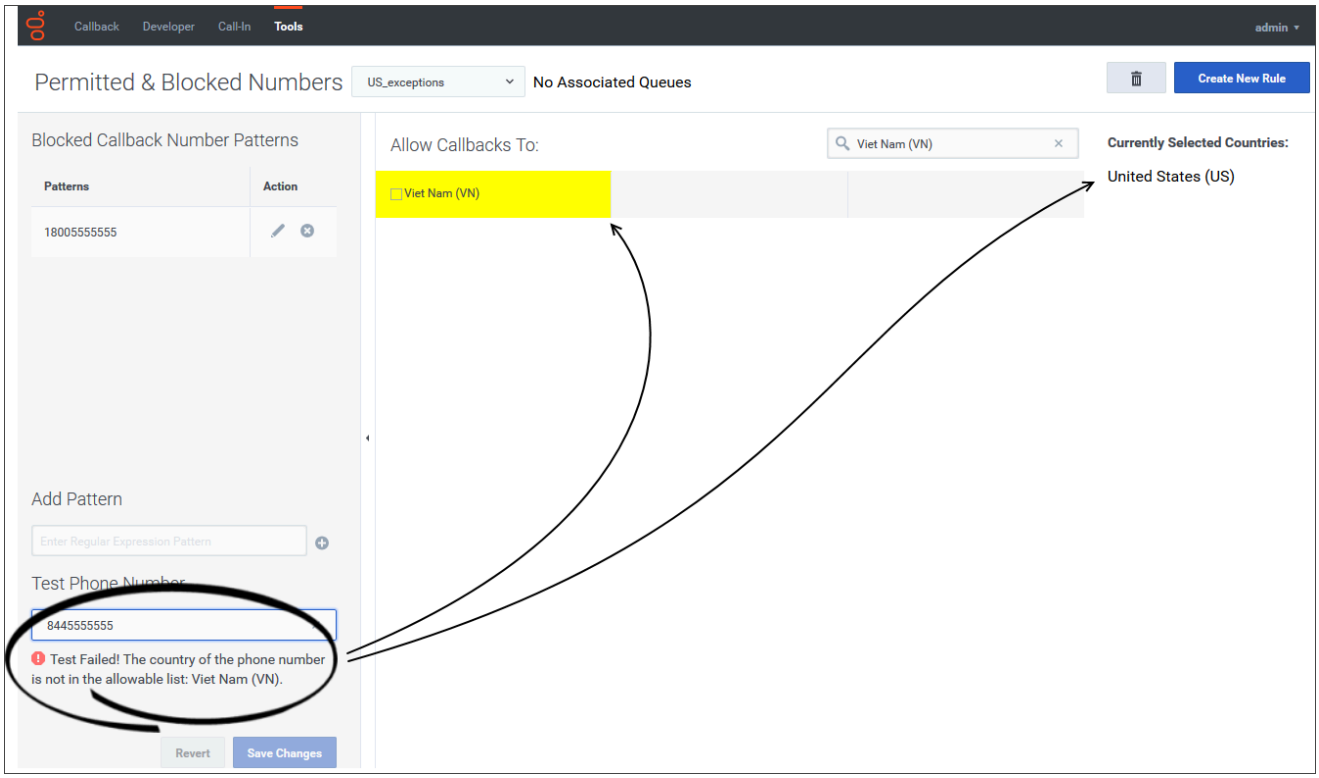


To help you manage impacts to queue activity, the Callback UI does not allow you to delete a rule that is assigned to a queue. Use the list of queues associated with the rule to help you determine if the rule can be safely removed from each assignment.

Navigate to the **Callback Queues** page in the Callback UI to remove a rule from a queue. Once the rule is removed from all queue assignments, you can delete it.



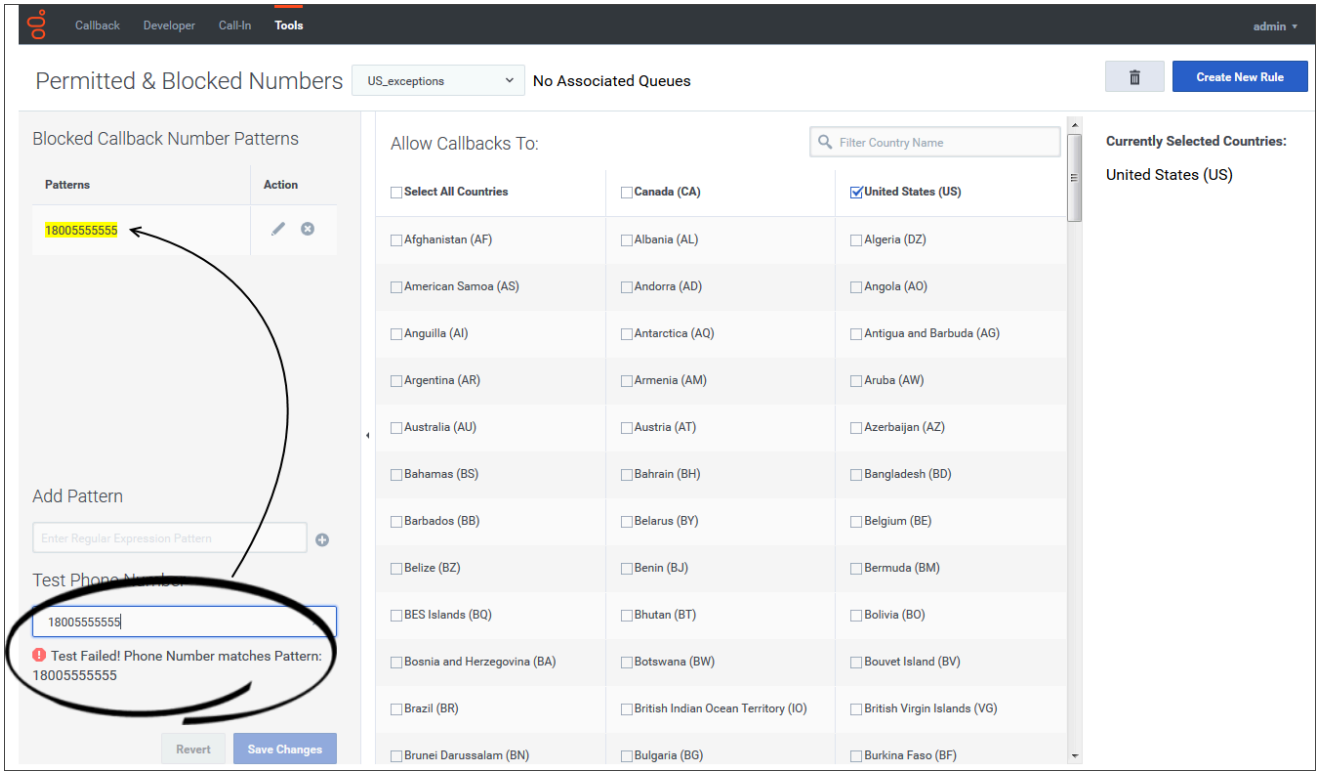
Testing a Number Pattern



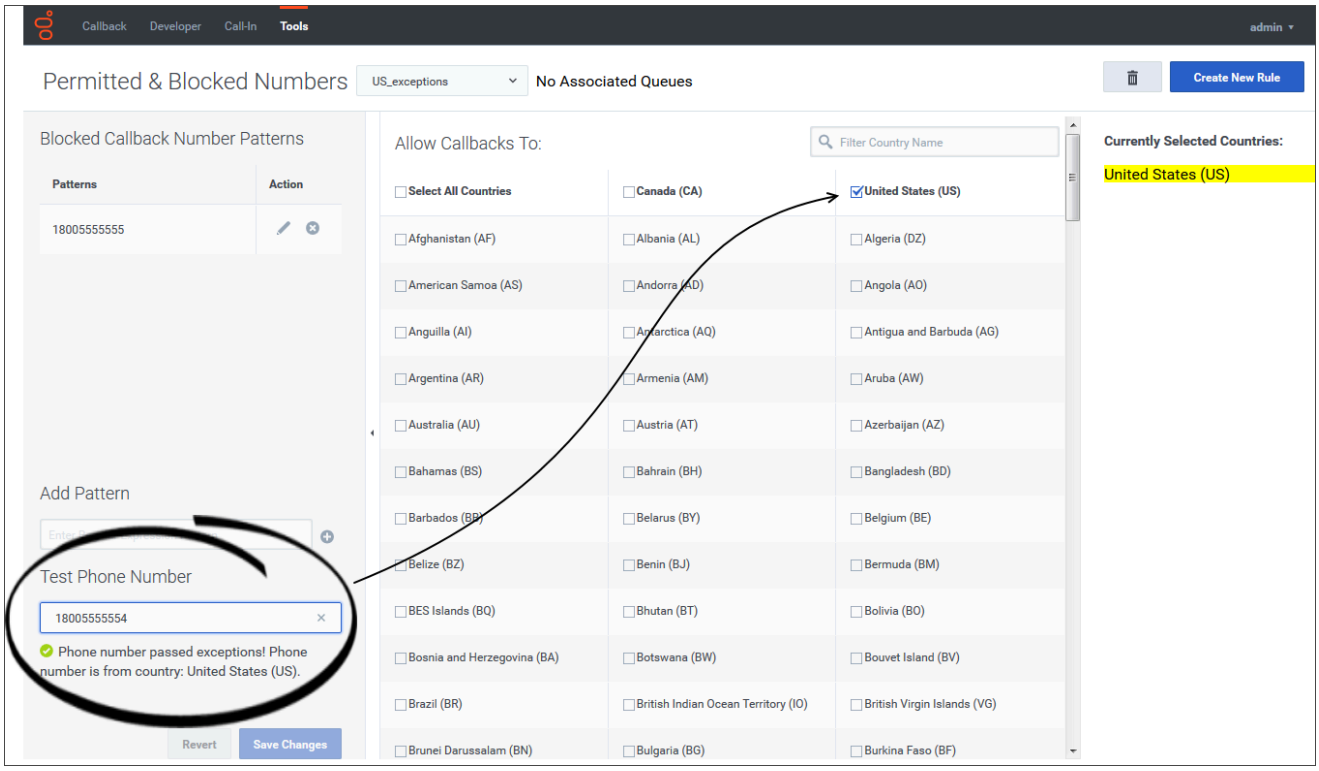
After you have created and saved a rule, you can use the rule to validate any number quickly using the testing feature on the **Permitted & Blocked Numbers** page. For example, you might be troubleshooting an error and you want to see if a rule might be interfering with the callback.

Enter a number in the **Test Phone Number** field. The number fails the rule's validation test if the number belongs to a country that is not added to that rule or if the number matches any one of the defined blocked number patterns.

If the number fails the check against allowable countries, then the system returns this reason for failure and does not validate the number against the blocked number patterns, even though the number might also fail that test.



If a number passes the country validation test, then it is checked against the blocked number patterns.



If the number belongs to a country to which callbacks are allowed for that rule and the number is not listed within the blocked number patterns, then the system validates that the number is acceptable for callback attempts.

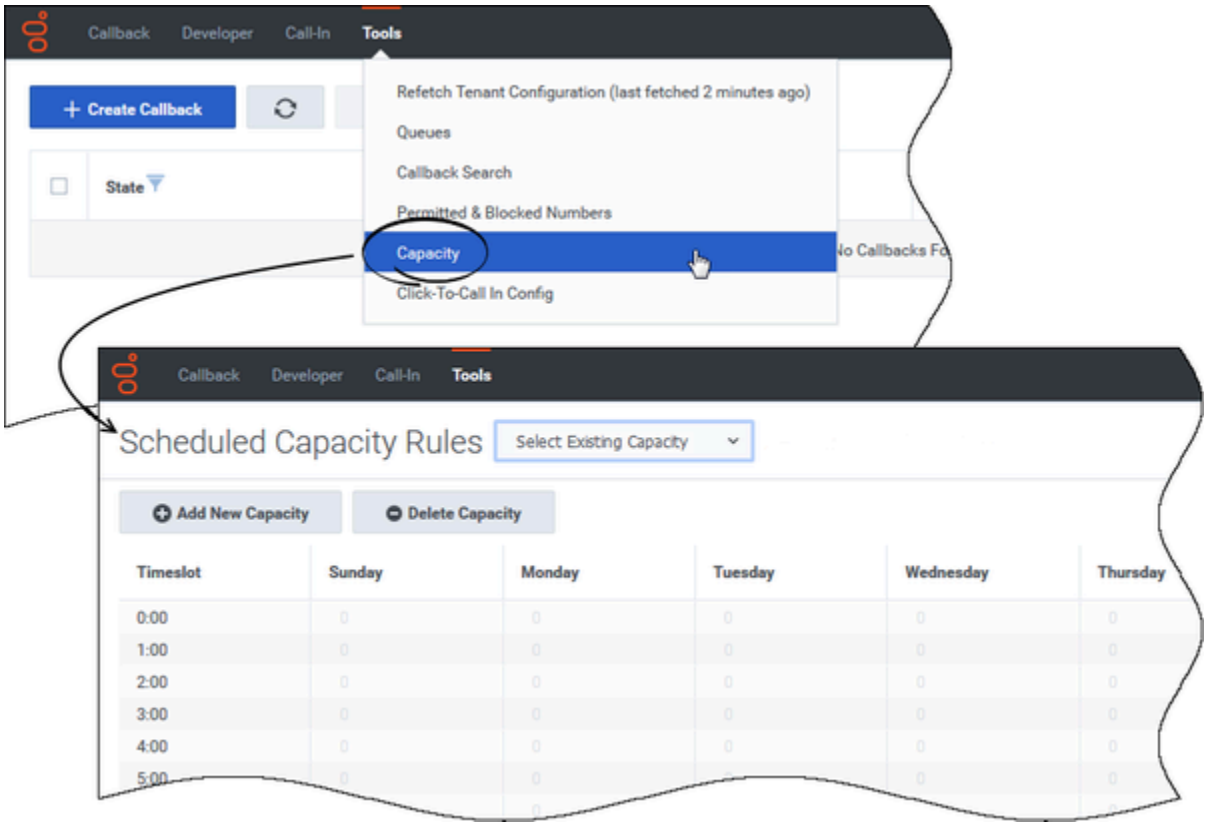
Configuring capacity rules

Important

This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Configuring Capacity rules.

Important

To create or edit capacity rules, you must be a member of the **Callback Administrator** role. The **Callback Developer**, **Callback Supervisor**, and **Callback Monitor** roles have read-only access. Access to queues is subject to line-of-business filtering where role-based access control is enabled.



Callback capacity refers to the maximum number of scheduled callbacks that are allowed in a given

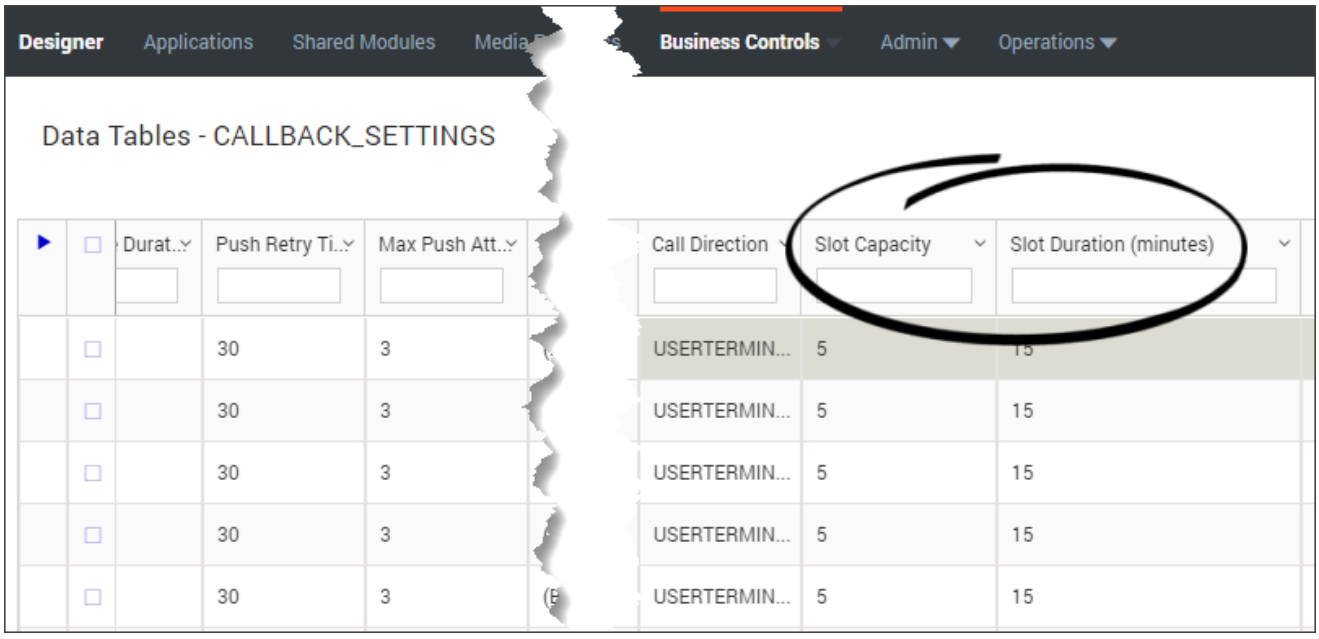
time interval. On the **Tools** > **Capacity** tab, you create *capacity rules*. A capacity rule contains a day-by-day weekly calendar on which you specify callback capacity for each time slot. A time slot is either a one-hour slot or a half-hour slot; you select the time slot duration from a drop-down menu. Once you have configured a capacity rule, you can assign it to queues. A capacity rule can be assigned to multiple queues, but you can assign only one capacity rule to each queue.

Time zone associations

There is no time zone associated with a capacity rule. During the evaluation process to determine if a scheduled callback can be booked or not, the time zone associated with the business hours service is used. Be sure you have the business hours, including the time zone, configured (in Designer) for the queue before you apply any capacity rules. For more information about configuring business hours, see Business Hours and Business Hours Block in the Designer documentation.

How capacity rules affect capacity settings configured in Designer

A capacity rule that you configure in the Callback UI and assign to a queue has an effect on the **Slot Capacity** and **Slot Duration** settings in Designer's CALLBACK_SETTINGS data table.



A value that you specify for a time slot on the **Capacity** page in Callback is the maximum number of scheduled callbacks that you can support *for each* ***Slot Duration*** (configured in Designer) within that time interval. For example, if you configure a capacity rule for 9:00 to 10:00 a.m. on Monday morning with a value of 5 and you assign that rule to a queue for which the **Slot Duration** in Designer is 15 minutes, then what the rule allows is a maximum of 5 scheduled callbacks every 15 minutes between 9:00 and 10:00 a.m. on Monday morning.

A capacity rule assigned to a queue *overrides* the **Slot Capacity** setting in the Designer CALLBACK_SETTINGS data table for that queue. For example, if you configure a rule for 3:00 to 4:00 p.m. on Wednesday afternoon with a value of 5 and you assign that rule to a queue for which the **Slot Capacity** in Designer is set to 3, then the rule allows a maximum of 5 scheduled callbacks for each **Slot Duration** during that hour on Wednesday afternoon, overriding the value of 3 that was set in the CALLBACK_SETTINGS table.

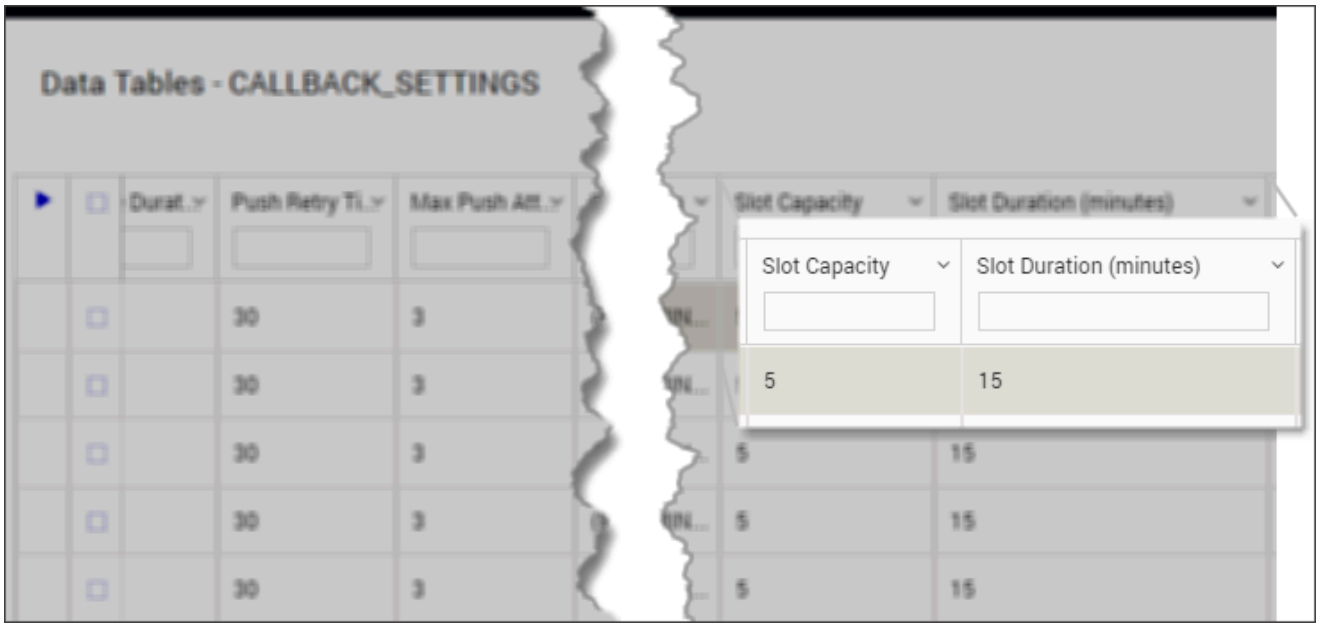
Before you start

Before you create capacity rules and assign them to queues, make sure that you have provisioned Callback in Designer. In particular, make sure that the following configuration is complete:

- In Designer, you have provisioned your business hours, including the time zone. See also Business Hours and Business Hours Block in the Designer documentation.
- In Designer, you have provisioned the callback services, which includes specifying a value for the **Slot Capacity** and **Slot Duration (minutes)** columns for your callback application. See also Callback Settings Data Table in the Designer documentation.

Using a rule to adjust capacity

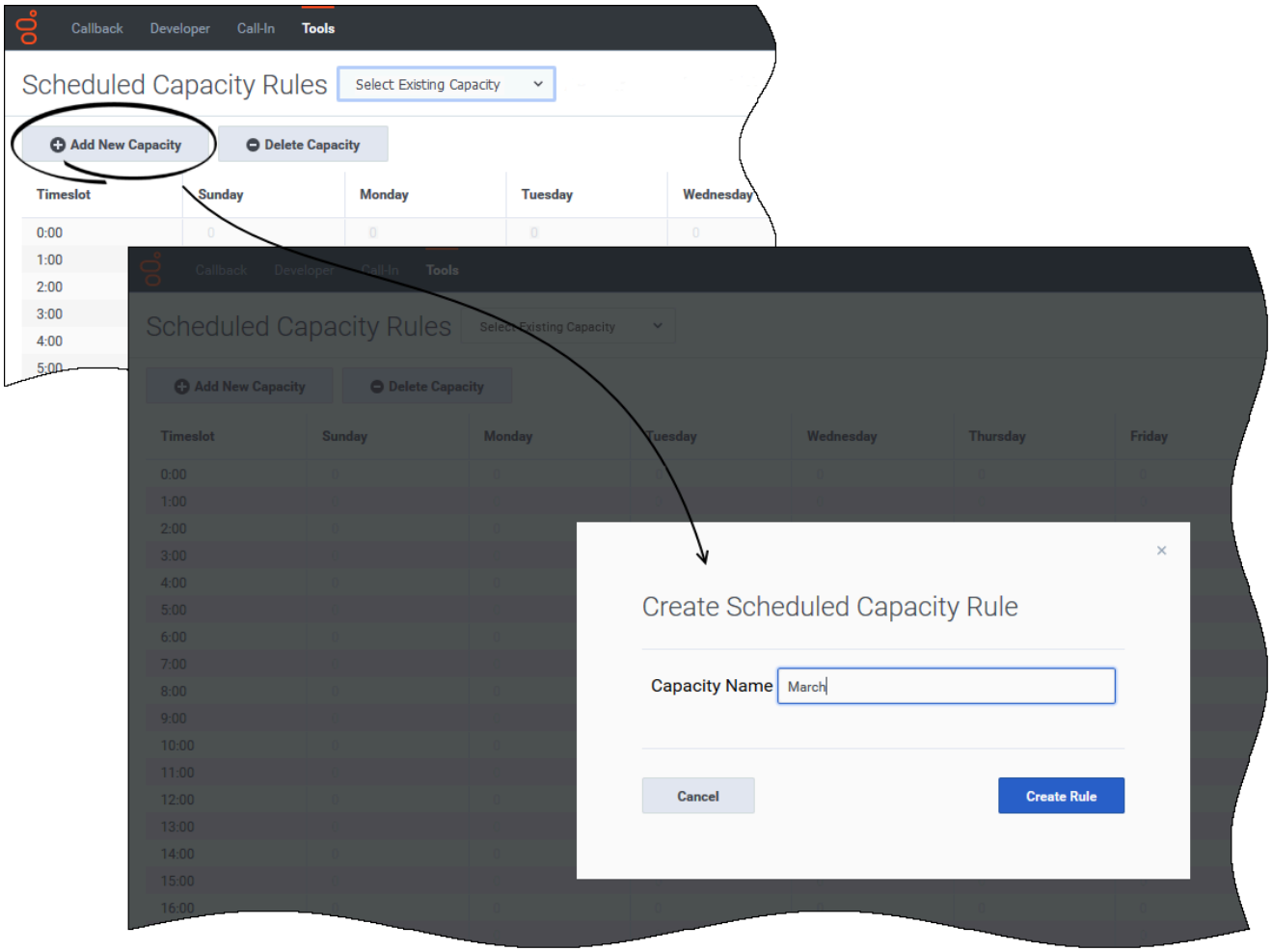
Let's assume that we have a special campaign running in the month of March because we are launching a new product feature. We anticipate higher than normal call volumes, so we plan to increase the number of skilled agents to support customer inquiries. We also anticipate more callback requests, so we plan to adjust the capacity for scheduled callbacks during the entire month of the launch.



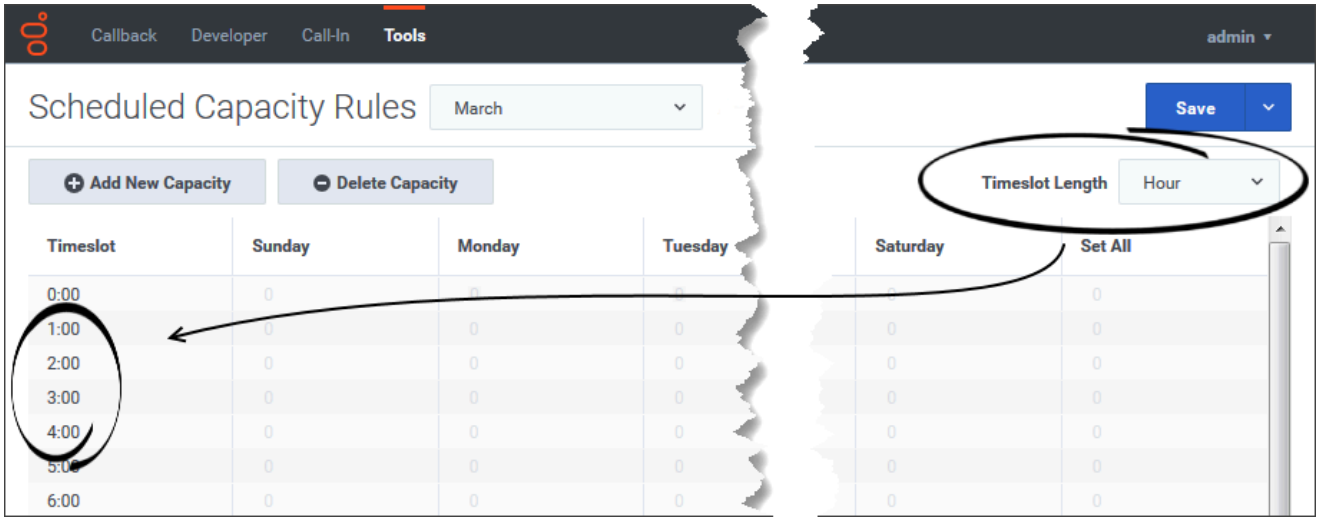
In the CALLBACK_SETTINGS data table in Designer, let's say that we have set the capacity for scheduled callbacks to 5 and specified a slot duration of 15 minutes for the callback application. In other words, on a normal day we can handle 5 scheduled callbacks every 15 minutes during open office hours. In Designer, we set this "flat rate" for capacity, but – for the month of March – we want to make some adjustments to that. We make the adjustments using a capacity rule that we'll configure and assign to queues in Callback.

Let's say that our business hours are 8:00 a.m. to 6:00 p.m., Monday to Friday. For the month of March, we plan to have 40 agents from 8:00 a.m. to noon, 50 agents from noon to 2:00 p.m., and back to 40 agents from 2:00 to 6:00 p.m. Slot duration in the CALLBACK_SETTINGS table is, as previously mentioned, set to 15 minutes.

Configuring the "March" capacity rule



To create a capacity rule, you must first give it a name. You cannot edit the name of a rule after you click **Create Rule**, so be sure that you have entered the name exactly as you intended. We'll call this new rule **March** because we are making adjustments to the capacity for that month. The rule can be applied to all relevant queues for the month of March to override the setting in Designer.



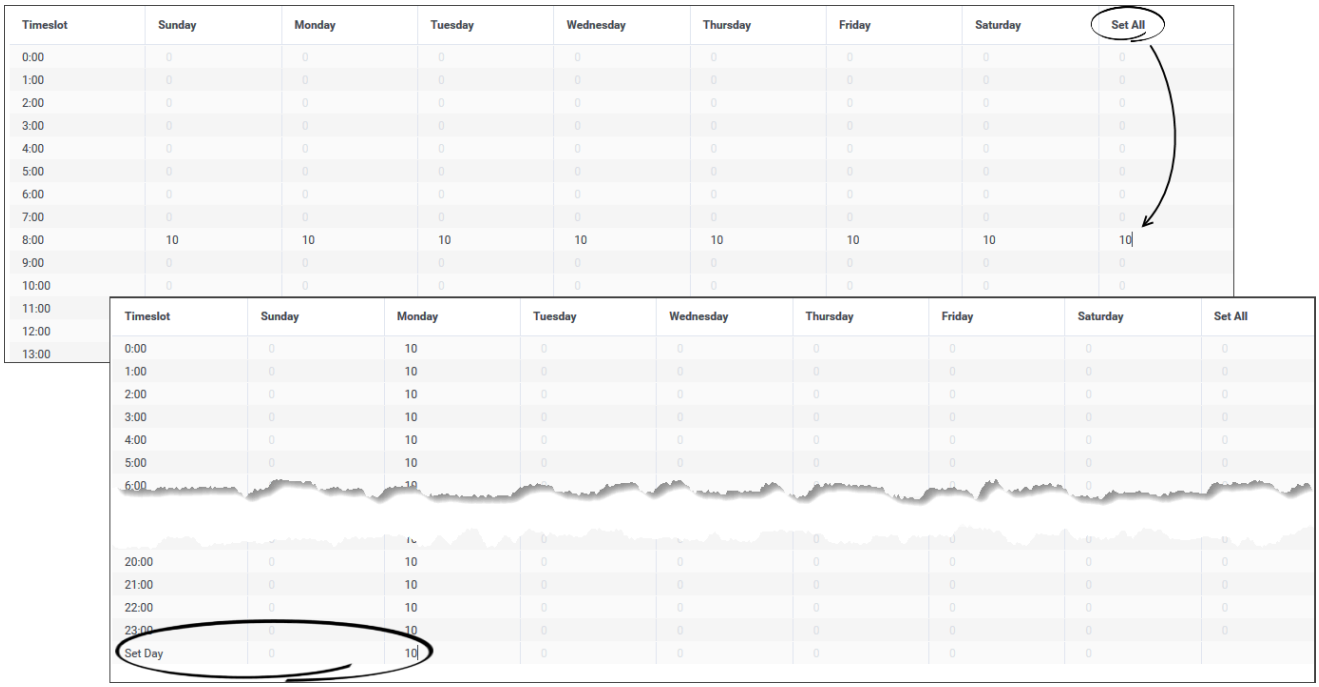
Select the **Timeslot Length**. This value sets the time slot duration on the *capacity calendar*, it does not change the **Slot Duration** that you set in Designer. This setting governs all time slots on the calendar; you cannot have some half hour-long slots and some hour-long slots.

For this example, we will set the timeslot length to `Hour`.

An hourly time slot begins at the top of every hour (:00). Half-hourly time slots begin at the top of every hour (:00) and at the bottom of every hour (:30). If you plan to use capacity rules with queues, Genesys recommends that you always set the **Slot Duration (minutes)** in Designer to a value that is evenly divisible into 60. Otherwise, you might experience unpredictable results when a capacity rule is applied.

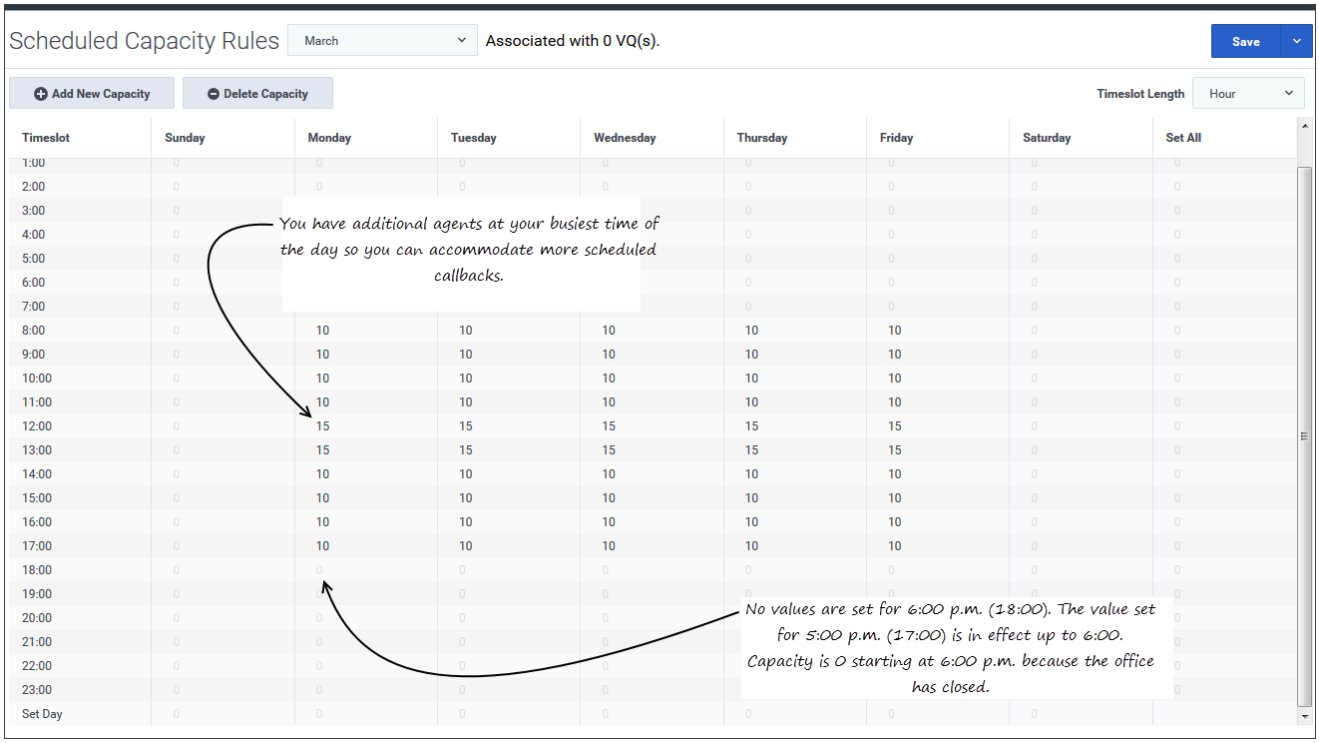
Once your capacity calendar is set up, then you can start adding the capacity values. Remember that the capacity values that you specify on the **Capacity** page indicate the maximum number of scheduled callbacks that you have the capacity to handle in each **Slot Duration** that occurs within the hour or half-hour that you're configuring. To find the **Slot Duration** that is configured for a particular queue, you must check the CALLBACK_SETTINGS data table in Designer.

If your contact center is open 24/7 and you want to set the same value for the same time slot on every day of the week, enter the value in the **Set All** cell for that time slot. To set the same value for every time slot on a particular day, enter the value in the **Set Day** cell for that day.



When creating a capacity rule, if you do not enter a value for a particular time slot, then the system reads that time slot as having 0 or *no capacity* when determining if a scheduled callback can be booked.

Currently, Callback does not support the definition of exceptions for specific dates; for example, statutory holidays.



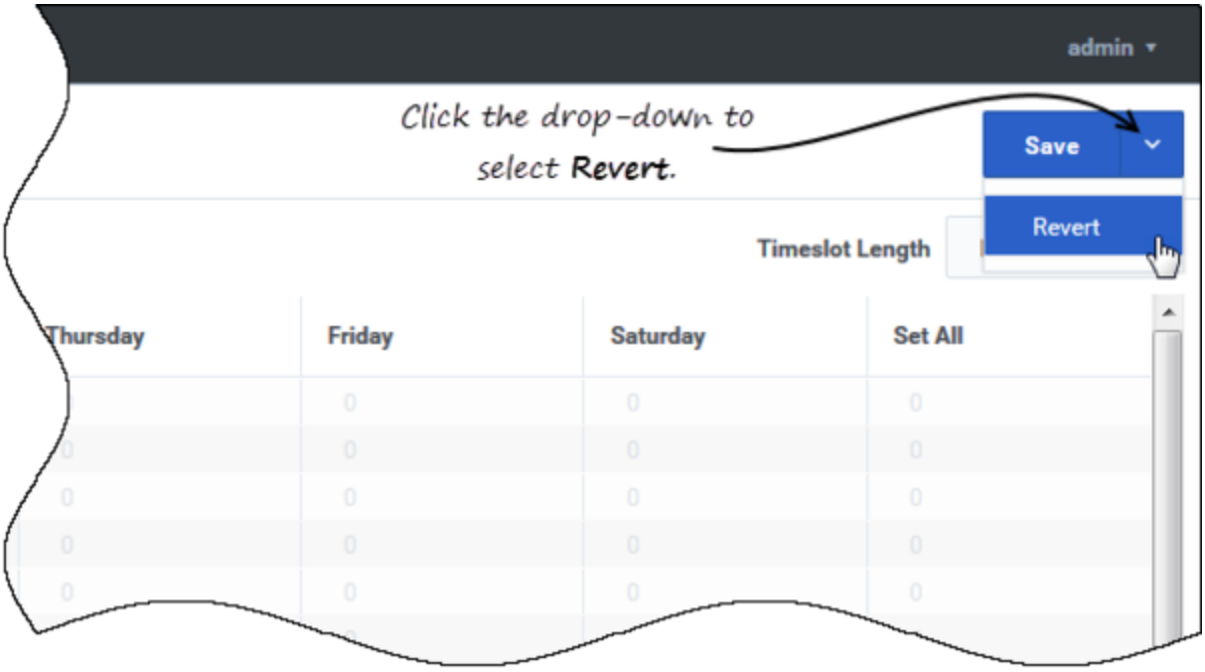
Returning to our sample capacity rule (**March**), we will set values for the days and hours that we're open (8:00 a.m. to 6:00 p.m., Monday to Friday) based on the additional agents that we have during the month of March. Slot duration in the CALLBACK_SETTINGS table is, as previously mentioned, set to 15 minutes. This means that we will be overriding the 5 scheduled callbacks that we normally have capacity to handle every 15 minutes with the following capacity values:

- up to 10 scheduled callbacks every 15 minutes from 8:00 a.m. to 12:00 p.m.
- up to 15 scheduled callbacks every 15 minutes from 12:00 p.m. to 2:00 p.m. (there are extra agents on phones during peak hours)
- up to 10 scheduled callbacks every 15 minutes from 2:00 p.m to 6:00 p.m.

Once this capacity rule is assigned to the callback virtual queue, the capacity specified in the rule takes effect.

After you have set some or all of the values, remember to save the rule. You can edit the values at any time. Saved changes to rules will be applied immediately to any queue to which the rule is assigned.

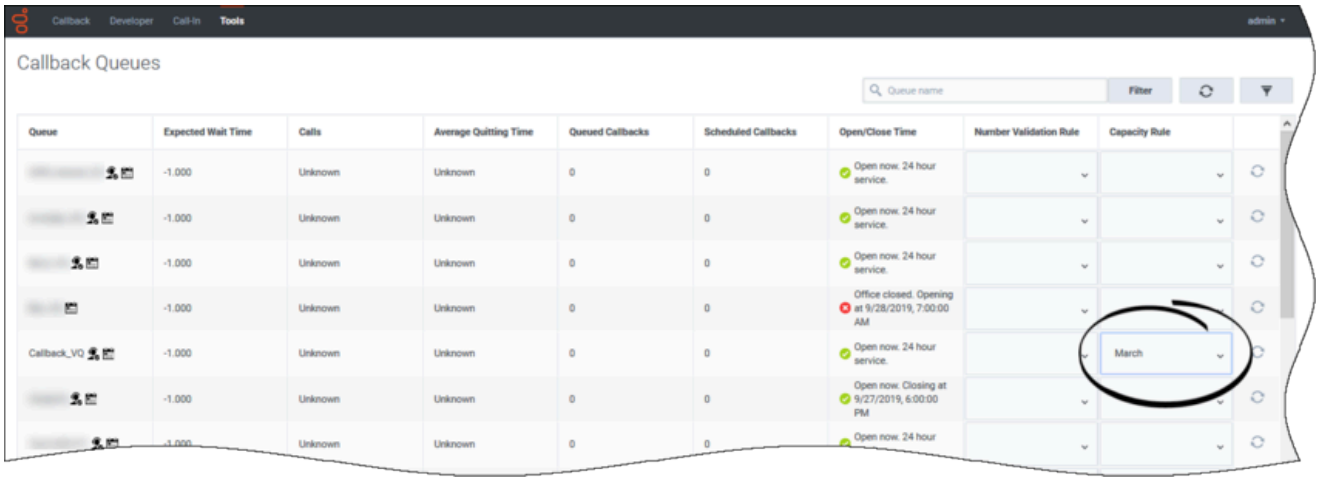
If you edit values in the calendar after it has been saved and you discover that you have made some mistakes – perhaps you are editing the wrong rule – click **Revert** to return to the last-saved values.

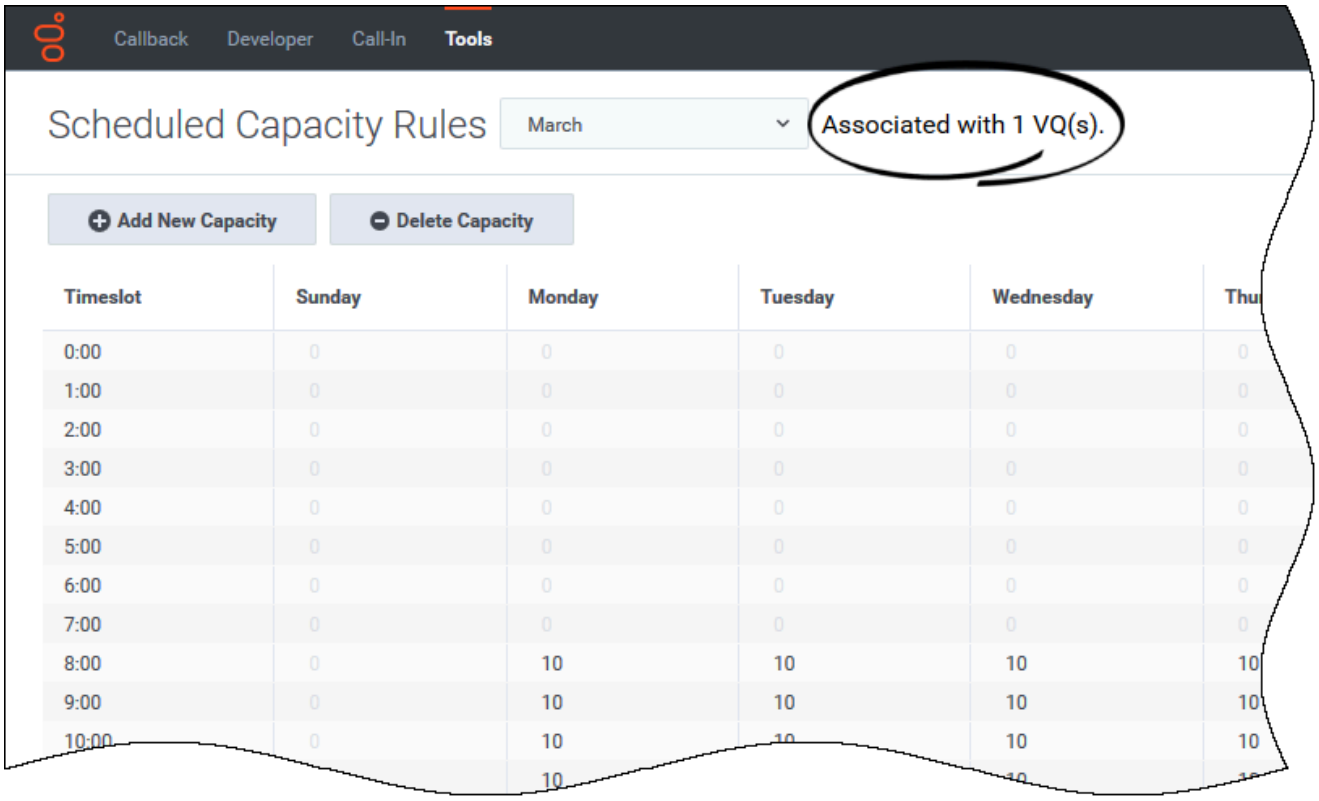


Assigning a capacity rule to a queue

A rule is created based on one week (Sunday to Saturday), but the capacity specified in the rule for each time slot of every day remains in effect on a queue for as long as the rule is assigned to that queue. After you have configured at least one rule, you can assign the rule or rules to your queues. You can assign a rule to multiple queues, but you can assign only one rule to each queue.

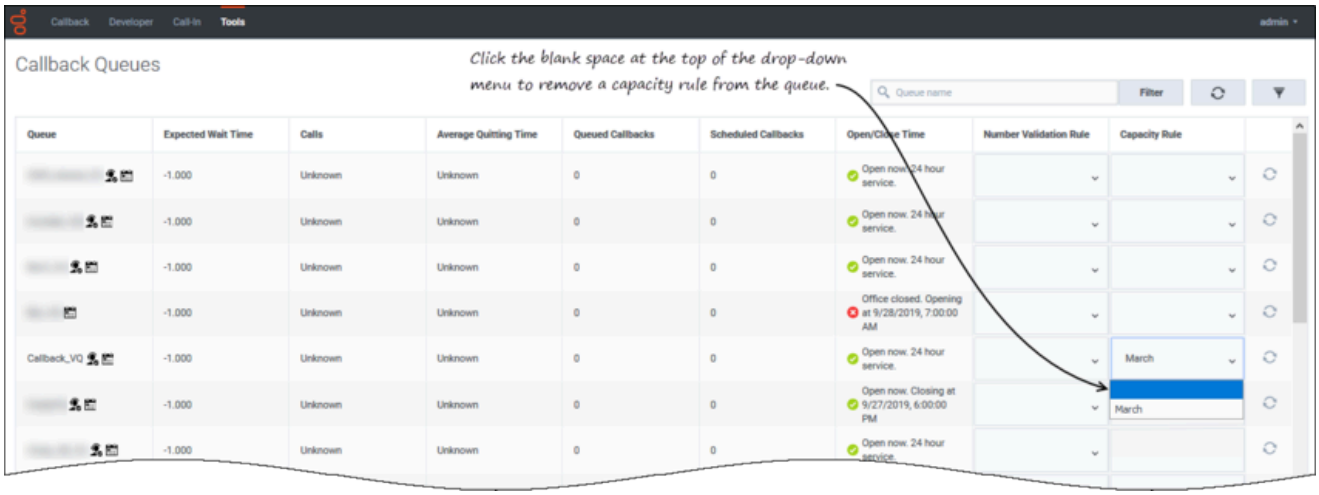
To assign a rule to a queue, navigate to the **Tools** > **Queues** page in your Callback UI. Locate a queue to which you want to assign a rule. You might need to filter the list of queues to locate what you're looking for. Once you find the queue, open the drop-down menu in the **Capacity Rule** column. Select the rule that you want to assign to the queue. Continuing our example, we will assign the **March** capacity rule to the callback virtual queue. The rule takes effect immediately on that queue.





TIP: You can select a rule on the **Scheduled Capacity Rules** page to see how many queues the rule is assigned to. This is important if you want to delete a rule. As long as it is assigned to at least one queue, you won't be able to delete it.

Once March is over and we want to return to our normal level of scheduled callbacks, we simply remove the assignment of the **March** rule from the queue.



Using JSON-format capacity statements to configure a capacity rule

You can create a capacity rule and manually enter your capacity for various hours and days or you can define your capacity in JSON format and then copy and paste the JSON capacity statement directly into an open and focused capacity rule using your usual pasting shortcut; for example, Command/Ctrl+V or the contextual menu. Make sure that no cells are selected in the capacity calendar. If a cell is selected, then the capacity data will be pasted into only the selected cell.

When you paste a JSON-format capacity statement into a rule that already has data, the pasted capacity data completely overwrites the existing data. For example, let's say that we have a capacity rule already defined and it includes a capacity of 5 on Tuesday from 9:00 a.m. to 10:00 a.m. We paste a JSON-format capacity statement into the rule, but the pasted capacity data does not include a value for the Tuesday, 9:00 a.m. to 10:00 a.m. time slot. After we paste in the JSON-format capacity statement, we notice that the capacity for that time slot has changed to 0. This happened because the pasted-in capacity statement did not merge with the existing data, it wrote over it. If nothing is defined for the 9:00 to 10:00 time slot on Tuesday morning, then the UI interprets that as a capacity of 0. If you want to maintain existing data, remember to include that data in your JSON-format capacity statement.

The following sample shows the correct format to use for JSON capacity statements:

```
{"days":{"sun":{"02:00":2},"mon":{"05:00":2},"tue":{"07:00":3,"06:00":4},"wed":{"08:00":5},"thu":{"11:00":6},"f
```

How capacity is evaluated for a "short" time interval

The callback system, when queried for availability, first checks the business hours that are configured in Designer. For more information about configuring Business Hours, see Business Hours and Business Hours Block.

As mentioned earlier on this page, the Callback capacity rules affect the **Slot Duration** and **Slot Capacity** settings that are configured in Designer. Because the capacity rules are based on callback capacity for time slots that begin at either the top of the hour or the bottom of the hour, there are situations in which a time slot might be ignored (no callbacks will be scheduled in that slot). For example, if your office hours begin at 7:50 a.m., **Slot Duration (minutes)** is set to 15 minutes in Designer, and a capacity rule assigned to the callback queue specifies a capacity of 10 scheduled callbacks between 7:30 and 8:30 a.m., then the system will begin scheduling callbacks for 8:00 a.m. There can be up to 10 scheduled callbacks between 8:00 and 8:15, and another 10 between 8:15 and 8:30. The system will not schedule callbacks between 7:30 and 7:45 because the business is not open during that slot, and the slot 7:45 to 8:00 is also considered to be unavailable because the business is still closed at 7:45. In other words, Callback checks the conditions only at the start of a **Slot Duration** period.

Switching the timeslot length in a saved capacity rule

If a capacity rule was initially configured with half-hourly time slots in Callback, and you later decide that you want to change it to hourly time slots, then you can certainly do so. The capacity values that were originally configured for the bottom of each hour are "dropped" and the capacity values that were set at the top of each hour are used instead.

Original half-hourly capacity rule settings

Time:	9:00	9:30	10:00	10:30
Capacity Value:	8	10	8	10

Half-hourly converted to hourly

Time:	9:00	(9:30)	10:00	(10:30)
Capacity Value:	8	(8)	8	(8)

When you convert a half-hourly capacity rule to hourly, then the value that was set at the top of the hour is duplicated at the half hour.

Original hourly capacity rule settings

Time:	9:00	10:00
Capacity Value:	8	8

Hourly converted to half-hourly

Time:	9:00	9:30	10:00	10:30
Capacity:	8	8	8	8

When you convert an hourly capacity rule to half-hourly, then the value that was set at the top of the hour is duplicated at the half hour.

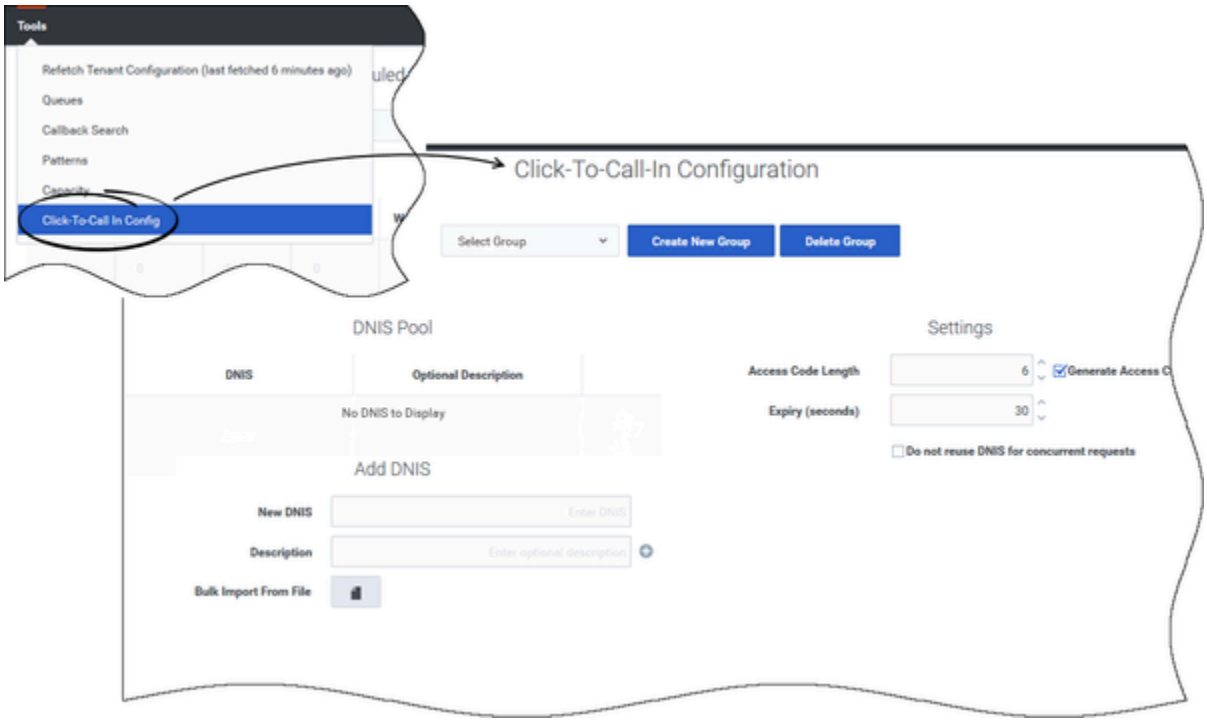
Configuring Click-To-Call-In Groups

> **Important**
>
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Configuring Click-To-Call-In Groups.

Callback supports the Click-To-Call-In scenario. To implement the scenario for use with Callback, you must configure at least one Click-To-Call-In group. A Click-To-Call-In group definition consists of a Dialed Number Identification Service (DNIS) pool and access code settings, including the choice to use or not use an access code with Click-To-Call-In requests. The system uses the DNIS pool associated with a group to find the number that it will provide to the customer who is making the Click-To-Call-In request. The system also uses the group configuration to determine whether or not an access code is required when the customer dials in.

You can find information about using the Genesys Engagement Service (GES) APIs, including the Click-To-Call-In API, on the Genesys Engage REST APIs and Tutorials for Callback page in this Guide. The page includes links to API documentation.

In Callback, you provision the Click-To-Call-In groups on the **Tools** > **Click-To-Call-In Config** tab.



There are two ways in which you can enter the DNIS pool of numbers for a Click-To-Call-In group:

- Enter each DNIS number individually using the fields on the **Click-To-Call-In Config** page.
- Bulk import the DNIS numbers for a group.

The information on this page describes both methods. For general information, including how to enter a DNIS pool manually, see Provisioning Click-To-Call-In Groups. For information about the bulk import of a DNIS pool, see Performing a bulk import of DNIS pool numbers below.

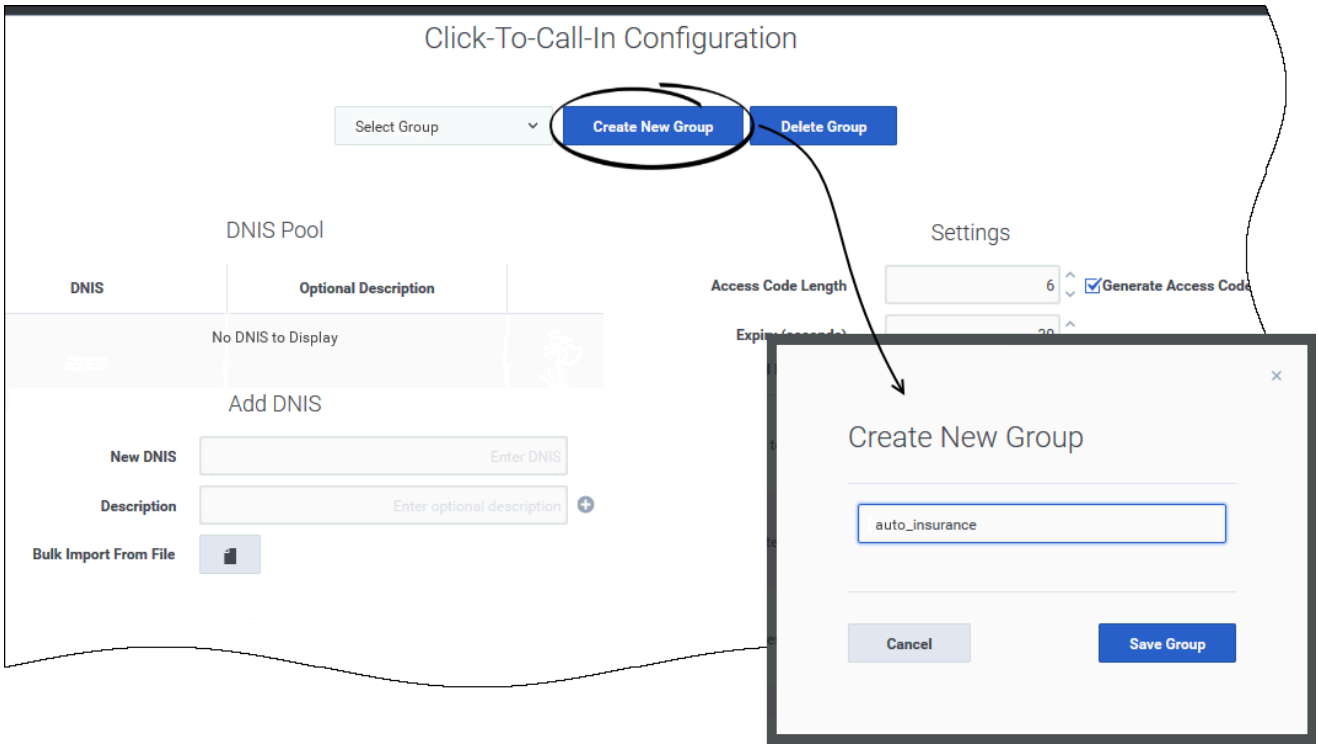
For information about the Callback Call-In tab, on which you can view the Click-To-Call-In records, see Viewing Click-To-Call-In records.

A Note about Migration

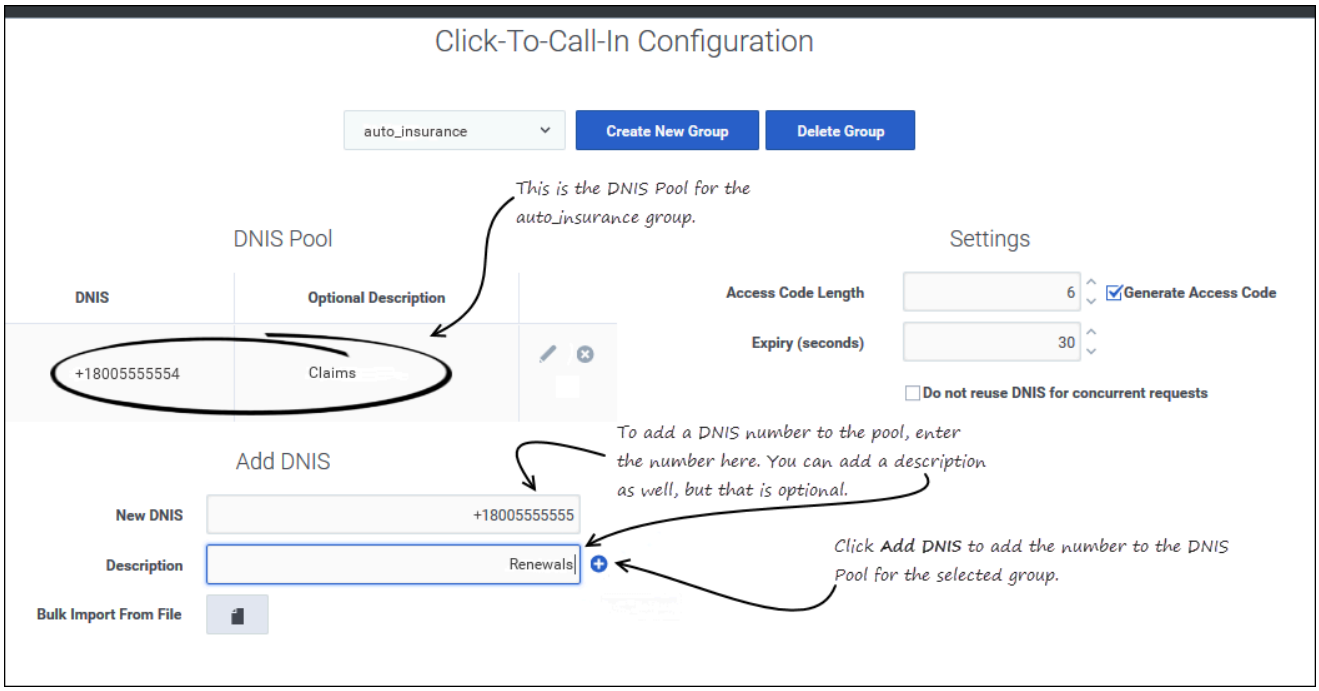
If you are migrating to Callback, make sure that you enter any existing Click-To-Call-In group information on the **Click-To-Call-In Config** page in the Callback UI. Previously, the Click-To-Call-In group information was configured in Platform Administration.

Provisioning Click-To-Call-In Groups

> **Important**
>
> To create or modify Click-To-Call-In groups, you must be a member of the **Callback Administrator** role. The **Callback Developer**, **Callback Supervisor**, and **Callback Monitor** roles have read-only access.



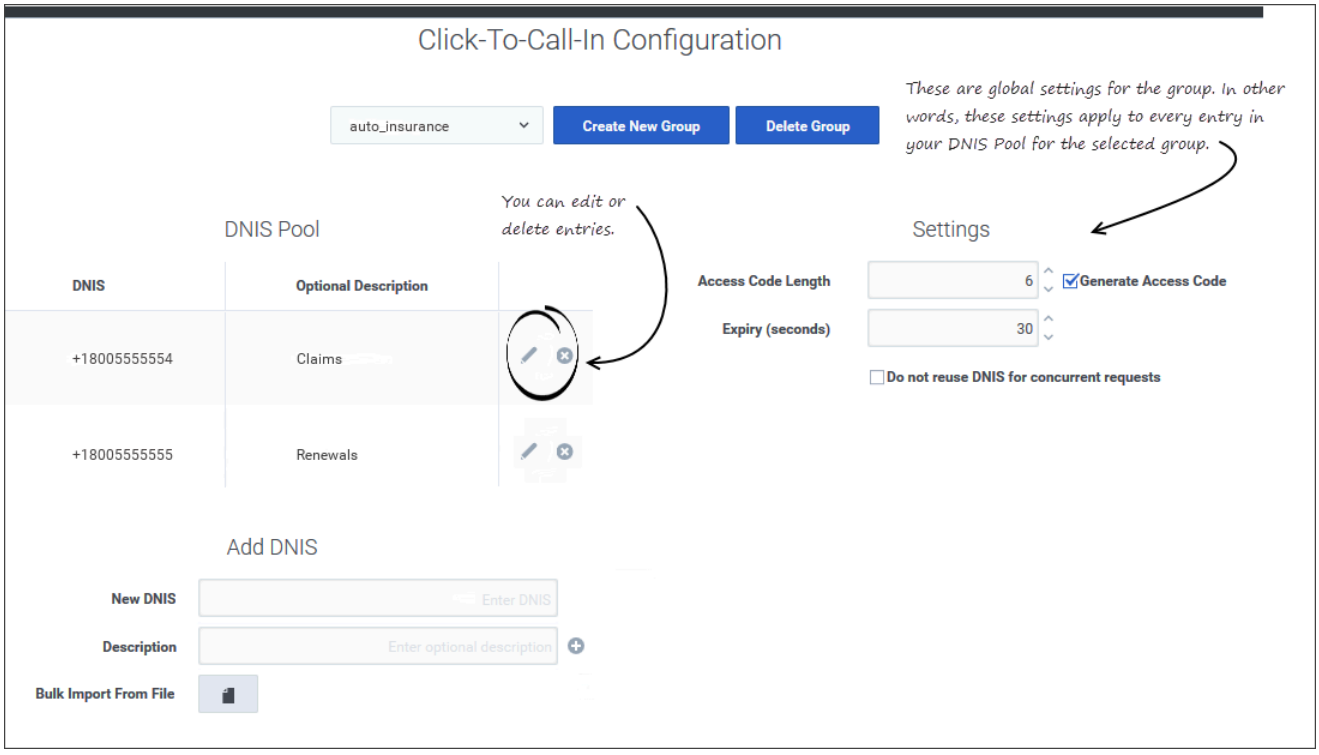
To create a new Click-To-Call-In group, navigate to the **Tools** > **Click-To-Call-In Config** page and click **Create New Group**. The UI accepts alphanumeric characters and symbols in the group name. You cannot edit the name of a group in the Callback UI after you click **Save Group**, so be sure that you have entered the name exactly as you intended before saving.



In the pane on the left side of the **Click-To-Call-In Configuration** page, you enter your DNIS pool of numbers. The group must contain at least one valid DNIS number (that is, it must conform to the E.164 format for telephone numbers). You can enter a maximum of 10,000 DNIS numbers for each group. A DNIS number cannot exceed 16 characters, and can include the "plus" sign (+) prefix.

You can include a description for a DNIS number, but that is optional. The UI accepts alphanumeric characters and symbols in the description. The description cannot exceed 32 characters.

If you have a large DNIS pool for a group, you can bulk import the numbers instead of entering each one individually. For more information about the bulk import method, see Performing a bulk import of DNIS pool numbers below.



In the pane on the right side of the page, you enable or disable the use of an access code. In other words, the access code is optional. If you enable the use of the access code, then you can specify settings for the code. If enabled, the system generates an access code for each Click-To-Call-In request and provides it to the caller at the same time that it provides the phone number that the caller must use to dial in. The caller must enter the access code correctly for the call to be directed to an agent.

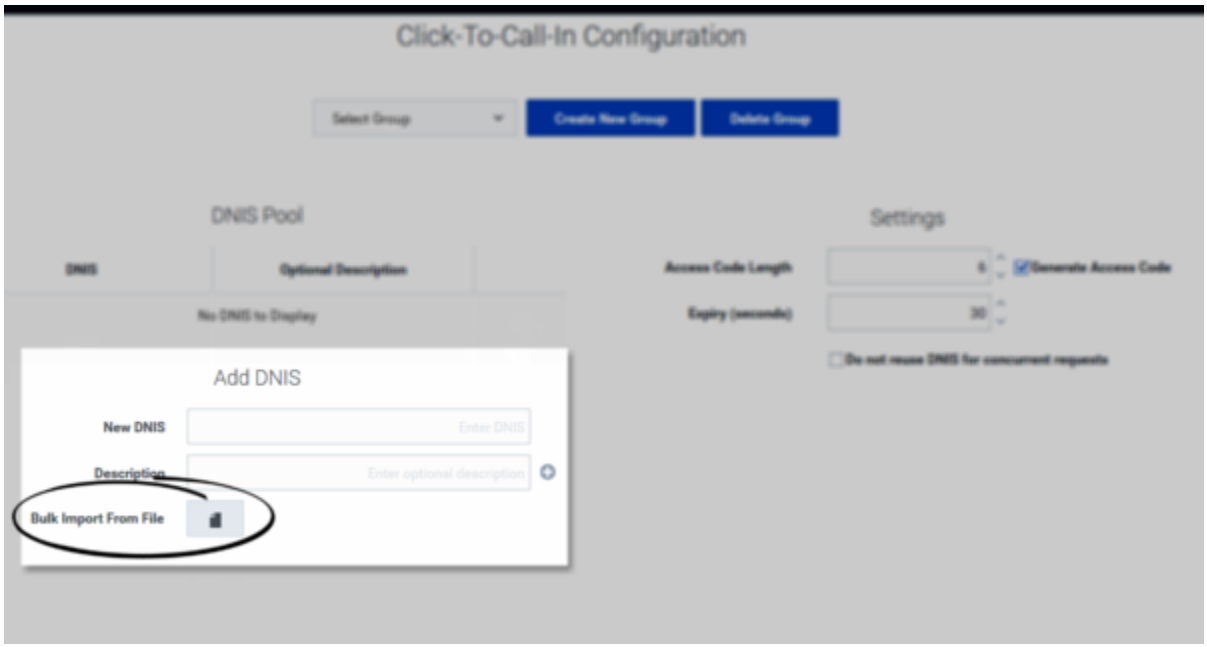
Additional information about the Settings

Setting	Description
Generate Access Code	Enable or disable the use of access codes with Click-To-Call-In requests.
Length	The length of an access code. Valid values are 4 to 12, inclusive. The default value is 6.
Expiry (seconds)	The lifetime, in seconds, of the Click-To-Call-In request. Valid values are 10 to 1800, inclusive. The default value is 30.
Unique DNIS	Specifies if a DNIS number is unique or not. "Unique" means that it cannot be reused while it is in use. Select the check box if the DNIS is unique. Otherwise, leave the check box clear. The default value is false, or "not unique" (the check box is not selected).

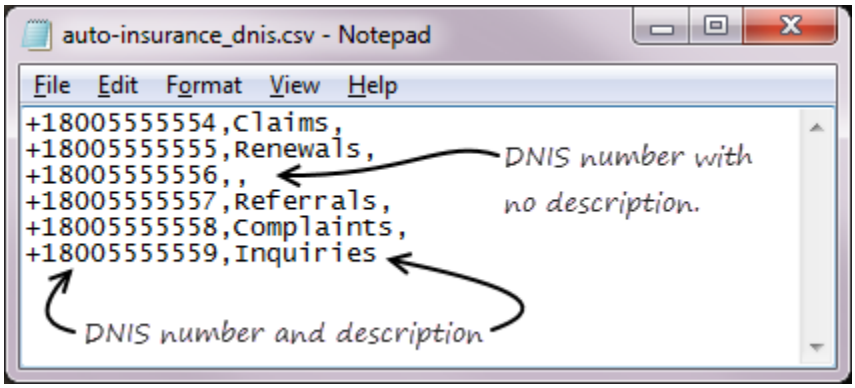
Performing a bulk import of DNIS pool numbers

If you have a large DNIS pool associated with a Click-To-Call-In group, you can import the DNIS pool as a bulk upload. After you create the group on the **Tools** > **Click-To-Call-In Config** page, then use

the **Bulk Import From File** button to import the DNIS pool from a .csv file.



If you use the bulk import method, you will require a separate .csv file for each group. You can upload the DNIS pool for a single group from a file; you cannot upload the DNIS pool for many groups from one file.



The bulk import method supports only .csv files (values must be separated by a comma). You can include a description, but it is optional. In the sample file shown here, there is no description for one entry, although you still require the commas.

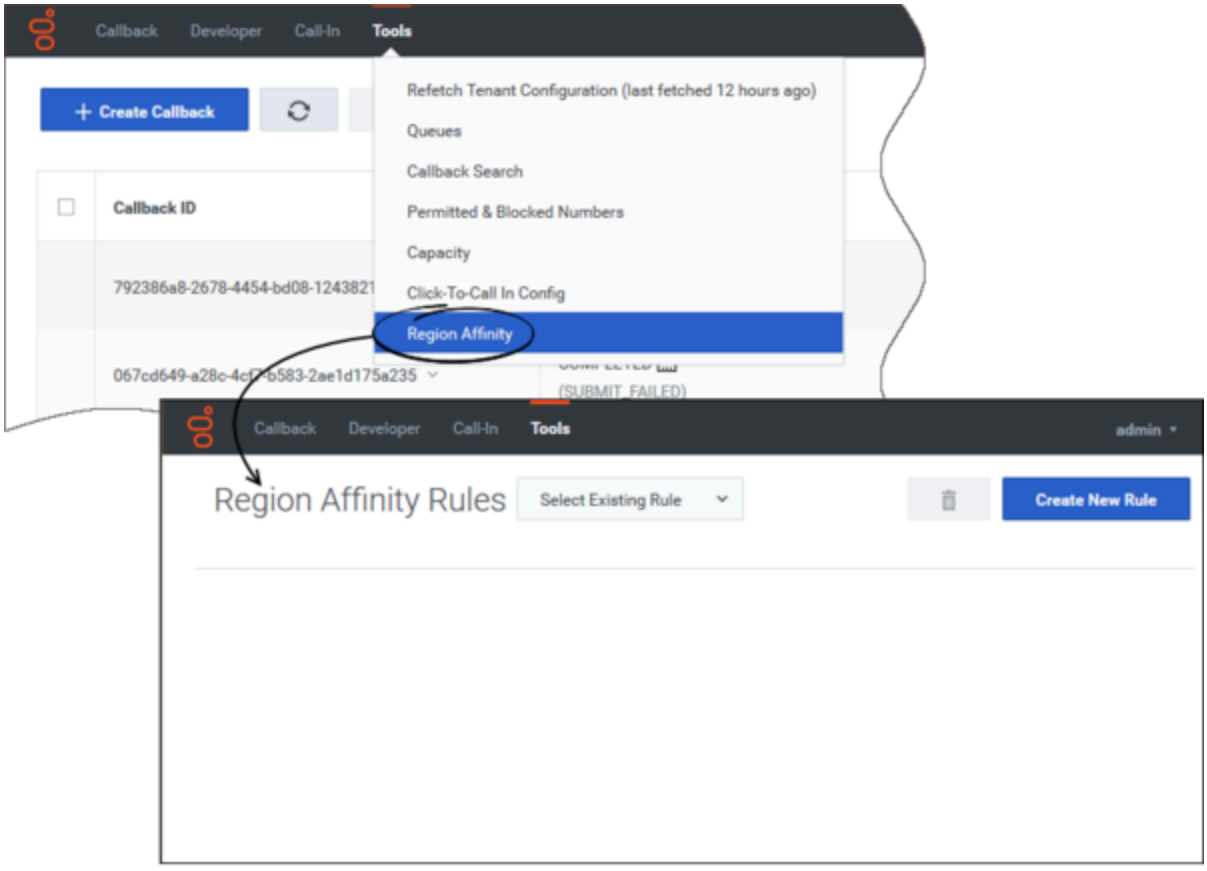
The DNIS numbers must be entered in the .csv file using the E.164 format for telephone numbers. The system will check the values for compliance and prompt you to fix any errors. All rules that apply when entering the DNIS numbers individually also apply when bulk uploading the pool of numbers.

Configuring Region Affinity rules

> **Important**
>
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Configuring Region Affinity rules.

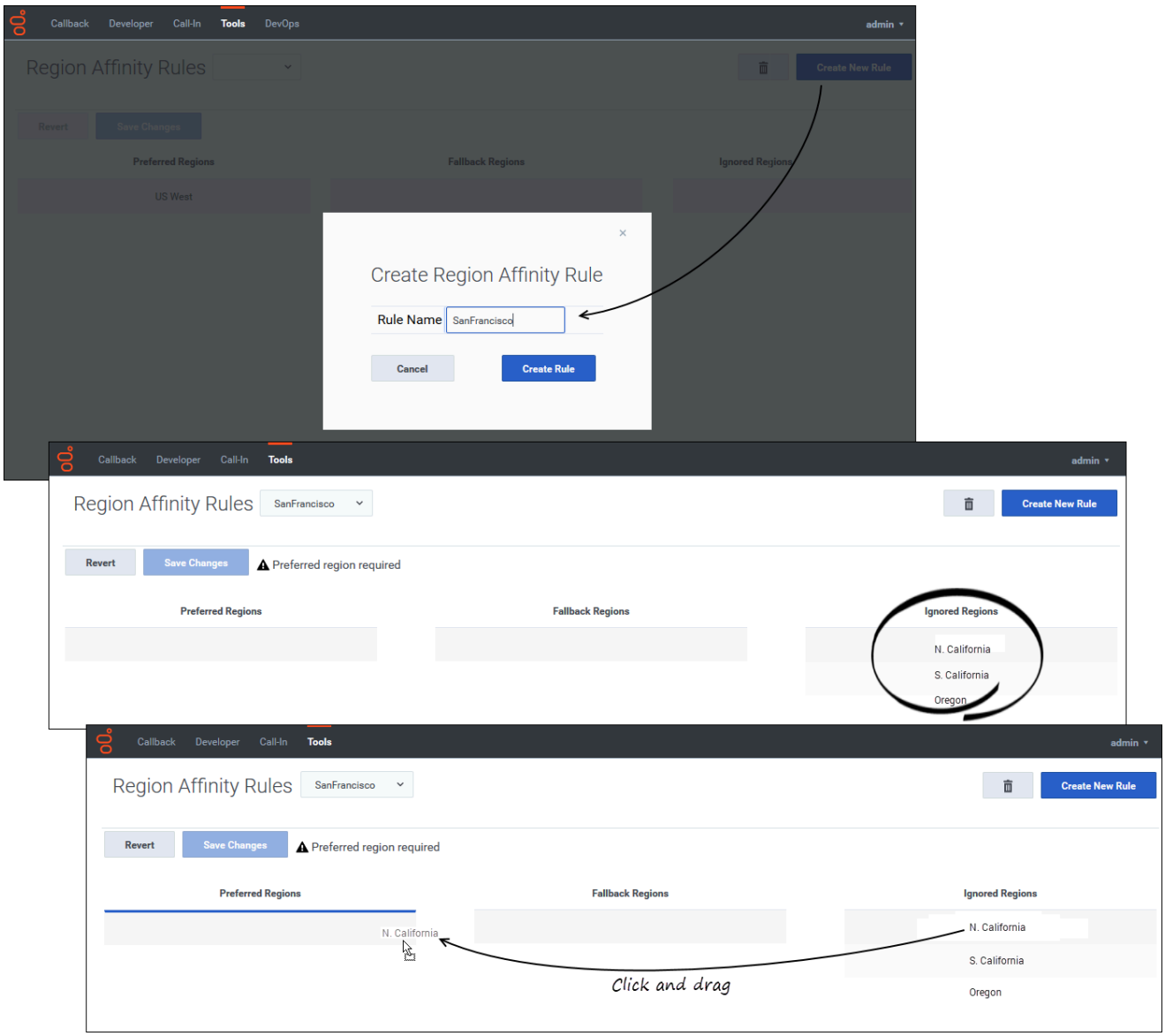
> **Warning**
>
> Region Affinity rules configuration is an advanced feature that requires an understanding of region configuration and processes that are typically managed internally by Genesys personnel. Do not use this feature unless you have consulted with your Genesys representative.

Use the **Tools** > **Region Affinity** page to create Region Affinity rules. To create Region Affinity rules, you must be a member of the Callback Administrator role.



Genesys has regional names for the locations at which there are *nodes* (servers with relevant applications) that manage the interactions for callback virtual queues. You use a Region Affinity rule to specify the regional nodes that you prefer to use and to control the order in which Callback attempts to connect to the available nodes. When there is activity on a queue to which a Region Affinity rule is assigned, Callback tries to connect to a node in the region designated as "preferred" first. Depending on the health status of nodes in the preferred region, Callback might fall back to other regions in the order of configuration until it finds a viable node. For information about how to configure regions in the order in which you want to attempt connection, see Specifying the order in which to connect to regional nodes.

Creating a Region Affinity rule



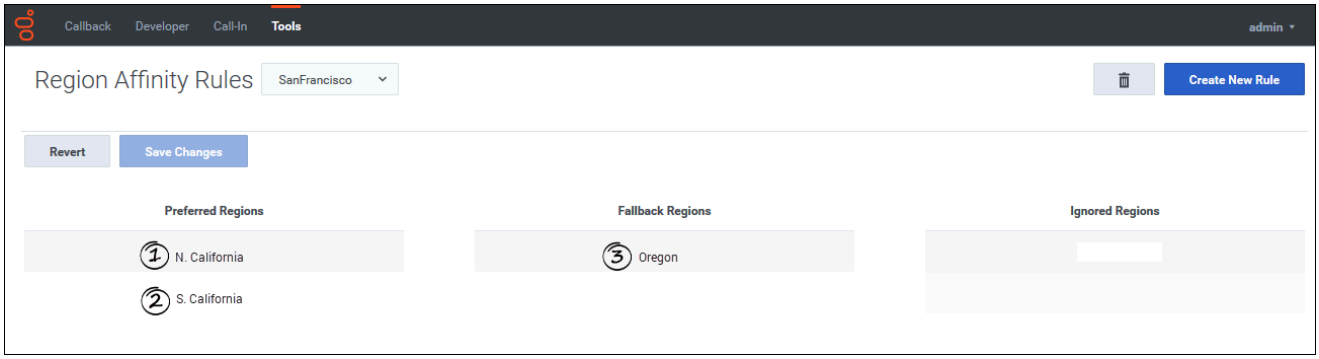
When you create a new Region Affinity rule, all of the regions that are available for your use are initially listed under the **Ignored Regions** column of the rule. To configure the rule, drag region names from the **Ignored Regions** column to the **Preferred Regions** and **Fallback Regions** columns. It is not mandatory to have a region in the **Fallback Regions** column and you can leave some regions in the **Ignored Regions** column. To save a rule, though, there must be at least one region in the **Preferred Regions** column. If there is a region that you want to use, but it does not show up on the **Region Affinity Rules** page in the UI, contact your Genesys Representative.

Specifying the order in which to connect to regional nodes

When handling queue interactions, Callback always tries to connect to a node in the region listed first in the **Preferred Regions** column. On occasion, there might be some reason why Callback cannot connect to that node. When this happens, Callback attempts to connect to a node in another region in the following order:

- If there are additional nodes specified in the **Preferred Regions** column, then Callback tries to connect to a node in the second region listed in the **Preferred Regions** column, and so on until it has attempted connection to every region – in descending order – in the **Preferred Regions** column.
- If Callback cannot connect to any node in the preferred regions, then it tries to connect to a node in the region listed first in the **Fallback Regions** column. If that fails, then Callback tries to connect to a node in the second region listed in the **Fallback Regions** column, and so on until it has attempted connection to every region – in descending order – in the **Fallback Regions** column.

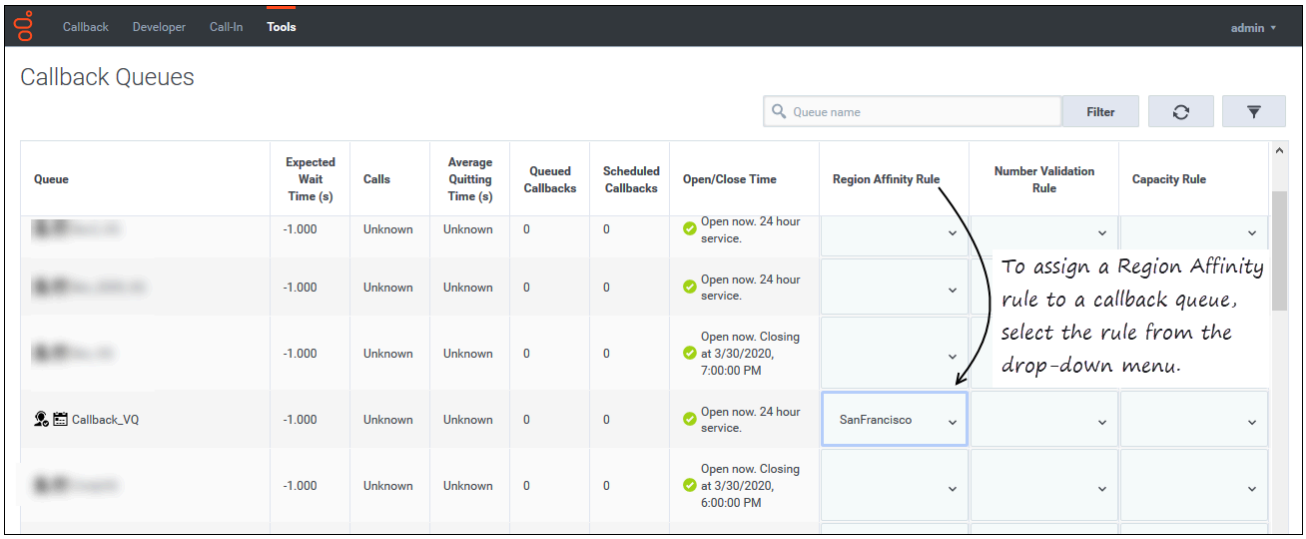
With this in mind, you can configure your Region Affinity rule to specify the precise order in which you want Callback to attempt connection to the other available regions. For example, if the following Region Affinity rule is applied to a callback queue, then Callback will always try to connect to the N. California region first, then the S. California region, and if both of those connections fail, then it tries to connect to the Oregon region.



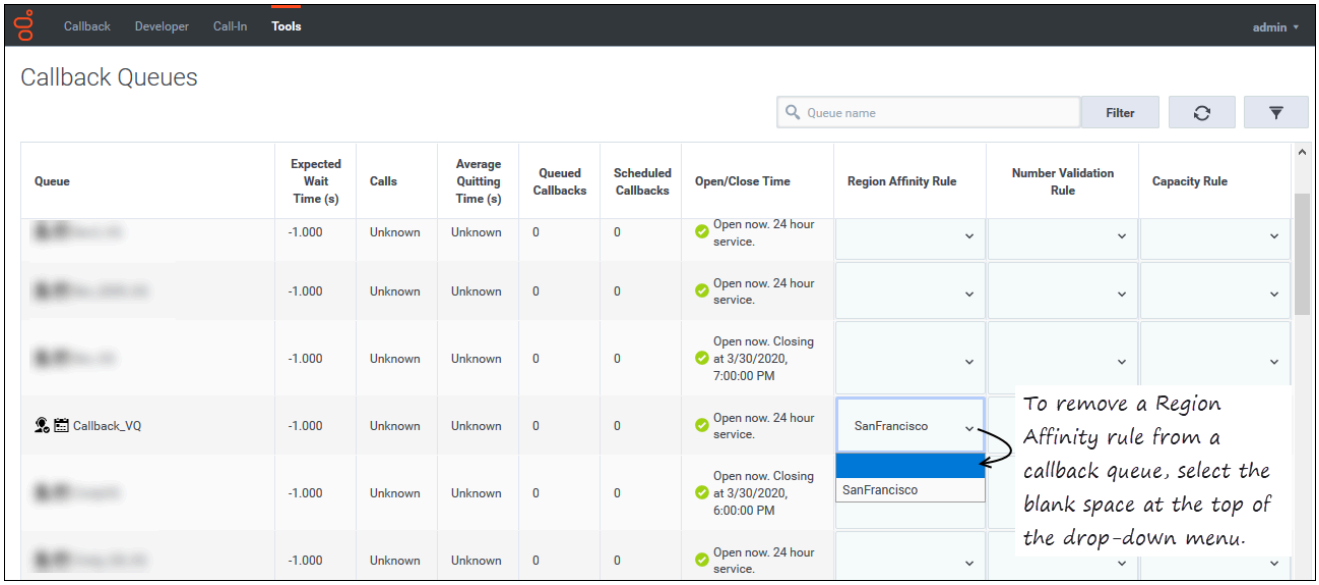
Assigning a Region Affinity rule to a callback queue

> **Tip**
> Access to queues is subject to line-of-business filtering where role-based access control is enabled.

To use a Region Affinity rule to specify the regional node at which to manage interactions on a given queue, you must assign the rule to the queue on the **Callback Queues** tab in the UI. The rule is in effect for that queue as soon as you assign it. You can assign a Region Affinity rule to multiple queues, but you cannot assign more than one rule to a queue.



If you want to delete a Region Affinity rule from the system, you'll first need to remove the rule from all queues to which it is assigned.



Genesys Engage REST APIs and Tutorials for Callback

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage REST APIs and tutorials for Callback.

Depending on your Role, you might work with the APIs that are available for Genesys Engagement Services (GES)/Callback. The documentation for REST APIs is available through the Genesys Engage cloud Developer Center.

The GES/Callback APIs are:

- Callbacks – Create, retrieve, and cancel callbacks.
- Estimated Wait Time – Retrieve Estimated Wait Time.
- Availability – Retrieve time slots for a Callback, matching Office Hours.
- Call In – Request the phone number to call in.
- Queue Status – Retrieve information about a queue's readiness to accept callbacks.
- Statistics – Provides a proxy to the GWS Statistics API. To use the Callback Statistics API, you must first register your GWS credentials in the Callback UI (**Developer** > **Credential Management** > **GWS Credentials** tab).

You can test queries in the Developer Center. However, if you receive error messages while testing your application, use the **Developer** tab to get more information.

This portal also includes Callback Tutorials; select **Engagement** in the tutorial list.

Accessing Callback APIs

To access Callback APIs from Web or Mobile applications, the following items are mandatory:

1. Request an API key from your Genesys Customer Care Representative. You need to send this as the value for `x-api-key` in the header of every request.
2. In your request to Customer Care, specify the list of domains from which you expect your API requests will originate. Genesys uses this list to configure cross-origin resource sharing (CORS) domains for you (this is not currently available as self-service configuration).
3. The base URL for your API endpoints is:
   `https://gapi-<region>.genesyscloud.com/engagement/v3`

You will receive the `<region>` information from Genesys at the same time that you receive your API key.

Consult the Genesys Engage cloud Developer Center for complete endpoint paths.

Cloud intelligent Workload Distribution

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to IWD.

Cloud intelligent Workload Distribution (iWD) allows you to dynamically prioritize the distribution of workitems from external source systems to the employee best suited to handle them.

What does Cloud iWD do?

iWD takes workitems from existing enterprise software applications (such as ERP, BPM, DCM, Salesforce) as well as homegrown legacy systems, analyzes the business context of the workitem—for example, the associated business process, product requested, or value of the customer making the request—and creates a single global task list, sorted on business value, that ensures that the most critical or highest-value workitem are distribued to the right resource at the right time, regardless of media type, system or location.

You'll use the Cloud iWD API to create, update or delete work items which are then categorized, prioritized and routed to employees according to configured rules and assigned routing application logic. A Genesys Designer application handles handles routing/distribution of the workitems, which are then handled by employees using Agent Desktop.

Monitoring of real-time activity is available in iWD Manager and also in Pulse, and historical reporting is also available through Genesys CX Insights.

With iWD, enterprises can effectively manage all customer service resources and business processes across the enterprise, going beyond the walls of the formal contact center and into other areas of the business like branch offices and experts in the back office.

Cloud iWD features and functions

- **Business Context Configuration**—Cloud iWD allows the source system to either:
  - Pre-classify workitems down to the Department and Process level and pass this information on to iWD to match with the appropriate Departments and business Processes in Genesys, against which tasks are then managed and reported on: or;
  - Use Cloud IWD Rules to apply rules that classify the workitems down to the Department and Process level. When using iWD Rules, adding new departments or business processes is a simple matter of configuration by a business user.

- **Service Level Agreement-based Prioritization Rules**—The Cloud iWD service level rules ensure workitems are completed according to the service level agreements (SLA) defined by business users. SLA rules quickly order the workitems from most important to least important, based on business value. Because information related to a workitem can change, iWD automatically re-assesses workitems throughout their lifecycle, ensuring the most important are at the top of the global task list.
- **Workitem Management**—Operations Managers and Team Leads with appropriate security permissions can:
  - View captured workitems
  - Hold, resume, and cancel workitems
  - Modify workitem attributes
  - Manage prioritization schemas

    The same workitem management capabilities (except managing prioritization schemas) are also exposed to source systems via the iWD API.
- **Business Insights**—Cloud iWD offers a set of workitem-based statistics providing insight into business performance and comparisons against key performance indicators configured in iWD by business users. Business insights are available for the current day, and intraday historically.

Set up iWD

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to IWD Setup.

iWD takes workitems from existing enterprise software applications and creates a single global task list, sorted on business value, that ensures that the most critical or highest-value workitems are distributed to the right resource at the right time, regardless of media type, system or location.

You'll use the Cloud iWD API to create, update or delete workitems which are then classified, prioritized and routed to employees according to configured rules and assigned routing application logic.

A Genesys Designer application handles routing/distribution of the workitems, which are then handled by employees using Agent Desktop.

Reporting is also available through Genesys CX Insights.

> **Important**
> You should be able to complete most of the steps on your own, but sometimes you'll need help from Genesys. We'll let you know when a step requires extra help from a Genesys representative.

High-level process

At a high level, you'll need to complete the following steps to implement Cloud iWD:

1. Define agents and agent groups.
2. Configure agent desktop options for your contact center.
3. Configure Role-Based Access Control (RBAC) for all users of iWD Manager.
4. Define a routing application in Designer to handle incoming workitems to iWD.
5. Use iWD Manager to:
    1. Create Endpoints that correspond to the target endpoints created in your Designer routing application.

2. Create Categories that correspond to specific departments or business processes and their hierarchies (business context), associate each with an Endpoint and create a set of Service Level Agreement-based rules for how workitems in each Category are to be handled.
3. Create Prioritization schemas that control the logic and speed with which workitems are (re-)prioritized both before and after their Service Level Agreement, and associate the schemas to Categories.
4. Ask your Genesys team to configure other global iWD settings, including setting up notifications back to the source system by configuring a webhook integration. Click here for more detail on webhooks.

6. Integrate your source systems to use the Cloud iWD API to submit workitems to Genesys (see Integrating source systems below).
7. Monitor the activity of your contact center as iWD manages the global queue of workitems.
8. Manage individual workitems or groups of them while they are being queued and distributed by iWD.
9. Report on historical contact center activity using Genesys CX Insights. Report on realtime activity through the iWD Manager Monitoring features on the **Summary** tab and through Pulse.

Integrating source systems

Create an application to submit workitems to Cloud IWD

A third-party application such as Salesforce or Netsuite can interact with Genesys Engage cloud IWD through the Cloud IWD API.

You'll need to develop an adapter for your third-party application that can create and stop workitems by using the Cloud IWD API. It should also update information about a workitem, if needed, and be capable of retrieving it, or receiving messages about it if webhooks are configured to communicate back to source systems. (Webhooks must be configured by Genesys.)

Add a webhook URL for each source system in IWD Manager

You'll need one webhook URL per source system to be configured so that your source system can receive messages about workitems back from Cloud IWD. You'll need to work with your Genesys representative to configure this.

Starting/updating/stopping a workitem/interaction

The typical workflow is as follows:

1. The application creates a workitem through the IWD API.
2. The application updates the workitem through the IWD API (if needed).

3. The application stops the workitem.

There are two ways for the application to stop an interaction:

- Implicit (recommended) — stop by invoking Mark Done. For this method, complete the following steps:
  1. Set up Agent Desktop to use the Service Client API.
  2. Disable the Mark Done button in Agent Desktop (as mentioned earlier in Configuring Agent Desktop) and instead invoke Mark Done.
  3. Set the Disposition Code
- Explicit — stop by using the IWD API. This requires additional configuration from your Genesys representative.

A sample application that demonstrates using the IWD API is also available through your Genesys representative.

Roles and privileges for iWD Manager users

> **Important**
>
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to IWD Roles and Privileges.

Roles and privileges will be configured initially by Genesys Professional Services. The table below show the roles and privileges they will create for iWD users.

Privilege name	Roles			
	iWD Business User	iWD Supervisor	iWD User	Notes
Workitems (**Workitems** tab in iWD Manager)				
iWD.Workitems.canView	X	X	X	Requires iWD.Workitems.Search.canExecute privilege.
iWD.Workitems.canRead	X	X	X	
iWD.Workitems.canDelete	X	X		
iWD.Workitems.canUpdate	X	X		
iWD.Workitems.canCreate	X			
iWD.Workitems.BulkActions.canExecute	X			
iWD.Workitems.Search.canExecute	X	X	X	
iWD.Workitems.Search.canExecute	X	X	X	
iWD.Workitems.Export.canExecute	X	X		
iWD.Workitems.Hold.canExecute	X	X		
iWD.Workitems.Cancel.canExecute	X	X		
iWD.Workitems.Resume.canExecute	X	X		
iWD.Workitems.Restart.canExecute	X	X		
Dashboard (**Summary** tab)				
iWD.Dashboard.canView	X	X	X	Requires iWD.Workitems.Search.canExecute privilege.
Settings (**Settings** tab)				
iWD.Settings.canView	X	X	X	
Settings/Categories				
iWD.Categories.canView	X	X	X	Requires iWD.Categories.canRead

Privilege name	Roles			
				privilege.
iWD.Categories.canRead	X	X	X	
iWD.Categories.canUpdate	X			
Settings/Prioritization				
iWD.Prioritization.canView	X	X	X	Requires iWD.Prioritization.canRead privilege.
iWD.Prioritization.canRead	X	X	X	
iWD.Prioritization.canUpdate	X			
Settings/Endpoints				
iWD.Endpoints.canView	X	X	X	Requires iWD.Endpoints.canRead privilege.
iWD.Endpoints.canRead	X	X	X	
iWD.Endpoints.canUpdate	X			

Test IWD provisioning

> **Important**
>
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Test IWD Provisioning.

You can test your iWD provisioning setup by using this procedure in conjunction with the supplied CIWD Designer sample application.

1. Create an interaction with this cURL command that will set the workitem's activation time to three minutes in the future.

```
curl --location --request POST 'https://<iwd-host>/iwd/v3/items' \
 --header 'Content-Type: application/json' \
 --header 'x-api-key: <iwd x-api-key>' \
 --data-raw '{
 "external_id":"e21c3332-eecd-4f3b-8a7f-c90",
 "FirstName":"Leonardo123",
 "LastName":"Doe",
 "EmailAddress":"Leonardo123@Doe.com",
 "PhoneNumber": "(925)555-1236499",
 "activate":"2020-02-18T11:35:03.077Z"    //Sample timestamp. Can be set to 3 minutes ahead of current time
 }'
```

2. Check that the interaction appears in real time reporting by creating iWD Queue Activity and iWD Agent Activity widgets based on the Genesys-supplied Pulse widget templates.
3. Check that the interaction appears as expected in iWD Manager.
    1. In the iWD Manager **Workitems** tab, check Search, filter and modify workitems.
    2. In the iWD Manager **Summary** tab, check Category-based graphs and statistics.
    3. In the iWD Manager **Settings** tab, check the configuration of Category tree, Priority and Endpoints.
4. Check workitem routing to an agent.
5. Login workspace employee #1 with the following skill profile:
    - `GSYS_skill_1 > 5`
    - `GSYS_skill_2 > 2`
    - `GSYS_skill_3 > 2`
6. Make employee #1 ready.
7. Three minutes after receipt, the workitem should be routed to employee #1.
8. Open the workitem.

9. Disposition the workitem.
10. View the transfer in real time reporting by using the iWD Queue Activity and iWD Agent Activity templates described above.
11. Wait for 15 minutes.
12. Check the Genesys CX Insights for iWD reports.

Integrate with Genesys Workforce Management (WFM)

> **Important**
>
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

Overview

You can integrate iWD with Genesys Workforce Management (WFM) to enable WFM capabilities to improve management of your contact center workforce. To enable this functionality, three metrics are provided to WFM for each activity every 15 minutes:

- **Interaction Volume** (also known as "new")—The total number of workitems that have been placed in Cloud iWD categories corresponding to the activity in the last 15-minute time interval.
- **Average Handle Time (AHT)**—Calculated as the total time agents spent working on workitems divided by the total number of workitems processed.
- **Actual Queue** (also known as "backlog")—The number of pending interactions (not completed yet) at the end of the last 15-minute time interval in all Cloud iWD categories corresponding to the activity

Interaction Volume and **Actual Queue** are provided by iWD Data Mart. **Average Handle Time** is read by WFM from Stat Server.

> **Important**
>
> - Do not specify Stat Server metrics for **Interaction Volume** and **Actual Queue** during iWD Activities configuration in WFM. This will cause values from Stat Server to override metrics obtained from iWD Data Mart and WFM will show incorrect numbers.
> - If Data Mart fails to send data to WFM, it will try to send metrics for all missed intervals during the next run.

Limitations

1. Only 1:1 mapping between iWD categories and WFM activities is supported.
2. iWD categories and WFM activities must have identical names.
3. Activities must have unique names—duplicate names are not supported. No data is provided to WFM in this case.
4. Tenant provisioning and configuration are done manually using Configuration Manager.
5. Exactly two levels of category tree are matched to WFM activity.
6. To match a one-level category activity, you must use **|Unknown** as the second level. For example, to match category Default activity **Default|Unknown** must be created.
7. Third level and lower cannot be used in activity-to-category mapping.
8. If a category has second-level entries, the first-level entry cannot be matched to an activity. For example, if **WFM|WFM_subcategory1** and **WFM|WFM_subcategory2** are configured, **WFM|Unknown** activity cannot be used.
9. If a workitem is reclassified after it was reported to WFM for the first time, it will not be reported as **PERF_ITEM_ACT_IV** but will be reported in **PERF_ITEM_ACT_QUEUE** on subsequent interval.

> **Important**
>
> Take into account the fact that iWD Data Mart is run every 15 minutes, which does not necessarily match WFM intervals' boundaries. So there could be delays in data becoming visible in WFM.

Configuration in WFM

1. Configure Filters for the selected iWD Categories to make it possible to calculate AHT for the corresponding Activities:
    1. Connect to the tenant Configuration Server.
    2. Go to **Applications > *region* > WFM > SS_WFM > Application Options**.
    3. Add new option under the **Filters** section:
        `<option_name>=PairExists("category_path","<iWD_Category>|<iWD_Subcategory>")`

        For example, `WFMTestSubcat1_Filter=PairExists("category_path", "WFMTestCategory|WFMTestSubcategory1")`.
    4. Save the changes.
    5. Open **Applications > *region* > WFM > SS_WFM_B > Application Options**.
    6. Repeat steps 3 to 5.

2. Create Activities as normal WFM activities taking into account the following guidelines:
    1. Create exactly one Activity of type **Deferred** per category with the following naming pattern:
       iWD_Category/iWD_Subcategory

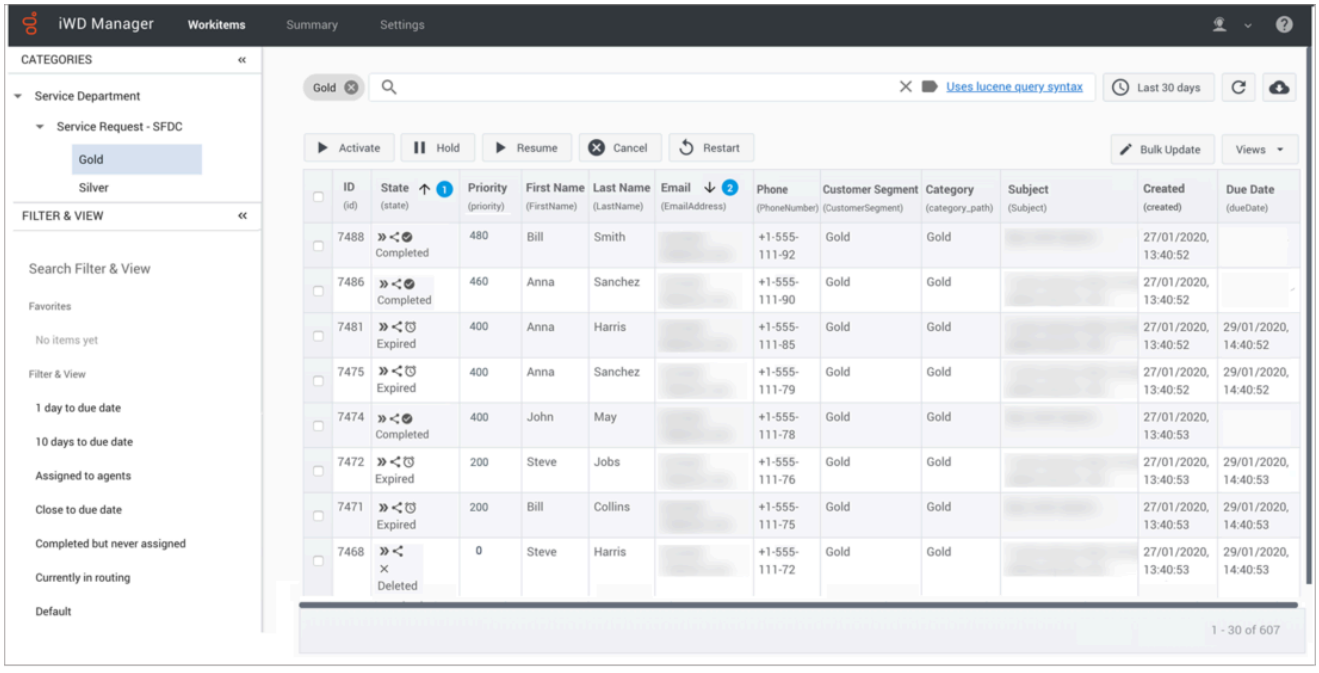
       and with exactly the same name as in iWD.
    2. Create a special Activity named **Default/Unclassified** to handle unclassified workitems. You also need to create a separate Filter for it as described in step 1.
    3. Configure the **Average Handle Time (AHT)** statistic for each Activity:
        1. Open **WFM > Configuration > Activities**.
        2. Open the **Activity > Statistics** tab.
        3. Click **Add**.
        4. Configure:
            1. **Type** = `Handle Time`.
            2. Either **Group of Agents** or **Group of Places** depending on your needs.
            3. **Total Handle Time** = `WFM_Workitem_Interactions_Processing_Time`.
            4. **Interactions Handled** = `WFM_Workitem_Interactions_Processed`.
            5. **Filters** = `<select the filter for the corresponding iWD Category/Subcategory>`.
        5. Click **Apply**.
        6. Save the Activity.

Use iWD Manager

> **Important**
>
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to IWD Manager.

Use iWD Manager to:

- Search, display, manage and update workitems (**Workitems** tab)
- Monitor workitems (**Summary** tab)
- Configure iWD global settings—Genesys use only—(**Settings** tab)



> **Important**
>
> You can only view and/or use features and functions in iWD Manager if you have the appropriate roles and privileges assigned to you. On-screen features that are grayed out are not available. Contact your administrator to update your privileges.

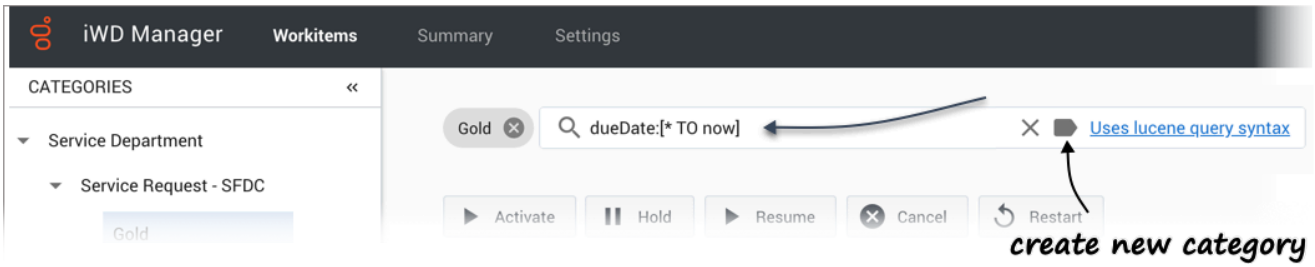
You can:

- Drag and drop columns across the screen.
- Sort multiple columns at once using the up and down sort arrows, which display on mouseover. Columns with sorting applied are indicated with a sequence number.
- Scroll continuously to the bottom of the display list without paging.

Search, display, manage and update workitems

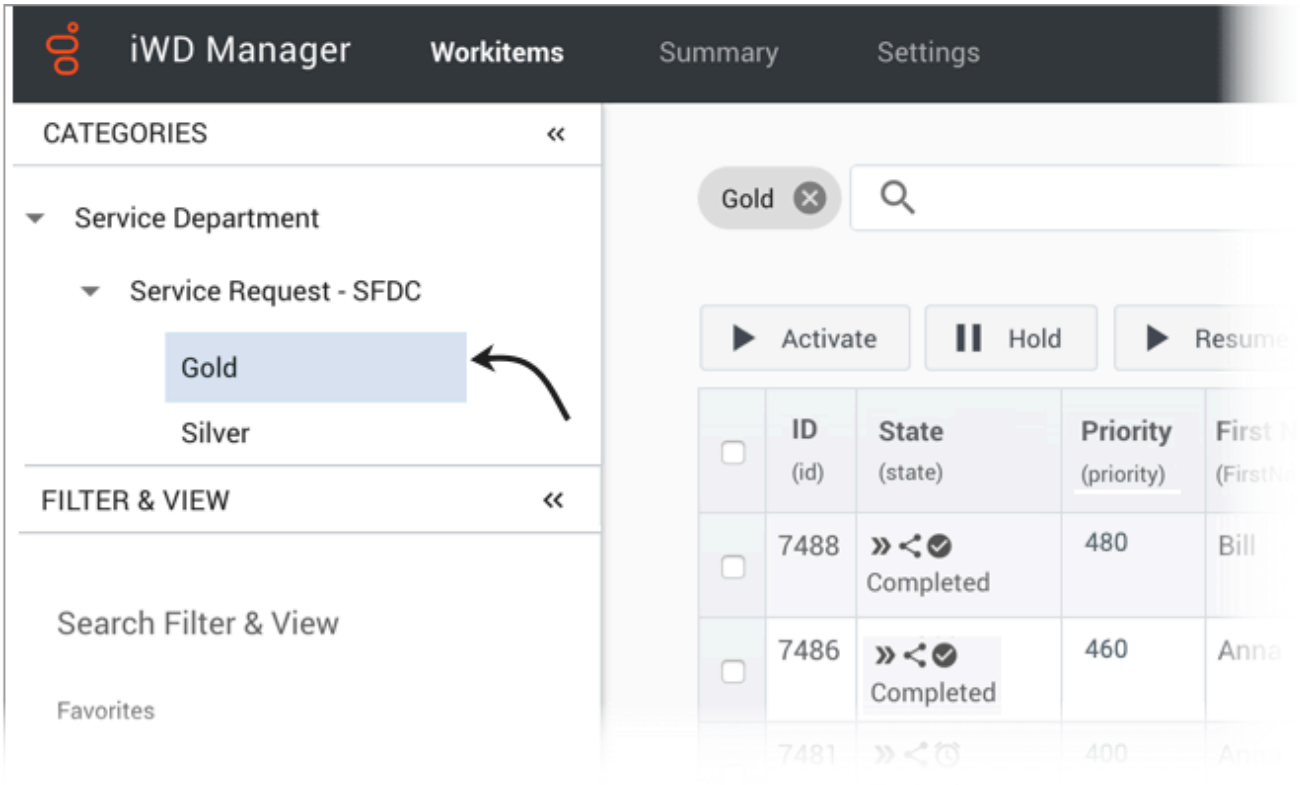
> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Search, display, manage and update workitems.

Search for workitems



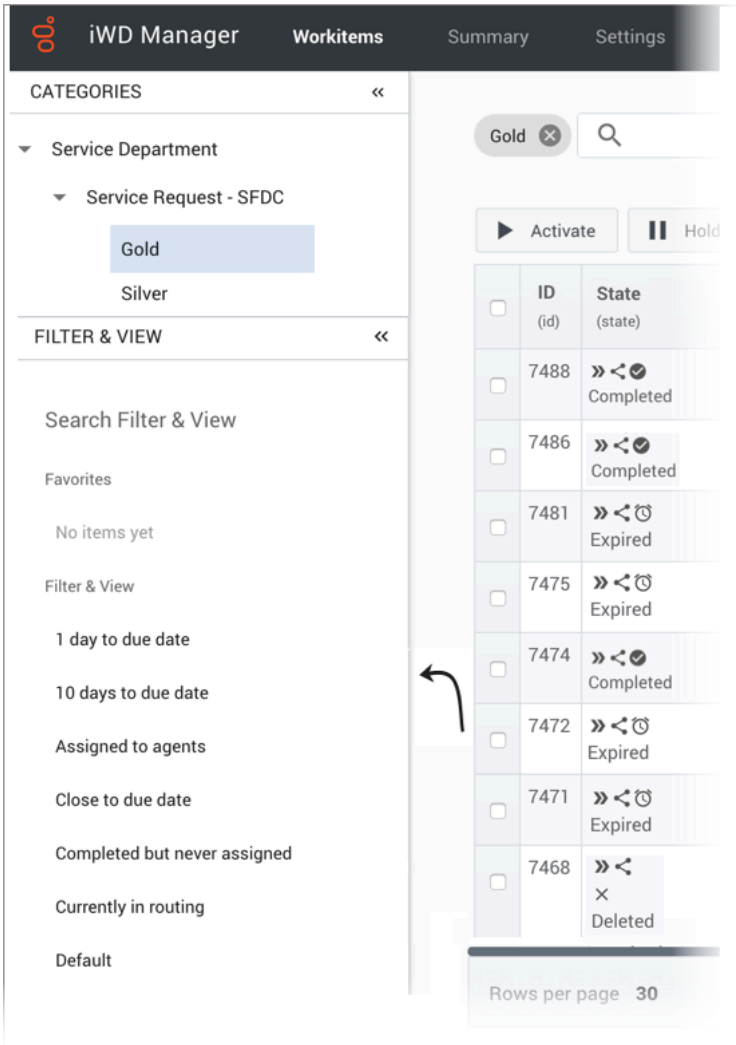
- Search for a **specific workitem** by entering its ID in the **Search** field at the top of the **Workitems** tab and clicking return.
- Search for any **list of workitems** by entering a text string in the **Search** field and clicking return.
- Search the **workitems database** directly by using the Lucene query syntax described here (vendor document).
- Clear the current search citeria by clicking **X** in the **Search** field.
- Create a new Category from the current search criteria by clicking the **Category** button in the **Search** field.

Filter by Category



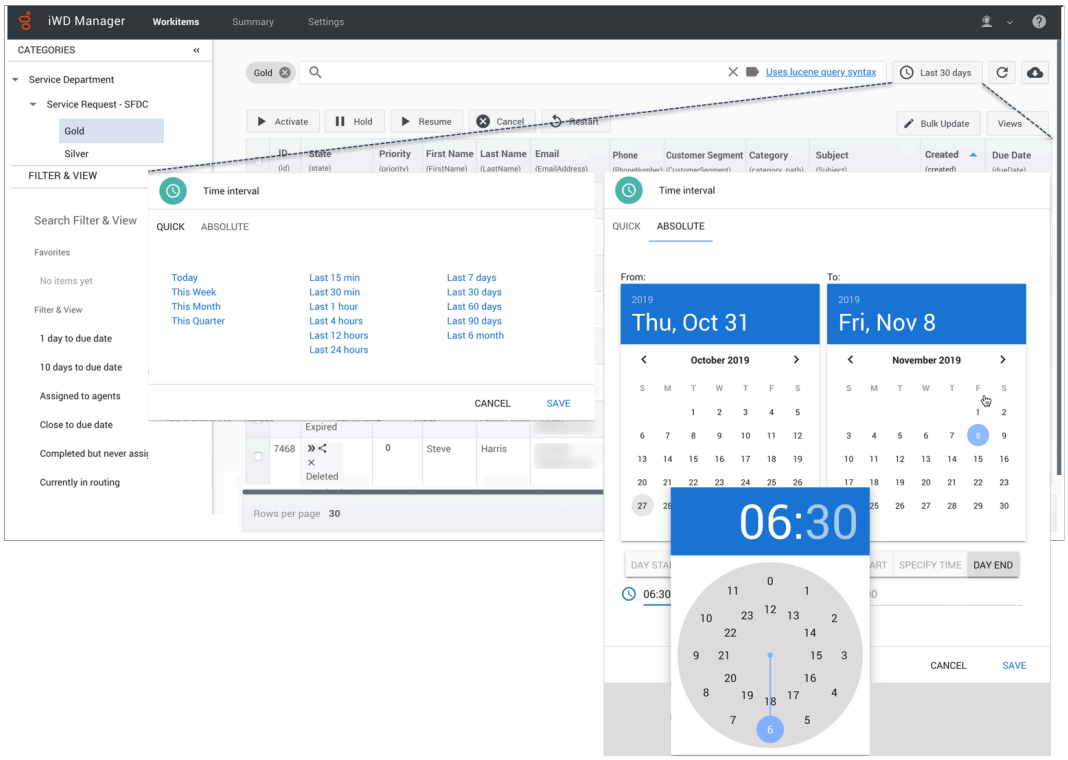
Select a Category on the collapsible **Categories** tab to quickly filter by workitems that match the selected Category or sub-Category only.

Using the Filter & View tab



Use a saved or favorited filter/view on the collapsible **Filter & View** tab to quickly apply filtering and view settings to workitems list. If you have previously selected a Category, any Category defined inside the filter/view overrides the earlier Category selection. If **All Categories** is selected inside the filter/view, any previous selection of Category is cleared.

Search by time interval



Use the time interval button (which shows the currently selected time interval) to display either:

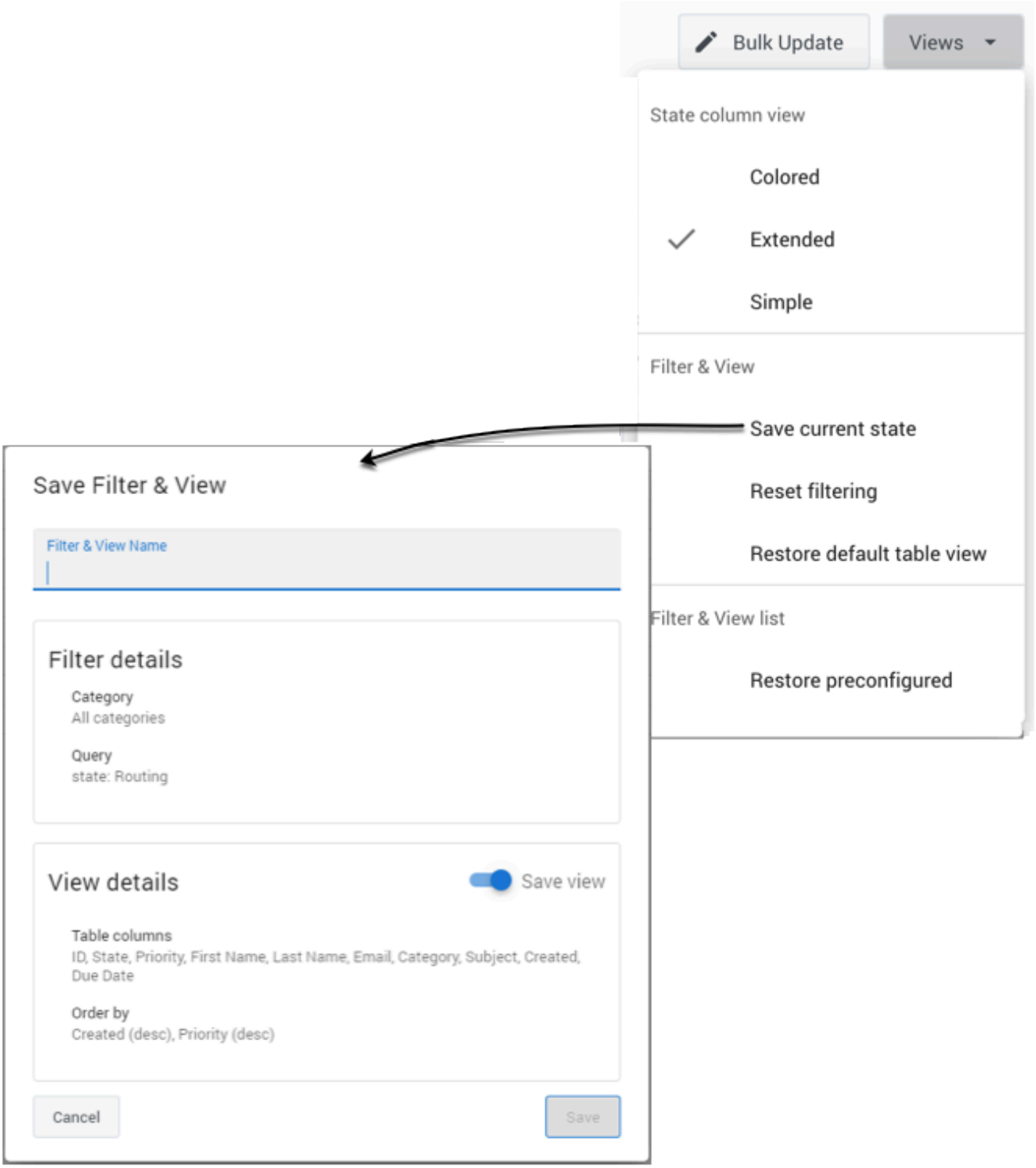
- Quick search using predefined time ranges
- Absolute date/time ranges for customized time intervals

You can click **DAY START** and **DAY END** selectors to select the day start and end times configured for your contact center or you can use the **SPECIFY TIME** selector to open a clock-style drag-and-drop selector in which you can click the displayed time, then change the **From** and **To** hour values. Drag the selector to select a new hour and minute value, then, on the second selector displayed, a new minute value.

> ## Tip
> You must select a minute value, even if it is `00`, to register the selection.

If you have a Category selected, only workitems in that Category are displayed.

Save filters and views

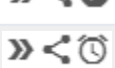


Use the **Views** button to do the following:

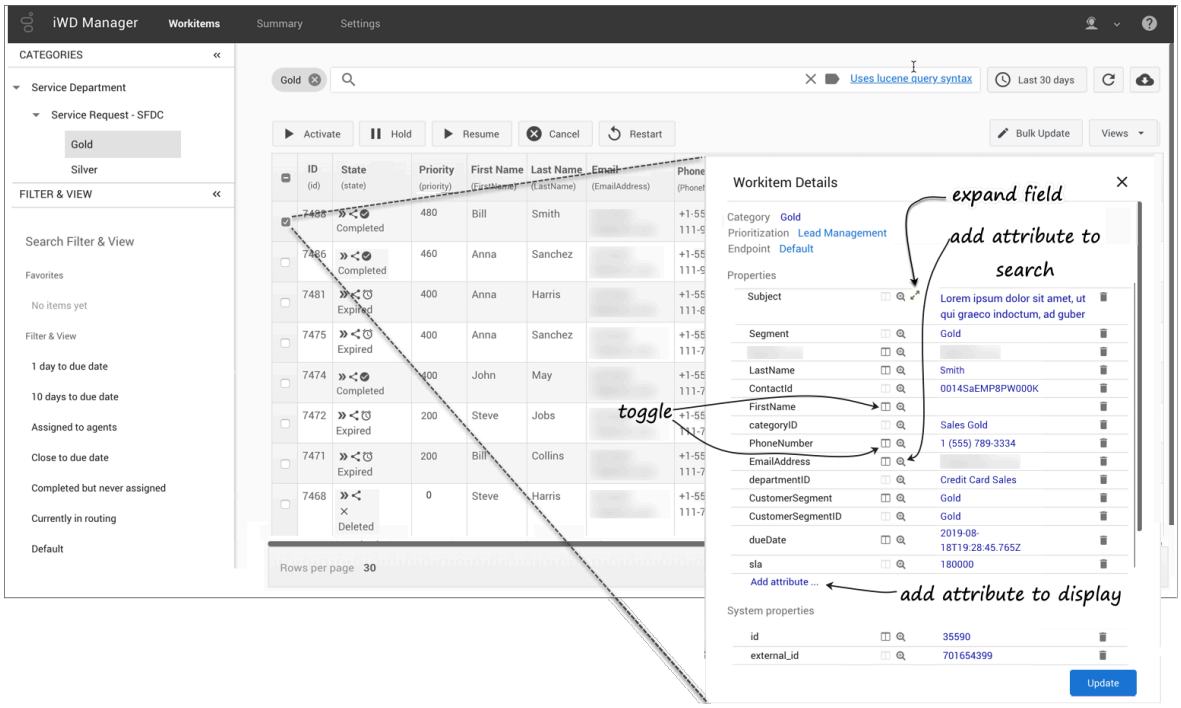
- Select a view for the **State** column.
- Save/reset filters (Category + query) and views (columns + sorting):
  - Save the current filter and/or view for future use (Supervisors, Business users and Administrators only).
  - Reset the filter to clear the filtering details from the page.
  - Reset the table view to the default.
- Restore the list of preconfigured filter and view states.

Extended States

The table below describes the meanings of the extended state symbols.

Extended	Description
	New
	Not yet distributed
	Held
	In routing state, not yet assigned
	Being processed by an agent
	Completed but not assigned to an agent
	Expired but not assigned to an agent
	Completed and has been processed by an agent
	Expired and has been processed by an agent

Display and update workitem attributes



For individual workitems

Select a workitem and/or click one of its attributes to display all its attributes.

Toggle the display of specific fields in the summary display by clicking the symbols indicated.

Expand a field with more than 200 characters by clicking the dynamic expand symbol next to the field label.

To update the attribute values of a workitem, click the value you want to change, add the new value, then click the **Update** button. System properties of the workitem are protected, so you can't update them.

You can add an attribute to the display by clicking **Add attribute...** at the bottom of the first panel of attributes.

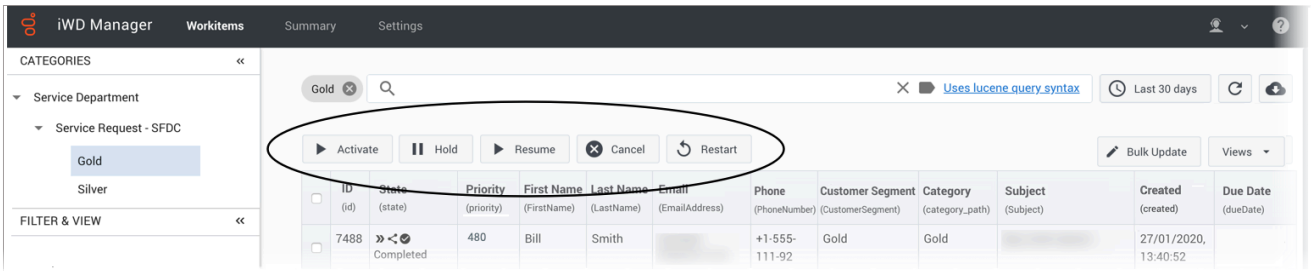
You can also search for other workitems corresponding to a specific attribute value by clicking the search symbol next to the attribute.

For multiple workitems

To make an identical value change to one or more attributes of multiple workitems, select the workitems by checking their checkboxes or by filtering workitems using a search query, click the value you want to change, add the new value(s), then click the **Bulk Update** button.

System properties of the workitem are protected, so you can't update them.

Manage workitems



Use the control buttons at the top of the **Workitems** window to do the following to either individual or multiple workitems:

- Activate workitem(s)—Select a workitem(s) that has been assigned an activation date in the future, and click **Activate** to activate it immediately.
- Hold workitem(s)—Select a workitem(s) and click **Hold**. Held workitems can't be reprioritized or distributed, but you can cancel, restart or resume them.
- Resume workitem(s)—Select a held workitem(s) and click **Resume** to resume processing. You can only resume held workitems.
- Cancel workitem(s)—Select a workitem(s) and click **Cancel** to permanently cancel it. You can't cancel a workitem if it's been completed, canceled, or rejected. But you can cancel a workitem that has already been assigned to an agent. Canceled workitems remain in iWD for an amount of time that's configured in the **Keep completed workitems for** field (by Genesys only) before being deleted. Workitems can only be completed through the source system or the API, so Genesys recommends canceling any of the workitems that you no longer want matched with an employee.
- Restart workitem(s)—Select a workitem(s) to send it back to the beginning of the process, where it will be treated as if it was new.

Any action can be applied to one or many individually selected items (checked in the table) or to all the items that satisfy the current query, including time frame and any applied Category filter.

Workitems that are in the **Default** category have not been segmented by any of the existing Categories. Genesys recommend that administrators monitor these workitems to decide whether a category should be established to segment them.

Monitor workitems

> **Important**
>
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Monitor workitems.

The **Summary** tab in iWD Manager gives you several options for displaying dashboards for monitoring the state of workitems controlled by iWD:

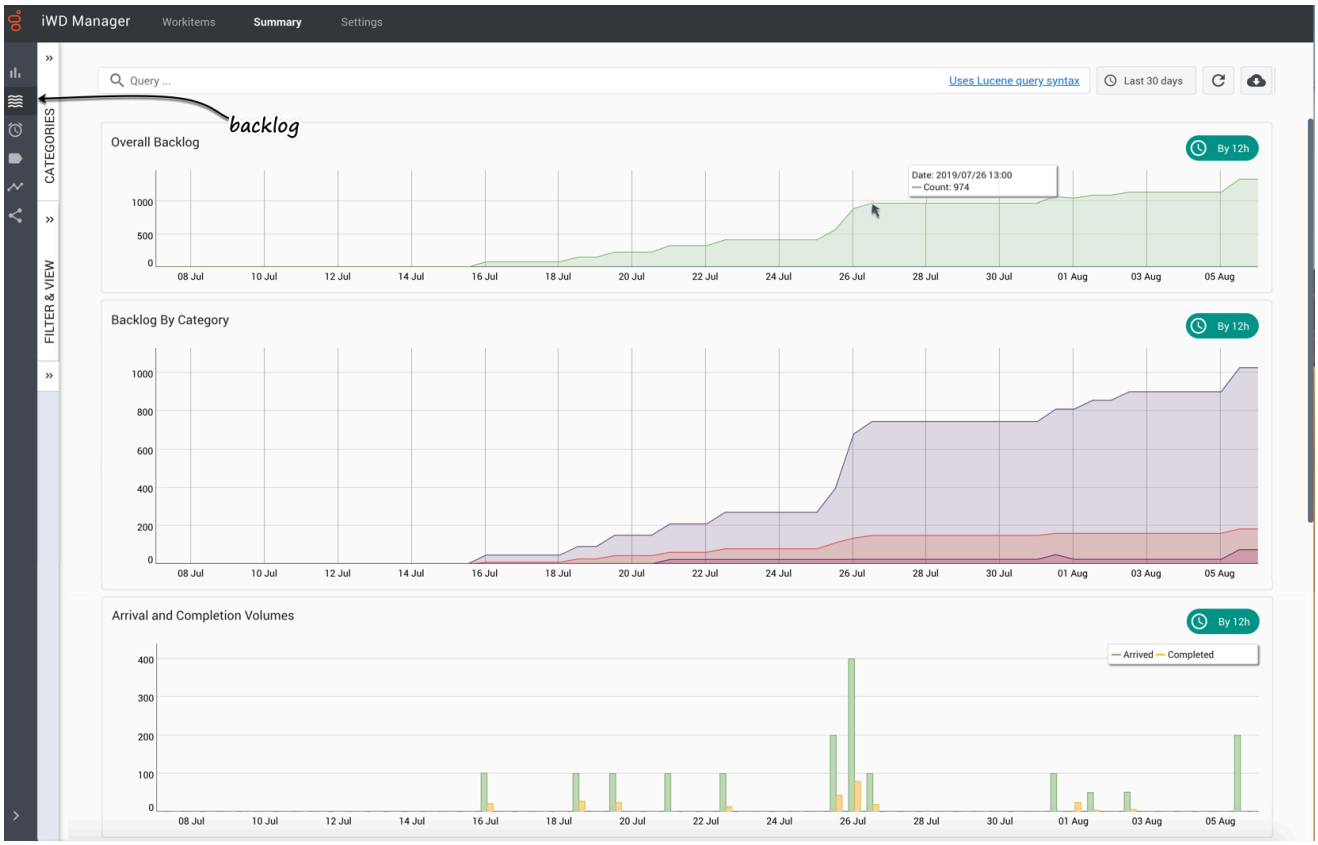
- By backlog
- By volume
- By Service Level Agreement
- By categories
- By timing
- By path

> **Important**
>
> - All the displays reflect the currently selected time frame and the selected Category and query.
> - Where very large numbers of workitems are displayed, scientific notation is used on the axes of display graphs.

Using saved and favorite search filters

Where filters have been created and saved by a Supervisor, Business User or Administrator you can apply them to the display of the summary dashboards, and favorite them. Expand the **Filter & View** panel to see available search filters.

By backlog



Provides an intraday high-level summary of the backlog for the period chosen. This is the starting point for analysis, enabling you to quickly spot trends in KPI and potential bottlenecks in workitems. For example, it can enable you to quickly recognize that the backlog is growing or shrinking. It displays dynamically the overall backlog, the backlog by category, and arrival and completion volumes.

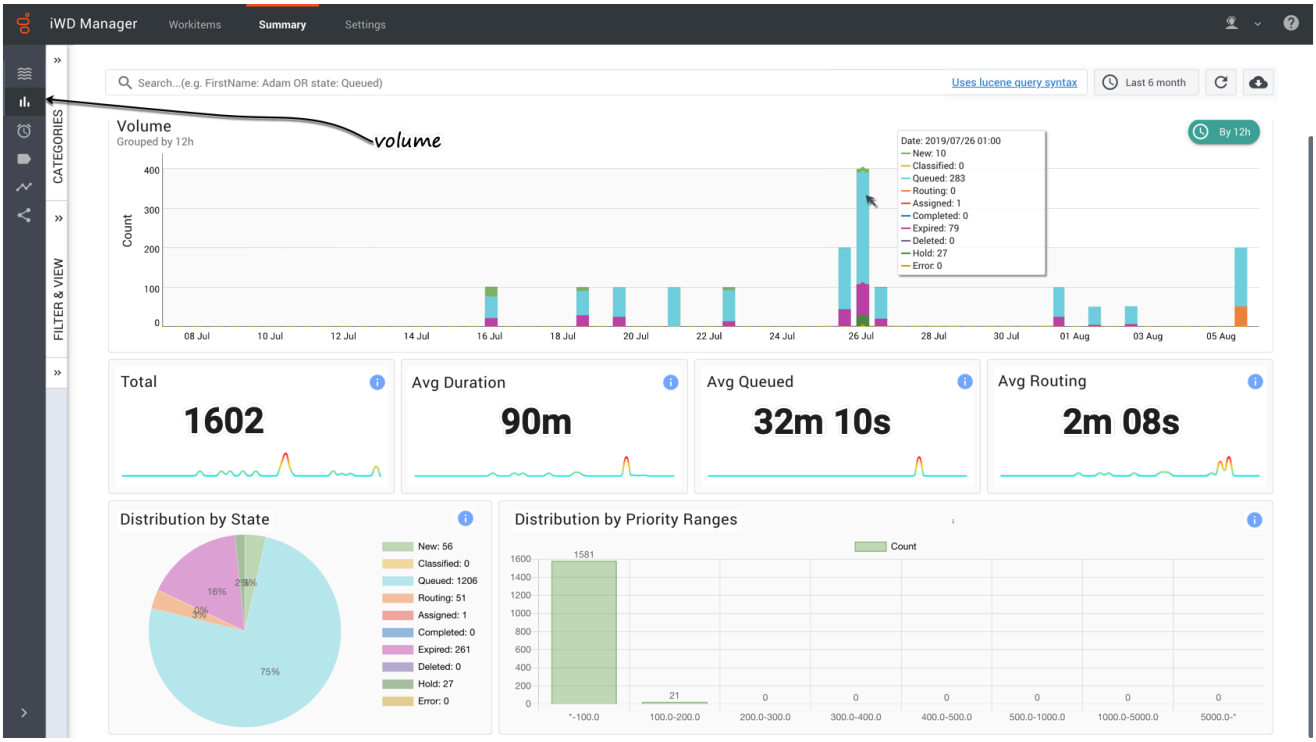
You can:

- Dynamically change the time interval and category for display.
- Dynamically select a saved search filter from the **Filter & View** tab for display.
- Hover over populated areas of the graph to display a detail panel for the date and time selected.
- Use your cursor to drag a box to select a range of columns and drill down to a lower level of detail.

Metric descriptions

- Overall backlog—The overall amount of workitems.
- Backlog by Category—The amount of workitems in distribution by category.
- Arrival and Completion Volumes—The amount of arrived and completed workitems in a time range.

By volume



Displays dynamic histogram and summary totals (total, average duration, average time in queue, average time in routing) for the selected time interval and category, and distributions by state (pie chart) and priority ranges (bar chart). You can:

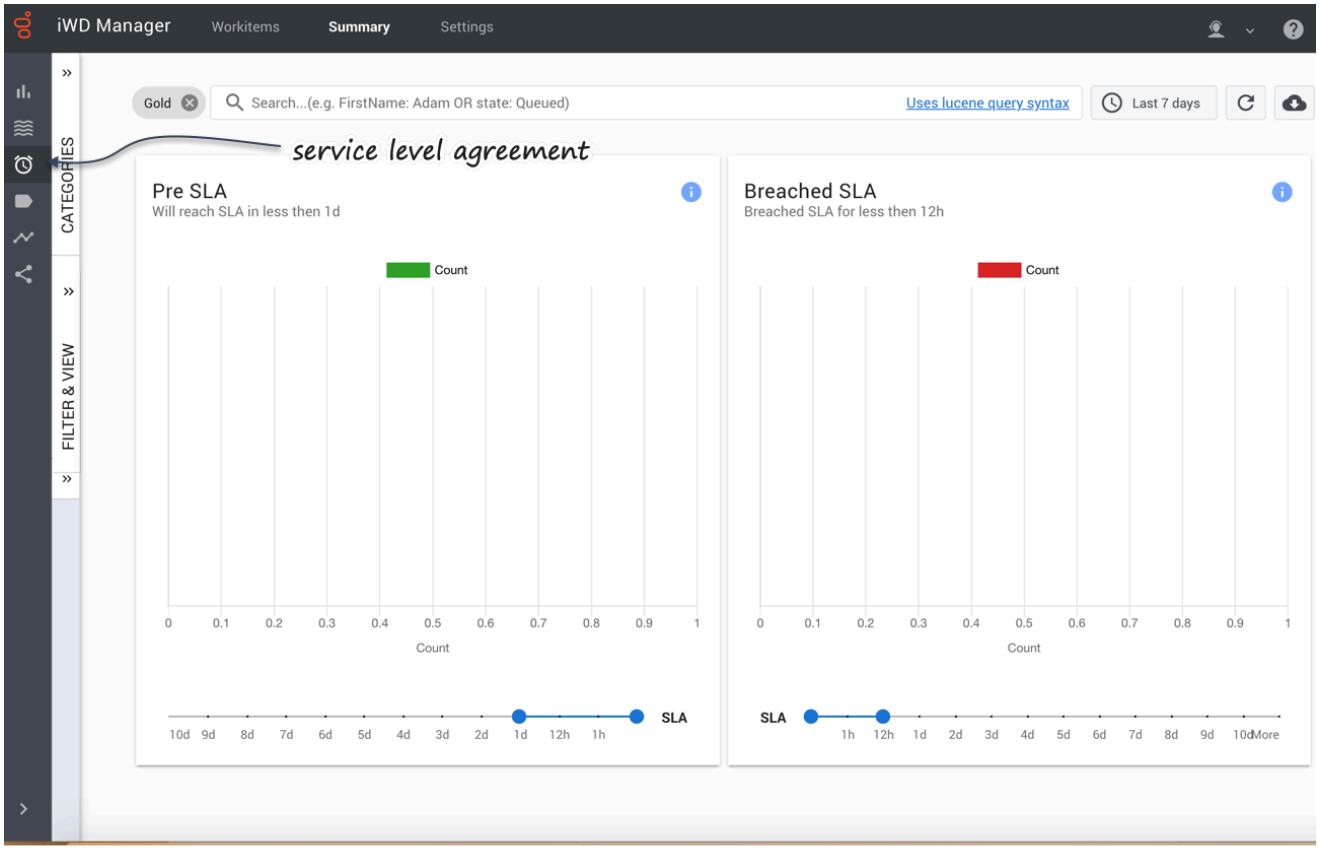
- Dynamically change the time interval and category for display.
- Dynamically select a saved search filter from the **Filter & View** tab for display.
- Hover over points in any graph to display a detail panel for the selected data point.
- Use your cursor to drag a box to select a range of columns and drill down to a lower level of detail.

Metric descriptions

- Volume graph—The total number of new workitems that were submitted to iWD in a time range.
- Total—The total number of new workitems that were submitted to iWD during the reporting interval.
- Average duration—The average amount of time that elapsed before agents completed workitems. This metric includes the time that workitems were backlogged, as well as work time.
- Average Queued—The average amount of time a workitem was waiting in the queue.
- Average Routing—The average amount of time a workitem was waiting to be routed.
- Distribution by state—The distribution of workitems by their state, in percent and quantity.

- Distribution by priority ranges—The number of workitems that have different priorities.

By SLA



Displays dynamically workitem counts for selected pre-SLA intervals, and for workitems that have breached their SLA by selected intervals. You can:

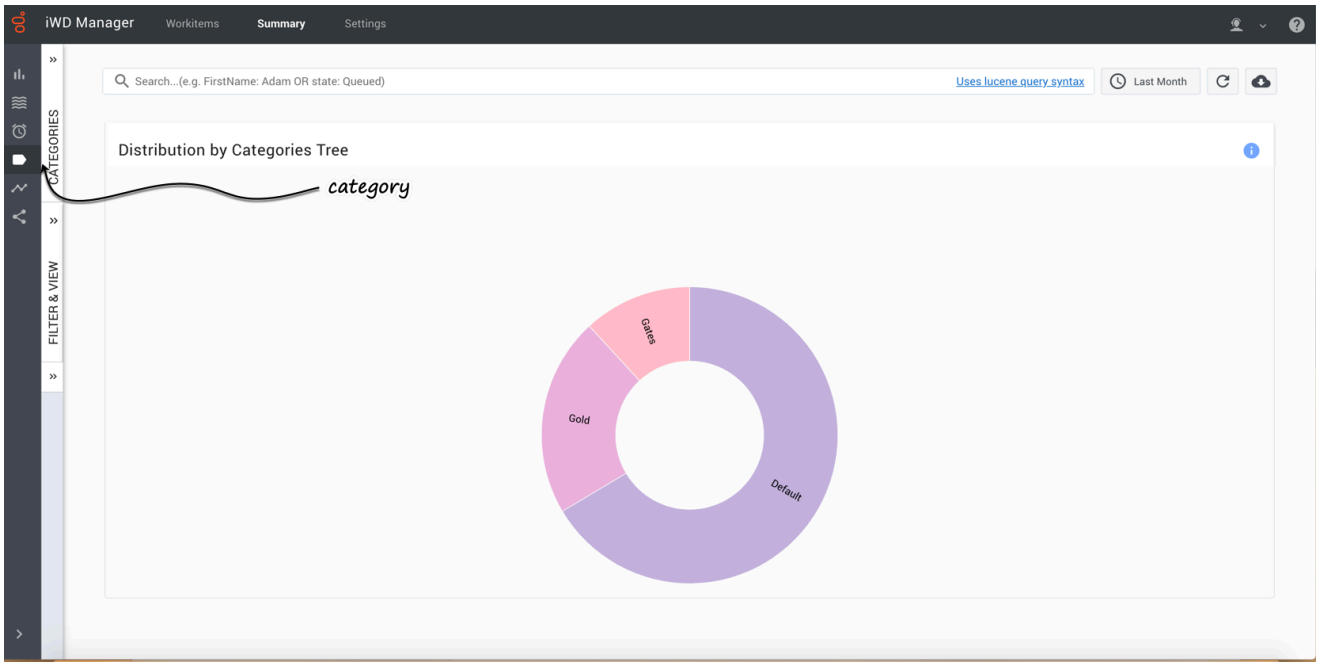
- Dynamically change the time interval and category for display.
- Dynamically select a saved search filter from the **Filter & View** tab for display.
- Hover over populated areas of the graph to display a detail panel for the date and time selected.
- Use the sliders at the foot of the panels to change the time ranges.

Metric descriptions

- Pre SLA item counts—The amount of workitems that should be serviced in a selected time range.

- Breached SLA item—The amount of workitems that should have been serviced but were not, in a selected time range.

By categories



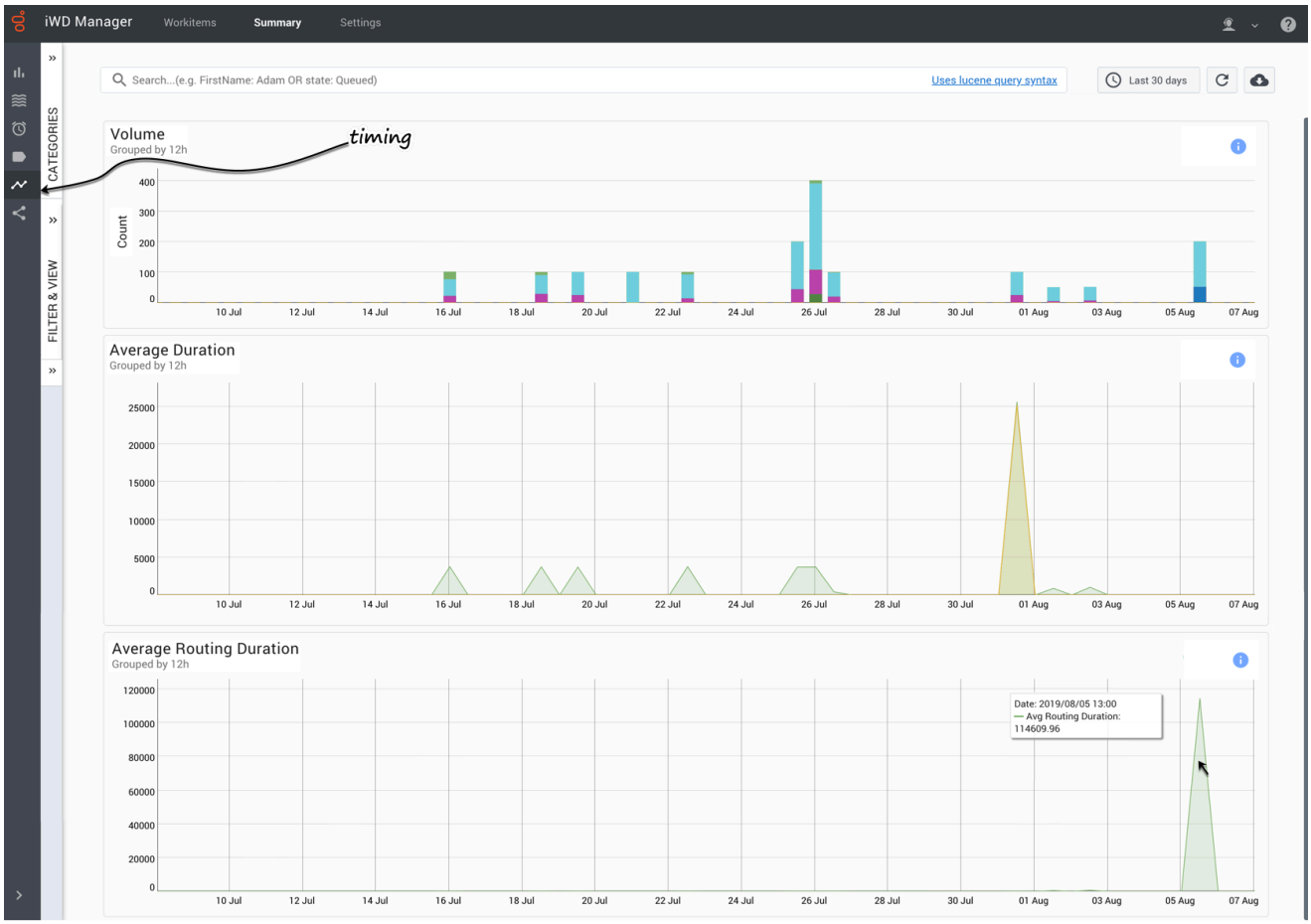
Displays dynamically the distribution of workitems across the available categories for the date and time selected. You can select current or historical time intervals for a rapid visual comparison.

You can also dynamically select a saved search filter from the **Filter & View** tab for display.

Hover over a Category chart to display information about that Category. **Metric description**

- Distribution by categories—Distribution of workitems by categories.

By timing



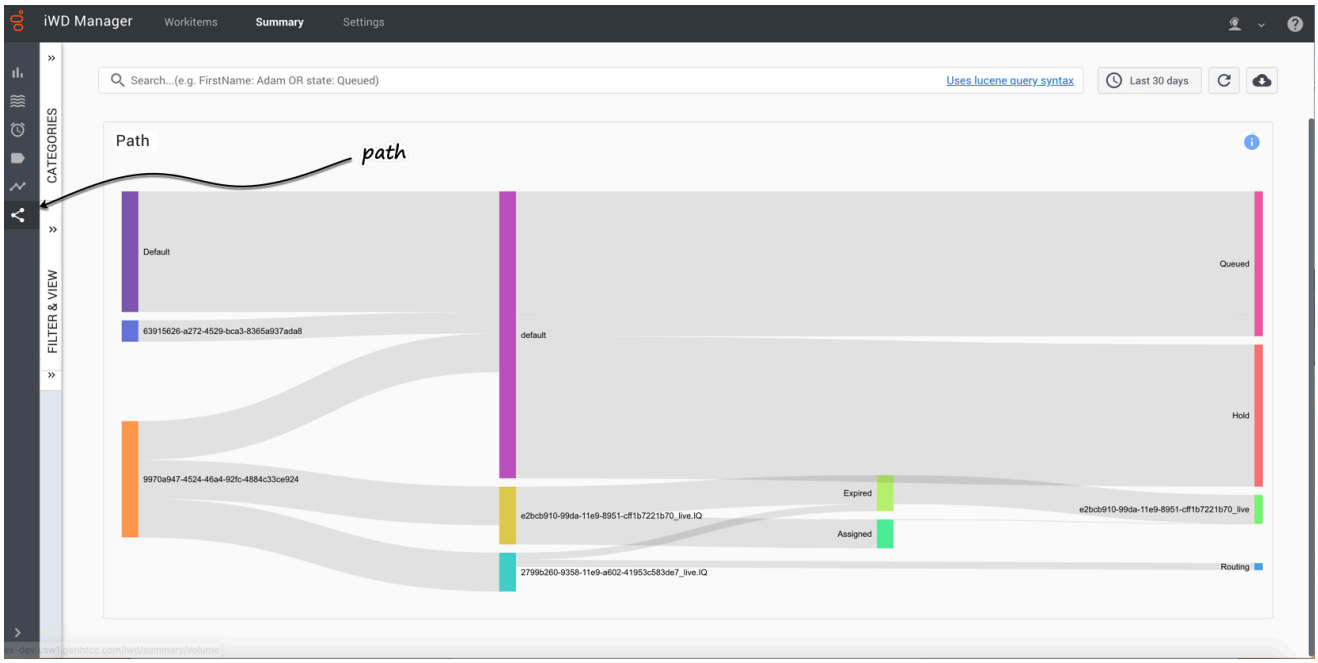
Displays dynamically the volume of transactions, plus average duration times and average routing duration times. You can:

- Dynamically change the time interval and category for display.
- Dynamically select a saved search filter from the **Filter & View** tab for display.
- Hover over populated areas of the graph to display a detail panel for the date and time selected.

Metric description

- Volume—Total number of received workitems in a time frame.
- Average Duration—Average workitem lifetime and time before workitem becomes queued.
- Average Routing Duration—Average time that workitems spent in routing before being routed to the agent.

By path



The Path dashboard enables users to see where the workitem has been segmented, queued in Designer and matched with an employee, through to completion. This dashboard is also useful for identifying workitems that have not been segmented and have been matched through the default path.

Configure iWD

> **Important**
>
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to IWD Settings.

The **Settings** tab of iWD Manager lets administrators do the following:

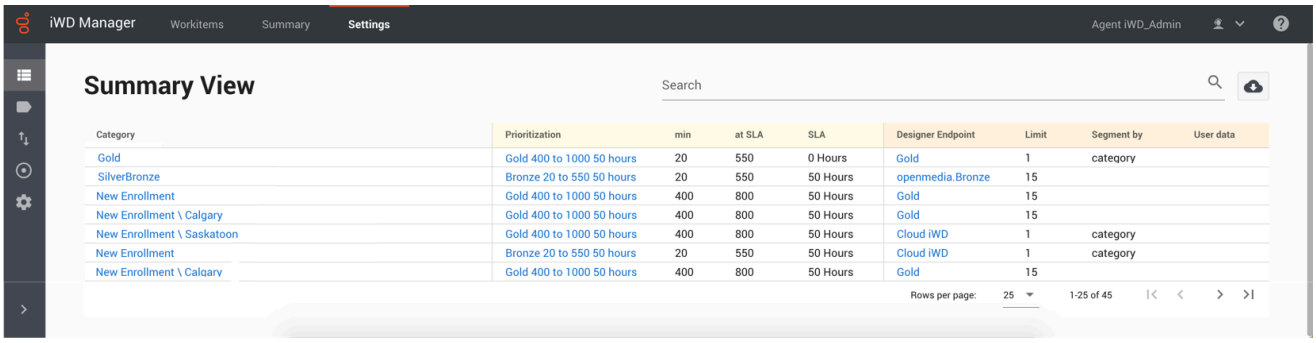
- View Summary details of all categories, prioritization schemas and endpoints. You can also search the displayed items and export displayed items to .CSV format.
- View, edit and create the Categories displayed in the collapsible Categories panel. Categories correspond to business units, lines of business, departments or business processes whose workitems iWD will control.
- View, edit and create different Prioritization schemas that will be assigned to Categories for handling the Prioritization and escalation of the workitems in those Categories.
- View, edit and create Endpoints that can be associated with different Designer applications and assigned to the Categories.

> **Important**
>
> Ask your Genesys team to configure other iWD settings on the Options toolbar, including notifications back to the source system.

View Summary

> **Important**
>
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to View Summary.

The Summary tab in iWD Settings shows a display of all categories, prioritization schemas, endpoints and service level agreements.

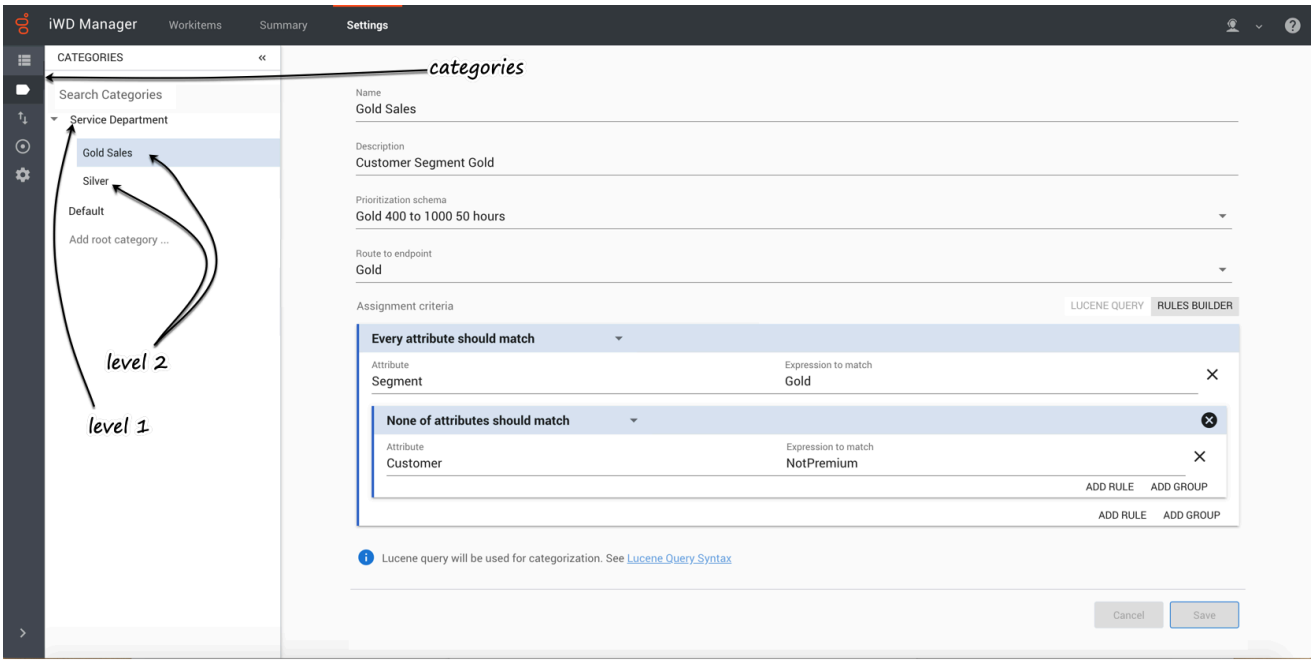


View, edit and create Categories and Rules

> **Important**
>
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to IWD Categories.

Each node in the Category tree corresponds to a business unit, department or process that handles a specific set of workitems for a specific purpose. Each node is associated with a Prioritization schema that controls how workitems are prioritized. Each node also specifies a Designer Endpoint at which its workitems are targeted, and enables you to define rules for how workitems are assigned to it.

View and edit a Category



1. Select the **Settings** tab.
2. From the collapsible **Categories** panel, select the Category you want to view or edit.

> **Important**
> Remember that changes can have large impacts on contact center operation.

The Prioritization schema and the Designer Endpoint associated with this Category are displayed. You can change them if you have the appropriate privileges. Designer applications are loaded on the Designer Endpoint in order to match the segmented workitems with employees.

Category levels and reporting

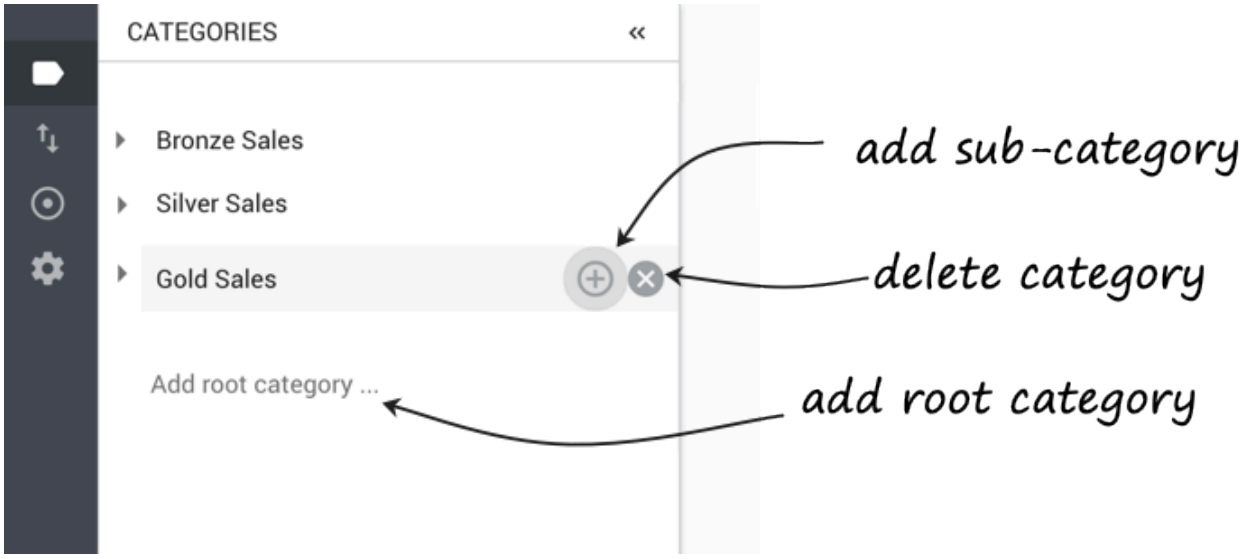
The first level of Category (**Service Department** in the screenshot above) translates to the Department in historical reporting (for example, here).

The second level of Category (**Gold Sales**, **Silver** in the screenshot above) translates to the Process in historical reporting (for example, here).

The third level, and any lower levels, do not translate to historical reporting. These levels add power to the capabilities, but you should establish a clear business reason why more than two levels need to be configured.

The first level filters all workitems that match the Category's criteria. Any sub-levels further filter workitems that match the first-level criteria as well as the sub-level's criteria. In this way you can create a hierarchy that funnels workitems through to lower levels of specificity to enable them to be handled in more specific ways.

Add/delete controls



- To add a new root category, at the bottom of the collapsible **Categories** panel, click **Add root**

category...

- To add a sub-category to an existing category, click the + button to the right of the parent Category's name.
- To delete a Category click the X button to the right of the Category's name.

> **Warning**
> You cannot undo deletion of a Category.

Create a new Category

1. Select the **Settings** tab.
2. Open the collapsible **Categories** panel.
3. Click either **Add root category...** or, to create a new sub-Category, the **+** symbol next to an existing Category .
4. Give the new Category a name. For a sub-Category, you should enter a name that follows the naming convention for the root or parent Category, for example:
   [source system] [department][process] high-level criterion
5. Give the Category a business-friendly description, to include the rule intention if possible (see the examples below).
6. From the drop-down list, select a Prioritization schema for this Category.
7. From the **Route to Endpoint** drop-down list, select a Designer Endpoint for this Category.
8. In the **Assignment criteria** section, create a rule or group of rules, or a Lucene query, that will determine which workitems are assigned to this Category. The examples below illustrate some relatively simple rules and rule groups.

> **Tip**
> You can use Elasticsearch query syntax in describing Categories and creating rules and rule groups.

9. Click **Save** when you're done.

Rules

The **Assignment criteria** panel shows which rules and rule groups have been set up for this Category (with the **RULES BUILDER** button selected). Rules determine which workitems are assigned to this Category. Rule groups are bundles of more than one rule. Rule groups can be coupled

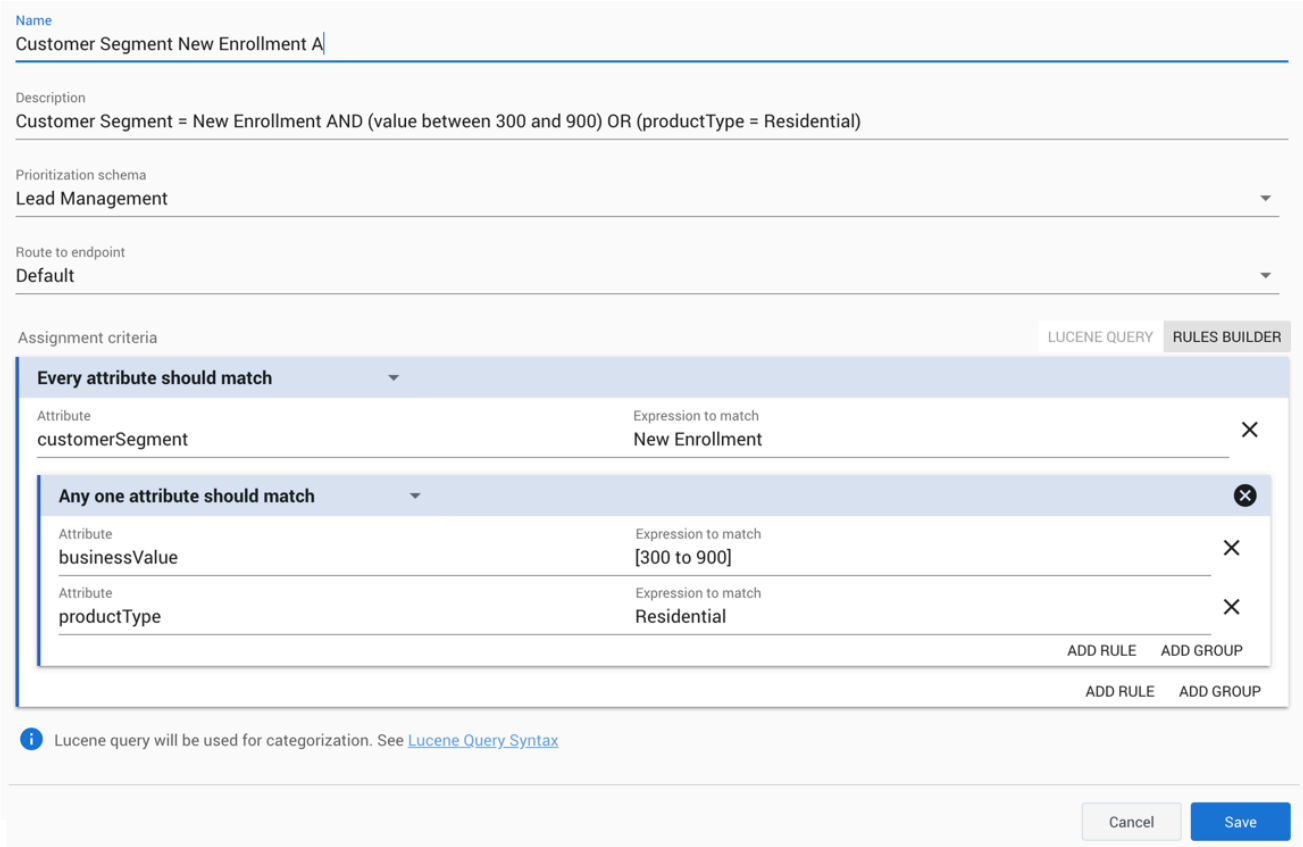
with standalone rules. (You can only nest rule groups down to the third level.) Every Category must have at least one rule configured. See the sample rules below.

Lucene queries

You can configure a Category by selecting the Lucene query option and entering Lucene query syntax in the field that displays:

Workitems retrieved by the Lucene search will be assigned to this Category.

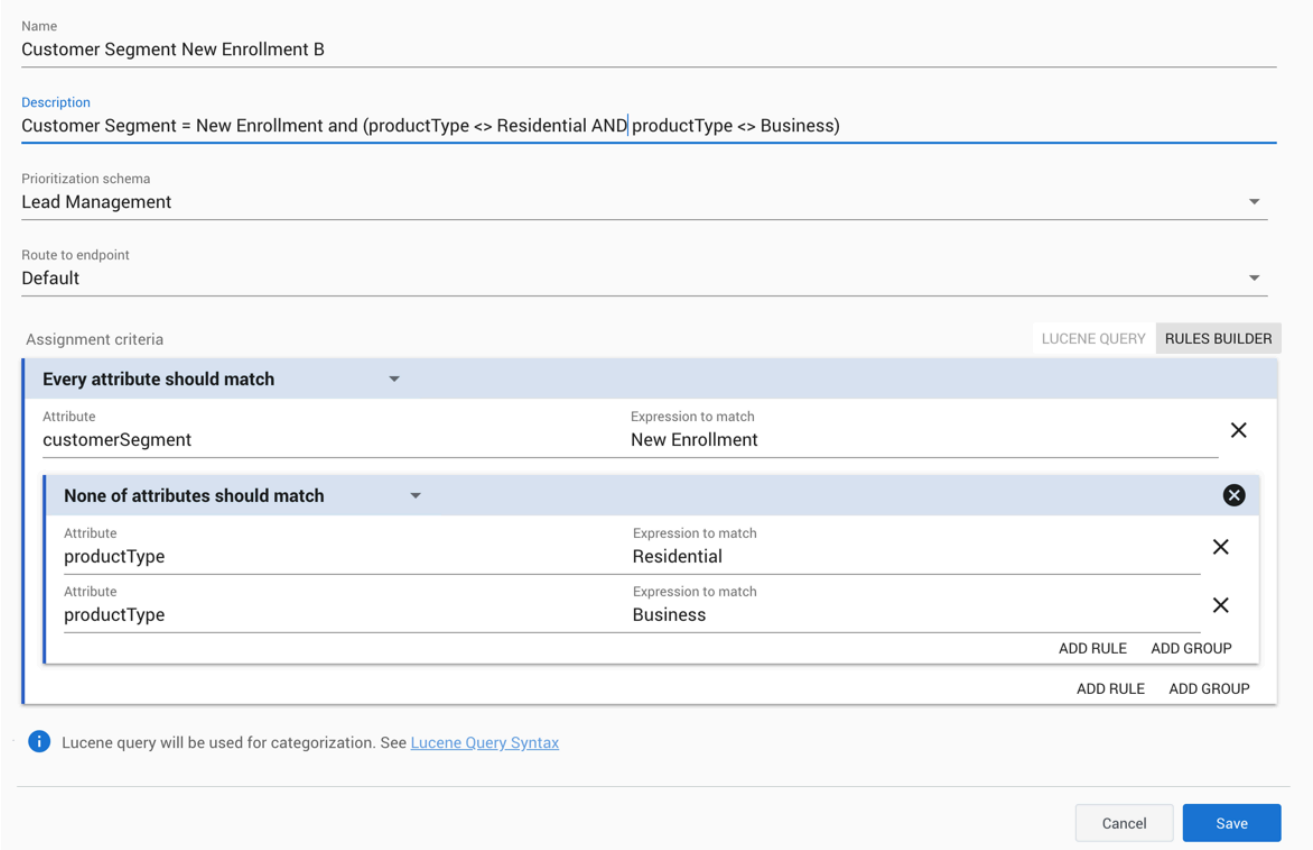
Rules examples



Example 1

This rule assigns workitems to this Category if they meet the following conditions:

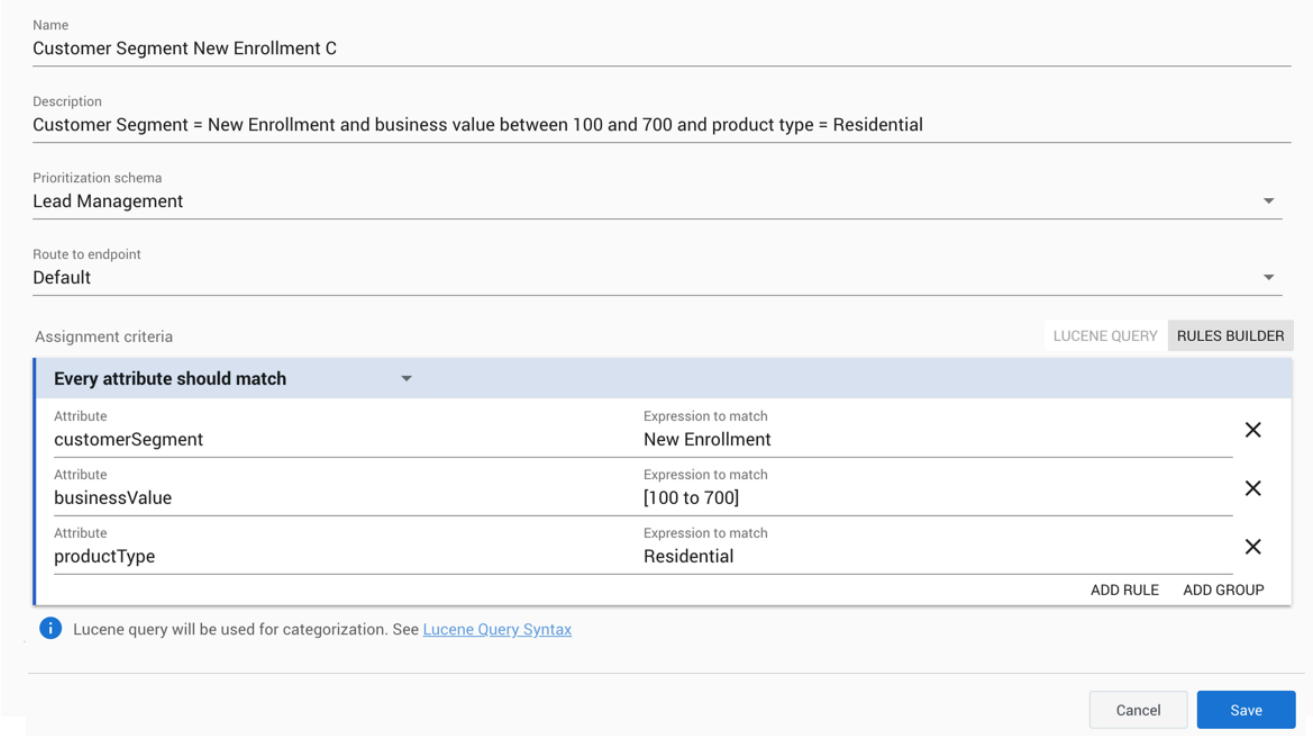
The value of their **customerSegment** attribute is `New Enrollment`
AND
EITHER
the value of their **businessValue** attribute is between `300` and `900`
OR
the value of their **productType** attribute is `Residential`.



Example 2

This rule assigns workitems to this Category if they meet the following conditions:

The value of their **customerSegment** attribute is `New Enrollment`
AND
NEITHER
the value of their **productType** attribute is `Residential`
NOR
the value of their **productType** attribute is `Business`



Example 3

This rule assigns workitems to this Category if they meet the following conditions:

The value of their **customerSegment** attribute is `New Enrollment`
AND
the value of their **businessValue** attribute is between `100` and `700`
AND
the value of their **productType** attribute is `Residential`.

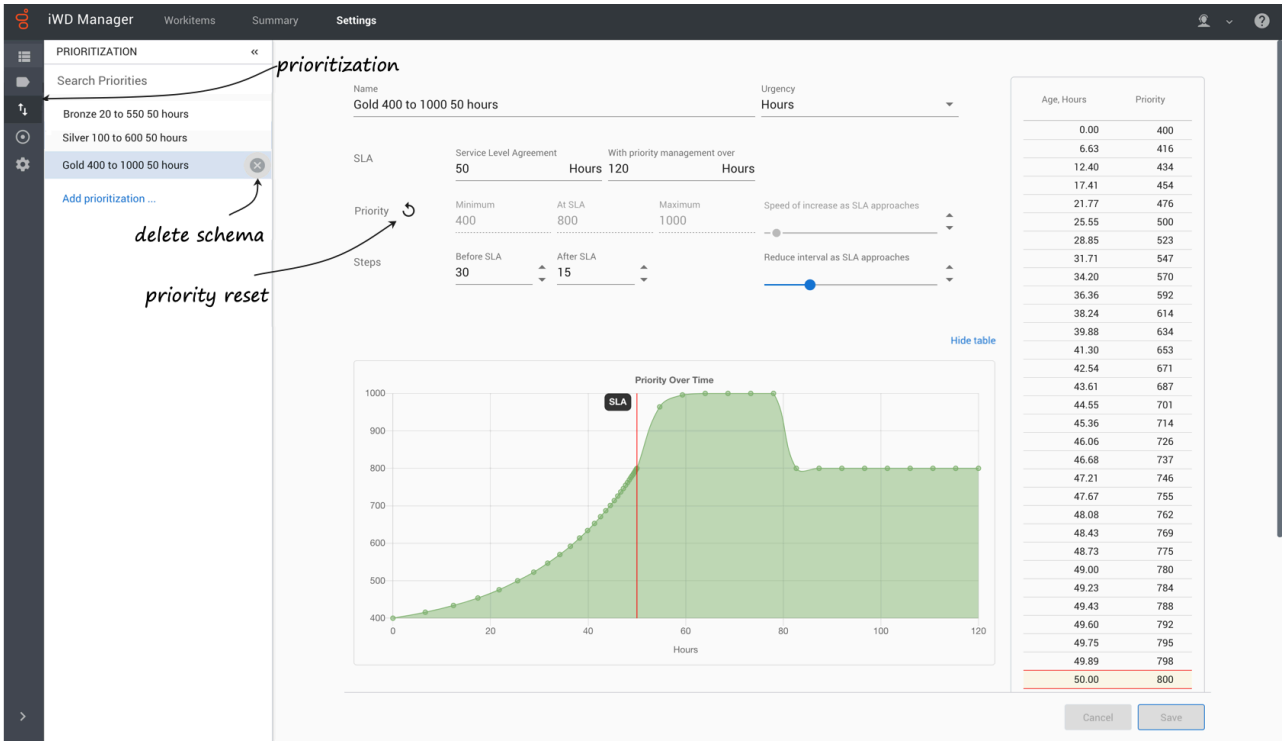
View, edit and create Prioritization schemas

> **Important**
>
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to IWD Prioritization.

A Prioritization schema defines an initial priority for new workitems in a Category, plus the timestamp (workitem age) at which subsequent reprioritizations take place and what the new priority value at each reprioritization will be. It also defines the Service Level Agreement (SLA) for the Category, and how workitems are prioritized post-SLA. A schema can be used by multiple Categories.

> **Important**
>
> Genesys recommends that you start with simple linear prioritization for their environments. As you become more accustomed to how prioritization works in your environment, you can evolve to a more complex prioritization graph. Genesys also recommends that you reduce the interval as the SLA approaches to increase the probability of workitems being assigned to an employee, as the priority is used to reflect the business value of the work. You can refine the graph by dragging and dropping the curve to represent the business value of the work item over the expected lifecycle. So if there are tiered penalties for work items that breach their due date and time, the graph after SLA can be arranged to reflect these tiers.

View and edit Prioritization schema



1. Select the **Settings** tab.
2. Select the Prioritization tool from the left toolbar.

The Prioritization schema details are displayed and can be changed if you have the appropriate privileges. Remember that such changes can have large impacts on contact center operation.

You can edit the schema by:

- Changing the values in any of the fields in the top part of the panel.
- Changing the priority values directly in the hideable table on the right of the panel.
- Dragging the individual data points up or down on the **Priority over time** graph.

Use the **Cancel** button to abandon all changes entered on the page.

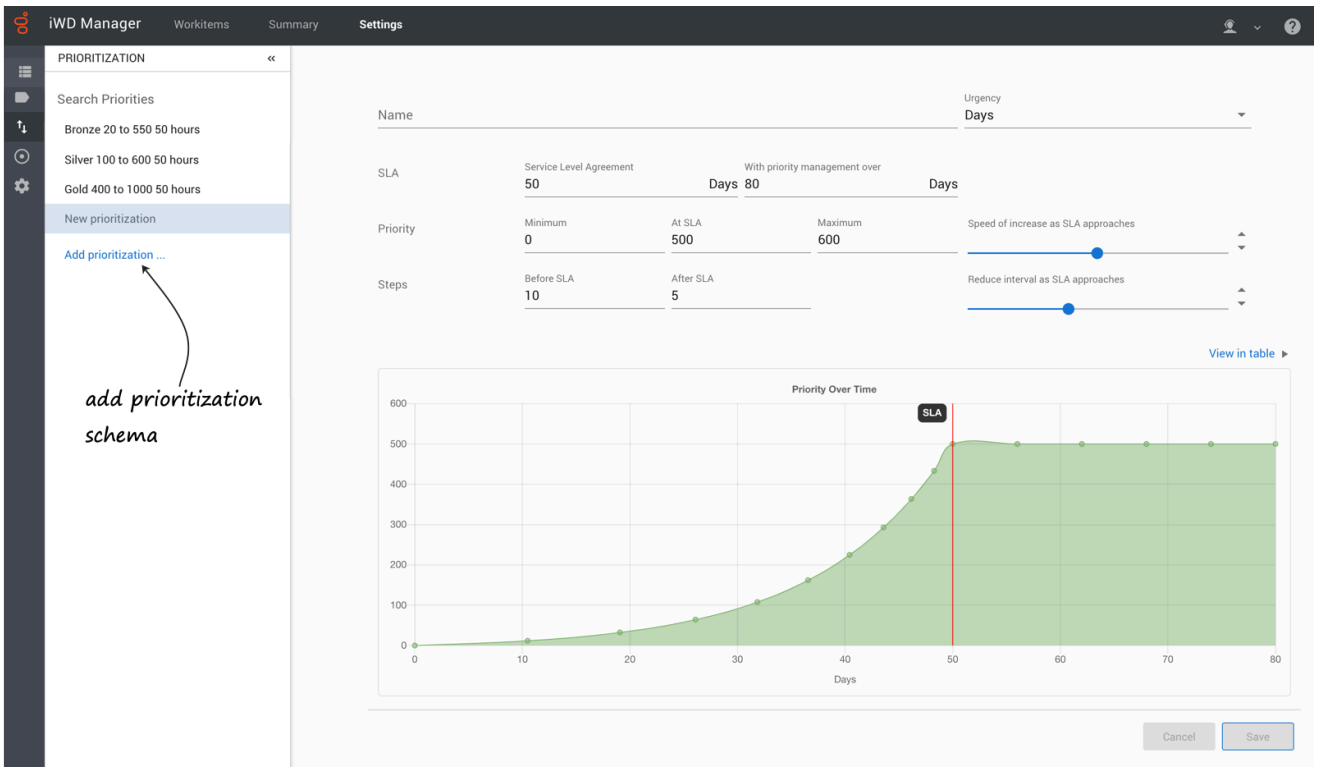
Use the **Reset** button to reset changes made in the graph and priority tables.

Field names

> **Important**
>
> Changes in the static fields and in the table are reflected dynamically in the **Priority over time** graph.

- **Name**—The schema name.
- **Urgency**—The units in which the intervals for calculation are denominated: Days, Hours or Minutes.
- **SLA**—The interval (Days, Hours or Minutes) by the end of which workitems must be completed.
- **With priority management over**—The post-SLA interval (Days, Hours or Minutes) during which the workitems are to be monitored and managed.
- **Priority**—Minimum, SLA and Maximum priorities for the workitems.
- **Speed of increase as SLA approaches**—Use the slider bar to accelerate or decelerate the change in priority when workitems under this schema are reprioritized before their SLA. As you move the slider bar, the curvature of the **Priority over time** graph changes dynamically to reflect the changes. A straighter line indicates a more single-speed regime. A more curved line indicates acceleration as the SLA approaches.
- **Steps**—The number of times the workitems under this schema are reprioritized, both before and after their SLA.
- **Reduce interval as SLA approaches**—Use the slider bar to alter the interval between reprioritization steps as the SLA approaches.

Add a new Prioritization schema



1. Select the **Settings** tab.
2. Select the Prioritization tool from the left toolbar.
3. Click **Add prioritization...** to create a new Prioritization schema.

Use the field definitions above to set up your new schema. Some fields have default values, listed here:

- **Urgency**—Days
- **SLA**—50
- **With priority management over**—80
- **Priority**
  - **Minimum**—0
  - **At SLA**—500
  - **Maximum**—600
- **Steps**
  - **Before SLA**—10
  - **After SLA**—5

View, edit and create Endpoints

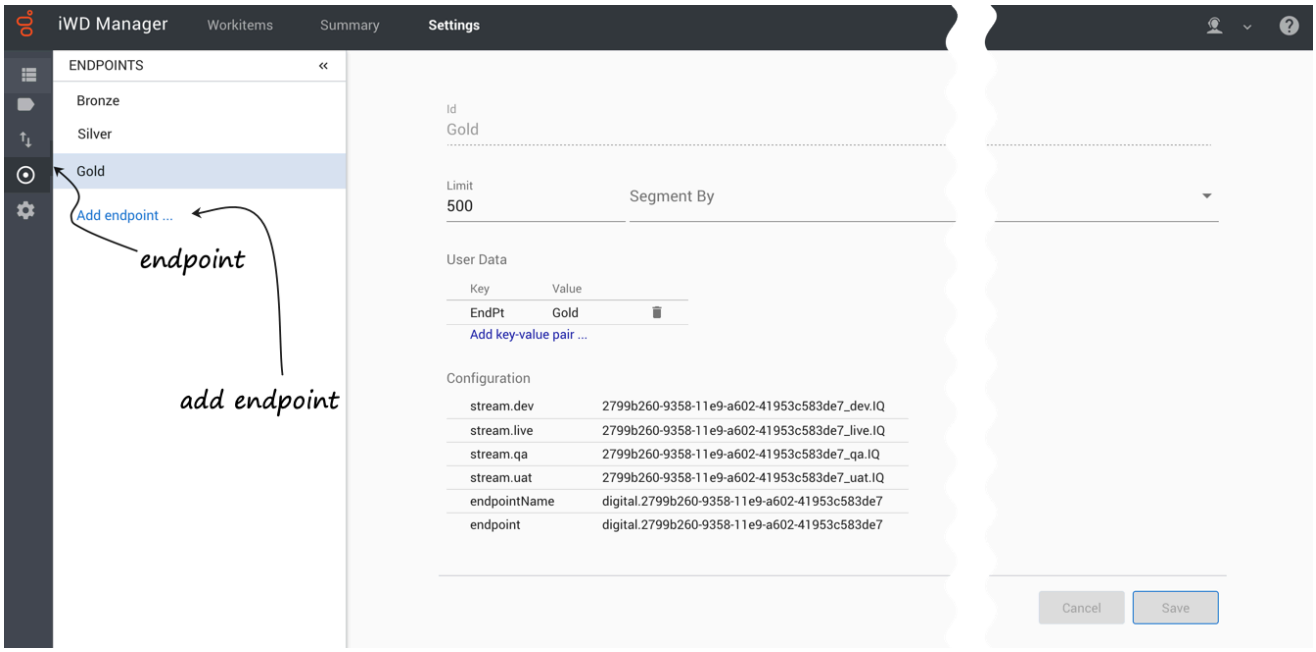
> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to IWD Endpoints.

A Designer Endpoint corresponds with the Designer application to which workitems will be distributed. The Designer application:

- Determines how this workitem is matched with the employee.
- Can add any information that needs to be attached if further information needs to be retrieved prior to sending the work to the employee. (Genesys recommends that the majority of the information for the employee is attached by the Designer application.)
- Determines how to distribute the workitem.

> **Important**
> Configuration of tenants to which Endpoints belong, and of all routing to Endpoints takes place in Designer.

View, edit and create Endpoints



Select the **Endpoint** tool from the left toolbar, then select a **Category** to display or edit its Endpoint details.

To add a new Endpoint click **Add endpoint**...

The **Limit** field is the limit for the number of unassigned/incomplete interactions that iWD creates for routing. iWD tracks the number of such interactions by subscribing and monitoring the Interaction Server events for each Interaction Queue associated with the Endpoint. If the limit is reached, iWD will not create any more interactions until some interaction is assigned to an employee, stopped or placed into the final queue (defined by the **Final queue** option in the iWD Options page, which is for use by Genesys Professional Services/Customer Care only).

The **Segment By** field displays an attribute by which workitems reaching this Endpoint can be segmented. To make changes to this value, either select from the displayed list or manually enter a different attribute.

The **User Data** field displays key/value pairs that will be attached to the user data of workitems when they are classified and targeted at this Endpoint. You can delete existing key/value pairs and add additional ones here.

The **Configuration** panel displays read-only details of the specific Designer application streams associated with this Designer Endpoint. You can read more about Designer Application streams here.

Designer

Important

This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Getting Started with Designer.

Important

These help topics are intended for the latest release of Designer (version 9).

- If you are running an older version of Designer (i.e. 8.x), some of the features and functionality described here might not apply. In that case, you should refer to this version of the Designer Help. (You can check your version in the top-right corner of the Designer interface.)
- If you are new to Designer 9 after running an 8.x version, refer to this section for important details about working in Designer 9.
- Contact your Genesys representative if you have any questions about the features that are available in your version of the software.

Designer is a web-based tool for developing self-service (IVR) and assisted service (routing) applications that run on the Genesys Engage cloud platform and enable you to manage voice and chat interactions.

This video shows a quick overview of Designer and demonstrates a few of its latest features:

Link to video

Supported browsers

Designer supports the latest versions of the following web browsers:

- Mozilla Firefox
- Google Chrome
- Microsoft Edge
- Apple Safari

The minimum display resolution supported by Designer is 1920 x 1080.

What's new

Here are some of the features and capabilities you'll find in the latest version of Designer:

Microservices platform	Designer is now built as a **microservice**, which means you get faster access to new features and zero downtime during upgrades.
Chatbots	You can use the Bot block in the Self Service phase of your application to add a chatbot to your application. Chatbots are software applications that apply automatic speech recognition and natural language understanding to listen and respond to customers in a way that resembles a conversation with a live agent. You can define and manage your bot resources in the new Bot Registry.
Streams-based workflow	Designer now features a streams-based workflow, where you can generate multiple builds of an application and assign them to different streams. For example, you can have one build of an application assigned to QA, and another build of the same application assigned to your Live operations. You can manage these builds from the application properties. In addition, you can now designate certain resources for use in testing, so that non-production builds do not reference resources that are being used in your live operations.
Omnichannel capability	"Build once, deploy anywhere." A single call flow now handles both voice calls and chat interactions.
Predictive Routing	If your site is enabled for Genesys Predictive Routing, you can use the new Predictive Routing block to add this powerful routing capability to your applications.
Rich media	For chats, customers can now click a "quick reply" button instead of manually typing a response. The quick replies offered are based on the DTMF Key options set up in the **Menu** block. In addition, the

	Play Message block has a new option that lets you specify an image to display within the chat window.
Social media channels	Designer is now capable of handling messages coming from **Facebook**, **WhatsApp**, **Twitter**, and **SMS**. If a customer contacts you from one of these social media platforms or services, Designer detects which channel the message is coming from and launches the appropriate application for managing and routing that type of interaction. If applicable, Designer can also send or post messages back to the incoming channel.
Personas	Designer now includes a variety of personas that you can use to control Text-to-Speech (TTS) services in your applications.

Check out the Designer page in the Genesys Engage cloud Release Notes for more information about these and other features and improvements to Designer. If you've been using a version of Designer older than 9, you should also review some of the changes in how Designer 9 works.

What do you want to do?

Use these topics as a starting point to find the information you need.

Getting started

Learn the basics.

- Getting started with Designer
- Using the Designer user-interface
- Understanding the application workflow
- Designer roles and permissions
- Managing variables

Managing resources

View and manage your application resources.

- Business Controls (Business Hours, Special Days, Emergency Flags, Data Tables)
- Media files and collections
- Digital messaging resources
- Speech grammars and Bots

Building applications

Using the various blocks to build applications in Designer.

Application phases

Working with blocks

Reporting and Analytics

Designer has an integrated set of dashboards that provide detailed information about your operations.

Designer Analytics

Managing the dashboards

For Designer Administrators

Manage users and their access to resources (Designer Administrators only).

Partition-based Access Control (PBAC)

For users of previous versions of Designer (8.x)

If you are new to Designer 9 after working in a previous version, there are some things you should be aware of when working in Designer 9:

Designer now uses a new application workflow based on builds and streams. You can learn more about this new type of workflow on the Application Workflow page. To gain familiarity, you can clone an existing application and experiment with making changes, generating builds, and assigning the builds to streams. When you are comfortable with the workflow, you can start to adopt it with your original applications.

- If you make any changes to a Business Control, Audio Resource, or Speech Grammar resource, those changes now take effect immediately if the resource is being referenced by the LIVE production build. However, if you make a change to a data table, you must publish the data table for the changes to take effect in any builds that are referencing it.
- In previous versions of Designer, any changes made to an application took effect immediately after the application was published. In Designer 9, you must generate a new build and assign it to the LIVE stream for the changes to take effect.
- Some permissions have changed. Refer to Permissions and Access for a list of Designer roles and their associated permissions.

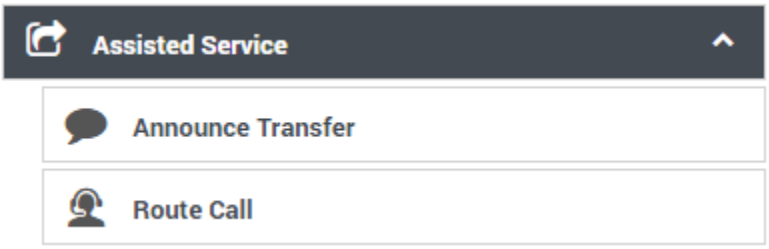
Getting Started

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

Designer provides easy-to-use, highly functional blocks that enable common tasks in a simple and straightforward manner. For example, there are blocks for building menus, playing messages, setting up decision points, routing interactions to agents, and offering surveys.

To build an application, you simply drag blocks from the **Palette** and drop them into the **Application Flow**, which is divided into phases and represents the application structure.

For example, you might use the following blocks in the **Assisted Service** phase, in which an agent helps a customer. The first block controls the announcement to the caller that they are being transferred, and the second block controls the routing function to an agent:



But before you start building applications, you should learn about some of the key concepts. The following topics provide an overview of things you should know when working with Designer:

- Using the Designer user-interface
- Designer roles and permissions
- Managing variables
- Understanding the application workflow

User Interface

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

Supported browsers

Designer supports the latest versions of the following web browsers:

- Mozilla Firefox
- Google Chrome
- Microsoft Edge
- Apple Safari

The minimum display resolution supported by Designer is 1920 x 1080.

> **Important**
> **Internet Explorer** is deprecated. Genesys recommends that IE users select one of the supported browsers when using Designer. Using non-supported browsers with Designer can produce unexpected results.

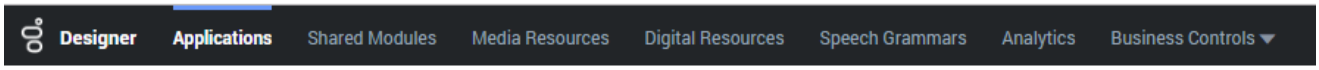
User interface overview

Watch this video to see an overview of the Designer user interface:

Link to video

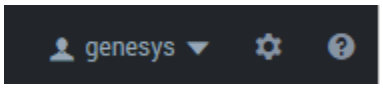
The various elements within the interface are described below.

Navigation bar



Provides one-click access to Applications, Shared Modules, Media Resources, Digital Resources, Speech Grammars, and Business Controls objects.

Workspace toolbar



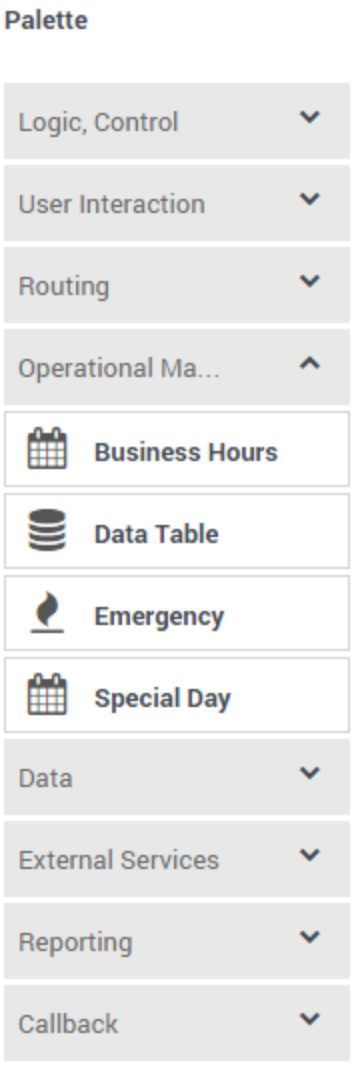
Provides buttons for common actions. Click your user name to log off. Click the settings icon to view or modify the global **Caching** settings for certain resources and to toggle certain **Features**. Click the Help icon to access the Designer Help.

Application toolbar

Provides buttons for common actions. Click Settings to set global settings for your application. Click Save Flow to save and validate your application, or click **Publish** to save and validate your application and prepare it for use by routing engines. Click **Build** to create and manage the builds associated with the application.

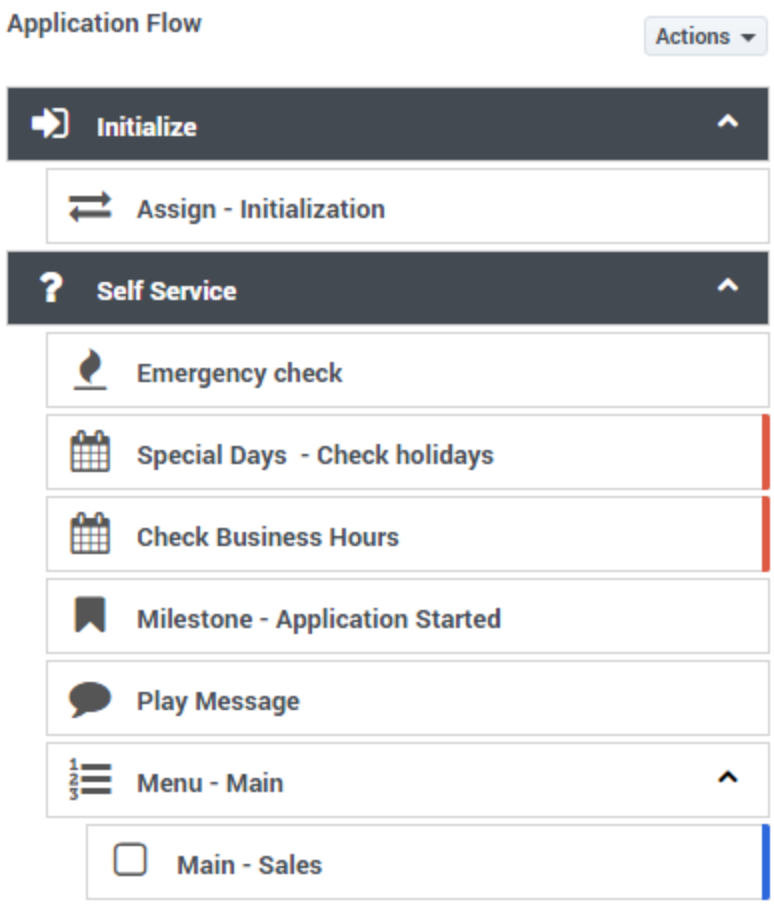
Palette

Provides all available blocks that you can use in your application, sorted by functional grouping:



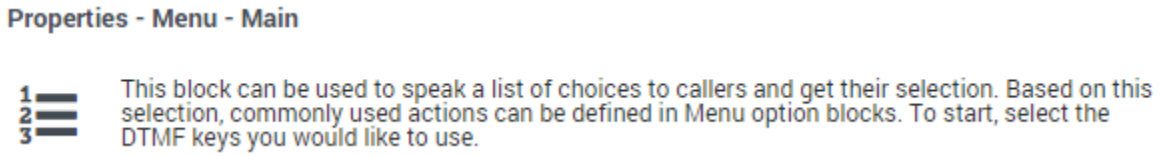
Application flow

Provides the main area to build your application by adding blocks vertically. (See Build Logic for more information.)



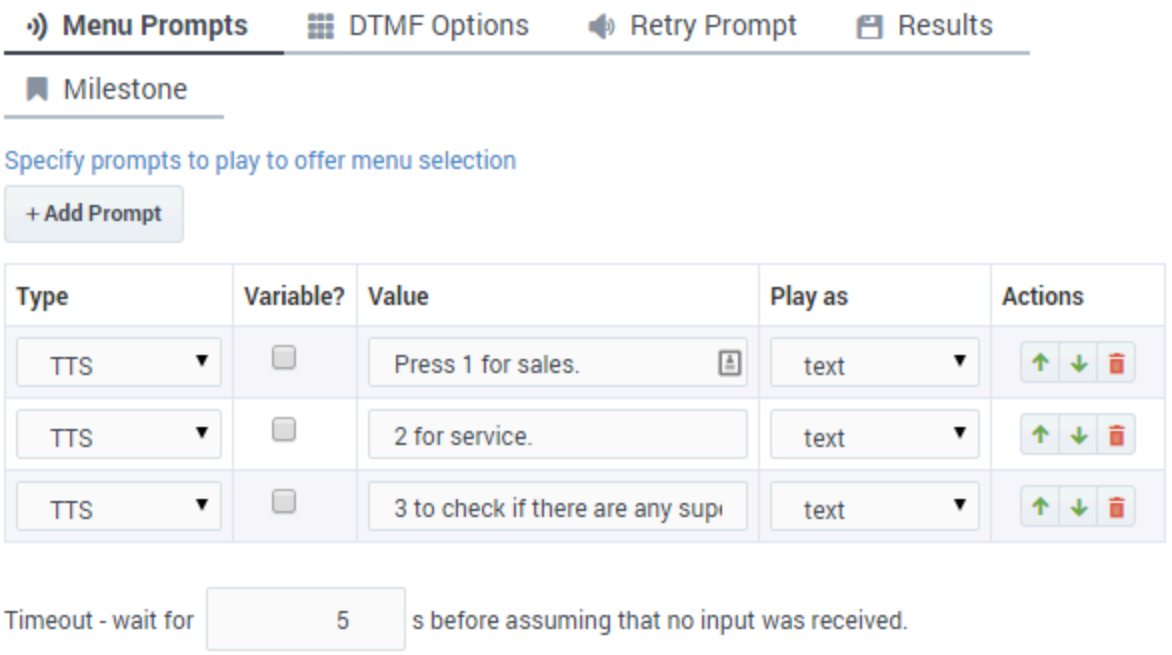
Help pane

Displays help information for the selected block:



Block properties

Displays all properties exposed by a block and provides assistance to set them:



Search bar

Most Designer pages have a search bar located near the top-right corner:

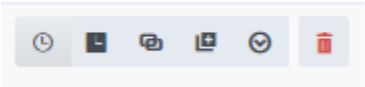
You can use this search bar to find items on that particular page. As you start typing, the items on the page are filtered to only show those items that match the terms you've entered.

There are additional levels of search available when you are working directly within an application flow. See Searching the application flow.

> ### Tip
> If you are searching on the **Applications** page, you can search for a specific phone number that is assigned to an application. The applications list is then filtered to only show the application(s) with that number assigned.

Actions toolbar

Many of the resource pages in Designer have an **Actions** toolbar:



The action items on the toolbar will vary based on the resource page you are viewing. Depending on the functions available for that particular resource, you can use this toolbar to do the following:

View History

Use the **View History** button ( ) to open the history view (or audit log) for a resource.

The history view shows you a list of each time the resource was viewed, edited, or published, the user who made the change, and the new and previous value of any properties that were changed.

You can use the buttons on the page to view the results for a specific time period (for example, **last 1W** to see the results for previous week), or use the date fields to specify a custom date range. Results can be sorted or searched and you can use the **Export** button to export the results as a CSV file.

You can drill-down further into each results item by double-clicking it. This opens an audit window opens that displays additional details for that particular event.

List Media/Message Collection Consumers

Use the **List Media/Message Collection Consumers** button ( ) to see information about which applications and modules are using that particular media or message collection.

Clone

Use the **Clone** button ( ) to clone the resource. You'll have to give the cloned resource a unique name.

> **Important**
> A cloned resource does not inherit the history and published versions of the original resource.

List Module Versions

Use the **List Module Version** button ( ) to view the version history for a shared module. You can then choose to make a version **public**, or view a version (in read-only mode) as it appeared at a particular time.

Export

Use the **Export** button ( ) to export a resource. This exports the selected resource for use in another Designer workspace. When you export a resource, all versions of that resource are exported, including the unpublished version.

> **Tip**
> If you are using a Safari browser when performing the export, the exported file is downloaded as *unknown*. The file is valid and can be imported successfully, but you might want to rename it to something more meaningful.

Import

This **Actions** item is only available on the Data Tables page. Use the **Import** button ( ) to import the contents of a CSV file into the selected data table.

Delete

Click **Delete** ( ) to delete **all versions** of the resource. However, published applications that were using the resource (in other words, applications that had previously generated their code) are not affected.

Quick filters

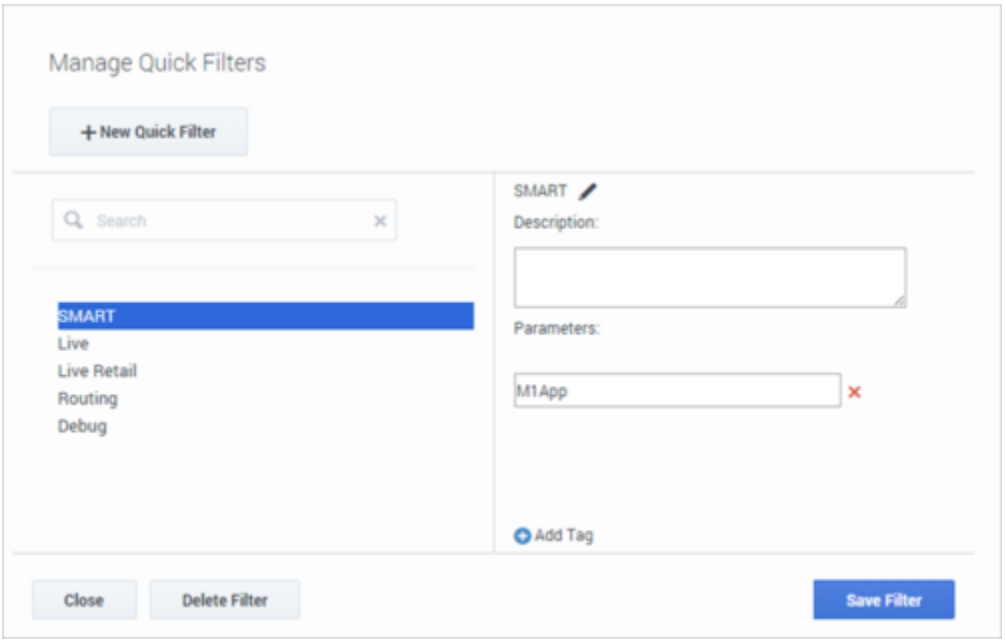
This toolbar enables you to filter a list of resource items by selecting one or more filters that are associated with tags. The list then refreshes to show only those items that match the selected filters.

In this example, the **SMART** and **Debug** filters are selected so that only data tables with those tags are shown:

Note that the filters will display any item in the list that has the associated tag, even if there are other tags associated with that item. If you navigate to a new resource page (such as going from **Data Tables** to **Business Hours**), any selected filters are automatically applied to the new page.

To add, modify, or delete quick filters, click the **Settings** icon to open the **Manage Quick Filters** window. To associate a filter with a specific tag, select it, and add the tag(s) under the **Parameters** section.

In the above example, the **SMART** filter was associated with the "M1App" tag, as follows:



The **Quick Filters** toolbar appears on the following resource pages: Special Days, Business Hours,

Data Tables, Applications, Shared Modules, Emergency Flags, Media Resources, and Digital Resources. The same filters appear on each page, and any filters that you create are visible to other Designer users.

> **Tip**
>
> Tags are a useful tool for keeping resources organized. For consistency, Genesys recommends that you define and use a similar set of tags across your various resource types.

Application Phases

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

An application flow consists of several sections of common blocks, known as application phases.

Each Designer application contains the following four phases:

- Initialize
- Self Service
- Assisted Service
- Finalize

Each phase is described in more detail below.

Initialize

This phase initializes application-level user variables and parameters to use when the application executes. The application initializes during this phase.

By default, the following actions take place:

- Initialize and set up user variables.
- Load application run-time parameters from external sources.

- Process interaction properties (for example, ANI and DNIS) and application run-time parameters. System variables or properties may be initialized internally.
- If configured, additional processing that was set up by the user.

Self Service

The **Self Service** phase is the IVR (Interactive Voice Response) portion of the interaction. This phase hosts blocks that provide automated interaction with the customer via speech, chat, and/or DTMF. It attempts to provide automated service and contain the interaction within an IVR, so there is no need to route the interaction to an agent.

If routing is necessary, this phase collects necessary data from the user through various questions and menus, and then determines how to route the interaction in the next phase, **Assisted Service**.

> ### Tip
> To enable voice call recording for the **Self Service** phase, set the **EnableSSRecording** variable to **true** in the **System Variables** section.

The following are typical actions that take place during the **Self Service** phase:

- Play Messages. These may be pre-recorded audio files or dynamic text spoken using TTS (see Media Resources).
- Check business hours and customize logic based on the outcome (for example, take *this* action if we are closed).
- Collect user input.
- Present choices to customers using menus.
- Navigate customers appropriately, based on their responses (i.e. segmentation and branches).
- Call external RESTful APIs and fetch data into user variables.
- Call Bots to interact with customers.
- Update user variables and write ECMAScript expressions (see Variables).
- Set up and process global commands and hot words.

The **Self Service** phase updates user variables with collected or calculated data. This data is later used by other blocks in the **Self Service** or **Assisted Service** phase.

Interaction processing might complete during the **Self Service** phase. In this scenario, the application control skips the **Assisted Service** phase and proceeds to the **Finalize** phase. For example, if the business hours check determines that the contact center is closed, the corresponding announcement is played to the customer and the interaction is terminated.

Assisted Service

This phase hosts blocks that route the interaction to a live agent, if necessary.

During the **Assisted Service** phase, the application attempts to route interactions to agents. Routing is performed based on data collected in previous phases. For example, target skills are taken from user variables.

The following are typical actions for this phase:

- Attempt to route the interaction while playing music or prompts.
- Call external RESTful APIs.
- Update user variables.

There may be multiple Route Call blocks in sequence. Each **Route Call** block might try to route the interaction to different targets with different timeouts. For example, it might expand a target by geographical location.

Each **Route Call** block has a timeout, after which the next **Route Call** block in sequence is executed. If any of the blocks successfully routes a call, the **Assisted Service** phase is complete and processing continues to the Finalize phase.

Finalize

This phase provides post-processing and interaction termination after the customer has been serviced.

When interaction processing is finished, the application goes to the **Finalize** phase to perform post-processing for various scenarios that are based on how the interaction was completed.

The following are examples of typical scenarios:

- Interaction was abandoned by the customer (while in either the **Self Service** or **Assisted Service** phase).
- Interaction was completed in **Self Service** phase.
- Interaction was routed to an agent in the **Assisted Service** phase.
- Interaction was delivered to voicemail in the **Assisted Service** phase.
- User opted to leave a queue and schedule a callback.

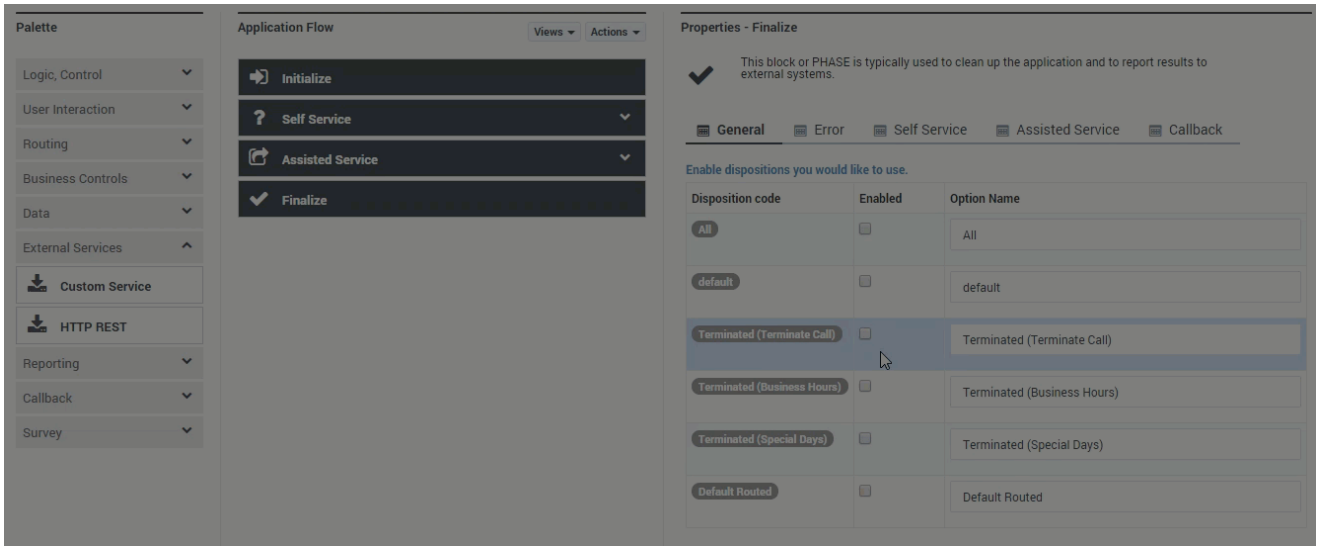
You can also use the **Finalize** phase to submit application data to an external system for reporting metrics, or to select a disposition code for post-processing.

Disposition codes

When you click on the **Finalize** block in the application flow, each of the tabs (**General**, **Error**, **Self Service**, **Assisted Service**, and **Callback**) has a list of disposition codes that you can enable.

When you select a disposition, a block for it is created below the main **Finalize** block. You can then drag other valid blocks (such as an HTTP REST block) below the disposition block to further customize the handling for that disposition.

Here's an example:



> ### Tip
> Setting up handlers for the **Finalize** phase is optional. There may be no need to do anything special for these cases.

This list contains additional information about some of the common disposition codes that can be enabled in the **Finalize** phase:

All

When an application enters the **Finalize** phase, it has only one disposition code, so only one disposition block is executed. However, the **All** disposition code is unique in that it is always executed, in addition to (and after) any disposition block related to the actual disposition code of the application. This is the only case where more than one disposition block is executed.

Typically, you would select the **All** disposition code when you want to execute some post-processing logic, no matter what the actual application disposition code is. This is more efficient than duplicating the same logic in every possible disposition block.

default

This code is used when no other disposition code is applicable. For example, it could indicate that a call was not routed, not terminated (by any party), and was likely still active when the session and/or application completed.

System Error

There was an unexpected error in the application (such as a script validation error).

Application Timeout

The application got stuck in a loop and reached the timeout limit.

Terminated (Terminate Call)

The application reached a Terminate Call block.

Terminated (Business Hours)

The call came in outside of regular business hours, as specified in the Business Hours block.

Terminated (Special Days)

The call came in on a special day, as specified in the Special Day block.

Terminated (Emergency)

The emergency flag was set, as specified in the Emergency block.

Terminated (Menu Option)

The caller chose a menu option to exit or end the call.

Abandoned in Self Service

The caller hung up before completing the **Self Service** phase of the application.

Abandoned in Queue

The caller completed the **Self Service** phase, but hung up while waiting to speak with an agent.

Completed in Self Service

The caller successfully completed their call in the **Self Service** phase — the application did not need to go to **Assisted Service** and went directly to **Finalize**. If the call ended in **Self Service** due to some other condition or event (such as the call being received outside of business hours or the customer hanging up), then the appropriate disposition code for that condition or event is used (**Abandoned in Self Service**, **Terminated**, and so on).

Routed to Agent

The interaction was successfully delivered to an agent.

Routed to DN

The call was successfully delivered to a direct number.

Permissions and Access

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

Genesys Designer provides layered access roles to ensure that your users only have access that is appropriate for your business needs—such as the ability to make changes to prompts, business hours, or set an emergency routing flag—without exposing control to the overall application logic.

Designer only permits one user at a time to open an application, shared module, or data table for editing. If you try to open one of these resources and it is already in use, Designer shows you which user has the resource locked. You can then choose to open it as read-only or go back to the previous screen.

If a user has opened an application or module for editing and then goes inactive, their browser eventually displays a timeout warning. If they do not respond, the resource is closed and the lock released.

Designer roles

Designer supports the following user roles:

Designer Developer

Designer Developers have the ability to view and modify all resource types within Designer, with the exception of features that are restricted to Designer Administrators (such as managing partitions and user accounts). Developers are also able to view the flows of application builds and previous versions of shared modules in read-only mode.

Designer Business User

Designer Business Users have limited access to Applications, Speech Grammars, and Shared Modules. They can view these resources, but cannot make any changes to them.

They have full access to Media Resources, Analytics, and most Business Controls resources (i.e. Business Hours, Special Days, and Emergency Flags). They can view and modify Data Tables, but

they cannot create or delete them.

Business Users can also assign phone numbers to applications.

Designer Administrator

Designer Administrators have full access to the Admin settings for Designer, which includes the ability to manage partitions and control which users have access to them. They can view most of the other resource types in Designer, but cannot make any changes to them.

> **Important**
>
> For full access to all resource types in Designer, a user would need to be assigned to *both* the Designer Administrator and Developer roles.

Designer Analytics

All Designer roles have access to Analytics. Users assigned *only* to the Designer Analytics role can view and modify the Analytics dashboards, but do not have access to any other Designer resources.

Designer permissions

The following tables provide a high-level overview of what each role can do with various Designer resources:

Media Resources

Task	Designer Developer	Designer Business User	Designer Administrator	Designer Analytics
View	✓	✓	✓	
Modify	✓	✓		
Create	✓	✓		
Delete	✓			

Message Resources

Task	Designer Developer	Designer Business User	Designer Administrator	Designer Analytics
View	✓	✓	✓	
Modify	✓			

Task	Designer Developer	Designer Business User	Designer Administrator	Designer Analytics
Create	✓			
Delete	✓			

Applications

Task	Designer Developer	Designer Business User	Designer Administrator	Designer Analytics
View	✓	✓	✓	
Modify	✓			
Create	✓			
Delete	✓			
Switch to last snapshot	✓		✓	
Assign phone numbers	✓	✓	✓	
Change block properties	✓			

Shared Modules

Task	Designer Developer	Designer Business User	Designer Administrator	Designer Analytics
View	✓	✓	✓	
Modify	✓			
Create	✓			
Delete	✓			

Speech Grammars

Task	Designer Developer	Designer Business User	Designer Administrator	Designer Analytics
View	✓	✓	✓	
Modify	✓			
Create	✓			
Delete	✓			

Business Controls

(Business Hours, Emergency Flags, Special Days, Data Tables)

Task	Designer Developer	Designer Business User	Designer Administrator	Designer Analytics
View	✓	✓	✓	

Task	Designer Developer	Designer Business User	Designer Administrator	Designer Analytics
Modify	✓	✓		
Create	✓	✓ (all except **Data Tables**)		
Delete	✓	✓ (all except **Data Tables**)		

Blocks

Task	Designer Developer	Designer Business User	Designer Administrator	Designer Analytics
View	✓	✓	✓	
Modify	✓	✓		
Create	✓	✓		
Delete	✓	✓		

Restricted mode during upgrades

During upgrades, Designer continues to provide full service, but goes into a restricted mode that only allows selective modifications to be made. While Designer is in this mode, certain objects (such as applications, shared modules, and workspace settings) are locked for editing until the upgrade is complete. You can, however, continue to make changes to objects such as business controls, audio resources, and grammars.

Application Workflow

> **Important**
>
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

This page describes the builds and streams workflow that Designer uses for developing applications. In this type of model, an application consists of multiple work streams, each of which represents a different stage in the life cycle of the application (such as the development or quality assurance stage).

At various times, you can "freeze" the current state of the application code as a build, which you can then assign to the appropriate stream. The builds are promoted through the various streams until a build is approved for use in the live production environment.

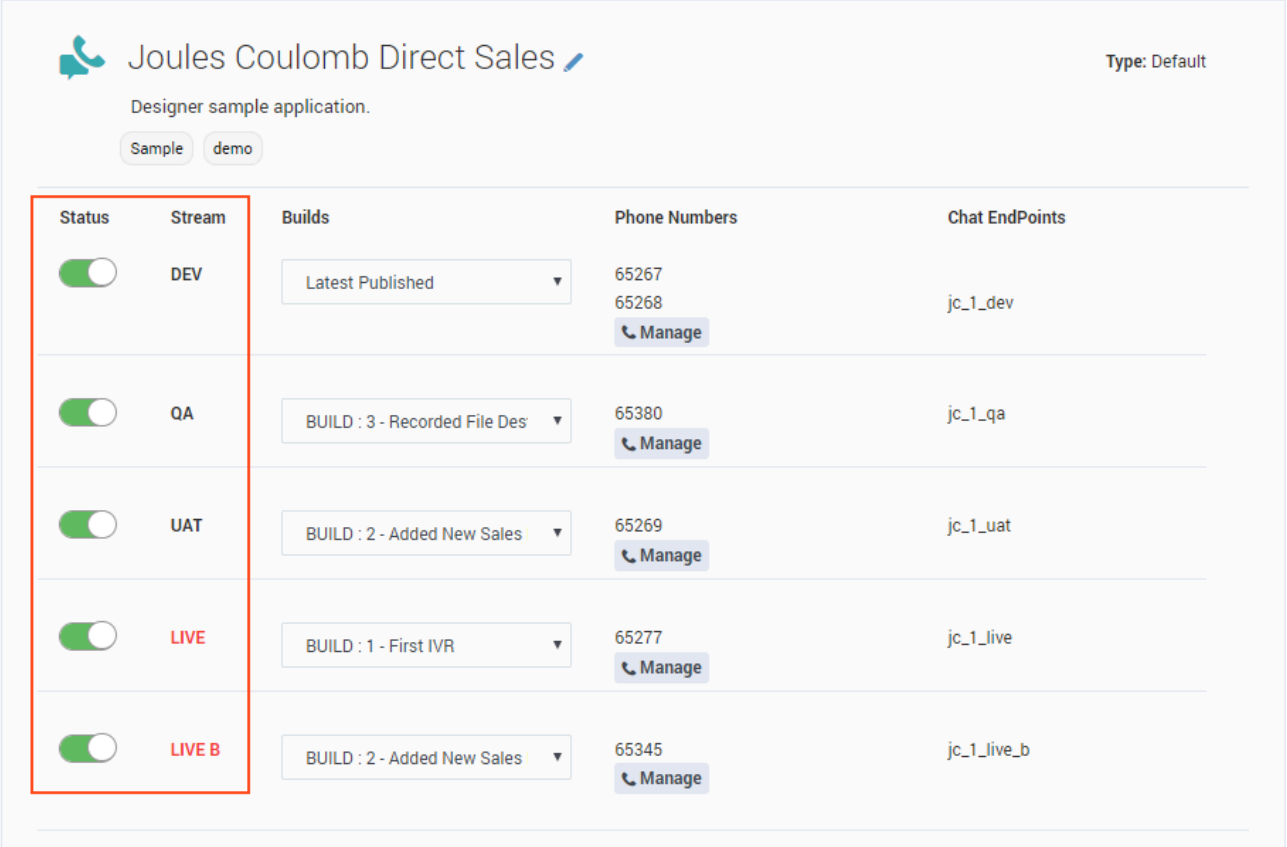
One of the primary benefits to this type of workflow is that each stream runs a build of the application that is completely independent of the others. For example, the development team can be working on a build without impacting the build being tested by the quality assurance team or the build that is actively running in production.

This video shows an example of a workflow in action:

Link to video

Application streams

You can view and manage the streams of an application from the application properties:



All streams are enabled by default. You can use the **Status** sliders to enable or disable them, as needed.

DEV (Development)

Use this stream to test the latest changes to your applications. You can then generate a build that can be promoted to **QA**, **UAT** (as applicable), and then to a **LIVE** stream. The DEV stream is the only one that can run the latest published version of the application code. As such, it is also the only stream impacted by Designer upgrades.

> ## Warning
> The **DEV** stream is intended for development testing purposes only and does not support live production traffic. Live production traffic must always be assigned to the **LIVE** or **LIVE B** streams.

QA (Quality Assurance)

This stream is for builds that require Quality Assurance testing.

UAT (User Acceptance Testing)

This stream is for builds that require beta-testing or some other variation of user-based trials.

LIVE

This stream is for builds that are actively running in the production environment.

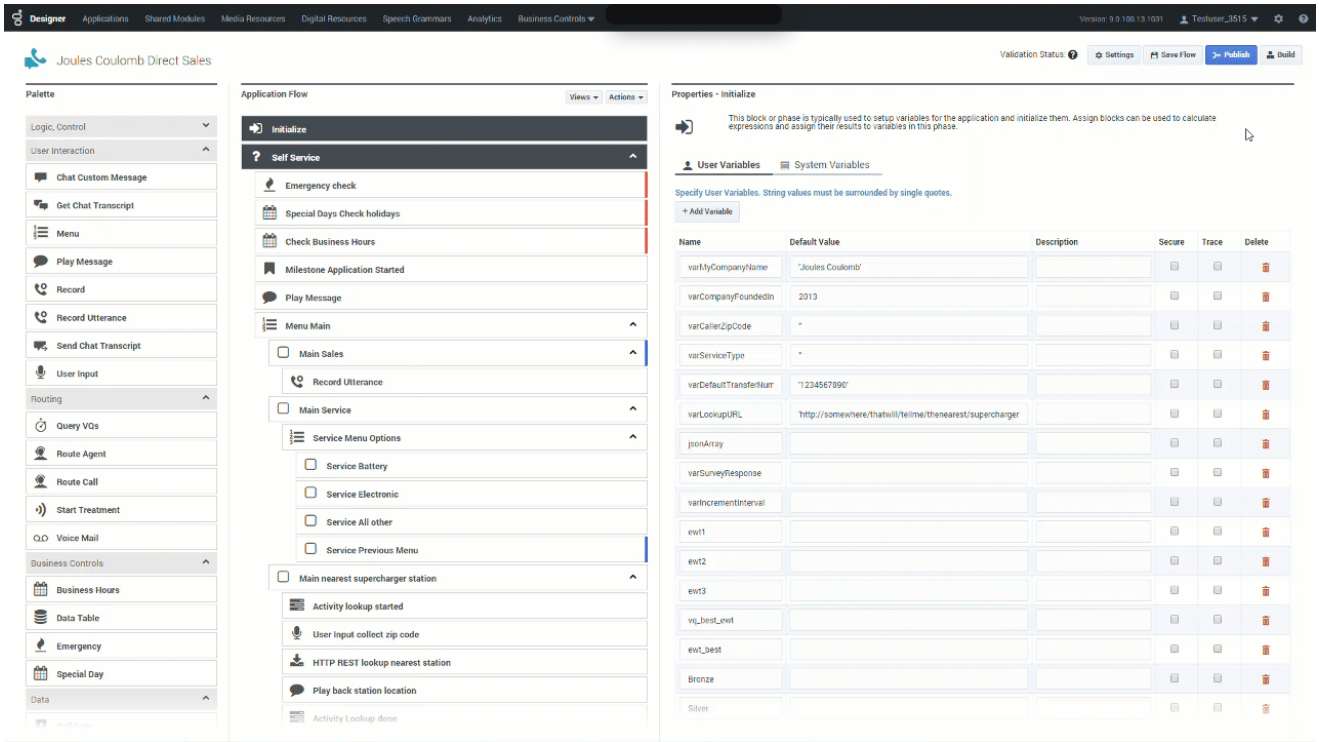
LIVE B

This is an optional stream that can run a second application build in the live production environment. For more information, see LIVE B streams.

Application builds

When you want to "freeze" a version of your application code, you can create a build. A build is basically a self-contained package of the application code. It has all of the resources it needs to run, so you can assign a build to a stream without worrying about impacts to the original resources or to the other builds.

You can create a build by clicking **Build** when the application is open in editing mode. Designer automatically increases the version number of the build, but you can add your own label and note:



You can manage the builds for an application by clicking Manage Builds from the application properties.

> **Important**
>
> - **Once generated, a build cannot be changed.** If you want to make changes, you must generate a new build. You can also choose to perform a rollback by selecting a previous build.
> - **There is a limit of 20 builds for each application.** Therefore, it's recommended that you generate a build only when required. If you exceed 20 builds, the **Build** button becomes disabled. You can re-enable the button by deleting builds that are no longer needed.

Parallel Test Environment (PTE)

The Parallel Test Environment enables you to use test resources instead of production ones when running an application build in a non-production stream. You can enable this option on the **Misc** tab in the application settings. When enabled, you can create copies of certain resources and add a special *test_* prefix to the resource name.

For example, if you create a business object resource to use in the LIVE Production stream (example: **Hours**), you would also create a similar business resource object with the *test_* prefix for use in the non-Production streams (example: **test_Hours**).

Then, you can reference the resource in your application as you would normally, using the original name (using the above example, you would just reference the **Hours** resource). Streams that are running non-production builds (DEV, QA, and UAT) will automatically reference the PTE versions of the resources (test_<*resource name*>) instead of the ones being used in production. The LIVE build continues to reference the original production resources.

> **Important**
>
> When making changes directly to the **test_<*data table*>**, you do not need to add the **test_** prefix to the resource names, as Designer adds this prefix automatically when PTE is enabled.

PTE is supported for Business Controls resources (**Business Hours**, **Emergency Flags**, **Special Days**, and **Data Tables**) and certain configuration resources (**Virtual Queues**, **Agents**, **Agent Groups**, and **Skills**).

LIVE B streams

You can run an optional LIVE B stream in addition to the existing LIVE stream. This allows you to run a second application in LIVE production mode, which gives you greater flexibility in how you can introduce new application builds into your production environment.

For example, each LIVE stream can have different contact points assigned to it, so one way you could use the LIVE and LIVE B streams is to allocate interactions coming from a certain region to a particular stream, or use the streams to balance (or gradually introduce) the number of interactions being handled by a new production build. In this way, you can implement a form of A/B testing.

Let's say you wanted to split or balance the number of interactions being handled between the LIVE and LIVE B builds. Typically, you would gradually introduce more interactions from LIVE to LIVE B, which could be done as follows:

- Assign **Build X** to the LIVE stream.
- Use the **Manage** button to assign phone numbers to the LIVE stream. At this point, 100% of the interaction load will be handled by **Build X** on the LIVE stream.
- Now, assign **Build X+1** to the LIVE B stream.
- Assign some of the phone numbers to the LIVE B stream so that 70% of the interaction load is handled by LIVE and 30% by LIVE B.

> **Tip**
>
> The Transfer from Others button is a great way to easily and quickly move several numbers from one stream to another.

- Check Designer Analytics to see if there are any issues.
- If there are no issues, we can then move more of the phone numbers from LIVE to LIVE B, so that 30% of the interaction load is being handled by LIVE and 70% load by LIVE B.
- Again, we continue to monitor Designer Analytics to see if there are any issues.
- If everything still looks ok, we can move the remaining phone numbers from LIVE to LIVE B, so that 100% of the interaction load is being handled by LIVE B.
- Assign **Build X+2** to the LIVE stream. This is a new build that we want to test in conjunction with **Build X+1** on LIVE B.
- Assign some phone numbers to the LIVE stream so that 30% of the interaction load is being handled by our new application build running on LIVE and 70% by LIVE B.
- Continue to check Designer Analytics for any issues, and adjust and balance the interaction load between the two builds as desired.

Variables

You can use two types of variables in Designer:

- User Variables - These are variables that you create. You can use these variables throughout the application and in all phases.
- System Variables - These variables are created with the application and cannot be deleted.

Tips

Variable names must be alphanumeric, but not start with a numeric character. For example:

- Valid variable names = **a**bcdef123 or **c**123badef
- Invalid variable names = **1**23abcdef or **3**abcdef21

Variable values may be:

- ECMAScript objects, such as `Date()`.
- Valid ECMAScript expressions. Do not add an ending semi-colon (;) as typically required by ECMAScript.
- Simple values, such as numeric or string.
  - If the value is a string, it must be surrounded by single quotes (for example, `'value'`). If the value also uses a single quote, you can use a backslash to escape the quote character (for example `'Joe\'s Pizza'`).

> **Important**
> The block properties page will indicate if single quotes are required.

User Variables

You can add user variables in the **Initialize** phase. You can assign initial values to these variables in the **Initialize** phase, or by setting values in an **Assign** block in the **Initialize** phase.

You can also assign a system variable as the default value of a user variable. For example, you might assign the system variable **DNIS** to a user variable you have created. (If the system variable does not have a value at the time of the call, the default values are used.) This is also supported for Self Service Shared Modules.

Important

User variables may not be initialized correctly if their value is set to one or more system variables in the **Initialize** phase itself. This phase should be used to declare variables, but their values should be assigned later using an **Assign** block if the value or the value expression involves a system variable.

Warning

You can delete a variable even if it is required by the application. Designer validates the application when you click **Publish**, at which time it checks for deleted variables.

Securing Variables

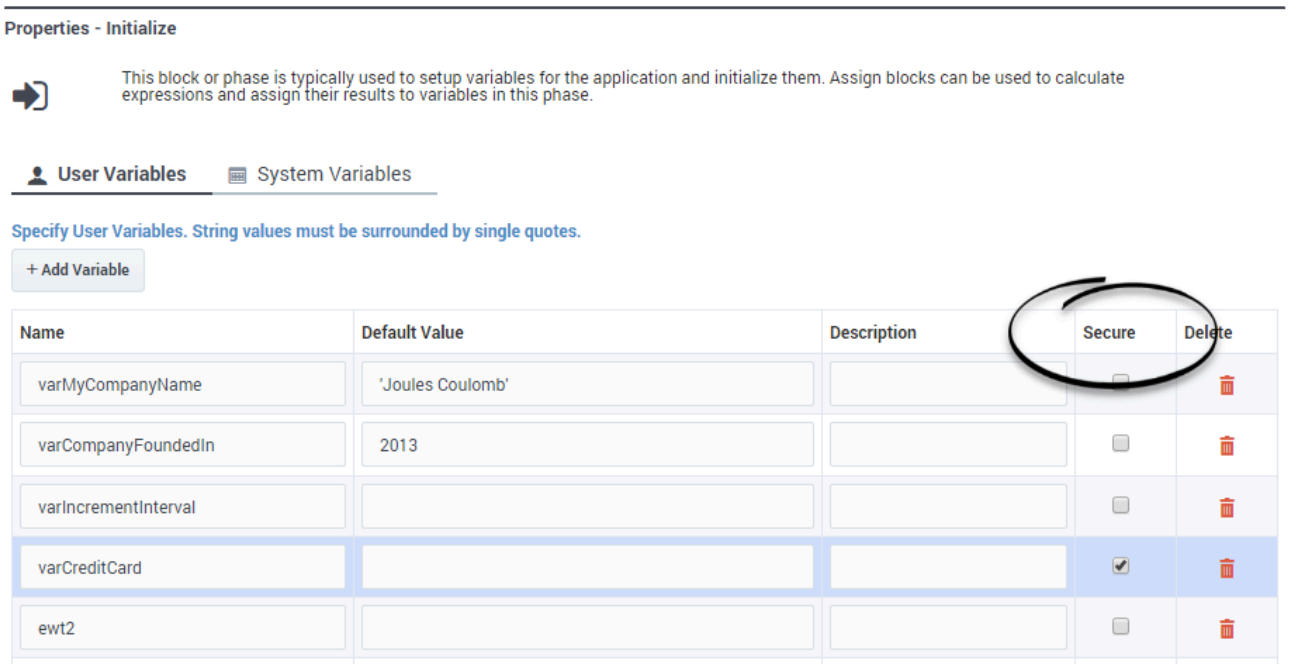
Warning

Variable values can be captured in a variety of data sources when Designer applications run on the Genesys platform. **To prevent the values of variables from being exposed as plain text in Designer and the Genesys platform, they must be marked as Secure.**

You can enable the **Secure** check box to indicate that a variable is secure and must not be logged or recorded. The application must be published for any changes to these settings to take effect.

Secure variables function as follows:

- Secure variable values are not logged in application logs.
- When you use secure variables to store the results of a user input, the user input is masked in MCP (Media Control Platform) logs.
- When you use secure variables to play back prompts, the prompt message is masked in MCP logs.
- Users cannot select secure variables for blocks that record reporting information, such as Call Data, Activity, and Milestone blocks.
- Secure variables are not reported in Analytics.



System Variables

The **Initialize** phase has a second tab that lists system variables - these variables are created with the application and cannot be deleted.

Most system variables are initialized when the application starts and can be used throughout the application, such as the **Last Milestone** variable. When your application starts, the initial value of **Last Milestone** is an empty string. While your application runs, the **Last Milestone** value is set to the last milestone that your application reaches.

The following system variables are available:

Variable Name	Description
DNIS	Specifies the dialed number.
ANI	The number associated with the calling party.
MaxTime	Maximum time (in minutes) to keep this session alive.
Timezone	The timezone used for this application, unless this value is overridden in other blocks. **Important** If you override this value—for example, by using Advanced Scripting in the Assign Variables block or setting up a **timezone** data type when defining columns for a data table—you *must* assign it a valid value (e.g. "America/Los_Angeles"). Otherwise, Designer might experience issues when performing

Variable Name	Description
	certain functions, such as executing Business Hours logic. The **Timezone** drop down contains a list of valid values you can reference.
Language	The default language for this application that is used for announcements.
AppLanguageName	The name of the default language for this application that is used for announcements.
AppCountryName	The name of the country for this interaction (can also be specified by the application).
RoutingSkills	A set of skills that might be specified in some blocks, such as **Menu Option** child blocks, that determine how the call is routed. For example, if you select a Skill in the **Call Handling** tab of a **Menu Option** block, this selection is stored in the **RoutingSkills** variable. Then, in a subsequent **Route Call** block, you can enable the **Use system variables RoutingSkills and RoutingVirtualQueue set already in Menu Options** check box to use the value of the **RoutingSkills** variable.
RoutingVirtualQueue	A virtual queue that might be specified in some blocks, such as **Menu Option** child blocks, that is used for routing unless a different queue is specified in Routing blocks. For example, if you select a Virtual Queue in the **Call Handling** tab of a **Menu Option** block, this selection is stored in the **RoutingVirtualQueue** variable. Then, in a subsequent **Route Call** block, you can enable the **Use system variables RoutingSkills and RoutingVirtualQueue set already in Menu Options** check box to use the value of the **RoutingVirtualQueue** variable.
EstimatedWaitTime	The estimated wait time for the call to be routed to an agent.
IVRProfileName	This application's calls will be associated with the given IVRProfile for VAR reporting.
GVPTenantID	This application's calls will be associated with the given tenant for VAR reporting.
SelectedTarget	This is the DN and the switch name of the target to which the interaction was routed or should be routed to definitively. The target format is `Name@SwitchName.Type`.
SelectedVirtualQueue	The virtual queue that was selected.
SelectedComponent	The agent-level target to which the interaction was routed or should be routed to definitively. If the target selected for routing is of type **Agent**, **Place**, **Queue**, or **Routing Point**, this variable contains the target. If the desired target type is **Agent Group**, **Place Group**, or **Queue Group**, the

<table>
<tr><th>Variable Name</th><th>Description</th></tr>
<tr><td></td><td>function returns the agent, place, or queue from the corresponding group to which the interaction was sent. The target format is <code>Name@StatServerName.Type</code>.</td></tr>
<tr><td>SelectedTargetObject</td><td>This is the high-level target to which the interaction was routed or should be routed to definitively. If a skill expression is used, the function returns: <code>?:SkillExpression@statserver.GA</code> or <code>?GroupName:SkillExpression@statserver.GA</code>. The target format is <code>Name@StatServerName.Type</code>.</td></tr>
<tr><td>SelectedAgent</td><td>This is the Employee ID of the agent to which the interaction was routed.</td></tr>
<tr><td>Access</td><td>(Optional) When present, this is an ECMAScript object that represents a switch access code. The table below show its properties and the corresponding switch access code fields:<table><tr><th>Access property</th><th>Switch access code field</th></tr><tr><td>prefix</td><td>Code</td></tr><tr><td>rtype</td><td>Route Type</td></tr><tr><td>destination</td><td>Destination Source</td></tr><tr><td>location</td><td>Location Source</td></tr><tr><td>dnis</td><td>DNIS Source</td></tr></table></td></tr>
<tr><td>CustomerSegment</td><td>The segment to which the customer belongs, based on information that the customer has provided.</td></tr>
<tr><td>CustomerId</td><td>A unique identifier for the customer, based on information that the customer has provided.</td></tr>
<tr><td>EnableSSRecording</td><td>Enable call recording in the Self Service phase.</td></tr>
<tr><td>CallbackReporting</td><td>An object containing key-value pairs for callback reporting.</td></tr>
<tr><td>PositionInQueue</td><td>The call's position in queue while waiting to be routed to an agent.</td></tr>
<tr><td>AppCountry</td><td>The country code for this call (can be specified by the application).</td></tr>
<tr><td>AppRegion</td><td>The region for this call (can be specified by the application).</td></tr>
<tr><td>AppCallType</td><td>The type of call (can be specified by the application).</td></tr>
<tr><td>AppUserDisposition</td><td>A custom disposition that the application can use to specify a user-specific outcome.</td></tr>
<tr><td>AppUserDispositionCategory</td><td>A custom disposition category that the application can use to categorize user-specific outcomes.</td></tr>
<tr><td>AppDeflectionMessage</td><td>The application can use this variable to track deflections by specifying the message played when a caller disconnected their call.</td></tr>
</table>

Variable Name	Description
AppLastMilestone	The last milestone that the application achieved.
AppStrikeoutMilestone	The last milestone that the application achieved before strikeout.
AppBailoutMilestone	The last milestone that the application achieved before the caller bailed out to an agent.
AppDeflectionMilestone	The last milestone that the application achieved before the caller was deflected.
ScriptID	The ScriptID as reported by the routing engine.
AppSelfHelpedMilestone	Used to contain a self help milestone.
SdrTraceLevel	Enables users to set the recording level. This variable accepts the following values: • `100` — Debug level and up. Currently, there are no Debug messages. • `200` — Standard level and up. This setting shows the existing blocks array in the SDR. • `300` — Important level and up. This setting filters out all blocks except those containing data that can change from call to call (such as the **User Input** block).
AppSessionType	The type of the session. The default value is `inbound` for inbound calls. Survey applications must use the value `survey`.
EnableRouteCallRecording	Set to `true` or `false` to enable or disable call recording for routed calls in the **Assisted Service** phase. Leave blank to use platform defaults.
GmsCallbackServiceName	The GMS Callback Service name.
GmsCallbackServiceID	The unique identifier that GMS assigns to a scheduled callback.
survey_sOffer	Set by the Setup Survey block to specify if a survey was offered, setup, or rejected.
survey_iAgentScore	Holds the user satisfaction score for the agent, if this question is asked by the survey.
survey_iCompanyScore	Holds the user satisfaction score for the company, if this question is asked by the survey.
survey_iCallScore	Holds the user satisfaction score for the overall call, if this question is asked by the survey.
survey_iProductScore	Holds the user satisfaction score for the product, if this question is asked by the survey.
survey_iRecommendScore	Holds the user's rating score (on a scale of 0-10) of the company, product, or service. Used to calculate Net Promoter Score (NPS).
ApplicationRevisionSerialID	A read-only variable that increments by 1 each time an application is revised.
ApplicationPath	The absolute path to the application.

Variable Name	Description
Interaction	Details about the interaction (`Interaction.Subject`, `Interaction.Type`, and so on).
Contact	Details about the customer contact (`Contact.EmailAddress`, `Contact.LastName`, and so on).
DefaultPartition	The default partition used to provide access control in GIR. This variable can be overridden by settings in the **Record** block.
Flow Entry Count	Number of times (including this run) a Designer application has been executed to handle this interaction.
TreatmentIterationCount	Keeps track of how many times a treatment has been executed.
AgentsTotalSize	The total number of agents that could possibly be available. For example, the total number of agents in a specified Agent Group.
AgentsLoginSize	The number of agents that are actually logged in.
ChatEntryPoint	(Digital only) Holds the point of entry for a chat interaction. Can be used in application logic at runtime to provide alternative processing or to facilitate the use of parallel testing environments.
InteractionSource	Source of the interaction. Expected values are `web` (i.e. desktop and mobile browsers) or `mobile` (i.e. apps).
ReferrerUrl	URL of the previous page (document.referrer) to detect where the customer came from (might be shortened).
UserAgent	Type of browser the customer is using (e.g. Chrome, Mozilla, Opera).
UserAgentOS	Type of operating system the customer is using.
AutoStopInteraction	(Digital only) Specifies whether or not the interaction is to be automatically terminated when the session ends. The default value is `auto` (to automatically terminate a *chat* interaction if it exceeds 20 application runs or exceeds the specified expiration time); other valid settings are `true` or `false`.
ChatOfferVQ	Name of the Virtual Queue that was queried for the Estimated Wait Time, to determine if chat will be offered.
ExpirationTime	Maximum time (in minutes) from the time this interaction was first processed by an application to keep this interaction alive. When an application terminates and the AutoStopInteraction variable value is `auto`, the application terminates the interaction if that interaction has been processed for longer than the specified ExpirationTime.

Variable Name	Description
Persona	The persona to be used for this application.
IsResumedFromParking	(Digital only) Holds the number of times a digital interaction has been parked.

VAR Metrics

> **Important**
> VAR action IDs are stripped of spaces and pipe characters (|). This includes implicit actions that are generated when a caller enters a shared module.

Use the **IVRProfileName** variable (User Data Key: **gsw-ivr-profile-name**) to associate the application VAR metrics with an IVR Profile. Use a value of `auto` to auto-detect the IVR Profile.

Use the **GVPTenantID** variable (User Data Key: **gvp-tenant-id**) to associate the application VAR metrics with a tenant. Designer attaches the value to user data. Use a value of `auto` to auto-detect the tenant.

These variables are listed in the properties of blocks once they are defined.

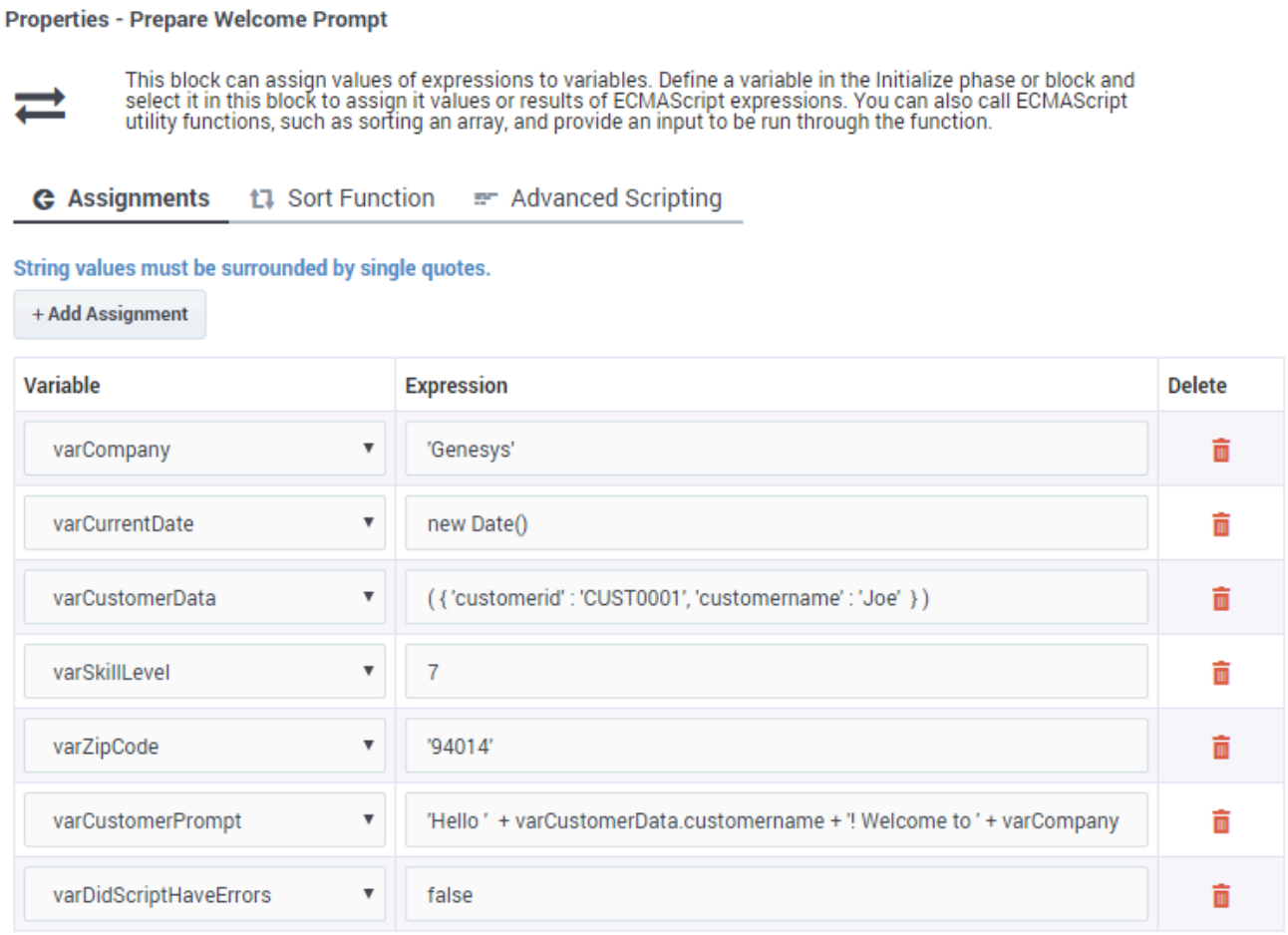
Assigning Values to Variables

Designer lets you assign values to variables in different ways. These examples show a few of the methods you can use to assign different types of values to a variable, including a JSON value.

Example 1: Simple Assignment

The easiest (and recommended) way is to assign a value to a variable using the **Assignments** tab on the Assign Variables block.

Click **Add Assignment** to add an assignment slot to the block, then choose a variable from the **Variable** column. For the **Expression**, you can use the name of another variable whose value should be copied in to the **Variable** column, a literal value (for example, "Sales Channel"), or an expression whose value should be calculated first and the results assigned to the **Variable**.



You must use single quotes (' ') when specifying string values, but you can assign numeric values without quotes. Note that the *varZipCode* above is a string data type (the single quotes tell JavaScript to treat it as a string), but it contains only numbers. It's important to remember that JavaScript treats string and numeric data types differently. For example, 1 + 2 = 3, but '1' + 2 = '12'.

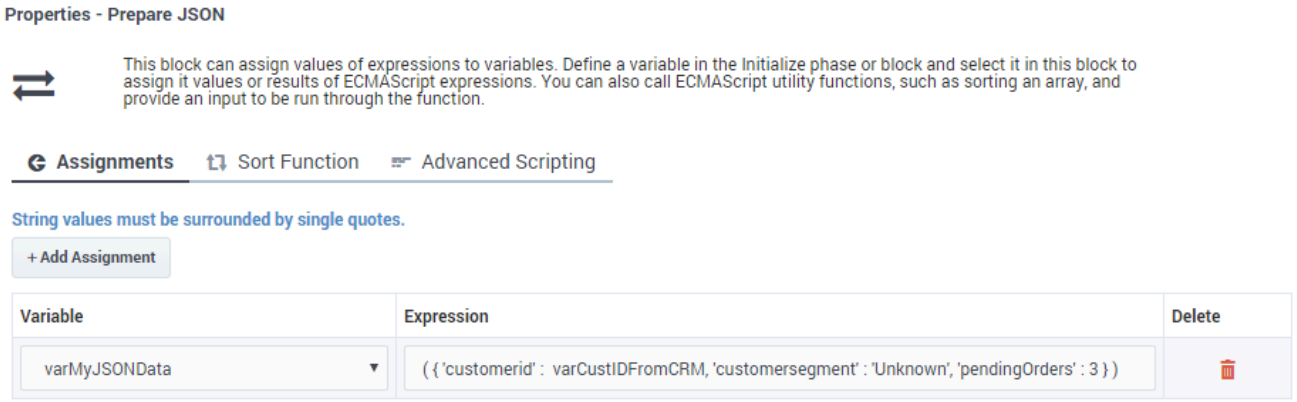
JSON data (for example, *varCustomerData*) is typically retrieved from an external data source, but you can also form a JSON string in the application. JSON strings must be enclosed in brackets ( ) and should follow the rules and syntax for JSON strings as defined by JavaScript. Note also that variables can easily be used to form part of the JSON string (as represented by *varCustIDFromCRM*, in the example shown below).

The *varCustomerPrompt* above shows a simple string expression, with the different string segments linked together by a **+**. If used in a **Play Message** block, it will play "Hello Joe! Welcome to Genesys." It accesses a property of the *varCustomerData* object using the "." notation and combines it with the welcome message.

> **Important**

> Although the terms ECMAScript and JavaScript are used interchangeably throughout this Help, Designer technically supports ECMAScript and does not support JavaScript functions that are typically used for web-browser based applications, such as pop-up windows, alerts, and so on.

Here is another example of how you could build a JSON expression. It contains mostly hard-coded strings, but also uses a variable to form part of the JSON string:



Example 2: Advanced Scripting

If your application requires you to go beyond simple assignments and use JavaScript constructs like loops or multiple nested conditions, you can use the **Advanced Scripting** tab, which allows you to compose valid ECMAScript or JavaScript.

> **Important**
>
> Advanced Scripting is an optional feature and might not be enabled on your system. To enable this functionality, contact Genesys.

To use this feature, you need a basic level of familiarity and understanding of ECMAScript syntax and rules. Any errors in the script can cause erratic behavior, so test your changes to make sure that your script works correctly.

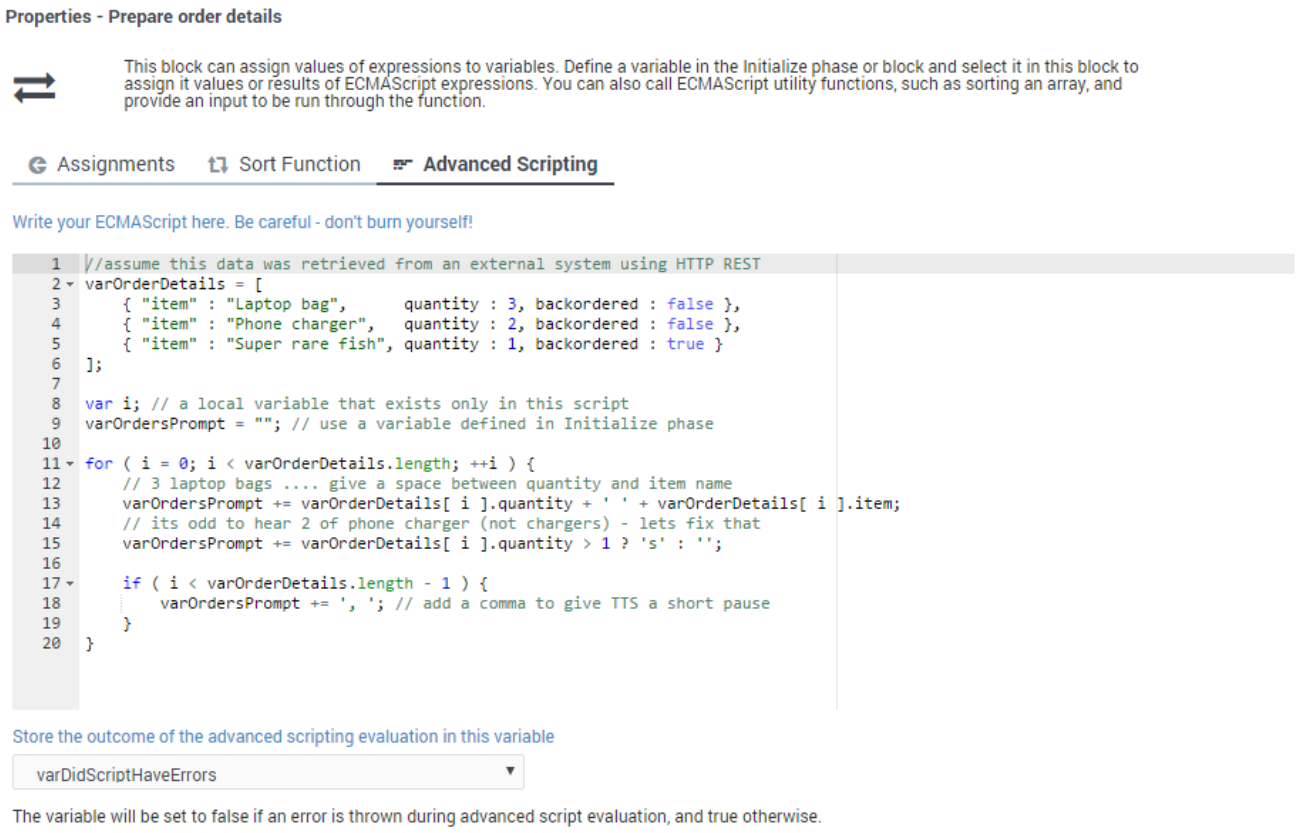
> **Warning**
>
> Use caution! Designer can check your script for syntax errors, but cannot validate it nor check for runtime errors that might occur when the script runs during a call.

In this example, the script sets the variable *varOrdersPrompt* to "3 Laptop bags, 2 Phone chargers, 1 Super rare fish". Here's how it works:

The sample script below first initializes JSON data in `varOrderDetails` so that it becomes an array of three JSON objects. Each JSON object has properties — item, quantity, backordered. The script then proceeds to loop through orders and forms a string to play back to the caller to notify them of their order status.

Note that this script uses variables in two scopes:

- A scope exclusive or local to this script itself ("`i`"). This variable remains available only while this script runs, and then it disappears.
- Top level variables that were defined in the **Initialize** phase — these remain available throughout this application flow, but not in any modules this application calls (such as `varOrdersPrompt`).



Saving and Publishing Your Application

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

It is a good idea to manually save your work often, especially after you have made important changes.

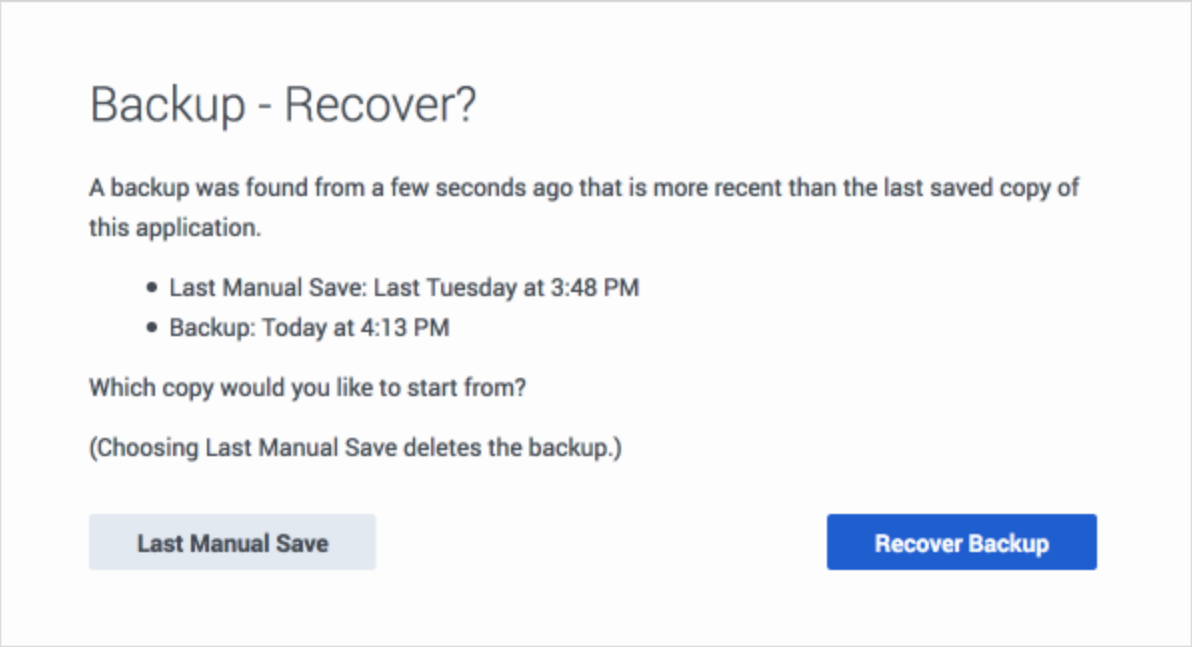
Click **Save Flow** to save your application. This action saves your work and performs some validation checks on your application. If no problems are found, a green check mark appears beside the **Validation Status** field. Otherwise, if problems are found, a warning icon appears beside the **Validation Status** field. You can click the warning icon to display the list of validation issues.

When you are ready to test and deploy your application, click **Publish**. Designer performs another validation test on your application and, if no errors are found, it generates the code that will run on Genesys platforms.

Designer also automatically saves a temporary backup of your work, which is refreshed periodically. If you forget to save your changes, and the editing session ends for some reason (for example, maybe you accidentally closed an active browser tab or were logged out due to inactivity), Designer provides a few recovery options.

Backup and recovery

If you try to edit an application or shared module that has unsaved changes from a previous session, Designer lets you know that it has found a local backup of your application, along with a comparison of timestamps between this local backup version and the version that is saved on the server.



You can choose one of the following recovery options:

Last Manual Save

Designer opens the last version that was manually saved to the server. If you select this recovery option, the local backup copy is discarded and any changes that were made *after* the last manual save are not included in the recovery.

Recover Backup

Designer opens the local backup version, which is recovered from the cache of your browser. You can then click **Save Flow** to save your changes to the server.

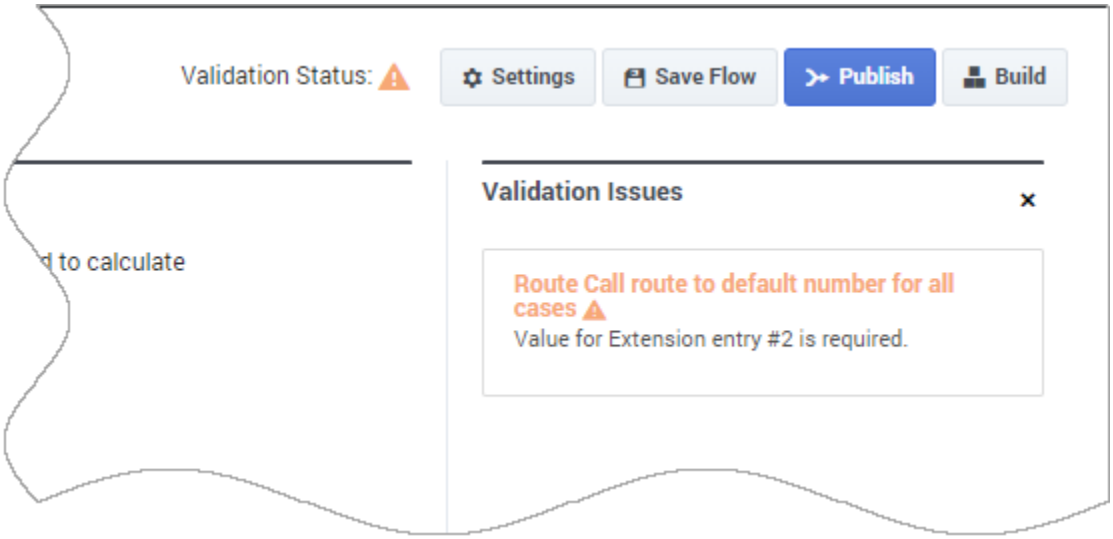
> **Important**
> This backup and recovery feature doesn't work in private browsing modes, such as "incognito" mode in Chrome or "private" mode in Firefox. These modes do not permit data to be stored or retrieved from the browser cache.

Validation Issues

If errors are found in your application, you can click the exclamation icon (⚠) beside the **Validation Status** field to display the **Validation Issues** list.

Warnings are displayed in yellow and errors in red. Code generation can complete successfully if there are warnings present, but fails if there are any errors.

Click a warning or error to go to the block containing the issue and address the problem.



File size limitations

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

This page provides information about the **maximum permitted file sizes** for resources being uploaded or imported/exported from Designer. These limitations help to ensure that your operations continue to run efficiently with minimal interruption.

Uploads

- Media (Audio) or Grammar resource = **10 MB**

Imports

- Application or Shared Module = **10 MB**
- Audio Collection = **15 MB**

Exports

- Application or Shared Module (with audio) = **150 MB**
- Application or Shared Module (without audio) = **50 MB**
- Audio Collection = **100 MB**

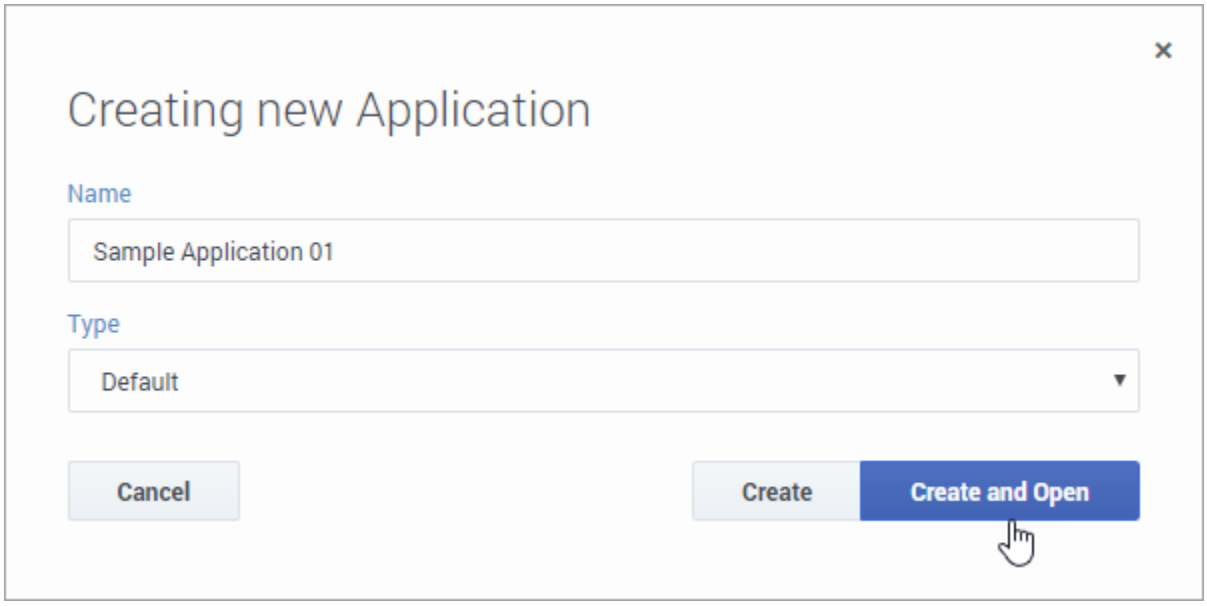
Applications

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

Go to the **Applications** page to manage your applications. From here, you can create new applications and make changes to existing ones.

Creating a new application

To create a new application, click **Add Application**. This opens the **Creating new application** window:



Enter a **Name** and select the **Type** of application that you want to build. You can select one of the following:

Default

Select this type to create an application that supports Interactive Voice Response (IVR) and voice call routing. After you create a default type application, you can enable it for digital interactions (i.e. chat) by enabling omnichannel support in the Application settings.

> **Important**
> When a default type application is enabled for omnichannel, it can service voice, chat, and SMS customers. However, note that the Menu block does not support SMS media.

Digital

Select this type if the application is only going to handle digital interactions, such as chat and/or email (no voice calls).

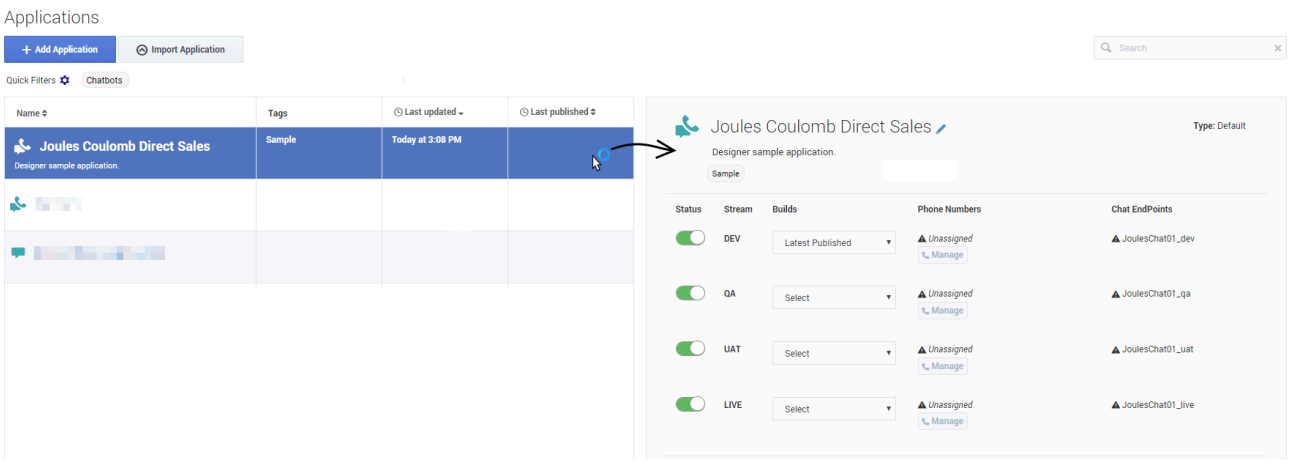
Callback

Select this type if the application is only going to be used for outbound callbacks.

Click **Create** to save it and go back to the **Applications** page. Or, click **Create and Open** to save the application and open it for editing.

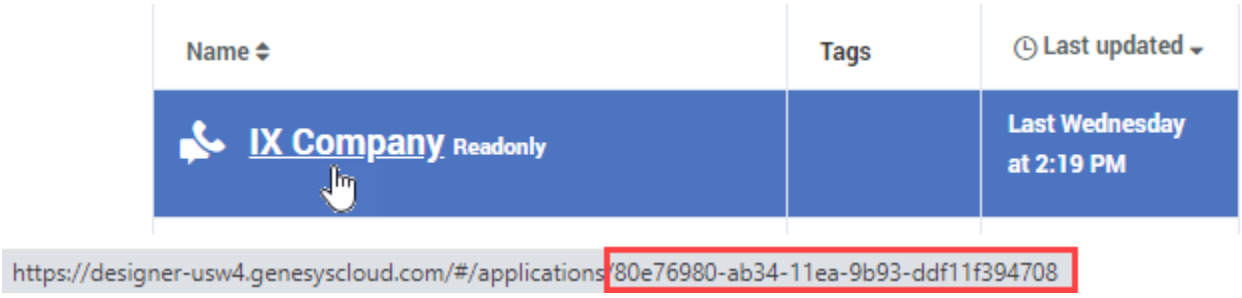
Viewing application properties

From the **Applications** list, click an empty spot in an application's row to display its properties on the right-hand side (if you click the application name, it will open for viewing or editing):



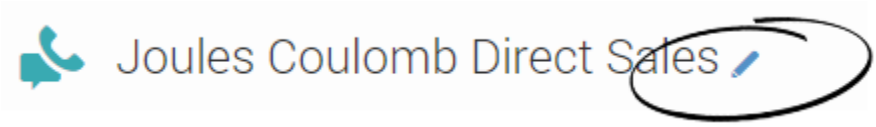
Application ID

Each Designer application has a unique application ID associated with it. You can view the application ID by mousing over the application link and checking the target URL that appears in the browser window:

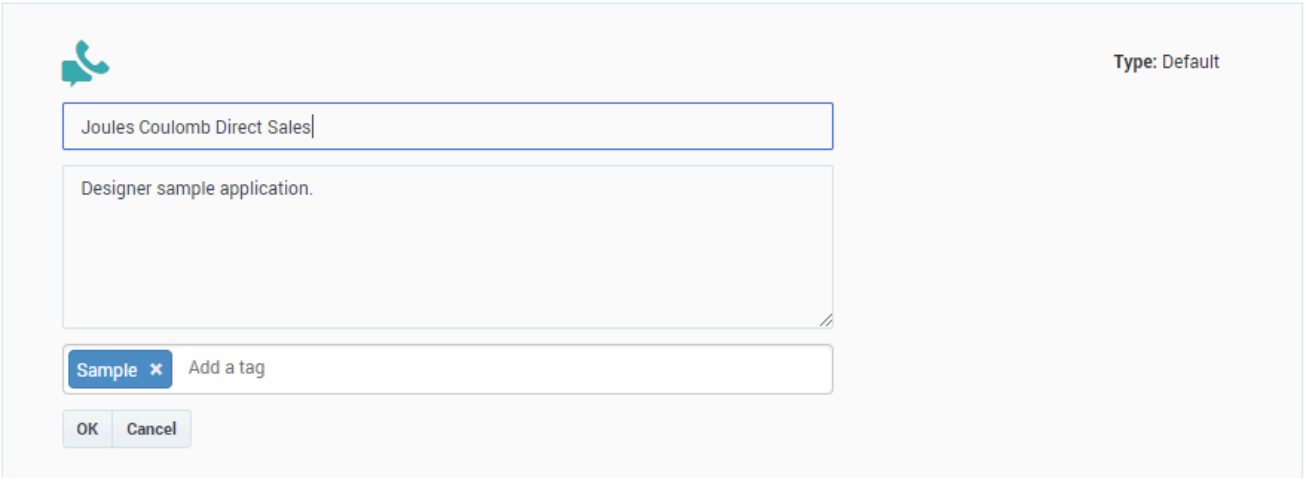


Edit the application name, description, or tags

Click the pencil icon to open the editing view.



From the editing view, you can change the application **name**, **description**, or manage the **tags** associated with the application.



> **Tip**
> You can also view or make changes to the application name and description by clicking the application name when its flow is open for editing.

Change the status of an application stream

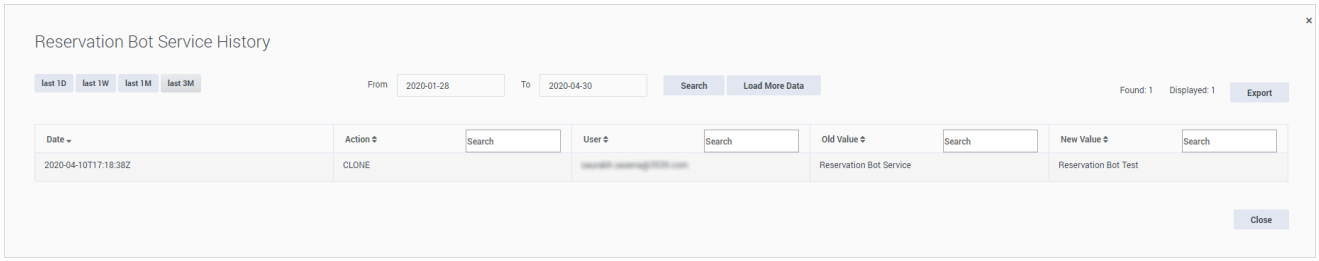
Use the sliders to change the status of an application stream to enabled or disabled.

To enable:

To disable:

View the application history

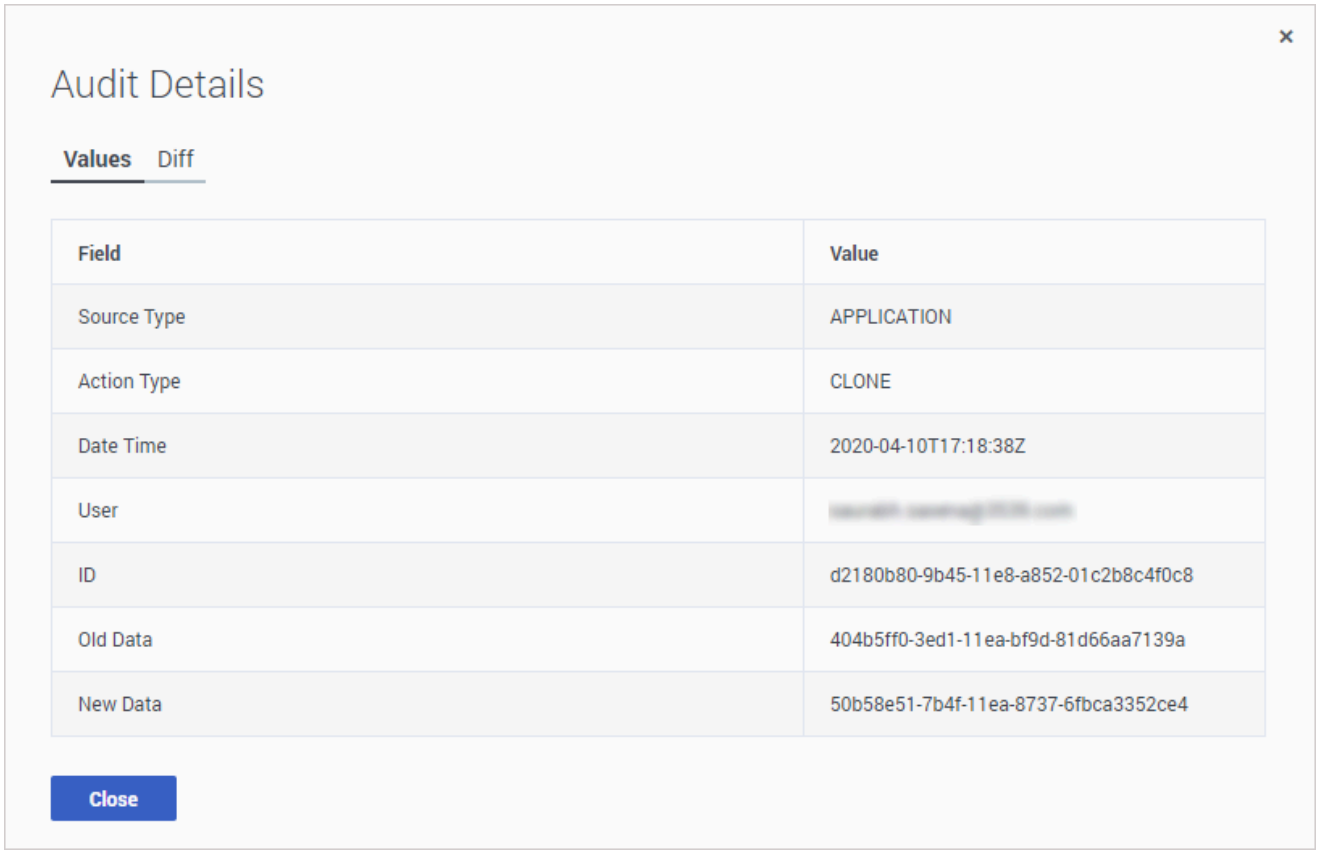
Click **Application History** to see a list of historical events for the application, such as each time it was viewed, edited, or published:



The history view shows you the date of each event, the action that was taken (for example, READ, EDIT, PUBLISH, CLONE, or DELETE), the user who made the change, and the new and previous value of any properties that were changed.

You can use the toolbar buttons to view the history for the last day, week, month, or three-month period. You can also specify a custom start and end date and click **Search** to generate the results. To download the results in a CSV file, click **Export**.

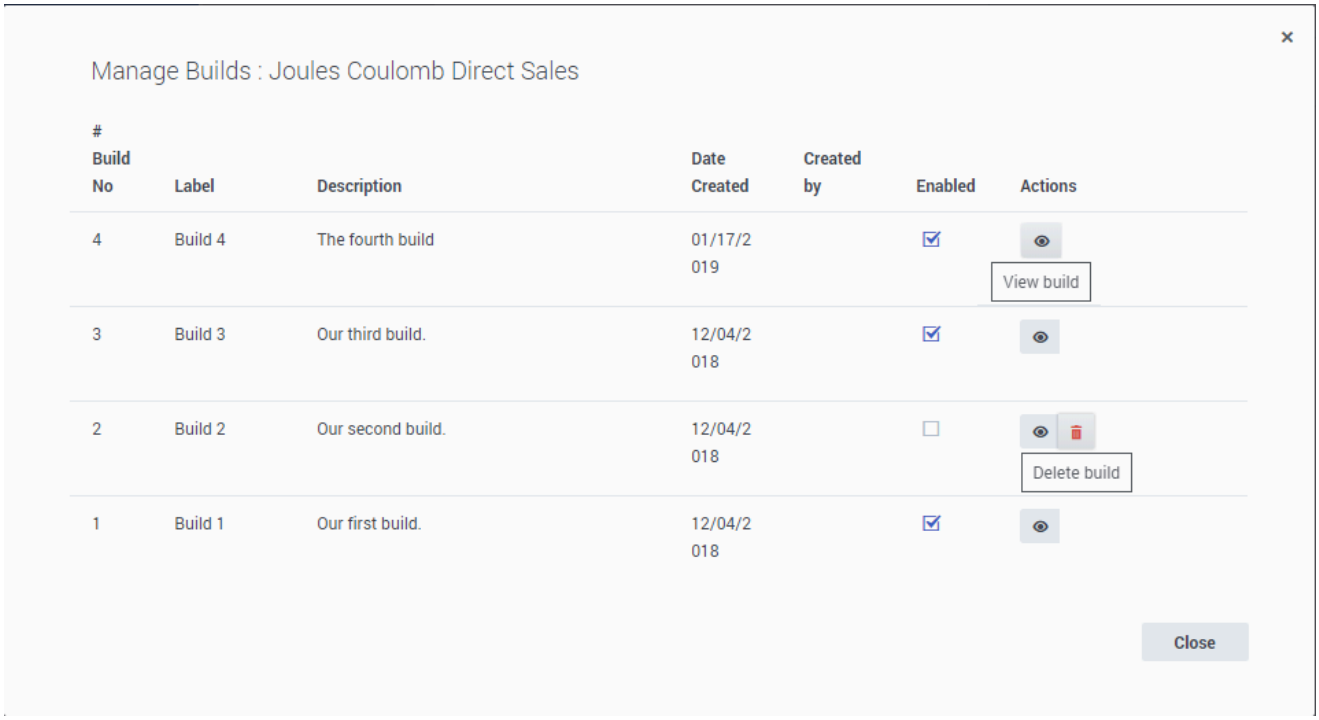
Double-clicking on an event opens an audit window that displays more details for that particular item:



Clone an application

Click **Clone application** to create a new application that is an exact copy of the selected application. Save the clone with a unique name and then manage it like any other application. This can be useful when you want to use the same structure as the selected application, but need to specify different settings or resources.

Manage Builds



Click **Manage Builds** to manage the application builds that are available for the application.

All builds are initially enabled by default.

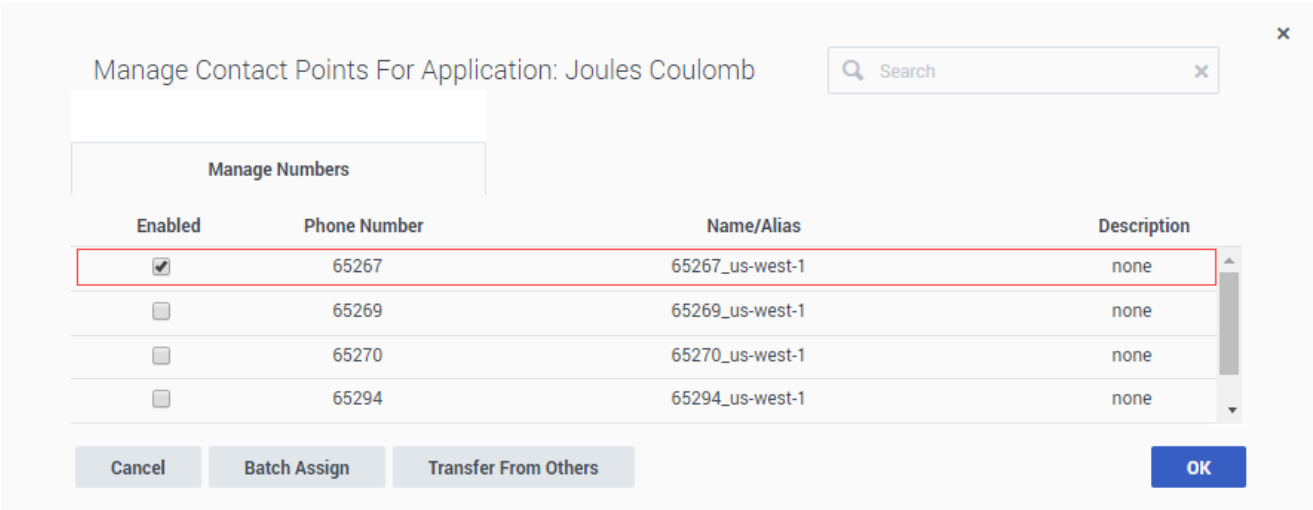
To disable a build, deselect it. After a build is disabled, you can click the trash icon to delete it.

If your user role has the required permissions, you can click **View build** to open the build in read-only mode. This can be useful when you want to review or compare the flows of different builds.

For more information about builds, see Application builds.

Manage Phone Numbers

Click **Manage** in the **Phone Number** column to select and assign a phone number(s) to the application. When you are finished, click **OK**.



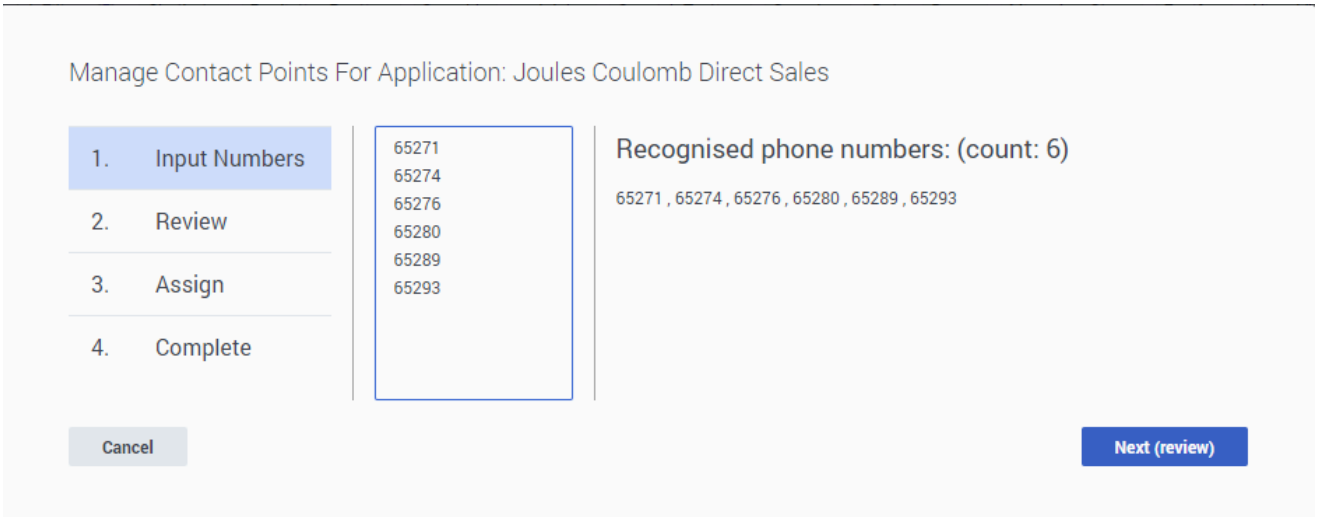
- You cannot assign a phone number to **IVR** type applications.
- You cannot assign a phone number to an application that has not been published at least once.
- To assign a large list of phone numbers, you can use the Batch Assign option.
- If you are moving phone numbers from one application to another, you can use the Transfer from Others option.

> ### Tip
> If you see a warning symbol beside a phone number that is assigned to your application, it might indicate that the phone number was not assigned by Designer or the application was updated but the changes were not published. The warning only indicates there is a possible problem with the phone number, not whether it is functional. You can ignore the warning if you are certain the phone number is functional. Otherwise, contact your Genesys representative.

Batch Assign

Let's say you have a large list of phone numbers that you want to assign to the application. With **Batch Assign**, you can copy and paste the phone numbers into a form and assign them in a single operation.

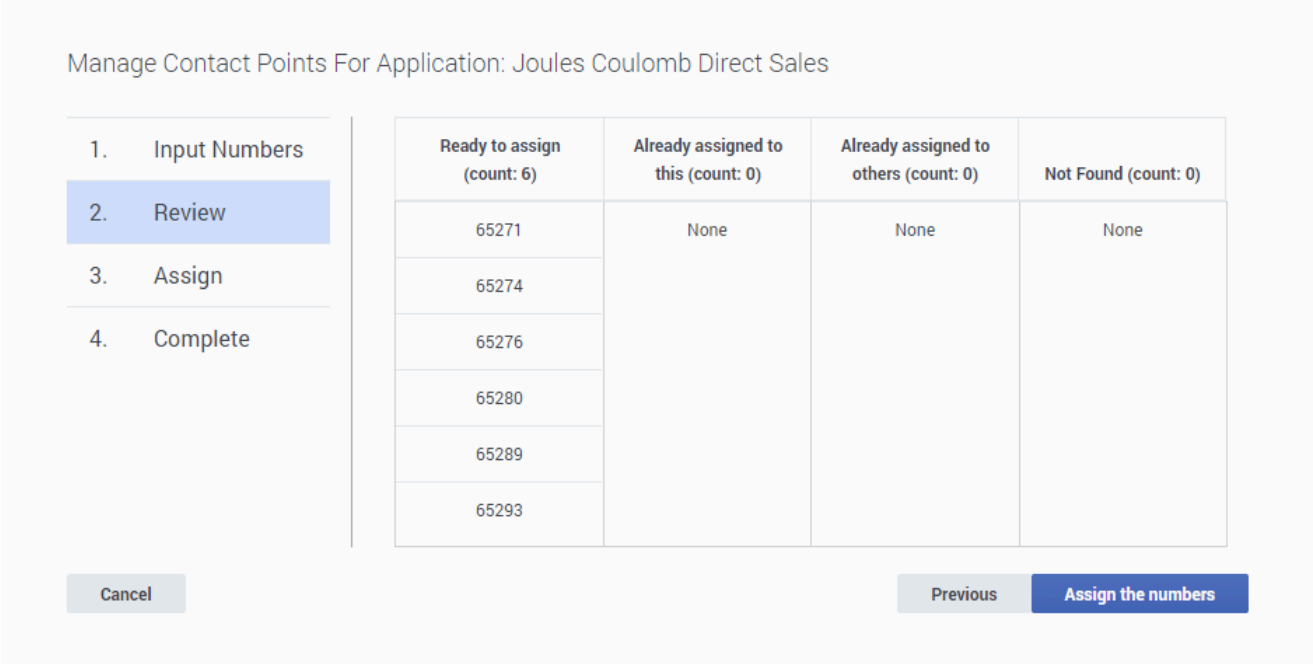
Here's how it works:



Input Numbers

Click **Batch Assign** and add the numbers you want to assign to the input column. Note that the numbers must be on separate lines. Designer keeps track of the numbers you've added in the **Recognized phone numbers** column.

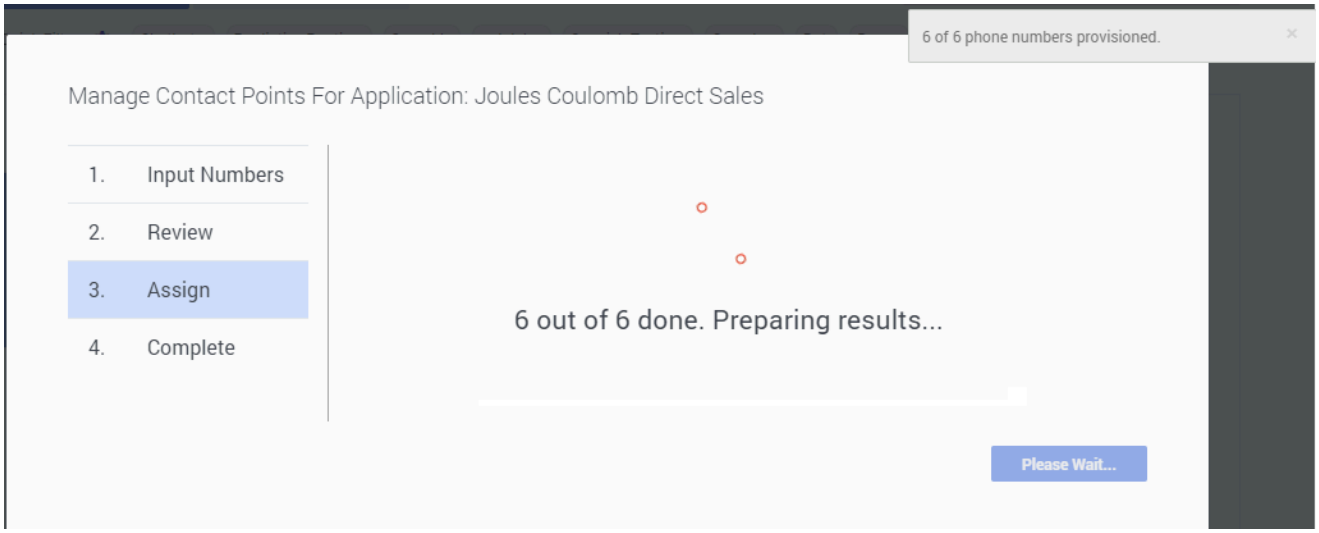
When you are finished adding numbers, click **Next (review)**.



Review

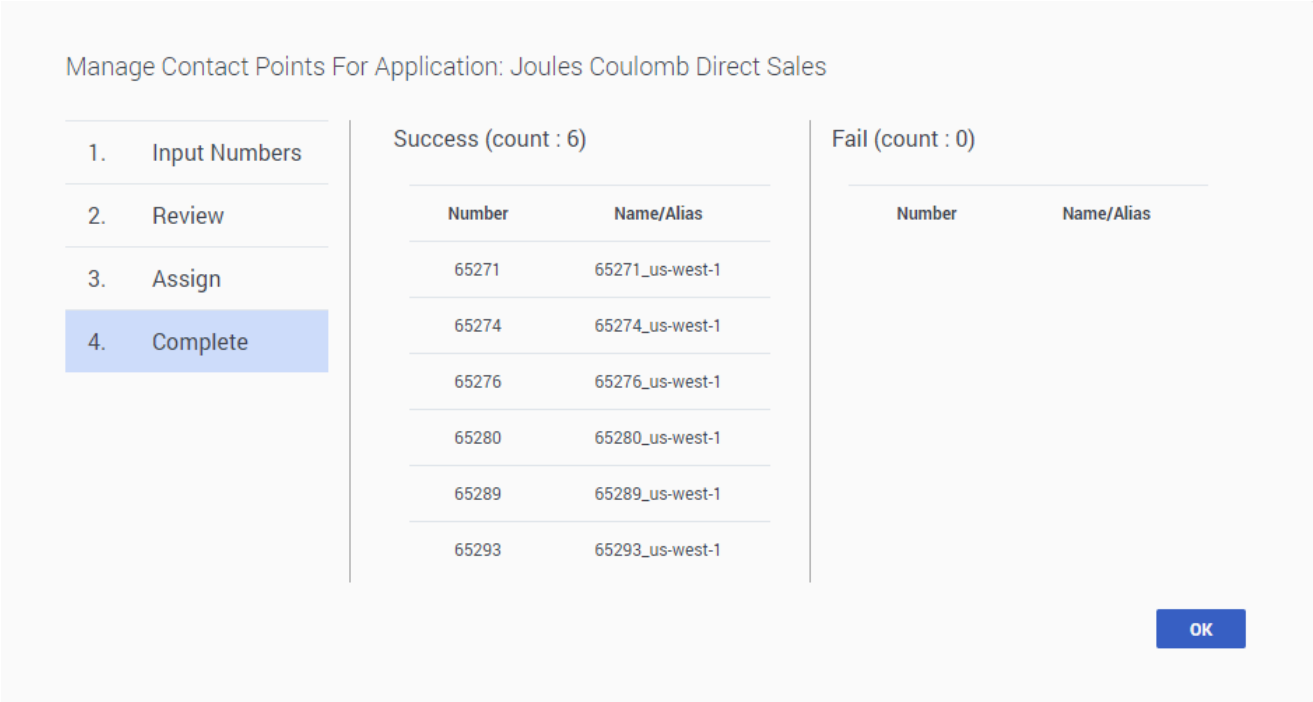
You can review the results to see the phone numbers that are ready to be assigned, are already assigned to this application (or another application), or were not found — maybe these were entered incorrectly or are not available for use.

If you need to make changes, click **Previous**. Otherwise, click **Assign the numbers**.



Assign

You can monitor the progress as Designer assigns the phone numbers to the application.



Complete

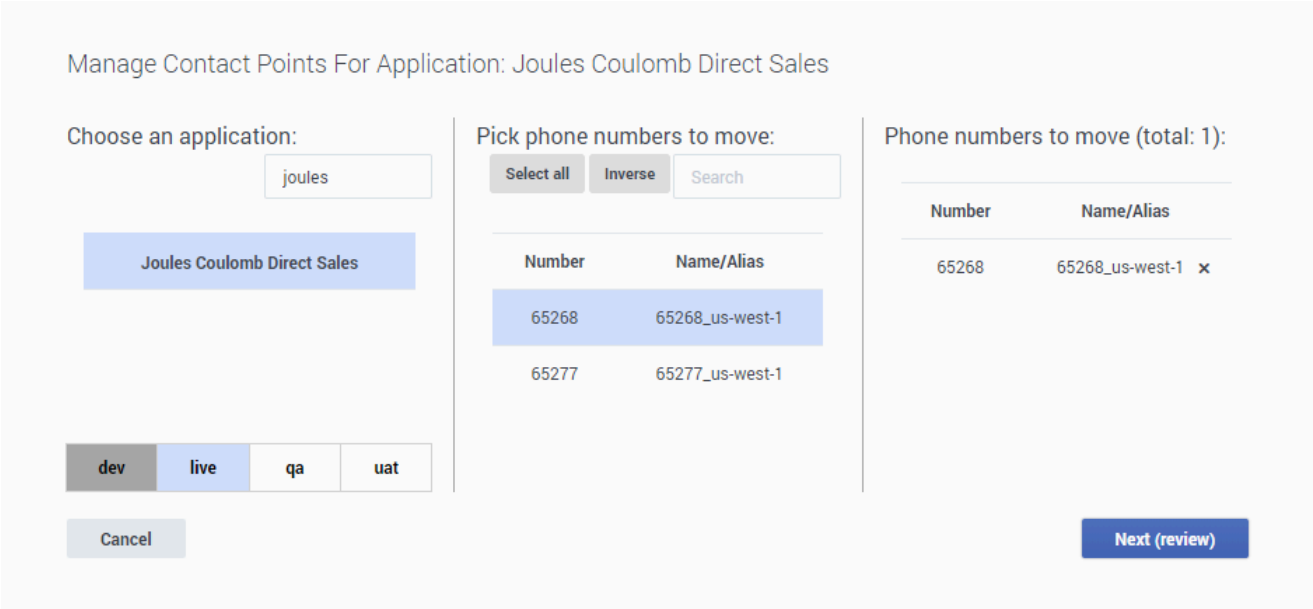
When the operation completes, Designer shows you which numbers were successfully assigned and which (if any) failed.

Click **OK** to close the window.

Transfer from Others

Use **Transfer from Others** to transfer multiple phone numbers to another application or to other stream of the current application. To show an example of how this works, let's transfer a phone number to another stream of an application.

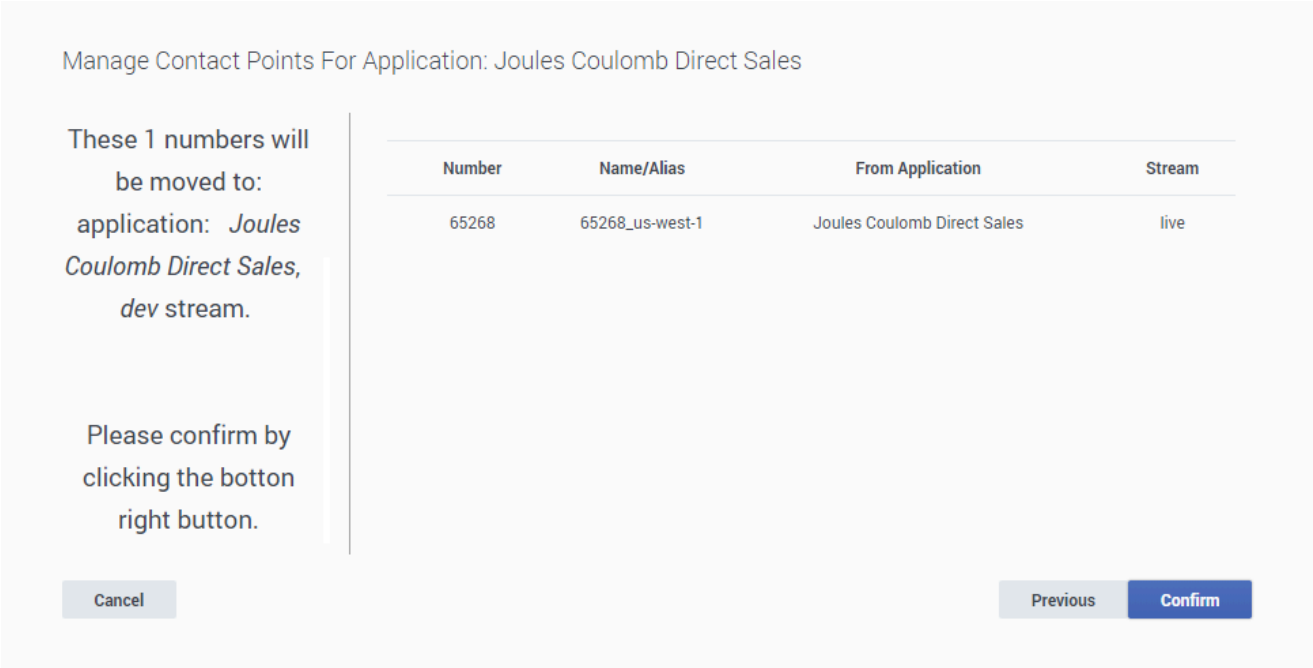
We've clicked **Manage Phone Numbers** for the Joules Coulomb application and selected **Transfer from Others**:



We want to transfer a number to a stream of the current application, so we use the search bar to filter the application list for "Joules". We select it, and the stream tabs appear at the bottom.

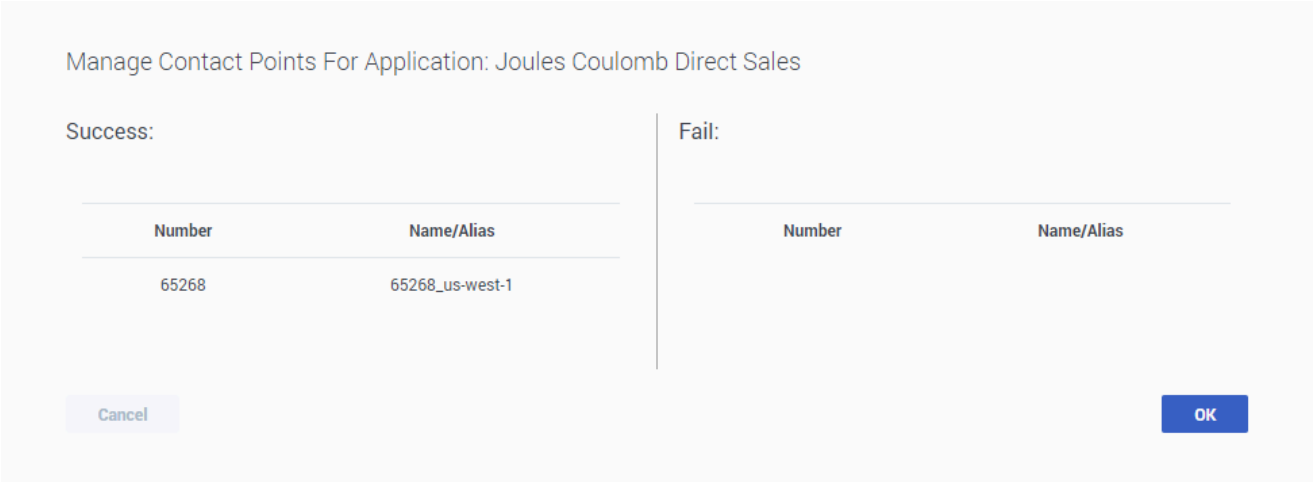
We want to move a number from LIVE to DEV, so we select the **LIVE** tab.

There are two numbers assigned to that stream, but we only want to move one of them. We select the number we want to move and click **Next (review)**.



Review the transfer details.

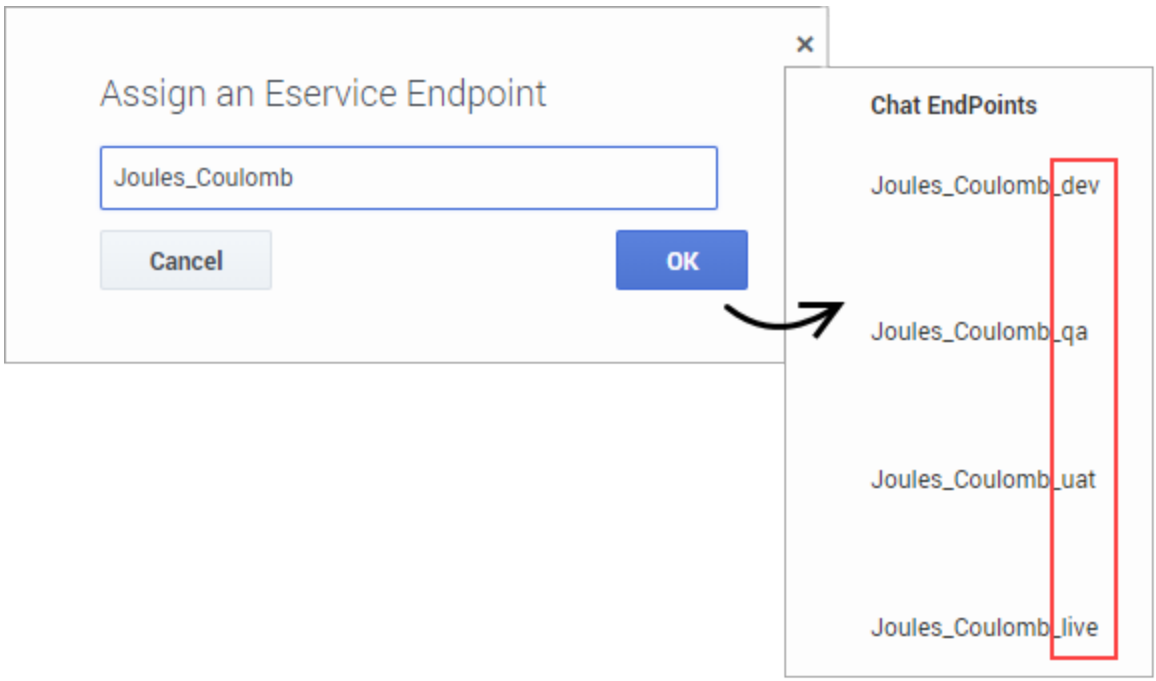
If everything looks ok, click **Confirm**. If you need to make any changes, click **Previous**.



Designer shows you the progress of the transfer. When it is complete, the summary lists the numbers that were successfully transferred and also any that failed.

Click **OK** to close the transfer results window and return to the application properties.

Manage Chat Endpoints



Click **Manage Chat Endpoint** to assign a chat endpoint to the application.

Note that the endpoint is assigned to each of the application streams, with a naming extension that corresponds to the stream.

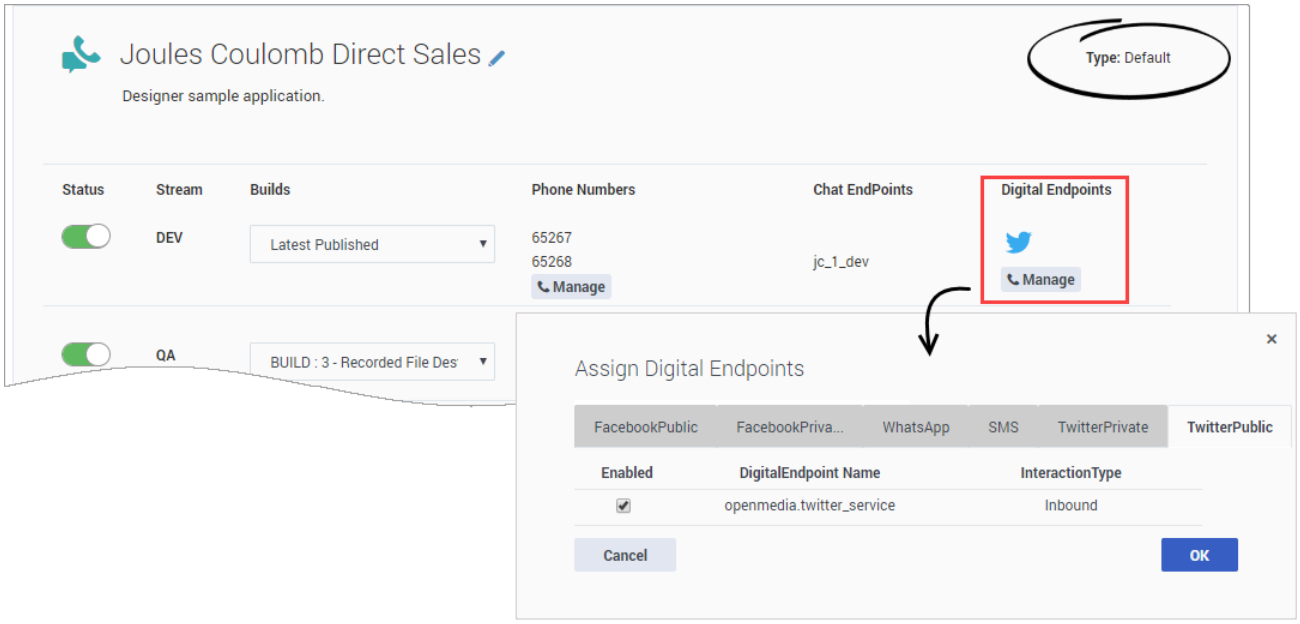
Some tips:

- Enter a name that is unique across all applications.
- The name should indicate the origination point of the chat (for example, *sales_page* or *mortgage_division*).
- Use alphanumeric characters only. Avoid using spaces or special characters (underscores are okay).

Manage Digital Endpoints

This option is only available for **default** application types.

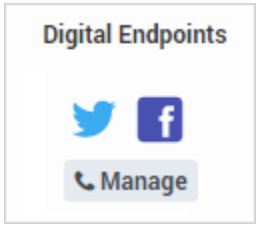
To assign a digital endpoint to a default application stream, click the **Manage** button under the **Digital Endpoints** column. (Digital endpoints are configured using Agent Setup. For more information, contact your Genesys representative.)



For example, you might want to add an endpoint for Facebook, WhatsApp, or Twitter, so that if a customer contacts you from one of these social media platforms or services, Designer detects which channel the message is coming from and launches the appropriate application for managing and routing that type of interaction. If applicable, Designer can also send or post messages back to the incoming channel.

Go to the tab for the appropriate service and select the endpoint to assign.

After you've assigned the endpoint, an icon for the respective service appears in the application properties, under the **Digital Endpoints** column:

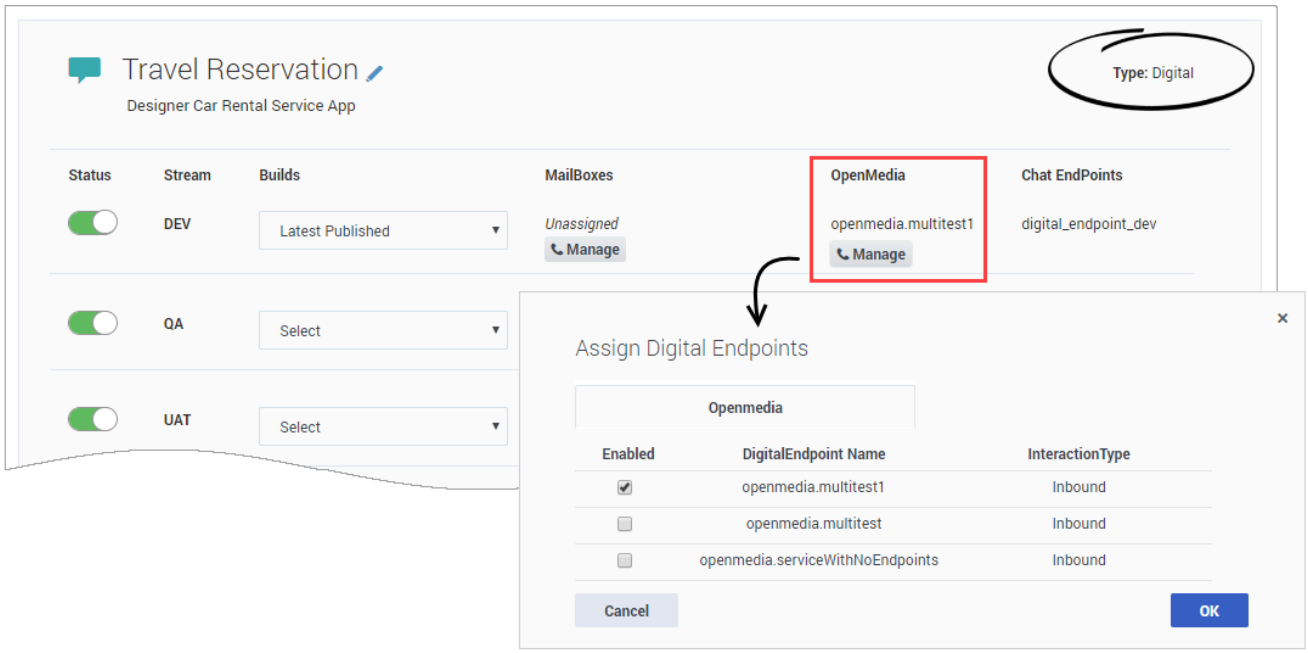


You can click the icon to view the details for the endpoint.

Manage OpenMedia Endpoints

This option is only available for **digital** application types.

To assign an OpenMedia endpoint to an application stream, click the **Manage** button under the **OpenMedia** column. (Digital endpoints are configured using Agent Setup. For more information, contact your Genesys representative.)



For example, you might want to add an endpoint for Facebook, Twitter, or WhatsApp, so that if a customer contacts you from one of these social media platforms or services, Designer detects which channel the message is coming from and launches the appropriate application for managing and routing that type of interaction. If applicable, Designer can also send or post messages back to the incoming channel.

Important

- Designer requires that OpenMedia endpoints contain the **openmedia** prefix (for example, **openmedia.banking**). Otherwise, Designer won't recognize the endpoint.

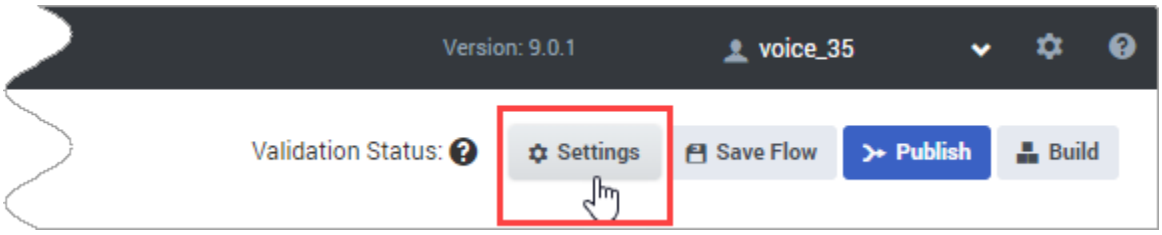
Disconnect Phone Numbers and Chat Endpoint

Click **Disconnect Phone Numbers and Chat Endpoint** to disconnect *all* phone numbers and the chat endpoint from the application (you might need to click **More** to see this option). Once disconnected, the phone numbers or endpoint can be assigned to other applications.

If you want to disconnect specific numbers from an application stream, click **Manage** under the **Phone Numbers** column for that stream to deselect the number(s) you want to disconnect.

Managing Application Settings

When an application is open for editing, click **Settings** to view and manage the Application Settings.



> **Important**
> Some of the features and settings described here might differ from what is available in your workspace. If you have any questions about the features and settings that are available for your workspace, contact your Genesys representative.

General tab

- **Application Reporting Title** — The name of the application to be used for reporting in Designer Analytics.
- **Application Version** — Specify the version number of this application. For example: `0.1`. You can then increase the version after making significant changes to the application.

Media tab

- **Media Resource Collection** - Select the Media Collection that this application will use. See the Media Resources page for more information.

Persona tab

If Persona functionality is available for your workspace, you can use this tab to enable the use of personas in the application. Personas are not enabled by default, so you'll need to enable this option for each application you want to use them in. For more information, see Personas.

Reporting tab

- **Milestone Path Prefix** - Specify a prefix to use with this application's milestone paths.

DTMF Options

This tab enables you to set global DTMF commands for your application. These DTMF keys can be used at any time within the application to trigger a specified action.

A common use case for this feature is a global command for the DTMF key **0** that routes the caller directly to an agent. In this example, you can set **0** as a global DTMF command that routes directly to the **Assisted Service** phase. In your application, you can add a Play Message block to announce that callers can press **0** at any time to speak to an agent.

Selecting **Enable Global Commands** enables global DTMF commands for the application.

To set a global DTMF command, select the drop-down menu beside the corresponding DTMF key that you want to use. In the drop-down menu, select a target block or phase for the DTMF key. Click **OK** when you are done setting global DTMF commands.

Global DTMF commands can target the **Self Service**, **Assisted Service**, or **Finalize** phase, or any block within the **Self Service** phase.

> ### Important
>
> - If the same DTMF key is also used by a block within your application, Designer first processes the command in the block.
> - You can also use global DTMF commands with **Self Service** type shared modules.

Speech Recognition tab

Configure settings for speech recognition (ASR). See the User Input block page for more information.

Global Retry tab

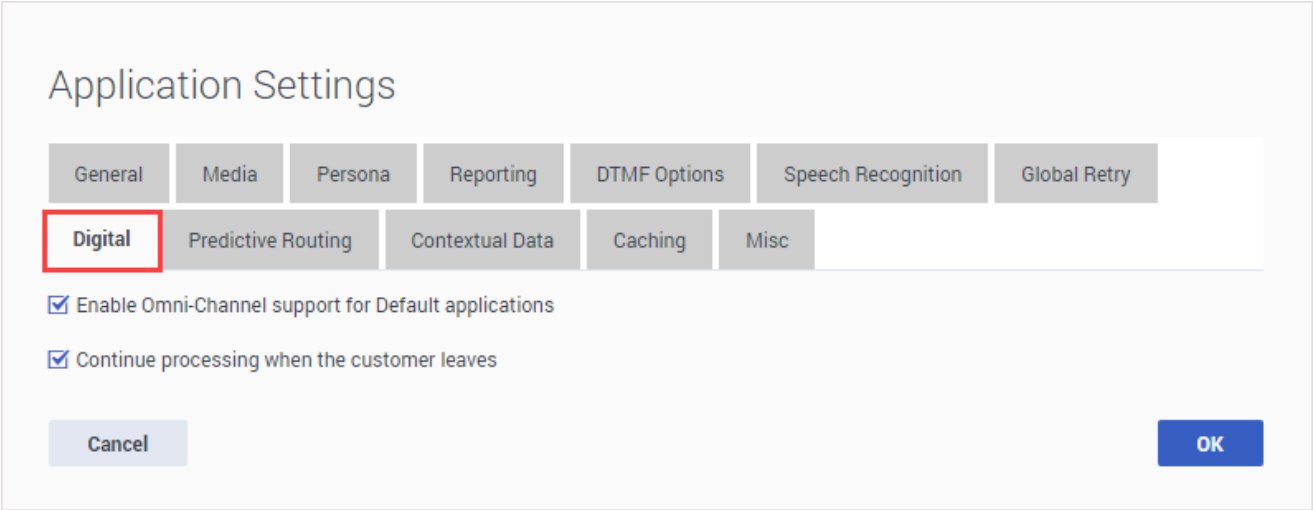
Configure global settings for menu retries. See the Menu block page for more information.

Digital tab

To enable a Default application type for chat interactions, select **Enable omni-channel support for Default applications**.

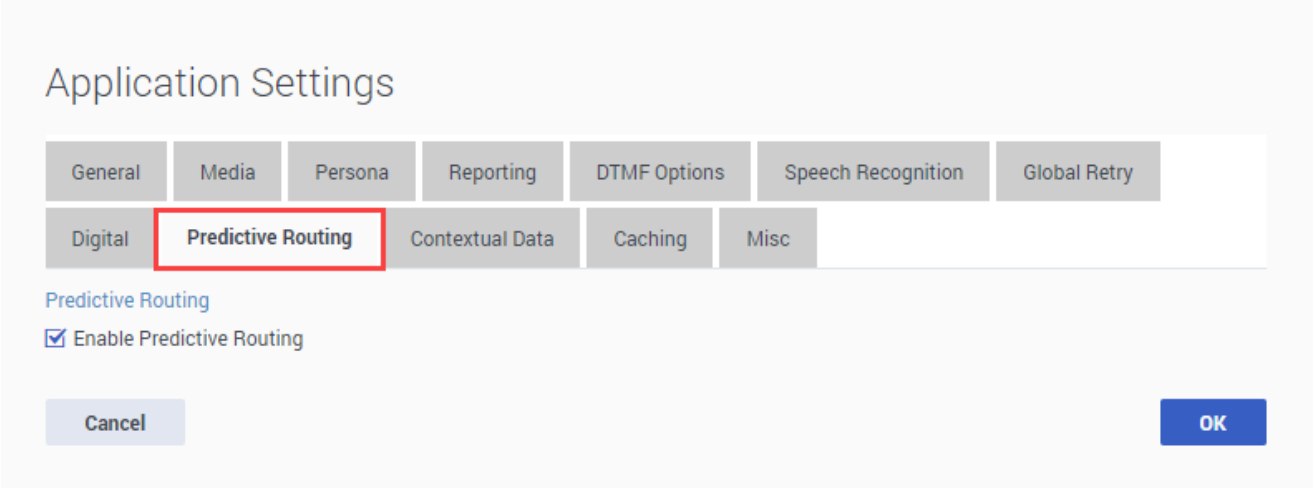
Select **Continue processing when the customer leaves** if you want Designer to continue processing the application after the customer leaves the chat session (this option does not apply to voice interactions). This option is only available for Default application types if omni-channel support is enabled.

> ### Important
>
> If **Continue processing when the customer leaves** is not enabled, Designer will terminate the chat session when the customer leaves and jump directly to the **Finalize** phase. This can interrupt any ongoing processing, including routing.



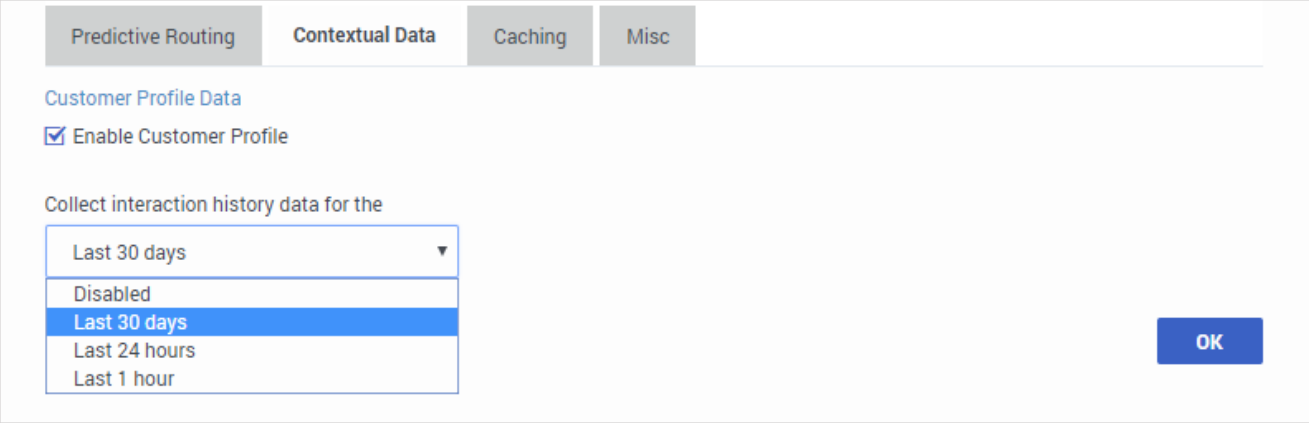
Predictive Routing tab

Select this option to enable support for predictive routing (as required for the Predictive Routing block).



Contextual Data tab

Select **Enable Customer Profile** to allow Designer to retrieve certain details about the customer, such as their name and contact information. When this option is enabled, you can also choose to select the amount of interaction history data that Designer should collect for the customer.



> **Important**
>
> This option must be enabled if you are using the **Last Called Agent** option in the Route Agent block.

Caching tab

Resource caching can improve overall system performance, but it can also cause a delay in how long it takes for changes in Designer to take effect (changes to Data Tables and Business Controls take effect immediately).

In most cases, it is best to leave **Use workspace cache settings** selected (it is already selected by default), as each application stage has optimized settings for how often a resource is checked. But you can disable it if you want the application to regularly check the cached resources for updates, and enter your own values for each resource.

The following resources are cached:

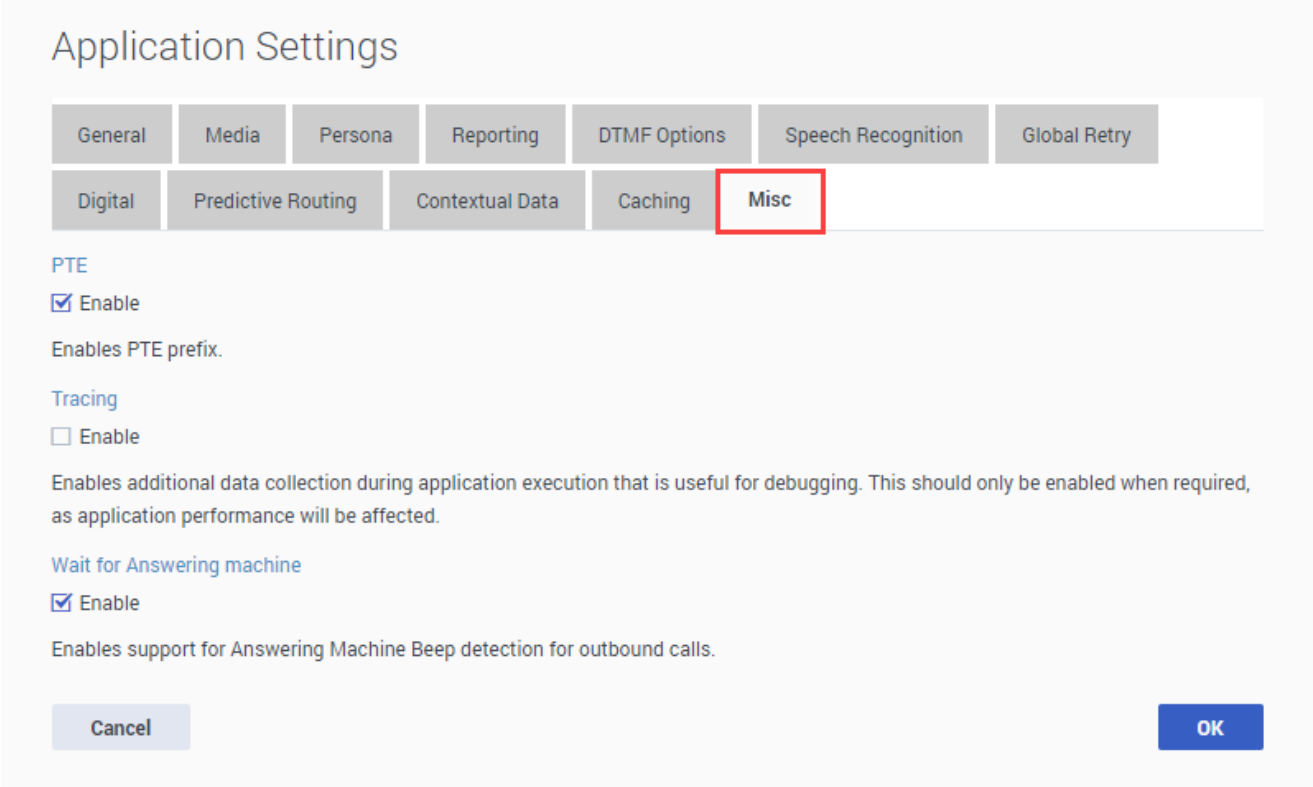
- Media Resources
- Speech Grammars

> **Important**
>
> Default values are set by Genesys for optimal performance. Although you can change these values, doing so might negatively impact application performance. Contact your Genesys representative for additional information that might apply to your environment.

Misc tab

The **Misc** tab contains options for enabling the Parallel Test Environment, Tracing, and Wait for

Answering Machine.



PTE (Parallel Test Environment)

Enable the **PTE** (Parallel Test Environment) option if you want non-production streams (DEV, QA, UAT) to use test versions of resources instead of the resources being used in the LIVE production environment.

For more information, see Parallel Test Environment (PTE) on the Application Workflow page.

Tracing

Enabling the **Tracing** option enables additional data to be collected while the application runs, which can later be used for debugging.

> **Important**
> This option should only be enabled when required, as it impacts application performance.

Wait for Answering Machine

This option tells Designer to wait for an answering machine beep (i.e. the tone that signals the caller to begin recording their message) during an outbound call. For example, if the customer does not answer the call, the application waits for an answering machine beep before playing a message.

Enabling Your Application

After you have assigned a phone number to your application, you can enable it by clicking the switch icon in the **Status** column. The switch icon turns green when the application is enabled.

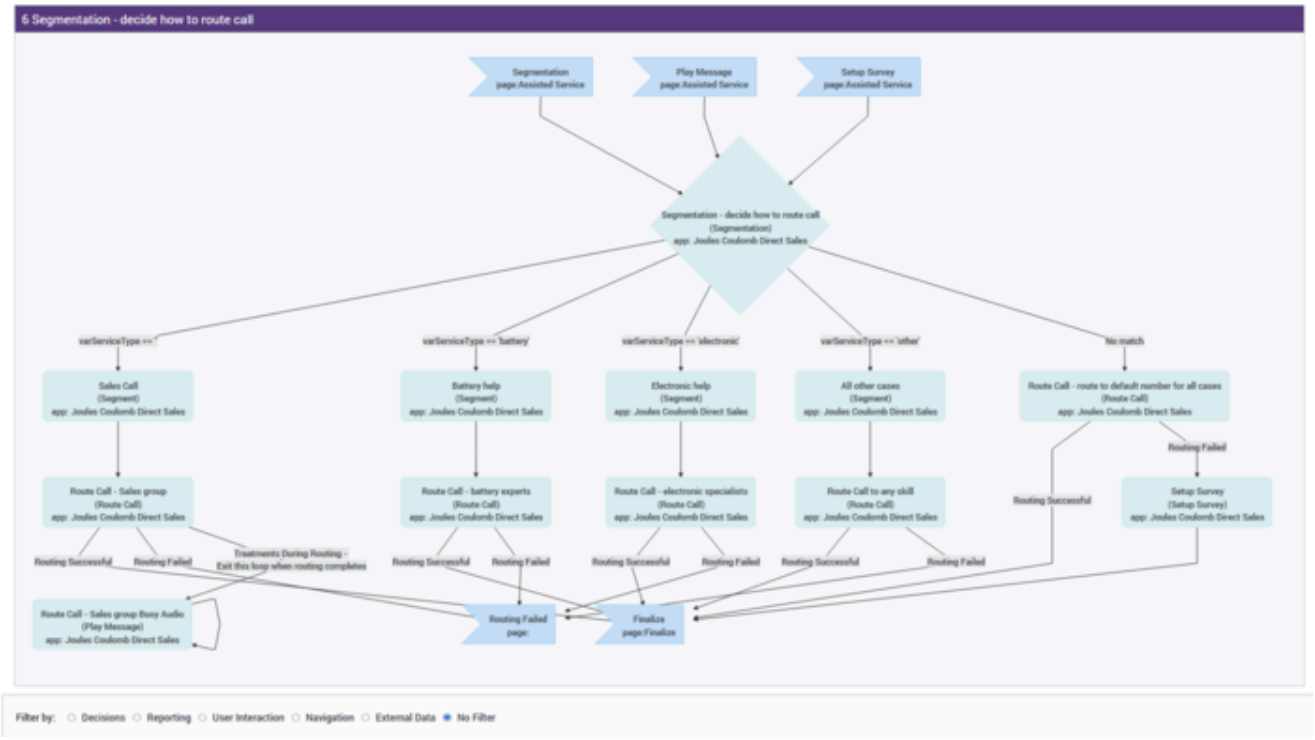
Viewing the application summary

When an application is displayed the flow section, you can use **Views** > **Summary** to generate a visualization of the application.

The resulting diagram is similar to a hierarchical flow-chart, with each node representing a block in the application flow:

The application summary view shows all the possible paths that an interaction can take through the application. The diagram is divided into sections for each application phase, and for nodes that need to be expanded into their own sections due to their size or complexity.

For example, this section shows a **Segmentation** node:



Filtering

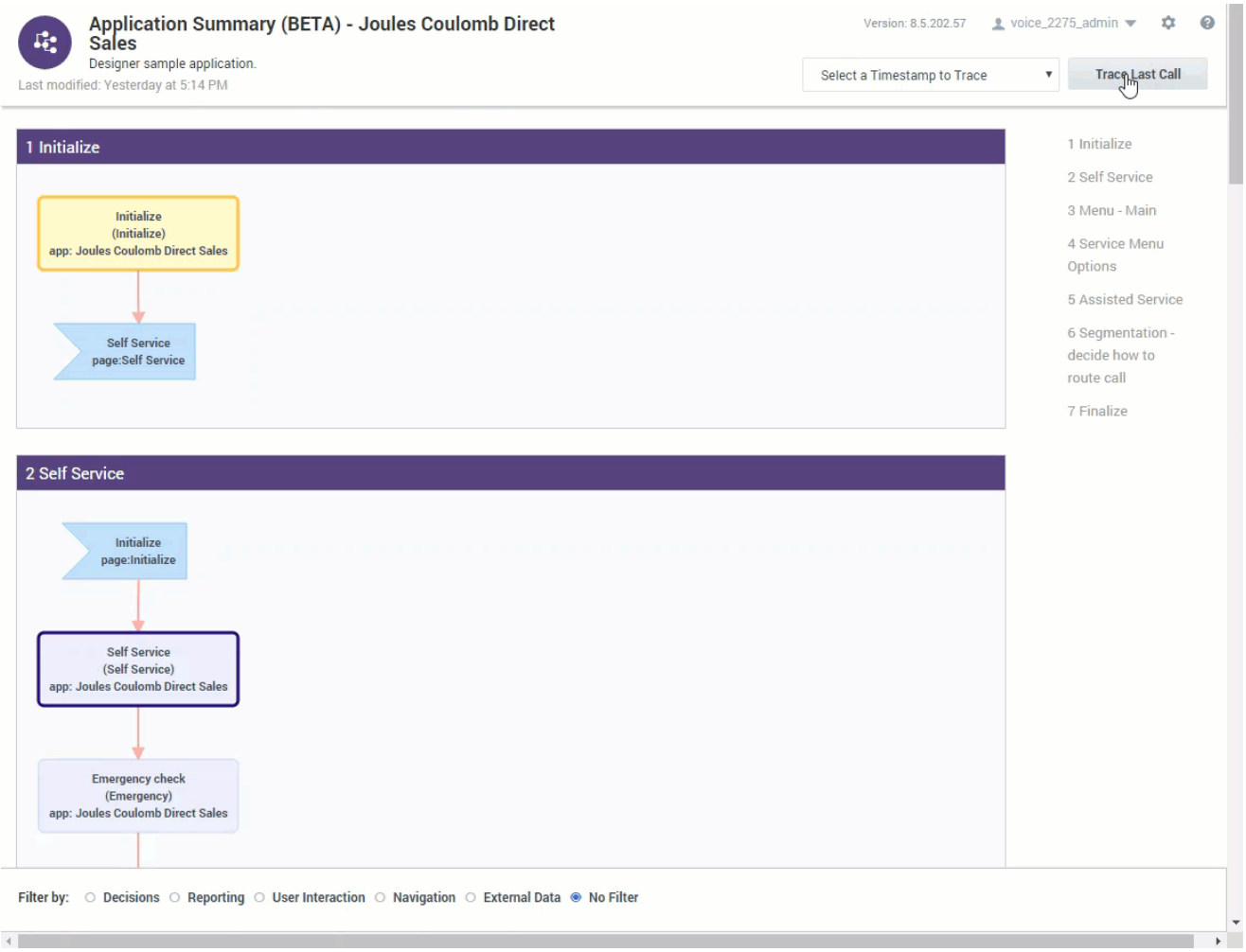
Use the **Filter by** options to focus on specific details. You can choose to filter the diagram by **Decisions**, **Reporting**, **User Interaction**, **Navigation**, or **External Data**.

Select **No Filter** to clear any selected filtering options.

Session playback

You can use **Select a Timestamp to Trace** to select and display the path that a specific session took through the application, or click **Trace Last Call** to load the path of the last session that was processed.

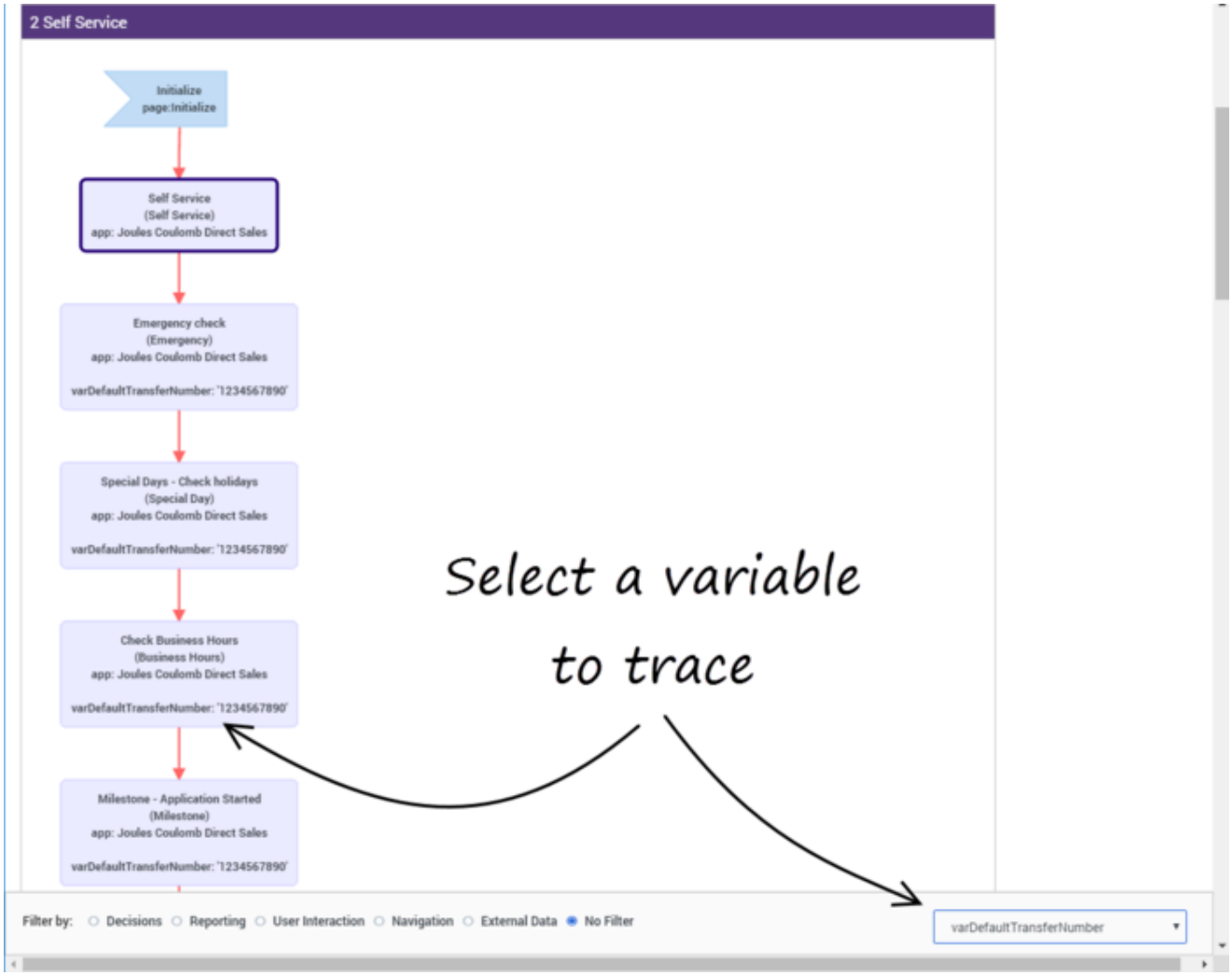
The path that the session took is indicated by red flashing connecting lines and highlighted nodes:



Click **Reset** to clear the playback details for the selected session.

Variable tracing

Likewise, you can also select a specific variable to trace:



This lets you track a variable as it moves and changes through the various nodes, which can be useful for discovering and resolving potential trouble spots.

Shared Modules

> **Important**
>
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

Click **Shared Modules** in the navigation bar to manage your shared modules. Shared modules are small pieces of applications that you can use in one or more applications. If you change a shared module, you are also changing all of the applications that use that module.

There are four types of shared modules:

- Self Service - Used within the **Self Service** phase of an application.
- Assisted Service - Used within the **Assisted Service** phase of an application, or within the **Initialize** phase with certain restrictions (see the note below).
- **Templates** - Used with the Callback block. These templates are read-only and cannot be edited or deleted. Click **Hide Templates** to hide these shared modules in the list.
- **Digital** - Used within Digital application types.

> **Important**
>
> - You cannot switch the type after you create the shared module.
> - Before you publish a module, you must ensure the module is adequately developed for use within applications. It should have well-defined input and output parameters that can be specified in the host application.
> - The **Route Agent**, **Route Call**, **Voice Mail**, and **Callback** blocks are not supported in **Assisted Service** type shared modules if the shared module is used in the **Initialize** phase.

Using Shared Modules

Modules can be used in different ways, depending on how you are planning your application. Currently, you can use modules with the following blocks:

Shared Module blocks

Larger application flows can often be difficult to manage. By dividing a larger application into smaller segments of individual flows, you can then "stitch" it back together using Shared Module blocks. This helps to make the flow size more manageable and also promotes reuse within and across applications.

Routing blocks (as "Busy Treatments")

You can use **Self Service** type modules to play *busy treatments* for callers (for example, play them some music while they wait to be connected with an agent). They can also be used to keep callers updated about their estimated wait times and, if the times seem excessive, offer them additional services (such as Callback). These modules will loop automatically until the call is routed, the caller hangs up, or the timeout specified in the routing block expires -- at which point the next block in the application is triggered.

- Learn more about routing blocks and busy treatments

Start Treatment block

The Start Treatment block also offers busy treatments to callers, but works a bit differently than the ones offered by routing blocks. Typically, you would use this block in the **Assisted Service** phase when you want to start a busy treatment (for example, play an audio file to callers while they wait to speak with an agent) and then move on to the routing blocks, without interrupting the playback to the caller.

- Learn more about the Start Treatment block and busy treatments

Creating a Shared Module

Click **Add Module** to create a new module. At this stage, it does not have a version number associated with it and it is not visible to applications. In the pop-up window, enter a name for the module in the **Name** field and specify in the **Type** drop-down whether this module is for the **Self Service** or **Assisted Service** phase.

When you are editing a module, you can perform the following actions:

- **Save Flow** - Save and validate changes to the module. Each save creates an incremented revision.
- **Create Version** - Create a new version of the module and make it available to applications. You must specify the following information:
  - **Version label** - The version number (for example, **1.0**).
  - **Notes** - Relevant information for use of this module.

Importing a Shared Module

You can import a shared module that was previously exported from another Designer workspace by clicking **Import Module**. Click **Choose file** to browse to the location of the archived file, then click **Upload**.

If everything looks ok, click **Confirm**. Designer will import all versions of the module that were exported to the archived file and upgrade any versions that need to be upgraded.

Settings

Click **Settings** in a Shared Module to access its settings.

General Tab

Managing Media Resources for Shared Modules

In the **Audio Resource Collection** drop-down menu, select **Inherits Audio Collection from Calling Context** if you want the shared module to inherit its audio collection from the host application. Otherwise, select an audio collection in the drop-down list if you want the shared module to only use a specific audio collection.

Example

You have one shared module that is used within two applications.

First, open the shared module and click **Settings**. In the pop-up window, select the **Audio Resource Collection** drop-down menu and choose **Inherits Audio Collection from Calling Context**.

Next, in the module's **Play Message** block, specify the type is **Announcement** and ensure the **Variable?** check box is checked. Choose a variable name (this example uses **greeting**).

Finally, in each application's Audio Collections, create an announcement called `greeting`. When playing the **Play Message** block, the call searches the inherited calling context audio collection for **greeting**.

Milestone Path Prefix

Specify a prefix to use with this application's milestone paths.

DTMF Options Tab

> **Important**

> This tab is only available for **Self Service** type shared modules.

This tab enables you to set global DTMF commands for your shared module. These DTMF keys can be used at any time within the shared module to trigger a specified action.

To set a global DTMF command, select the drop-down menu beside the corresponding DTMF key that you want to use. In the drop-down menu, select a target block for the DTMF key. Click **OK** when you are done setting global DTMF commands.

> **Tip**
> You can also set global DTMF options for Applications. In this case, when the shared module is running, global DTMF options are first processed within the shared module and then within the application.

Global DTMF Options Among Shared Modules

Your shared module (*Shared Module A*) might interact with another shared module (*Shared Module B*) that also has global DTMF commands. In this case, Designer processes DTMF commands in this order:

1. By block. For example, a DTMF command within a **User Input** block or a **Menu** block that expects this DTMF command.
2. By global setting in *Shared Module B*.
3. By global setting in *Shared Module A*.
4. By global setting in the host application.

If the DTMF command is not used by one of the above, Designer discards the command.

Other Actions

The following additional operations are available in the **Actions** column:

View module history

Click  to view the **Module History**. The history view shows you a list of each time the module was viewed, edited, or published, the user who made the change, and the new and previous value of any properties that were changed. The results can be sorted, filtered (for example, you can use the buttons to see only the history for a day, week, or month, or manually enter a specific start and end date), searched, and exported to a file. If you double-click an event row, an audit window opens that displays details for that particular event.

List module versions

Click  to **List module versions**. You can then choose to make a version **public**, or view a version (in read-only mode) as it appeared at a particular time.

List module consumers

Click  to **List module consumers**. This lists any applications that are currently using any version of this module.

> **Important**
>
> It is critical that you review the list of applications that use a module before the module is deleted. A module should not be deleted if it is used by an application.

Clone a module

Click  to **Clone the module** (similar to the cloning function for applications). Clone the selected module and save it with a new name.

> **Important**
>
> A cloned module does not inherit the history and published versions of the original module.

Export a module

Click  to **Export the module**. This exports the selected module for use in another Designer workspace. When you export a module, all versions of that module are exported, including the unpublished version.

> **Tip**
>
> If you are using a Safari browser to export a module, the exported file is downloaded as *unknown*. The file is valid and can be imported successfully, but you might want to rename it to something more meaningful.

Delete a module

Click  to **Delete the module**. This deletes all versions of this module. *Published* applications that already use the module (in other words, applications that have already generated their code) are not affected.

Media Resources

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

Click **Media Resources** in the navigation bar to manage your media resources.

Each **Media Collection** contains a set of **Media Resources**, which are TTS (Text-to-Speech) prompts and recorded audio files that you can use in any block in your application or shared module that supports playing audio, and also text and standard response messages that can be displayed in blocks that support chat prompts.

For example, a banking company might have a media collection assigned to an application that receives customer inquiries. Within that collection are several media resources for each language in which the company operates. Each media resource contains language-specific audio for greeting the caller, communicating announcements, and offering special promotions.

From this page, you can centrally manage these media resources for all of your applications. If you change a media resource, the change takes effect the next time that the media collection is published.

Designer also includes the following media collections:

- **Callback V2 Audio** contains media resources used with callback. You can modify the resources in this collection if you want to customize them for your own requirements. For example, you might want to add additional languages or replace the existing recordings with new ones.
- **Shared Resources** contains media resources that can be used in any application or module. You can make changes to this collection.
- **System Resources** is a read-only collection that is managed by Designer. You cannot make any changes to this collection.

See the Applications or Shared Modules page for more information on how to assign a media collection to an application or shared module.

Creating a media collection

To create a new media collection, click **Add Media Collection** and enter a name. When you are done, click **Create and Open** to open the new collection and add media resources.

To specify a media collection for an application or a shared module, go to the **Media** tab in the

application settings and select the **Media Resource Collection** you want to use. Each application or shared module can be linked to a single media collection. If you don't specify a collection for a shared module, it inherits the collection of the invoking application.

To improve media resources management, consider linking shared modules to their own media collections.

> ### Tip
>
> A media collection can't have more than 5000 prompts—if you need more than that for an application, organize the prompts into multiple collections. You can then split your application into modules, with each module referencing the appropriate collection.
>
> - Keep this limit in mind when planning your applications. It is much more difficult to go back and split an application later if this limit is reached.
> - Splitting an application into multiple shared modules lets you use multiple media collections in the application (each shared module can work with only one user-defined media collection).
> - **Important**: Designer won't warn you or display an error if there are too many prompts. But as more prompts are added, and the limit is reached or exceeded, your customers might start to experience quality issues due to the additional processing required.

Adding a media resource

After you've created a media resource collection, you can add media resources to it by clicking its link. You can then upload a single resource, or use the Import Bulk Audio option to import a zip file that contains multiple resources you want to add to the collection.

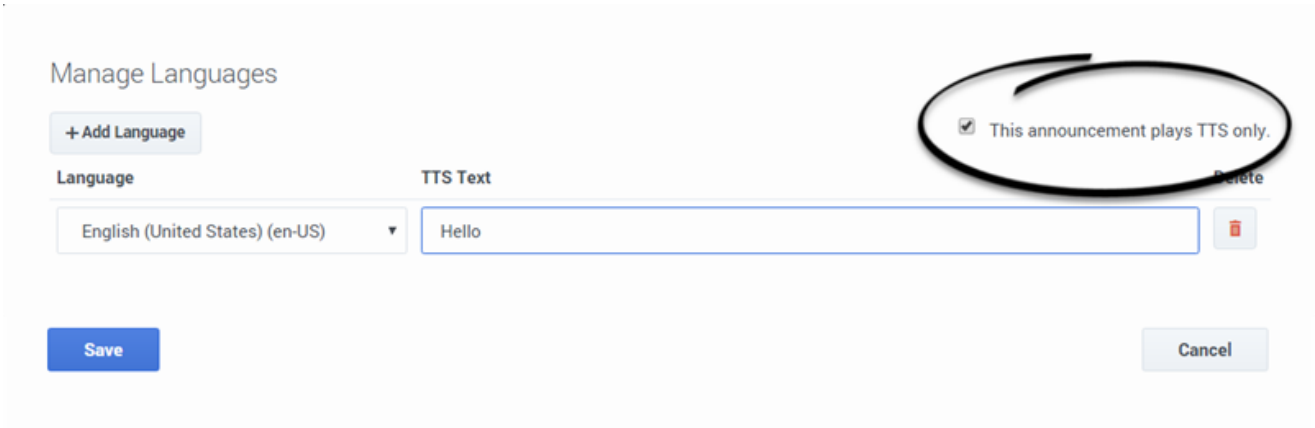
To add a single resource, click **Add Media Resource** and enter a name for the media resource. Make sure to use a unique name, as you won't be able to add it if it has the same name as an existing system resource. Click **OK**.

Properties of the new media resource are displayed to the right of the list. You can choose to add a **Description** to describe the purpose of this resource. You can also add **Tags** to associate this resource with media resources that share the same tags.

Click **Manage Languages** to continue setting up the resource. Select the language to be used for this resource (you can click **Add Language** if you want to add more). You can then set up the **Audio** and **Message** properties of the resource:

Audio properties

If this resource is only to be played as TTS (Text-to-Speech), select **This announcement plays TTS only**. In the **Alernate TTS Text** field, enter the text to be spoken by the TTS engine and click **Save**.



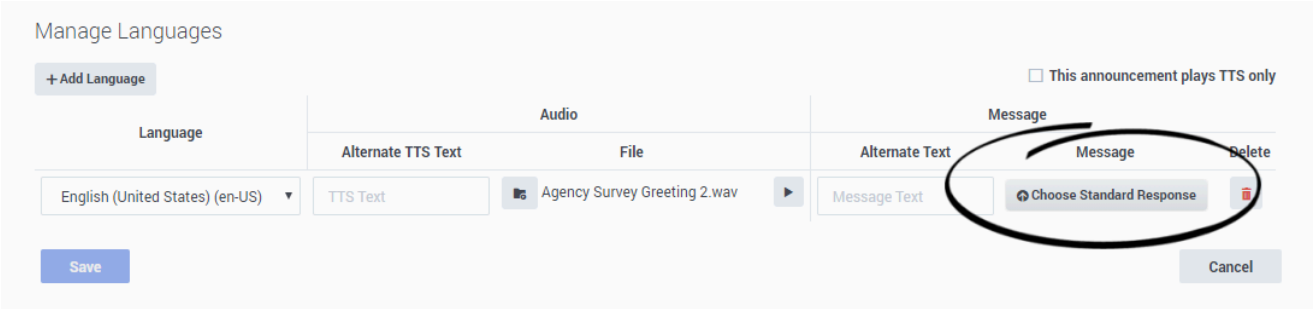
If adding an audio recording, make sure that the **This announcement plays TTS only** box is NOT selected. There are a few requirements for audio files:

- The file size must not exceed 10 MB.
- Files should be named as follows: *<media resource name>_<short language name>.<file extension>*. Example: welcome_en-US.wav. The file name must only contain alphanumeric characters, underscores (_), hyphens (-), or dots (.). File names with special characters are not supported.
- Audio files must be recorded in (or converted to) the G711 u-Law 8 kHz (also known as G.711Mu or G.711µ) audio codec. They must also be single-channel (mono) and saved as a .wav file.

Click the **File** button to select and upload your recording. A checkmark appears if the file is uploaded successfully. After you have saved the audio resource, you can click the play button to listen to the audio file.

Message properties

If the media resource is going to be used for chats, you can enter text to display in the **Alternate Text** field or select a **Standard Response** message.



Speech Synthesis Markup Language (SSML)

Designer supports the use of **Speech Synthesis Markup Language** (SSML) in TTS prompts. You can use these tags to control how TTS phrases are spoken by the TTS engine, such as reading back a prompt in a specific type of format or placing an emphasis on certain words or syllables. The tags use a syntax similar to HTML tags, in that they enclose the text value to be speech formatted in an opening and closing tag.

For example, the following prompt uses SSML tags to specify a *date* format:

```
<say-as interpret-as="date">1960/01/02</say-as>
```

The TTS engine reads this back as:

```
"January second, nineteen sixty."
```

Or, if you wanted to specify an *ordinal* format:

```
You are the <say-as interpret-as="ordinal">5</say-as> customer in the queue.
```

The TTS engine reads this back as:

```
"You are the fifth customer in the queue."
```

When enabled for an application, SSML handling is in effect for any block that plays or consumes TTS messages, such as the Bot (includes responses received from the bot service), Menu, and Play Message blocks.

TTS service providers might not support all SSML tags, so you'll need to check with your provider to confirm which ones you can use.

SSML settings

You can make changes to the SSML settings in the application system variables. For new applications, SSML handling is enabled by default, but you can set the **enableSSML** variable to `true` (enabled) or `false` (disabled), as desired.

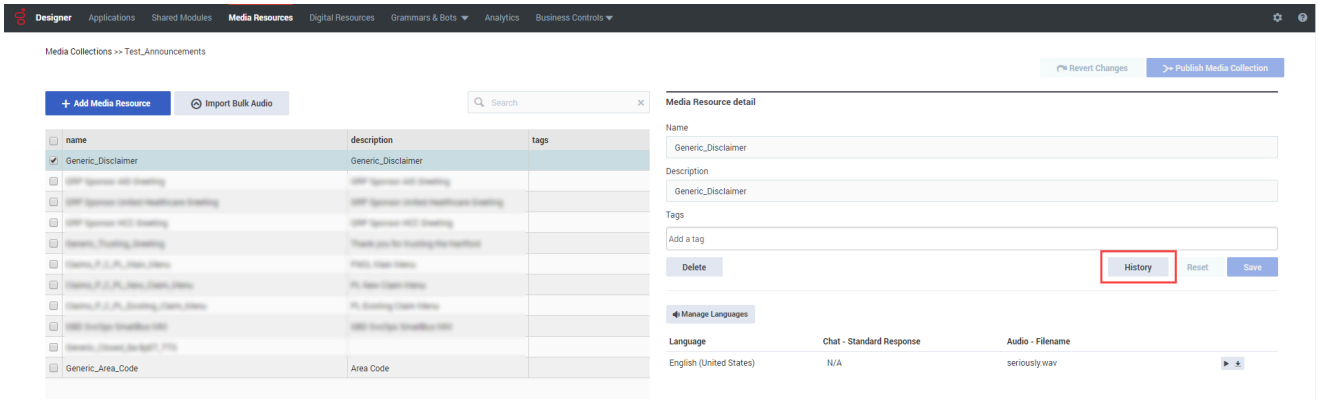
If your application is using the same TTS prompts and messages for both voice and chat channels, you can set the **removeSSMLInChat** system variable to `true`. This removes the SSML tags from prompts when they are displayed to customers during chats.

> **Important**
>
> Designer does not validate SSML tags. If you are using them in your prompts, make sure that you are using the correct syntax and that the tags are supported by your TTS services provider.

To learn more about SSML standards, see the Speech Synthesis Markup Language page at the World Wide Web Consortium (W3C) website.

History view

To view the history of a media resource, select it and click **History**.



The history view shows you a list of each time the resource was viewed, edited, or published, the user who made the change, and the new and previous value of any properties that were changed. The results can be sorted, filtered (for example, you can use the buttons to see only the history for a day, week, or month, or manually enter a specific start and end date), searched, and exported to a file. If you double-click an event row, an audit window opens that displays details for that particular event.

Import Bulk Audio

This option enables you to upload a package of media resources to a collection. This can be useful if you've exported a collection to make changes—maybe you've added new resources, made new audio recordings, or updated the properties of several resources specified in the associated CSV file—and now you want to refresh the collection with the changes.

From the **Media Resources** page, select the media collection you want to add resources to. Click **Import Bulk Audio**, select the zip file that contains the resources you want to import, and click **Submit**. Designer checks the zip file to see if there are any errors, and lets you know if anything needs to be corrected before it imports the resources into the collection.

To learn more about working with CSV files for media collections, see Offline management (CSV file).

Publishing a media collection

Media collections must be published before they can be used by applications. When you are finished adding media resources, click **Publish Media Collection** to publish the collection. If you want to undo the changes you have made, click **Revert Changes**. This will discard any changes made and restore the collection back to the last published version.

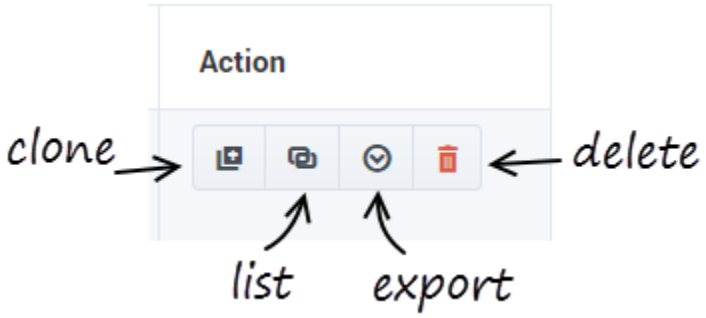
> **Important**
> If you make changes to a media collection, it must be re-published for the changes to take effect in the applications using those resources.

After you click **Publish Media Collection**, Designer opens the **Publish - Media Collection difference** window, where you can use the **Added**, **Deleted**, and **Modified** tabs to review the changes made to media resources since the last time the collection was published.

When you are ready to publish, click **Publish** to publish only those items in the collection that have been changed. Select **Publish Entire Media Collection** if you want to publish all resources contained in the collection.

Managing media collections

Use the **Action** toolbar to manage your media collections:



Clone a media collection

Creates a copy of the media collection while preserving the original. For example, you might want to

use a media collection in another application but need to make a few modifications to its resources, such as using an alternate media file or language.

List media collection consumers

Displays a list of the applications that are using the media collection. This lets you see which applications would be impacted if you make any changes to the collection.

Export a media collection

Use this option to download a zip file containing all of the media files in the collection. The zip file includes a CSV file that you can edit to make offline changes to the media collection before importing the collection back into Designer.

Delete a media collection

Click the trash icon to delete a media collection. Designer will ask you to confirm this action, and display a list of applications and modules that are using any of the media resources contained in the collection.

Offline management (CSV file)

You can export a media collection to make changes to it offline. The collection can then be imported back into Designer.

When you export a media collection, a CSV file is included with the exported zip file. You can use an application like Microsoft Excel to edit this file and make changes to the media collection. For example, you can add new resources, mark existing resources for removal, or modify certain resource properties.

There are certain rules to follow when working with the CSV file:

- Don't make any changes to the spreadsheet schema. For example, do not make changes to the column headings or re-arrange the columns.
- When adding a new resource, you can add a new row. However, do not add (or remove) any columns. Also, do not remove rows for existing resources — there is a proper way to remove resources (see Removing resources from a collection).
- All mandatory fields must contain a value before the audio collection can be imported.
- Do not remove or rename any of the extracted files or folders.
- Make sure any *new* media file resources you are adding have been placed in the **media** folder before preparing the zip file and uploading the collection to Designer.
- Items in CSV files are separated (or delimited) by commas. If you need to use a comma within a value, you must enclose it in double-quotes (","").

CSV file fields and descriptions

Field	Description
Mark Media Resource For Deletion	**Mandatory**. Indicates if the media resource should be removed from the collection. By default, all values are set to FALSE. If you change this value to TRUE for a resource, it will be deleted during import.
Audio Resource ID	**DO NOT MODIFY**. This value represents the unique ID generated by Designer for each audio resource. If you are adding a new audio resource, leave this field blank.
Audio Resource Name	**Mandatory**. Name of the audio resource. You can modify this value.
Audio Resource Description	Description of the audio resource. You can modify this value.
Audio Resource Tags	Tags used to group the audio resources for easy identification. Tags should always be enclosed within double-quotes ("tag"). Multiple tags can be separated by a comma, but kept within the double-quotes: "tag1, tag2, tag3".
Mark Language For Deletion	**Mandatory**. Indicates whether the language should be deleted from an audio resource. By default, all values are set to FALSE. If you change this value to TRUE for a language, it will be removed from the associated audio resource during import.
Play Only Text	**Mandatory**. Indicates whether the audio resource should play only text. If set to FALSE, the resource plays as an audio file. If the audio file cannot be played, the text specified in `Text to Be Played` is played.
Audio Resource Language	**Mandatory**. Indicates the language that the audio file/text supports. The language can be any one of the languages supported by Designer.
Audio File Name/Text to Play (if Play only Text is true)	**Mandatory**. If `Play Only Text` is set to TRUE, enter the text to be played. If `Play Only Text` is set to FALSE, enter the name of the audio file to be played. Make sure that the audio file specified here is contained within the **audio** folder.
Text to Be Played	Indicates the text to be played if there is issue playing the audio file. Enter the TTS text to be played.

Adding new resources to a collection

To add a new resource to a collection, add a new row to the CSV file. Specify values for all of the mandatory fields, and any optional fields as desired. Do not enter a value for **Audio Resource ID** as this value is added by Designer after import.

If you are adding an audio file resource, make sure that the file is placed in the **audio** folder before performing the import.

Removing resources from a collection

To remove an audio or language resource from a collection, change the value of the **Mark Audio Resource/Language for Deletion** field to TRUE.

Supported Audio Formats in Browsers

The types of audio files that you can play is dependent on the web browser that you are using. The following table shows which audio file formats are supported by each browser.

> **Important**
> The table below applies to your browser only and does not indicate whether the file format is supported by the Genesys platform. It is important to note that the Genesys platform is responsible for playing the audio resource when you run an application.

Format	Chrome	Firefox	Internet Explorer	Safari
mp3	Yes	Yes	Yes (IE 9 and later)	Yes
wav (16 bit mono)	Yes	Yes	No	Yes
wav (u law)	Yes	No	No	Yes
vox	No	No	No	No

Intelligent Prompts

As an alternative to TTS, intelligent prompts play back prerecorded audio files that are read to the customer using human-sounding audio, so that the prompt sounds similar to natural human speech.

They can be useful when you need to arrange a prompt to be presented in a format that is specific to a region, such as a *currency* denomination or a date standard. For example, you might need to specify **Euros** instead of **Dollars**, have the day spoken before the month, or you might want a number to be spoken as an *ordinal*, where a number such as "21" is spoken to the customer as "twenty-first".

Intelligent prompts can be used with the following:

Item	Description
Alphabetical characters	For example, **"P"**.
Cardinals	Speaks a number as a quantity. For example, **1234** would be spoken as **"one thousand, two hundred, thirty-four."**
Currencies	Speaks a number as a specified currency denomination. For example, in the U.S. English (US-en) locale where dollars ($) are used, **11234** would be spoken as **"eleven thousand, two hundred and thirty-four *dollars*."**
Days	Speaks the specified day of the week. Days are numbered from 0-7, starting with Sunday. For example, **1** would be spoken as **"Monday"**.
Months	Speaks the specified month. Months are numbered from 1-12, starting with January. For example, **5** would be spoken as **"May"**.
Ordinals	Speaks a number as an ordinal. For example, **21** would be spoken as **"twenty-first"**.

For intelligent prompts, Designer includes support for the following languages:

- English, Australian (en-AU)
- English, American (en-US)
- Spanish, Mexican (es-MX)
- French, Canadian (fr-CA)
- Punjabi, Indian (pn-IN)
- Chinese, Hong Kong (zh-hk)

> **Important**
> If Designer is not able to play an intelligent prompt in the caller's preferred language, it will play the prompt in American English (en-US).

Personas

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

Choosing a persona for your application can provide customers with a more personalized experience.

What is a persona?

Personas are artificial "voices" that you can use to control Text-to-Speech (TTS) services in your applications. They are designed to sound more natural and conversational than the "robotic" voice typically associated with IVRs, and portray certain personality characteristics or traits that customers might find more appealing or appropriate. For example, you might choose the persona of a formal-sounding male in his 40's for one particular type of application or scenario, and a cheerful female in her 20's for another.

Designer includes the following persona types:

- Samantha (female, 30-40s, professional, calm) – this is also the default persona
- Tom (male, 30-40s, polite, professional)
- Gabriela (female, 20-30s, fresh, engaging)
- Michael (male, 20-30s, curious, geeky
- Diane (female, 40-50s, soothing, silky)
- David (male, 40-50s, professional, confident)

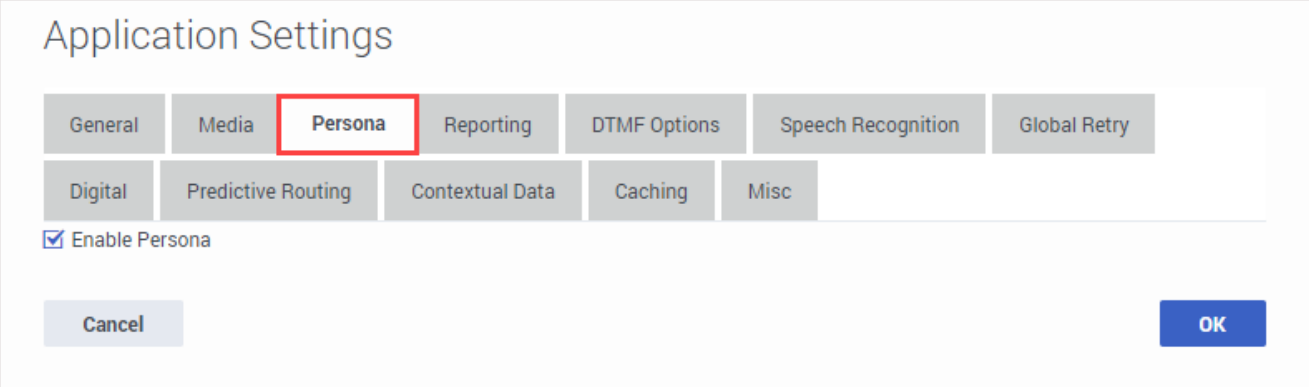
Your site might also be set up to use additional personas that have been customized for your particular business needs.

Depending on how you want to use personas, you can set the related System Variable to apply one globally to the entire application, or use a Change Persona block to switch to a different persona at a specific point in the application.

Enable personas

Personas are not enabled by default. To enable personas for an application, go to the Persona tab in

the application settings and select the **Enable Persona** check box.



After personas are enabled, you can select a persona for the application.

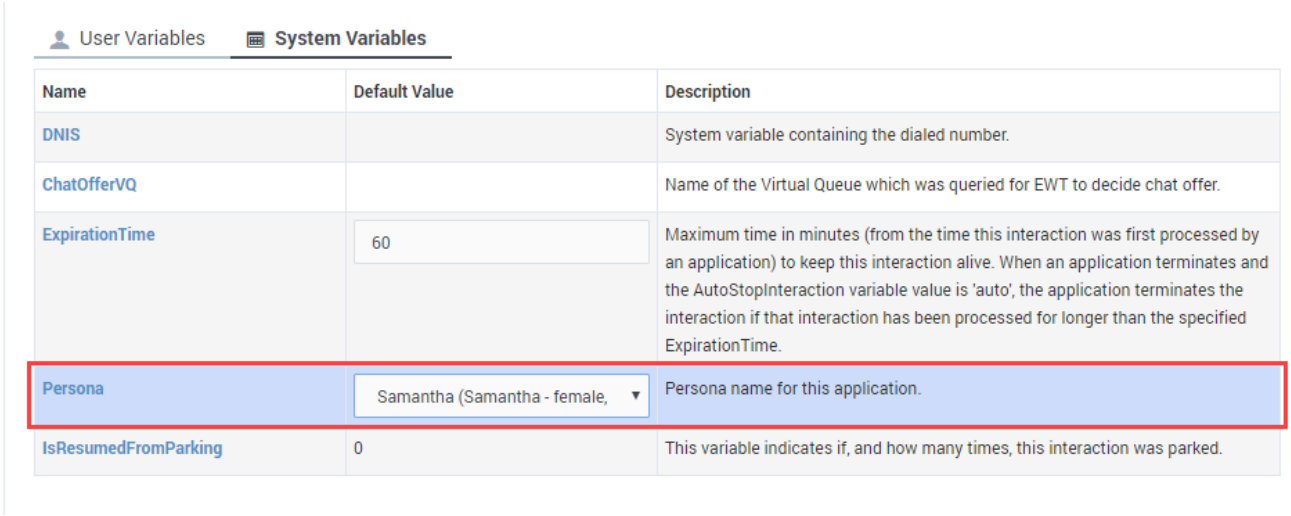
Disable personas

If you want to disable personas for an application, clear the **Enable Persona** check box. Any personas specified by the **Persona** system variable or **Change Persona** blocks will then be ignored and all TTS services will revert to the default voice that is provided by that service.

Select a persona

To set a persona that will be used globally throughout the entire application, go to the System Variables tab in the **Initialize** phase. Locate the **Persona** variable and select the one you want from the list of available personas.

For example:



The persona that you select will be used by any blocks that are using Text-to-Speech (TTS) services, such as **Play Message**, **User Input**, **Menu**, **Bot**, **Route Call**, and **Route Agent**. If desired, you can change the persona at various points in the application.

If personas are enabled for the application and you don't select a specific persona (or choose **use default** from the list) the default persona is used. Typically, the default persona is "Samantha", but this can vary depending on your configuration.

Change the persona

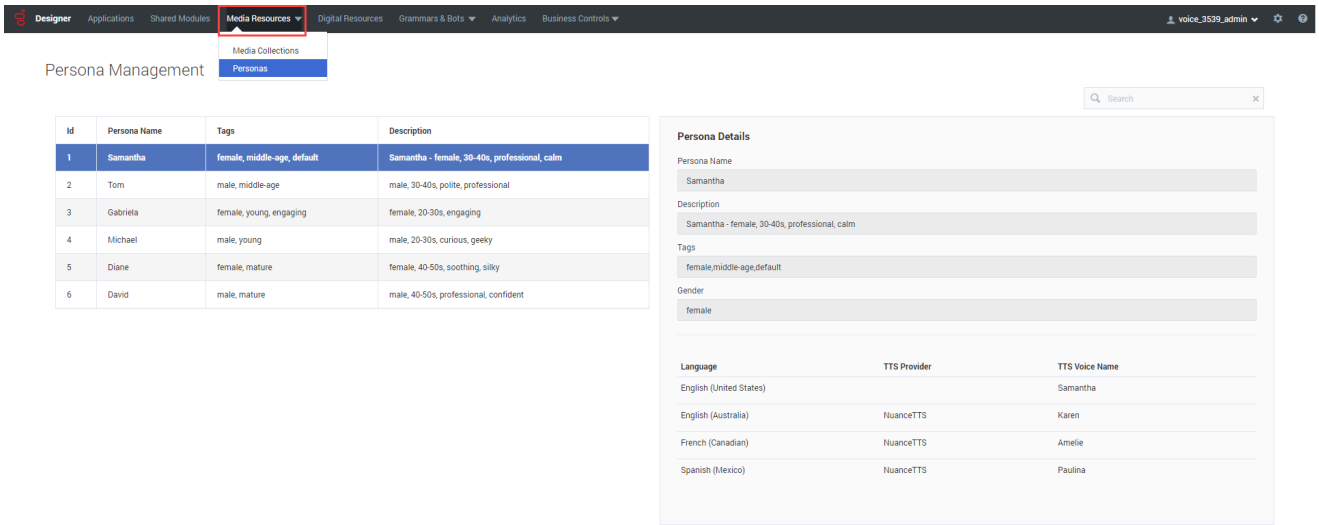
You can use a Change Persona block to dynamically change the persona. You can add this block to the Self Service or Assisted Service phase of your application or use it in a shared module. The new persona takes effect with the next prompt that is played.

Switching personas can be useful when you want to use certain personas for specific situations. For example, you might want to use a more formal persona when dealing with sensitive customer issues, or switch to different personas depending on the customer segment being served ("Gold" customers get one type of persona, "Blue" ones get another).

For more information about using this block, see the Change Persona block page.

Managing personas

You can view a list of available personas on the **Media Resources** > **Personas** page. From this page, you can select a persona to view additional details about it, such as the **Language**, **TTS Provider**, and **TTS Voice Name**.



What happens if the language changes?

If the language of the application is changed (such as by a Change Language block), the TTS "voice" continues to use the selected persona. For example, if you are using the persona of a young female and the language changes from English to Spanish, the voice will still use that type of persona. If there is no voice available that matches the selected persona, an alternate voice or TTS service is used for that language.

Digital Resources

> **Important**
>
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

Digital Resources are predefined standard responses and user-defined messages that you can use in digital applications and shared modules. A **Message Collection** is a collected set of individual message resources that can be accessed by a digital application.

From this page, you can centrally manage these message resources for all of your applications. If you make any changes to a message resource, the change takes effect the immediately across affected applications.

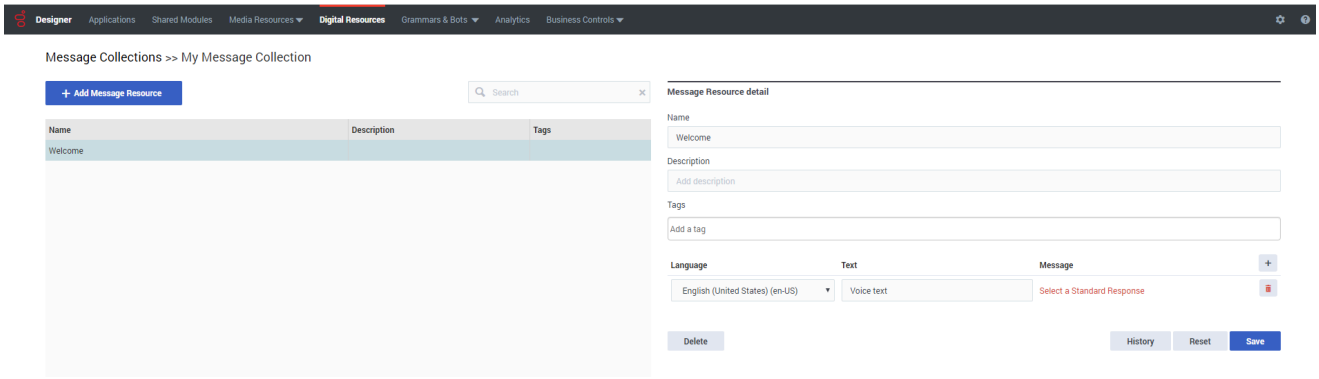
(See the Applications or Shared Modules page for more information on how to assign a message collection to an application or shared module.)

Creating Message Collections and Message Resources

To create a new message collection, click **Add Message Collection** and enter a name. When you are done, click **Create and Open** to open the new message collection and add message resources.

Next, click **Add Message Resource** and enter a name for the message resource. Make sure to use a unique name as you won't be able to add it if it has the same name as an existing system resource. Click **OK** to save the new message resource.

Selecting a message resource enables you to view and make changes to the resource properties:



You can:

- Add a **Description** for the resource.
- Add some **Tags** to associate the resource with similar resources.
- Add a **Language** to create a custom **Text** message or select a standard response **Message** from the Standard Response repository.
- Use the **History** button to view the history of the digital resource.

History view

To view the history of a digital resource, select it and click **History**.

The history view shows you a list of each time the resource was viewed, edited, or published, the user who made the change, and the new and previous value of any properties that were changed.

The results can be sorted, filtered (for example, you can use the buttons to see only the history for a day, week, or month, or manually enter a specific start and end date), searched, and exported to a file.

If you double-click an event row, an audit window opens that displays details for that particular event.

Grammars & Bots

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

From the **Grammars & Bots** menu you can manage your Speech Grammars and the Bot Registry.

Speech Grammars

Click **Speech Grammars** to upload and manage speech grammars for use in your applications. Genesys Designer supports **voice** and **dtmf** speech grammars in either **SRGS** (GRXML) or **SLM** format.

A speech grammar defines the list of phrases or options that the caller can input when they use your application. You might use a **voice** speech grammar to tell your application which words or phrases might be used by the caller, to help the application determine how the call should be routed.

For example, an automotive company might upload a speech grammar that contains phrases that a caller might use when they contact the company, such as "I want the parts department" or "I am interested in buying a new car." Your application can use these phrases to determine the best routing target for this customer.

Creating a Speech Grammar

Click **Add Grammar** to create a speech grammar. In the pop-up window, enter a name for the speech grammar and click **OK**.

You can now upload your speech grammar file. In the **Grammar detail** area, click the **Language** drop-down to select the language of the speech grammar. Next, click the **Choose file to upload** button that appears when you hover over the **No file** text. Choose a file to upload.

After the speech grammar file has uploaded, you can click to view the contents of the grammar file in a read-only window.

Next, you can tag the speech grammar or enter a description. You can also specify the format and mode of the speech grammar.

> **Important**
>
> Genesys Designer does not validate the contents of the speech grammar. The **Grammar Format** and **Grammar Mode** settings only determine how Designer treats the speech grammar in an application.

When you are done, click **Save** to save your changes.

Using Speech Grammars

You can use the User Input block to reference your speech grammars in applications.

In the **Input** tab, you can select **External Grammars** and click **Add Grammar** to add your speech grammar to the block. You can use multiple speech grammars at once.

In the **ASR Settings** tab, you can enable the **Use application-wide ASR settings** check box to use the default ASR (Automatic Speech Recognition) settings. You can define these settings by clicking **Settings** in the Toolbar.

Alternatively, you can disable the **Use application-wide ASR settings** check box to fine-tune the ASR settings for this **User Input** block.

Refer to the User Input block page for more information.

Bot Registry

Use the **Bot Registry** to add and define the bot resources you want to use in your applications.

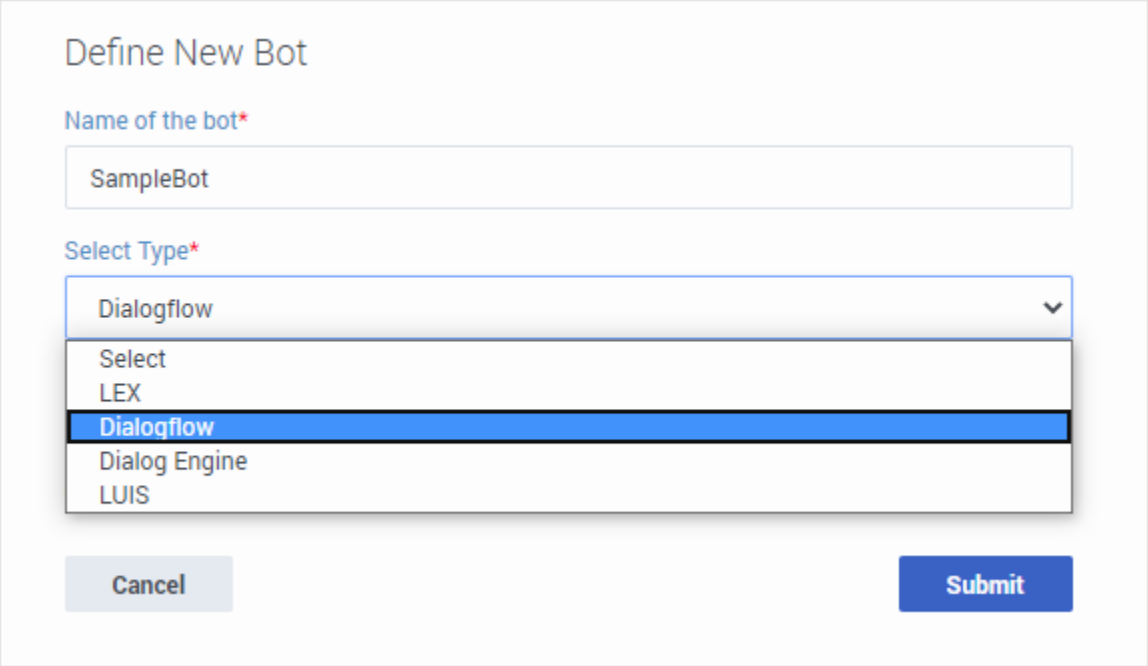
Designer supports several bot services providers, such as Google Dialogflow, Amazon Lex, and Genesys DialogFlow. If you have a bot resource set up with one of the supported providers, you can add it to the registry and use it in your applications. See the Bot block page to learn more about how to use bot resources in your applications.

Adding a new bot resource

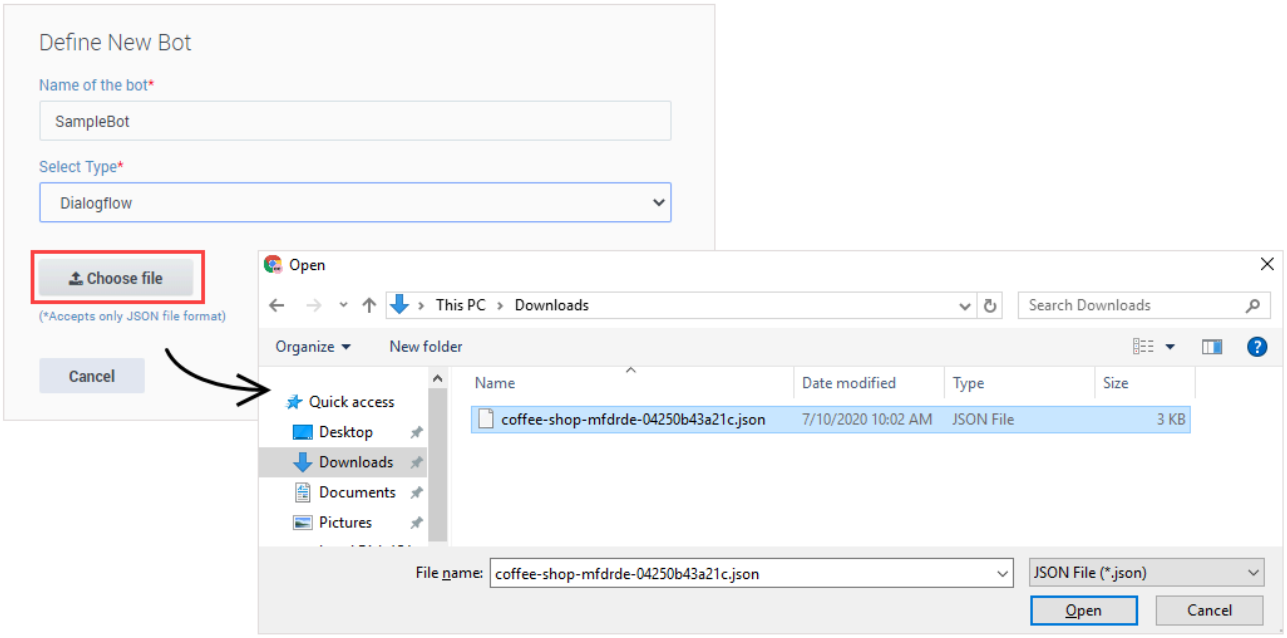
Watch: Add a bot resource to Designer

Link to video

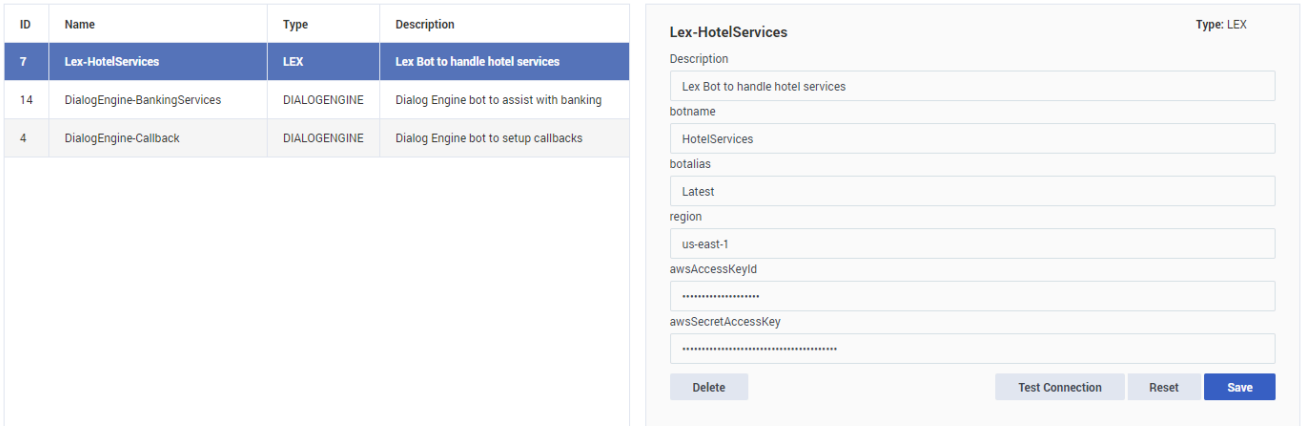
Click **Add Bot Definition**. Enter a **Name** for the bot resource and select the **Type** from the list of supported bot services providers.



For Dialogflow bots, you can import a JSON file that contains all of the required settings for your bot resource. (You can export this file from your Dialogflow account page — if you need help, see the documentation provided for your bot service.)



For other bot services providers, you can click **Submit** and enter the details manually:



Use the **Test Connection** button to check if Designer can communicate with your bot. When you are finished setting up the properties, click **Save** to add it to the registry.

After a bot resource is added to the registry, you can start using it in your applications by selecting the resource in a Bot block.

Business Controls

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

The **Business Controls** pages enable you to manage resources and settings that are specific to your site or business, such as Emergency Flags, Business Hours, Special Days, and Data Tables. Any changes made on these pages are immediately applied to your applications and modules—you do not have to update each application or module.

Learn more about:

Emergency Flags

Check for and react to an emergency condition.

Business Hours

Specify when your business is open or closed.

Special Days

Create holidays and other special days.

Data Tables

Create and manage data tables.

Callback Settings

Business Controls is also where you can manage the settings for Callback V2 (if applicable to your environment).

When you add or modify the Special Days or Business Hours for a callback virtual queue, Designer stores the values for each virtual queue in a system-defined data table called CALLBACK_SETTINGS. You can then manage the settings for each callback virtual queue by editing its related row in the table.

For more information about Callback, see the Callback V2 blocks page.

> **Tip**
> The CALLBACK_SETTINGS data table contains default values for parameters that are not set in **Business Controls** or by the Callback V2 blocks.

Emergency Flags

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

The Emergency Flags feature lets your applications check for and react to an emergency condition.

For example, if your business might be closed due to a storm, you can create a common Emergency Flag in Business Controls and have one or more of your applications check it. If the Emergency Flag is set, you can add special handling for that condition.

When the Emergency Flag is activated, the applications receive the new status when they check it, and start handling the emergency accordingly.

Adding an Emergency Flag

Click **Add Emergency Flag** to create an Emergency Flag. Enter a name in the provided field and click **OK**.

Activating or Deactivating an Emergency Flag

To activate or deactivate the Emergency Flag, click the On/Off slider.

Checking for Emergency Flags

To check the Emergency Flag in an application, add an Emergency block to the **Application Flow**.

In the properties of the **Emergency** block, enable the **Use Emergency Flags defined in Business Controls** check box and select the Emergency Flag that you previously created. The block will then show the current status of the Emergency Flag (this status display is for informational purposes only; you can't modify it from within the block).

The next time you activate this Emergency Flag from the **Business Controls** menu, the Emergency Flag handling is also activated in your application.

> **Important**
>
> Activating an emergency flag can impact existing interactions if the check was done prior to activation.

Business Hours

> **Important**
>
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

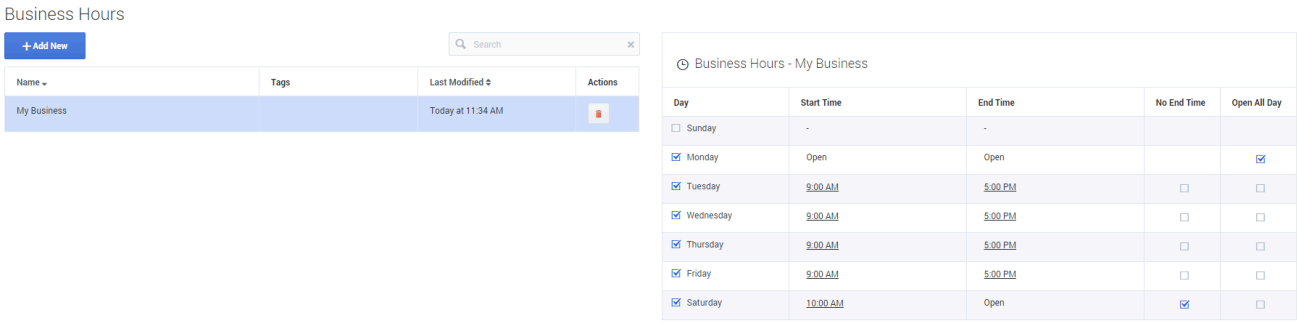
You can use the Business Hours feature to create various sets of business hours for use in your applications.

You must use a unique name for each set of business hours. You might want to use a combination of company names and departments. For example, you could use `CompanySales` and `CompanyService`.

It is also recommended to use tags to help organize your business hours. Once Business Hours are set, you can use them in your applications with a Business Hours block.

Specify open and closed times

Use the check boxes to indicate which days your business is open. Click the **Start Time** and **End Time** values to specify the opening and closing times, or use the **No End Time** or **Open All Day** options.



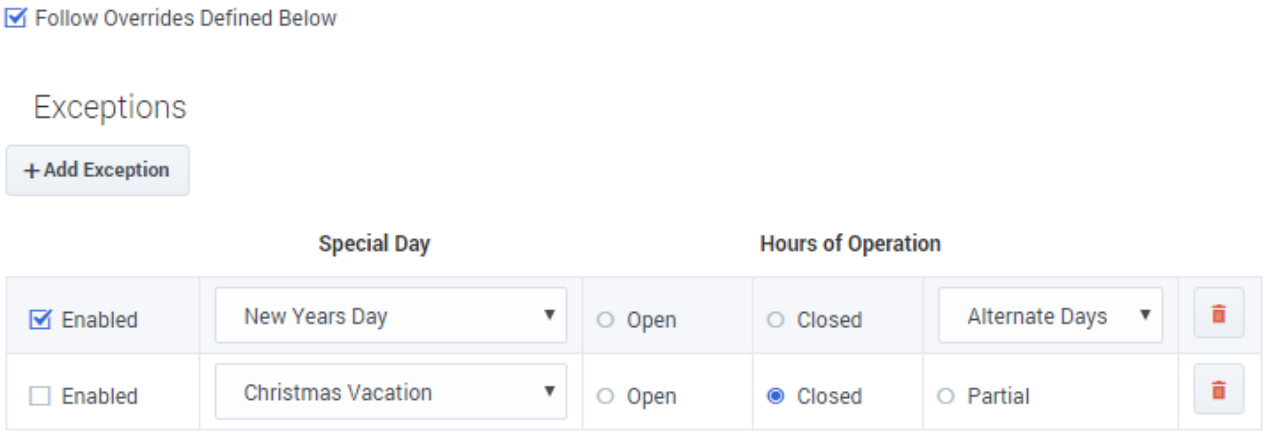
Important information for Callback

If you are using this business hours entry for Callback (i.e. in the Callback Settings data table), you must also specify the **Time Zone** setting. See Time Zone, below, for more information.

Specify alternate business hours

You can use these settings to override the normal Special Day business hours with alternate business hours.

To specify alternate business hours for a Special Day, select **Follow Overrides Defined Below** and click **Add Exception**. Choose a **Special Day** and select the **Hours of Operation**.



When you add an exception (or override), it is enabled by default. You can deselect an exception to disable it.

Important

- You must select a **Special Day** for each exception. Otherwise, the exceptions will not take effect.
- Even if a **Special Day** exception is enabled, it won't take effect unless the **Follow Overrides Defined Below** option is selected.

Time Zone (for Callback only)

If you are using Callback, select the appropriate **Time Zone** to use for booking scheduled callbacks. This setting ensures that customers who request callbacks are offered time slots that correspond to the time zone of the business hours that are being used for callbacks, and not the time zone of the Designer application (as set in the system variables).

The **Time Zone** setting only applies when determining the available time slots for booking scheduled callbacks. It does not affect any other callback features or settings, such as determining if the business is open or closed for offering callbacks or when to initiate **Immediate Blackout**. These

calculations are always based on the time zone that is specified in the **System Variables** of the Designer application.

Example

Let's say our application is running in the Eastern Daylight Time (EDT) zone (i.e. **America/New_York**), as specified in the **System Variables**:

System Variables

Name	Default Value	Description
Timezone	(GMT-5:00) America/New_York	TimeZone for this application used unless it is overridden in other blocks.

However, our contact center is located in California and is open from Monday to Friday, 9 am to 5 pm (Pacific Daylight Time). In this case, we would specify the business hours accordingly:

Business Hours - My Business

Day	Start Time	End Time	No End Time	Open All Day
Sunday	-	-		
Monday	9:00 AM	5:00 PM		
Tuesday	9:00 AM	5:00 PM		
Wednesday	9:00 AM	5:00 PM		
Thursday	9:00 AM	5:00 PM		
Friday	9:00 AM	5:00 PM		
Saturday	-	-		

Then, for the **Time Zone**, we would select **America/Los_Angeles** to use for booking scheduled callbacks. This setting will override the time zone that is currently set in the Designer application and be used for determining the available callback times:

Time Zone - Required only for Callback

(GMT-8:00) America/Los_Angeles

If a customer calls and requests a callback, they are offered an available time slot that is based on the **America/Los_Angeles** time zone. All other features and settings continue to use the **America/New_York** time zone that was set for the application in the **System Variables**.

Special Days

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

You can use **Special Days** to create holidays and other special days for use in your applications.

Add a Special Day

Click **Add Holiday** to add a special day. Use a name that describes the special day, such as `New Years` or `Thanksgiving Day`, and then use **Add Date Range** to create and specify a date range. You can create and define multiple date ranges for a Special Day, as well as enable or disable them.

As you make your changes, the **Special Day** is automatically saved. If the special day occurs on the same day every year, select the **Occurs every year** check box.

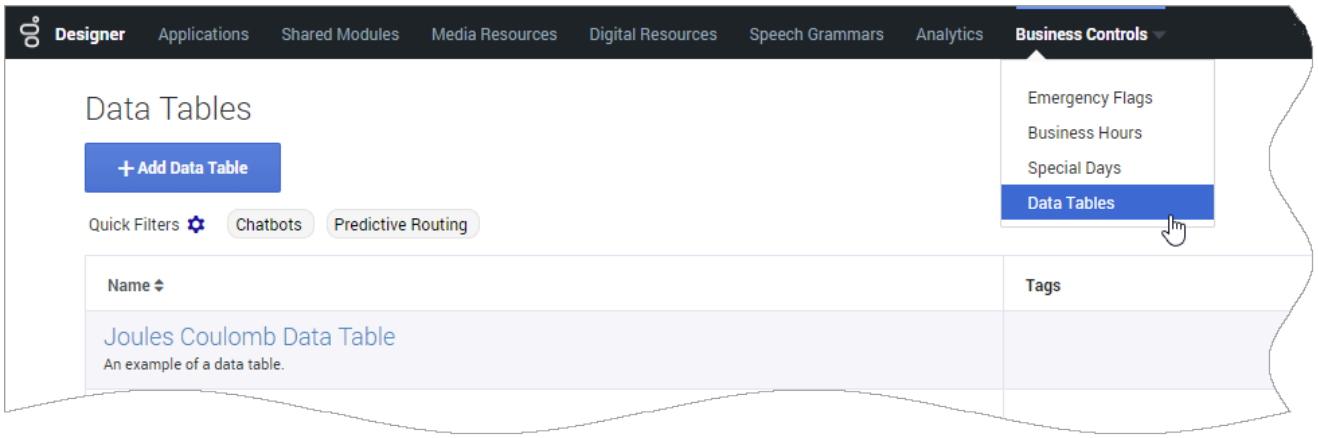
If you make an error, or want to make changes, you can edit or delete the Special Day by selecting it.

Once you have set Special Days, you can use them in your applications by using the Special Days block.

Data Tables

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

Go to **Business Controls** > **Data Tables** to view and manage your data tables.



What is a data table?

A data table contains values that can be read by a Designer application. It has rows and columns and looks similar to a spreadsheet (in fact, you can even export a data table and edit it in a program such as Microsoft Excel), but operates more like a database. Each data table has at least one primary *key* column, which Designer uses to lookup and retrieve (or store) a value from the table.

When should I use a data table?

Data tables are useful when you want an application to refer to values that are stored outside of the application, or if you want Designer to update values without actually changing them in the application.

For example, you might want customers from a specific region to receive a different welcome message than other customers. Or you might want to specify additional routing handling based on a particular condition being met, such as the business hours for that day or a customer's ID.

Also, certain Business Controls (such as Business Hours and Special Days) have corresponding data types, which means you can add them to a data table and then create and edit their values directly

from within the data table.

How do I add a data table to my application?

When you want Designer to reference a data table, just add a Data Table block to the application flow and select the data table you want from the list. To learn more about using this block, see the Data Table block page.

Creating a new data table

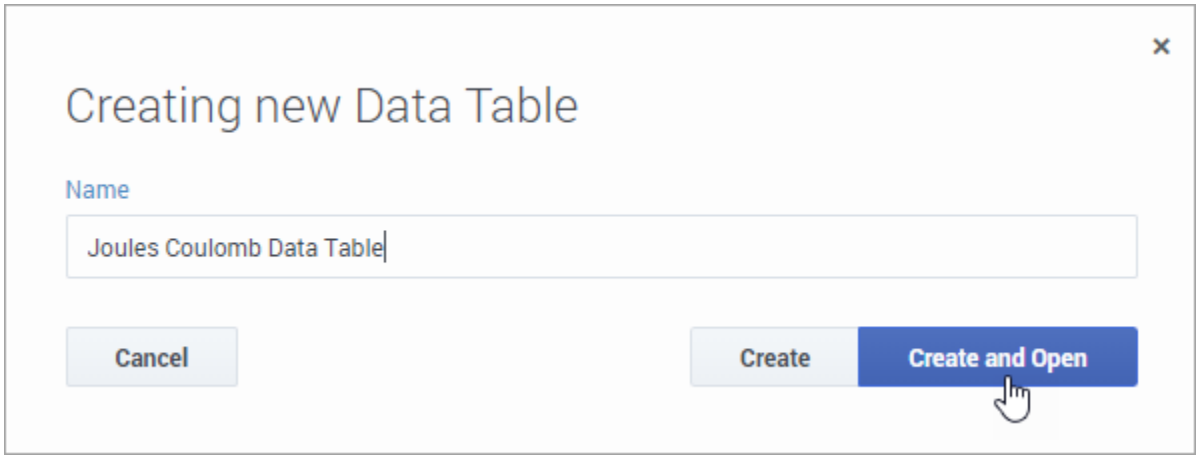
To create a new data table, click **Add Data Table** and enter a unique name for it.

You can click **Create** to save the new data table and return to the main **Data Tables** page, or **Create and Open** to save it and start configuring the data table properties.

Some things to keep in mind when planning or creating a data table:

- Limit the number of rows to 1000 and the total size of the data table to no more than 10,000 cells. If the number of rows is less than 1000, you can increase the number of columns until the 10,000 cell limit is reached. For example, a 200 row table can have up to 50 columns, and a 1000 row table can have a maximum of 10 columns.
- A data table is not intended to be used as a full-scale database—there is a limit as to how much data can be stored. Focus on data that is frequently updated or critical to your operations.

As an example, let's create a data table that tells the application to perform some special handling for a voice call based on a dialed number (DNIS). We'll call it **Joules Coulomb Data Table**:



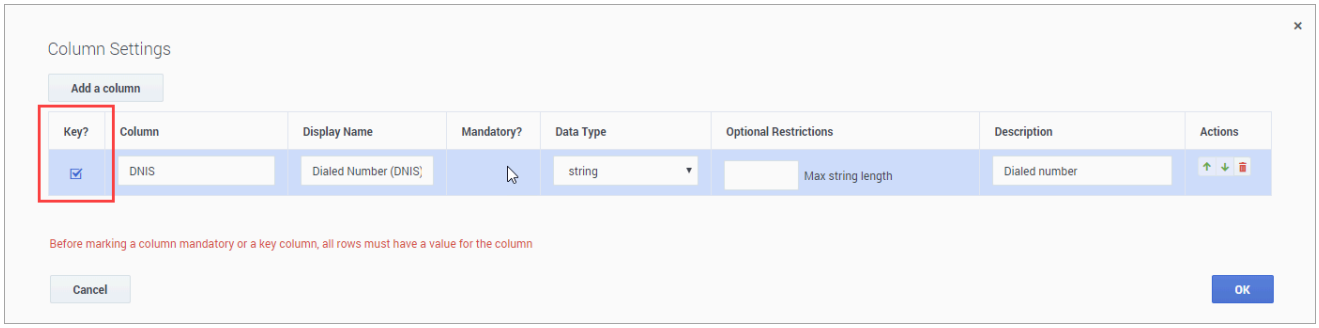
Click **Create and Open** to create the table and open it for editing. Our table does not have a schema (structure) yet, so Designer asks us to create one, which we'll do in the next step.

Defining the column settings

When a data table is first opened for editing, Designer asks you to define the structure, or "schema".

Click **Manage Schema** to define the **Column Settings**. (If the data table already has a schema defined, you can get to the column settings by clicking **Column Settings** when the data table is open for editing.)

In this example, we want to use the **DNIS** as the *lookup key*. The lookup key is the column that holds the value that Designer will search for when referencing the data table. We'll create a column called **Dialed Number** and select it as a **Key** column. We can then define the additional properties for this column, as follows:



Key?

Indicates if this column is a key column used to look up a row of values. In the example above, we want our column to be a key, so we've selected this option. **Note:** You can only select this option *after* you have entered values for the other column properties.

Clearly define the lookup keys, as these are important for searching for (and locating) the target data.

> **Tip**
>
> **Composite Keys**: You can select more than one **Key** column to create a composite lookup key. But note that if a key column is used for a lookup query and contains a blank or invalid value, Designer ignores it when returning the results. In effect, Designer treats it as if it is a wildcard, not as a "no match".

> **Important**
>
> You can't use a value of "0" (zero) in a **numeric** or **integer** key column. This causes a validation error.

Column

The name of a column to add to your data table. In the above example, we've entered `DNIS`.

Clearly categorize the data that you want to store. For example, if you are storing customer profiles, some various categories could be **Name**, **Address**, and **Phone**. Then you could set up **Name** and **Address** as a *string* data type and **Phone** as a *numeric* data type.

Display Name

Lets you customize how the column name is to be shown in the data table (this does not overwrite the actual **Column** value). In the above example, we've entered `Dialed Number (DNIS)`.

Data Type

Specifies the type of value(s) that will be used by this column. Supported data types include **string**, **numeric**, **boolean**, **announcement**, **integer**, **datetime**, **datetimerange**, **skillexpression**, **timezone**, **businesshours**, and **specialdays**. For this example, we've selected **string**.

> **Tip**
> When specifying integer values, the **numeric** data type does support integers, but data table lookups complete faster if you use the **integer** data type for these values.

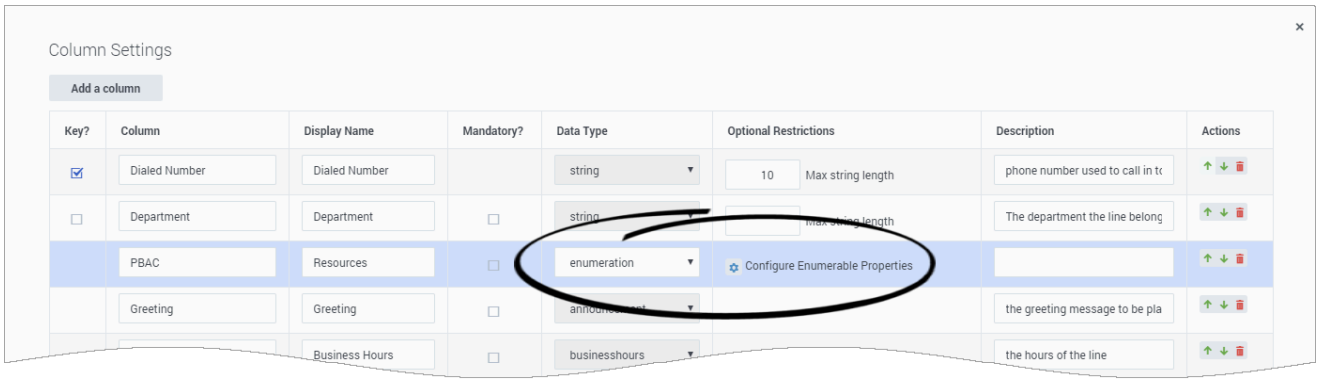
Description

An optional description of the column.

Optional Restrictions

For certain data types, you might want to specify any special restrictions. For example, you could limit a **string** value to a certain number of characters or make it mandatory to select a timezone when specifying business hours.

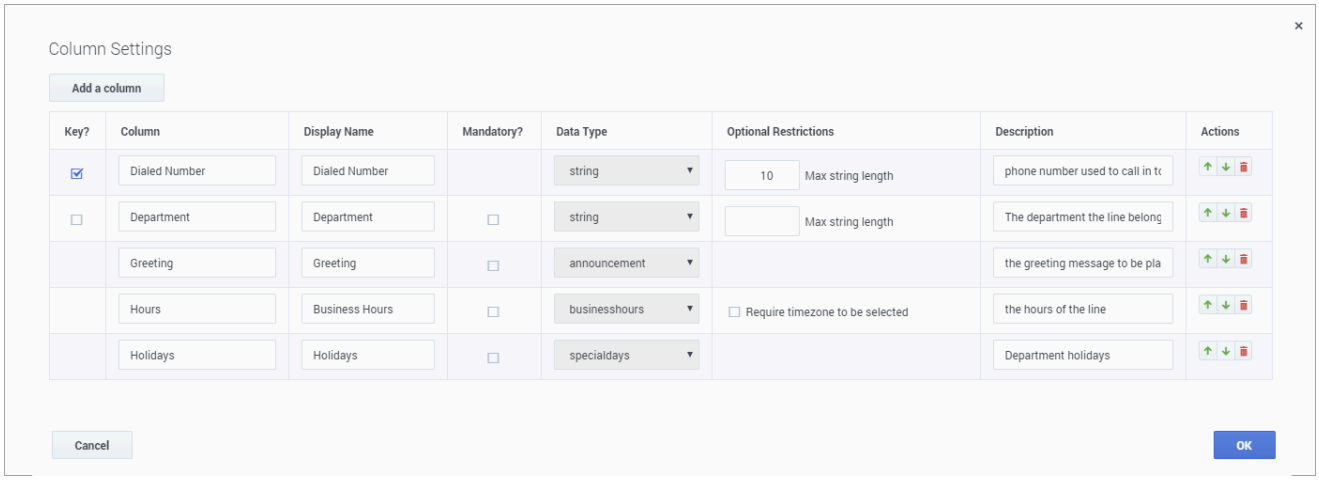
If you have Designer Administrator privileges and are setting up a column that uses an *enumeration* data type, you can also use Partition-Based Access Control (PBAC) to control the items that appear in the drop-down list for each Designer user. To do this, select the gear icon to **Configure Enumerable Properties** and use the **Enumeration Settings** window to manage the partition resources you want to display.



Finishing up

Add and define any additional columns that are needed. When done, click **Save**. You now have a data table with a key column of **Dialed Number**. You can go to Editing data tables for information about how to make changes to the data table, such as modifying its settings and adding rows and values.

Here is our example table with some additional columns added:



When you are ready to use the data table in an application, click **Publish Table** and use a Data Table block to add it to your application.

Editing a data table

To open a data table for viewing or editing, go to **Business Controls** > **Data Tables** and click the link for the data table you want to edit. (If another user has the data table open for editing, you will only be able to view it in **read-only** mode.)

> **Tip**
> Some browsers might display a script error or seem to freeze when opening large data tables. This is normal, and usually only temporary. You can let the script continue or wait for the browser to finish loading the data table.

From the editing mode, you can

- add or remove rows
- change column settings
- update values
- locate a specific value or row
- save and/or publish the data table

Don't forget! You can save your changes, but they won't take effect until the data table is published.

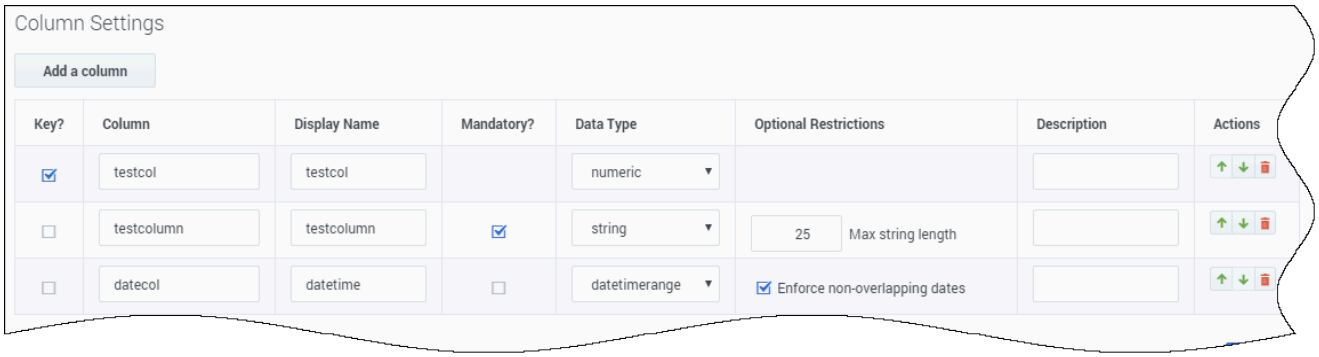
Adding or removing rows

To add a row, click **Add Row**. To remove a row, select it and click **Mark For Deletion**. Any rows that you mark for deletion are removed the next time you save the table.

> **Tip**
> For best system performance, limit the number of rows to 1000 and the total size of the data table to no more than 10,000 cells.

Changing column settings

Click **Column Settings** to add new columns or update the properties of existing columns.



For example, you can update the **Display Name** of a column, indicate if it is **Mandatory**, or specify any **Optional Restrictions** for that particular data type, such as a **maximum string length** for **string** types or whether to enable **Enforce non-overlapping dates** for **datetimerange** types.

Use the options under **Actions** to change a column's position in the grid or delete it.

> ### Important
> If you change the data type of a column, make sure that after saving the data table, you refresh or reload the page before entering or editing any cell values. Otherwise, the cell values under the modified columns might not display correctly. **After a data table is published, you cannot change the data types of the existing columns.** You can, however, still modify the schema of the data table and change the data types of columns that have not yet been published.

Changing data table values

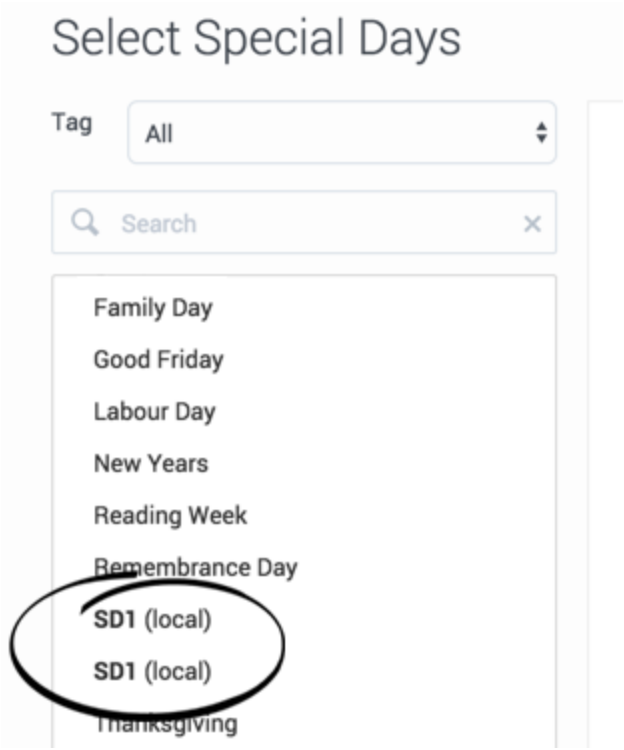
You can change the value of a table cell by clicking on it. As soon as you start editing a cell, the row is automatically selected and the updated text is displayed in blue:

Updating Business Hours and Special Days

You can change these directly in the data table. If you click on a Business Hours or Special Days value, you can select a different item or create and add a new one.

One thing to keep in mind — if you add new **Business Hours** or **Special Days** to a data table, the new business object is *local* to that data table. In other words, it can only be used by the data table it was created in. It won't appear in the global **Special Days** list and it won't be available to select in other data tables.

Local business objects appear in **bold** with a (local) label:



Searching the data table

There are a couple of ways you can quickly locate a specific value or row:

- Each column header has a search box. As soon as you start typing, Designer shows only those rows that contain a match to what you have entered.
- The **Row Count** box at the bottom of the page lets you jump directly to the specified row number.

Reviewing your changes

At the bottom of the data table, a tracker displays how many rows you have added, modified, or marked for deletion.

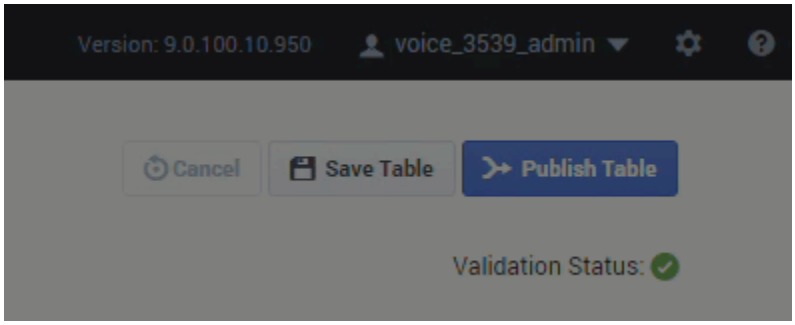
To view only the rows that were added, changed, or marked for deletion, check the **Show Modified Rows Only** box. Uncheck it to go back to editing mode.

You can also review changes to local business objects by clicking **Display Business Object Diffs** and selecting **Business Hours** or **Special Days**. You can then select an item from the list to view the original version side-by-side with the revised version. On the original version, edited properties are highlighted in red to indicate edits and deletions. On the revised version, edited properties are highlighted in green to indicate edits and additions.

Saving the data table

When you are ready to commit your changes, click **Save Table**. Make sure to review your changes! After you click **Save Table**, the changes can't be undone.

Designer validates your changes and lets you know if there are any errors.



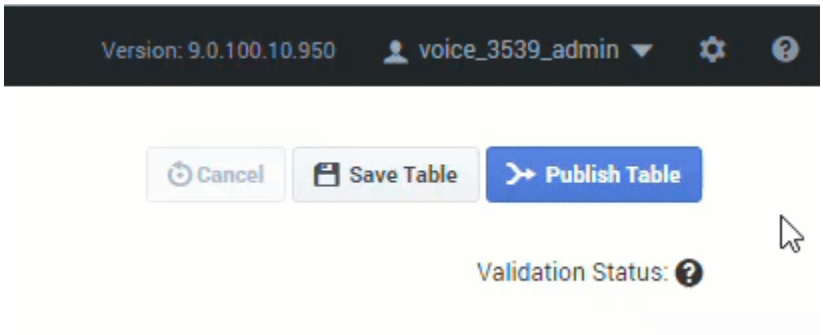
Tip

During the save operation, you might see some values (particularly for **Business Hours** or **Special Days**) suddenly change to **N/A**. This is just temporary, and the correct values will re-appear after the save completes.

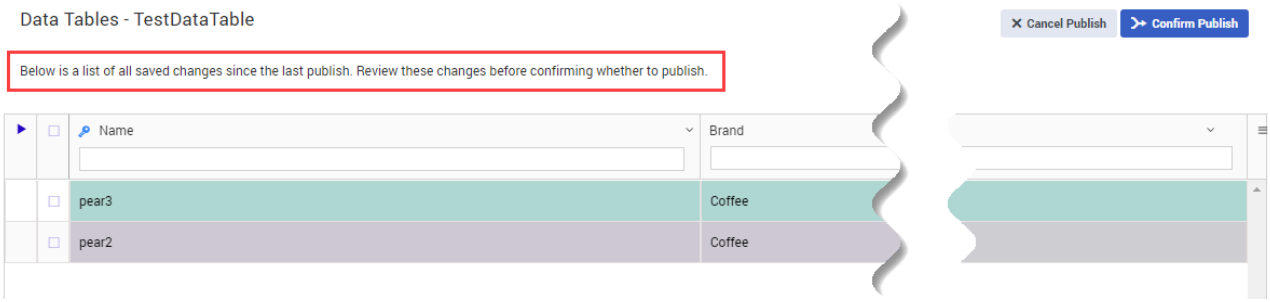
Don't forget! Saving a data table only preserves the changes you have made. To activate the changes, you must publish the data table.

Publishing the data table

Click **Publish Table** to publish (i.e. activate) the data table in the live production environment. Click **Confirm Publish** to start the publishing process.



If you are publishing changes to an existing data table, Designer also lists the differences between the previously published version of the data table and the one you are publishing now. You can review these changes before clicking **Confirm Publish**.



After you publish the data table, the applications that reference it have access to the latest changes.

> **Important**
> Designer may not always be able to display a complete list of detected changes on the confirmation page. However, you can still proceed with publishing the data table.

Viewing the history

Click **View History** in the **Actions** toolbar to open the history view (i.e. audit log) for a data table:

The history view shows you a list of each time the data table was viewed, edited, or published, the user who made the change, and the new and previous value of any properties that were changed.

You can use the buttons on the page to view the results for a specific time period (for example, **last 1W** to see the results for previous week), or use the date fields to specify a custom date range. Results can be sorted or searched and you can use the **Export** button to export the results as a CSV file.

You can drill-down further into each results item by double-clicking it. This opens an audit window opens that displays additional details for that particular event.

Exporting and importing data tables

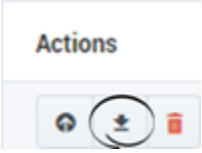
You might prefer to use another program, such as Microsoft Excel, to edit data table values. If so, you can export a data table from Designer into a CSV file that can be edited in Excel. When you are done, you can import the edited CSV file into Designer.

Warning

- DO NOT open an exported data table in Excel before reading the information contained in Editing exported data files in Excel. This will help you to avoid any unexpected data changes in your file.
- When importing the edited CSV file, do not change the column headers in Designer. The headers of the data table columns must remain unchanged between the export and the subsequent import. Otherwise, the import will fail.

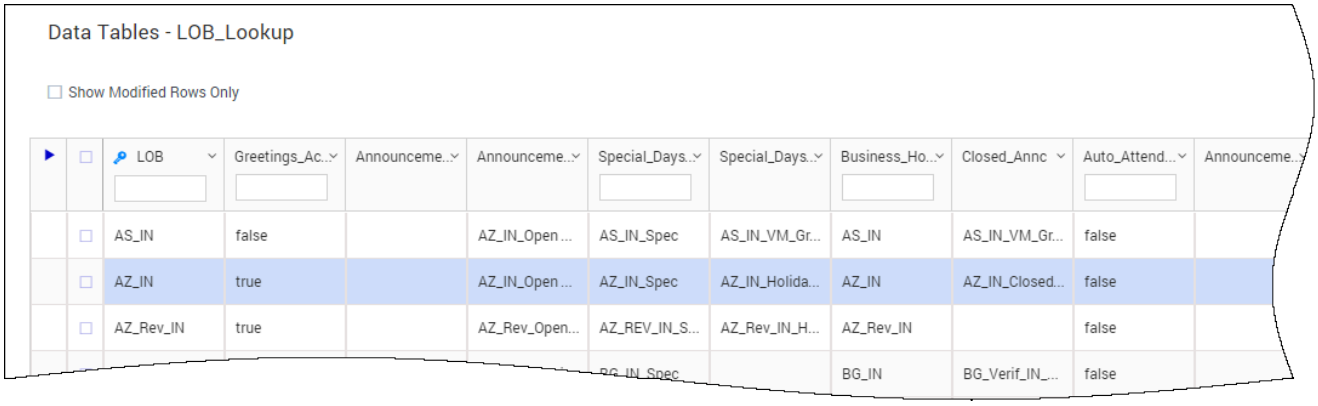
Export

Click **Export** in the **Actions** column to export a Data Table from Designer into a CSV file.



Below is a sample Data Table, its generated CSV file, and the CSV file in Microsoft Excel.

Data Table



CSV File

Here is how the row that is highlighted above would appear in the exported CSV file:

```
AZ_IN,true,,dbc63d70-37d6-11e6-a888-e53edc8cf09b,AZ_IN_Spec,2165b9f0-37d7-11e6-a888-e53edc8cf09b,AZ_IN,0508d4e0-37d7-11e6-a888-e53edc8cf09b,false,,,,VQ_AZ_IN,AZ_IN,30,AZ_IN,30,,,true,EstimatedWaitTime,78e070d0-37d7-11e6-a888-e53edc8cf09b,a38864a0-37d7-11e6-a888-e53edc8cf09b,,true,false,,Arizona_IN_transaction,e5ffb100-37d5-11e6-a888-e53edc8cf09b
```

Note that some of the items are represented by their resource ID and not their actual name. For example, the audio resource *AZ_IN_Open Greeting* appears as "dbc63d70-37d6-11e6-a888-e53edc8cf09b". This ensures that the correct resource is being referenced (names of resources can be changed, but their assigned resource IDs always remain the same).

> **Tip**
>
> Items in CSV files are separated (or delimited) by commas. If you need to use a comma within a value (such as for the text in a script dialog) you must enclose it in double-quotes (","").

Data Table in Microsoft Excel

Here is how the CSV file appears when viewed in a program like Microsoft Excel:

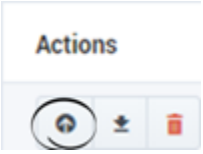
	A	B	C	D	E	F	G	H	I	J	K	L	M	N
1	LOB	Greetings	Announce	Announce	Special_D	Special_D	Business_	Closed_Ar	Auto_Atte	Announce	Priority_S	Priority_Ir	Target_Vi	Target_1
2	AS_IN	FALSE		dbc63d70-	AS_IN_Sp	bc6259f0-	AS_IN	bc6259f0-	FALSE				VQ_AS_IN	AS_IN
3	AZ_IN	TRUE		dbc63d70-	AZ_IN_Sp	2165b9f0-	AZ_IN	0508d4e0-	FALSE				VQ_AZ_IN	AZ_IN
4	AZ_Rev_I	TRUE		006bd7b0	AZ_REV_I	44e28ba0-	AZ_Rev_IN		FALSE				VQ_AZ_Re	AZ_REV_I
5	BG_IN	TRUE		2774bfe0-	BG_IN_Spec		BG_IN	90a14ca0-	FALSE				VQ_BG_IN	BG_IN

After you have edited the CSV file, you can import it into Designer.

> **Tip**
>
> While you can edit any item listed in the CSV file, it is more practical to edit items referenced by resource IDs from within the actual data table.

Import

To import a CSV file into a data table, locate it in the Data Tables list and click **Import** in the **Actions** column.



> **Important**

- Import is disabled for data tables that contain **menu** data types. If you do not see the **Import** icon in the **Actions** column for a data table, it indicates that the data table is using the **menu** data type.
- If you are importing a CSV file into an empty data table, Designer designates the first column as the key column.
- If you are importing a CSV file into a populated data table, make sure that the CSV file and the data table use the same table headers. If the headers do not match, Designer displays an error.

Editing exported data tables in Excel

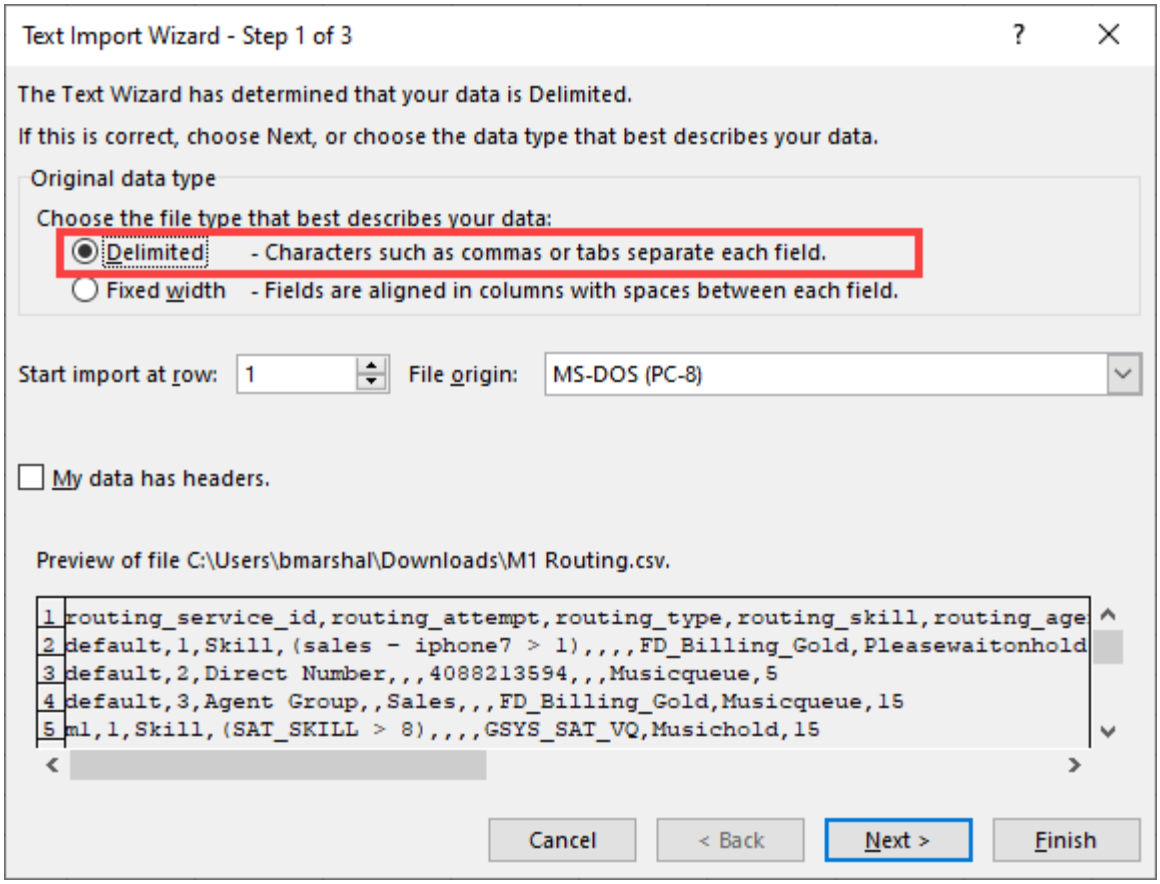
If you use Microsoft Excel to edit a CSV file, the program might re-format some of the data without any indication that these changes are being made. This can cause unexpected issues when the file is later imported back into Designer.

You can use the following guidelines to safely open a CSV file in Excel for editing. (The steps as described might differ slightly from your version of Excel. If you get stuck, you can use the help tool in Excel to find more information about how to perform that step in your version.)

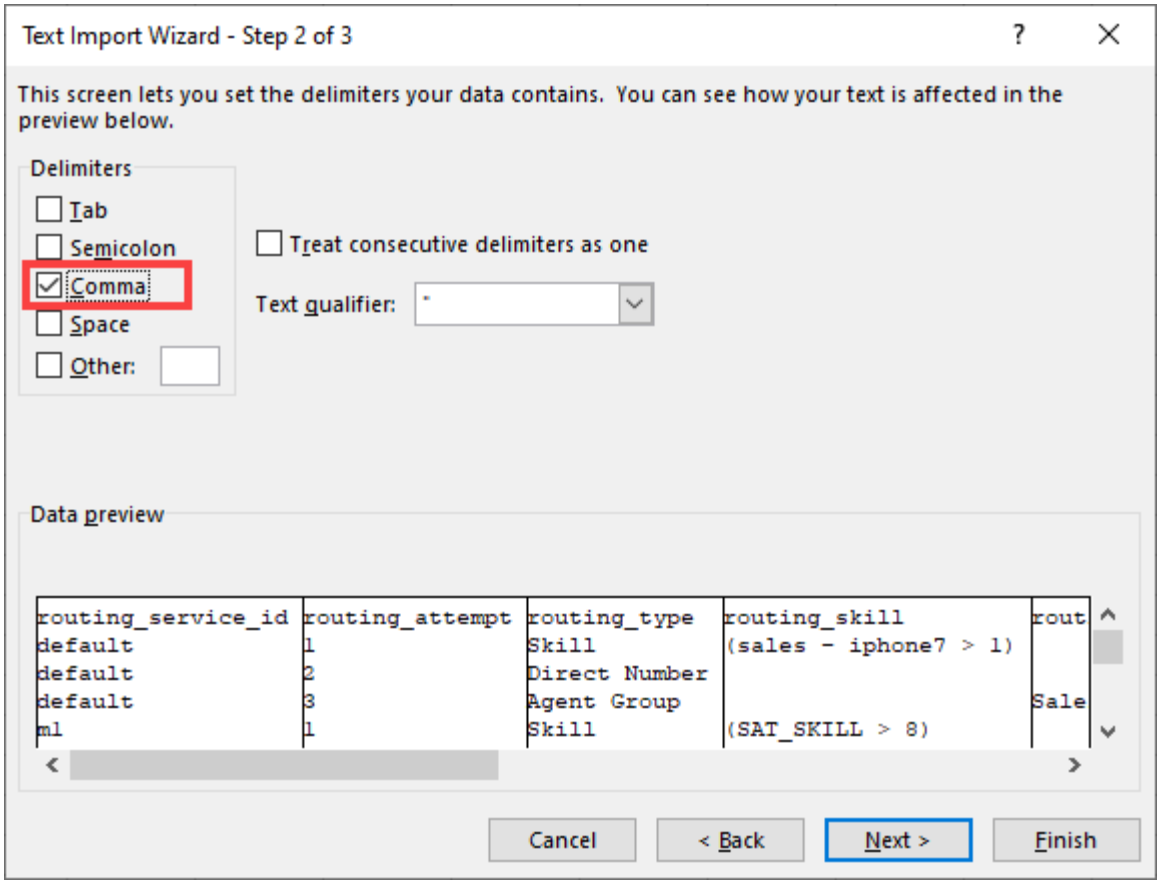
- In Excel, open a new blank workbook. Go to the **Data** tab and select **Get Data** > **From Text**.

> **Tip**
>
> If you only have an option for combined **Text/CSV**, you can enable the legacy wizard for importing a text file by going to **File** > **Options** > **Data** and enabling the **From Text (Legacy)** import wizard. Once enabled, you can then select **Get Data** > **Legacy Wizards** > **From Text (Legacy)** to open your file.

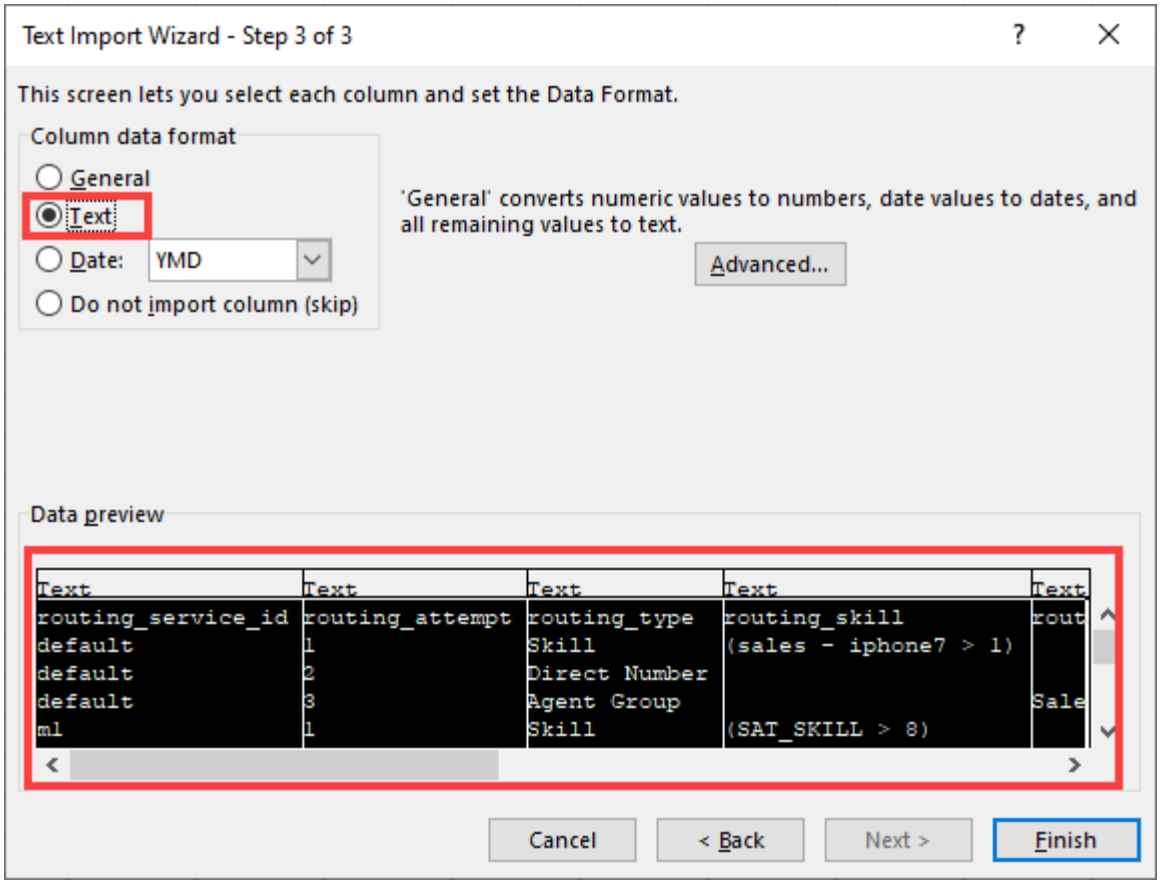
- Browse to the CSV file for the data table you want to open and select **Import**.
- In the **Import Wizard**, choose **Delimited** as the original data type and click **Next**.



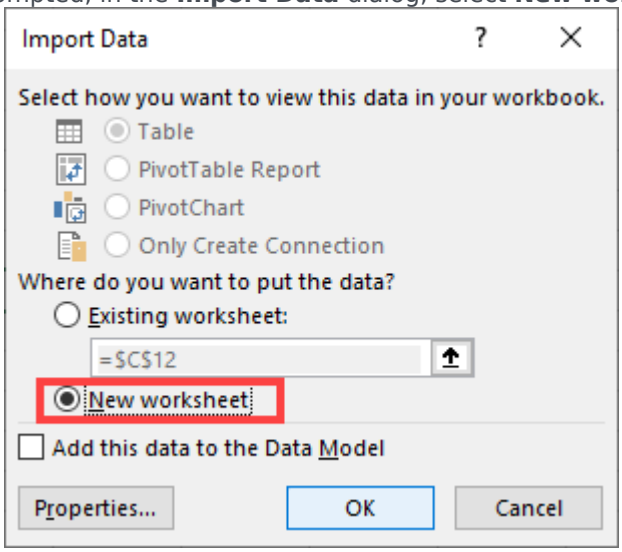
- Select **Comma** as the **Delimiter**. Make sure to deselect any other checkboxes that are selected. Click **Next**.



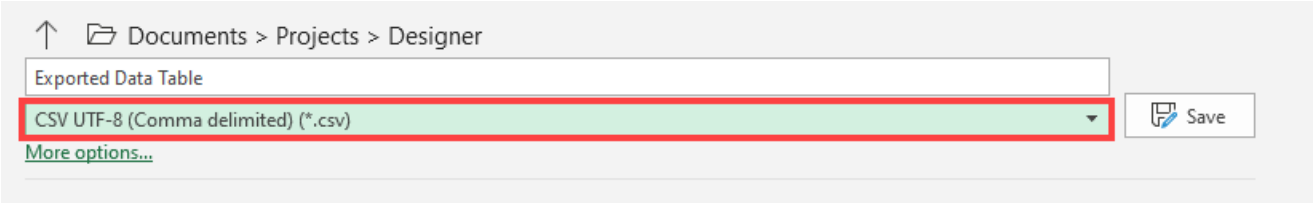
- In the **Data preview**, the first column should now be highlighted. Hold down the **Shift** key and click on the last column to select and highlight all columns. You might need to scroll to the right, depending on how many columns your data table has.
- With all columns selected, select **Text** as the **Column data format**. Click **Finish**.



- If prompted, in the **Import Data** dialog, select **New worksheet** for the data destination and click **OK**.



The file is now opened safely for editing. When you are ready to save the file and import it back into Designer, make sure to save it as a **CSV** (comma-delimited) type of file. This is the only type of file that Designer allows for data table imports.


Documents > Projects > Designer
Exported Data Table
CSV UTF-8 (Comma delimited) (*.csv)
Save
More options...

Admin

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

> **Important**
> The **Admin** settings are only available to users who are assigned to the **Administrator** role in Designer. For more information about roles in Designer, see Permissions and Access.

Click **Admin** in the navigation bar to access the Designer Administrator settings.

Partitions

Use partition-based access control (PBAC) to manage partitions, resources, and users.

Partition-Based Access Control

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

Use the settings on the **Partitions** page (under the Admin tab) to manage partitions, resources, and users. To view this page, you must be assigned to the **Designer Administrator** role in Designer. For more information about roles in Designer, see Permissions and Access.

As a Designer Administrator, you can control the resources that users have access to through **Partition-Based Access Control (PBAC)**. With PBAC, you can create a partition and assign certain resources to it. In Designer, "resources" are the various objects used during interaction sessions, such as Applications, Shared Modules, Business Hours, Special Days, Emergency Flags, Data Tables, Speech Grammars, and Media and Message Resources.

For each partition, you can then select the users who will belong to it. Users will only be able to see and manage those resources that are assigned to the partitions they belong to.

> **Tip**
> You can also use PBAC to control which resources are displayed to users in drop-down lists when setting up columns with *enumeration* data types in Data Tables.

All user permissions defined by their assigned roles within Designer remain in effect, and each user's PBAC details are stored in their Workspace settings and retrieved during login.

This video provides a quick overview of how PBAC works:

Link to video

Watch this video to see an example of how PBAC can be set up:

Link to video

> **Important**
> By default, PBAC works by **inclusion**. If a user is not assigned any partitions, it is assumed that PBAC is not in effect for that user and they will have access to ALL resources, including those that have partitions assigned to them. Similarly, if a

> resource is not assigned any partitions, it is considered a public resource that is accessible to ALL users.

In general, partitioning can be set up as follows:

- Define a private partition. Assign it to all resources that you intend to control using PBAC. You can leave out any resources that should remain globally visible.
- **Don’t assign this partition to any users.** This private partition will ensure that resources under partitioning control will NOT be visible to a user who has at least one partition defined.
- For each department, set up a dedicated partition and assign it to users from that department. Then assign each partition to the resources those users need access to. (Here's an example.)
- New resources inherit the partitions of the users who created them, and remain accessible only to users who belong to that partition.

For example...

You might create a partition for each of the following departments:

- Sales
- Finance
- Marketing

Then add users as members of their appropriate partitions:

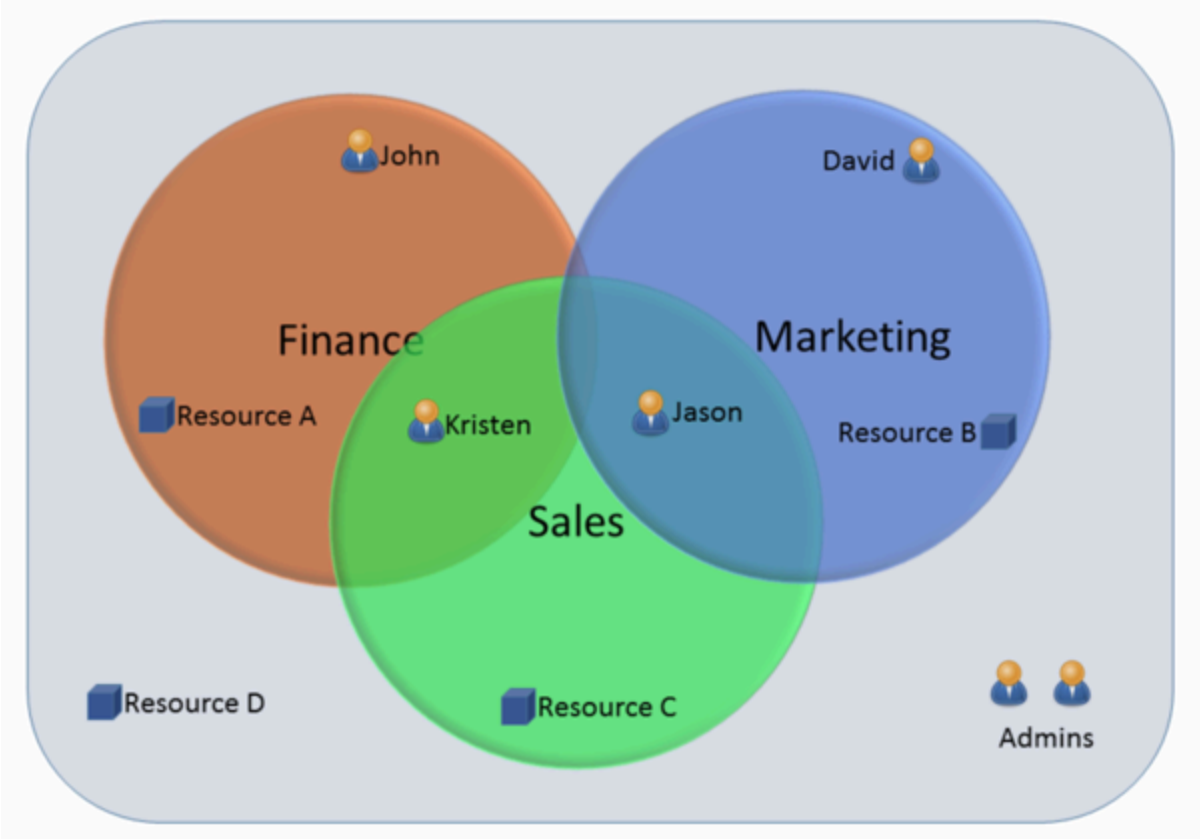
- John to Finance
- David to Marketing
- Kristen to Sales and Finance
- Jason to Sales and Marketing

Remember: Users who are Designer Administrators do not need to be assigned to a partition as they already have full access.

You can then assign certain resources to each partition:

- Resource A to Finance
- Resource B to Marketing
- Resource C to Sales
- Resource D to "none" (remember that non-assigned resources are visible to ALL users)

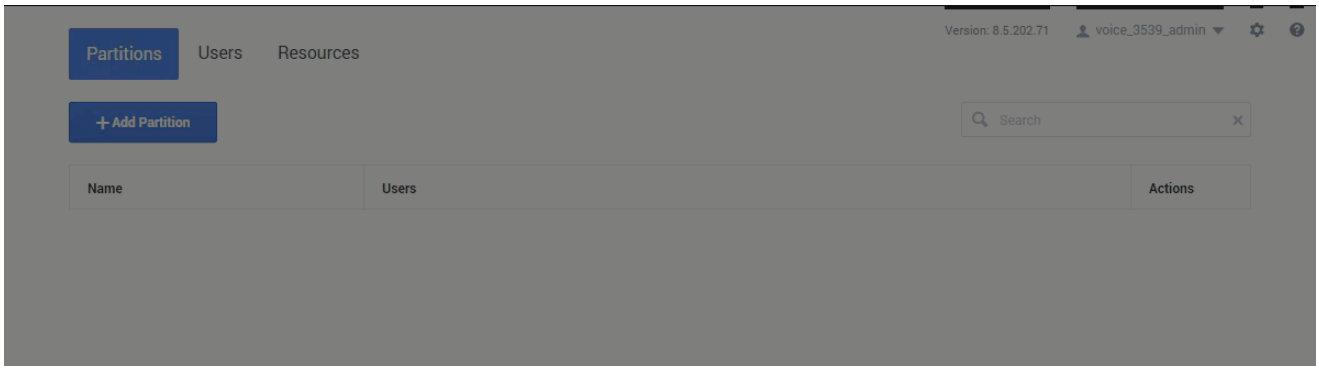
This diagram illustrates the relationships between the users, resources, and partitions described in this example:



Partitions tab

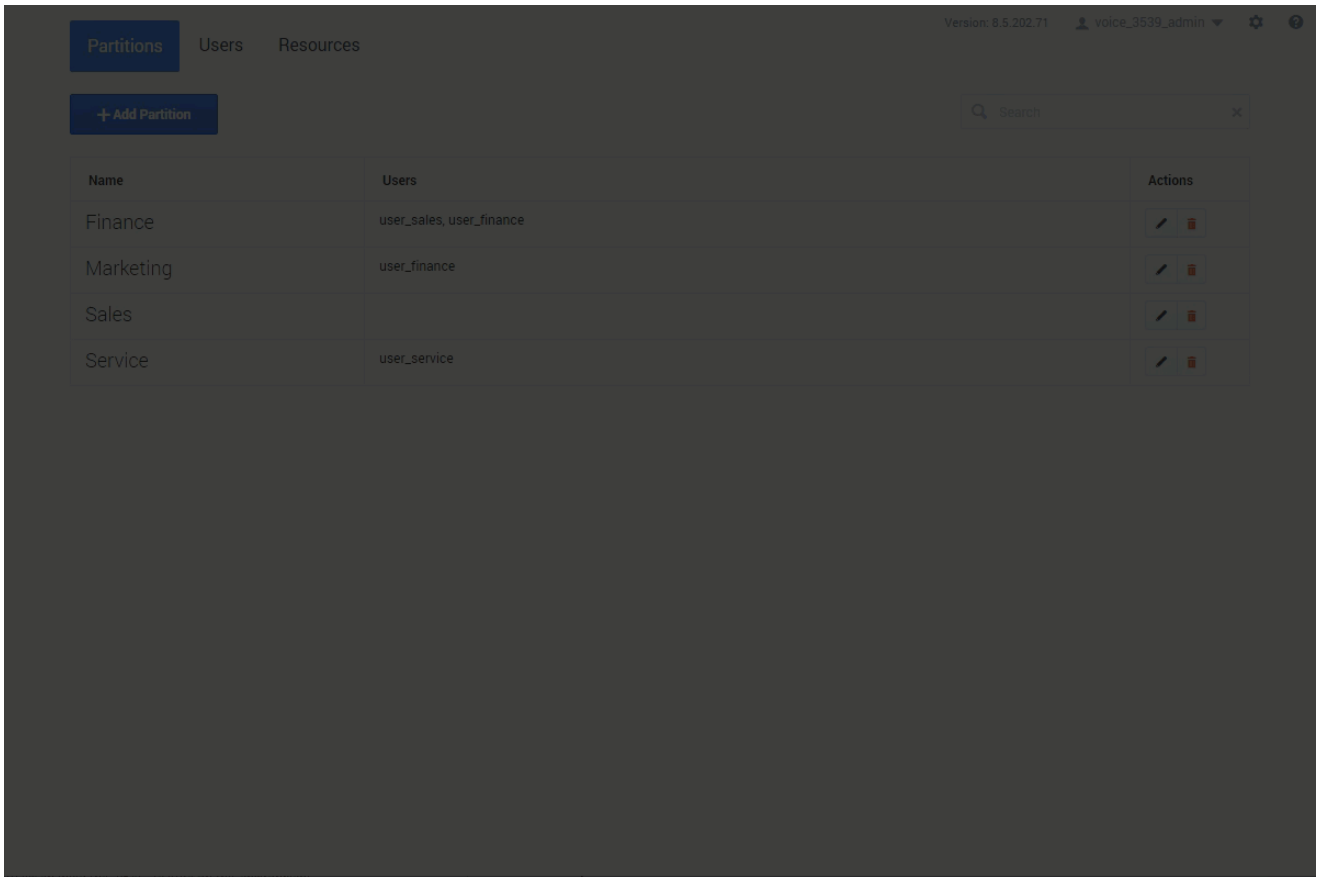
Use this tab to add or manage partitions and select the users who can access them.

For example, to add a new partition called **Sales**:



After the partition is added, you can use the **edit users** action to select the users who can access it:

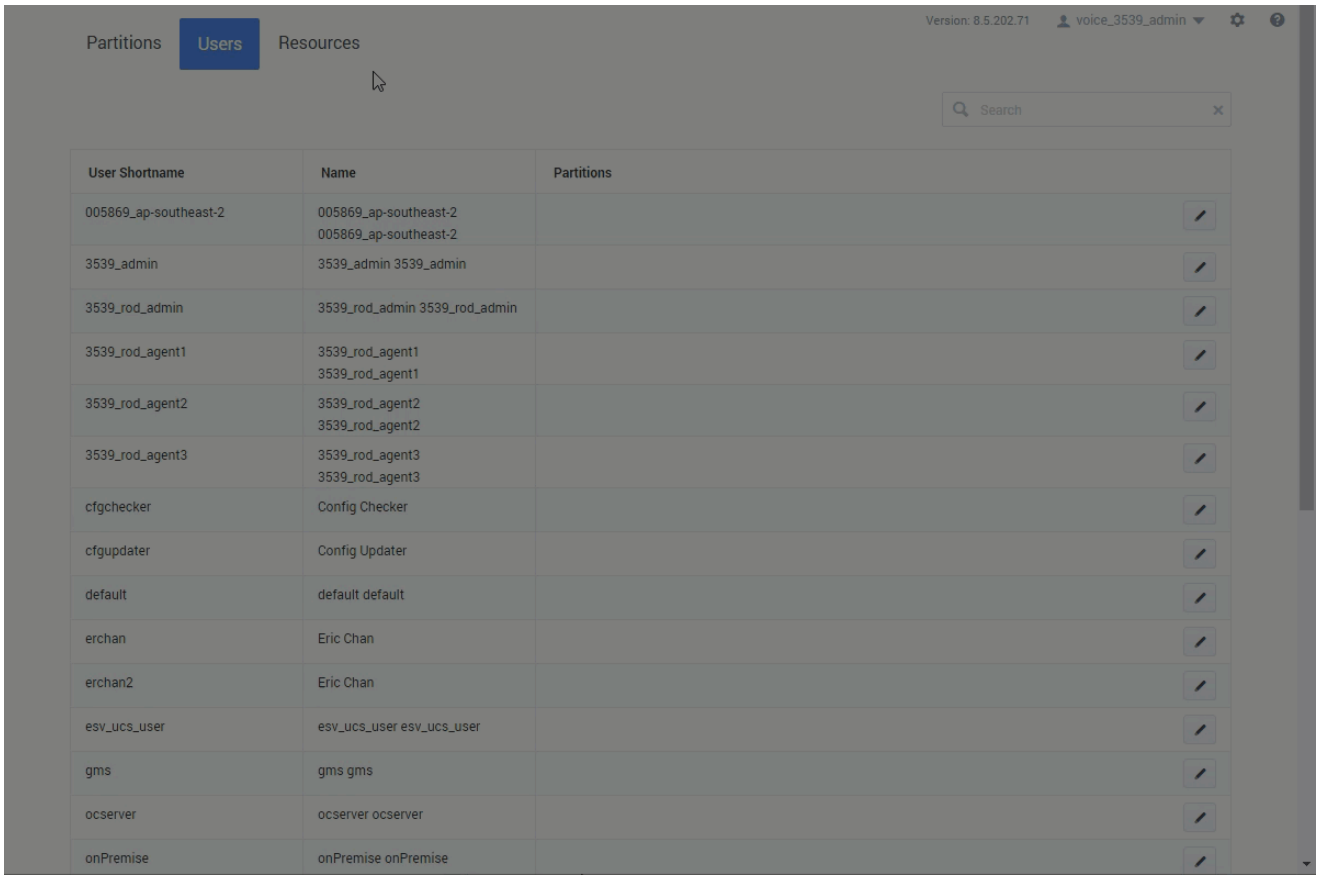
> **Tip**
> Users who are also Designer Administrators don't need to be assigned to partitions as they already have full access. Even if they are assigned to partitions, they will continue to see all resources.



Users tab

Use this tab to view the list of users and manage their assigned partitions.

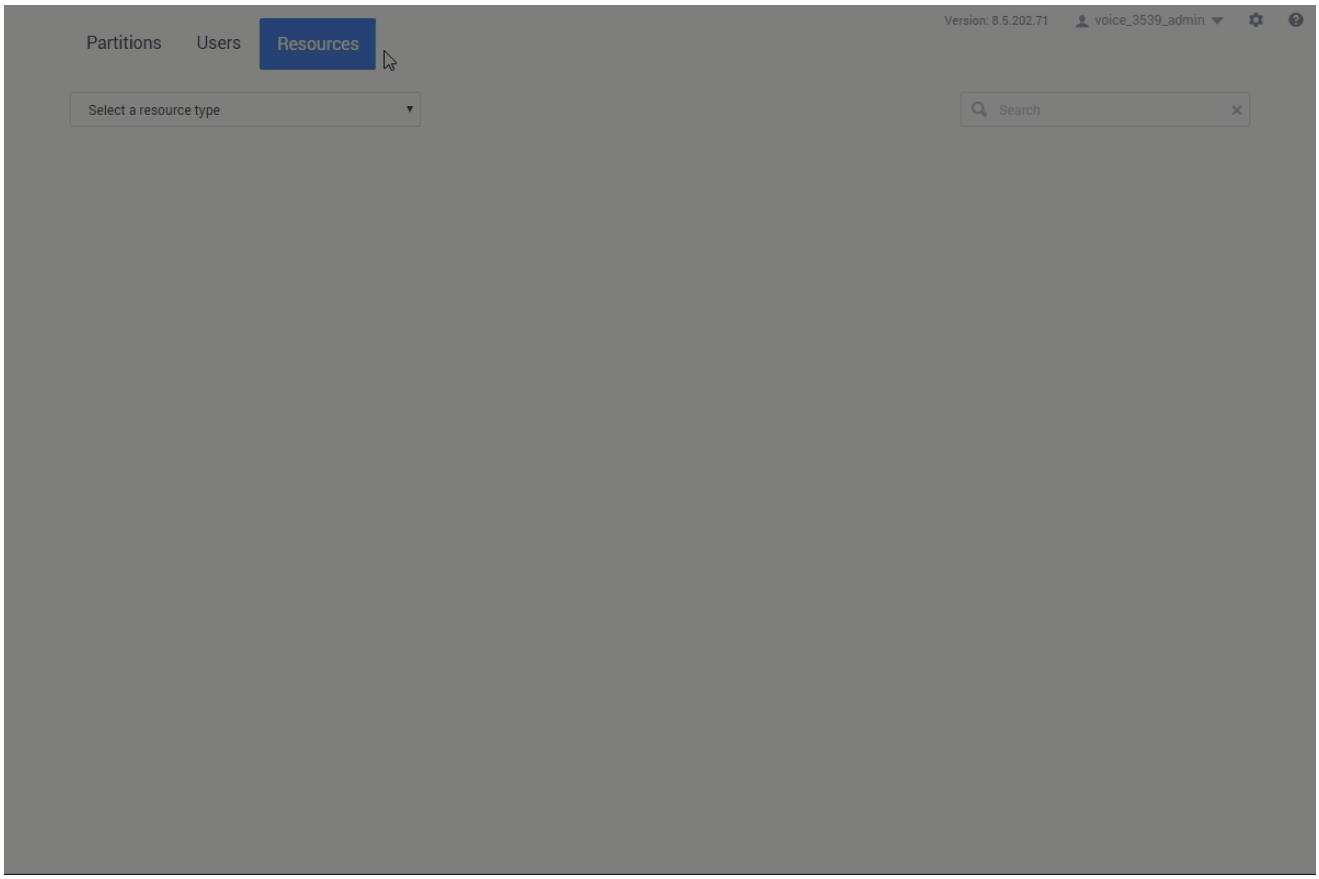
For example, to assign **user_sales** to the **Sales** partition and remove them from **Finance**:



Resources tab

Use this tab to view the list of resource types and their associated partitions.

For example, let's say the Business Hours resource **regularhours** is already associated with the **Service** and **Sales** partitions, but now we want to also associate it with **Marketing**:



Important

There are certain Designer resources that cannot be assigned to a partition because they are used by the system or are common resources that are shared across multiple applications. These include templates, shared audio resources, and some system-based data tables (such as CALLBACK_SETTINGS and NUMBER_VALIDATION_CONFIGURATIONS). All users have access to these resources.

Using Blocks

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

This chapter provides information about the blocks that are available in Designer.

> **Important**
> Some blocks might not be available if you are creating an **IVR** type application. Refer to the Applications page for a list of blocks that cannot be used in **IVR** type applications.

Application phases

An application flow consists of several sections of common blocks, known as application phases:

- Initialize
  This phase initializes application-level user variables and parameters to use when the application executes.
- Self Service
  This phase hosts blocks that provide automated interaction with the customer via speech, chat, and/or DTMF.

- Assisted Service
  This phase hosts blocks that route the interaction to a live agent, if necessary.
- Finalize
  This phase provides post-processing and interaction termination after the customer has been serviced.

Each phase is described in more detail below.

Initialize

The application initializes during this phase. By default, the following actions take place:

- Initialize and set up user variables.
- Load application run-time parameters from external sources.
- Process interaction properties (for example, ANI and DNIS) and application run-time parameters. System variables or properties may be initialized internally.
- If configured, additional processing that was set up by the user.

Self Service

The **Self Service** phase is the IVR portion of the interaction. This phase attempts to provide automated service and contain the interaction within an IVR, so there is no need to route the interaction to an agent.

If routing is necessary, this phase collects necessary data from the user through various questions and menus, and then determines how to route the interaction in the next phase, **Assisted Service**.

> ### Tip
> To enable voice call recording for the **Self Service** phase, set the **EnableSSRecording** variable to **true** in the System Variables section.

The following are typical actions that take place during the **Self Service** phase:

- Play Messages. These may be pre-recorded audio files or dynamic text spoken using TTS.
- Check business hours and customize logic based on the outcome (for example, take *this* action if we are closed).
- Collect user input.
- Present choices to customers using menus.
- Navigate customers appropriately, based on their responses (segmentation and branches).
- Call external RESTful APIs and fetch data into user variables.
- Update user variables and write ECMAScript expressions.
- Set up and process global commands and hot words.

The **Self Service** phase updates user variables with collected or calculated data. This data is later used by other blocks in the **Self Service** or **Assisted Service** phase.

Interaction processing might complete during the **Self Service** phase. In this scenario, the application control skips the **Assisted Service** phase and proceeds to the **Finalize** phase. For example, if the business hours check determines that the contact center is closed, the corresponding announcement is played to the caller and the call is terminated.

Assisted Service

> **Important**
>
> This phase does not appear if you are using an **IVR** type application. See the Applications page for more information on application types.

During the **Assisted Service** phase, the application attempts to route interactions to agents. Routing is performed based on data collected in previous phases. For example, target skills are taken from user variables.

The following are typical actions for this phase:

- Attempt to route the call while playing music or prompts.
- Call external RESTful APIs.
- Update user variables.

There may be multiple **Route Call** blocks in sequence. Each **Route Call** block might try to route the interaction to different targets with different timeouts. For example, it might expand a target by geographical location.

Each **Route Call** block has a timeout, after which the next **Route Call** block in sequence is executed. If any of the blocks successfully routes a call, the **Assisted Service** phase is complete and processing continues to the **Finalize** phase.

Finalize

When interaction processing is finished, the application goes to the **Finalize** phase to perform post-processing for various scenarios that are based on how the interaction was completed.

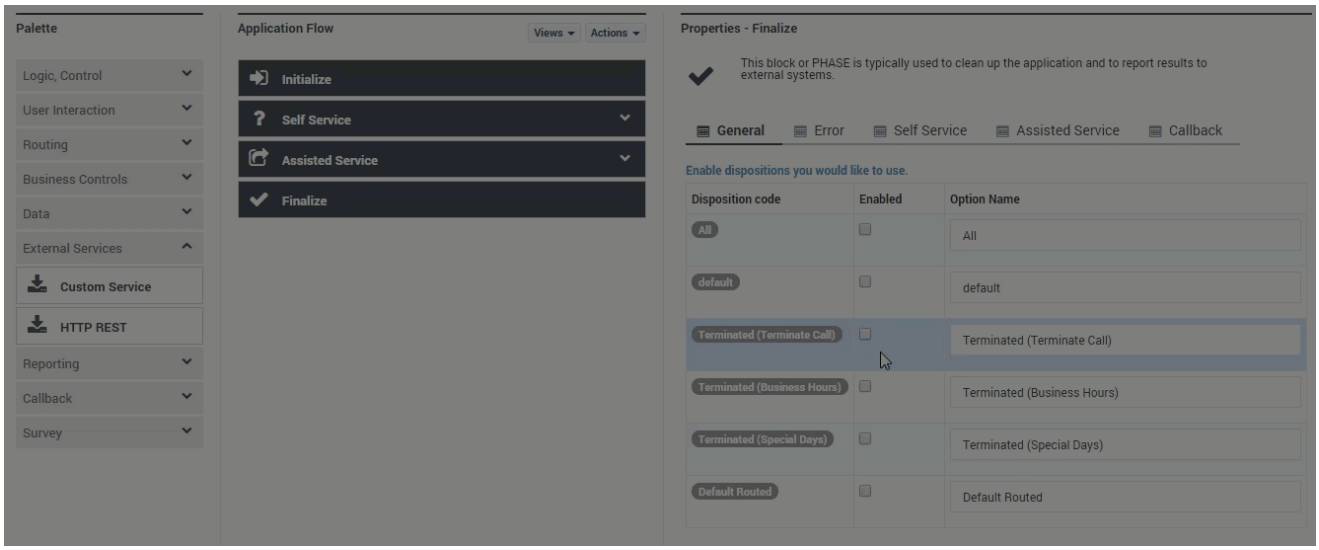
The following are examples of typical scenarios:

- Interaction was abandoned by the customer (while in either the **Self Service** or **Assisted Service** phase).
- Interaction was completed in **Self Service** phase.
- Interaction was routed to an agent in the **Assisted Service** phase.
- Interaction was delivered to voicemail in the **Assisted Service** phase.
- User opted to leave a queue and schedule a callback.

You can also use the **Finalize** phase to submit application data to an external system for reporting metrics, or to select a disposition code for post-processing.

When you click on the **Finalize** block in the application flow, each of the tabs (**General**, **Error**, **Self Service**, **Assisted Service**, and **Callback**) has a list of disposition codes that you can enable. When you select a disposition, a block for it is created below the main **Finalize** block. You can then drag other valid blocks (such as an HTTP REST block) below the disposition block to further customize the handling for that disposition.

Here's an example:



When an application enters the **Finalize** phase, it has only one disposition code, so only one disposition block is executed. However, the **All** disposition code is unique in that it is always executed, in addition to (and after) any disposition block related to the actual disposition code of the application. This is the only case where more than one disposition block is executed.

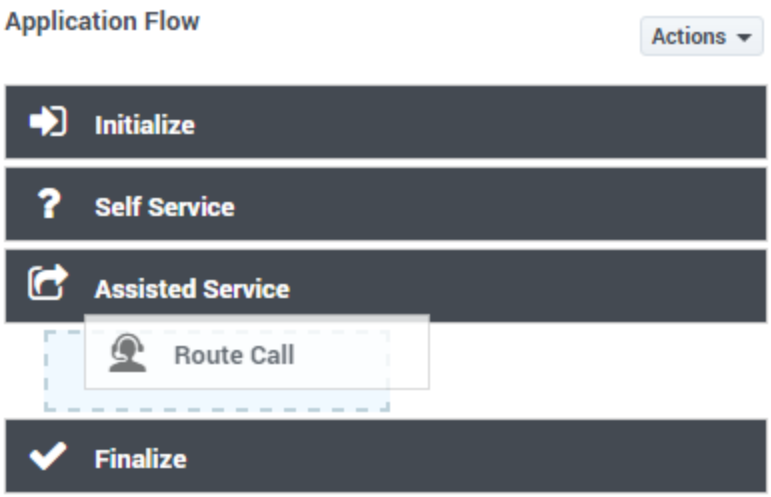
Typically, you would select the **All** disposition code when you want to execute some post-processing logic, no matter what the actual application disposition code is. This is more efficient than duplicating the same logic in every possible disposition block.

Setting up handlers of the **Finalize** phase is optional. There may be no need to do anything special for these cases.

Build Logic

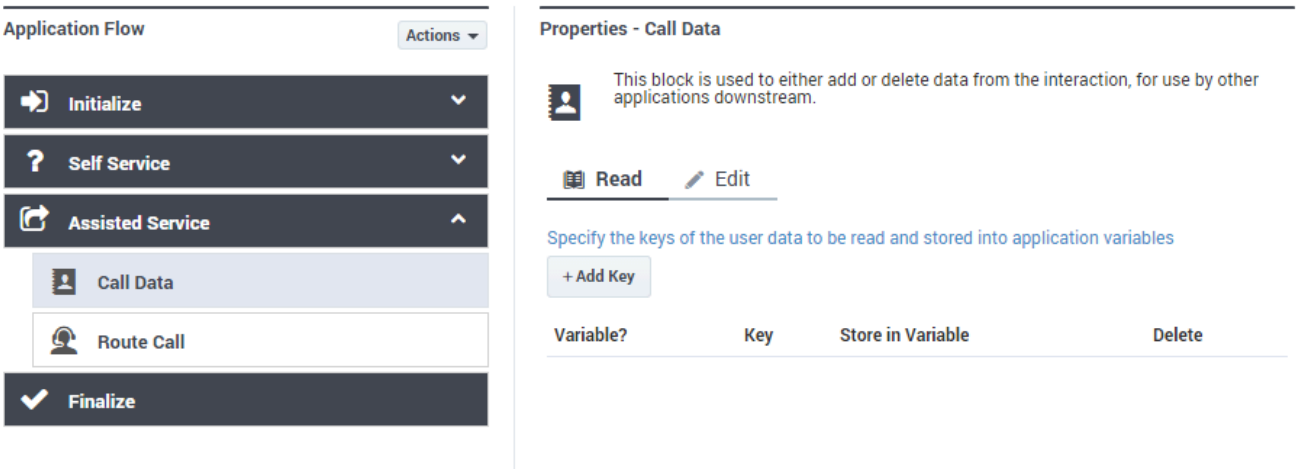
Using the Palette

The blocks are grouped into various categories on the **Palette**. You can drag any block from the **Palette** into the **Application Flow** and place it under the phase in which you want it to execute. If the block can be used in that phase, a blue placeholder block appears and you can drop the block to place it in that phase.

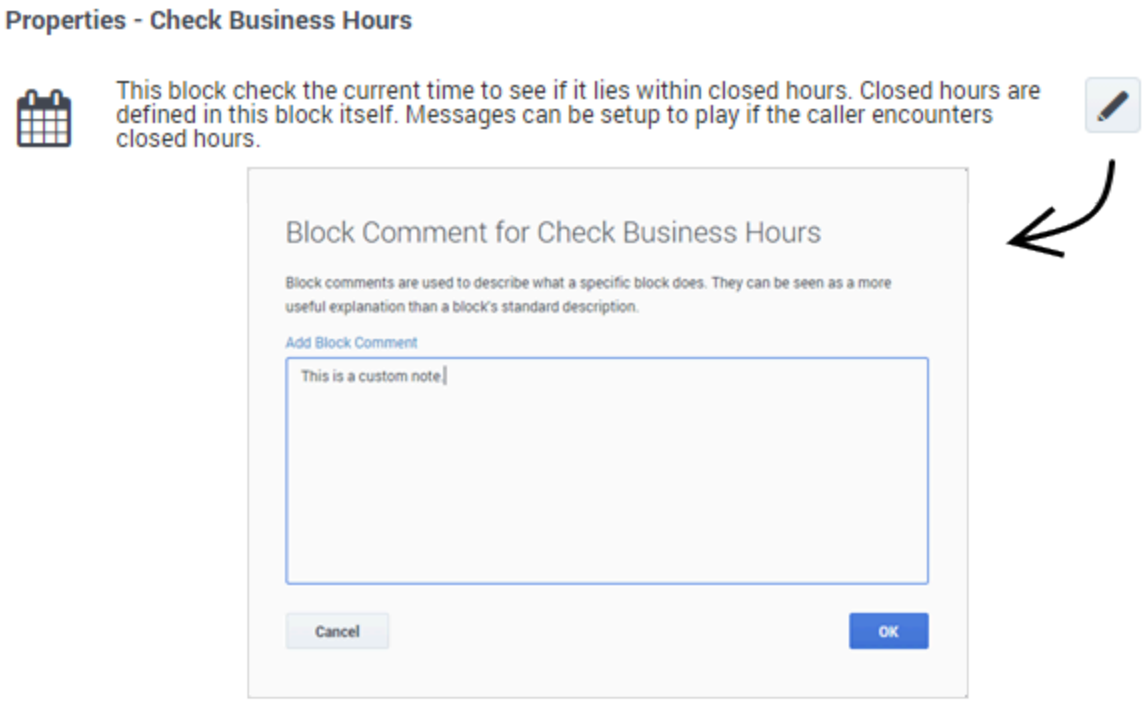


After placing a block, its details are shown in the **Property** view and you can configure the block and provide your application logic.

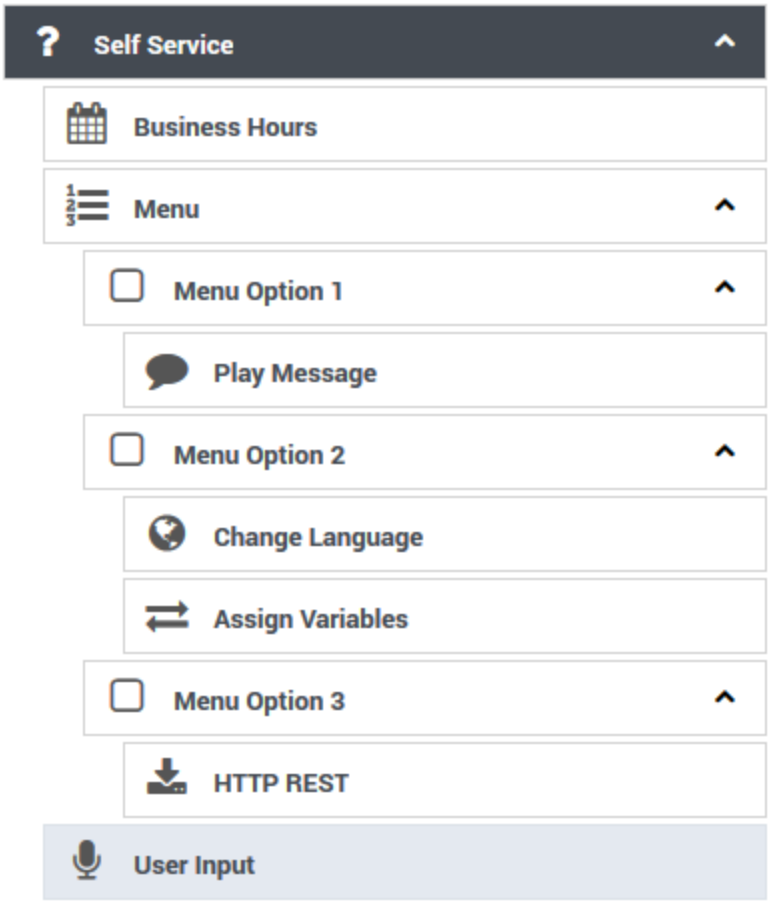
Click a block in the **Application Flow** at any time to select it, highlight it, and show its details.



Each block has a default description, which you can edit to add your own description or comment.



You can place child blocks underneath some blocks. Child blocks are indented underneath their parent block. With the **Menu** and **Segmentation** blocks, this indicates that several outcomes are possible but only one path is followed when the application runs.



In the above image:

1. A user hears a menu prompt.
2. The user enters input corresponding to one of the available menu options.
3. The child blocks of that menu option execute before the application continues with the next block after the **Menu** block.

In this case, if the user chooses **Menu Option 2**, the **Change Language** block runs, followed by **Assign Variables**, and then the menu completes and moves onto **User Input**. The child blocks under **Menu Option 1** and **Menu Option 3** do not execute in this scenario.

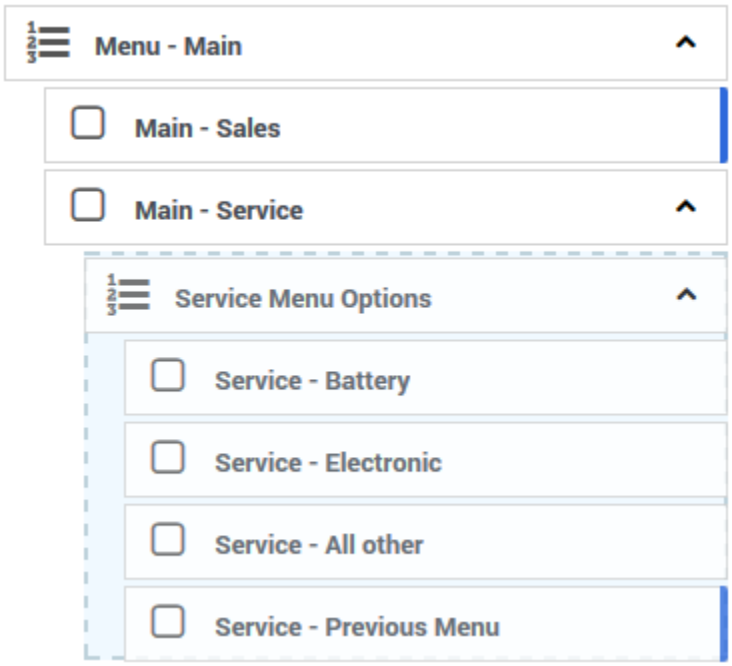
In general, an indentation on the **Application Flow** canvas indicates that a decision or branch occurs, and one of several mutually exclusive paths is followed.

It is also possible for **Menu** or **Segmentation** blocks to be nested within one another - that is, a top-level menu option can lead to a second menu. This is indicated with multiple levels of indentation on the **Application Flow**. Once an option or a branch completes, the application returns one level higher and continues execution.

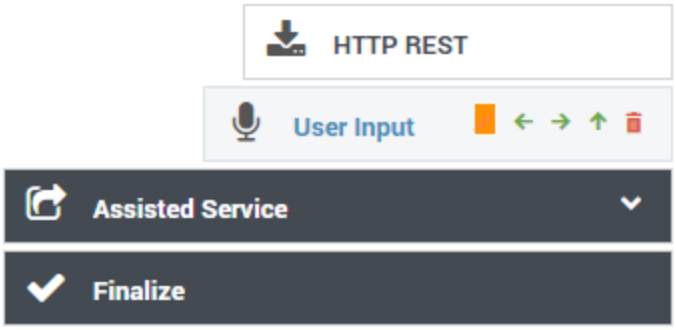
> **Tip**
> When you save your application, Designer also remembers if a group of nested blocks was expanded or collapsed. So the next time the application is opened for editing, the blocks appear in the same state as they were during the last save.

Moving and arranging blocks

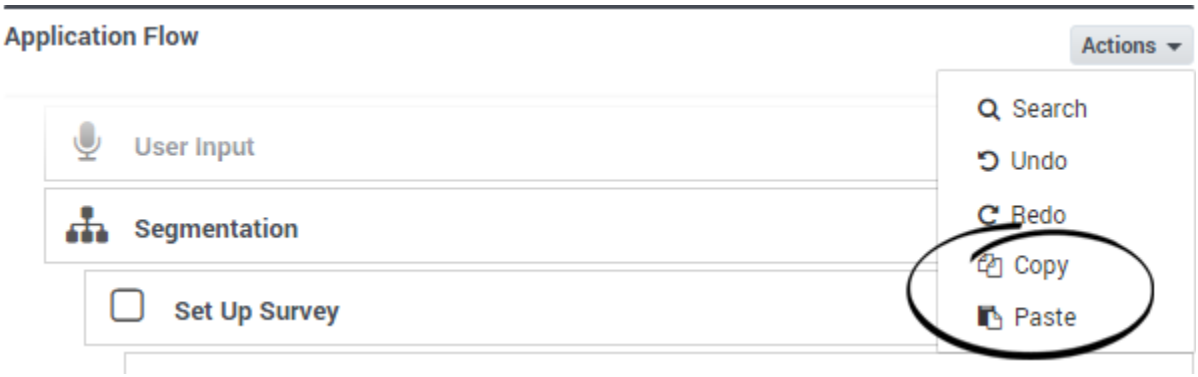
To rearrange the order of blocks, you can drag and drop blocks around the **Application Flow**. Moving a parent block also moves any child blocks under it, so you can move entire groups of blocks together in one operation.



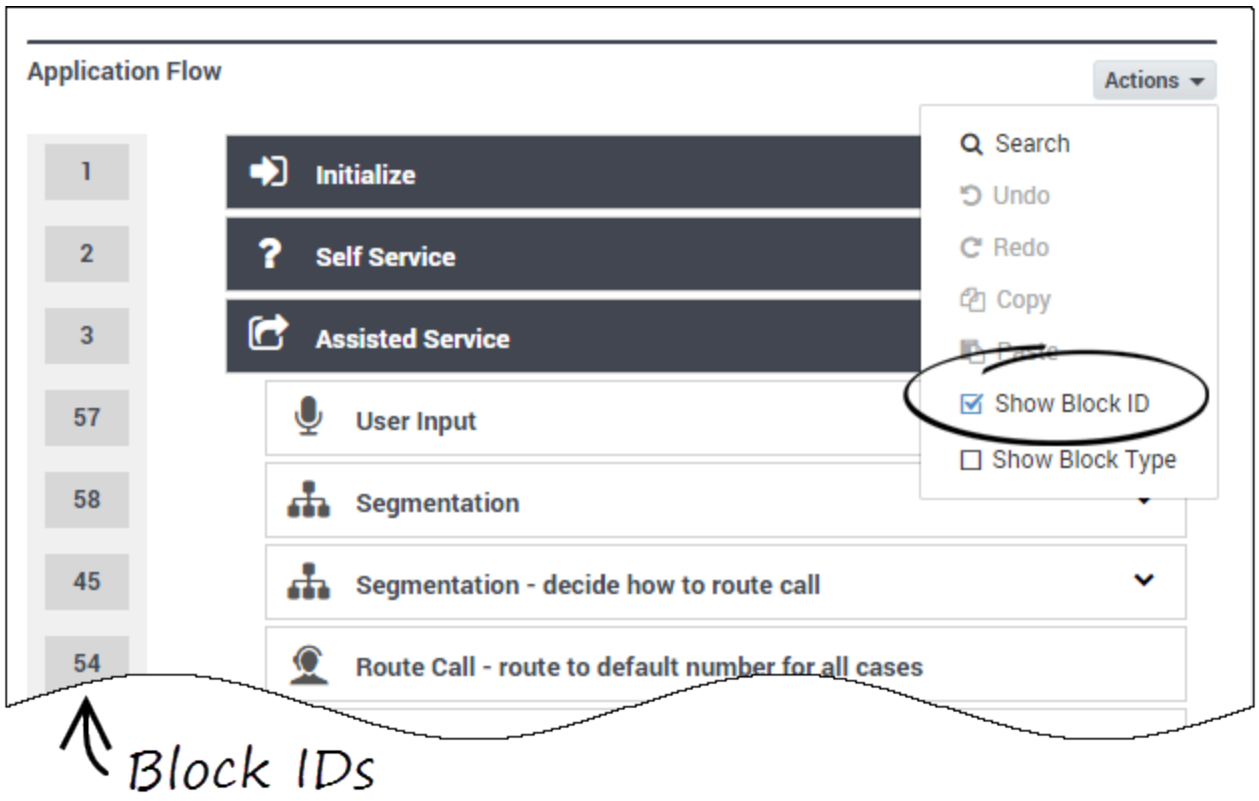
You can also move groups of blocks by clicking the arrows that appear when you hover over a block. This is useful if an application becomes large enough that dragging and dropping is unfeasible. These arrows show allowable operations on a block: up and down to move a block backward and forward within a phase, and left and right to change the indentation of a block underneath a parent block.



You can also use the **Copy** and **Paste** functions (under **Actions**) to copy a block to another location in the flow, or to another module or application. Copying a parent block also copies any child blocks under it, so you can copy entire groups of blocks together in one operation. Keep in mind that blocks can only be copied to locations where that type of block is permitted.



Under **Actions**, you can select **Show Block ID** or **Show Block Type** to toggle the visibility of those attributes. For example:

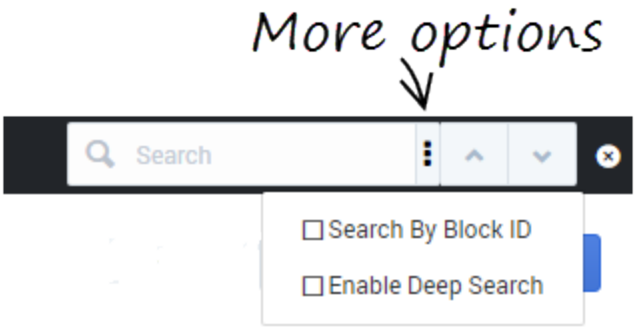


Searching the application flow

To search the blocks in the application flow, select **Search** from the **Actions** menu.

The search box appears in the main navigation bar and you can start typing the search term you are looking for. As you type, the items on the page are filtered to show only those items that match the text you've entered. The results are highlighted in the application flow, and you can use the up/down buttons to jump to the next or previous result.

If you click the **More Options** button, you can also select the **Search by Block ID** or **Enable Deep Search** options.



Search by Block ID can be useful if a particular block ID is flagged by Designer Analytics as having an issue. You can toggle the block IDs to be visible in the application flow and then use this search option to quickly locate the block in question.

The **Enable Deep Search** option enables you to search the entire application flow for a specific value or property. For example, you might search for blocks that contain a certain virtual queue.

For more about the search tools available in Designer, watch this video:

Link to video

Limiting Application Indentation

Although Designer allows you to use several levels of indentation, you might not be able to access the block properties element if you have more than 10 levels of indentation.

To prevent your application from becoming too deeply indented, use Menu and Segmentation blocks to jump to specific points in the application. This takes control back to the main *trunk* and prevents the application from being too indented and difficult to understand. **Menu** and **Segmentation** blocks that do not terminate the application within a reasonable depth should include a Go To block to jump to a different part of the application.

> **Important**
>
> In certain cases, you might need to skip over certain parts of the main application. In these cases, use a **Go To** block to forward processing to the correct block in the application.

ECMAScript Expressions

Some block properties accept ECMAScript expressions that are executed by the application at runtime. This allows the application to perform dynamic operations, such as calling an ECMAScript function or combining the values of other variables.

In general, block properties do not support ECMAScript expressions unless otherwise stated. If ECMAScript expressions are not supported, you must enter a value (a string that is taken as a literal). This value is not evaluated at runtime. For example, in the Play Message block, the value of a TTS prompt is taken as a literal string.

Supported Properties

The following lists blocks properties that support ECMAScript expressions:

- Initialize phase - User Variables
- Activity block - Values in key-value pairs
- Assign block - Assignments
- Data Table block - Lookup Key
- Menu Option block - Set Variables
- Milestone block - Values in key-value pairs
- Return block - Assigning values to output variables
- Shared Module block - Assigning values to input variables

Tips

- See ECMAScript Documentation for more information on using ECMAScript expressions.
- Strings must be quoted. For example, `'hello'` is a string, whereas `hello` is a reference to a variable called **hello**.
- Single quotes (`'`) are recommended, as opposed to double quotes (`"`).
- When specifying an object in JSON notation, surround the JSON with parentheses. For example: `({'abc': 'def'})`.

Examples

Below are examples of how you might use an ECMAScript expression in a Designer application.

Building a Dynamic TTS Prompt

You can use the Assign block to concatenate a string to be spoken by the application. The expression below reads the caller's phone number or ID.

```
'You are calling from ' + ANI
```

Control the Application Flow

A Segmentation block can take ECMAScript expressions that evaluate to a Boolean value, and thus control the flow of the application. For example, you might want to inform your customers about upcoming seasonal events and you need a way to determine the current season and whether the

event is occurring within the coming week. The expression below determines whether the call was received within seven days of the event, and whether the current season is summer or autumn.

```
numDays > 7 && (isSummer || isAutumn)
```

ECMAScript User Functions

Designer also has built-in ECMAScripts that you can invoke from a Designer application, such as from an Assign or Segmentation block, to perform certain functions at runtime.

isDataTableValueValid

You can use this function to determine if a value returned from a data table query is valid. For example, you might use the following function in an Assign block:

`isDataTableValueValid(`*`value`*`, `*`datatype`*`)`

This function has two arguments:

- *value* is a single value returned from a data table query
- *datatype* is the data type of the data table column, such as 'string', 'boolean', 'integer', 'announcement', or 'numeric' (this argument is optional)

If the data table value is valid, the script returns `true`. Here is a list of values that this function can return:

- `isDataTableValueValid(varStr, 'string')` on a valid (or empty) string returns `true`. Anything else returns `false`.
- `isDataTableValueValid(varNum, 'numeric')` on a valid number or `0` returns `true`. Anything else returns `false`.
- `isDataTableValueValid(varNum, 'integer')` on a valid integer or `0` returns `true`. Anything else returns `false`.
- `isDataTableValueValid(varBool, 'boolean')` on `true` or `false` returns `true`. Anything else returns `false`.
- `isDataTableValueValid(varAudio, 'announcement')` on a valid (or null) announcement returns `true`. Anything else returns `false`.

Busy Treatments

A busy treatment is a special form of voice call handling that tells Designer what to do while a caller is waiting for their call to be connected with an agent. For example, you can play music for callers or provide them with updates about their estimated wait times.

Certain blocks allow you to specify audio files or self-service type shared modules as busy treatments.

Route Call and Route Agent blocks

The Route Call and Route Agent blocks both have a **Treatments** tab where you can specify an audio file or a shared module as a busy treatment.

If you choose to add an audio-based treatment, a Play Message block is automatically nested below the routing block. Use this block to select and configure the audio options.

> ### Important
> If multiple consecutive **Play Message** blocks are added beneath a routing block as treatments, Designer considers them as one single treatment.

If you choose to add a module-based treatment, a Shared Module block is automatically nested below the routing block. Use this block to select the shared module that will be used as a busy treatment. Note that you can only select a Self Service shared module.

Busy treatments defined in routing blocks will loop automatically until a certain condition is met – such as the call is routed, the caller hangs up, or the timeout specified in the routing block expires – at which point the next block in the application is triggered.

> ### Important
> After a busy treatment has been executed at least 10 times, Designer exits the routing block and moves to the next block if the average duration of the treatment is less than 1000 ms (for example, due to a missing audio file).

Start Treatment block

The Start Treatment block also lets you specify a busy treatment, but it works a bit differently than the treatments used in the routing blocks.

Typically, you would use this block in the Assisted Service phase when you want to start a busy treatment — for example, play an audio file to callers while they wait to speak with an agent — and then move on to the routing blocks, all without interrupting the playback to the caller.

Things to keep in mind when using this option:

- **Don't define any additional treatments in the routing blocks that directly follow the Start Treatment block.**
  You want the audio started by the **Start Treatment** block to continue playing while the routing blocks do their job. If a routing block starts another treatment, the treatment that is playing stops.
- **The Start Treatment block does not loop a module automatically.**

If you want to set up looping, you might need to use a GoTo block in the module or find another way to loop it back. For example, you might add additional logic in the module to detect which mode it is being used in. This will expose an input parameter that controls whether the module loops internally or not. Or, you could clone the module and have a different looping logic defined in the cloned module. (However, take note that this option creates copies of the same logical module and might make maintenance more difficult.)

Validation

Designer enforces a drag-and-drop policy to ensure that you can only place blocks into applicable phases. In rare scenarios, a block might be placed in an invalid phase. In these cases, the validation process that occurs after you click **Publish** will report this failure with an error that includes the blocks placed into invalid phases.

You can update many options without regenerating the code:

- In the Business Hours block, you can:
  - change business hours of operation.
  - determine whether to terminate the call if it is outside business hours.
  - change the closed message (prompts).
- Update the Emergency block.
- Update prompts in the Menu block.
- Update prompts in the Play Message block.
- Specify the audio in the **Play Audio** tab of the Route Call block.
- Update the Special Day block.
- Update the input and retry prompts in the User Input block.

> ### Tip
> When nesting blocks, Genesys recommends that you do not go beyond ten (10) levels. Otherwise, you will receive a validation warning.

Block Categories

The blocks are grouped into the following categories:

- Logic and Control Blocks
- User Interaction Blocks

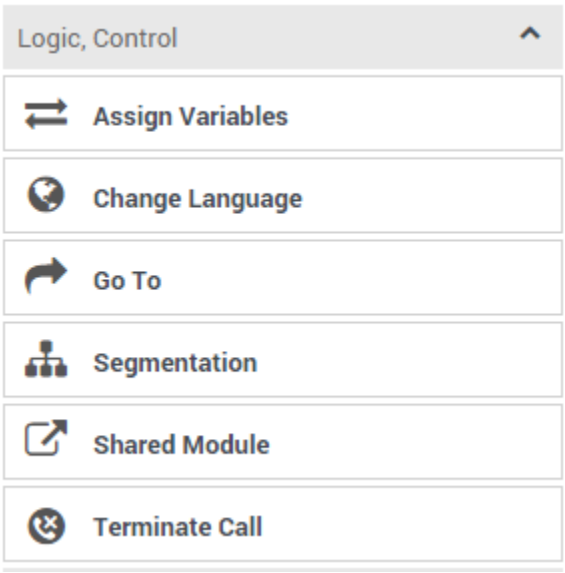
- Business Control Blocks
- Routing Blocks
- Data Blocks
- External Services Blocks
- Reporting Blocks
- Callback Blocks
- Survey Blocks

Logic and Control Blocks

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

The blocks in this category are used to add *logic* functions to an application, such as to assign variables, change the language (usually based on the customer's preference), and to provide *control* mechanisms within an application, such as to transition to another block, direct the application to follow a certain path, or end the interaction.

You might not see all of the blocks listed here on your Palette. The blocks shown depend on the features that are enabled and the type of application that is being built. For example, the **Terminate** block is only available for Digital application types.



Use the links below to learn more about each block.

Assign Variables

Assigns a new value or expression to user variables.

Used in: **Initialize**, **Self Service**, **Assisted Service**, **Finalize**

Change Language

Changes the language of the application and media resources.

Used in: **Initialize**, **Self Service**, **Assisted Service**, **Finalize**

Go To

Enables transitions to other blocks.

Used in: **Self-Service**, **Assisted Service**

Return

Returns control from the Shared Module to the application or Shared Module that called it.

Used in: **Shared Modules**

Segmentation

Selects a path based on a specific runtime condition.

Used in: **Initialize**, **Self Service**, **Assisted Service**, **Finalize**

Shared Module

Splits larger applications into smaller pieces.

Used in: **Self-Service**, **Assisted Service**

Terminate Call

Disconnects the caller and stops the call.

Used in: **Initialize**, **Self Service**, **Assisted Service**, **Finalize**

Terminate (Digital only)

Ends the chat session.

Used in: **Assisted Service**, **Finalize**

Assign Variables Block

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

You can use the **Assign Variables** block in any phase of the application to assign a new value or expression to any of the user variables. Those variables can be used in other blocks whose properties support variables (for example, TTS prompts). The last-known state of variables is captured in metrics just before the SCXML session ends.

> **Tip**
> You specify user variables in the **User Variables** tab of the **Initialize** phase. When the application starts, those user variables are declared and assigned the user-specified default value.

You can use the **Sort Function** tab to sort the elements of a JSON array in a specified order. A maximum of three keys can be specified with each array. The same array can be sorted multiple times; therefore, the number of sort keys is unlimited.

> **Important**
> The **Assignments** tab is processed before the **Sort Function** tab when your application executes the **Assign Variables** block. Do not assume that assignments are processed after sorting within the block. To extract specific parts of data after sorting, add another **Assign Variables** block after the one that performs sorting.

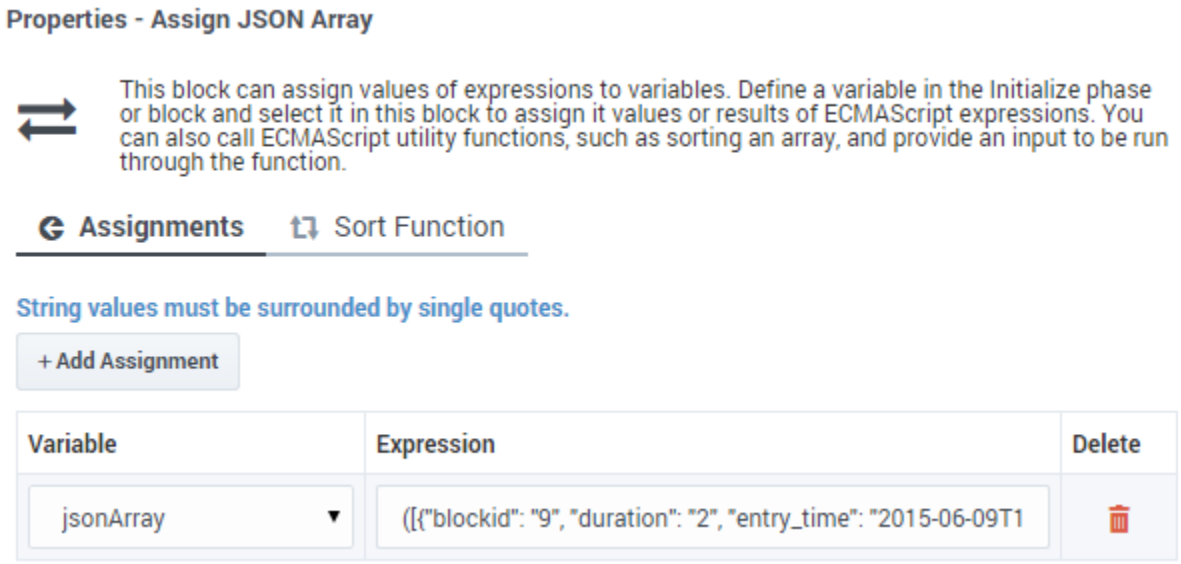
Assignments tab

Click **Add Assignment** to assign a value or expression to a variable.

- Select a variable from the **Variable** drop-down menu.
- Enter a value or expression for the variable in the **Expression** field. The value can be a simple literal value (such as a string, integer, or Boolean) or any valid JavaScript expression. The value expression

can refer to other variables.

> **Tip**
> When assigning a string value to a variable, you must ensure that you enclose the string value with quotation marks. Otherwise, the string is interpreted as a reference to a variable.



Sort Function tab

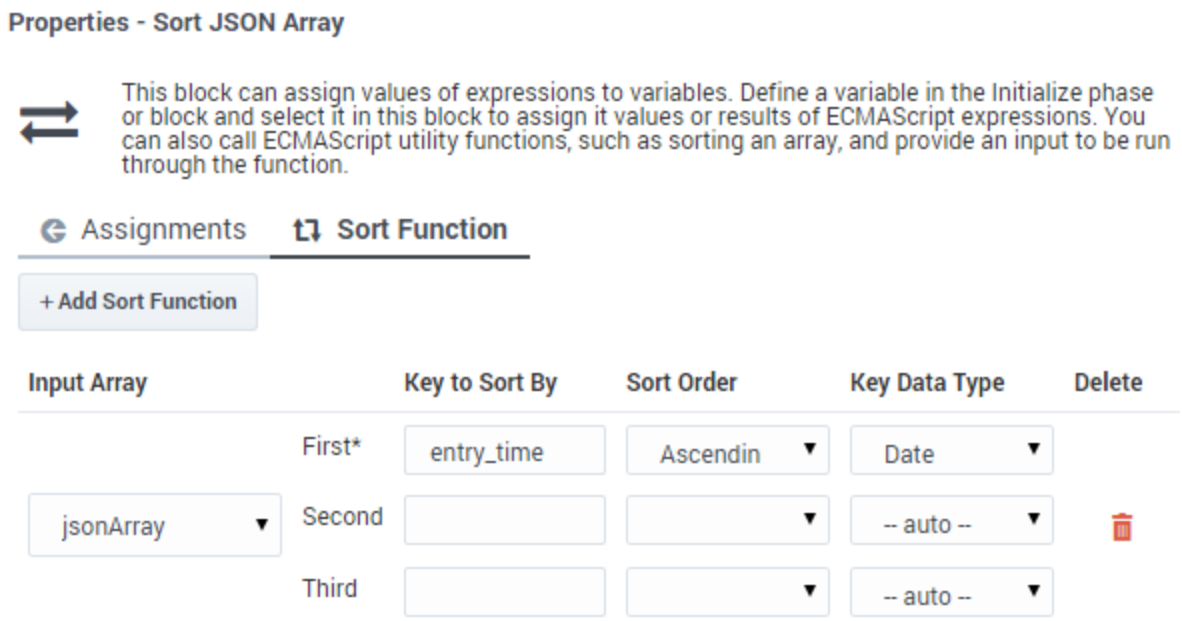
Click **Add Sort Function** and select an **Input Array** that contains a valid JSON array with values that you want to sort.

Enter up to three values in the **Key to Sort By** fields. These values must exist in the array. If an element does not have the specified value, it is skipped by the sort function and the value appears towards the end of the list.

Optionally, specify a **Sort Order** and **Key Data Type**. For the **date** type, you can use the following formats:

- *yyyy-MM-ddTHH:mm:ssZ*
  - Example in Greenwich Mean Time (Zulu): `2015-06-01T12:13:14Z`
- *yyyy-MM-ddTHH:mm:ss[+-]HHmm*
  - Example in Pacific Time: `2015-06-01T12:13:14-0800`

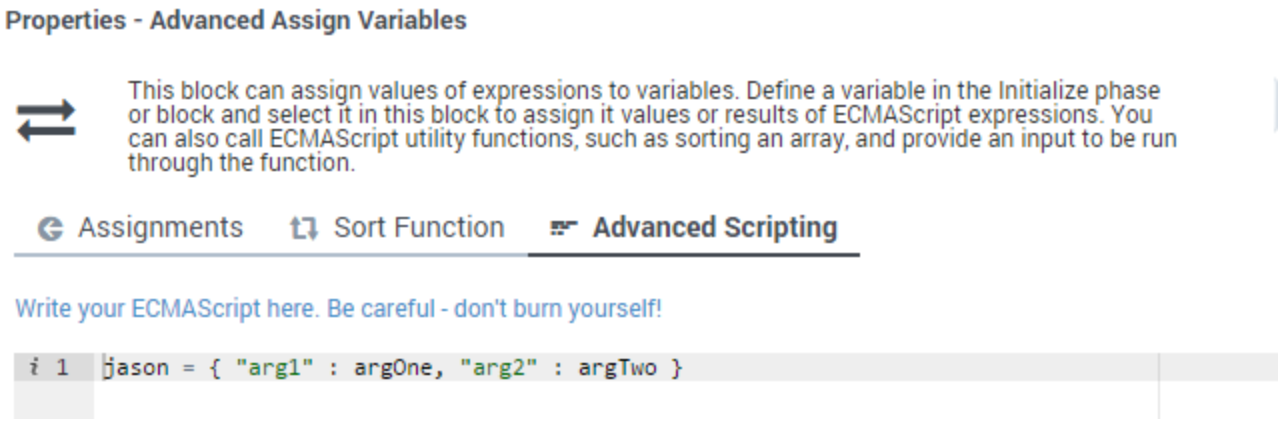
> **Tip**
> You can specify a **Key Data Type** to use data-aware sorting to treat different keys differently. If you do not specify a data type, the sort function treats all sort keys as strings and sorts those strings.



Advanced Scripting tab

> **Important**
> Advanced Scripting is an optional feature and might not be enabled on your system. To enable this functionality, contact Genesys.

Click **Advanced Scripting** to enter your own ECMAScript expression.


Properties - Advanced Assign Variables
This block can assign values of expressions to variables. Define a variable in the Initialize phase or block and select it in this block to assign it values or results of ECMAScript expressions. You can also call ECMAScript utility functions, such as sorting an array, and provide an input to be run through the function.
Assignments
Sort Function
Advanced Scripting
Write your ECMAScript here. Be careful - don't burn yourself!
1
jason = { "arg1" : argOne, "arg2" : argTwo }

Change Language Block

> **Important**
>
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

You can use the **Change Language** block to change the language of the application. This also changes the language in which audio resources are played.

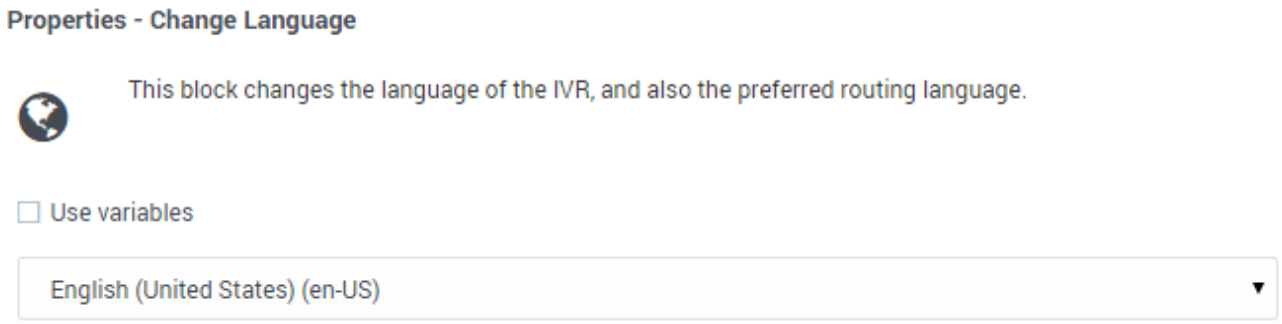
Typically, you use this block to switch languages once the caller's language preference is determined. This may be determined by prompting the caller to select his preferred language, using logic in the application (for example, a call that is routed to a regional contact center might use a default language setting for each region), or a RESTful API call into a customer preferences database that returns the preferred language.

You can use the **Change Language** block in any phase of the application.

> **Important**
>
> - If a Self Service shared module called from the Self Service phase of the application changes a language, that language stays in effect when the module returns to the calling flow.
> - You must upload audio resources for all languages used in the application before the application is run.
> - If you set the language of an application to a variable, you must ensure that the associated audio files have been uploaded before the application is run. Otherwise, if these audio files are missing, no audio is available to play during the call. Designer cannot detect this error when you click **Publish** to validate and save your application.

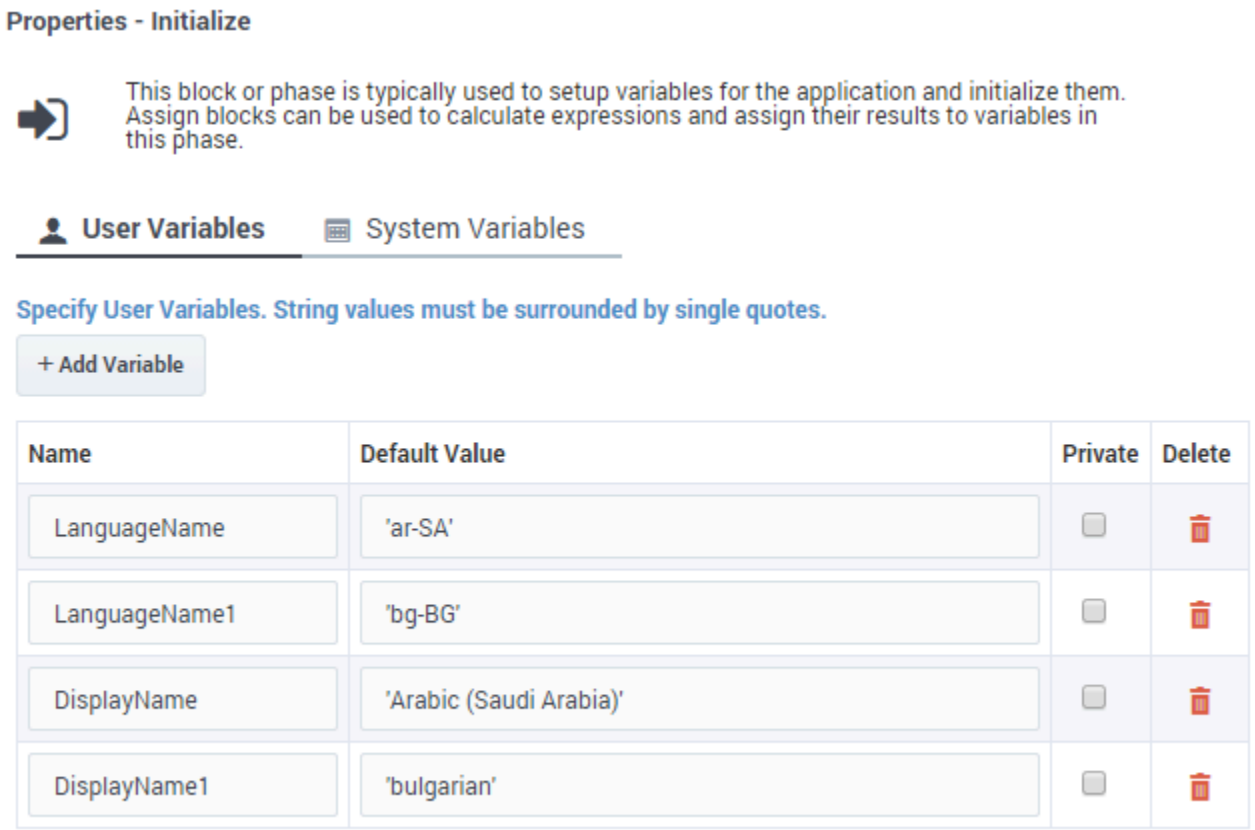
Using this Block

You can select a language from the drop-down menu:

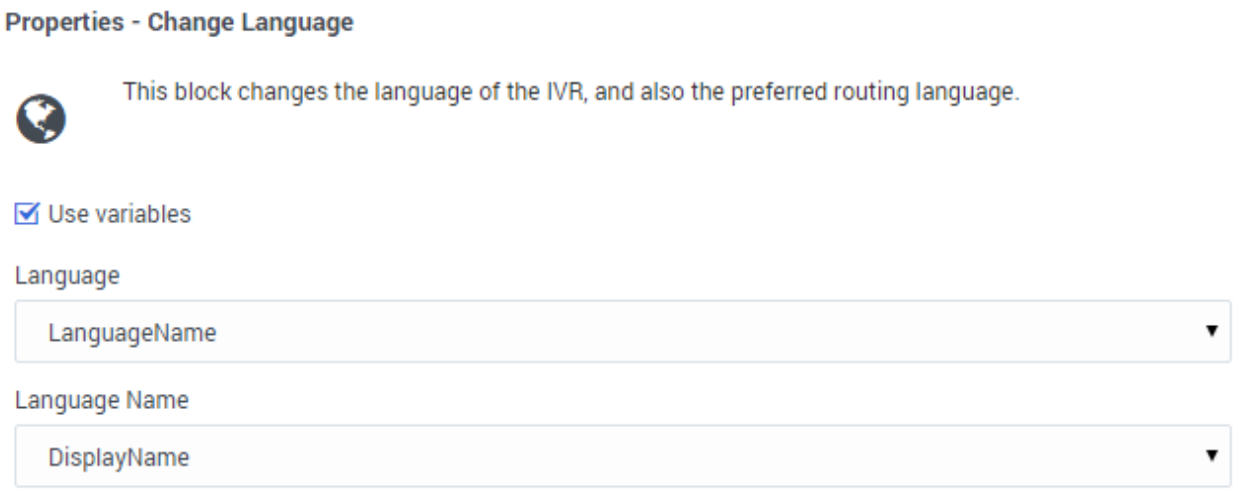


Or, you can select the **Use variables** check box to specify variables for the Language and Language Name. Here's an example of how to do this:

- First, specify your language variables in the **Initialize** phase:



- Then, select them in the **Change Language** block:


Properties - Change Language
This block changes the language of the IVR, and also the preferred routing language.
Use variables
Language
LanguageName
Language Name
DisplayName

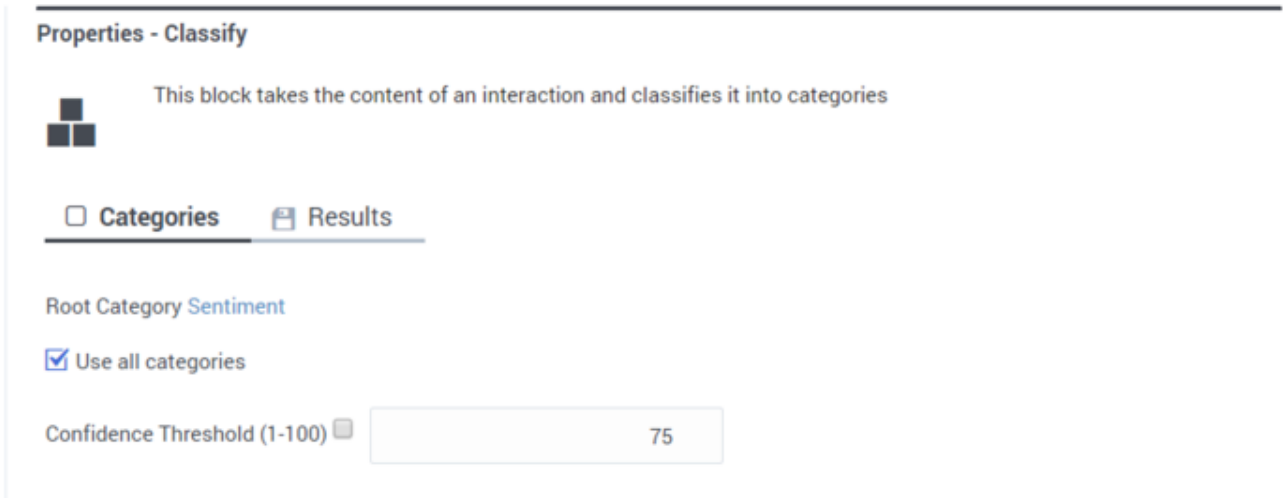
Classify Block

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

You can use this block to classify a digital interaction based on certain keywords in the content.

Categories tab

Use the settings on this tab to select the categories that Designer will use to classify the interaction and the confidence threshold to be applied.



If you use individual categories, segment blocks are created. When the application runs, the segment associated with the highest scored category is executed (similar to a Segmentation block).

Results tab

Specify the variables in which to store the **outcome** and **results** of the classification operation and the **name** and **relevancy** of the most relevant category.

Click-to-Call-In Match Block

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

You can use this block to support the **Click-to-Call-In** feature. In a typical **Click-to-Call-In** scenario, a customer dials a contact center and provides some additional information during the Self Service phase, such as the reason why they are calling. They are then given a phone number to dial along with a special access code.

The application uses this block to compare the customer's information with the details of the original request, such as the access code they entered, the number they called from, or the number they dialed. If a match is found, the additional information that was submitted with the original request is returned to the block.

Depending on the matching criteria used, the match block then generates all combinations for the output as segmentation branches, where each segmentation has its own **Navigation** tab that specifies how to handle the interaction.

For more information about Click-to-Call In functionality, see Provisioning the Click-to-Call-In Scenario.

Using this block

Typically, you would add a User Input block ahead of this block to prompt the caller for their access code and specify the variable that will hold it. Then use the **Click-to-Call-In Match** block to specify the matching criteria and the variable that will hold the returned data, if a match is made.

For example, if all three criteria are used, this creates eight possible outcomes (or segmentation branches), including an outcome where **Nothing Matched**. You can then use a Call Data block to attach the returned data to the current interaction.

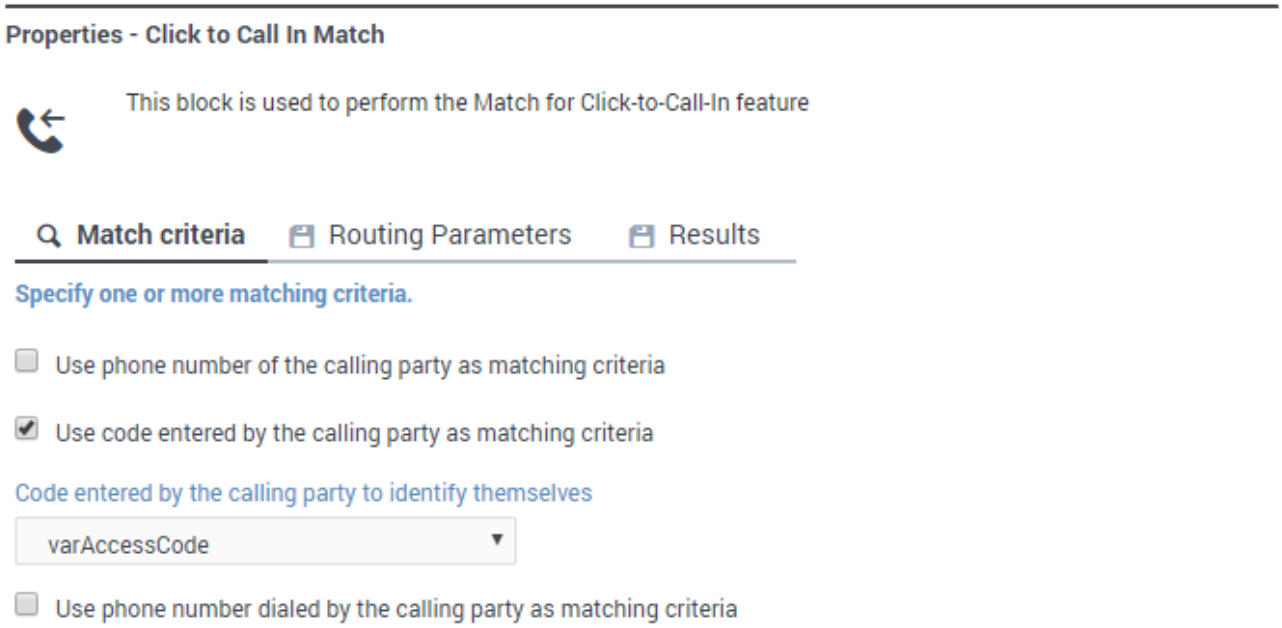
If there are no matches, you can use the **Navigation** tab on the **Nothing Matched** block to go to a specified block, skipping the **Call Data** block.

Match Criteria tab

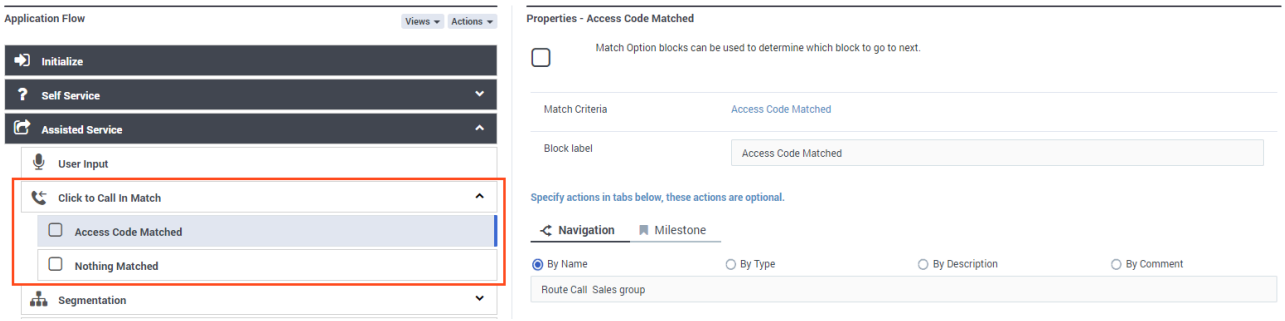
Use the settings on this tab to specify which criteria to compare to the original request, and then

specify the variable which holds this value. You must specify at least one item.

Example



Notice that Designer automatically creates child blocks for each outcome, which you can then use to specify the action to take if a match is found:



Routing Parameters

For Click-to-Call-In delayed scenarios, routing parameters are returned if there is a unique match. Use the settings on this tab to specify the variables that will hold the values for each parameter.

You can select variables for the following parameters:

- Outbound VQ
- Target Skill Expression
- Current Priority
- Priority Increment Interval
- Priority Increment
- Maximum Priority
- Notification Timestamp (when the push notification was initiated)

For more information, see Click-to-Call-In (Delayed).

Results tab

Specify the variable that will hold the retrieved storage data if a match is found.

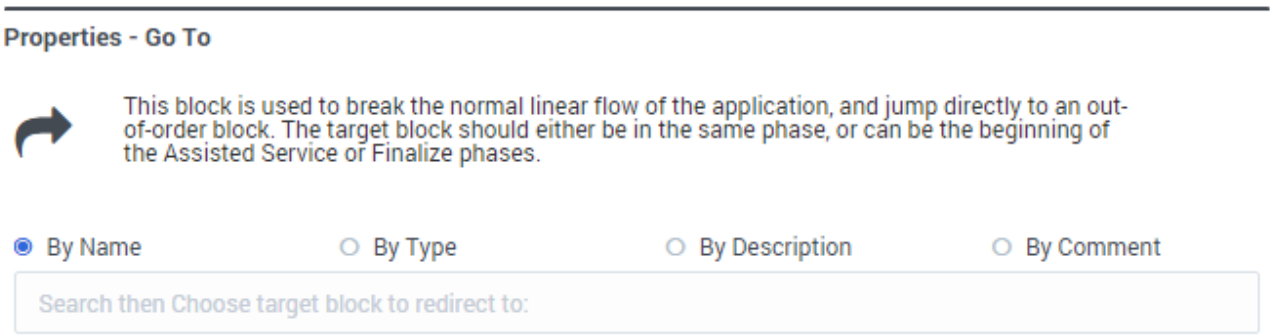
Go To Block

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

You can use the **Go To** block to enable transitions to other blocks in the same phase of an application or to the beginning of the **Assisted Service** or **Finalize** phase. You cannot use the **Go To** block to transition directly to **Menu Option** or **Segmentation Option** blocks.

Using this Block

Use the radio buttons to select a search criteria (for example, by **Name**, **Type**, **Description**, or **Comment**), then start entering the term you are looking for. Designer starts returning the results as you type.



Return Block

> **Important**
>
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

The **Return** block is available only for Shared Modules and is used to return control from the Shared Module to the application or Shared Module that called it. Multiple **Return** blocks can be used in a single Shared Module.

This block can return values of any output variables from the Shared Module. In the **Initialize** phase, variables can be marked as **Output** variables that are expected to be returned from this Shared Module. Only those variables can be assigned return values in the **Return** block.

Using this Block

Click **Add Assignment** and specify output variables.

Segmentation Block

> **Important**
>
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

You can use a **Segmentation** block to take a different path depending on the specific values of application variables. A valid ECMAScript expression containing application variables, ECMAScript operators, and Designer functions can be used to define a Segmentation Option. If this condition is evaluated to a *true* (Boolean) value while the application executes, the application flow takes the path of that Segmentation Option.

You can define multiple Segmentation Options, each with their own conditions. For example, the condition can be a variable with a Boolean value, a call to a function that returns a Boolean, or a combination of variables with logic operators that evaluates to a Boolean.

The first condition that evaluates successfully is selected as the segmentation path, and any blocks under that Segmentation Option are executed. If no condition expression evaluates successfully, none of the Segmentation Options execute, and the application executes the block that follows the **Segmentation** block.

Application variable values can be set based on logic in the application, by querying external data sources from blocks (such as the HTTP REST block), or by collecting input from a caller in the User Input block.

Conditions are ordered and exclusive, which means:

- Condition expressions are evaluated in the order they are defined.
- If one condition evaluates to true and the corresponding path is selected, then the following condition expressions are not tested. After executing this segment path, the application executes the block that follows the **Segmentation** block.

> **Tip**
>
> If the same logic needs to be executed in multiple segmentation paths, Genesys recommends that you keep the paths for each option independent and avoid using **GoTo** blocks to jump between paths. The common logic can be moved into a Shared Module, which can then be called from multiple paths. This improves the structure and reliability of your application.

The **Segmentation** block selects the first segment whose condition is a valid ECMAScript expression

that evaluates to *true* (Boolean). If none of the conditions evaluate to *true*, no segment is executed, and processing moves on to the next sibling of the **Segmentation** block.

> ### Warning
> You must use condition expressions that evaluate to a Boolean value. Expressions that evaluate to a different data type can result in errors.

The following are valid expressions:

- Using a variable whose value is *true* or *false* and comparing it to a Boolean value, such as the variable used to hold the result of a **Special Days** block:

  ```
  isSpecialDayVar == true
  ```

  or:

  ```
  isSpecialDayVar == false
  ```

- Using a Boolean property of an object stored in a variable, such as the **Route Call** block outcome variable:

  ```
  routeCallOutcomeVar.success == true
  ```

- An expression using Boolean variables and logical operators (&&, ||):

  ```
  var1 == false || (var2 == true && var3 == true))
  ```

- An expression using comparison operators (==, ===, !=, !==, >, <, >=, <=):

  ```
  var1.length > 3 || var2 === 'stop'
  ```

> ### Important
> Do not use an expression that does not have explicit comparison operators (such as `varIsHoliday`).

When using condition expressions that do not evaluate to a Boolean value, the following rules apply:

> ### Important
> - These rules are general ECMAScript/Javascript rules, and apply as-is to Designer. (This not the regular "flavor" of Javascript that runs in a browser. Instead, this Javascript is executed by Genesys platform components and has certain restrictions.)
> - It is mandatory to ensure these expressions evaluate to a Boolean value and not to other data types or values, such as *undefined*.

- The condition expression evaluates to an object => the condition is considered *true* (this applies to arrays, even if they are empty). Instead, use the following:

  ```
  typeof myVar === 'object'
  ```

- The condition expression evaluates to *undefined* => the condition is considered *false*. Instead, use the following:

  ```
  myVar !== undefined
  ```

- The condition expression evaluates to *null* => the condition is considered *false*. Instead, use the following:

  ```
  myVar !== null
  ```

- The condition expression evaluates to a number => the condition is considered *false* if the value is `+0`, `-0`, or `NaN`; otherwise, the condition is considered *true*. Instead, use the following:

  ```
  myVar === 3
  ```

- The condition expression evaluates to a string => the condition is considered *false* if the string is empty; otherwise, the condition is considered *true*. Instead, use the following:

  ```
  myVar.length > 0
  ```

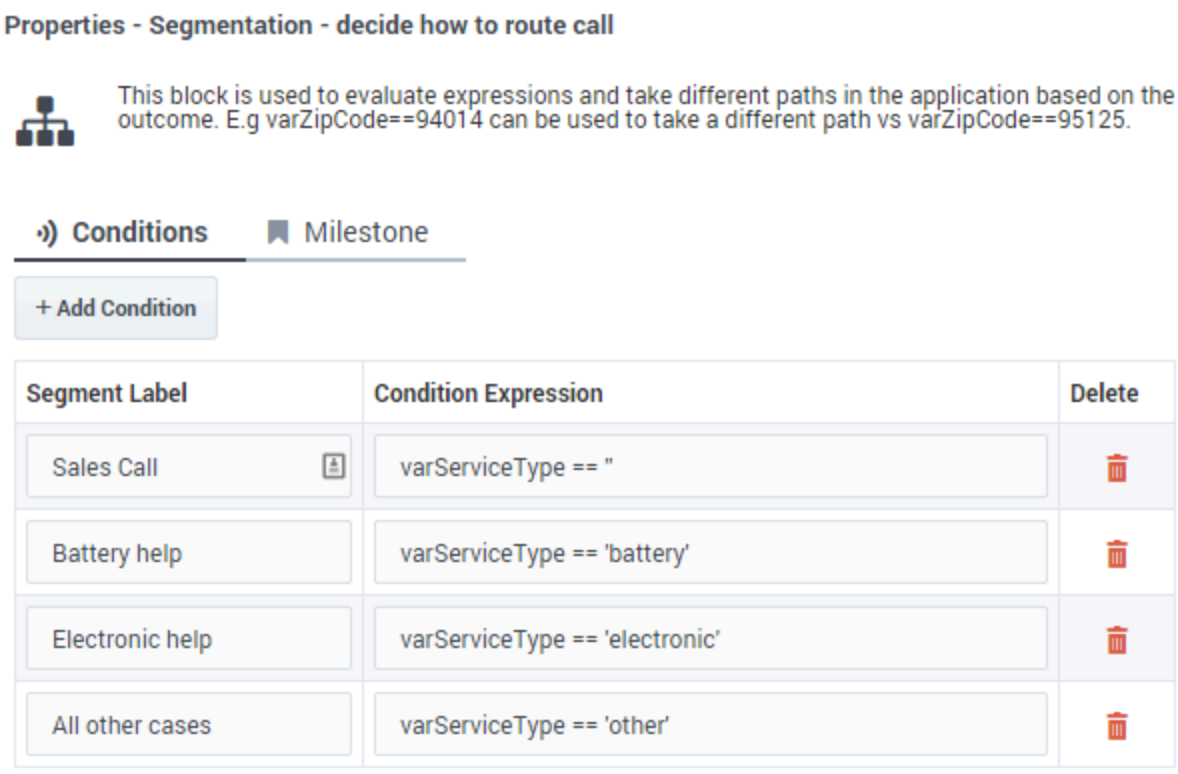
Conditions tab

Click **Add Condition** and type the condition to evaluate in the **Condition Expression** field. The value can be a simple Boolean value, a variable with some Boolean content, or any valid JavaScript expression that evaluates to a Boolean. The condition expression can refer to other variables.

You can edit the **Segment Label** field to give a meaningful label to your segment. The child segment block will be named accordingly.

To remove a condition, click the trash icon for that condition in the **Segmentation** block or click the trash icon on the related child block.

> ### Important
> Always make sure the condition evaluates to a Boolean value at runtime.



Milestone tab

Add a Milestone to mark this key moment while the application is running, similar to within the Milestone block.

Shared Module Block

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

Shared Modules are useful for reusing code from multiple applications, as well as for splitting larger applications into smaller and more manageable chunks. Once you have created a Shared Module, you can use the Shared Module block to invoke the module into your application.

If you change a Shared Module, you also change all of the applications that use that module. If an application uses the **Latest** version of a module, and the application is published, it starts using the new state of the Shared Module. If an application uses a specific version of the Shared Module (not the **Latest**), it does not receive the latest changes in the Shared Module, even if the application is published again.

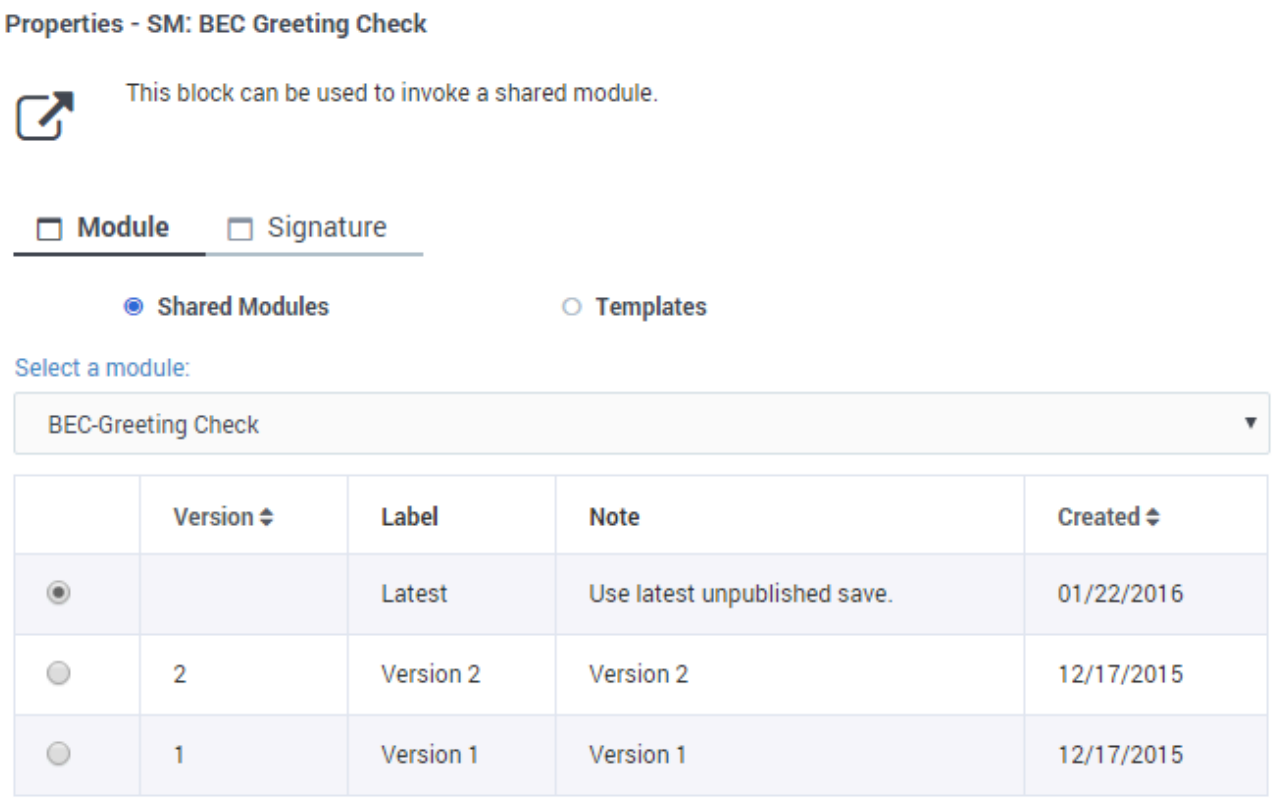
For more information about how to create and manage shared modules, see Shared Module.

Module tab

Select a Shared Module or Template.

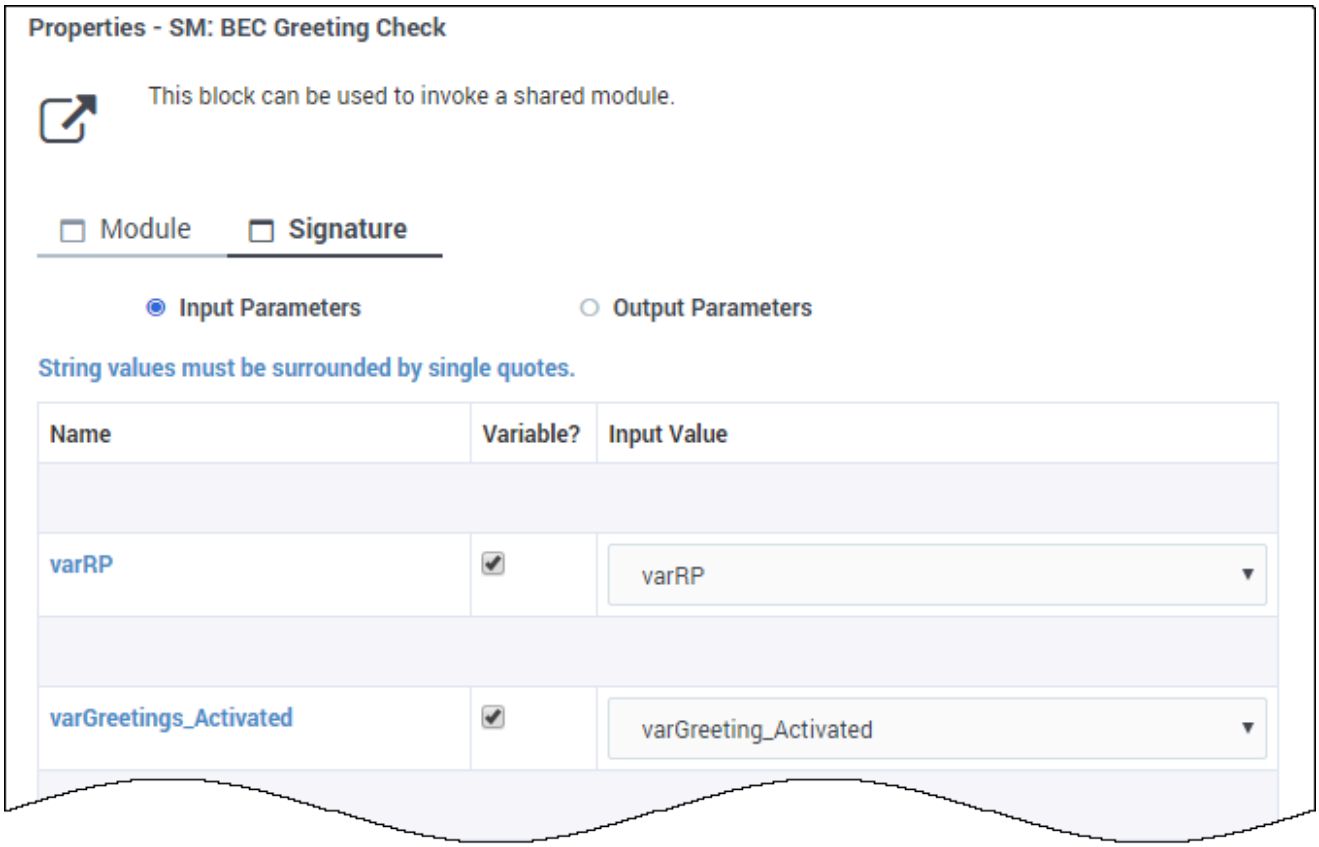
All Shared Modules that have at least one version are listed. Once a Shared Module is selected, all published versions of the module are shown. Usually the latest version should be selected, unless there is a incompatibility with the latest version.

Templates are used only with the Callback block. They are read-only and cannot be edited or deleted.



Signature tab

Specify the values for the **Input** and **Output Parameters**. You can use literals, variables, or expressions.


Properties - SM: BEC Greeting Check
This block can be used to invoke a shared module.
Module
Signature
Input Parameters
Output Parameters
String values must be surrounded by single quotes.
Name
Variable?
Input Value
varRP
varRP
varGreetings_Activated
varGreeting_Activated

Terminate Call Block

> **Important**
>
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

You can use the **Terminate Call** block to disconnect the caller and stop the call. Everything after the **Terminate Call** block is skipped, and the application moves straight to the **Finalize** phase.

As a visual aid, the right edge of the **Terminate Call** block is capped in red, to show that the application will stop if and when it reaches this block. This visual aid also applies to any block that might end the call, such as Business Hours or Special Day, when the Terminate Call option is enabled.

Terminate Block (Digital)

> **Important**
>
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

You can use the **Terminate** block in a Digital application type to end an interaction.

When used in the **Self Service** or **Assisted Service** phases, everything after the **Terminate** block is skipped and the application moves straight to the **Finalize** phase.

When used in the **Finalize** phase, the application sets the termination flag and moves to the next block.

As a visual aid, the right edge of the **Terminate** block is capped in red to show that the application will stop if and when it reaches this block. This visual aid also applies to any block that might end the call, such as Business Hours or Special Day, when the Terminate option is enabled.

Change Persona block

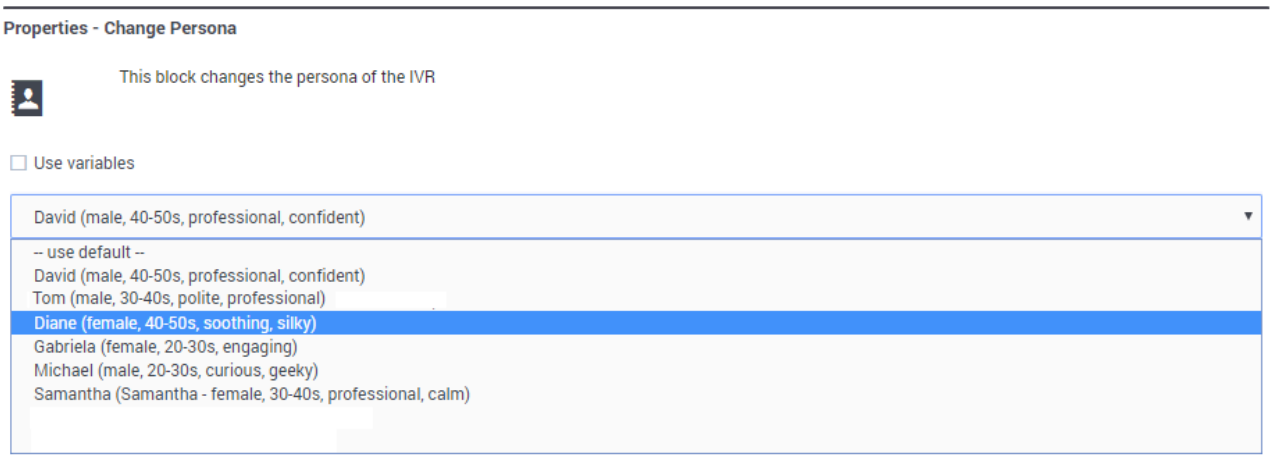
> **Important**
>
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

This block enables you to dynamically change the persona being used by the application throughout the course of an interaction.

Switching personas can be useful when you want to use certain personas for specific situations. For example, you might want to use a more formal persona when dealing with sensitive customer issues, or switch to a different persona depending on the customer segment being served ("Gold" customers get one type of persona, "Blue" ones get another).

Using this block

You can add this block to the Self Service or Assisted Service phases of the application, or use it in a Shared Module. After you've added the block, select the persona (or the variable) you want to use.



The selected persona will apply to any blocks that are using Text-to-Speech (TTS) services, such as Play Message, User Input, Menu, Bot, Record Utterance, Business Hours, Emergency, Special Day, Route Call, and Route Agent.

For more information, see Personas.

User Interaction Blocks

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

The blocks in this category are used to interact with callers in various ways, such as to offer them a list of menu options ("Press 1 to speak with an agent"), collect their information (such as an account number), play them a message, or record their call (or a selected portion of the call).

The blocks shown depend on the features that are enabled and the type of application that is being built. For example, only Digital type applications will see blocks related to email.

Use the links below to learn more about each block.

Bot

Add a chatbot resource to your application.

Used in: **Self Service**

Menu

Presents a list of choices to callers.

Used in: **Self Service**

Play Message

Plays audio messages to callers.

Used in: **Self Service**, **Assisted Service**

Record

Starts or stops a call recording.

Used in: **Self Service**

Record Utterance

Records a user's voice or DTMF inputs.

Used in: **Self Service**

User Input

Collects information from callers.

Used in: **Self Service**, **Assisted Service**

Chat Message

Sends a chat message to a contact.

Used in: **Assisted Service**

Chat Transcript

Emails the chat transcript to a contact.

Used in: **Assisted Service**, **Finalize**

Get Chat Transcript

Provides access to the latest chat transcript.

Used in: **Assisted Service**, **Finalize**

Send Email

Sends an email containing a standard-response message to a user.

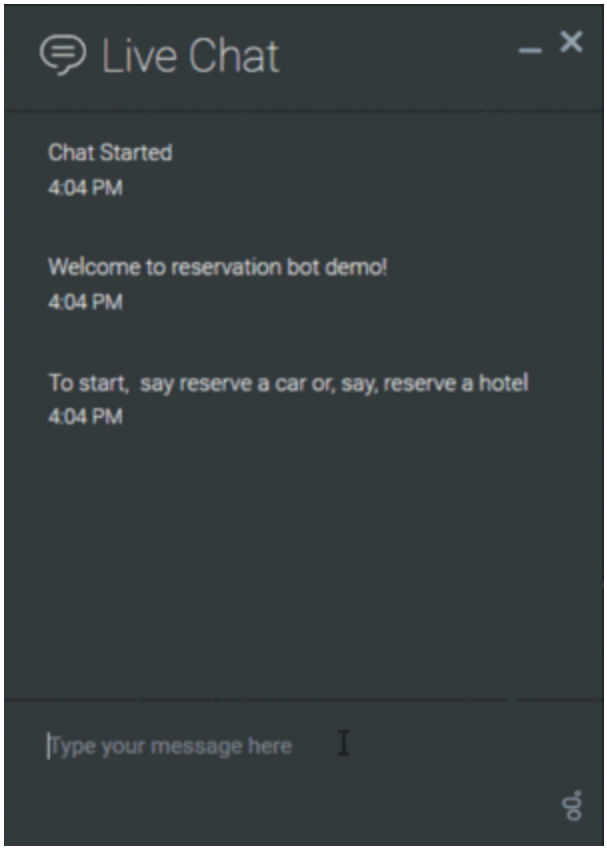
Bot Block

> **Important**
>
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

Add a **Bot** block to the Self Service phase of your application when you want to use a bot resource in your application. If you add a bot to a default type application that is enabled for omni-channel, the same bot resource can service both voice and chat customers.

Bots are software applications that apply automatic speech recognition and natural language understanding to listen and respond to customers in a way that resembles a conversation with a live agent. They can determine what a customer wants to do and then collect the information required to fulfill the request.

For example, they can help a customer set up a reservation:



Click to view larger

Designer bot resources are managed from the **Bot Registry**, where you can add and manage bot resources from your bot services provider (for a list of supported providers and details about adding bot resources to Designer, see the Bot Registry page). You can use multiple bots in an application, from any of the supported bot services providers.

Using this block

From the **User Interaction** section of the palette, drag and drop a **Bot** block to the **Self Service** phase of your application. After you have set up the block, you can then customize the application logic to specify additional handling, such as call a **Shared Module** for managing a specific type of intent or send the customer to a live agent for more assistance.

Watch: Use bots in applications

Link to video

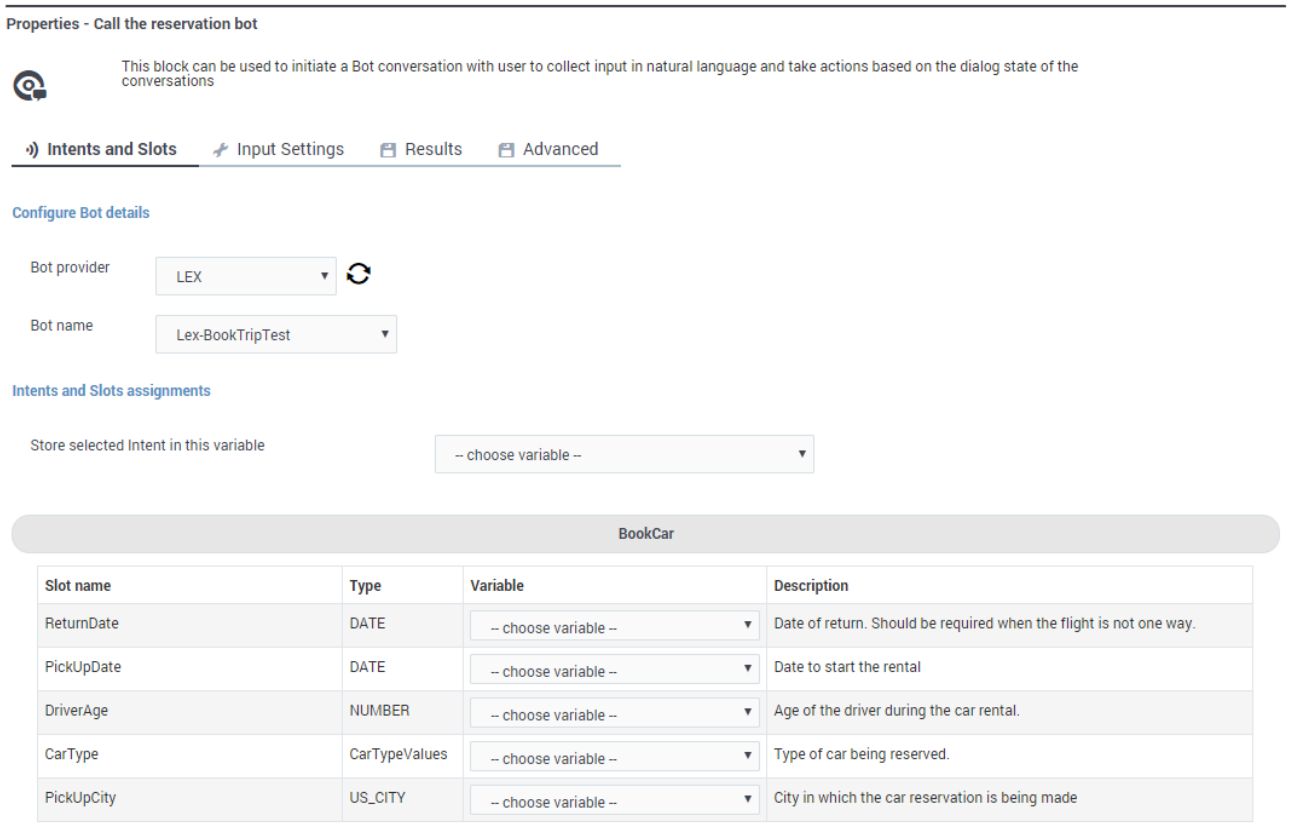
Intents and Slots tab

Use the settings in the **Configure Bot details** section to select the **provider** and **name** of the bot resource you want to use.

Designer then retrieves the details for your bot from the bot services provider and automatically populates the block properties with **Intents and Slots**. For each Intent, Designer automatically creates an associated child block (similar to how the Menu block creates child blocks for each menu option).

In the **Intents and Slots assignment** section, you can select the variable that holds the **Intent** (what does the customer want to do?) and values for the **Slots**. The **Slots** (or *entities*) help provide additional context to the intent.

For example, let's say our selected bot resource has an **Intent** called "Book Car". The bot detects that the customer wants to reserve a car, but it also needs to know other details about the customer's request, such as the type of vehicle they want, where they are picking it up, and when they plan to return it. The **Slots** are used to guide the bot as to the questions it needs to ask the customer (using natural-language) so it can collect this information and thereby fulfill the intent.



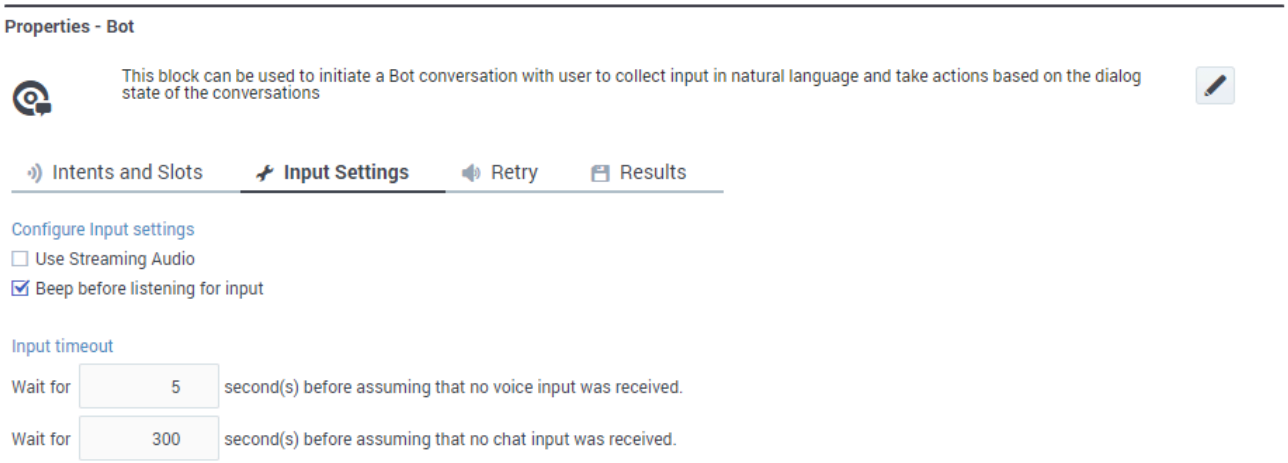
Input Settings tab

If you are setting up a DialogFlow bot, you can select **Use Streaming Audio** to have Designer stream the audio inputs directly to the bot services provider. This enables the bot provider to handle the transcribing of audio inputs and assigning them to the appropriate intents and slots, which can improve the performance of these types of bots.

The **Beep before listening for input** option plays a "beep" tone after the bot asks the customer for input. When enabled, the Bot block only recognizes the input that is received from the customer after the beep has played.

You can also use this tab to adjust the **Input timeout** values for both voice and chat inputs. These settings tell Designer how long to wait (in seconds) before assuming that the customer did not provide any input to the bot.

Use these settings to specify how many seconds should elapse before Designer assumes that no voice or chat input has been received from the customer.



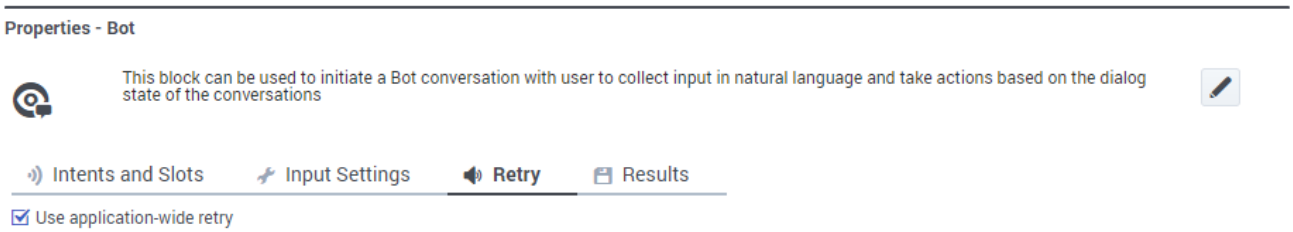
Retry tab

If a bot doesn't understand a response from a customer, it asks the customer to try again until it understands the input being provided.

The retry settings on the **Bot** block work a bit differently than the retry settings on other blocks, such as **Menu** and **User Input**, in that you specify how the **Bot** block will respond to input that isn't recognized on a *conversational* level, for each of the question and response exchanges that take place between the bot service and the customer.

Use application-wide retry

Select this option if you want to use the retry settings that are specified on the **Global Retry** tab in the Application Settings.



Allow retries

Select this option to specify specify retry rules for this block. When enabled, you can set the following options:

Number of No Input retries allowed

Select the number of retries to allow for each question and response sequence that occurs in the conversation between the bot service and the customer. For each retry, you can specify whether a prompt is played by clicking the corresponding section beneath this field.

For example, if you allow two no-input retries and you want to play a prompt after the first retry, select the **No Input #1** line and add a prompt. Enable the **Play original menu prompt after this retry prompt** check box to repeat the menu prompts for the customer.

Number of No Match retries allowed

Most bots will follow-up with the customer if they don't understand the input that's been provided. For example, the bot will simply ask the customer to repeat the information until it successfully captures the response.

As this type of handling is typically built-into the bot by the bot services provider, you may not need to specify this setting in the **Bot** block.

After Final No Input

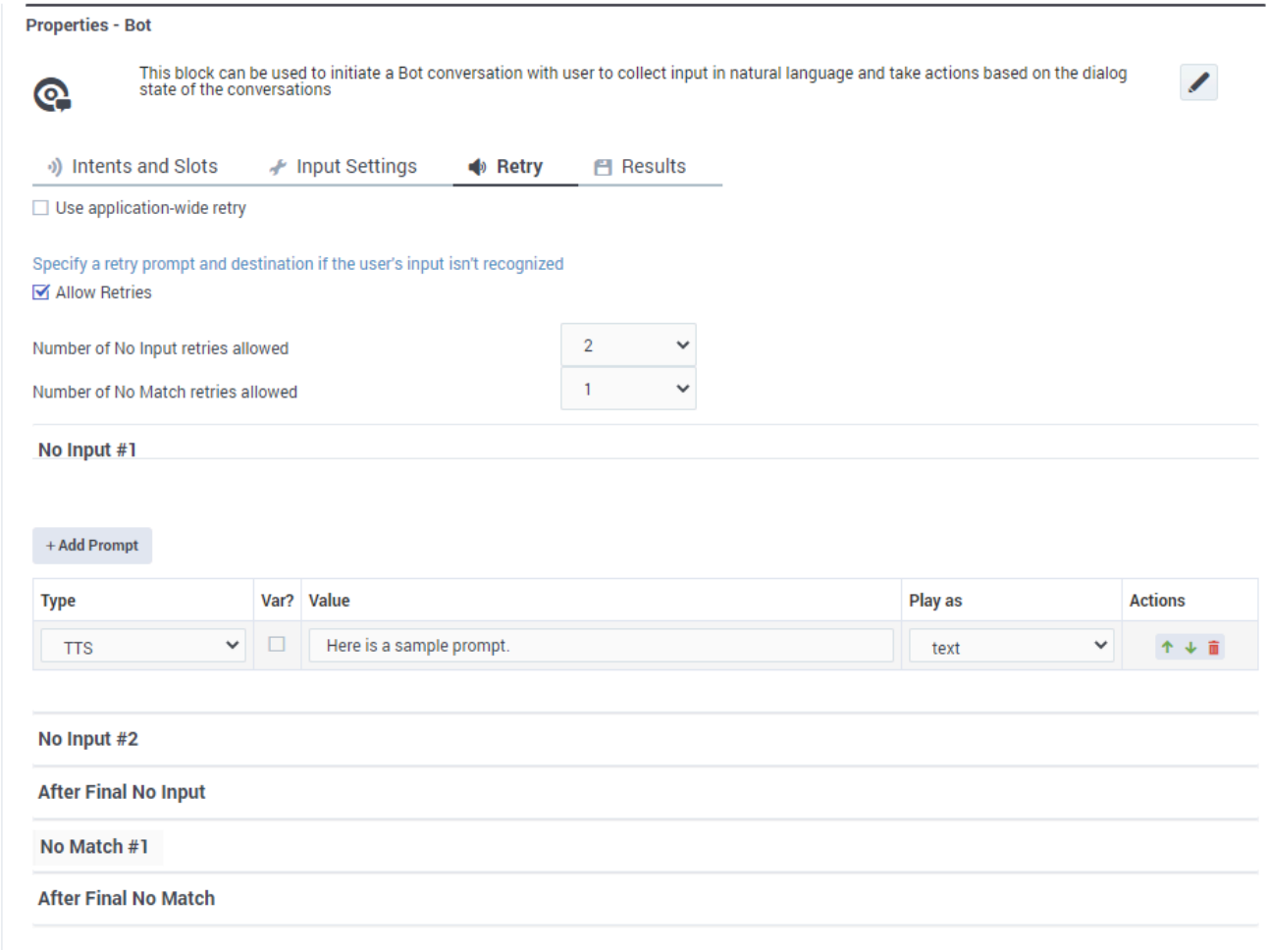
Add the prompt to play after the maximum number of permitted **No Input** retries is reached.

As this block is in the Self Service phase, you can also specify a target destination for the application to jump to, such as another block in the Self Service phase or to the Assisted Service or Finalize phase of the application.

After Final No Match

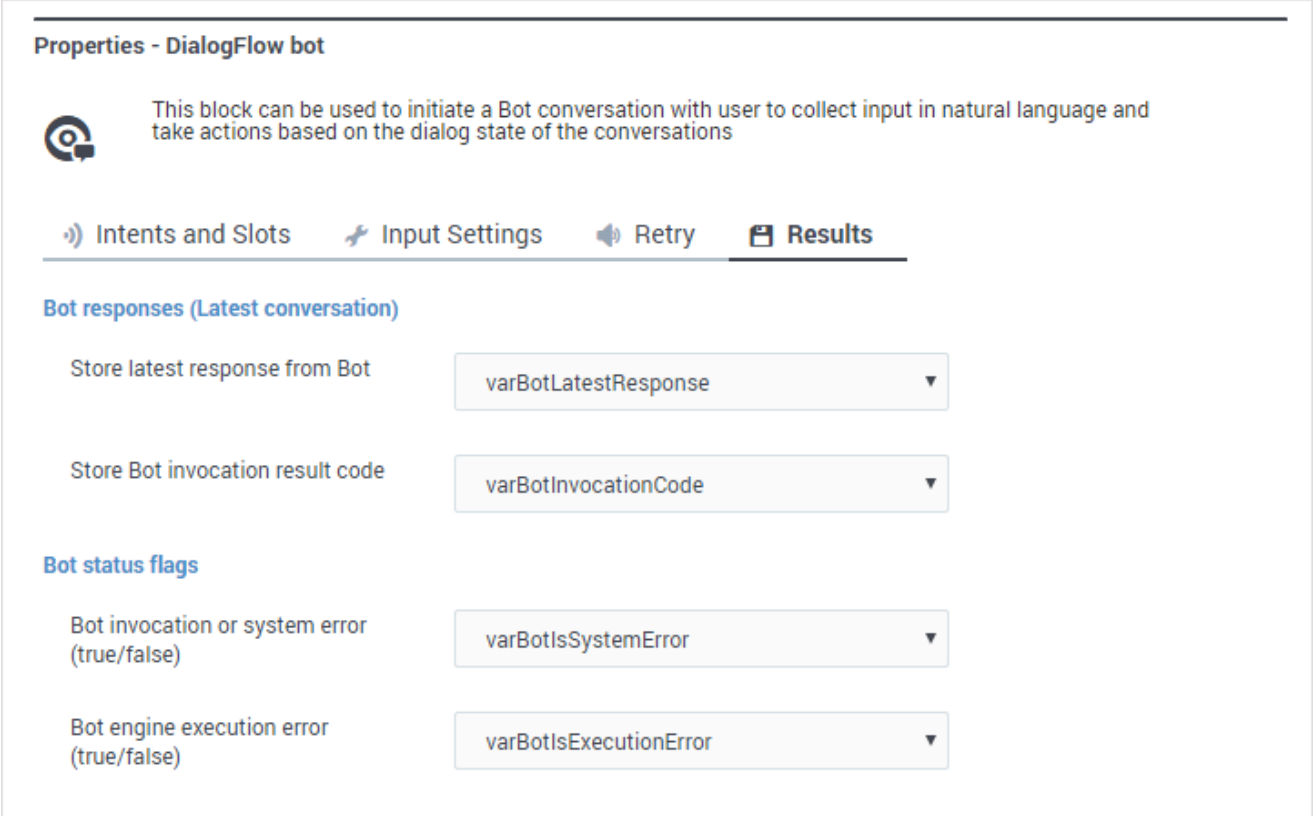
Add the prompt to play after the maximum number of permitted **No Match** retries is reached.

As this block is in the Self Service phase, you can also specify a target destination for the application to jump to, such as another block in the Self Service phase or to the Assisted Service or Finalize phase of the application.



Results tab

Specify the variables that will hold the **bot responses** and **status flags**, as returned to the **Bot** block from the bot services provider. Each variable is described in more detail below.



Bot responses

Store latest response from bot

This variable stores details about the latest conversation that the bot engine had with a customer. For example, the results for booking a ghost removal service might look like this (JSON formatted):

```
{
  "status": {
    "code": 0,
    "message": null
  },
  "data": {
    "botName": "MySampleServiceBookingBot",
    "botAlias": null,
    "sessionId": "ABC123",
    "state": "SUCCESS",
    "intent": "Book a Ghostbuster",
    "intentScore": 1,
    "slots": {
      "neighbourhoods": "Queens",
      "location": "backyard",
      "date": "2020-01-11T12:00:00-05:00",
      "ghost": "Zuul the Gatekeeper"
    },
```

```
    "slotsData": null,
    "inputTranscript": "today",
    "message": "",
    "attributes": {},
    "error": null,
    "recognitionConfidence": null,
    "stability": null
  }
}
```

In the example above, some of the details that were returned include:

- `botName`
- `sessionID`
- `state` – indicates SUCCESS if everything worked.
- `intent` – the *intent* that was detected (i.e. what the customer wanted to do).
- `slots` – these are the details that the bot collected from the customer to fill the associated *slots* (or *entities*) and fulfill the intent.
- `inputTranscript` – the *utterance* (voice or chat input) that the bot received from the customer.

Store bot invocation result code

This variable stores the HTTP status code received from the bot when it was last invoked by the application. For example, a result code of `200` (OK) indicates that the bot was successfully invoked.

Any other result code received, such as `401` (Unauthorized) or `403` (Forbidden), can indicate there was a problem reaching the bot service.

Bot status flags

Bot invocation or system error

If `true`, this indicates that Designer was not able to successfully reach or invoke the bot service. There could be an issue with the system, or you might need to check the credentials provided for your bot service in the Bot Registry. Otherwise, this returns `false`.

Bot engine execution error

If `true`, this indicates that Designer was able to communicate with the bot, but an error occurred while the bot engine was processing the request. For example, the bot returned an incorrect response and triggered the **Error Handler** block. Otherwise, this returns `false`.

Viewing the results data

You can view the results data in Designer Analytics by going to the Session Data Records dashboard. In the **All Events** panel, find the application instance you want to check and then filter or search for the data you want to view.

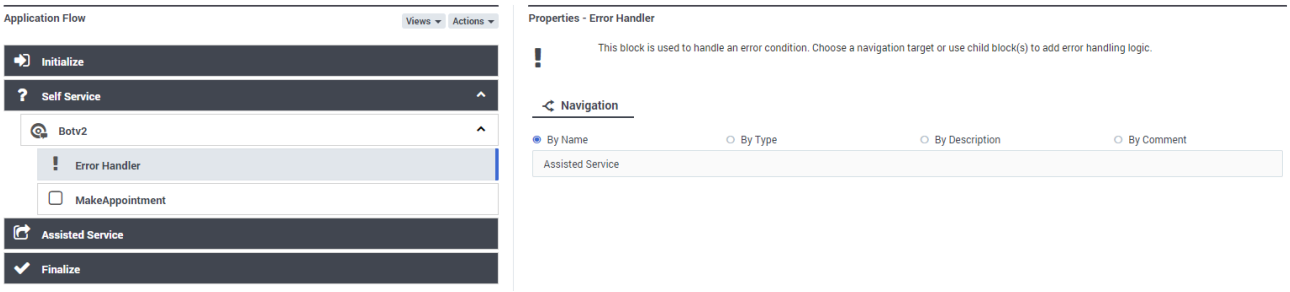
For example:

variables.varBotInvocationResultCode		200
variables.varBotIsExecutionError		false
variables.varBotIsSystemError		false
variables.varBotLatestResponse		{"status":{"code":0,"message":null},"data":{"botName":"w29-agt8-Ghostbusters-GDF","botAlias":null,"sessionId":"01LHORINN8F3B5BMI8BKG2LAES007A53","state":"SUCCESS","intent":"Book a Ghostbuster","intentScore":1,"slots":{"neighbourhoods":"Queens","location":"backyard","date":"2020-05-11T12:00:00-05:00","ghost":"Zuul the Gatekeeper"},"slotsData":null,"inputTranscript":"today","message":"","attributes":{},"error":null,"recognitionConfidence":null,"stability":null}}

Bot Option blocks

Additional **Bot Option** child blocks for each intent are also created, along with an **Error Handler** block. For each **Intent**, you can go to its related **Bot Option** child-block to edit its label and give it a more meaningful name.

In the properties for the **Error Handler** block, you can specify the action to take if an error occurs (for example, go to another block or the Assisted Service phase):



We can also add other blocks below the intent when you want to add some additional logic or processing, such as a **Shared Module** that will be called when that intent is detected.

Menu Block

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

You can use the **Menu** block only in the **Self Service** phase to present a list of choices to the customer and accept a selection that the customer provides.

You can choose to enable certain DTMF keys and associate specific processing or logical flow with those keys. For each DTMF key that is enabled, a new **Menu Option** block is shown in the **Application Flow**. You can then add new blocks to each of these **Menu Option** blocks.

> **Warning**
> The **Menu** block is not supported with **SMS** media.

DTMF Options tab

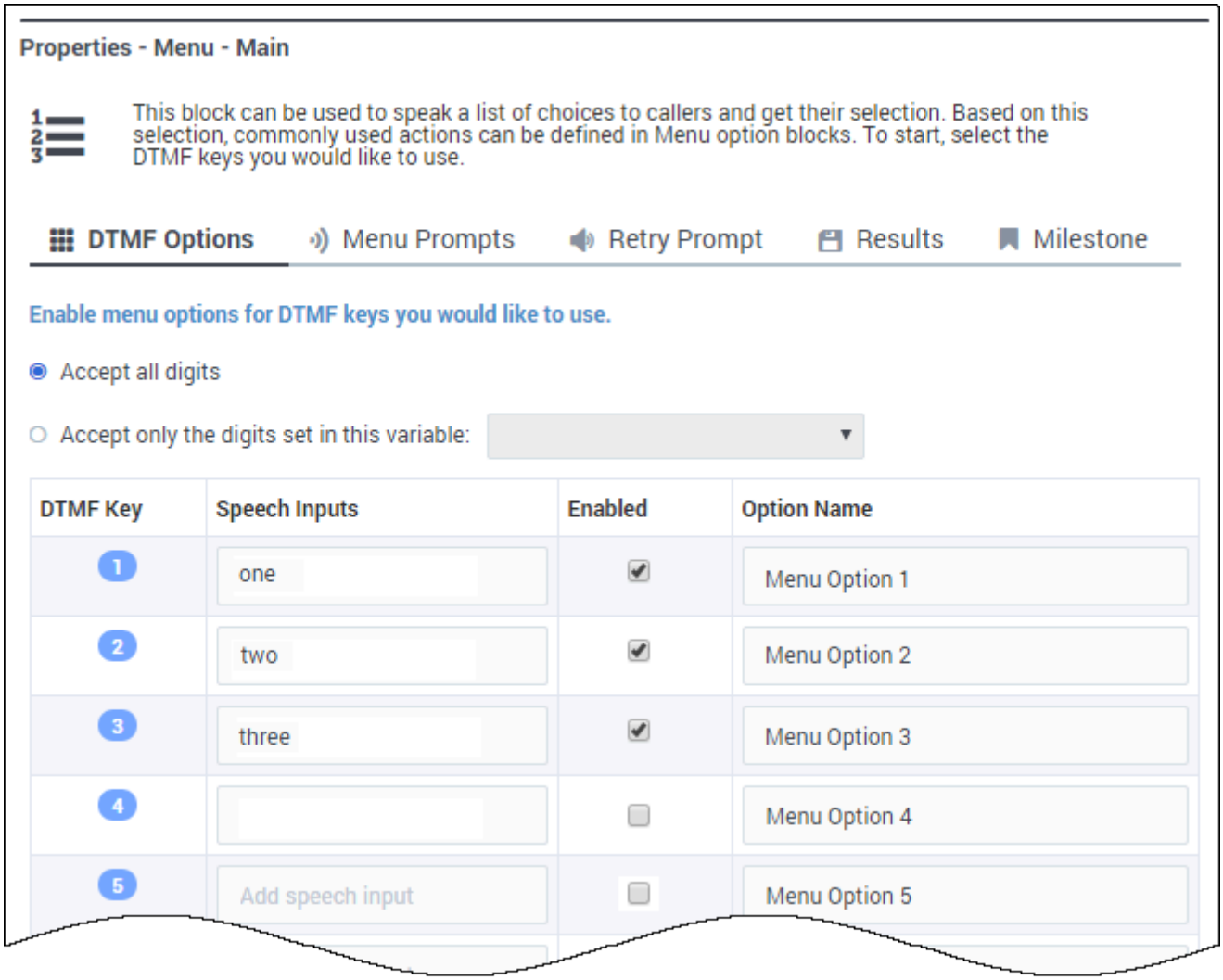
Select one or more DTMF keys, which enables a **Menu Option** block for each key.

Select **Accept all digits** or **Accept only the digits set in this variable**. Using the variable option allows you to set conditions for enabling or suppressing specific menu options while the application is running.

Use a descriptive **Option Name** to make it is easier to understand the flow.

Optionally, enter a valid speech input for each DTMF key in the **Speech Inputs** field.

Refer to the Menu Option block page for more information on how to configure **Menu Option** blocks.



Menu Prompts tab

Input timeout

Specify the number of seconds that the application should wait before assuming that no voice or chat input was received from the customer.

Disable barge-in

Select this option to prevent customers from interrupting a prompt while it is still playing. For example, you might want a "Welcome" message to play all the way through before the customer can enter another command and skip to the next menu prompt.

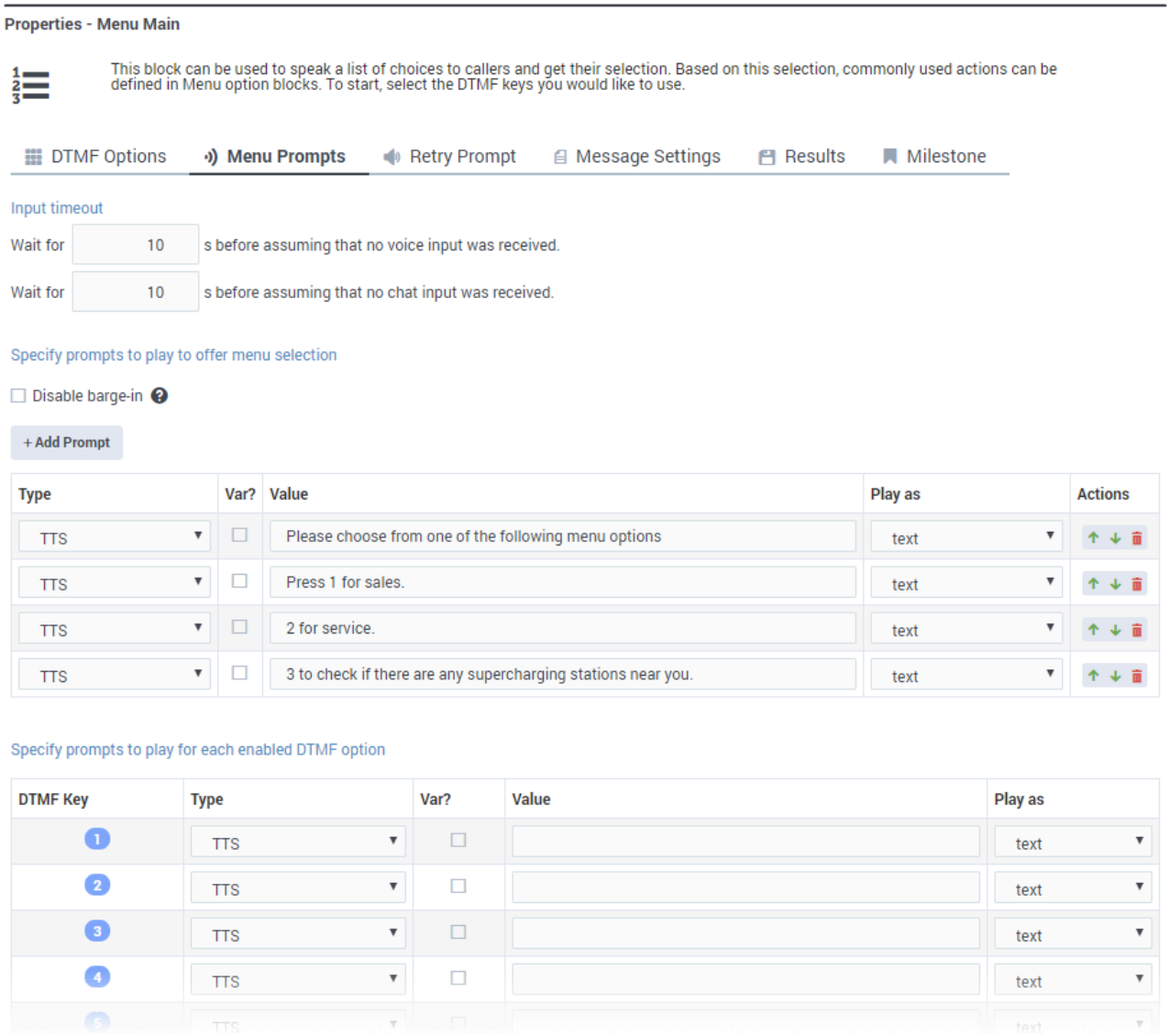
If this option is not selected, barge-in is enabled, and the prompt can be interrupted by the customer.

> **Important**
> The selected barge-in setting applies irrespective of whether global DTMF commands are used or not.

Click **Add Prompt** to play prompts when the menu starts.

> **Tip**
> See the Play Message block page for more information on how to create prompts.

You can also specify prompts to play for each enabled DTMF option. For chat applications, the chat widget displays the **DTMF Key** prompts as "quick reply" buttons. The customer can then select one of the quick replies instead of manually typing a response.



Retry Prompt tab

Allow Retries

Select to allow customers to provide late input or an unrecognized input. If enabled, you can set the following options:

- **Number of No Input retries allowed**
  Enter the number of retries to allow for customers whom do not provide input. For each retry, you can specify whether a prompt is played by clicking the corresponding section beneath this field. For example, if you allow two no-input retries and you want to play a prompt after the first retry,

select the **No Input #1** line and add a prompt. Enable the **Play original menu prompt after this retry prompt** check box to repeat the menu prompts for the customer.

- **Number of No Match retries allowed**
  Enter the number of retries to allow for customers whom do not provide a match for a **Menu Block**. For each retry, you can specify whether a prompt is played by clicking the corresponding section beneath this field. For example, if you allow two no-input retries and you want to play a prompt after the first retry, select the **No Match #1** line and add a prompt. Enable the **Play original menu prompt after this retry prompt** check box to repeat the menu prompts for the customer.
- **After Final No Input**
  Add the prompt to play after the maximum number of permitted No Input retries is reached. You can also specify a target destination for the application to jump to, such as another block in the Self Service phase or to the Assisted Service or Finalize phase of the application.
- **After Final No Match**
  Add the prompt to play after the maximum number of permitted No Match retries is reached. You can also specify a target destination for the application to jump to, such as another block in the Self Service phase or to the Assisted Service or Finalize phase of the application.

Results tab

Select variables to store the user's DTMF selection and the outcome of the interaction.

Milestone tab

Add a milestone to mark this key moment while the application is running. See the **Milestone** block page for more information.

Menu Option Block

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

Menu Option blocks appear in the **Application Flow** after you enable at least one DTMF key in a Menu block.

> **Important**
> When configuring Menu block options, Genesys recommends that you keep the branches of each option independent and use Shared Modules to share any functionality between them (rather than pointing to the child block of another option within the same branch). This improves the efficiency and reliability of your application.

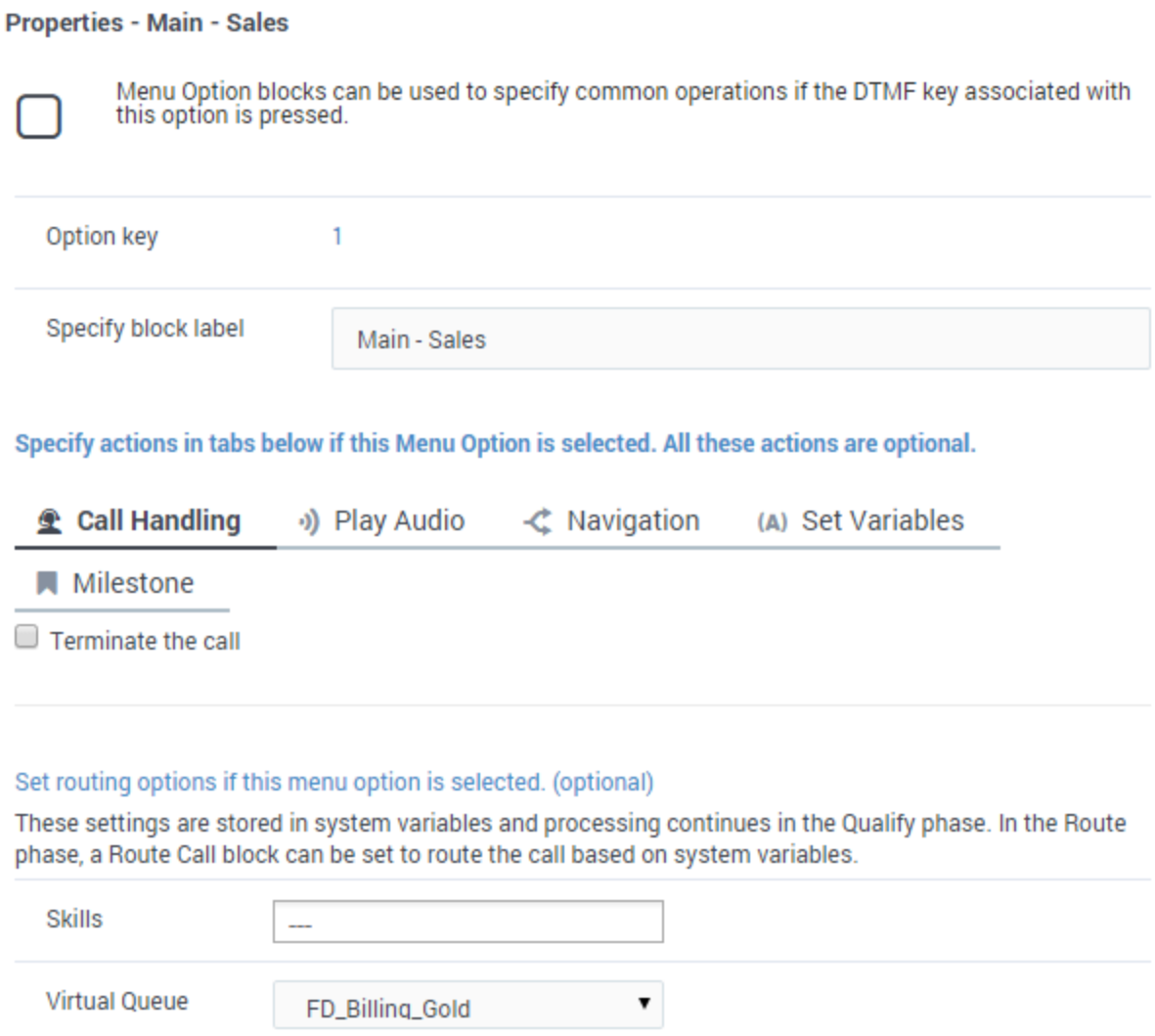
Call Handling tab

Terminate the call

Enable this option to terminate the interaction if this menu option is selected by the user.

Optionally, you can choose to route this interaction if this menu option is selected by the user. If so, select a Skill and Virtual Queue to which the interaction will be routed. These selections are stored to the **RoutingSkills** and **RoutingVirtualQueue** system variables, respectively.

> **Important**
> If you set these routing options, Designer does not route the interaction unless a Route Call block is added to the **Assisted Service** phase that routes based on menu options.



Play Audio tab

Disable barge-in

Select this option to prevent users from interrupting a prompt while it is still playing. For example, you might want a "Welcome" message to play all the way through before the user can enter another command and skip to the next menu prompt.

If this option is not selected, barge-in is enabled, and the prompt can be interrupted by the user.

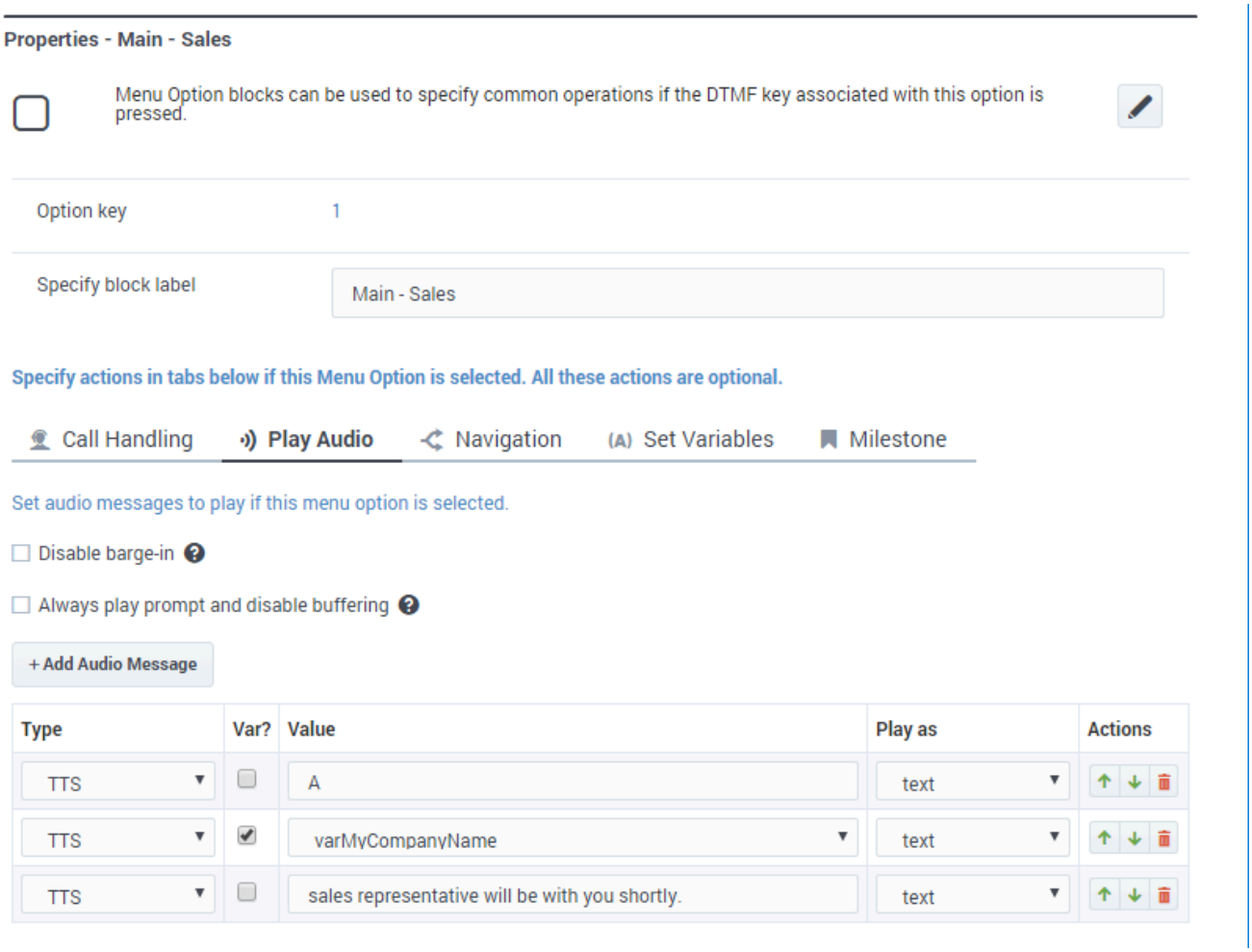
> **Important**
> The selected barge-in setting applies irrespective of whether global DTMF commands are used or not.

Always play prompt and disable buffering

Select this option if you want users to be able to interrupt a prompt while it is playing, but not have those inputs applied to subsequent User Input or Menu block prompts. For example, if this option is enabled for a voice call and the user interrupts a "Welcome" message by pressing 3, the input is ignored by the next User Input or Menu prompts.

If this option is not enabled, the input is buffered and applied to the next block accepting input.

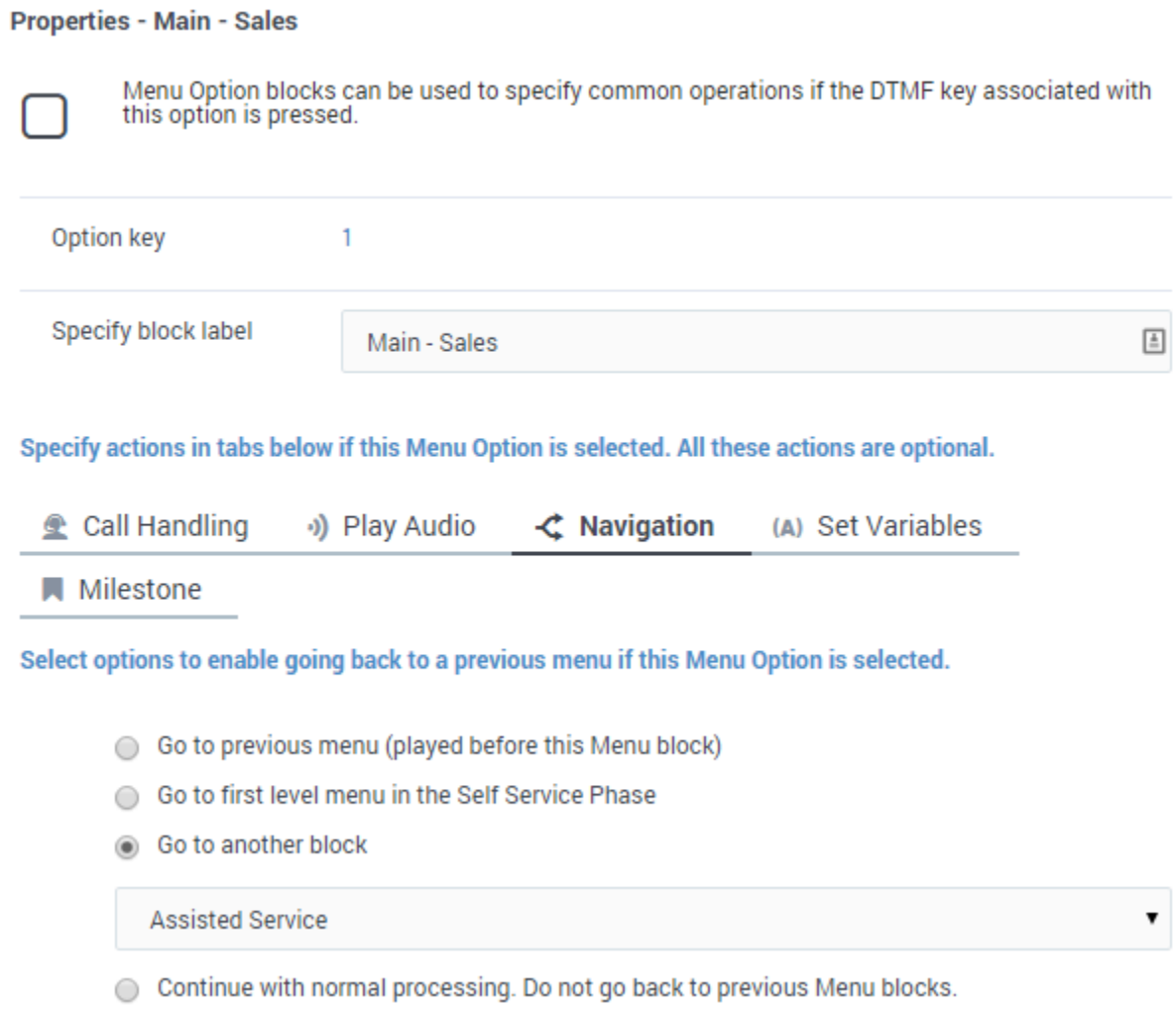
Click **Add Audio Message** to play audio if this specific menu option is selected.



Navigation tab

Select where the application proceeds after this menu option is selected by the user.

> **Tip**
> If there are hierarchical menus in your application, it is a good idea to provide users with an option to go to a previous menu.



Set Variables tab

Assign variables to use when this menu option is selected by the user, without having to add an

Assign Variables block.

Milestone tab

Add a milestone to mark this key moment while the application is running. See the Milestone block page for more information.

Play Message Block

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

You can use the **Play Message** block in the **Self Service** and **Assisted Service** phases to play audio messages to the customer. These messages or prompts might be an introductory welcome message or instructions on how to proceed through the application.

These audio messages are defined as either:

- Text-to-Speech (TTS) — Strings entered directly in the block, or variables.
- Announcements — Audio files that were previously uploaded in the Media Resources page, or variables played as TTS.

Prompts tab

Disable barge-in

Select this option to prevent callers from interrupting a prompt while it is still playing. For example, you might want a "Welcome" message to play all the way through before the caller can enter another command and skip to the next menu prompt.

If this option is not selected, barge-in is enabled, and the prompt can be interrupted by the caller.

> **Important**
> - The selected barge-in setting applies irrespective of whether global DTMF commands are used or not.
> - This option is only supported if the **Play Message** block is used during the Self Service phase. For Assisted Service, you can use the User Input block to control barge-in settings.

Always play prompt and disable buffering

Select this option if you want callers to be able to interrupt a prompt while it is playing, but not have those inputs applied to subsequent User Input or Menu block prompts. For example, if this option is enabled and the caller interrupts a “Welcome” message by pressing 3, the input is ignored by the next User Input or Menu prompts.

If this option is not enabled, the input is buffered and applied to the next block accepting input.

> **Important**
>
> This option is only supported if the **Play Message** block is used during the Self Service phase.

To create a new prompt, click **Add Prompt** and follow the instructions below.

In the **Type** column, select the type of prompt:

- **TTS** — Read a text or variable value to a user through TTS.
- **Announcement** — Play a prerecorded announcement. When using a variable, the variable value should be the name of the audio resource to play.
- **Intelligent Prompt** — Intelligently convert a number into an item such as an alphabetical letter, date (day, month), or currency and then read it with human audio to a customer.

> **Important**
>
> If Designer is not able to play an Intelligent Prompt in the caller's preferred language, it will play the prompt in American English (en-US).

- **Record Utterance** — Play the recorded file captured by the Record Utterance block. (This option is only supported in the Self Service phase, as the recording file captured by the Record Utterance block is no longer available after the Self Service phase.)
- **External Audio** — Play an audio file hosted outside of Designer. For example, you could define a variable that points to an audio file hosted on another server, and specify that variable here.

In the **Variable?** column, enable or disable the check box to identify the **Value** as a variable.

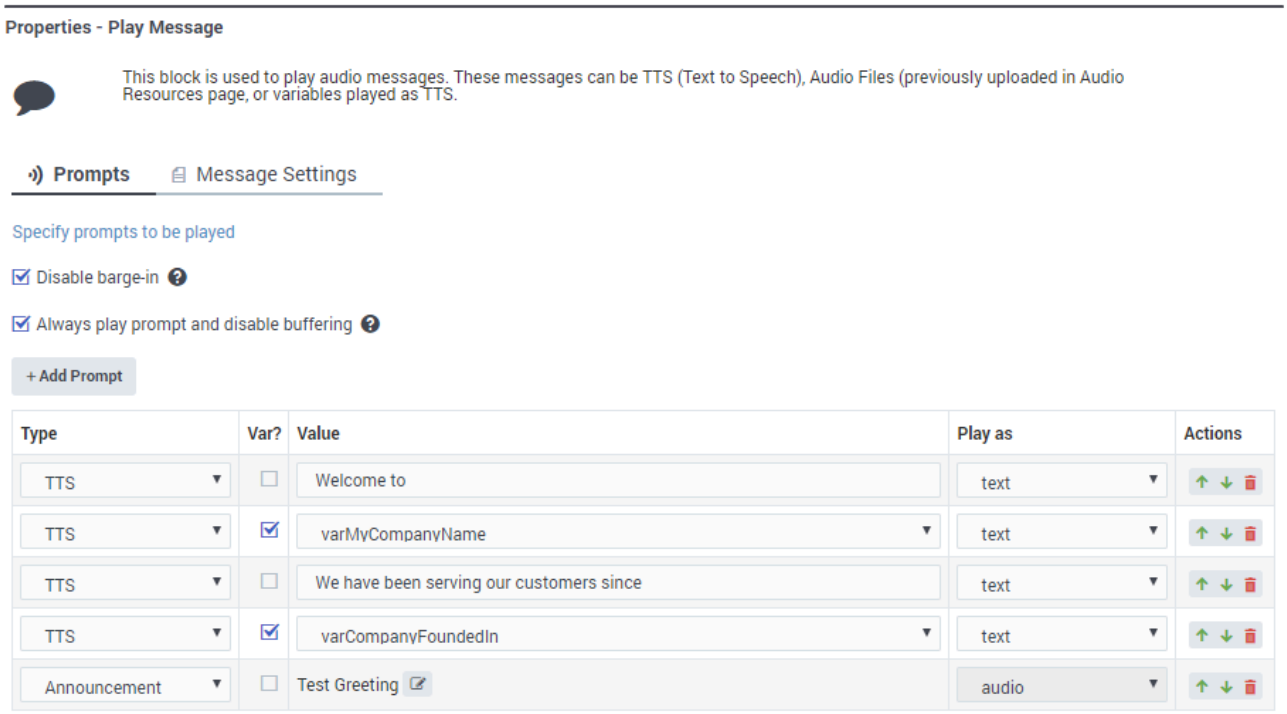
In the **Value** column, specify the prompt value. If **Variable?** is enabled, choose a variable in the drop-down menu.

In the **Play as** column, select an option:

> **Important**

> Some **Play as** options might not be available for certain prompt types.

- **alphanumeric** - The value is read as a series of letters and/or numbers.
- **currency** — Use the following format: `UUUMM.NN`, where `UUU` is the ISO4217 currency code. You can omit the currency code to use the default currency for the current locale.
- **date** — Use the following format: `YYYYMMDD`. You can use `??` or `????` for unspecified fields.
- **day** - A day of the week.
- **dtmf** - A menu item.
- **ordinal** — A positive integer.
- **cardinal** — A positive or negative integer or decimal number.
- **character** - A character.
- **text** - Text that should be read without special formatting (for example, a sentence or phrase).
- **time**
  - TTS prompt - You must use the following format: `hh:mm`. For example, use `09:00` for 9 a.m. or `21:00` for 9 p.m.
  - Intelligent Prompt - You can use the TTS format or the following format: `hhmm[aph?]`, where a is a.m., p is p.m., h identifies 24-hour time, and ? is unspecified. For example, you can use `0900` for 9 a.m. or `0900p` for 9 p.m.
- **telephone** or **phone** — Use a sequence of digits (`0` - `9`), optionally followed by an "x" and then extension digits (`0` - `9`).

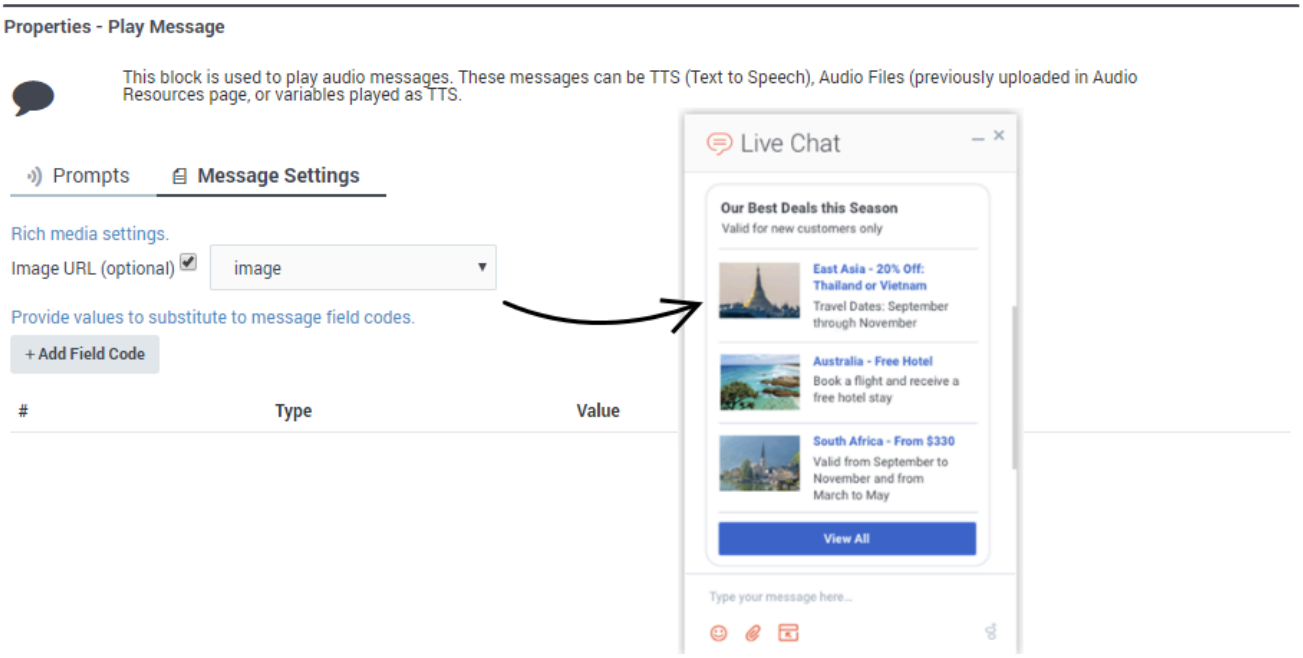


Message Settings tab

Rich media settings

Image URL

(Optional) Enable this option if you want to display an image to customers during chats. Specify the variable that contains the URL to the image that you want to display.



Add Field Code

(Optional) If you are using field codes, you can specify them here.

Scenarios

If you want to repeat the account number that the caller just entered:

1. First Prompt
   - **Type**: TTS
   - **Variable?**: Disabled
   - **Value**: `The account number you just entered is`
   - **Play as**: text
2. Second Prompt
   - **Type**: TTS
   - **Variable?**: Enabled
   - **Value**: account_number_variable
   - **Play as**: telephone

If you want to allow barge-in on a "Welcome" message, followed by an informational prompt for a Menu block input that has barge-in and buffering disabled:

1. In the properties for the **Play Message** block for the "Welcome" message:
   - Do not select **Disable barge-in**.
   - Select **Play prompt and disable buffering**.
2. In the properties for the Menu block that prompts for the caller's input:
   - Select **Disable barge-in**.

Record Block

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

You can use the **Record** block to control a call recording during the **Self Service** phase of an application.

After recording is started, all interactions are recorded until the specified stop setting is reached, the **Self Service** phase ends, or the call is completed.

The **Record** block is useful when you want to

- record the entire IVR call flow but avoid capturing any sensitive or private information (such as a caller's SSN or credit card number) — you could stop recording just before the caller is asked for personal information, and then start it afterward.
- record only a specific part of a call, such as when a customer is asked to confirm or agree to make a payment — you could start recording for the part of the call flow where the customer is asked for this information, and then stop it afterward.

After the recording stops, the platform mixes the input and output audio channels into a single audio file.

Things you should know

Depending on whether your environment uses **Full Call Recording (FCR)** or **Genesys Interaction Recording (GIR)**, the **Record** block might not support all of the features or options described on this page (in most cases, you won't see an option on your screen if it isn't supported).

The Designer instance can only support one type of call recording. If you are not sure which recording type applies to your site, please check with Genesys.

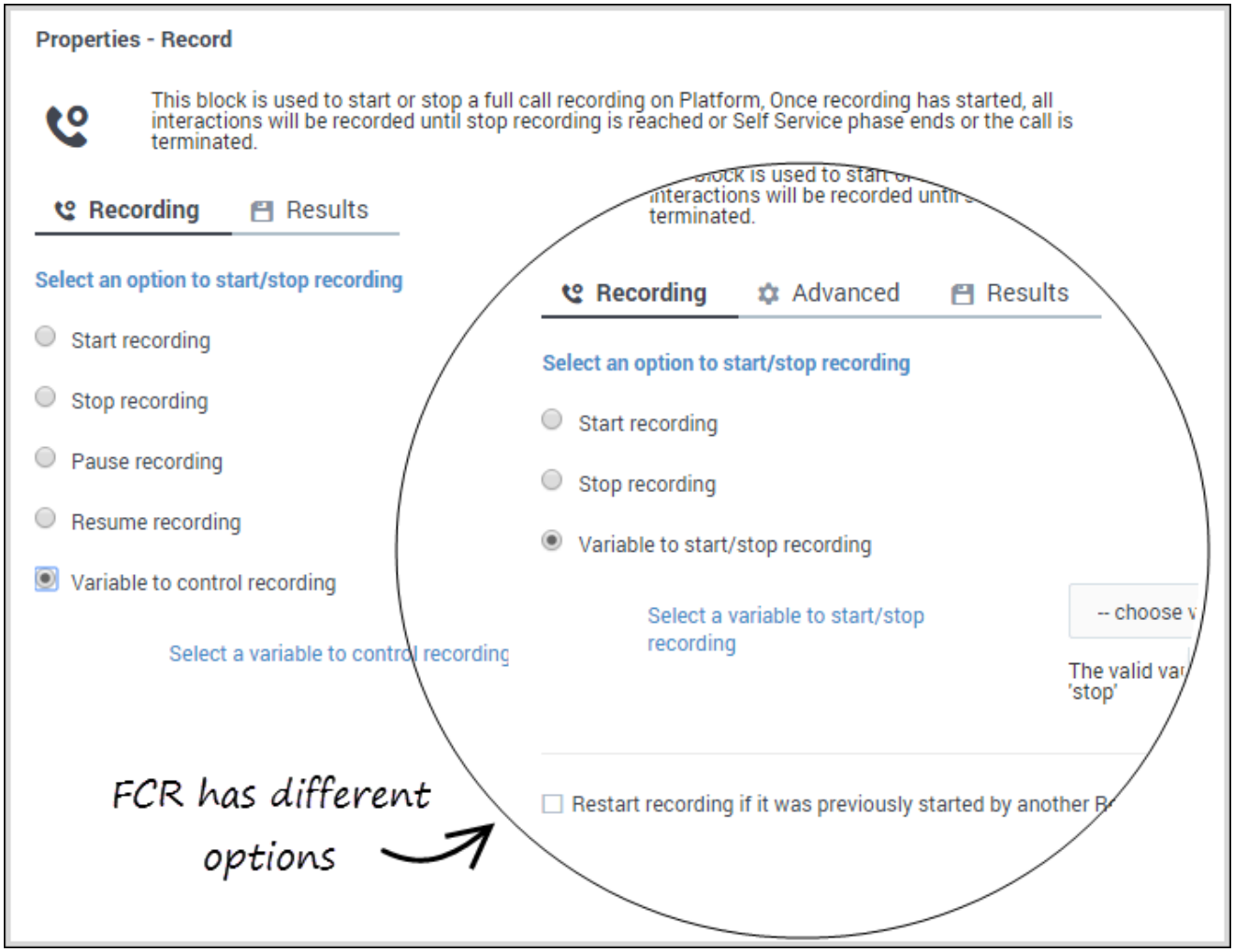
If your recording type has changed, you might also need to review Migrating Applications for information about migrating your existing applications to the new recording type.

> **Important**
> If you are using FCR and have enabled call recording using the *EnableSSRecording*

variable, the application ignores the **Record** blocks.

Recording tab

Select an option to **start** or **stop** the recording, or choose a variable. If your environment uses GIR, you will also have additional options available to **pause** or **resume** the recording.



Select the **Variable to start/stop recording** when you need to start or stop the recording dynamically based on a condition during run-time.

If you are using GIR, this option appears as **Variable to control recording**, and includes additional options to pause or resume the recording.

You would then select a user variable that will be assigned the value of the recording action you

want, depending on the condition experienced during the application flow (such as to *stop* recording if a caller chooses a certain menu option).

The recording is controlled based on the value assigned to the variable. If the variable does not contain a valid value, the application skips the block and continues with the flow.

Select the **Restart recording if it was previously started by another Record block** option if you want to start a new recording file instead of continuing with the recording file that was already started. (This option isn't available for environments using GIR.)

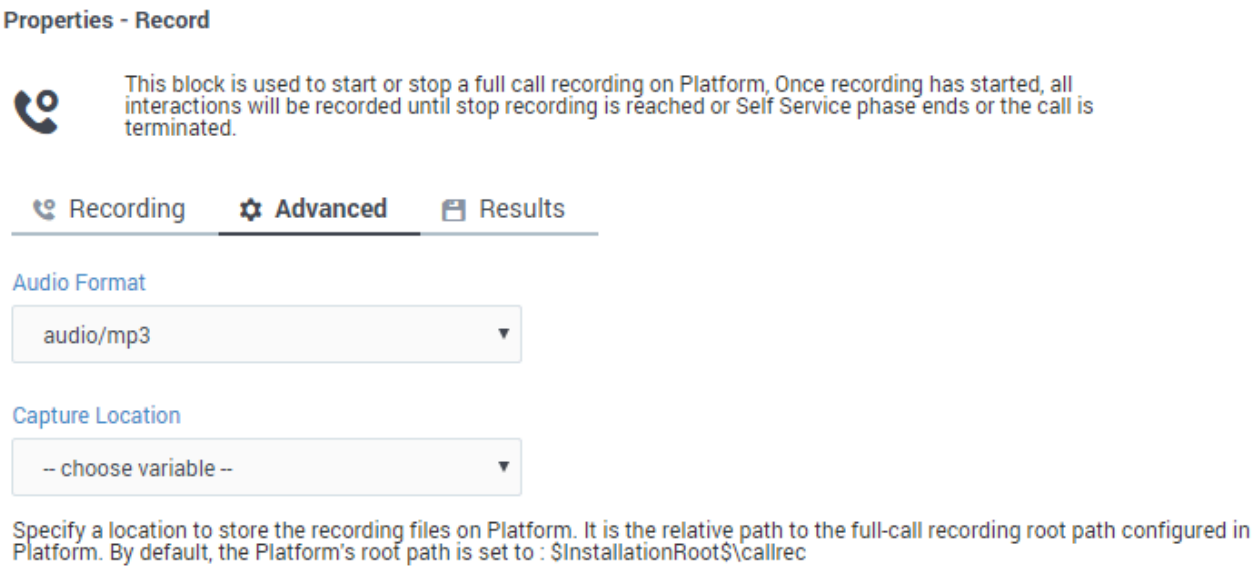
Advanced tab

If you selected a variable to control recording on the **Recording** tab, the **Advanced** tab is available.

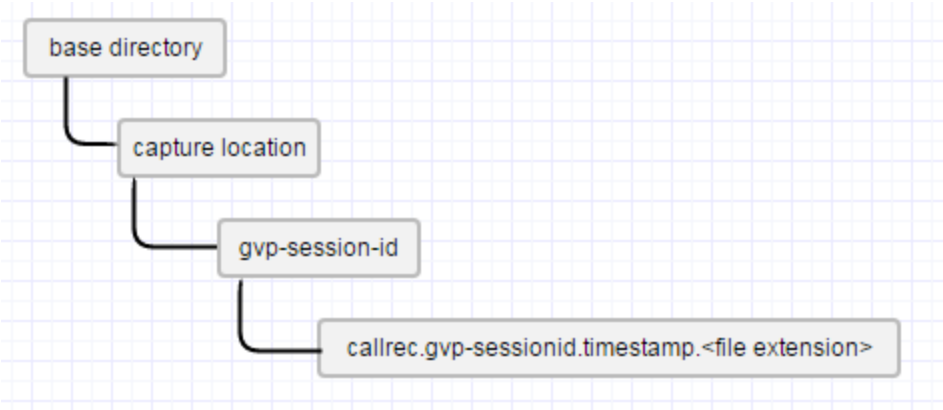
The options shown on this tab are based on whether your site is using a FCR or GIR recording type. Scroll down to the section that applies to your installation.

FCR

Select an **Audio Format** (the default audio format is **mp3**) and a **Capture Location**.



The audio file will be stored on the platform, in a location relative to the base directory (typically, this is `$InstallationRoot$\callrec`). This diagram shows the directory and file-naming structure:



> **Important**
> If the **Capture Location** is not specified, this sub-folder is not created.

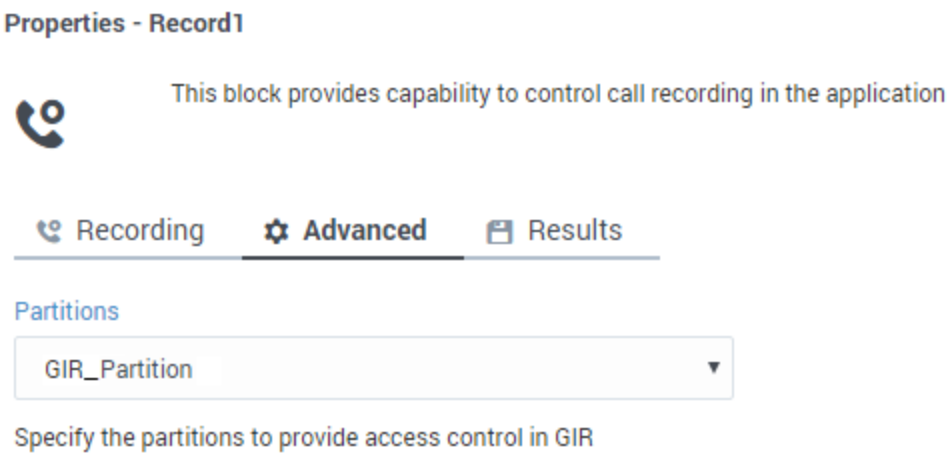
GIR

GIR uses *partitions* for controlling access to media files, such as call recordings.

Designer applications will tag each recording with the partitions specified in this block so that GIR receives both the recordings and their associated partitions. GIR will then use them to control access to recordings. Only those users who belong to the partition will have access to the recordings tagged with that partition.

The partition for the overall application is set by the *DefaultPartition* system variable during the initialization phase. However, if you want this block to use a different partition for GIR access control, you can select one from the **Partitions** list.

This will override the default partition setting, but only for this block.

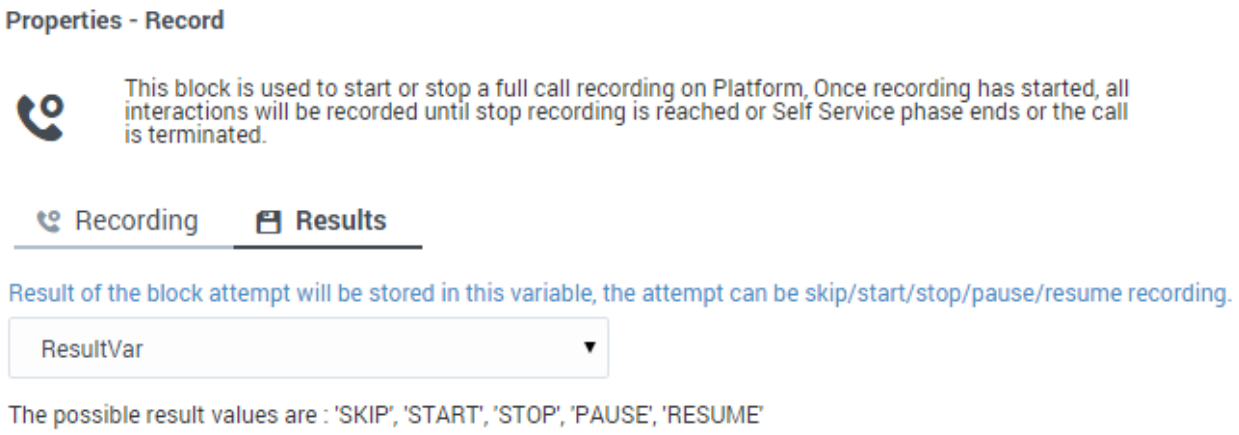


Results tab

Select a variable that will store the result of the block attempt.

If you are using FCR as a call recording type, the result indicates whether the block has attempted to **start**, **stop**, or **skip** recording.

If you are using GIR as a call recording type, the result can also indicate an attempt to **pause** or **resume** recording.



Updating applications to use a new recording type

The **Record** block can use either FCR or GIR for call recordings. GIR offers more capabilities, such as **pause** and **resume**, that are not offered by FCR.

If the recording type is changed in your environment (this change can only be done by Genesys), the previous recording type will continue to be in effect until you publish your applications. You might also need to adjust your applications to the new set of capabilities, as follows:

Moving from FCR to GIR

The existing FCR-based **Record** blocks will continue to work with GIR without changes. To use the additional capabilities offered by GIR, change your blocks to select the new options.

When you are finished making changes, publish the updated applications.

Moving from GIR to FCR

FCR offers fewer options, so you will no longer have access to **pause** or **resume** recording options. Therefore, **Record** blocks that specify these actions will no longer work and must be changed to use either **start** or **stop**.

When you are finished making changes, publish the updated applications.

> ### Important
> The **Advanced** tab will only show those options that are applicable to the current recording type.

Record Utterance block

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

Use this block to capture a voice recording of the caller. You can then use the HTTP REST block to send the recording to an external API, or play it back using the Play Message block.

This block can only be used in the **Self Service** phase of an application. After the Self Service phase is completed, the recording is no longer available.

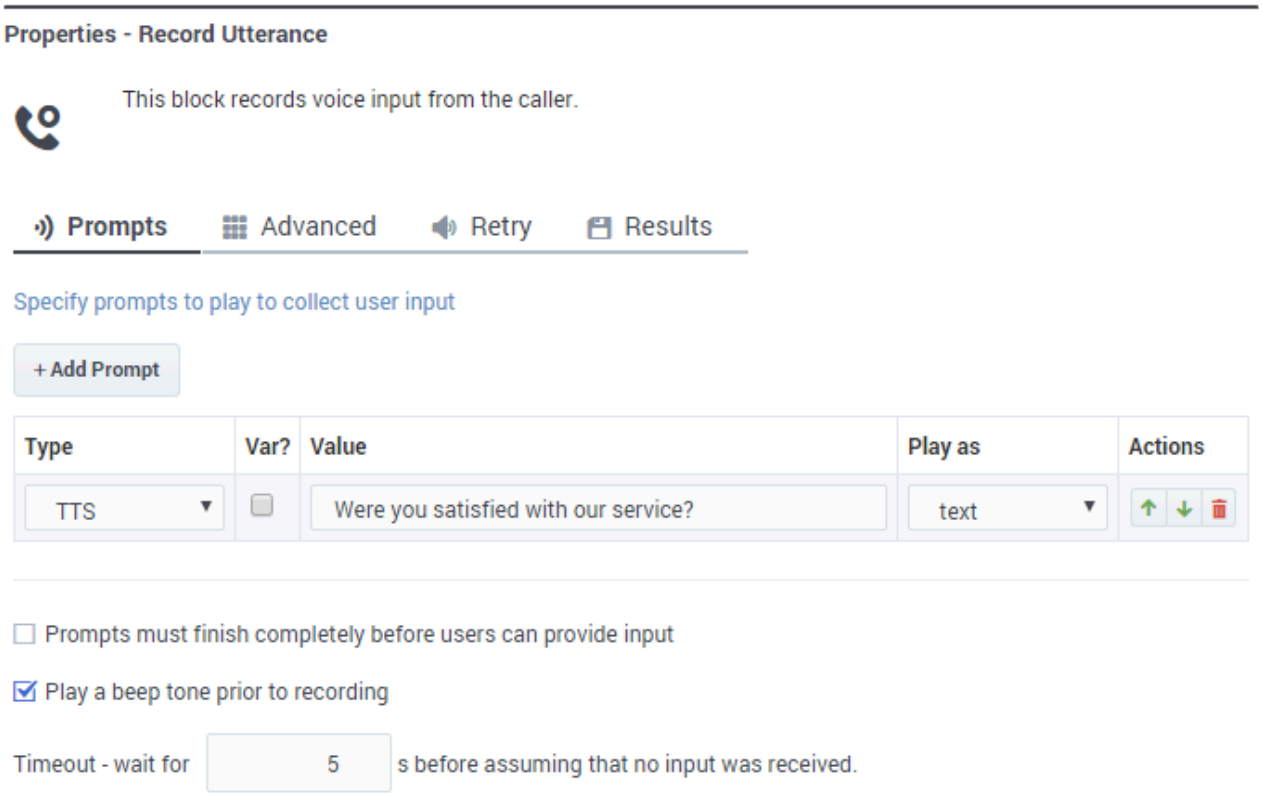
Prompts tab

Click **Add Prompt** to specify the prompts that will be played to the caller.

Select **Prompts must finish completely before users can provide input** to prevent users from responding to the prompt before it has finished.

Select **Play a beep tone prior to recording** to indicate that recording is about to begin.

You can also specify a timeout value to indicate how long Designer should wait for the user to provide a voice input before moving to the next block.



Advanced tab

Use the settings on this tab to specify the recording duration and timeout settings.

Maximum recording duration

Specify the maximum duration (in seconds) of the recording. The default maximum recording duration value is 10 seconds.

Minimum recording duration

Specify the minimum duration (in milliseconds) of the recording. The default minimum recording duration value is 250 ms (milliseconds).

End of recording timeout

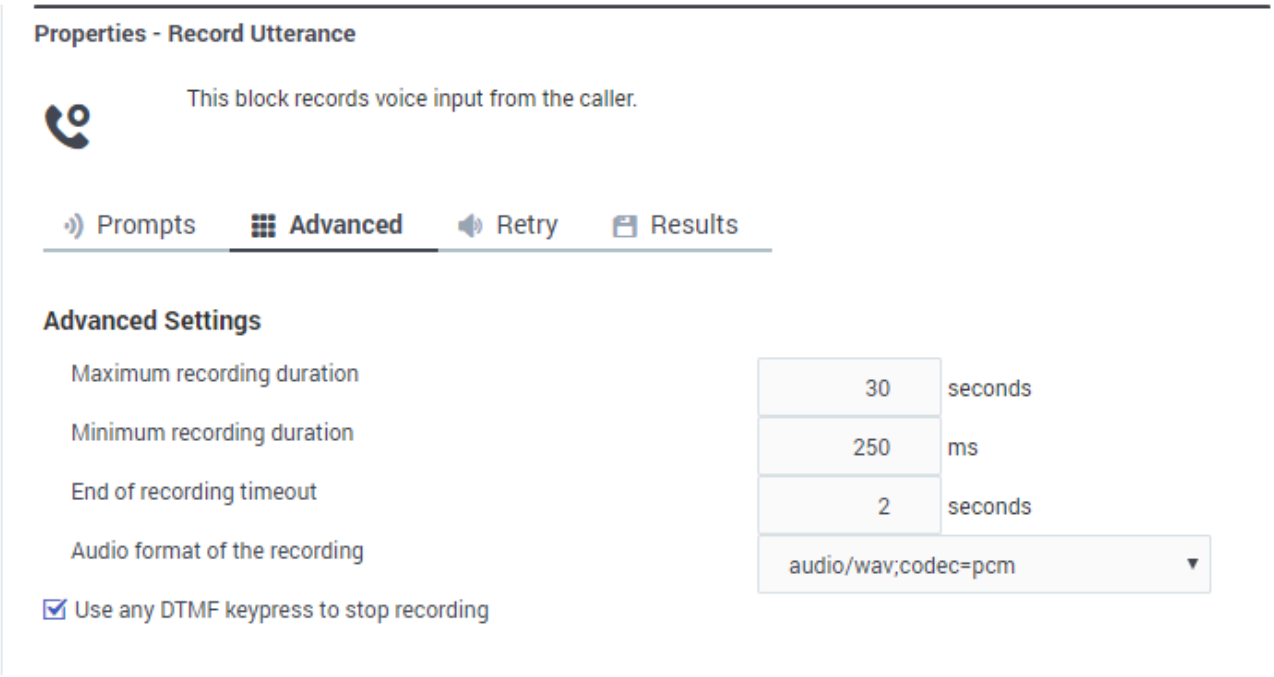
Specify how many seconds of silence should elapse before recording stops. The default recording timeout value is 2 seconds.

Use any DTMF keypress to stop recording

Enabling this option allows users to stop the recording with any key press.

Audio format of the recording

Select the audio format to use for the recording file. The default audio format is **audio/wav;codec=pcm**.



Retry tab

Enable **Use application-wide retry** to use the default retry settings specified in your application settings.

If you disable **Use application-wide retry**, you can enable **Allow Retries** to use the standard input retries if no input was detected during recording.

Results tab

Specify the variables that will store the recording and its details.

User Input Block

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

You can use the **User Input** block in the **Self Service** phase to collect information from the user, such as an account number or credit-card information, and store it in a variable for processing. In the **Assisted Service** phase, you can use this block to gather more information from the user.

Optionally, you can specify whether retries are allowed if the input is not recognized, and whether to play a retry message, along with the original prompt message.

> **Tip**
> If the user enters invalid information or no input, then the value of the results variable is undefined. This must be considered before any later block can process the result. For example, a **Segmentation** block could determine whether the results variable stores a valid value and, based on the result, branches to different paths.

Prompts tab

Disable barge-in

Select this option to prevent callers from interrupting a prompt while it is still playing. For example, you might want a "Welcome" message to play all the way through before the caller can enter another command and skip to the next menu prompt.

If this option is not selected, barge-in is enabled, and the prompt can be interrupted by the caller.

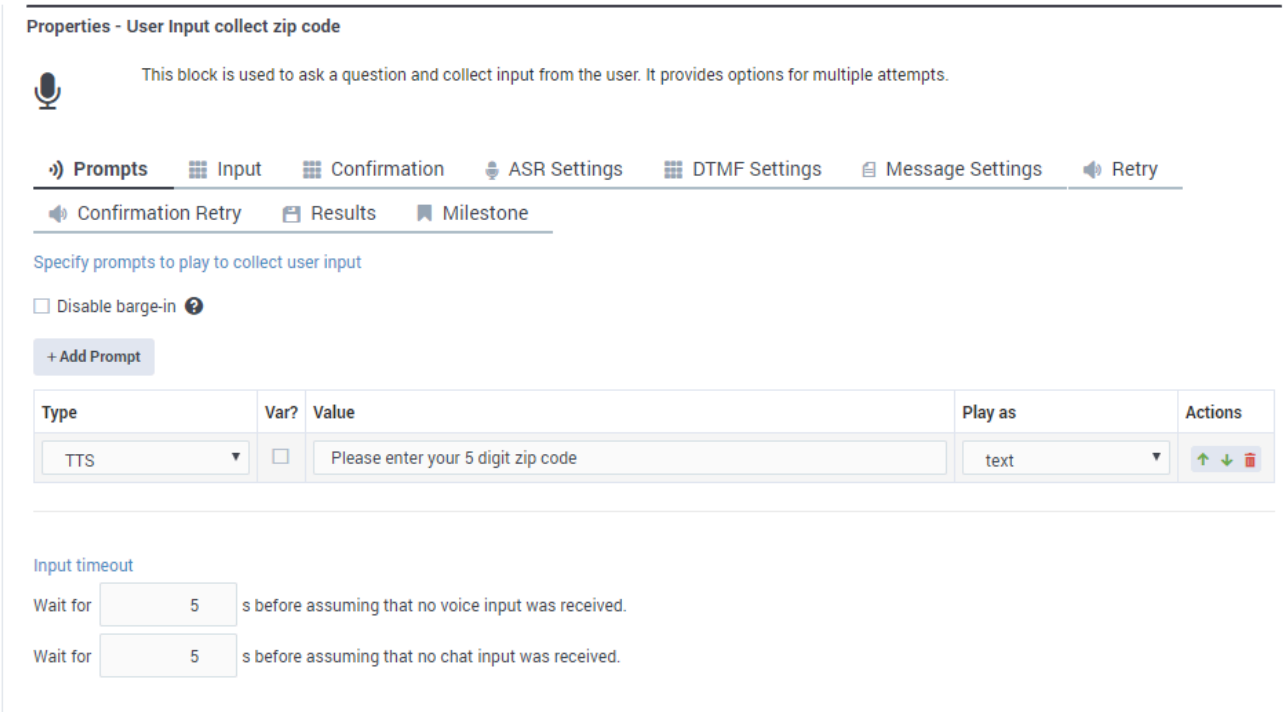
> **Important**
> The selected barge-in setting applies irrespective of whether global DTMF commands are used or not.

Click **Add Prompt** to play prompts when the menu starts.

Input timeout

Set the timeout period, in seconds, to wait before assuming that no voice or chat input was received from the customer. Refer to the **Retry** tab to specify which actions are taken if the timeout period is reached. If retries are not permitted and the timeout period is reached, the application moves onto the next block.

Example



Input tab

Choose one of the following options:

Built-in Grammar

Select this option to use a built-in grammar. You can select from the following types:

- boolean
- currency
- date
- digits
- number
- phone

- time

If you select **digits**, you can also set the following options:

- **Minimum number of input digits** — Specify the minimum number of digits that the caller must enter.
- **Maximum number of input digits** — Specify the maximum number of digits that the caller can enter.

Next, specify the input mode for the grammar. You can select **DTMF**, **Speech**, or both. (If you select only **Speech** mode, **DTMF** grammars remain active but are not matched.)

Languages for built-in grammars can be managed using the *AppLanguageName* system variable (see System variables) or the Change Language block.

External Grammars

Select this option if you have created your own speech grammar or if you want to use a grammar that is hosted outside of Designer.

To add a grammar to use with this block, click **Add Grammar** and specify its settings:

- **Var?** - Enable this check box to indicate that the selected speech grammar will be determined by a variable.
- **Type** - Select the type of speech grammar you are adding:
  - **Not in Designer** - Indicates that the selected speech grammar is hosted outside of Designer. To select this option, check the **Var** box and use the **Name** setting to specify the variable that holds the location where the grammar is hosted. For more information, see Externally-hosted Grammars.
  - **Dynamic?** - Indicates that the selected speech grammar contains dynamic values that can change over time (for example, an employee directory). For more information, see Dynamic Grammars.
  - **Service?** - If selected, **Var?** is enabled and you can select the variable for the desired service.

> **Important**
>
> A grammar can be either **Dynamic** or a **Service** — it cannot be both at the same time. In addition, if you are planning to use more than one service grammar with the same **User Input** block, ensure that no grammar rule names are repeated across two or more grammars.

- **Name** - Select the speech grammar name (or variable) that you want to add to this block.
- **Mode** - Specifies whether this speech grammar is for **voice** or **DTMF**.
- **Arg** - This setting only applies if you selected **Service** as the speech grammar type. You must specify the variable to be matched against the query string contained within the **Service** variable. **Important:** If you don’t specify an **Arg** variable, you'll get a validation error prompting you to include an argument in the **Service** grammar.
- **Actions** - Click the trash icon to delete the selected grammar.

Transcription (Google ASR)

Select this option to use Google Automatic Speech Recognition (ASR). Speech input provided by the

customer will be transcribed by Google's Speech-to-Text engine.

Confirmation tab

Use the settings on this tab to enable and configure user input confirmation. This allows callers to review and confirm any spoken or DTMF inputs they have provided.

Never

Disable user input confirmation.

Always

Enable user input confirmation.

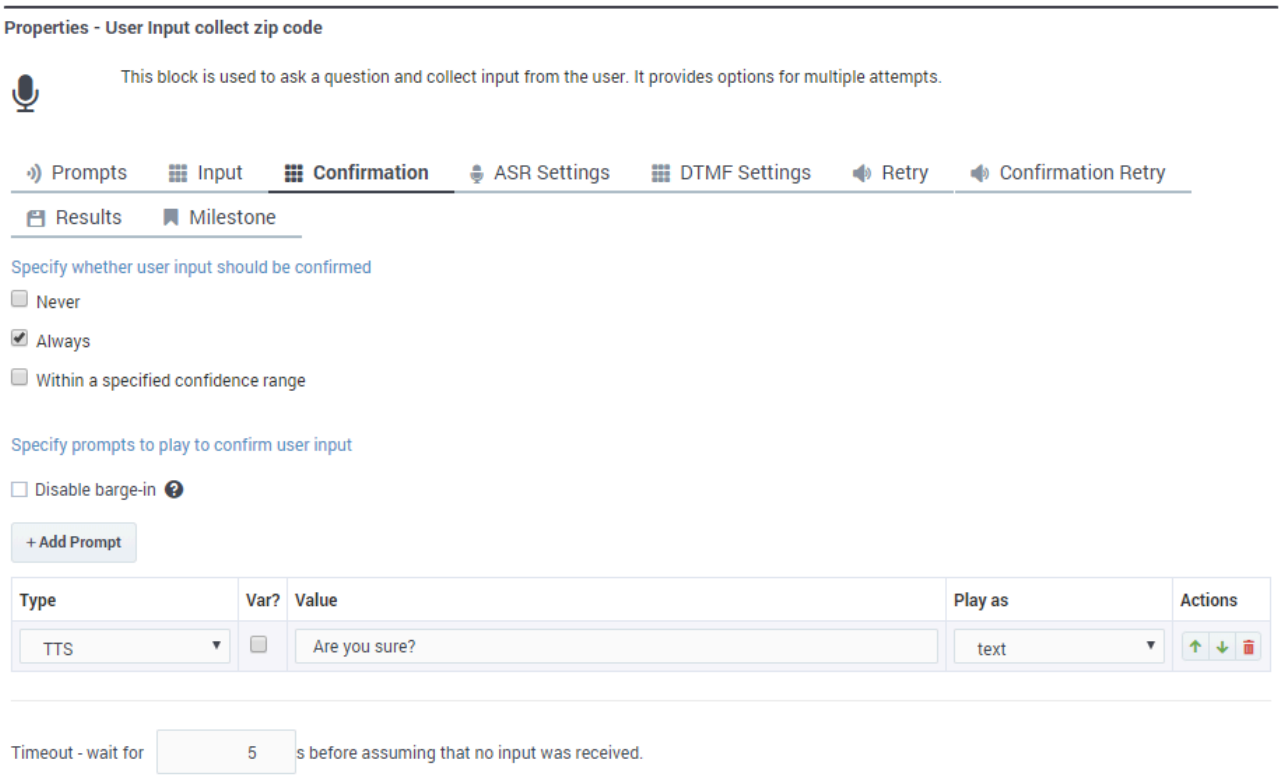
Within a specified confidence range

Enable user input confirmation for callers that meet the defined confidence threshold. If selected, you can then define the lower and upper thresholds.

Specify prompts...

If user input confirmation is enabled, specify the prompt(s) to play. You can also enable or disable barge-in (see the Prompts tab for more information about this setting).

Example



ASR Settings tab

Use application-wide ASR settings

Enable this check box to use the default ASR (Automatic Speech Recognition) settings for your application. You can view or change these settings by clicking **Settings** in the Toolbar.

If you disable the **Use application-wide ASR settings** check box, you can set the following options for this block:

- **Confidence Level** - Specifies the speech recognition confidence level. If the caller's input is below this threshold, the input is determined as **No Match**. A value of `0.0` specifies that minimum confidence is needed for a match. A value of `1.0` specifies that maximum confidence is required before a match is determined.
- **Sensitivity** - Specifies the sensitivity level. A value of `1.0` specifies that speech recognition is highly sensitive to quiet input. A value of `0.0` specifies that speech recognition is least sensitive to noise.
- **Speed vs. Accuracy** - Specifies the balance between how fast the application responds to the input versus how accurate the response is interpreted. A value of `0.0` specifies that quick recognition is preferred. A value of `1.0` specifies that high accuracy is preferred.
- **Complete Timeout** - Specifies the required length of silence, in seconds, following user speech before the application determines a result (match, **No Match**, or **No Input**).

- **Incomplete Timeout** - Specifies the required length of silence, in seconds, following user speech before the application determines a result. This property is used in the following situations:
  - If the speech prior to the silence does not match all active grammars, this property specifies how long to wait before the partial result is rejected as **No Match**.
- If the speech prior to the silence matches an active speech grammar, but it is still permissible to continue speaking and match the speech grammar. By contrast, **Complete Timeout** applies when the speech prior to the silence matches an active speech grammar and no further words are permissible.

- **Max Speech Timeout** - Specifies the maximum amount of time, in seconds, for which speech input is allowed before it is determined to be **No Match**.

> **Tip**
> If you change a setting and you later want to revert the setting to the default value, click **Global**.

- **Max N-best** - Specifies the speech recognition threshold level for N-best. This is an algorithm that enables Designer to formulate a "best guess" as to what it thinks a customer intended to say in response to a prompt, if the input they provided isn't clear.

DTMF Settings tab

Configure the following settings for DTMF input:

- **Input termination character** - Specify a termination character that the caller can enter to mark the end of the input string. Commonly, * or # are used as termination characters. If the caller does not enter a termination character, the application waits until the **Terminating Timeout** period has passed before processing the input.
  Example: You set this value to #. The caller enters 1234#. The input is **1234** and **#** signals that no more characters will be entered.
- **Interdigit Timeout** is the amount of time, in seconds, that the application waits between digit inputs before assuming the end of the input string. If the user entered too few or too many digits, a retry is attempted. If retries are not allowed, the application moves on to the next block.
  Example: You set this value to 3 and specify that the input can be between three and five digits. The caller enters 1234. The application waits **3** seconds before assuming a fifth digit will not be entered.
- **Terminating Timeout** is the amount of time, in seconds, that the application waits for the **Input Termination Character** before processing the input string. If the input is always a static length (for example, four characters), then you can set this value to `0` for the application to immediately process the input after the last digit is entered.
  Example: If this value is **5** and the caller enters 1234, the application waits **5** seconds before processing the input.

Retry tab

Use application-wide retry

Enable this option to use the default retry settings for your application. You can view or change these settings by clicking **Settings** in the Toolbar.

Allow retries

If you disable the **Use application-wide retry** check box, you can enable **Allow retries** to specify retry rules for this block. You can set the following options:

- **Number of No Input retries allowed**
  Select the number of retries to allow for callers whom do not provide input. For each retry, you can specify whether a prompt is played by clicking the corresponding section beneath this field. For example, if you allow two no-input retries and you want to play a prompt after the first retry, select the **No Input #1** line and add a prompt. Enable the **Play original menu prompt after this retry prompt** check box to repeat the menu prompts for the caller.
- **Number of No Match retries allowed**
  Select the number of retries to allow for callers whom do not provide a match for a **Menu Block**. For each retry, you can specify whether a prompt is played by clicking the corresponding section beneath this field. For example, if you allow two no-input retries and you want to play a prompt after the first retry, select the **No Match #1** line and add a prompt. Enable the **Play original menu prompt after this retry prompt** check box to repeat the menu prompts for the caller.
- **After Final No Input**
  Add the prompt to play after the maximum number of permitted No Input retries is reached. If this block is in the Self Service phase, you can also specify a target destination for the application to jump to, such as another block in the Self Service phase or to the Assisted Service or Finalize phase of the application.
- **After Final No Match**
  Add the prompt to play after the maximum number of permitted No Match retries is reached. If this block is in the Self Service phase, you can also specify a target destination for the application to jump to, such as another block in the Self Service phase or to the Assisted Service or Finalize phase of the application.

Confirmation Retry tab

Enable and configure the settings for user input confirmation retries.

Use default confirmation strategy

Select this option to use the default settings.

Allow retries

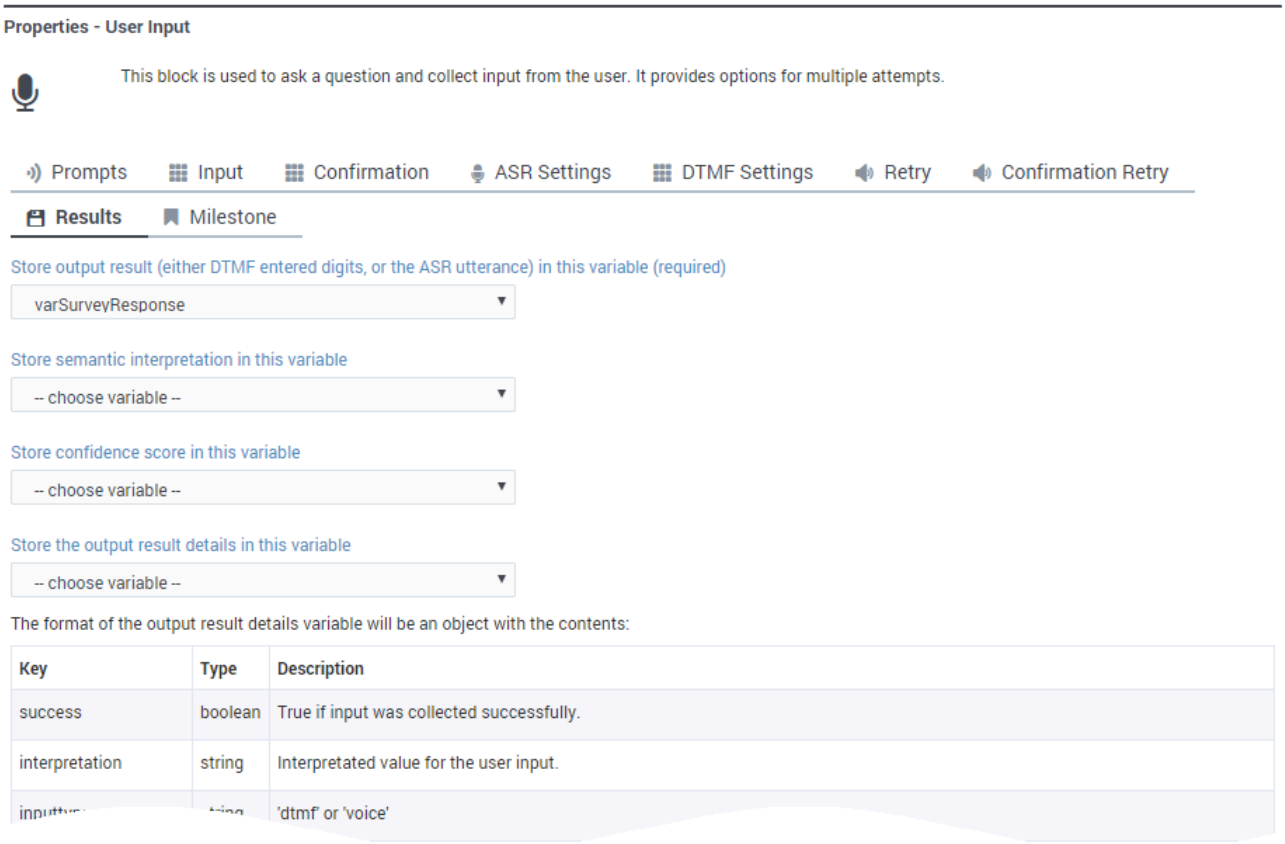
Enable retries. Specify the number of retries that are allowed for **No Match**, **No Input**, and

Disconfirmations.

Results tab

Specify the variables in which to store the results of the interaction, semantic interpretation, confidence score, and output result for speech recognition (such as the **Confidence Level** value).

Example



Milestone tab

Specify a milestone for this block.

Example scenario

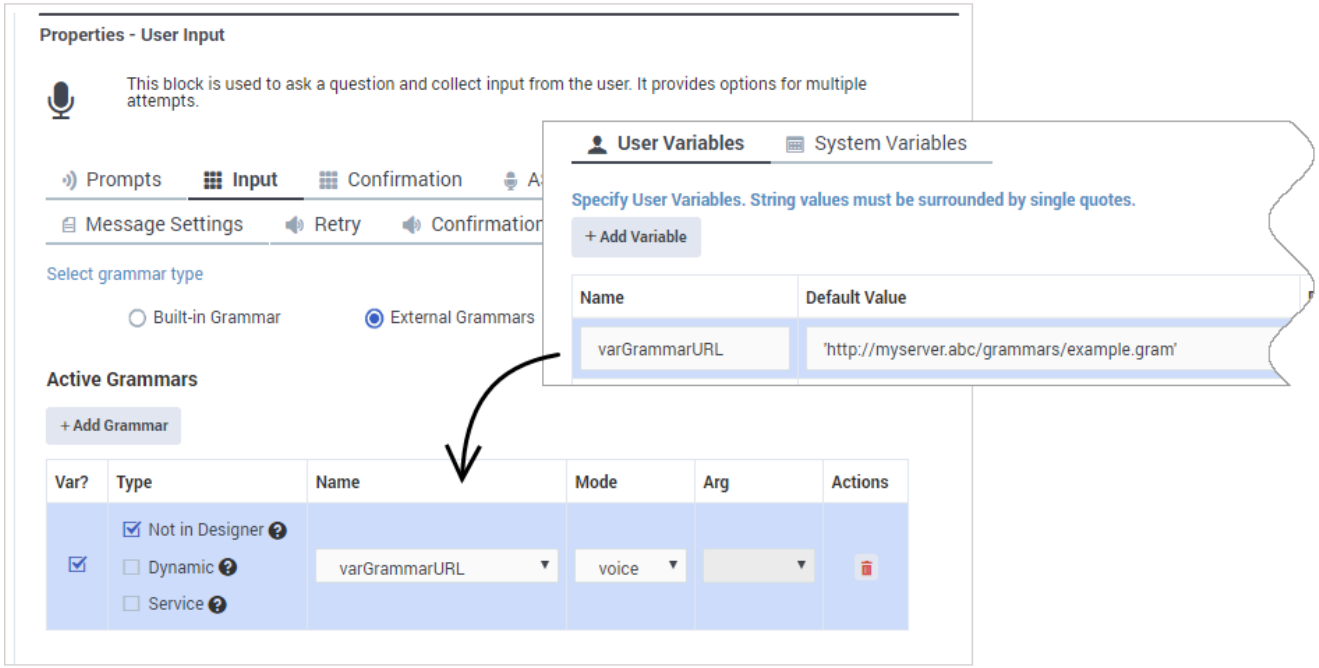
If you want to:

- Collect a single digit and allow for two retries. The caller can start collecting input only after the entire audio prompt has finished playing.
  - **Disable barge-in:** Enabled
  - **Minimum number of input digits**: 1
  - **Maximum number of input digits**: 1
  - **Allow retries**: Enabled
  - **Number of No Input retries allowed**: 2
  - **Number of No Match retries allowed**: 2

Externally-hosted Grammars

Unlike other types of external grammars, these grammars are managed remotely and not uploaded to Designer.

Here's an example of how an external speech grammar hosted outside of Designer might be set up. In this case, the user-defined variable *varGrammarURL* contains the path to the location where the grammar is hosted (e.g. http://myserver.abc/grammars/example.gram):



If you select the **Not in Designer** option, and Designer is unable to locate the grammar (for example, it is missing from the remote server or the path was defined incorrectly), the associated prompt won't be played and the value of the output result variable will be set to *false*. The reason will be captured in the output result details variable as *missing_grammar*.

Applications should always check for the result of this block before processing any of the user input that was collected.

Dynamic Grammars

A dynamic grammar contains an array of values that automatically update in response to external input. Whereas traditional grammars are static and must be updated manually for each change, dynamic grammars are always current and do not require manual updates. This is useful for situations in which the grammar contents change frequently, such as employee or customer lists.

Consider the following example. You have created a voice application with a **User Input** block that allows the caller to contact specific employees in your company. You are using an external grammar that contains the names of all of your employees as valid inputs for this block. However, if an employee leaves the company or your company hires a new employee, the grammar is no longer current and must be updated. A dynamic grammar, however, can use an array of values from a variable and update itself based on external input. You do not need to manually update a dynamic grammar each time there is a change to the list.

There are two ways to create an array for use with dynamic grammars:

- Use an HTTP REST block.
- Use an Assign Variables block.

Using an **HTTP REST** Block

This method uses an **HTTP REST** block to fetch an array from a web service.

In this example, we are using a web service that returns the following JSON data:

```
{
        "success": true,
        "employees": ["John", "Julie", "Mark"]
}
```

Next, in the **Output Parameters** section of the **HTTP REST** block, we can assign **employees** to a User Variable that we previously created, called **varEmployees**.

Finally, in the **User Input** block, we can select **varEmployees** as a dynamic grammar. Each time the application runs, the **HTTP REST** block fetches the employee list and uses its contents as a dynamic grammar for the **User Input** block.

Using an **Assign Variables** Block

This method uses an **Assign Variables** block to push values to an array.

In this example, we have initialized a variable called **varEmployees** with a value of `'[]'` (an empty array).

Next, in the **Assign Variables** block, we use the expression `varEmployees.push('John')` to add an employee, **John**, to the employee list.

You can use multiple **Assign Variables** blocks to add items to the array.

Automated Message Block

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

When a customer sends an email to your business, you can use the **Automated Message** block to send an automated message back to the customer to confirm that their email was received.

Message tab

Specify From Address

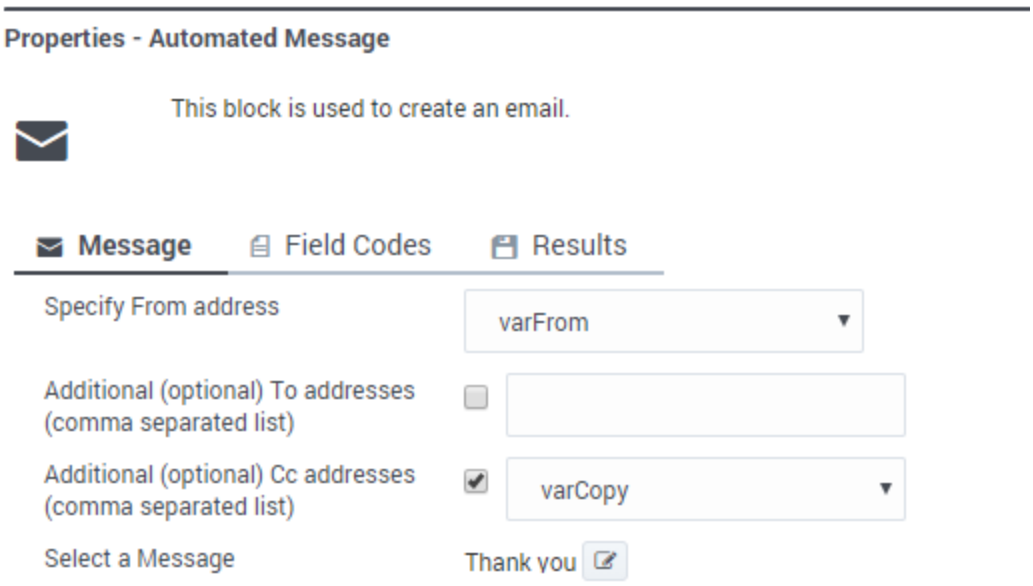
Select the variable that contains the email address to be used as the **From** address.

You can also choose to specify any additional **To** and **CC** email addresses, using variables or comma-separated lists.

Select a Message

Use the picker to browse the **Digital Resources** collection and select the standard message you want to use.

> **Important**
> Custom messages are not supported. You must select one of the standard response messages from the **Digital Resources** collection.



Field Codes tab

(Optional) If you are using field codes, you can specify them here.

Results tab

Specify the variable that will store the results of the email operation.

Chat Message Block

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

You can use the **Chat Message** block to send a chat message to a contact. You can create a custom plain text message, or use one of the standard responses.

This block can also be used as a busy treatment.

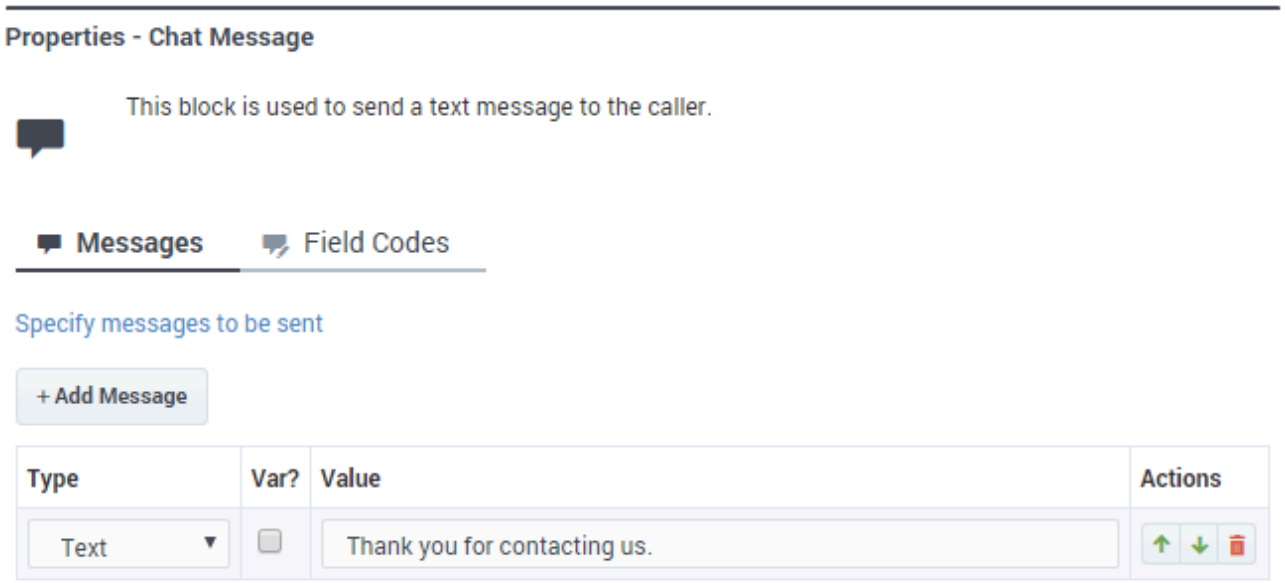
Messages tab

Use the **Messages** tab to add and manage chat messages.

Select **Text** if you are writing a custom text message. Enter the message in the **Value** field.

If you want to specify the value with a variable, select **Var** and choose the appropriate variable.

Select **Message** if you want to use a standard response. Click the "picker" icon to open the Chat Resource Set and select the message you want to use.



Field Codes tab

(Optional) If you are using standard responses, you can use the **Field Codes** tab to specify the field codes being used.

Chat Custom Message Block

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

You can use the **Chat Custom Message** block to send a custom chat message to a contact. You can create a message using plain text, or specify a variable that contains the message you want to send.

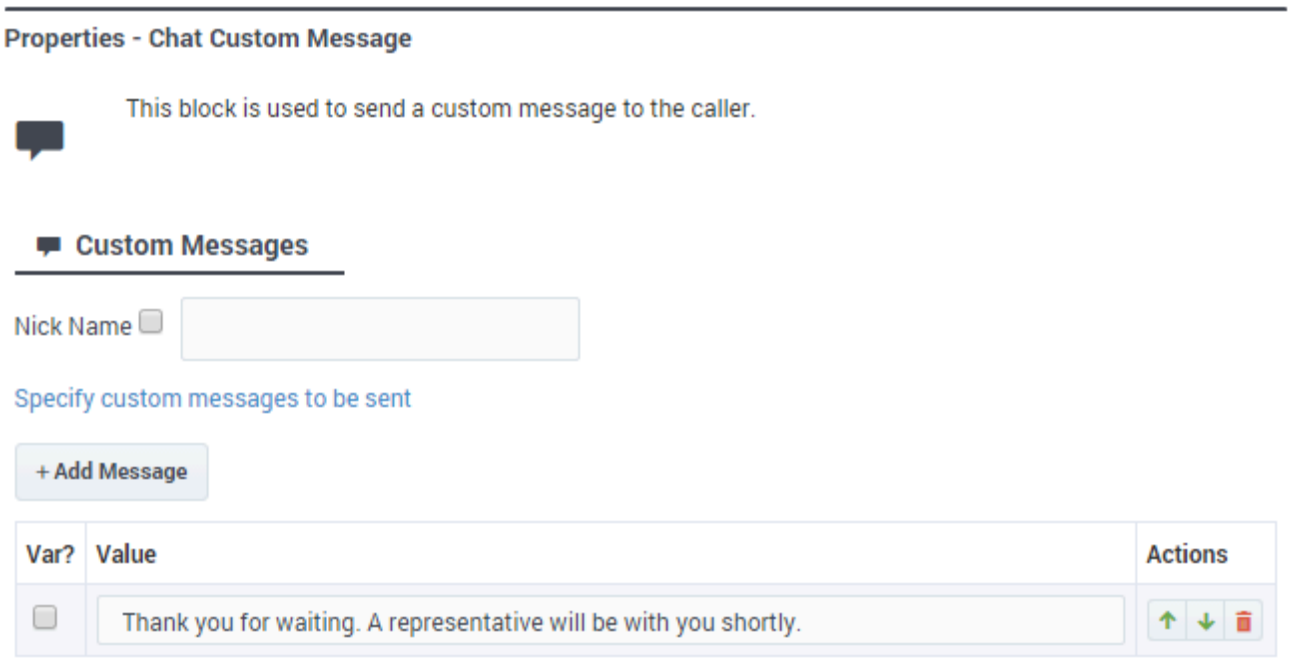
> **Important**
> The **Chat Custom Message** block does not support Standard Responses or Field Codes. It also cannot be used as a busy treatment. This block is only supported for Designer deployments that are using *legacy* chat services (i.e. Chat Version 8.x). If you are not sure if this applies to your deployment, contact your Genesys representative.

Custom Messages tab

Use the **Custom Messages** tab to add and manage chat messages.

Nick Name (optional) is the name that chat contacts will see as being the sender of the custom chat message. You can enter a name, select a variable, or leave this field blank.

Click **Add Message** to enter the value of the message. If you want to specify the value with a variable, select **Var** and choose the appropriate variable.


Properties - Chat Custom Message
This block is used to send a custom message to the caller.
Custom Messages
Nick Name
Specify custom messages to be sent
+ Add Message
Var?
Value
Actions
Thank you for waiting. A representative will be with you shortly.

Send Chat Transcript Block

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

The **Send Chat Transcript** block lets you send a transcript of the chat to the email address specified in the contact's profile.

When used within the initial application flow, the transcript message is sent right away. Note that if routing is not yet completed, this transcript will not include any messages or conversations that take place between the agent and the contact after that point.

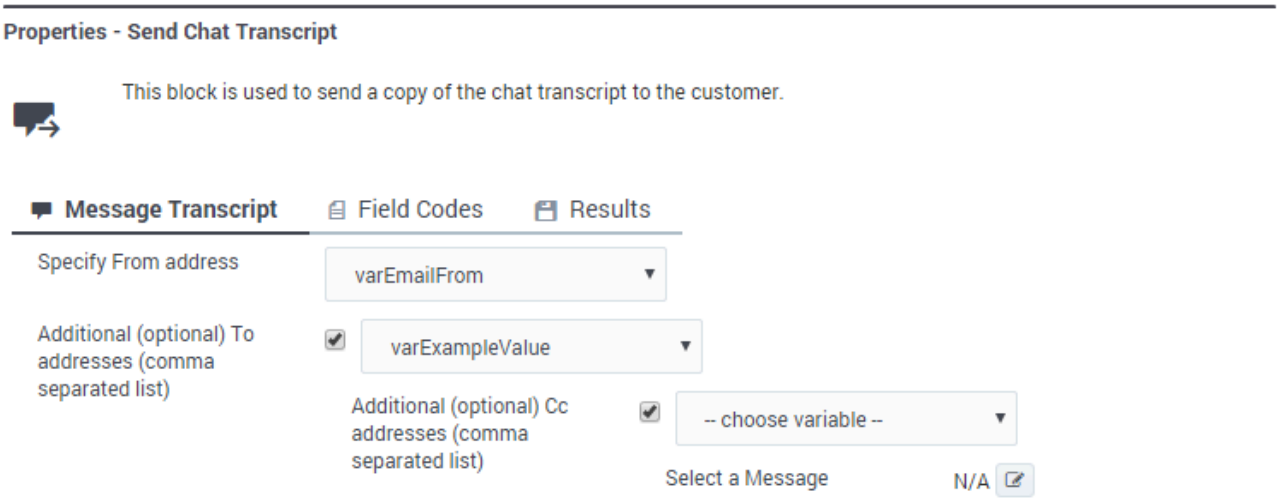
If you want all messages or conversations that take place between the agent and the contact to be included in the transcript, add this block to a post-processing application that you have specified in the **Advanced** tab of the Route block.

Message Transcript tab

Use the **Message Transcript** tab to select the variable that contains the email address you want to use as the *From* address in the emailed transcript.

(Optional) If you are sending the transcript to multiple recipients, you can select the variables containing a comma-separated list of email addresses to be added as additional recipients or copied on the email.

For **Select a Message**, click the "picker" icon to open the Chat Resource Set and select a standard response message to include with the transcript.



Field Codes tab

(Optional) If you are using standard responses, you can use the **Field Codes** tab to specify the field codes being used.

Results tab

Select the variable in which to store the results of the email operation.

Get Chat Transcript Block

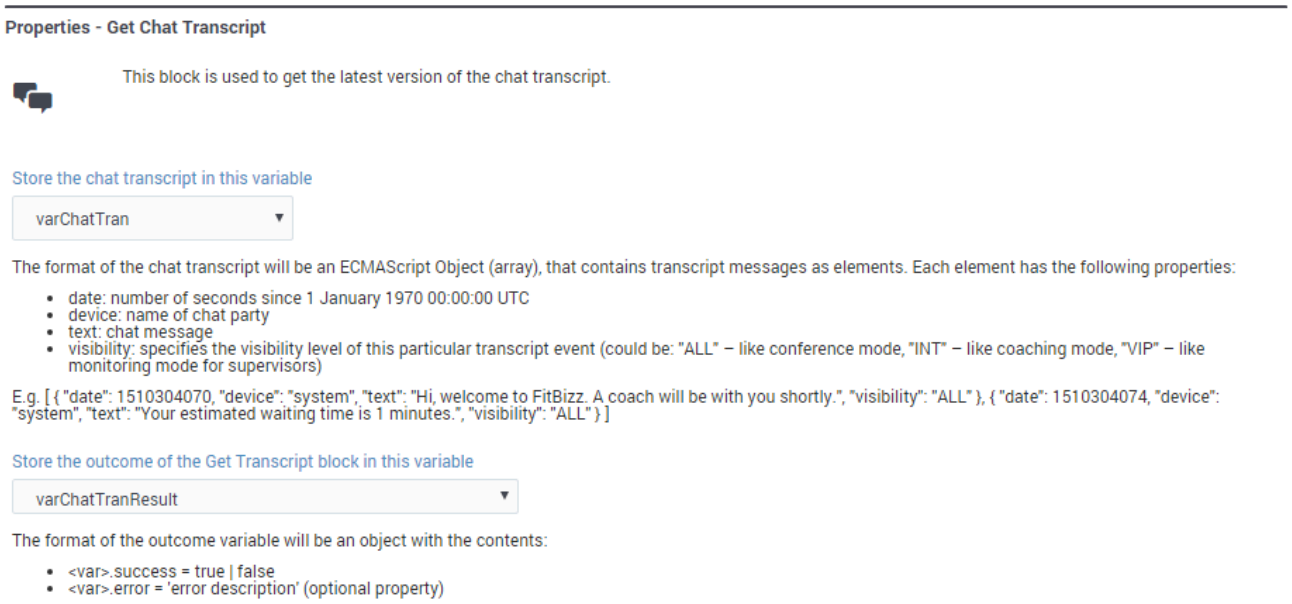
> **Important**
>
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

The **Get Chat Transcript** block enables you to store the contents of the latest chat transcript in a variable which can then be referenced at a later point in the application flow. For example, you might want to retrieve the chat transcript and send it to multiple email recipients.

Using this block

This block can be used in the **Assisted Service** and **Finalize** phases.

Select a variable to store the chat transcript and a variable to store the result of the **Get Chat Transcript** request.



Send Email Block

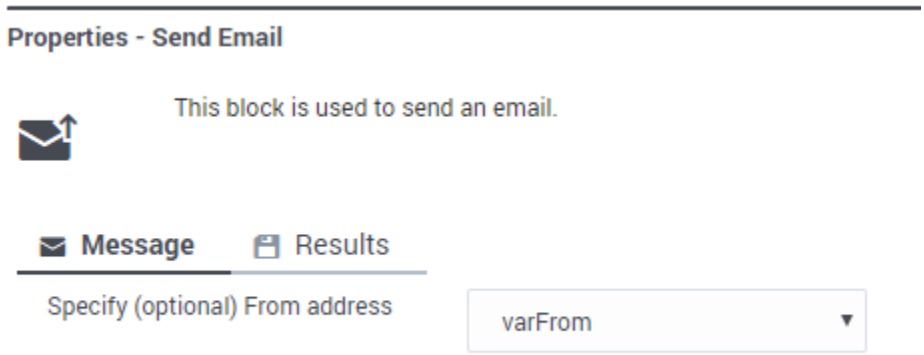
> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

The **Send Email** block lets an agent send an email message to the customer.

Message tab

Specify From Address

Select the variable containing the email address to be used as the **From** address.



Results tab

Specify the variable in which to store the results of the email operation.

Send SMS block

> **Important**
>
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

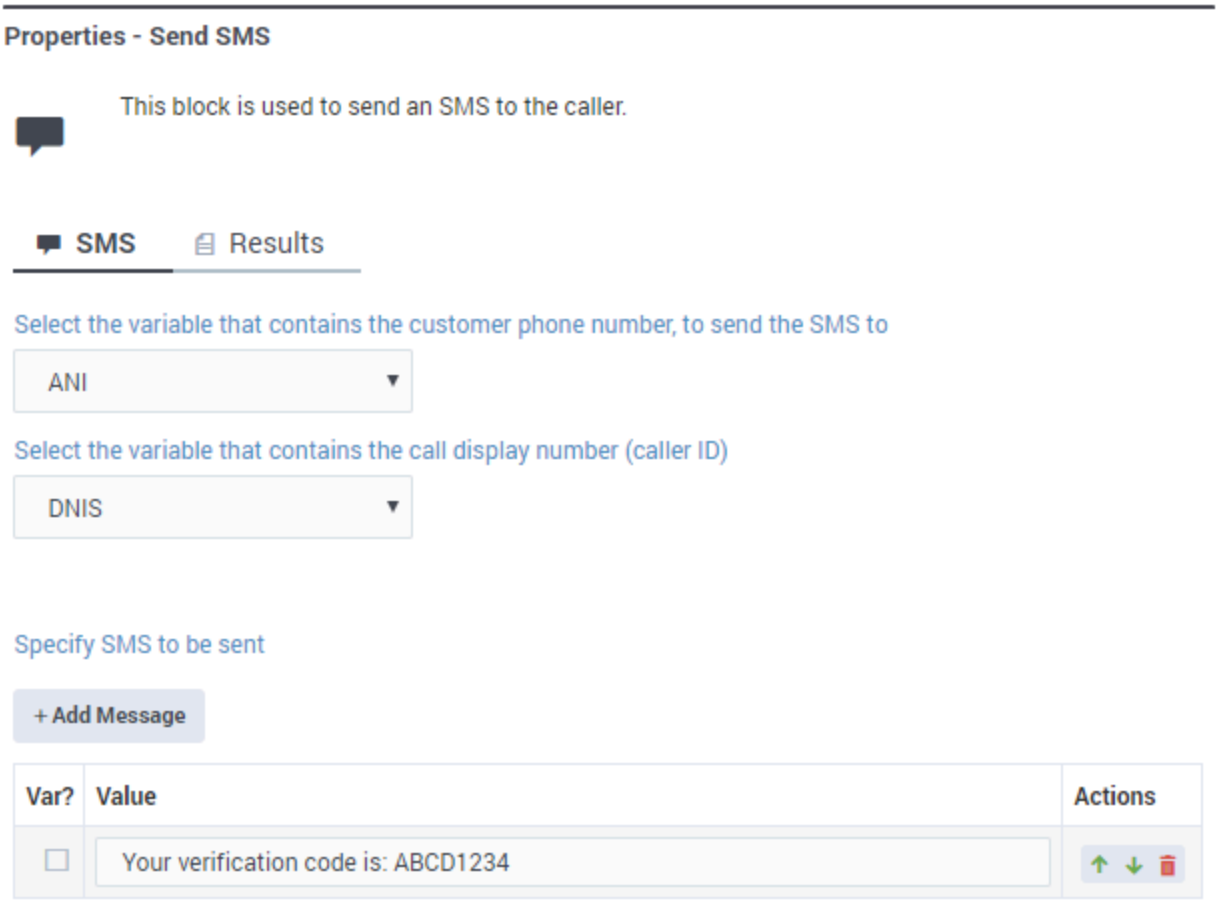
> **Important**
>
> To use this block, your site must be enabled for Digital Channels. For more information, contact your Genesys representative.

The **Send SMS** block lets you send an SMS (Short Message Service) message to a customer.

SMS tab

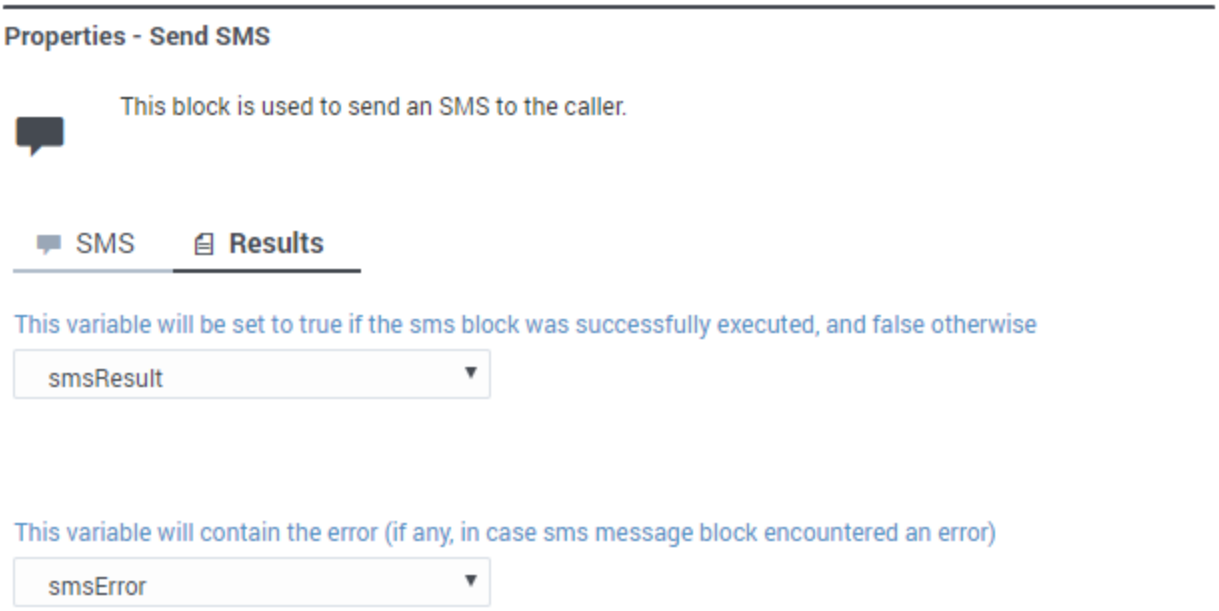
Use the settings on this tab to specify the variables that hold the values for the customer's phone number (i.e. ANI) and the call display number, or caller ID (i.e. DNIS).

Click **Add Message** to specify the SMS to be sent to the customer (or the variable that contains the message).



Results tab

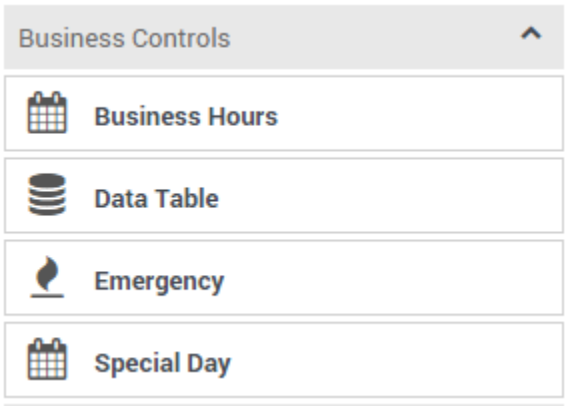
Use the settings on this tab to specify the variables that will hold the results of the **Send SMS** block.


Properties - Send SMS
This block is used to send an SMS to the caller.
SMS
Results
This variable will be set to true if the sms block was successfully executed, and false otherwise
smsResult
This variable will contain the error (if any, in case sms message block encountered an error)
smsError

Business Controls Blocks

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

The blocks in this category are used to control various operational aspects of your business, such as setting up your hours of operation, emergency flags, data tables, special days, and so on.



Business Hours

Announce when your business is closed.

Used in: **Initialize**, **Self Service**, **Assisted Service**

Data Tables

Read values from a data table.

Used in: **Initialize**, **Self Service**, **Assisted Service**, **Finalize**

Emergency

Add a conditional emergency option to your application.

Used in: **Initialize**, **Self Service**, **Assisted Service**

Special Day

Define holidays and other special days.

Used in: **Initialize**, **Self Service**, **Assisted Service**

Business Hours Block

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

You can use the **Business Hours** block in the **Initialize**, **Self Service**, or **Assisted Service** phase to inform customers that your business is closed. You can also choose to end the interaction at this point.

> **Tip**
> The hours that you define in the **Business Hours** block are based on the application's time zone setting. To set the application time zone, select the **Initialize** phase and open the **System Variables** tab. Click the drop-down menu in the **Timezone** row and select a value. You can override the default time zone setting by selecting a variable in the **Set Timezone** section of the **Business Hours** tab.

Business Hours tab

To set your business hours, select each **Day** you are open and specify the **Start Time** and **End Time** for each day.

Open All Day

Select if your business is open for that entire day.

No End Time

Select if there is no set end time for the given day (this option only appears if the business is not closed for that day and the **Open All Day** option is not enabled).

Terminate the call if it is outside Business Hours

Select to end calls that come in outside of business hours.

Use Business Hours defined in Business Controls

If you prefer to use a specific shared business hours resource that you've defined on the Business Controls page, enable this option and select it from the list of defined business hours resources.

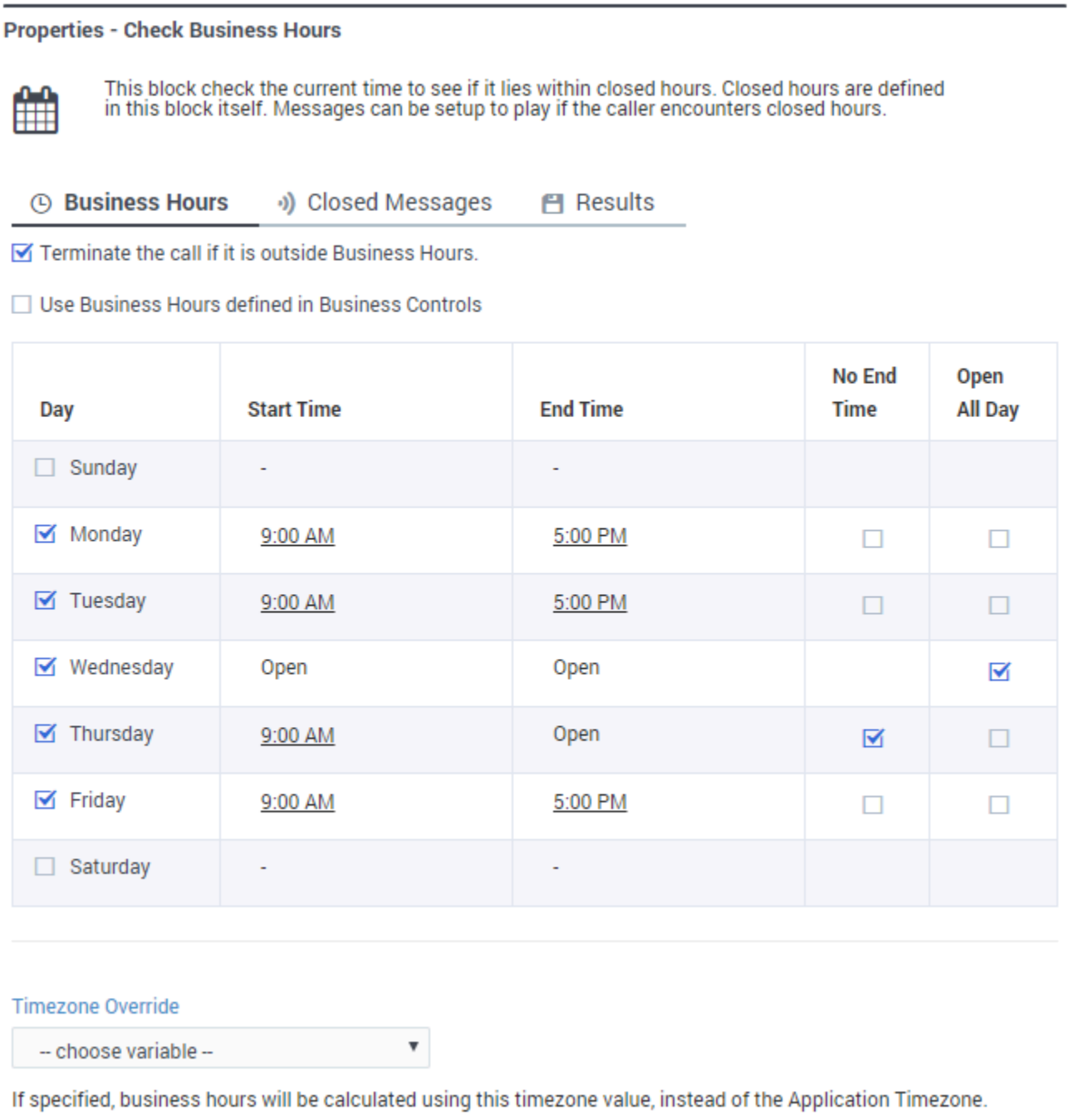
Alternatively, you can specify a variable to be used dynamically while the application runs. Select **Variable?** to specify a user-defined variable that holds the name of a Business Hours resource. If this resource will be read from a data table, you must also select **From Data Table?** to indicate that the variable holds the result of a data table lookup. Otherwise, the result will not be evaluated correctly.

> **Important**
>
> - If the Business Hours are being determined dynamically at runtime, you can't mix user-defined variables (for example, `varDepartmentName + "_PrimaryHours"`) with variables retrieved from data table lookups. Make sure you check the appropriate box to indicate the type of variable being used.
> - No matter which method you use, the name stored in the variable must match one of the Business Hour objects you created on the Business Controls page.

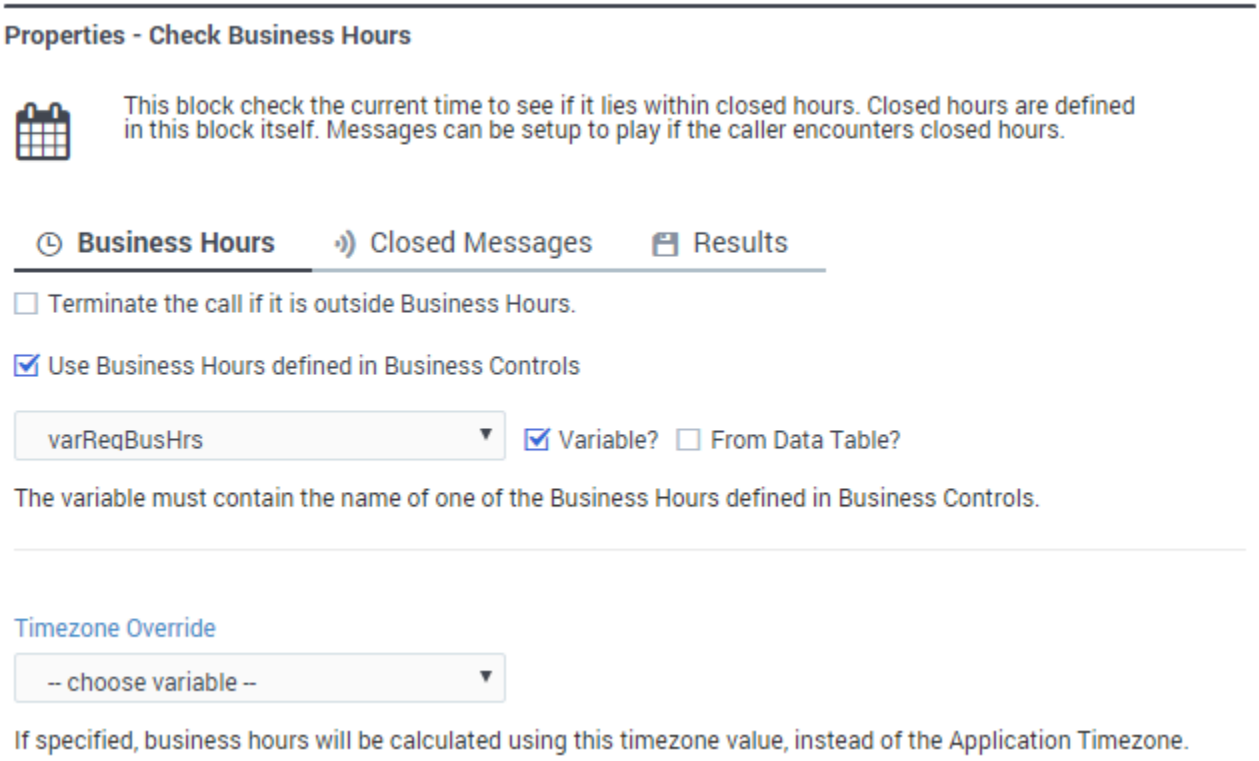
Timezone Override

Select a variable that will override the time zone setting for the application.

Example 1


Properties - Check Business Hours
This block check the current time to see if it lies within closed hours. Closed hours are defined in this block itself. Messages can be setup to play if the caller encounters closed hours.
Business Hours
Closed Messages
Results
Terminate the call if it is outside Business Hours.
Use Business Hours defined in Business Controls
Day
Start Time
End Time
No End Time
Open All Day
Sunday - -
Monday 9:00 AM 5:00 PM
Tuesday 9:00 AM 5:00 PM
Wednesday Open Open
Thursday 9:00 AM Open
Friday 9:00 AM 5:00 PM
Saturday - -
Timezone Override
-- choose variable --
If specified, business hours will be calculated using this timezone value, instead of the Application Timezone.

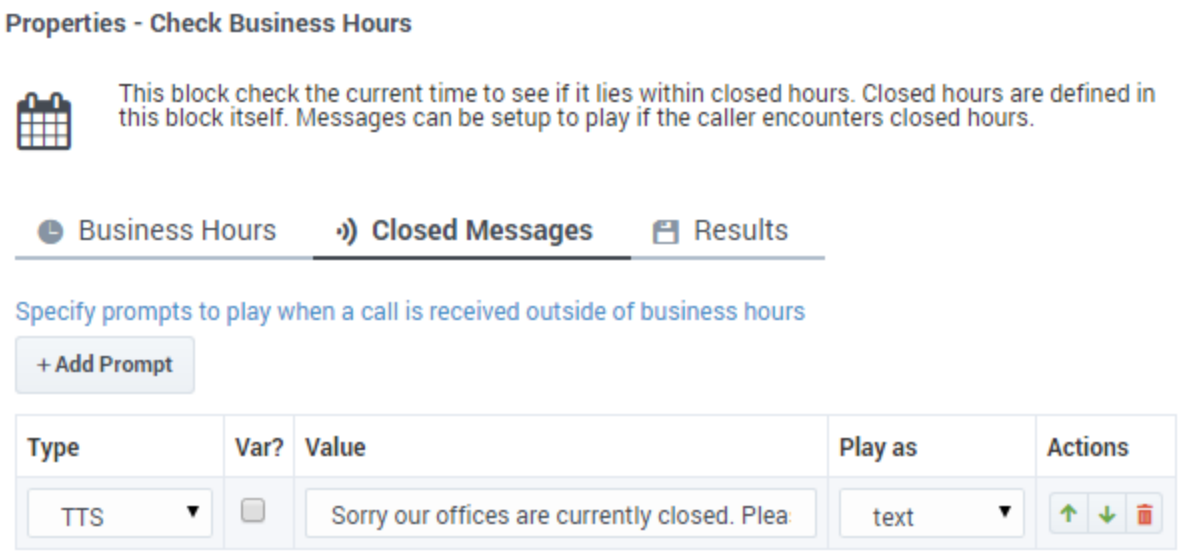
Example 2



Closed Messages tab

In the **Closed Messages** tab, you can select the messages to play or send to customers when your business is closed. If your application is a Default application type, you can specify the prompts to play for voice or chat customers. For Digital application types, you can select an acknowledgement message from the Digital Resources Collection that will be emailed to customers.

> **Tip**
> See the Play Message block page for information on how to use prompts.



Results tab

You can assign a variable to the **Set the result of Block operation status in this variable** property to store the result of the block operation check. If specified, the variable is assigned the Boolean value of **true** or **false**, to indicate if the block operation completed without errors.

You can assign a variable to the **Store the result of Business hours check in this variable** property if you need to use the result of the check later in application. If specified, the variable is assigned the Boolean value of **true** or **false**.

You can assign a variable to the **Store the number of minutes before the business is opened** to track how long until the business is open, if it is currently closed. If the business is open, this variable is set to 0. The maximum value returned is the number of minutes until the end of week (Sunday to Saturday). (This feature is supported only for Business Hours defined in Business Controls.)

You can assign a variable to the **Store the number of minutes before the business is closed** to track how long until the business is closed, if it is currently open. If the business is closed, this variable is set to 0. (This feature is supported only for Business Hours defined in Business Controls.)

You can assign a variable to the **Store the business hours schedule in this variable** property if you need to read the business hours schedule later in the application. If specified, the variable is assigned a JSON object. The JSON object structure is:

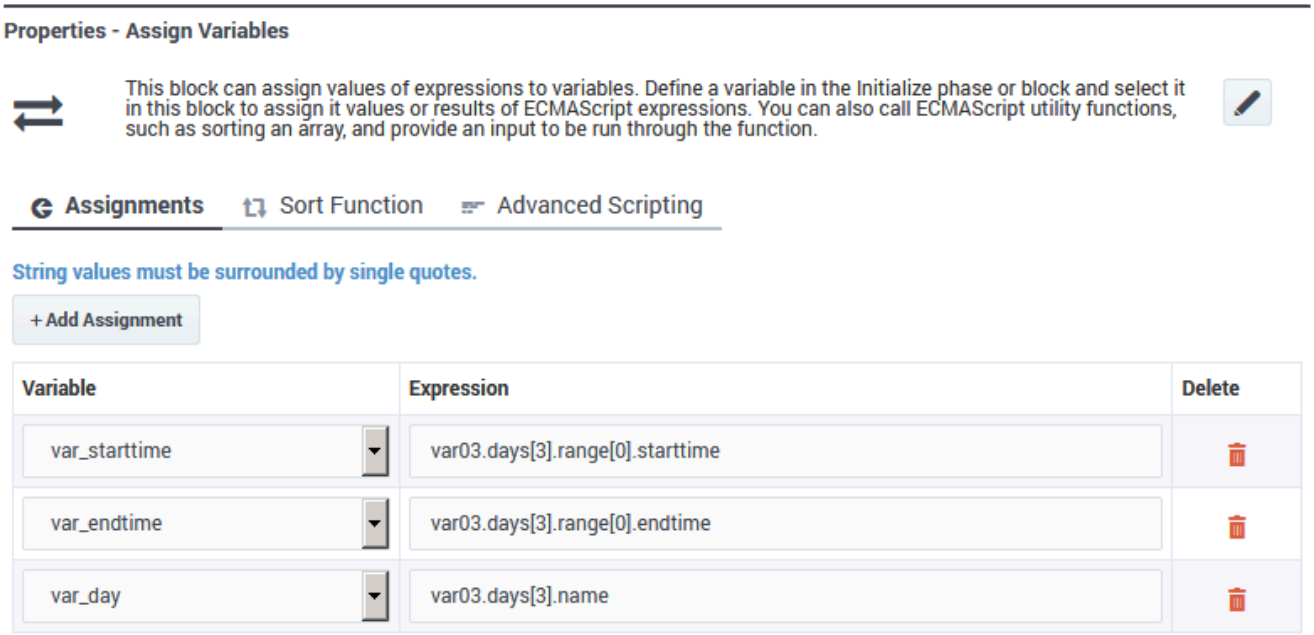
```
{
    "days": [
        {"name": "Sunday","closed": true,"range": [{"starttime": "","endtime": ""}]},
        {"name": "Monday","closed": false,"range": [{"starttime": "1000","endtime": "1800"}]},
        {"name": "Tuesday","closed": false,"range": [{"starttime": "1000","endtime":
"1800"}]},
        {"name": "Wednesday","closed": false,"range": [{"starttime": "1000","endtime":
"1800"}]},
        {"name": "Thursday","closed": false,"range": [{"starttime": "1000","endtime":
```

```
"1800"}]},
        {"name": "Friday","closed": false,"range": [{"starttime": "1000","endtime": "1800"}]},
        {"name": "Saturday","closed": false,"range": [{"starttime": "1100","endtime":
"1700"}]}
    ]
}
```

This JSON object can be used in expressions for the **Assign Variables** block and prompt values for the **Play Message** block.

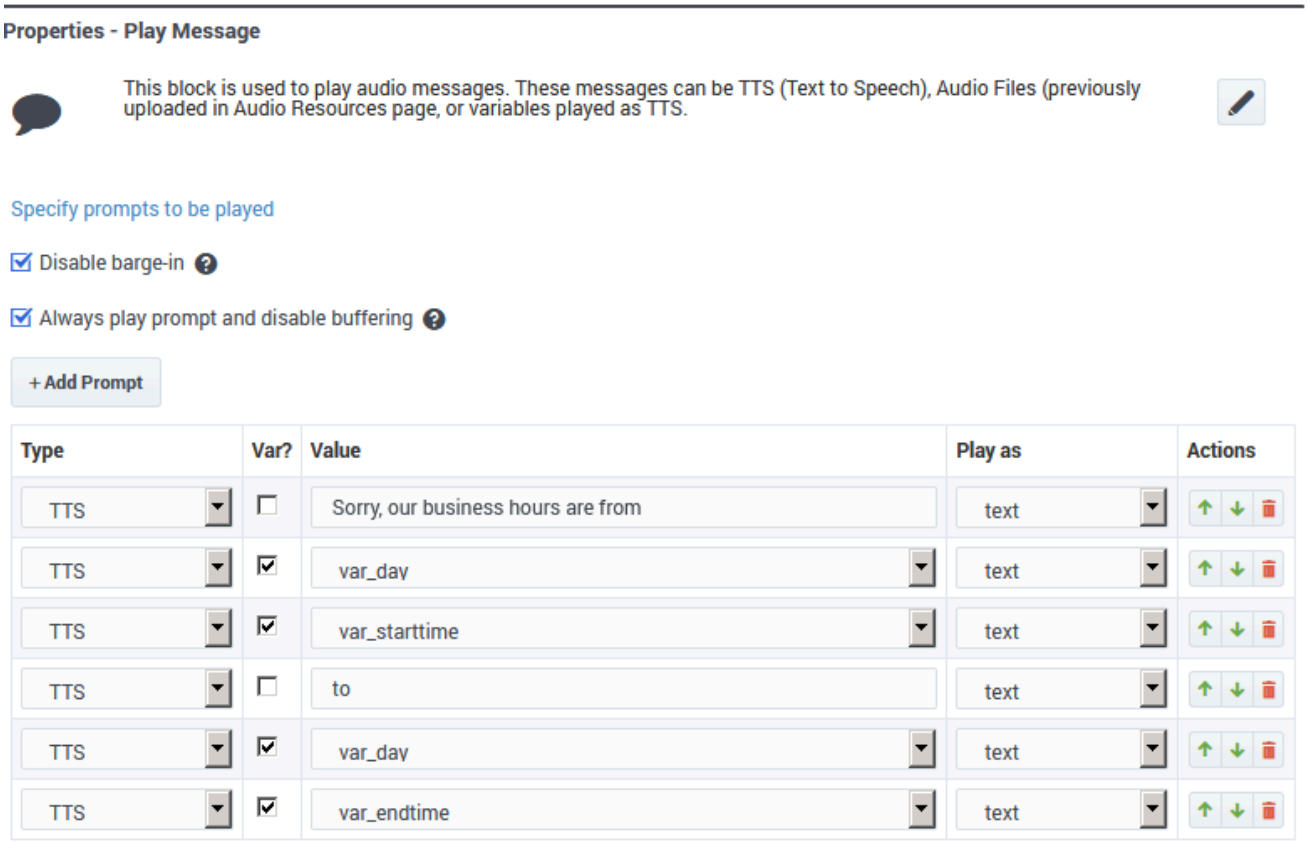
Using Business Hour Values in Prompts

You can also use the **Business Hours** block to announce business hours in prompts. For example, you could use an Assign Variables block to assign the following variables:



Then, set up the prompt to announce the **End Time** value for Wednesday:

- In the **Play Message** block, click **Add Prompt**.
- Choose **TTS** as the **Type**.
- Enable the **Var?** checkbox, and select **var_endtime** as the **Value**.



> **Important**
>
> The `days[n]` range is from from `0` to `6`, with `0` representing Sunday and `6` representing Saturday.

Scenarios

If you want to:

- Specify that the business is closed after 4 p.m. on Thursdays
  - Set the **End Time** value on Thursday to 4:00 PM
- Play a message if a customer calls after business hours, and then end the call
  - On the **Business Hours** tab, enable the **Terminate the call if it is outside Business Hours** checkbox.
  - On the **Closed Messages** tab, create a prompt.

- Specify business hours for a different timezone than the one the application is running in
  - In the **Intialize** phase, assign a value to the **timezone** variable (remember that string values must be surrounded by single quotes — for example, 'UTC').
  - On the **Business Hours** tab, select the **timezone** variable from the **Set Timezone** section.

Data Tables Block

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

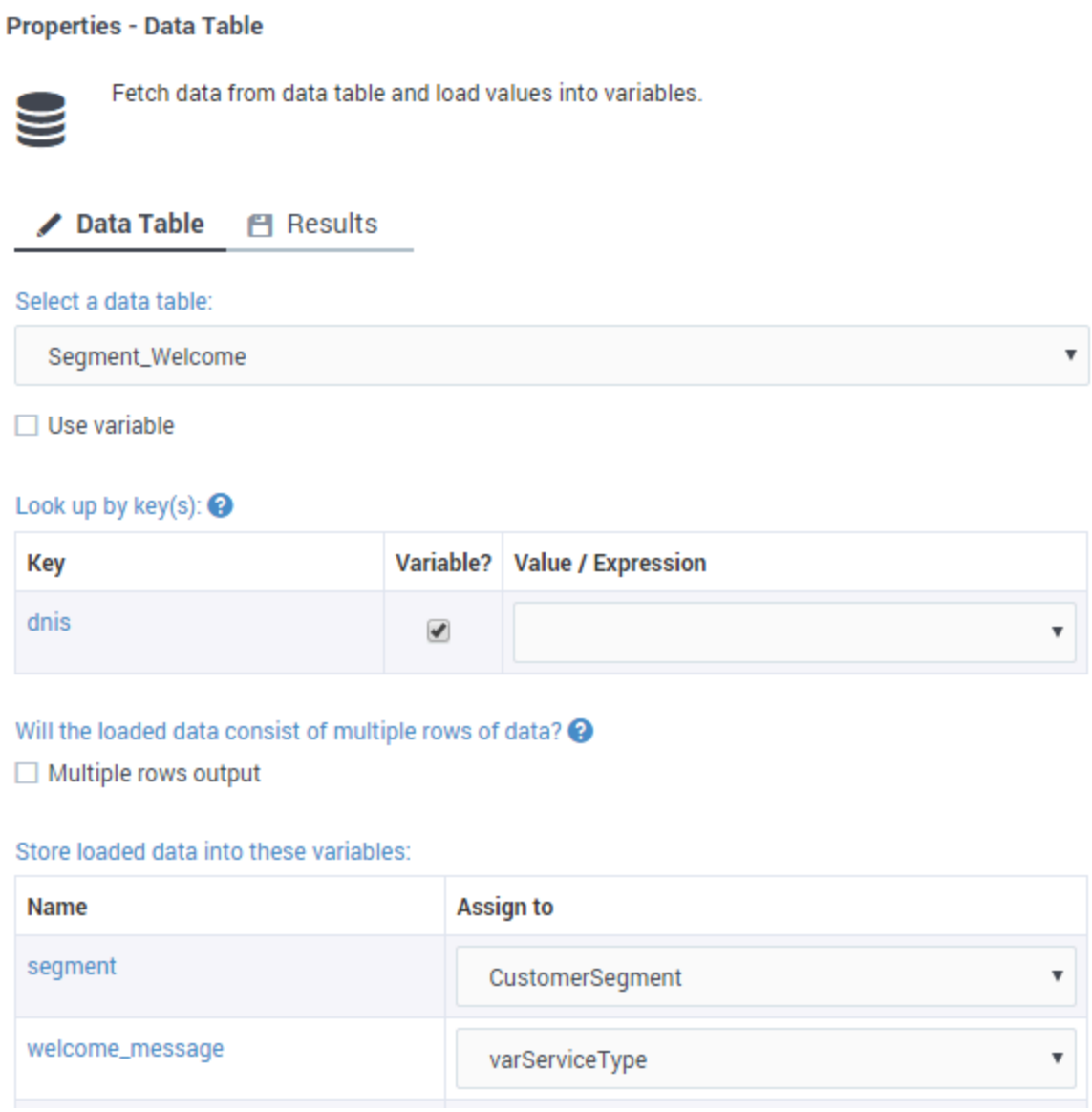
You can use the **Data Tables** block in any phase of your application to read values from a Data Table.

Data Table tab

Select a Data Table in the drop-down list. If you want to enable the option to use the data table as a variable, select **Use variable**. For example, you might select this option if you are using the same application in multiple locations, and each location needs to refer to a data table that is specific to that location.

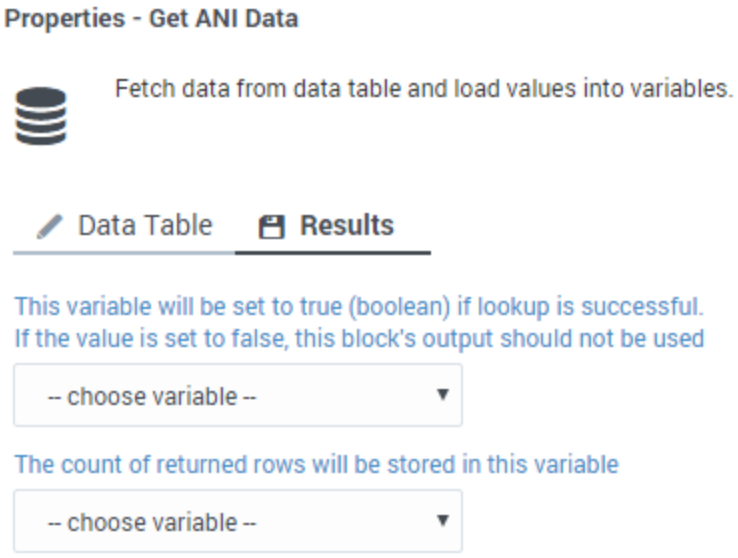
Once you have selected a Data Table, you must configure the following:

- **Look up by key(s)** - For each key in your Data Table, enter a value (or variable, if **Variable?** is enabled) to use as an input for the **Data Table** block.
- **Multiple rows output** - Select this option if the lookup key is of a date/time range data type, or if the data table contains multiple keys.
- **Store loaded data into these variables** - For each column in your Data Table, select a variable to hold the output value of your **Data Table** block.



Results tab

Select a variable to store the outcome status of the lookup (**true** or **false**). You can also select a variable to store the number of returned rows.


Properties - Get ANI Data
Fetch data from data table and load values into variables.
Data Table
Results
This variable will be set to true (boolean) if lookup is successful.
If the value is set to false, this block's output should not be used
-- choose variable --
The count of returned rows will be stored in this variable
-- choose variable --

Emergency Block

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

You can use the **Emergency** block in the **Initialize**, **Self Service**, or **Assisted Service** phase to implement a conditional emergency option in your application.

You can configure this block to play or send an emergency message and then optionally terminate the interaction. This process works only if the emergency mode switch is set to **ON** in either the **Emergency** block or in the Emergency Flags section.

If the switch is set to **OFF**, the block has no effect and it is skipped by the application.

For simple applications, a user typically places this block at the start of the **Self Service** phase. If service is disrupted, the **Emergency** block is easy to locate and enable.

For complex applications that branch into multiple geographic areas, you can place this block in certain segments of a Segmentation block that uses logic to detect branches that are affected by emergency conditions. This allows selective enabling of emergency mode for interactions that require services from affected branches. For example, if your company has two offices and one is closed due to an emergency, you can route interactions to the other office.

> **Tip**
> Remember to set the emergency mode switch to **OFF** once normal operation resumes.

Using this Block

If you have defined an Emergency Flag, enable the **Use Emergency Flags defined in Business Controls** check box. Otherwise, follow the instructions below.

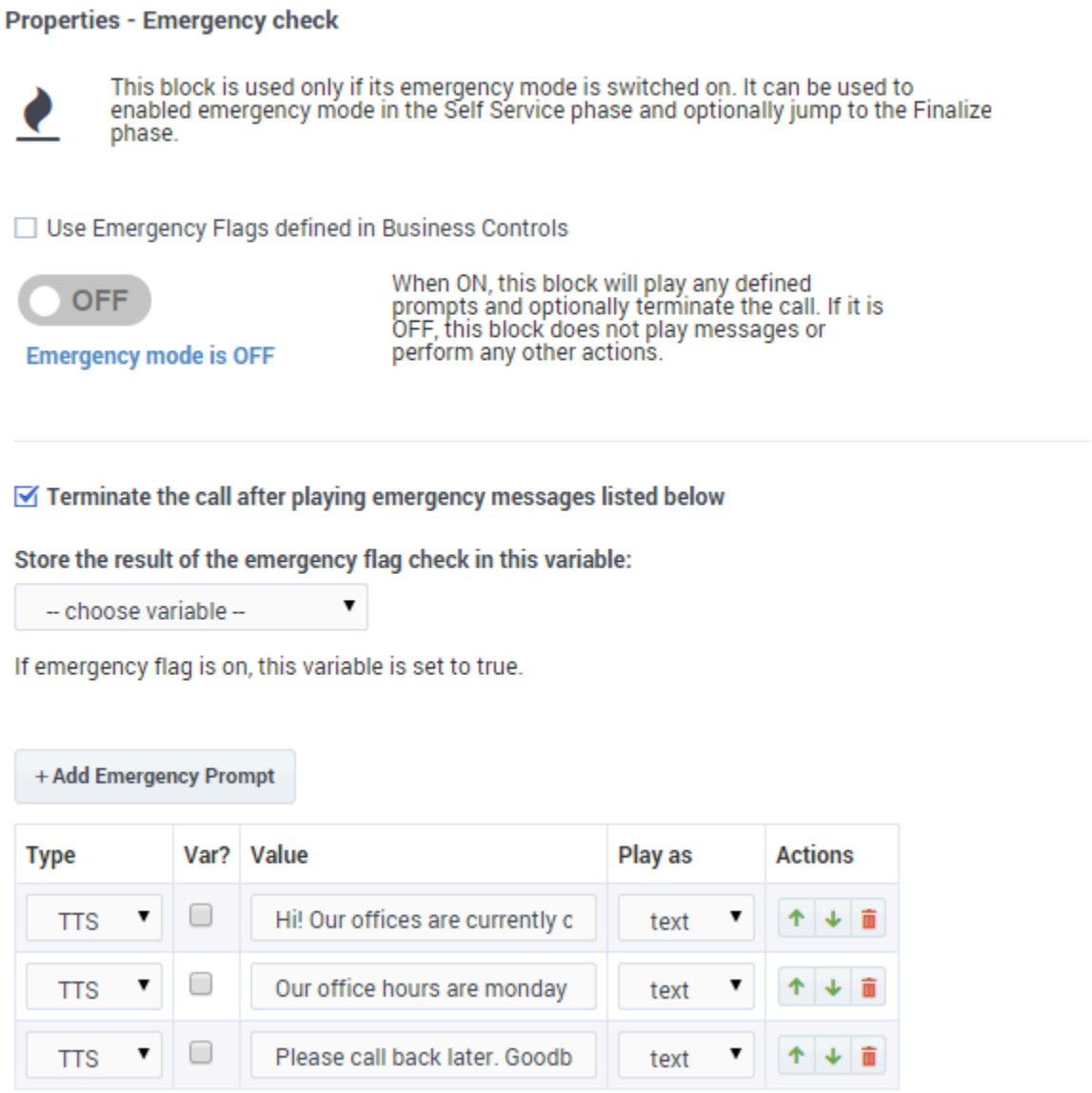
To start, ensure the emergency mode switch is set to **OFF**. You can set this switch to **ON** in an emergency situation.

For Default application types, click **Add Emergency Prompt** to add one or more emergency prompts to play to callers when the emergency mode switch is set to **ON**. For Digital application

types, select the message(s) to be emailed to customers.

Enable the **Terminate the interaction after playing emergency messages listed below** check box if you want the **Emergency** block to end the interaction after playing the emergency prompts.

If you want to store the result of the emergency flag in a variable, select a variable from the list.



Scenarios

If you want to:

- Enable emergency mode:
  - Open your application.
  - Locate the **Emergency** block.
  - Toggle the emergency mode switch to **ON**.
- Control emergency mode from a web service:
  - Add an HTTP REST block in the **Initialize** phase.
  - Assign relevant output to a variable (for example, `varEmergency = true`).
  - In the **Self Service** phase, add a Segmentation block.
  - Add a condition/branch (`varEmergency == true`).
  - Add an **Emergency** block in this segment.
  - Set the emergency mode switch to **ON** permanently in this block.
  - Specify any emergency prompts.
  - Enable or disable the **Terminate the interaction after playing emergency messages listed below** check box.
  - Specify a variable to store the result of the emergency flag check.

Special Day Block

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

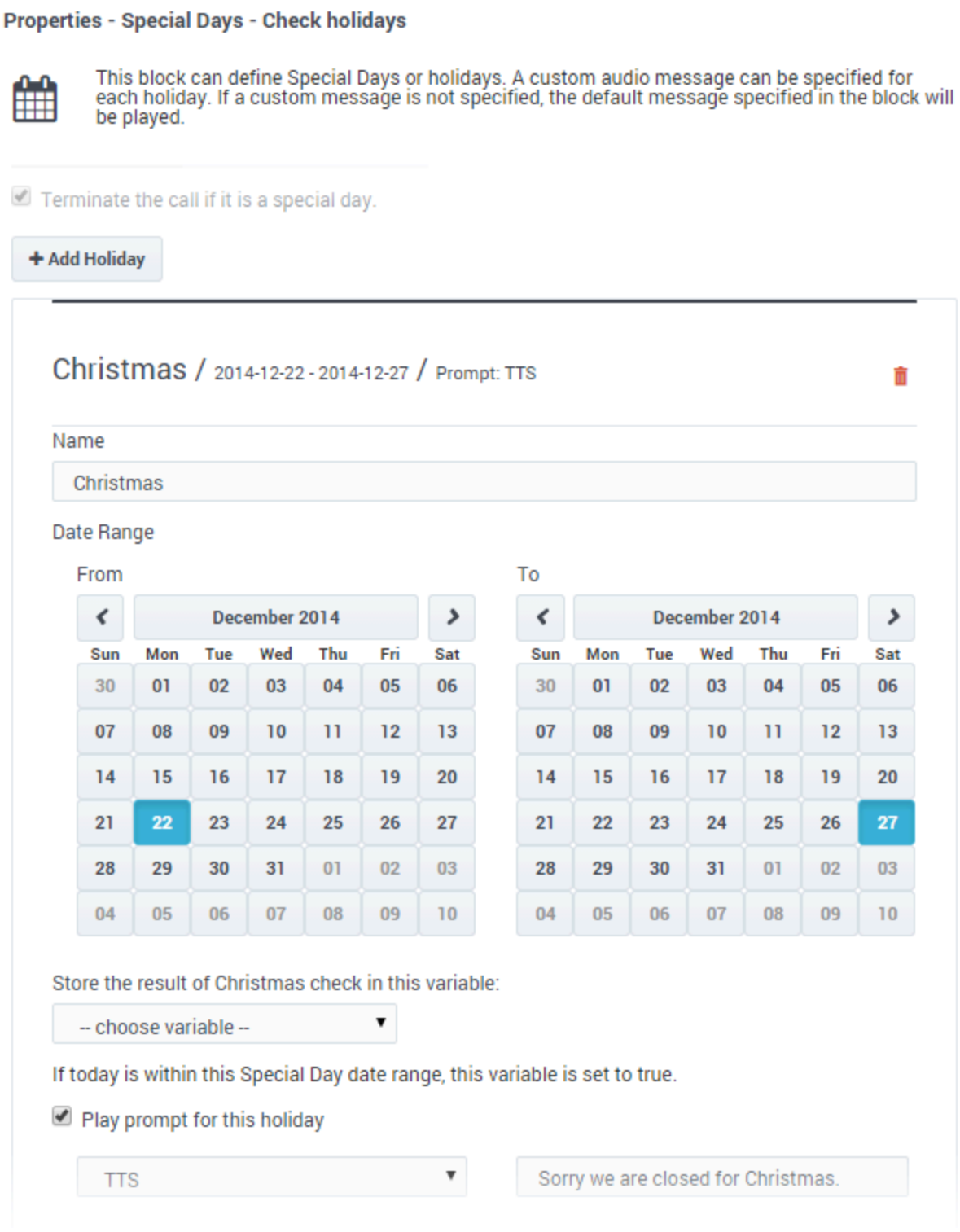
You can use the **Special Day** block in the **Initialize**, **Self Service**, or **Assisted Service** phase to define holidays and other special days, and play prompts or send messages to announce closures or greetings. It can also terminate the interaction if your business is closed.

Holiday tab

Click **Add Holiday** to add a holiday. A holiday entitled **New** appears in the list.

Next, click on the **New** holiday to edit its settings. Configure the following options:

- In the **Date Range** section, use the provided calendars to select the **From** and **To** dates for the holiday.
- Assign a variable to the **Store the result of *Special Day Name* in this variable** property if you want to use the result of this check later in application. If specified, the variable is assigned the Boolean value of true or false.
- For Default application types, you can enable **Play prompt for this holiday** to play a special greeting to customers during a special day.
- For Digital applications type, you can enable **Send message for this holiday** to email a special message to customers during a special day.


Properties - Special Days - Check holidays
This block can define Special Days or holidays. A custom audio message can be specified for each holiday. If a custom message is not specified, the default message specified in the block will be played.
Terminate the call if it is a special day.
+ Add Holiday
Christmas / 2014-12-22 - 2014-12-27 / Prompt: TTS
Name
Christmas
Date Range
From
To
December 2014
Sun Mon Tue Wed Thu Fri Sat
30 01 02 03 04 05 06
07 08 09 10 11 12 13
14 15 16 17 18 19 20
21 22 23 24 25 26 27
28 29 30 31 01 02 03
04 05 06 07 08 09 10
Store the result of Christmas check in this variable:
-- choose variable --
If today is within this Special Day date range, this variable is set to true.
Play prompt for this holiday
TTS
Sorry we are closed for Christmas.

Default Prompts/Messages tab

Click **Add Prompt** to specify a prompt (or message, if the application is a Digital type) to play or send if the application receives an interaction on a special day, and that particular day does not have a custom prompt or message.

Results tab

You can select a variable that will be set to **true** if any of the special days listed in the block evaluate to **true**.

You can also select a variable that will be set to **true** if the special days evaluation processing completed correctly. If it did not, it will be set to **false**.

Scenarios

If you want to:

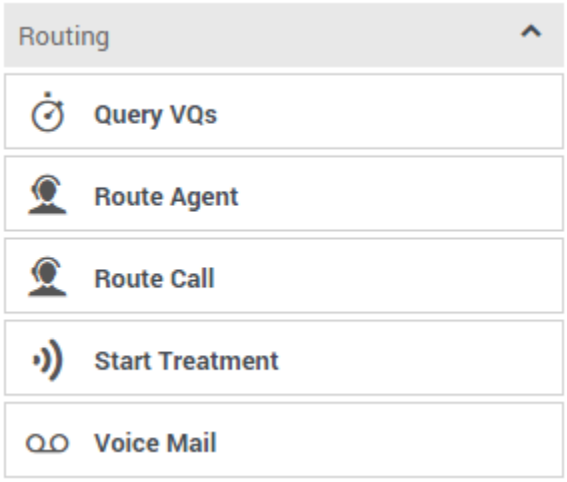
- Play a special greeting during Thanksgiving.
  - Click **Add Holiday** and set the **From** and **To** dates.
  - Enable the **Play prompt for this holiday** check box.
  - Select **TTS** and enter text to speak, or select **Announcement** to choose a predefined announcement.
- Play the same greeting for all holidays.
  - Click **Add Holiday** and create one or more holidays.
  - Do not enable the **Play prompt for this holiday** check box.
  - In the **Default Prompts** tab, add prompts to the table.

Routing Blocks

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

These blocks specify where the interaction should be *routed* when certain conditions are met.

You might not see all of the blocks listed here on your Palette. The blocks shown depend on the features that are enabled and the type of application that is being built. For example, the **Route Digital** block is only available for Digital application types.



Use the links below to learn more about each block.

Query VQs

Queries virtual queues and assigns their Estimated Wait Times.

Used in: **Initialization**, **Assisted Service**

Route Agent

Specifies routing to a particular agent.

Used in: **Assisted Service**

Route Call

Specifies routing to an agent based on various criteria.

Used in: **Assisted Service**

Start Treatment

Plays uninterrupted audio to customers while their interaction is being routed.

Used in: **Assisted Service**

Voice Mail

Routes calls to voicemail.

Used in: **Assisted Service**

Transfer

Transfers a call to another destination.

Used in: **Self Service**

Route Digital

(Digital application types only) Routes a multimedia interaction to a target.

Used in: **Assisted Service**

Predictive Routing

Routes to the agent best equipped to handle the request.

Used in: **Assisted Service**

Query VQs Block

> **Important**
>
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

You can use the **Query VQs** block in the **Initialize** or **Assisted Service** phases to determine the Estimated Wait Time for several virtual queues.

Before a call is queued, the **Query VQs** block can check the Estimated Wait Time for all targeted virtual queues to help determine which virtual queue receives the call. The results are stored in user variables.

> **Important**
>
> If the query isn't able to obtain a result from a targeted virtual queue, the value of the Estimated Wait Time is set to zero (0).

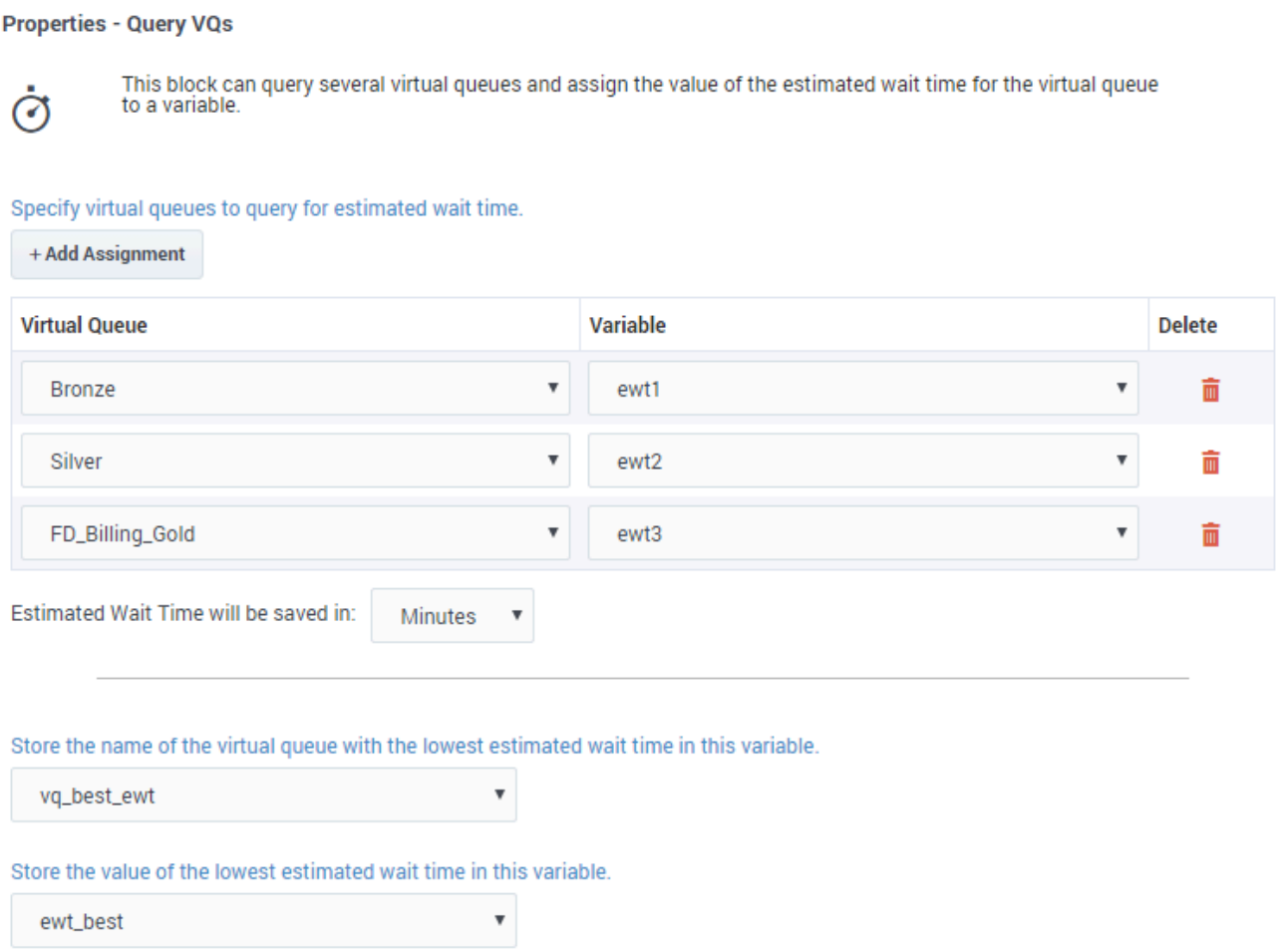
Using this Block

In the block properties, click **Add Assignment**.

Next, select the **Virtual Queue** to query and the **Variable** to which Designer assigns the Estimated Wait Time value for that virtual queue.

Choose whether to save the Estimated Wait Time in **minutes** or **seconds**.

In the drop-down menus, select the variable to store the **name** of the virtual queue with the lowest Estimated Wait Time value and the variable to store the **value** of the lowest Estimated Wait Time.


Properties - Query VQs
This block can query several virtual queues and assign the value of the estimated wait time for the virtual queue to a variable.
Specify virtual queues to query for estimated wait time.
+ Add Assignment
Virtual Queue
Variable
Delete
Bronze
ewt1
Silver
ewt2
FD_Billing_Gold
ewt3
Estimated Wait Time will be saved in:
Minutes
Store the name of the virtual queue with the lowest estimated wait time in this variable.
vq_best_ewt
Store the value of the lowest estimated wait time in this variable.
ewt_best

Route Agent Block

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

You can use the **Route Agent** block in the **Assisted Service** phase to route calls to an agent based solely on:

- a specified **Agent ID** and **Virtual Queue**
- a specified **Agent Login** and **SIP Switch**, or
- the **Last Called Agent** that the caller spoke to

To route calls based on skills and other criteria, use the Route Call block.

You can sequentially place multiple **Route Agent** blocks with different settings, so that if routing fails in one block, your application proceeds to the next block. When a **Route Agent** block successfully routes the call to an agent, the application moves to the **Finalize** phase, ignoring any subsequent blocks in the **Assisted Service** phase.

> **Important**
> The **Route Agent** block can only be used to route voice calls. It does not support the routing of chat interactions.

Agent Routing tab

Agent Routing

Select a Routing type:

- **Agent ID** - Select the **Agent ID** (specified as a variable) and **Virtual Queue** that the call will be routed to.
- **Agent Login** - Select the **Agent Login** (specified as a variable) and **SIP Switch** (where the selected agent is logged on, also specified as a variable) that the call will be routed to.

- **Last Called Agent** - Select this option to route the call to the agent that the caller last spoke with.

> **Important**
>
> If you select the **Last Called Agent** option, you must also select the **Enable Customer Profile** option in the Contextual Data tab of the Application Settings.

Other Routing Settings

- **Overall timeout** - If enabled, you can specify how long the application must wait before moving on to the next block. Optionally, you can enable the check box to specify a variable.
- **Route only if the Agent is ready** - If enabled, a call is routed to an agent only if his status is set to Ready. If disabled, the call is routed to an agent regardless of his status.

Treatments tab

Specify a busy treatment to execute while waiting for an agent to become available. You can choose to play audio and/or execute a shared module.

- Learn more about busy treatments

> **Important**
>
> After a busy treatment has been executed at least 10 times, Designer exits the **Route Agent** block and moves to the next block if the average duration of the treatment is less than 1000 ms (for example, due to a missing audio file).

Audio

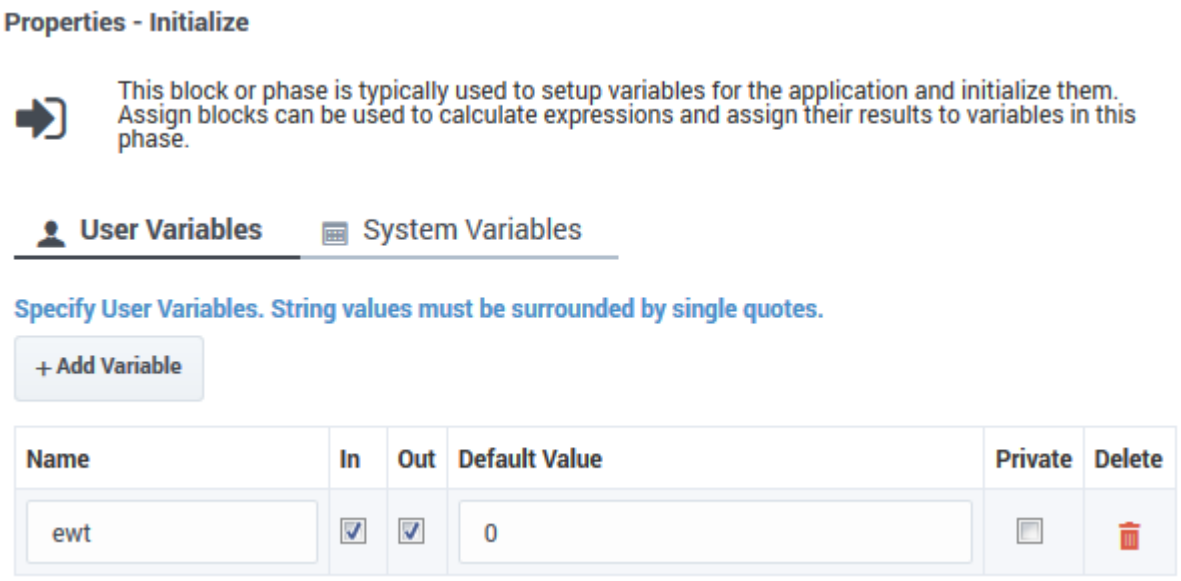
Click **Add Audio** to add a Play Message child block underneath this **Route Agent** block. The collection of audio plays repeatedly until the call is successfully routed or times out.

Shared Module

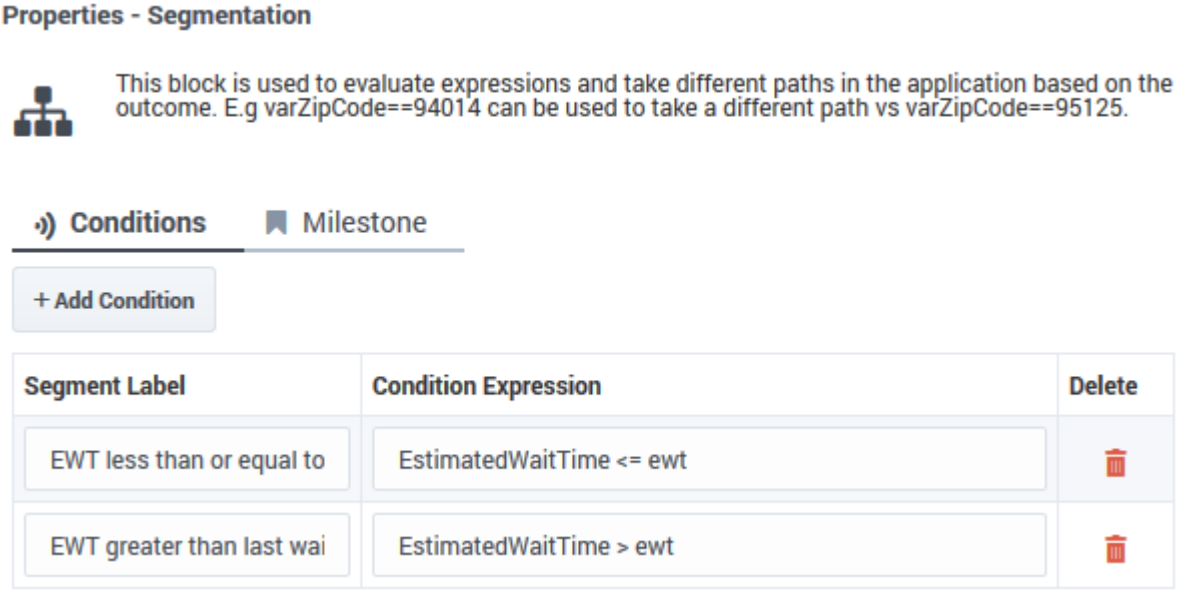
Click **Add Module** to add a Shared Module child block underneath this **Route Agent** block. In the child block, you can select a shared module to execute.

A potential use case is to execute a shared module based on a specified set of conditions that can change over time and respond to external factors. For example, you might use a shared module that can play one announcement for callers if the estimated wait time (EWT) is beyond a certain threshold, and another announcement for when they are the next caller in the queue. To set up this feature:

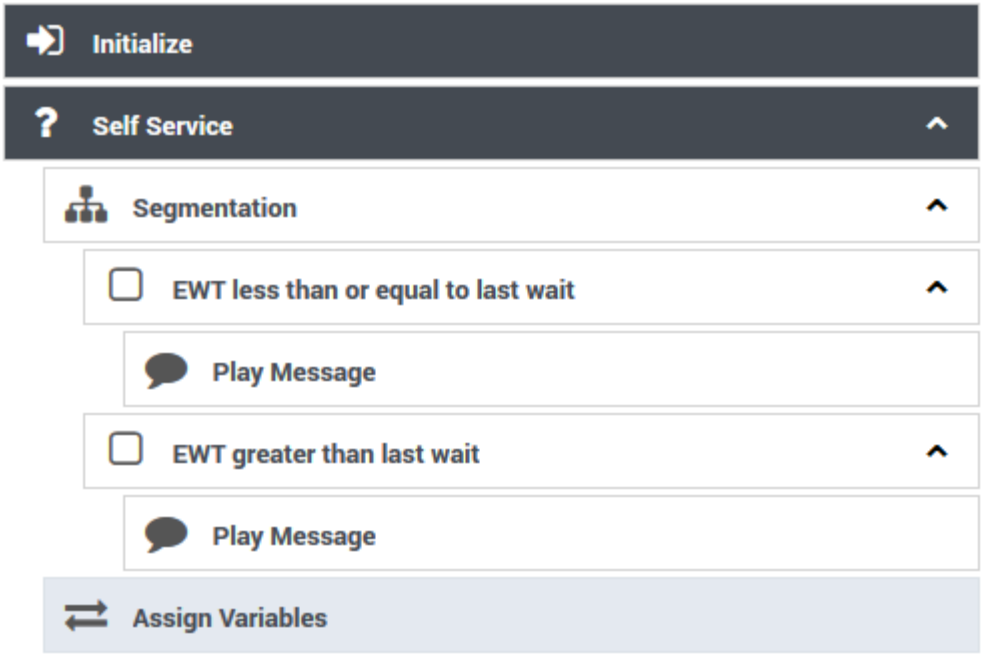
1. In the application, create a user variable, `ewt`, and set its default value to `0`.
2. Create a **Self Service** type shared module.
3. In the shared module, create a user variable, `ewt`, and set its default value to `0`.



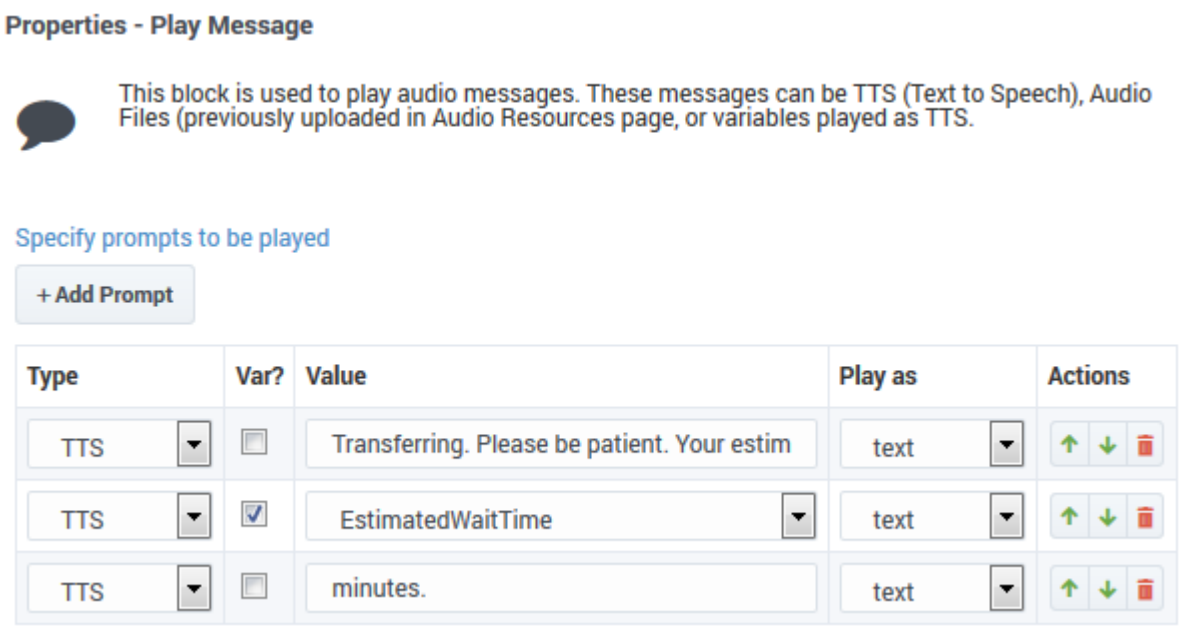
4. In the **Self Service** phase of the shared module, add a **Segmentation** block. Add the conditions as shown below:



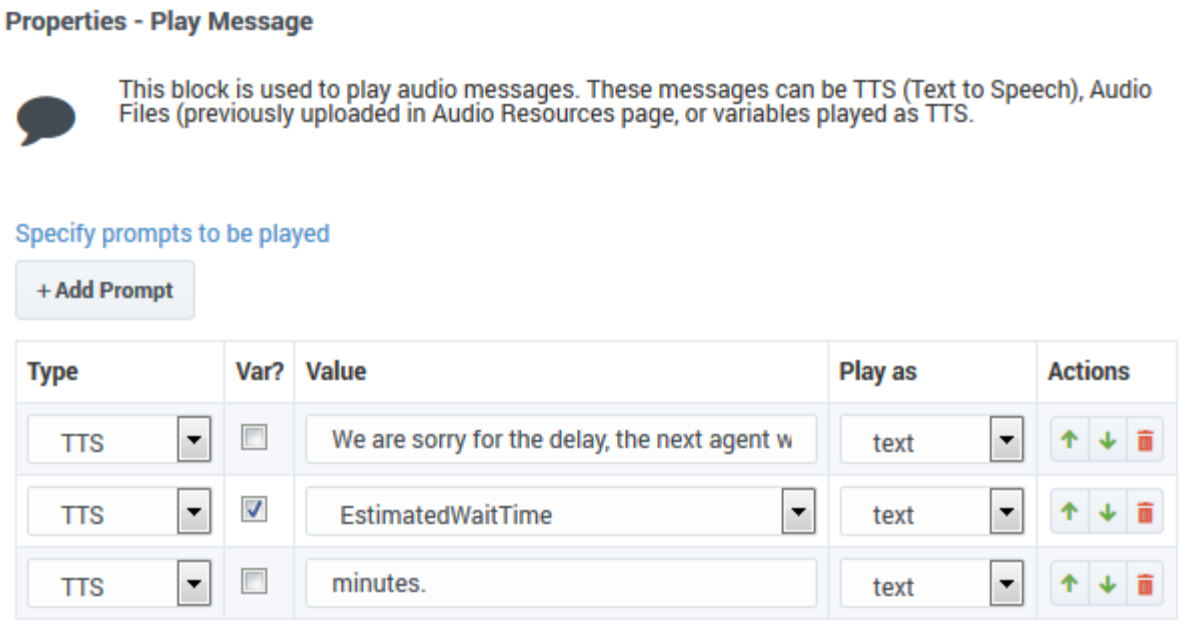
5. Add two **Play Message** blocks as child blocks of the condition blocks, and add an **Assign Variables** block at the end. Your shared module should appear as shown below:



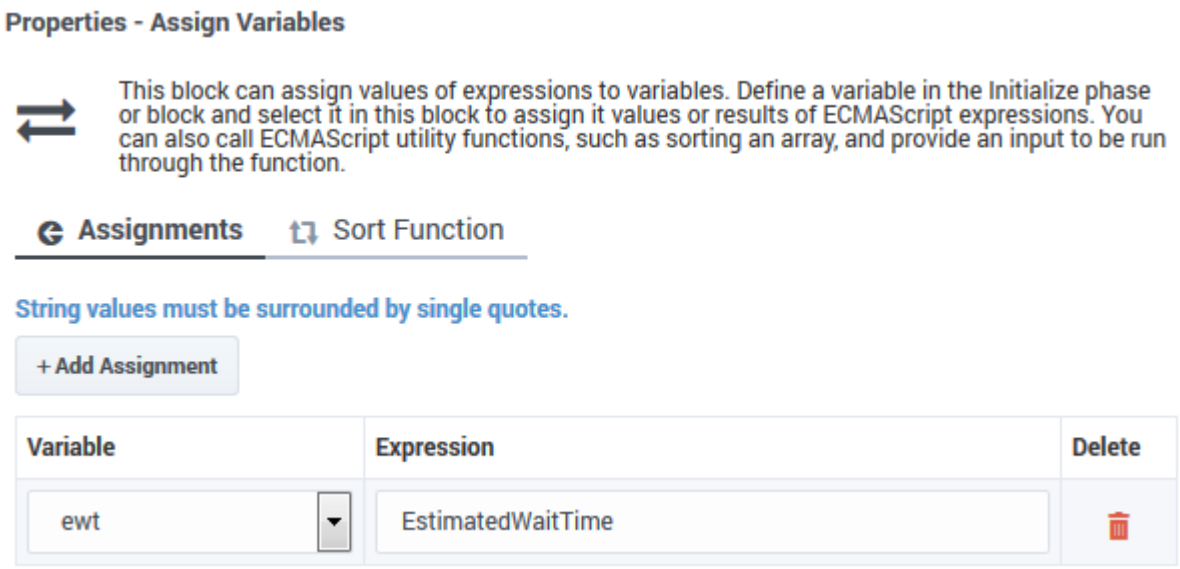
6. Configure the first **Play Message** block. An example is below:



7. Configure the second **Play Message** block. An example is below:



8. Configure the **Assign Variables** block as shown below:



9. In your application, select the **Route Agent** block and click the **Treatments** tab.
10. Click **Add Module**. A child **Shared Module** block appears beneath the **Route Agent** block.
11. In the child **Shared Module** block, select the shared module that you created in Step 2.

The application passes **ewt** to the shared module, along with the system variables, which includes **EWT**. The shared module compares **ewt** and **EWT** in the **Segmentation** block and executes a **Play Message** block, depending on which variable is larger. At the end, the shared module sets **ewt** to **EWT** before returning to the application.

Advanced tab

Targeting

Use Agent Status

If checked, the routing engine uses the **status** flag to route the call to an agent. If not checked, the routing engine uses the **loggedin** flag to route the call to an agent.

> **Important**
> This check box has no effect if you enabled the **Route only if the Agent is ready** check box in the **Agent Routing** tab.

Threshold Expression

This option enables you to use an ECMAScript (or JavaScript) expression to further refine a routing threshold for the specified target(s). Threshold expressions for the **Route Agent** block can be used for the following routing types:

- Agent ID
- Agent Login
- Last Called Agent

Threshold expressions can contain variables or reference queue-specific values, such as *sdata(target, statistic)* or *callage()*. Strings must be enclosed in single quotes. For example:

For more information about using ECMAScripts in Designer, see Assigning values to variables.

> **Important**
> For routing types that have multiple targets, the script defined in **Threshold Expression** applies to all targets.

Extensions

Click **Add Extension Data** to add an extension as a key-value pair to this block. The value type can be a string or integer.

If you want to use a variable for the **Key** or **Value**, select the **Variable** checkbox and then select a variable from the drop-down menu. If the **Value** is an integer, select the **Integer** checkbox.

Important

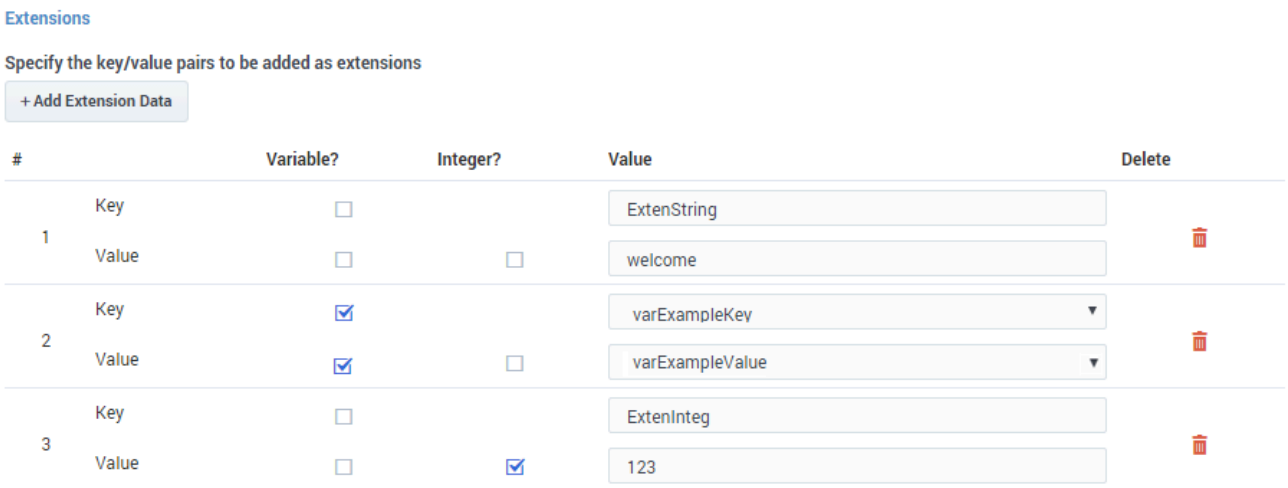
You do not need to enclose extension values in quotes. However, if the quote is part of the value, you must escape the quote character by using a preceding backslash. For example:

- Incorrect: `Joe's Pizza`
- Correct: `Joe\'s Pizza`

Important

Designer displays an error message if Extension Data is added, but the **Key** and **Value** settings are not defined.

This example shows a few different ways that key-value pairs can be added as extensions:



Results tab

Select a variable to store the result of this **Route Agent** block execution.

Route Call Block

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

You can use the **Route Call** block in the **Assisted Service** phase to route calls to an agent based on various criteria, such as Skills and Agent Groups.

You can sequentially place multiple **Route Call** blocks with different settings, so that if routing fails in one block, your application proceeds to the next block. When a **Route Call** block successfully routes the call to an agent, the application moves to the **Finalize** phase, ignoring any subsequent blocks in the **Assisted Service** phase.

Call Routing tab

Select Routing Type section

Choose between the following routing options:

Skill based routing with relaxing criteria

Routes the call to an agent that has the required skills. If selected, you can choose from the following options:

- **Use system variables RoutingTarget / RoutingVirtualQueue set already in Menu Options** - Use system variables that were set in a Menu Option block.
- **Specify Skills in this block** - Specify one or more skills and a Virtual Queue to use to route this call. If you specified more than one skill, you can choose whether the routing engine considers any or all of the selected skills:
  - **all skills** - The application must use all of the selected skills to route the call.
  - **any skill** - The application can use any of the selected skills to route the call.

> **Important**
> This option uses the skill level specified in the **Use Skill Proficiency level** setting (documented below). For

> example, if you set an initial skill level of 8, Designer only routes the call to agents with the specified skills that have a level of 8 or greater. You cannot set an individual level for each specified skill.

- **Use Skill Proficiency level** - Enter a Skill level. The call is routed to an agent that has a skill level equal to or higher to the value you provide. If you enable **Reduce skill requirements**, the required skill level is gradually decremented by a specified skill level, until it reaches the specified minimum skill level. This option allows you to expand the group of agents that can receive this call if other agents are busy.

Skill expression based routing

Enter a skill expression in the **Skill Expression** tab, or click the drop-down menu to select a variable that specifies a skill expression.

Agent Group routing

Route the call to a specific Agent Group or a variable that holds the name of an Agent Group at runtime.

Agent routing

Route the call to agents by using a variable that holds the ID of an agent at runtime. You must use the following format: *agentid*`@`*optional_statserver*`.A`.
Example: `1001@StatServer.A`.

Campaign Group routing

Route the call to a specific Campaign Group or a variable that holds the name of a Campaign Group at runtime.

Route to another Application

Transfer the interaction to the selected application. You can then specify a Phone Number (or the variable containing the Routing Point name) that is associated with the target application.

Direct number routing (voice only)

Transfer a call to a number. You can use a variable to hold the number to use at runtime or add direct number elements. Specify the weight for each number and Designer displays and uses the percentage ranking based on the weightings.

> **Important**
> Genesys recommends using **Force route (voice only)** instead of this option.

Force route (voice only)

Force the call to route to a direct number. When using this routing type, Designer routes the call in a way that is similar to how an interaction is redirected by Route Point routing. When selected, you can specify the target as a *literal* value, or as a variable that holds a *string*, *number*, or *object* value.

> **Important**
>
> When using **Force Route**:
>
> - The **Routing Priority** tab and the **Targeting Options** in the **Advanced** tab (**Clear targets from queue if this block times out** and **Early exit from this block if no agents are logged in**) are not applicable.
> - The overall timeout for the **Route Call** block is limited to 30 seconds.

Parking Queue (digital only)

If a digital interaction arrives when the business is closed, send the interaction to a parking queue until the business is open. If selected, you can then specify a variable that tells Designer how long the interaction is to be parked (for example, the variable that holds the number of minutes until the business is open). When regular business hours resume, the interaction is retrieved and processed.

Other Routing Settings section

Routing Algorithm

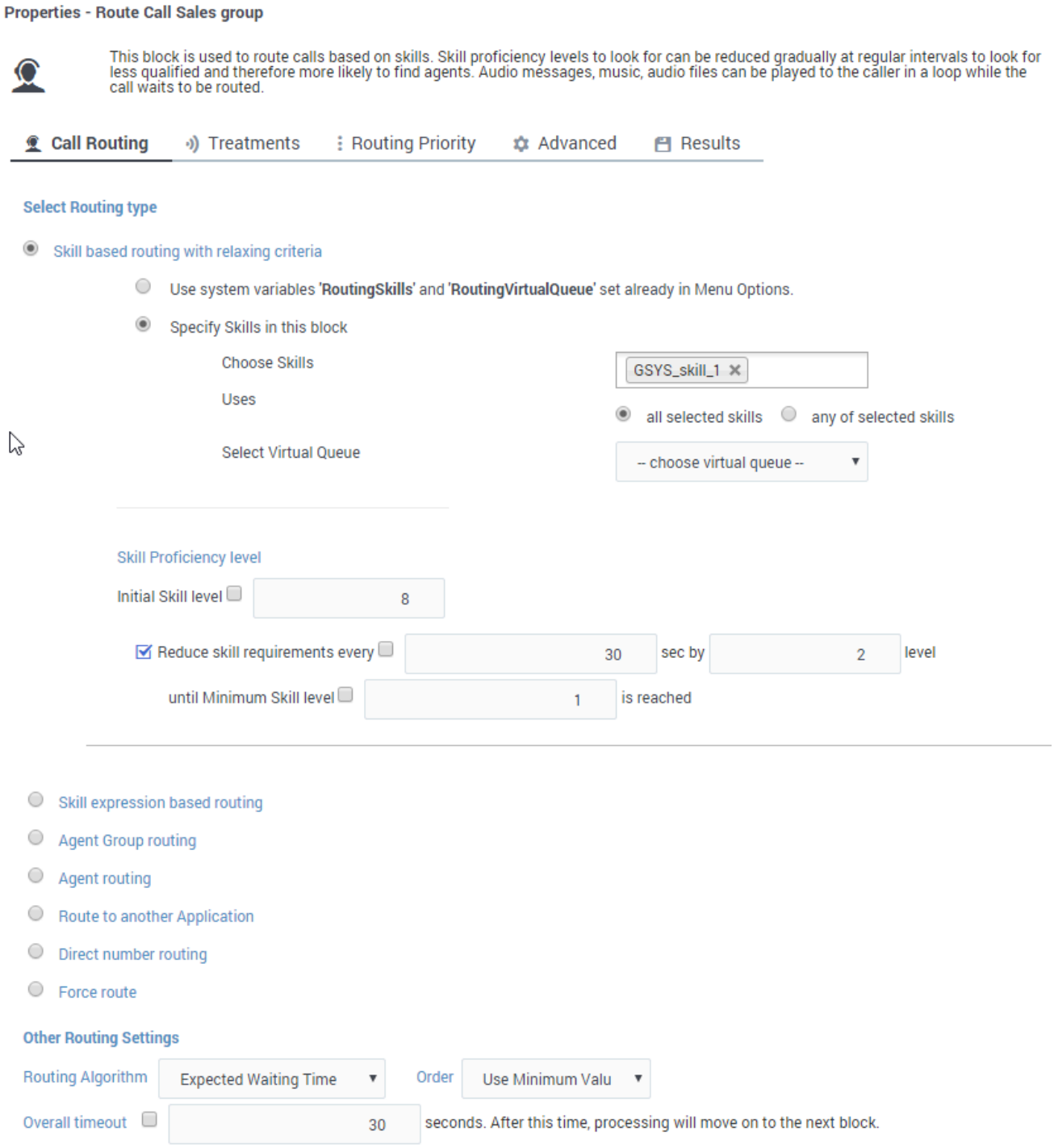
Select which algorithm is used to choose an agent when more than one agent is available. You can also specify this value with a variable. For more information, see Routing Algorithms.

Overall timeout

Enter the maximum time (in seconds) to wait for an agent to be available before moving to the next block. Optionally, you can enable the check box to specify a variable.

> **Important**
>
> System variables **SelectedTarget**, **SelectedVirtualQueue**, **SelectedComponent**, **SelectedTargetObject**, **SelectedAgent**, and **Access** are automatically set when the call is routed to an agent and can be used later in the application. Refer to the **Initialize** phase's **System Variables** tab to read a detailed description for each of these variables.

Example


Properties - Route Call Sales group
This block is used to route calls based on skills. Skill proficiency levels to look for can be reduced gradually at regular intervals to look for less qualified and therefore more likely to find agents. Audio messages, music, audio files can be played to the caller in a loop while the call waits to be routed.
Call Routing
Treatments
Routing Priority
Advanced
Results
Select Routing type
Skill based routing with relaxing criteria
Use system variables 'RoutingSkills' and 'RoutingVirtualQueue' set already in Menu Options.
Specify Skills in this block
Choose Skills
GSYS_skill_1
Uses
all selected skills
any of selected skills
Select Virtual Queue
-- choose virtual queue --
Skill Proficiency level
Initial Skill level
8
Reduce skill requirements every
30
sec by
2
level
until Minimum Skill level
1
is reached
Skill expression based routing
Agent Group routing
Agent routing
Route to another Application
Direct number routing
Force route
Other Routing Settings
Routing Algorithm
Expected Waiting Time
Order
Use Minimum Valu
Overall timeout
30
seconds. After this time, processing will move on to the next block.

Skill Expression tab

> **Important**
>
> This tab only appears if you selected the **Skill expression based routing** option in the **Call Routing** tab.

If you selected the option **Skill expression based routing** in the **Call Routing** tab, you must build the skill expression to identify the best agent to handle the call. The skill expression consists of a list of skills for which you must individually set an operator and an integer value.

Arrange individual skill conditions in the conditions sets. You can specify skills by name or variables that contain the name of the skills at runtime.

> **Important**
>
> When using **Skill expression based routing** and you are building the entire skill expression within a variable, you must manually add the single quotes around the skill names.
>
> For example, use this:
>
> ```
> "'New iPhone' > 7"
> ```
>
> instead of this:
>
> ```
> "New iPhone > 7"
> ```

Treatments tab

Specify a busy treatment to execute while waiting for an agent to become available. You can choose to play audio and/or execute a shared module.

- Learn more about busy treatments

If your application supports digital interactions, you can specify the number of seconds that will elapse between each treatment iteration (this setting is ignored for voice interactions).

> **Important**
>
> After a busy treatment has been executed at least 10 times, Designer exits the **Route Call** block and moves to the next block if the average duration of the treatment is less

than 1000 ms (for example, due to a missing audio file). (However, this does not apply if the **Force Route** option is selected.)

Audio

Click **Add Audio** to add a Play Message child block underneath this **Route Call** block. The collection of audio plays repeatedly until the call is successfully routed or times out.

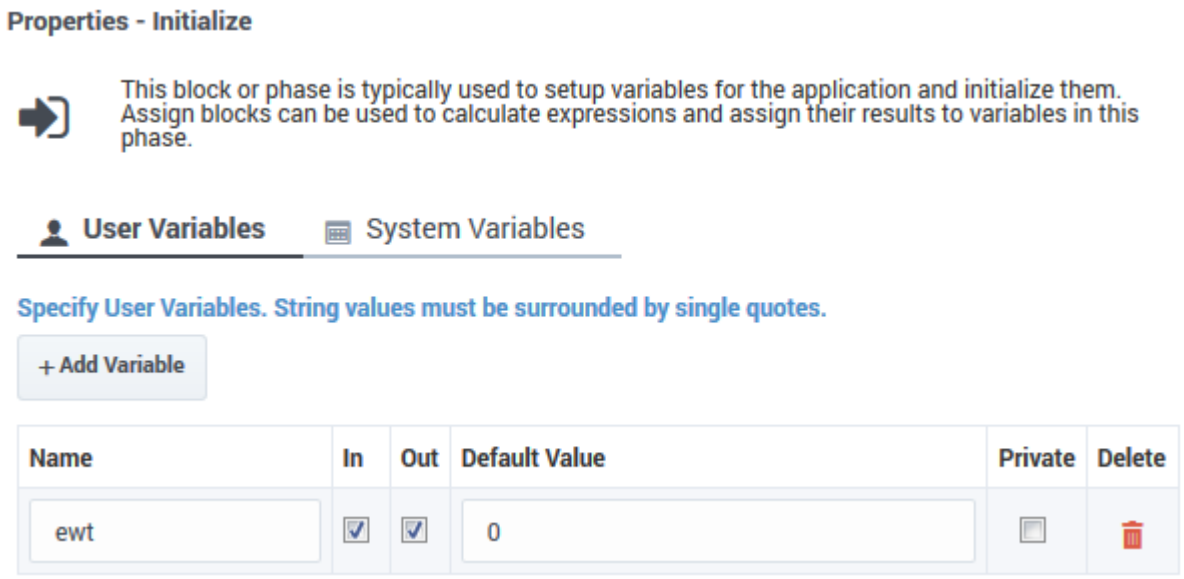
Shared Module

Click **Add Module** to add a Shared Module child block underneath this **Route Call** block. In the child block, you can select a shared module to execute.

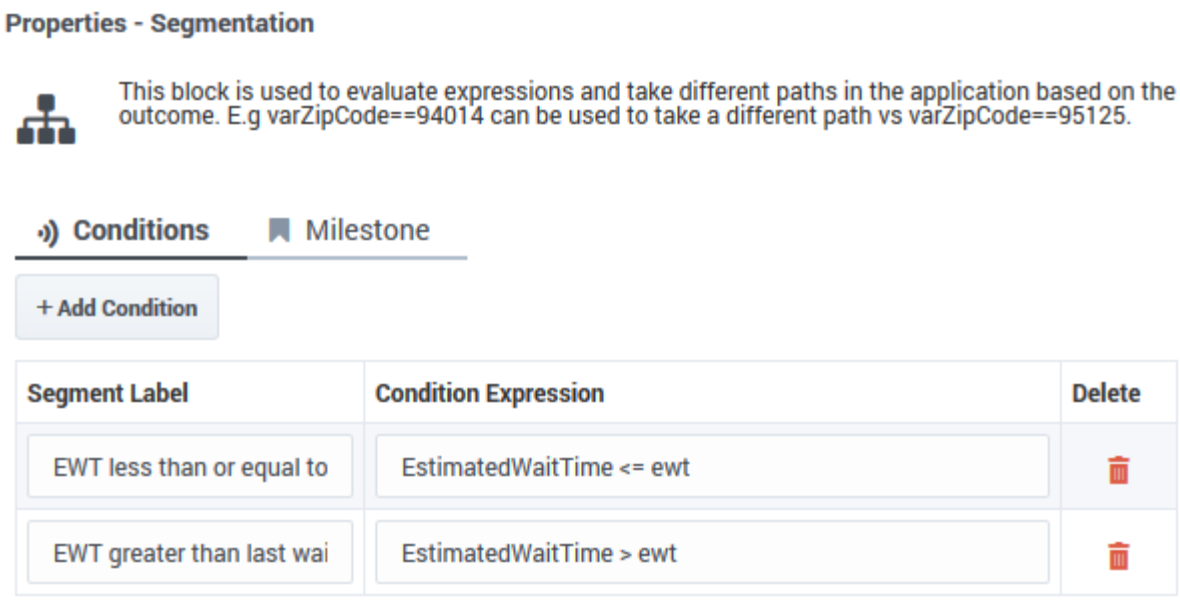
A potential use case is to execute a shared module based on a specified set of conditions that can change over time and respond to external factors. For example, you might use a shared module that can play one announcement for callers if the estimated wait time (EWT) is beyond a certain threshold, and another announcement for when they are the next caller in the queue.

To set up this feature:

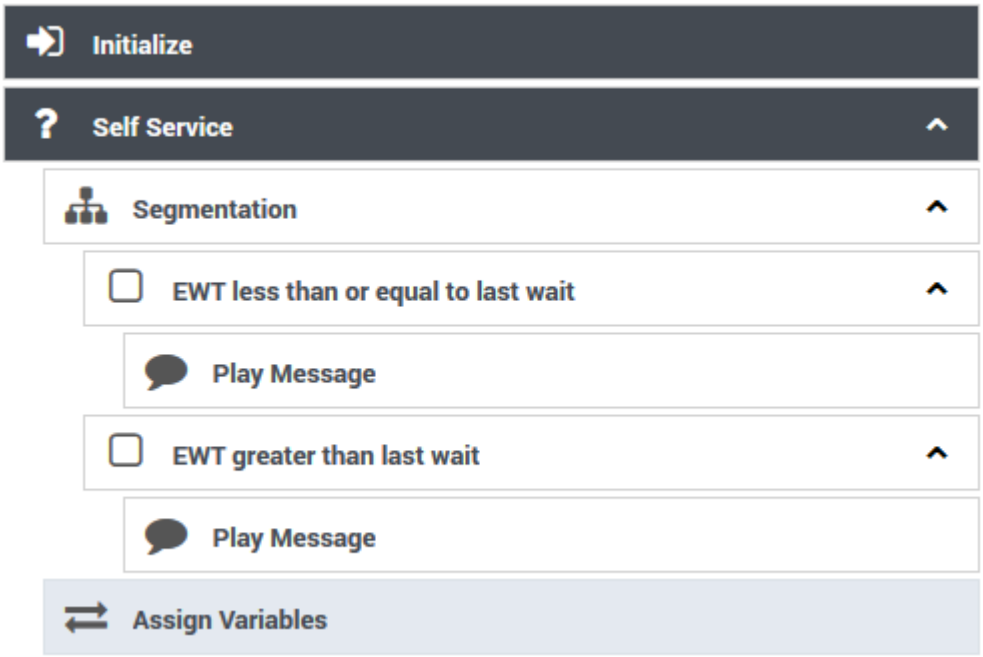
1. In the application, create a user variable, `ewt`, and set its default value to `0`.
2. Create a **Self Service** type shared module.
3. In the shared module, create a user variable, `ewt`, and set its default value to `0`.



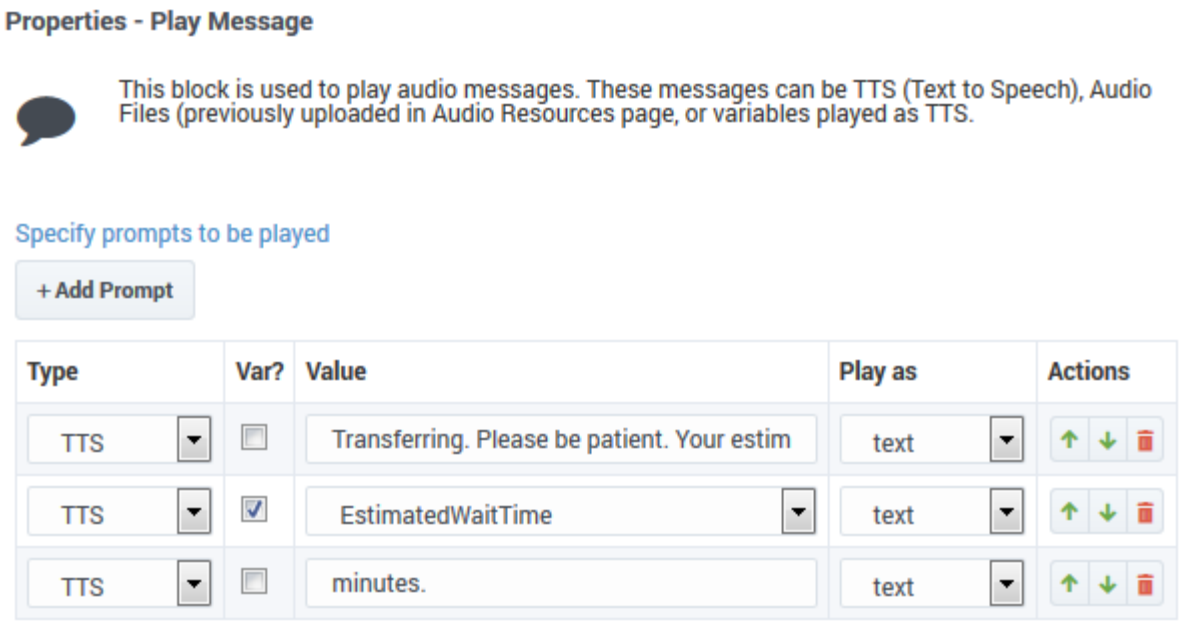
4. In the **Self Service** phase of the shared module, add a **Segmentation** block. Add the conditions as shown below:



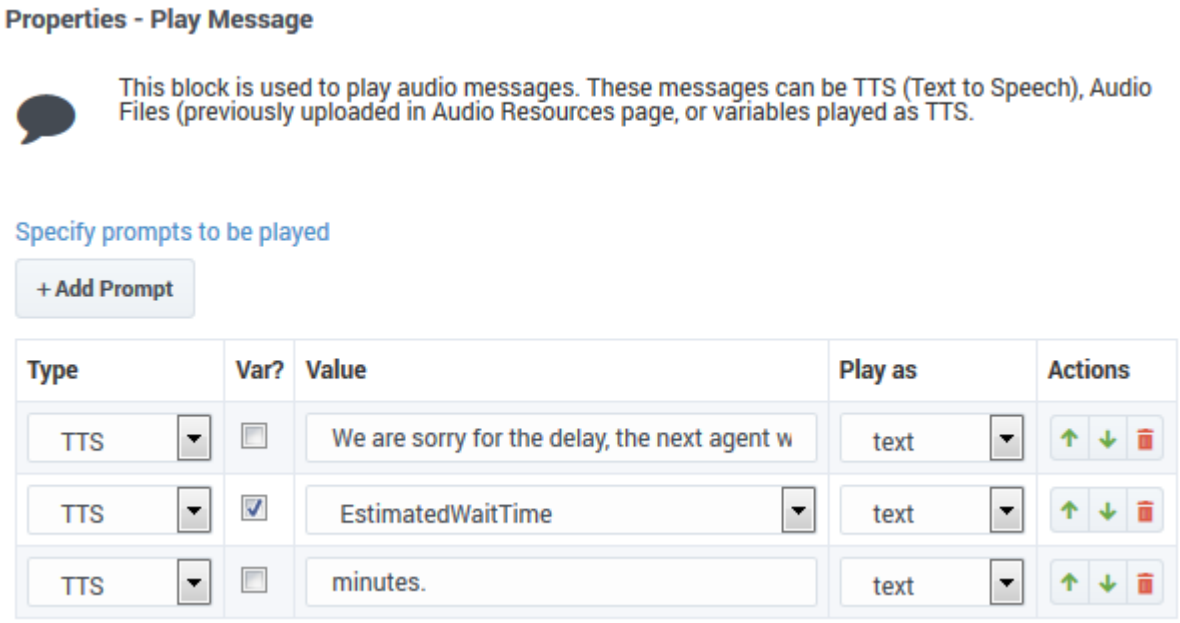
5. Add two **Play Message** blocks as child blocks of the condition blocks, and add an **Assign Variables** block at the end. Your shared module should appear as shown below:



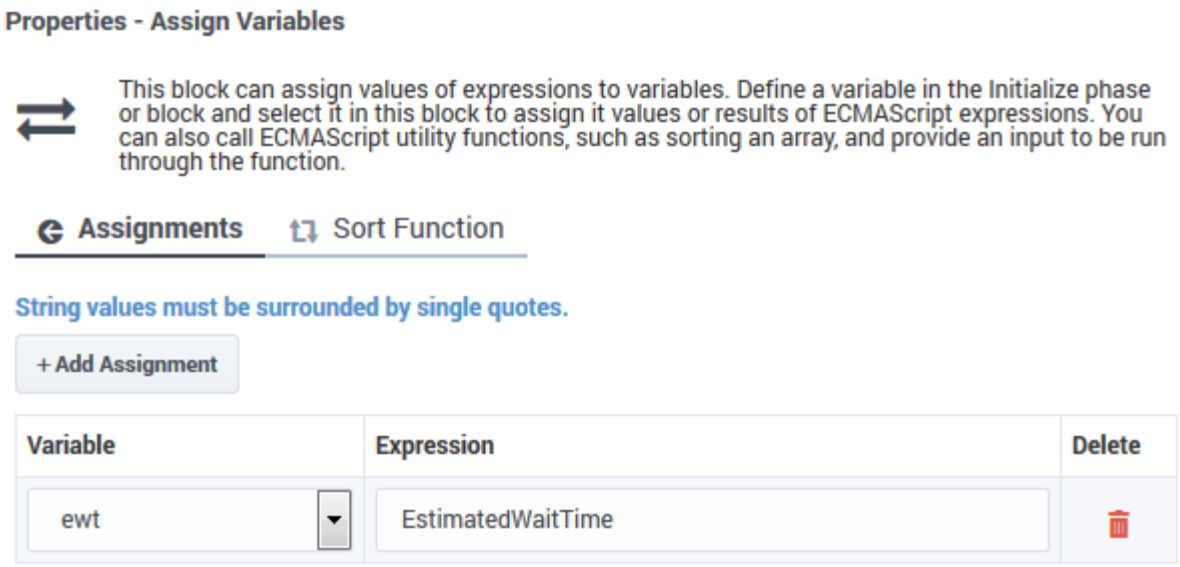
6. Configure the first **Play Message** block. An example is below:



7. Configure the second **Play Message** block. An example is below:



8. Configure the **Assign Variables** block as shown below:



9. In your application, select the **Route Call** block and click the **Treatments** tab.
10. Click **Add Module**. A child **Shared Module** block appears beneath the **Route Call** block.
11. In the child **Shared Module** block, select the shared module that you created in Step 2.

The application passes **ewt** to the shared module, along with the system variables, which includes **EWT**. The shared module compares **ewt** and **EWT** in the **Segmentation** block and executes a **Play Message** block, depending on which variable is larger. At the end, the shared module sets **ewt** to **EWT** before returning to the application.

Routing Priority tab

Use Priority during Routing

Enable this check box to use priority-based routing, which prioritizes your calls depending on your business requirements.

To prioritize calls, you must segment calls and assign the name of that segment to a variable. You must select this variable in the **Lookup Priority table based on this variable** drop-down menu.

You can customize this table with your own segment definitions to fit your business needs. If the specific segment is not found, then the value specified for **Initial priority** is used. Enter a value in **Increment size** to increase the priority of a call that remains in a queue over time. The priority increment is defined for each segment, but a default increment is configurable with the **Increment Size** property.

Increment Priority every ___ seconds

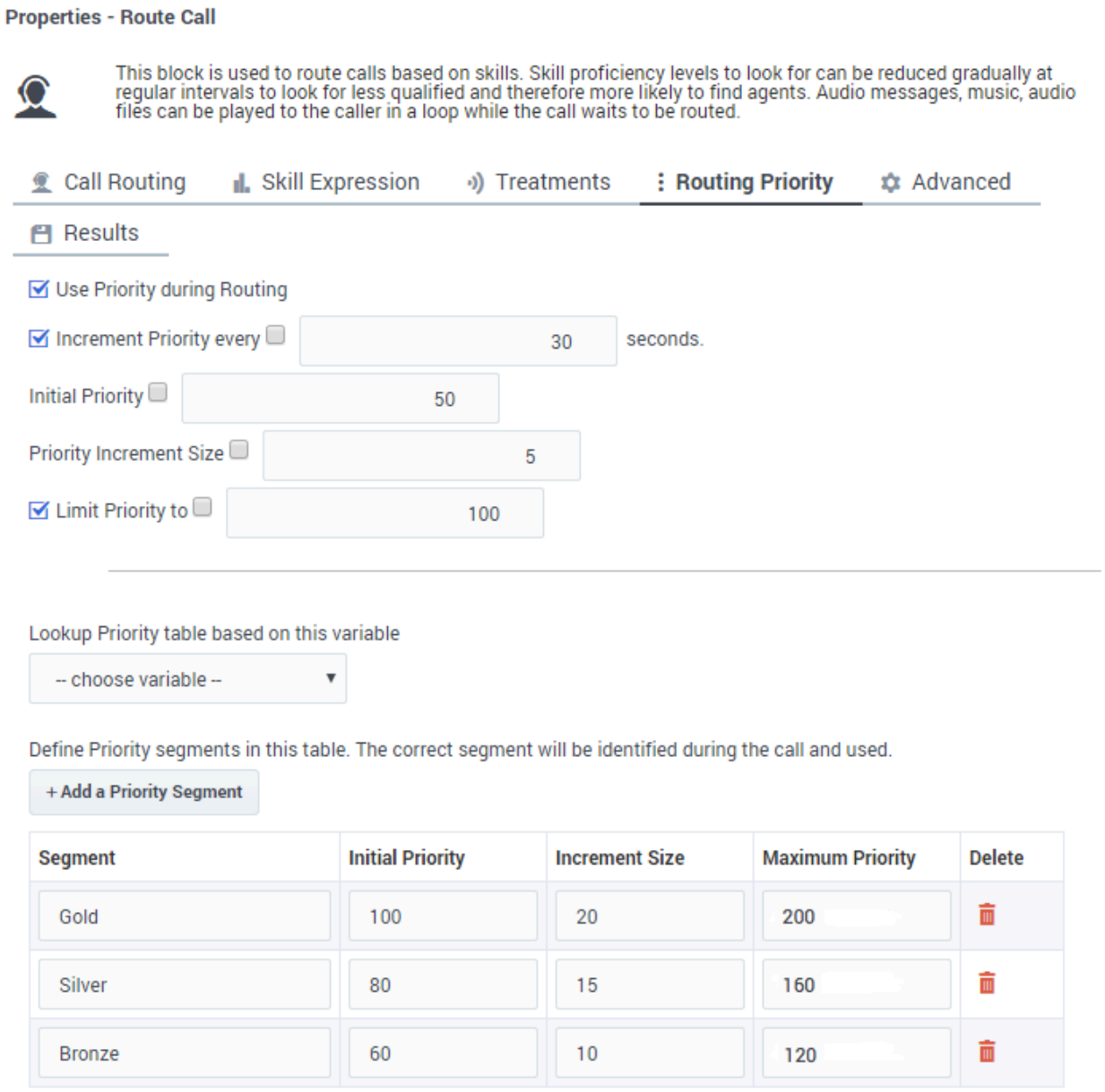
Enable this check box to specify the time interval between priority increments. If you enable the other check box beside the field, you can select a variable that specifies the overall **Routing**

Timeout and **Priority Increment Interval** properties.

Limit Priority to

If the **Increment Priority every ___ seconds** option is enabled, you can use this option to set a maximum priority value. For example, if the initial priority is `50`, you can use this option to not let the priority value increase beyond `100`, as shown here:

Example



If you enable the other check box beside the field, you can select a variable for this option.

Advanced tab

Targeting section

Clear targets from queue if this block times out

Enable this check box to specify whether the pending request for a target should be kept active or not after exiting this block on timeout. When the request is kept active (check box is disabled), an agent may be selected after the block times out if, for example, an agent with the matching criteria is ready after the block was exited.

Early exit from this block if no agents are logged in

Enable this check box to exit the block if no agents are logged in for the selected routing target (such as Agent or Agent Group, skill expression based, or skill based routing with relaxing criteria).

> **Important**
> Designer runs periodic checks to determine if any agents are logged in. Therefore, it can take up to 60 seconds for the application to actually exit the block.

Prioritize routing to local agents

If you have selected **Skill based routing with relaxing criteria** or **Skill expression based routing**, you can enable this option. When enabled, the call is routed to a local agent who matches the target skill.

> **Tip**
> If you want to route to local agents as the preferred option, but then route to all agents if there are no local agents available with the required skill, you can set up cascaded routing.
>
> Here's a way you can do that:
>
> - Set up the **Route Call** block with **Route only to local agents** enabled, a short **Overall timeout** property value, and **Clear targets from queue if this block times out** deselected.
> - Then, set up any **Route Call** blocks that are further down the application flow with **Route only to local agents** not selected.

You can watch this video to see a short demonstration of how to set this up.

Link to video

You might also want to modify skill relaxing settings to run faster on routing blocks that target local agents.

Threshold Expression

This option enables you to use an ECMAScript (or JavaScript) expression to further refine a routing threshold for the specified target(s). Threshold expressions for the **Route Call** block can be used for the following routing types:

- Skill
- Skill Expression
- Agent Group
- Agent
- Campaign Group
- Direct Number

Threshold expressions can contain variables or reference queue-specific values, such as when used with functions like **sdata** or **callage** (see below for more details about using these functions). Strings must be enclosed in single quotes. For example:

- **sdata**(*target*, *statistic*)
  This function applies additional routing criteria based on a statistic. You can use it to specify a target and a predefined statistic, such as **StatPositionInQueue**, **StatCallsWaiting**, or **StatCallsCompleted**.

  For example, this expression would route an interaction to a particular Agent Group when there are more than two agents ready and available:

  ```
  sdata(Group2.GA, StatAgentsAvailable)>2
  ```

  You can also combine multiple **sdata** functions using a pipe (|) as an **OR** operator or an ampersand (&) as an **AND** operator. For example:

  ```
  sdata(VQ_1@.Q,StatServiceFactor)>30 & sdata(VQ_2@.Q,StatServiceFactor)>40
  ```

  For a list of common statistic types used by Designer, see Statistic Types. Note that when adding a statistic to your expressions, you must use the **Stat** prefix, such as **StatTimeInReadyState**, **StatAgentsAvailable**, **StatEstimatedWaitingTime**, and so on.

- **callage()**
  This function returns the age of an interaction in seconds. You can use it when building an expression that specifies a time-based routing condition, such as routing an interaction if it is older than 60 seconds:

  ```
  callage()>60
  ```

> ### Important
>
> - For expressions that reference a variable, the **callage** function is treated as a string. Therefore, you must enclose the function and its operator in single quotes (e.g. 'callage() < ' + *variable*).
> - For routing types that have multiple targets (such as Agent Group or Agent), the script defined in **Threshold Expression** applies to all targets.

For more information about using ECMAScripts in Designer, see Assigning values to variables.

Greetings section

Enable the check box beside **Customer Greeting** and/or **Agent Greeting** to play an audio file to that person while the call is being connected.

For customers, you might use this feature to play a legal disclaimer, or to announce that the call might be recorded (if you use call recording in your contact center). For agents, you might use a variable to announce the customer name or other relevant information.

After you enable **Customer Greeting** and/or **Agent Greeting**, you can select an audio file to play by clicking the icon in the **Announcement** field. This is useful for customer greetings that play a static disclaimer audio file.

Optionally, enable the **Var?** check box to use a variable to dynamically select the audio file. This is useful for agent greetings that use a variable to provide call-specific information, such as the customer name.

Note that:

- The **Customer Greeting** plays continuously until the **Agent Greeting** finishes playing.
- When the **Customer Greeting** and **Agent Greeting** contain different prompt values, each prompt is played to the customer and the agent as specified.
- When only one option contains a value, the same prompt is played to both the customer and the agent.
- If the **Customer Greeting** or **Agent Greeting** cannot be played, the customer is immediately connected to the agent. No greetings are played.

Extensions section

Click **Add Extension Data** to add an extension as a key-value pair to this block. The value type can be a string or integer.

If you want to use a variable for the **Key** or **Value**, select the **Variable** checkbox and then select a variable from the drop-down menu. If the **Value** is an integer, select the **Integer** checkbox.

Important

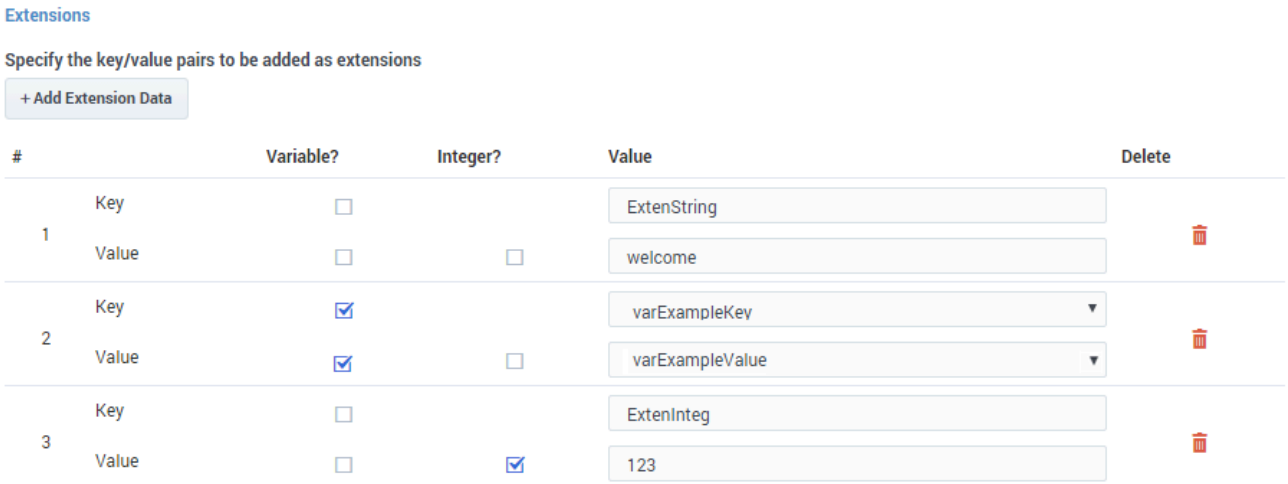
You do not need to enclose extension values in quotes. However, if the quote is part of the value, you must escape the quote character by using a preceding backslash. For example:

- Incorrect: `Joe's Pizza`
- Correct: `Joe\'s Pizza`

Important

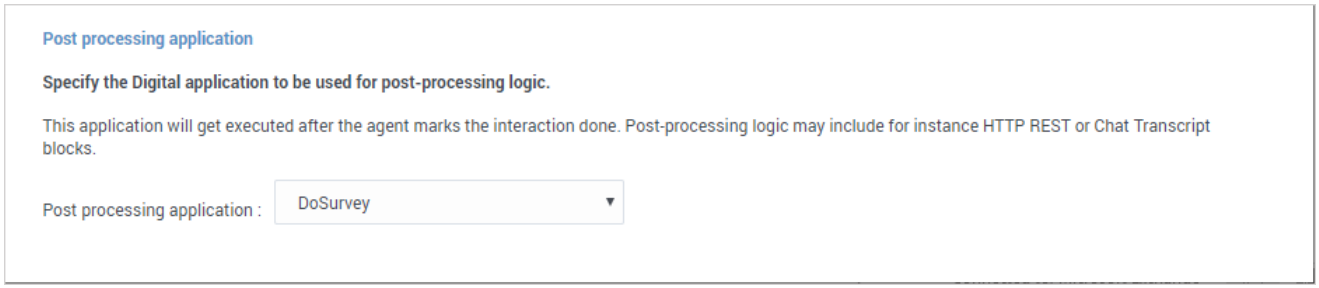
Designer displays an error message if Extension Data is added, but the **Key** and **Value** settings are not defined.

This example shows a few different ways that key-value pairs can be added as extensions:



Post processing application section

Use this section to specify a Digital type application that will apply post-processing after the agent has marked the interaction as done. For example, you could specify an application that provides a survey.



Results tab

Select a variable in the **Store selected agent ID in this variable** drop-down menu to keep track in a specific variable the ID of the agent selected as a result of this **Route Call** block execution. The **SelectedAgent** system variable is transparently assigned this same agent ID value.

You can also select a variable in the **Store the outcome of the Route Call block in this variable** drop-down menu to store the result of this **Route Call** block execution.

Routing Algorithms

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

The Route Call and Route blocks allow you to select certain statistic types that can be used as routing algorithms. (For more information about statistic types, see Statistic Types on the Statistic block page.)

Agent Loading

Selects agents within an agent group and calculates a vector based on three values:

- The number of busy DNs.
- The agents' time in Ready state (the same value as **Time In Ready State**)
- A random number.

Agent Occupancy

Enables the routing engine to route interactions to the least occupied agent (the agent with the lowest occupancy rate). The occupancy rate is determined by the ratio between the time the agent has been busy since last login compared to the agent’s total login time.

Agent Occupancy enables the routing engine to evaluate multiple available agents and select the least occupied agent, which balances the workload among available agents.

This statistic is defined and calculated when the agent logs in and can be used only in statistics that are applied for an agent. This statistic cannot be used with agent groups.

How it works

Consider the following scenario:

- After login, Agent 1 was on a call for five minutes and was in a Ready state for five minutes. His occupancy is 50 per cent (5 / (5+5)).
- After login, Agent 2 was on a call for one minute and was in a Ready state for two minutes. His

occupancy is 33 per cent (1 / (1+2)).

When using **Agent Occupancy**, the routing engine distributes an incoming interaction to Agent 2, as he is the least occupied agent.

Expected Waiting Time

Provides wait-in-queue estimates for the last interaction that entered a virtual queue. This statistic has been designed for the multimedia model, but assumes agents cannot handle more than one simultaneous non-voice interaction at a time.

This statistic applies to all types of media and is calculated and refreshed internally for the last 600 seconds.

Load Balance

Load balancing between routing targets helps to ensure there is an equal distribution of interactions among queues, DNs, agent groups and queue groups. This statistic considers the number of calls distributed between objects, the percentage of busy agents, the expected waiting time, or other routing features.

When this statistic is defined, the routing engine automatically counts calls that are routed to different DNs and queues and adjusts the logic so there is an equal distribution of interactions across the specific DNs or queues.

Most Skilled Agent

This routing algorithm finds the most skilled agent among all available agents who match the skills selected in the Route Call (or Route) block.

How it works

Consider the following scenario:

- A **Route Call** block has three skills selected.
- Agent A has skill levels of 7/7/7 for those skills.
- Agent B has skill levels of 10/5/5, respectively.

The algorithm finds the agent with the highest *combined* skill among the selected skills. Therefore, if both agents are available, the algorithm selects Agent A as the most skilled agent.

> **Important**
> This routing algorithm only works with *skill*-based routing; *skill expression*-based routing is not supported.

Randomly

Controls the mechanism for selecting a default target. If the target list has more than one available target, this option functions as follows:

- If this option is set to **random**, the routing engine picks a target according to its internal rules.
- If this option is set to **FIFO**, the routing engine picks the first available target in the list.

Round Robin

Quickly selects a target from an agent group by applying a round-robin algorithm. The routing engine searches through a list of agents in the target object and returns the first available agent (Agent 1). For the next call, the routing engine bypasses the previously selected agent and picks the next available agent.

When the routing engine reaches the end of the list of agents, it returns to the beginning of the list.

Time in Ready State

Provides the total current time of an agent in Ready state. It is calculated based on availability - that is, there is no activity currently in progress on a particular DN and the agent has placed the DN in Ready state. **Time in Ready State** aggregates the total time for the state and this information can be used to build the target list.

You can use **Time in Ready State** to route interactions to the agent that has been waiting in Ready state for the longest time.

Route Digital Block

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

You can use the **Route Digital** block in the **Assisted Service** phase to route interactions to an agent based on various criteria, such as skills.

You can sequentially place multiple **Route Digital** blocks with different settings, so that if routing fails in one block, your application proceeds to the next block.

When a **Route Digital** block successfully routes the interaction to an agent, the application moves to the **Finalize** phase, ignoring any subsequent blocks in the **Assisted Service** phase.

Click here to watch a video showing an example of skill-based routing for chat interactions.

Routing Tab

Select Routing type

Choose between the following routing options:

Skill based routing with relaxing criteria

Routes the interaction to an agent that has the required skills. If selected, you can choose from the following options:

- **Use system variables *RoutingSkills* and *RoutingVirtualQueue* set already in Menu Options** - Use system variables that were set in a Menu Option block.
- **Specify Skills in this block** - Specify one or more skills and a Virtual Queue to use to route this interaction. If you specified more than one skill, you can choose whether the routing engine considers **any** or **all** of the selected skills.

> **Important**
> This option uses the skill level specified in the **Skill Proficiency level** setting

(documented below). For example, if you set an initial skill level of 8, Designer only routes the interaction to agents with the specified skills that have a level of 8 or greater. You cannot set an individual level for each specified skill.

- **Skill Proficiency level** - Enter the initial and minimum skill levels. The interaction is routed to an agent that has a skill level equal to or higher to the values provided. If you enable **Reduce skill requirements every**, you can have the skill level decremented by a certain amount every *x* number of seconds, until the minimum skill level is reached. This option allows you to expand the group of agents that can receive this interaction if other agents are busy.

Skill expression based routing

Enter a skill expression in the **Skill Expression** tab, or click the drop-down menu to select a variable that specifies a skill expression.

Route to another Application

Route the interaction to the interaction queue of the selected destination application.

Agent Group routing

Route the interaction to an Agent Group.

Agent routing

Route the interaction to agents by using a variable that holds the ID of an agent at runtime. You must use the following format: *agentid*`@`*optional_statserver*`.A`. Example: `1001@StatServer.A`.

Parking Queue

If a digital interaction arrives when the business is closed, send the interaction to a parking queue until the business is open. If selected, you can then specify a variable that tells Designer how long the interaction is to be parked (for example, the variable that holds the number of minutes until the business is open). When regular business hours resume, the interaction is retrieved and processed.

Other Routing Settings

Routing Algorithm

Select which algorithm is used to choose an agent when more than one agent is available. (For more information about the routing algorithms, see Statistic Types on the **Statistic** block page.)

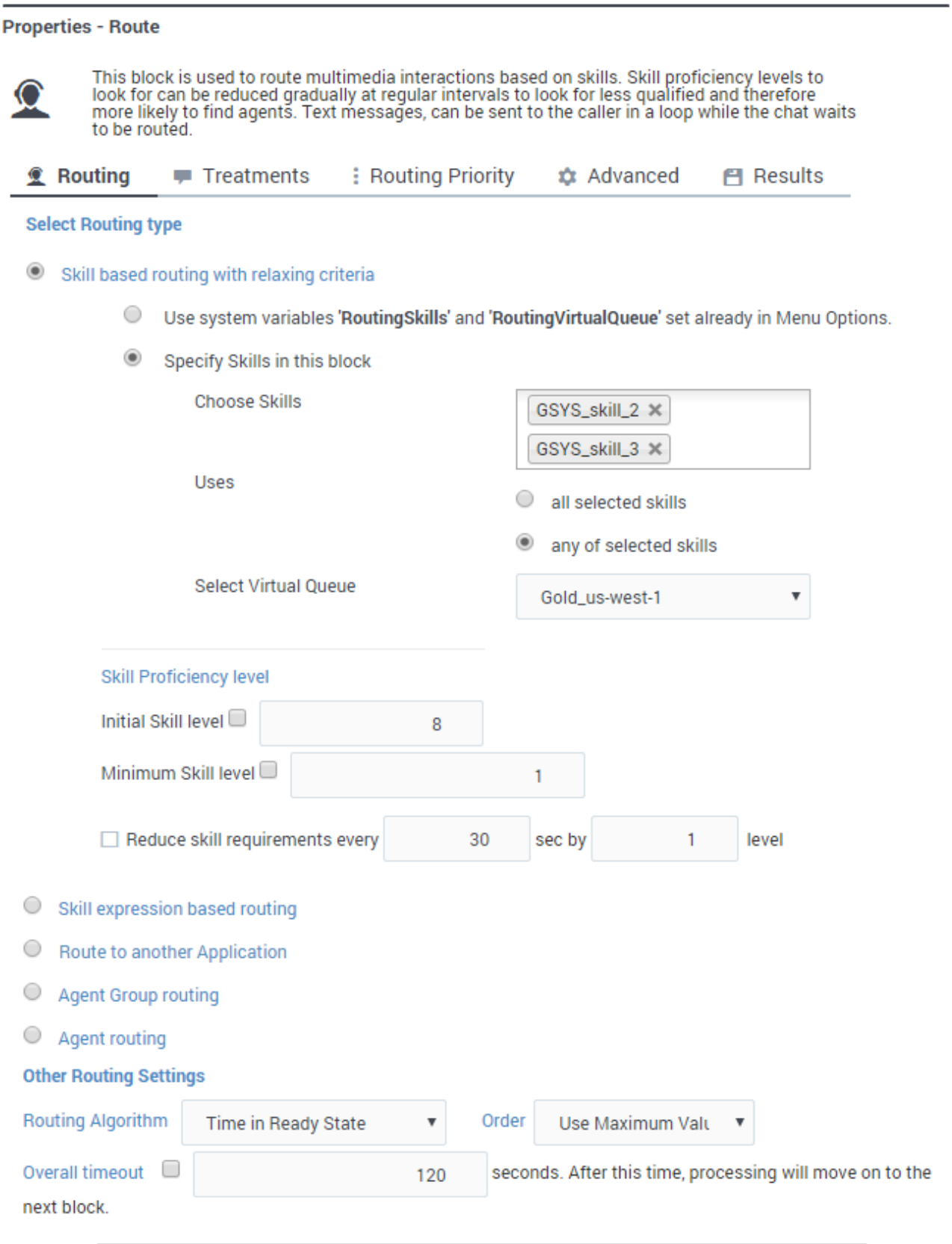
Overall timeout

Enter the maximum time (in seconds) to wait for an agent to be available before moving to the next block. Optionally, you can enable the check box to specify a variable.

Important

System variables **SelectedTarget**, **SelectedVirtualQueue**, **SelectedComponent**, **SelectedTargetObject**, **SelectedAgent**, and **Access** are automatically set when the interaction is routed to an agent and can be used later in the application. Refer to the **Initialize** phase's **System Variables** tab to read a detailed description for each of these variables.

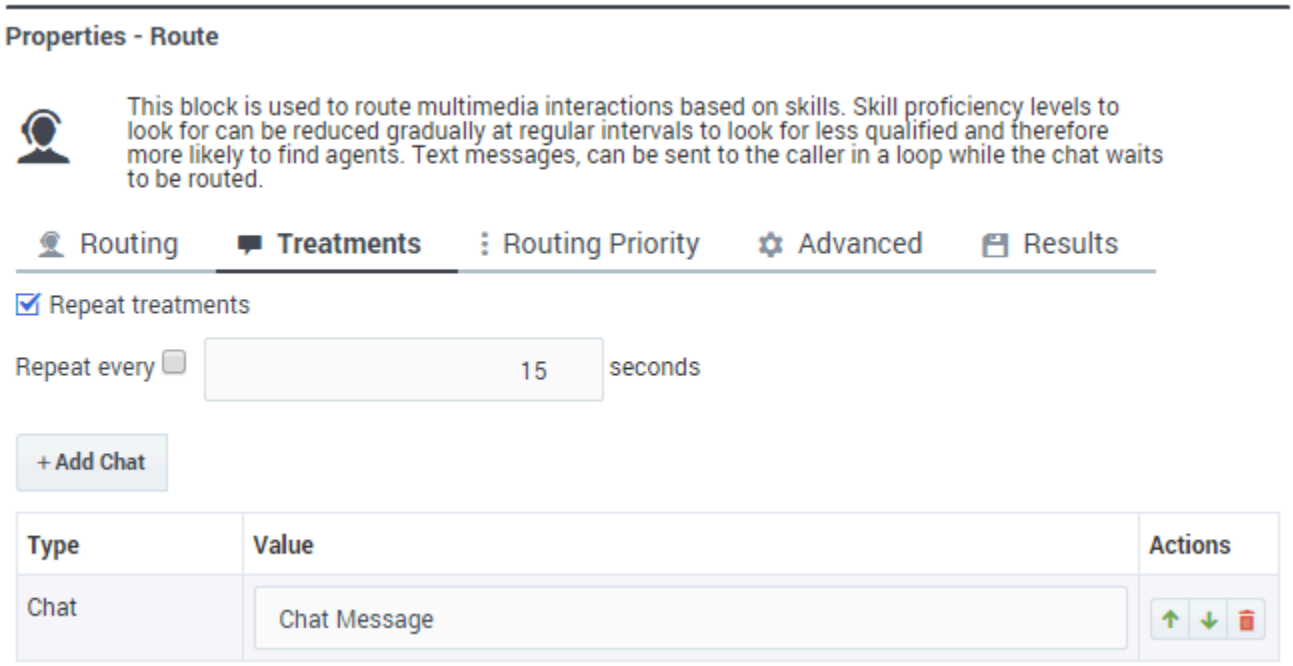
Example


Properties - Route
This block is used to route multimedia interactions based on skills. Skill proficiency levels to look for can be reduced gradually at regular intervals to look for less qualified and therefore more likely to find agents. Text messages, can be sent to the caller in a loop while the chat waits to be routed.
Routing
Treatments
Routing Priority
Advanced
Results
Select Routing type
Skill based routing with relaxing criteria
Use system variables 'RoutingSkills' and 'RoutingVirtualQueue' set already in Menu Options.
Specify Skills in this block
Choose Skills
GSYS_skill_2
GSYS_skill_3
Uses
all selected skills
any of selected skills
Select Virtual Queue
Gold_us-west-1
Skill Proficiency level
Initial Skill level
8
Minimum Skill level
1
Reduce skill requirements every
30
sec by
1
level
Skill expression based routing
Route to another Application
Agent Group routing
Agent routing
Other Routing Settings
Routing Algorithm
Time in Ready State
Order
Use Maximum Valu
Overall timeout
120
seconds. After this time, processing will move on to the next block.

Treatments tab

Use this tab to specify a busy treatment to execute while waiting for an agent to become available. The busy treatment can be a **Chat** message that you can set to repeat at regular intervals.

Example



Routing Priority tab

Use Priority during Routing

Enable this check box to use priority-based routing, which prioritizes your interactions depending on your business requirements.

To prioritize interactions, you must segment interactions and assign the name of that segment to a variable. You must select this variable in the **Lookup Priority table based on this variable** drop-down menu.

You can customize this table with your own segment definitions to fit your business needs. If the specific segment is not found, then the value specified for **Initial priority** is used. Enter a value in **Increment size** to increase the priority of an interaction that remains in a queue over time. The priority increment is defined for each segment, but a default increment is configurable with the **Increment Size** property.

Increment Priority every ___ seconds

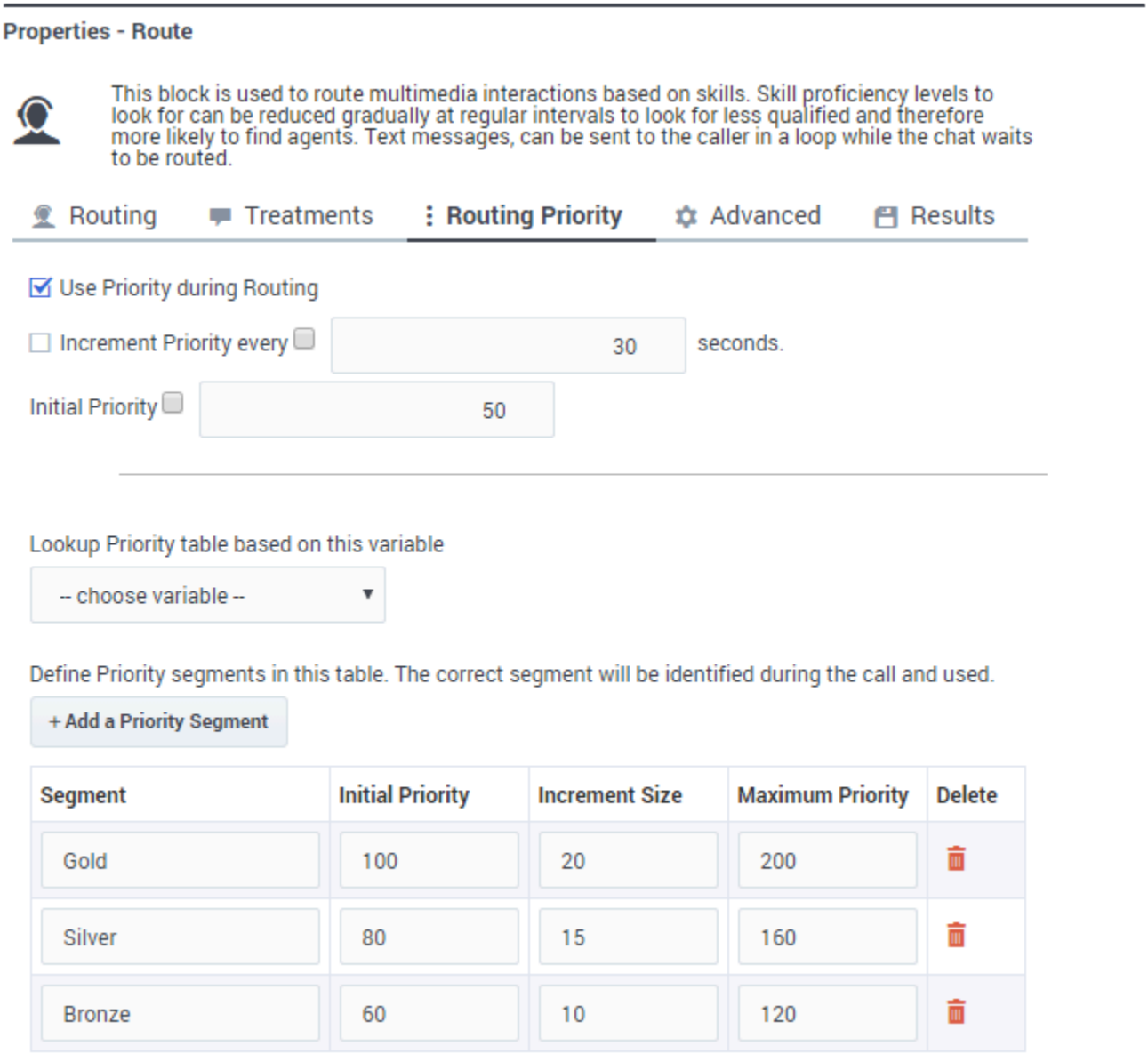
Enable this check box to specify the time interval between priority increments. If you enable the other check box beside the field, you can select a variable that specifies the overall **Routing Timeout** and **Priority Increment Interval** properties.

Limit Priority to

If the **Increment Priority every ___ seconds** option is enabled, you can use this option to set a maximum priority value. For example, if the initial priority is `50`, you can use this option to not let the priority value increase beyond `100`.

If you enable the other check box beside the field, you can select a variable for this option.

Example


Properties - Route
This block is used to route multimedia interactions based on skills. Skill proficiency levels to look for can be reduced gradually at regular intervals to look for less qualified and therefore more likely to find agents. Text messages, can be sent to the caller in a loop while the chat waits to be routed.
Routing
Treatments
Routing Priority
Advanced
Results
Use Priority during Routing
Increment Priority every
30
seconds.
Initial Priority
50
Lookup Priority table based on this variable
-- choose variable --
Define Priority segments in this table. The correct segment will be identified during the call and used.
+ Add a Priority Segment
Segment
Initial Priority
Increment Size
Maximum Priority
Delete
Gold
100
20
200
Silver
80
15
160
Bronze
60
10
120

Advanced tab

Targeting

Clear targets from queue if this block times out

Enable this check box to specify whether the pending request for a target should be kept active or not after exiting this block on timeout. When the request is kept active (check box is disabled), an agent may be selected after the block times out if, for example, an agent with the matching criteria is ready after the block was exited.

Early exit from this block if no agents are logged in

Enable this check box to exit the block if no agents are logged in for the selected routing target (such as Agent or Agent Group, skill expression based, or skill based routing with relaxing criteria).

Threshold Expression

This option enables you to use an ECMAScript (or JavaScript) expression to further refine a routing threshold for the specified target(s). Threshold expressions for the **Route Digital** block can be used for the following routing types:

- Skill
- Skill Expression
- Agent Group
- Agent
- Last Called Agent

Threshold expressions can contain variables or reference queue-specific values, such as *sdata(target, statistic)* or *callage()*. Strings must be enclosed in single quotes. For example:

For more information about using ECMAScripts in Designer, see Assigning values to variables.

> **Important**
> For routing types that have multiple targets (such as Agent Group or Agent), the script defined in **Threshold Expression** applies to all targets.

Route only to local agents

If you have selected **Skill based routing with relaxing criteria** or **Skill expression based routing**, you can enable this option. When enabled, the call is routed to a local agent who matches the target skill.

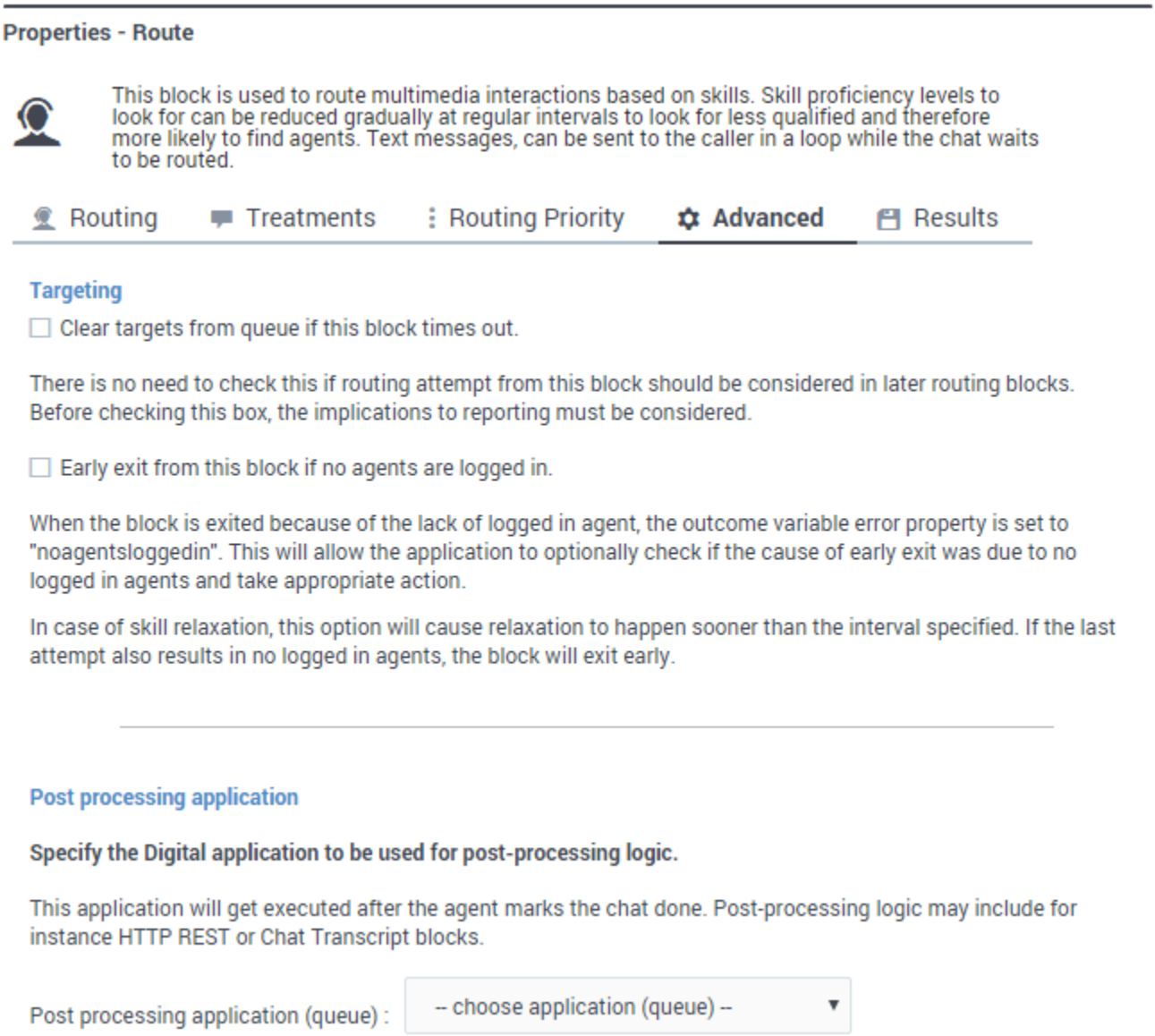
> **Tip**
>
> If you want to route to local agents as the preferred option, but then route to all agents if there are no local agents available with the required skill, you can set up cascaded routing.
>
> Here's a way you could do that:
>
> - Set up the **Route Digital** block with **Route only to local agents** enabled, a short **Overall timeout** property value, and **Clear targets from queue if this block times out** deselected.
> - Then, set up any **Route Digital** blocks that are further down the application flow with **Route only to local agents** not selected.
>
> You can watch this video to see a short demonstration of how to set this up. (The video demonstrates this using a Route Call block, but the steps are the same.)
>
> Link to video
>
> You might also want to modify skill relaxing settings to run faster on routing blocks that target local agents.

In the **Post processing application** section, you can specify the digital application (queue) that will be used for post-processing logic. This application will be executed after the agent marks the chat as *done*.

Post-processing logic may include, for example, HTTP REST or Chat Transcript blocks.

> **Tip**
>
> If the purpose of your post-processing application is to only send a Chat Transcript, Genesys recommends that you use a Terminate block in the post-processing application to prevent it from sending of multiple copies of the transcript.

Example



Results tab

Select a variable in the **Store selected agent ID in this variable** drop-down menu to keep track in a specific variable the ID of the agent selected as a result of this **Route Digital** block execution. The **SelectedAgent** system variable is transparently assigned this same agent ID value.

You can also select a variable in the **Store the outcome of the Route block in this variable** drop-

down menu to store the result of this **Route Digital** block execution.

Videos

This video shows an example of using skill-based routing to route chat interactions.

Link to video

Start Treatment Block

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

You can use the **Start Treatment** block to play uninterrupted audio (a "busy treatment") to callers while their calls are being processed by more than one Route Call or Route Agent blocks.

Typically, busy treatments are played by the Route Call or Route Agent block, but the audio stops playing when the flow moves on to the next block.

- Learn more about busy treatments

> **Important**
> After a busy treatment has been executed at least 10 times, Designer exits the Route Call or Route Agent block and moves to the next block if the average duration of the treatment is less than 1000 ms (for example, due to a missing audio file).

With the **Start Treatment** block, callers won't hear any breaks in the audio as their call is being routed. The audio will continue to play until another treatment is started (for example, the flow reaches another **Start Treatment** block, or **Play Message** is started by one of the routing blocks), the call is routed, or the Assisted Service phase ends.

> **Tip**
> Remember that when you start a new treatment, it immediately stops any treatment that is running.

Module tab

Use this tab to select the Shared Module that will play the audio file.

Is Synced

- If enabled, the **Start Treatment** block will remain active while the treatment is running. The application will not move to the next block in the phase until a condition specified in the Navigation tab causes the application to jump to another block.

> **Important**
>
> When exiting the block on a navigation condition, the treatment keeps playing (like with an asynchronous block) although the **Start Treatment** block is exited.

- If not enabled, the **Start Treatment** block is exited as soon as the treatment starts, and processing moves to the next block in the phase.

> **Important**
>
> When **Is Synced** is not enabled, an asynchronized treatment can continue to play after the **Start Treatment** block has been exited, and will continue playing until a new treatment is started or the Assisted Service phase ends.

Signature tab

This tab displays any **Input Parameters** and **Output Parameters** that are returned by the Self Service Shared Module running the busy treatment.

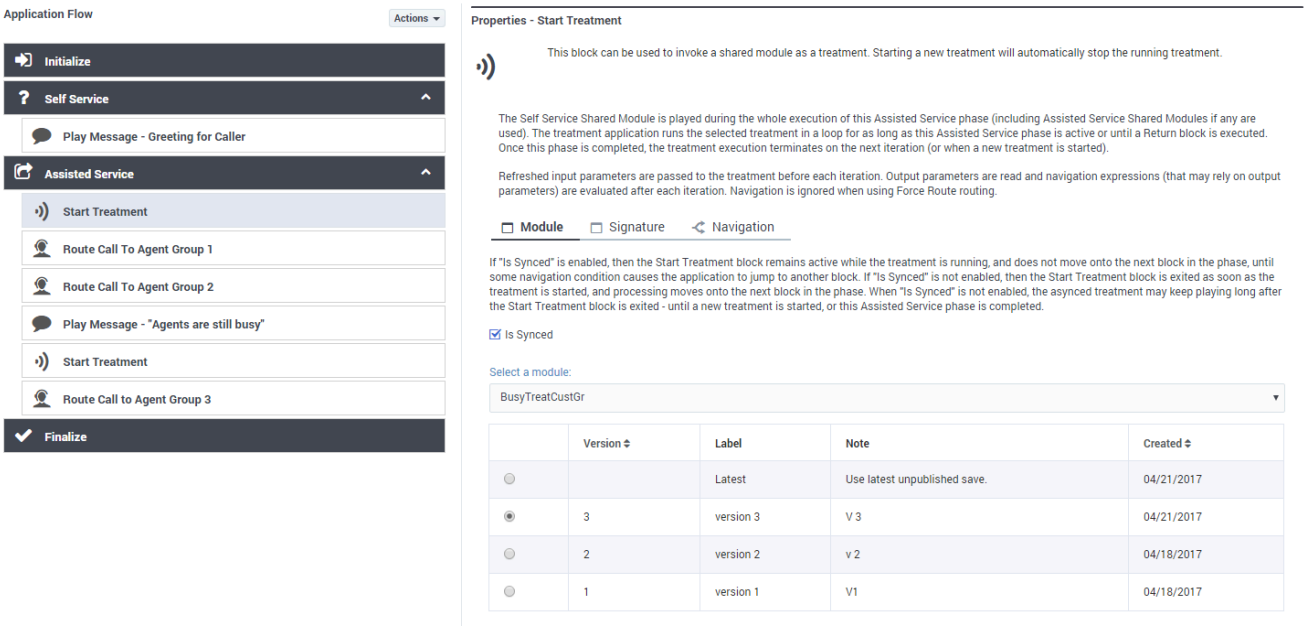
Navigation tab

Use this tab to add a **Condition Expression** that will redirect the application to the selected target block. You can select a target by **Name**, **Type**, or **Description**.

> **Important**
>
> The busy treatment will continue to play to callers during the redirect, as the target block is part of the same Assisted Service phase as the **Start Treatment** block.

Example

Here's a look at how this block can be used in an application flow (click to enlarge):


Application Flow
Actions
Initialize
Self Service
Play Message - Greeting for Caller
Assisted Service
Start Treatment
Route Call To Agent Group 1
Route Call To Agent Group 2
Play Message - "Agents are still busy"
Start Treatment
Route Call to Agent Group 3
Finalize
Properties - Start Treatment
This block can be used to invoke a shared module as a treatment. Starting a new treatment will automatically stop the running treatment.
The Self Service Shared Module is played during the whole execution of this Assisted Service phase (including Assisted Service Shared Modules if any are used). The treatment application runs the selected treatment in a loop for as long as this Assisted Service phase is active or until a Return block is executed. Once this phase is completed, the treatment execution terminates on the next iteration (or when a new treatment is started).
Refreshed input parameters are passed to the treatment before each iteration. Output parameters are read and navigation expressions (that may rely on output parameters) are evaluated after each iteration. Navigation is ignored when using Force Route routing.
Module
Signature
Navigation
If "Is Synced" is enabled, then the Start Treatment block remains active while the treatment is running, and does not move onto the next block in the phase, until some navigation condition causes the application to jump to another block. If "Is Synced" is not enabled, then the Start Treatment block is exited as soon as the treatment is started, and processing moves onto the next block in the phase. When "Is Synced" is not enabled, the asynced treatment may keep playing long after the Start Treatment block is exited - until a new treatment is started, or this Assisted Service phase is completed.
Is Synced
Select a module:
BusyTreatCustGr
Version
Label
Note
Created
Latest
Use latest unpublished save.
04/21/2017
3
version 3
V 3
04/21/2017
2
version 2
v 2
04/18/2017
1
version 1
V1
04/18/2017

Transfer Block

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

> **Important**
> This block is only available for **IVR** type applications. Refer to the Managing Applications page for more information.

You can use the **Transfer** block in the **Self Service** phase to transfer a call to another destination.

You can sequentially place multiple **Transfer** blocks with different settings, so that if a transfer fails in one block, your application proceeds to the next block. When a **Transfer** block successfully transfers the call, the application moves to the **Finalize** phase, ignoring any subsequent blocks in the **Self Service** phase.

Transfer tab

Click the **Destination** drop-down menu to select a target destination for the call.

Next, enter a timeout value for the application to wait for the transfer to proceed before moving to the next block.

Results tab

Select the variable that will store the result of this **Transfer** block execution.

Voice Mail Block

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

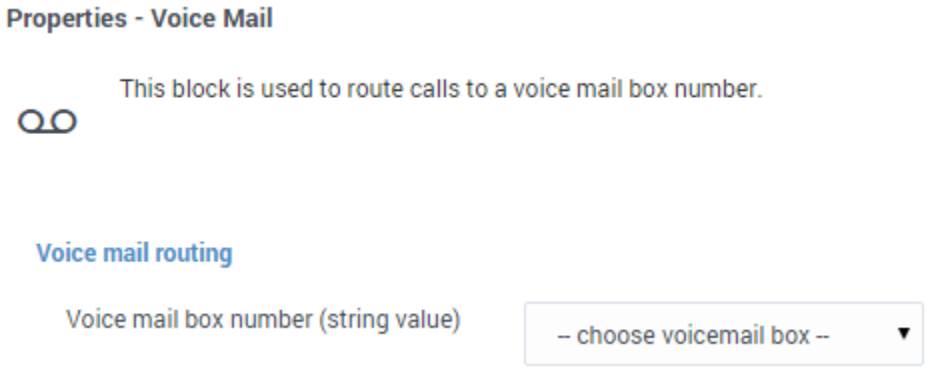
You can use the **Voice Mail** block in the **Assisted Service** phase to route calls to voicemail.

You can sequentially add multiple **Voice Mail** blocks with different settings, so that if routing fails, the next block is used. When a **Voice Mail** block successfully routes the call to voicemail, the application moves to the **Finalize** phase, ignoring any subsequent blocks in the **Assisted Service** phase.

Using this block

Voice mail routing

Select the voicemail number to which you want to route the call.



Predictive Routing Block

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

> **Important**
> You can only use this block if your environment is enabled for Genesys Predictive Routing. Contact your Genesys representative if you have any questions about using this feature.

The **Predictive Routing** block enables a Designer application to retrieve a list of agents that are best equipped to handle a specific type of call and then route the call to the agent with the highest likelihood of a successful outcome.

Typically, you would add this block when you want to improve routing metrics for an application that is using traditional (or "prescriptive") routing methods and has an established amount of historical data that can be used for generating accurate predictors and models.

How it works

Genesys Predictive Routing (GPR) uses machine learning to analyze your accumulated history of agent, customer, and call data to generate scores for each available agent that indicate probable outcomes for incoming calls. Using these scores, you can ensure that calls are routed to agents with the highest likelihood of producing a satisfactory result.

When you add the **Predictive Routing** block to an application, you can select a *Predictor* from the list of Predictors that Designer retrieves from GPR. The Predictor indicates the metric you want to optimize (such as first-contact resolution, sales conversion, and so on). The Predictor also specifies which customer and agent qualities (or *features*) have the most impact on the chosen metric.

When a call arrives, Designer retrieves a list of currently-available agents (ranked according to the parameters specified in the Predictor), and then routes the call to the best available match. Designer continues to periodically retrieve these lists of agents for the duration of the session—even if the application has moved on from the Predictive Routing block to the next routing block—until the call is finally routed or the session ends. If the "predicted" outcome differs from the actual outcome, Predictive Routing "learns" from this result, improving the future accuracy of its predictions.

For more information about setting up Predictors, see Creating and Updating Predictors in the *Genesys Predictive Routing Help*.

> **Important**
>
> Only Predictors with active Models are available for selection. See Configuring, Training, and Testing Models in the *Genesys Predictive Routing Help* for more information.

Using this block

This block can be placed in the Assisted Service phase of an application. For the block to take effect, you'll also need to enable predictive routing in the application settings.

> **Tip**
>
> Genesys recommends that you use this block in a mixed-use setting, where Predictive Routing is attempted first (say, for a period of 2 seconds) and then traditional routing acts as a fallback method of routing the call. This also helps to train the models used by Predictive Routing, which can effectively "learn" from the traditional routing outcomes.

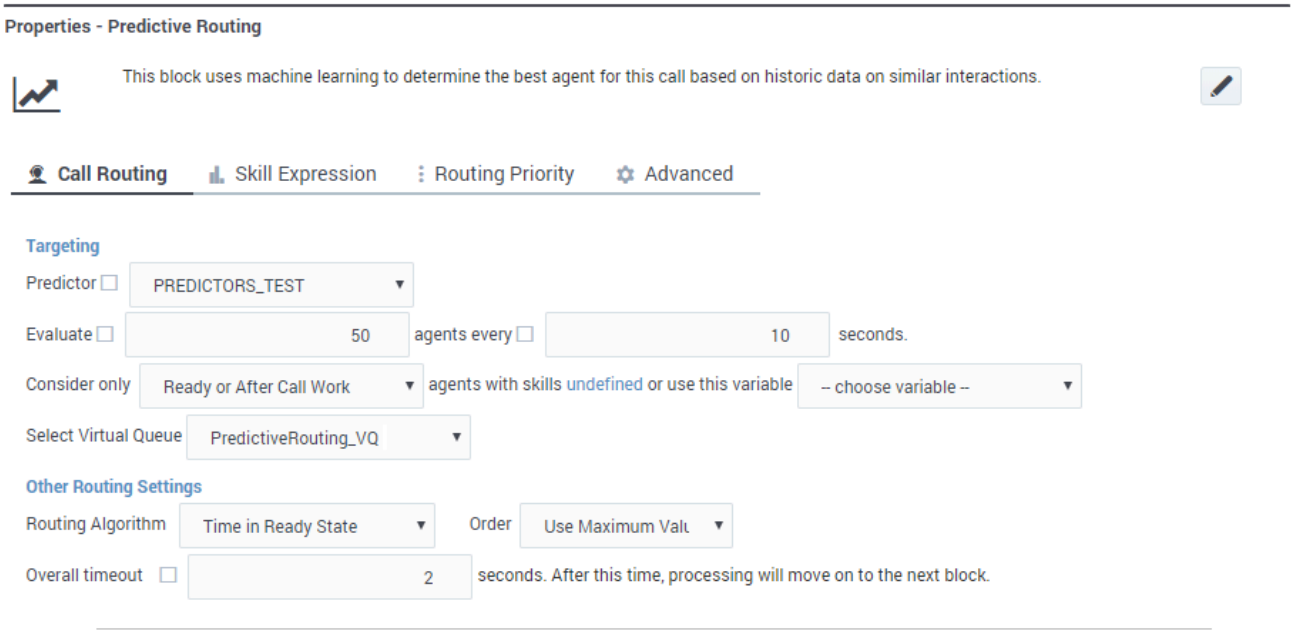
Call Routing tab

In the **Targeting** section, select the **Predictor** you want Designer to use. You can also choose to specify this value using a variable.

In the **Evaluate** field, enter the number of agents to submit as a batch for routing, and how often to submit each batch. For example, you might choose to submit a batch of 50 agents every 10 seconds.

You can then specify additional criteria to be used when evaluating possible agents, such as their state or skills, and select the **virtual queue** for routing.

In the **Other Routing Settings** section, you can specify which routing algorithm to use and the block **timeout** value.



Skill Expression tab

Use this tab to specify a skill expression for the **Predictive Routing** block. The skill expression consists of a list of skills for which you must individually set an operator and an integer value.

> ### Important
> When you configure a skill expression in the **Skill Expression** tab of the **Predictive Routing** block, the **Consider Only** option (for agent status) is enabled on the **Call Routing** tab. Also, the built-in agent status filter is ignored if you use a variable-defined skill expression.

For more information about setting up skill expressions, see the Skill Expression settings for the **Route Call** block.

Routing Priority tab

For information about these settings, see the Routing Priority settings for the **Route Call** block.

Advanced tab

Greetings section

Enable the check box beside **Customer Greeting** and/or **Agent Greeting** to play an audio file to that person when the interaction is connected.

For more information, see the Advanced tab settings for the **Route Call** block.

Extensions section

Click **Add Extension Data** to add an extension as a key-value pair to this block. The value type can be a string or integer.

For more information, see the Advanced tab settings for the **Route Call** block.

Predictive Settings section

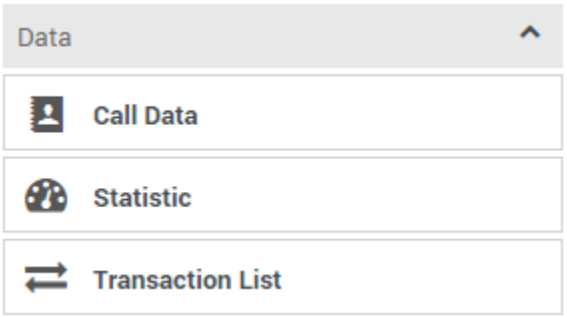
If desired, you can specify key-value pairs that will provide additional context to the prediction service.

> **Tip**
>
> The **Predictive Routing** block does not have an option for treatments, but you must specify a treatment to play during routing. Genesys recommends that you do the following:
>
> - Place a Start Treatment block and a Self Service module ahead of this block to provide a treatment to play during routing.
> - In the settings for the **Start Treatment** block, disable the **Is Synced** option so that Designer will start the treatment and then immediately move on to the next block while the treatment keeps playing. This ensures that the same treatment continues for any prescriptive routing blocks that follow.

Data Blocks

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

These blocks are used for various data handling functions, such as retrieving information about an interaction that was created before the application started running, saving some information about a customer (or an interaction) to a database so that it can be processed later by another application, specifying keys to be read from Call Data and stored into application variables, or using statistics in the segmentation logic to route calls.



Use the links below to learn more about each block.

Call Data

Reads or writes data that can be accessed from both inside and outside an application.

Used in: **Initialization**, **Self Service**, **Assisted Service**

Statistic

Gets statistic values for queues or agent groups.

Used in: **Initialization**, **Assisted Service**

Transaction List

Assigns transaction list values to variables.

Used in: **Initialization**, **Assisted Service**

Call Data Block

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

You can use the **Call Data** block to read and write data that can be accessed from both inside and outside the application.

You can add the block to the **Self Service** or **Assisted Service** phase, but the block can behave differently depending on the phase in which it is used. These differences are noted where applicable, so be sure to review the information carefully when setting up the block.

The data format consists of **key-value pairs (KVP)**. That is, for every **key** in the call data, you can associate a **value** with it. This value can be represented as an integer, character string, binary (boolean) type, or as a list of KVPs (see an example of how to do this).

While data in application variables only exist within the application, information stored in call data can persist and be retrieved even after the application finishes running. This is useful for retrieving information about the call that was created before the application started running, or for saving some information about either a caller or a call to a database so that it can be processed later by another application.

There are certain variable names used by the Designer application that must not be updated by the **Call Data** block. See Restricted Variable Names for a list of these variables.

Syntax requirements

When entering values, you must escape quote characters in values by using a preceding backslash. For example:

- **Correct:** `Joe\'s Pizza`
- **Incorrect:** `Joe's Pizza`

The **Call Data** block only accepts *string* type keys and values, so you do not need to enclose these values in single quotes.

Read Data tab

Use the **Read Data** tab to specify keys to be read from Call Data and stored into application variables. You can toggle the key being read between a variable and a string.

Edit Data tab

Self Service phase

If this block is in the **Self Service** phase, use the **Edit Data** tab to specify key-value pairs to be written to the call data (see an example).

When used in **Self Service**, the **Call Data** block has the following restrictions:

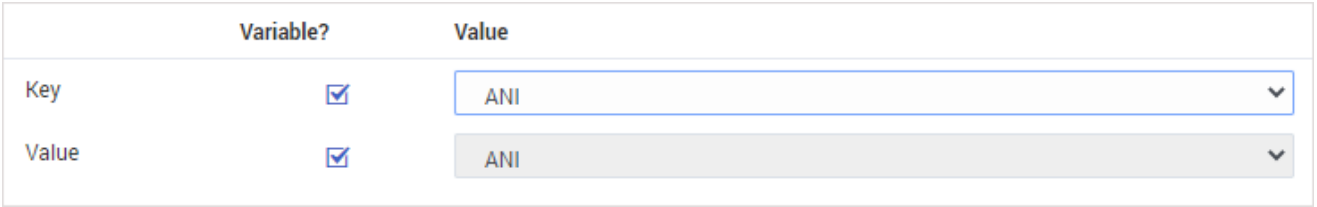
- Non-variable key names must follow standard JavaScript rules for variable names, even if the keys are not variables. For example, key names must only contain alphanumeric characters (no spaces) and not match any of the restricted variable names.
- Call data cannot be deleted. The **Edit Data** tab does not have the **Remove** option, as is available in the **Assisted Service** phase.

When a key-value pair list is used as a value in the user data for a **Call Data** block in the **Self Service** phase, the user data is attached to the call, but the platform encodes any special characters that appear in the user data, such as single quotation marks ('), double-quotation marks ("), backslashes (\), commas (,), colons (:), semicolons (;), and so on.

You can specify *literal* or *variable* values (or a combination) for each KVP, using one of the following options:

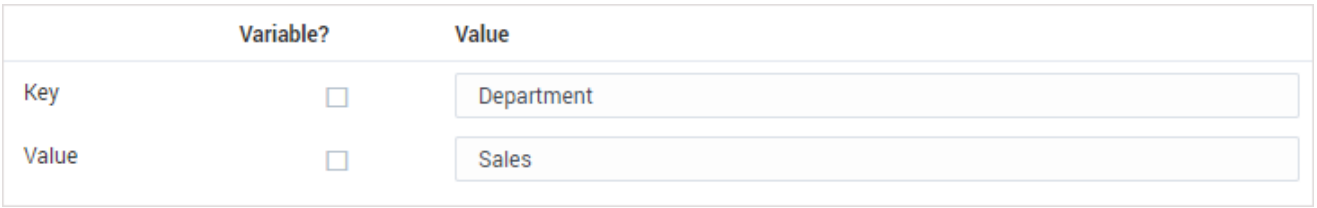
Variable Key + Variable Value

If you enable the variable checkbox for **Key**, the variable checkbox for **Value** is automatically selected and its dropdown menu is disabled (i.e. grayed out). Whichever variable you select for **Key** is also applied to **Value**. In this case, Designer will update the **Key** in the call data with the name of the variable and the **Value** with the value that is stored in it:



Literal Key + Literal Value

You can also enter literal values for both **Key** and **Value**. In this case, the entries are not required to match; each value is written to the call data as specified:



Literal Key + Variable Value

If using a literal value for **Key** and specifying a variable for **Value**, make sure that the name of the variable you select does **NOT** match the entry provided for **Key**:

Otherwise, the call data won't be updated correctly.

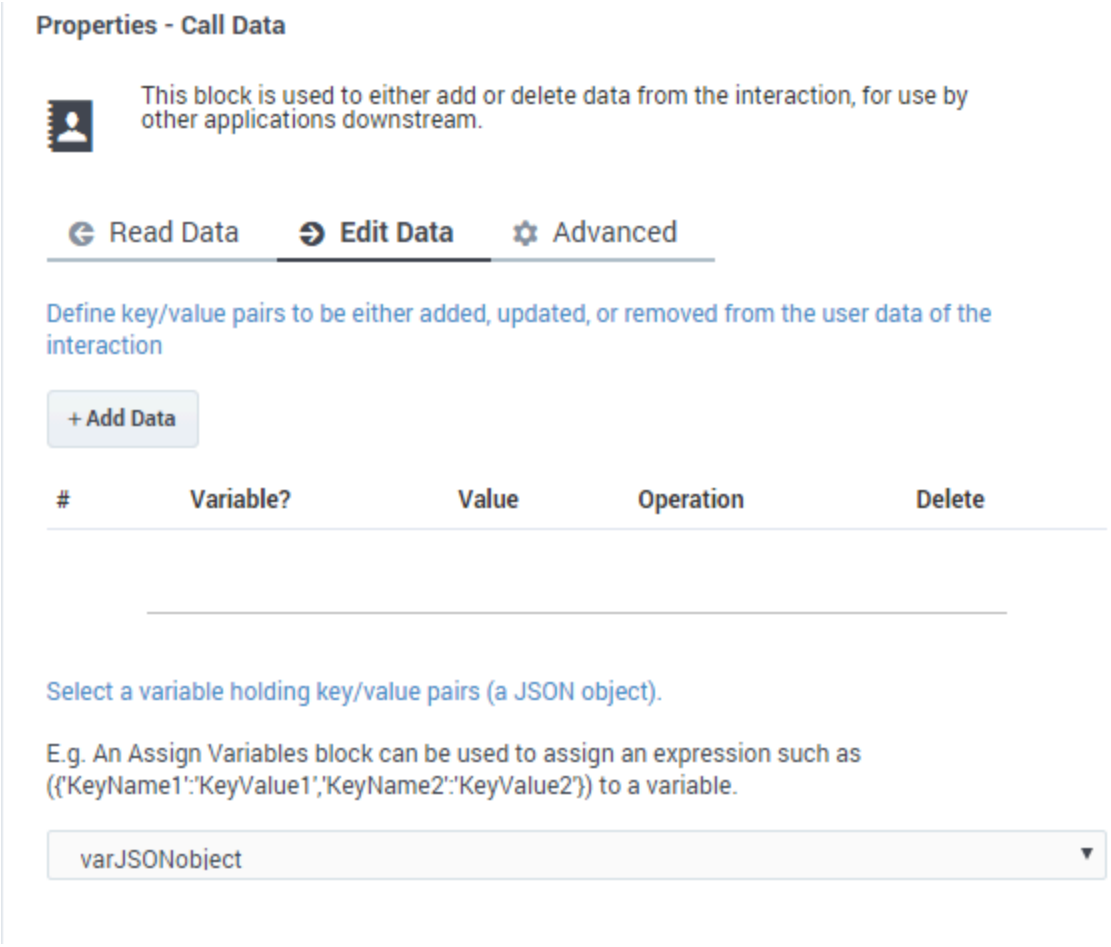
Assisted Service phase

If this block is in the **Assisted Service** phase, use the **Edit Data** tab to specify **key-value pairs** to be written to the call data, either by adding data with new keys (**Add/Update** operation), updating data for existing keys (**Add/Update** operation), or deleting data for existing keys (**Remove** operation):

- **Add/Update** - Add or update data with the specified key. This operation automatically adds the key-value if the specified key does not yet exist in the Call Data, or, if the provided key does exist in the Call Data, it automatically updates data for the key. You can toggle both the key and the value independently between a variable and a string.
- **Remove** - Provide the key for data you want to delete, which you can toggle between a variable and a string. At runtime, if the key you try to delete does not exist, nothing happens.

For operations that involve large amounts of key-value pairs (i.e. 30 or more), see Adding or Removing Large Amounts of Key-Value Pairs.

You can also add a list of key-value pairs by selecting a variable that holds key-value pairs in a JavaScript object.



Advanced tab

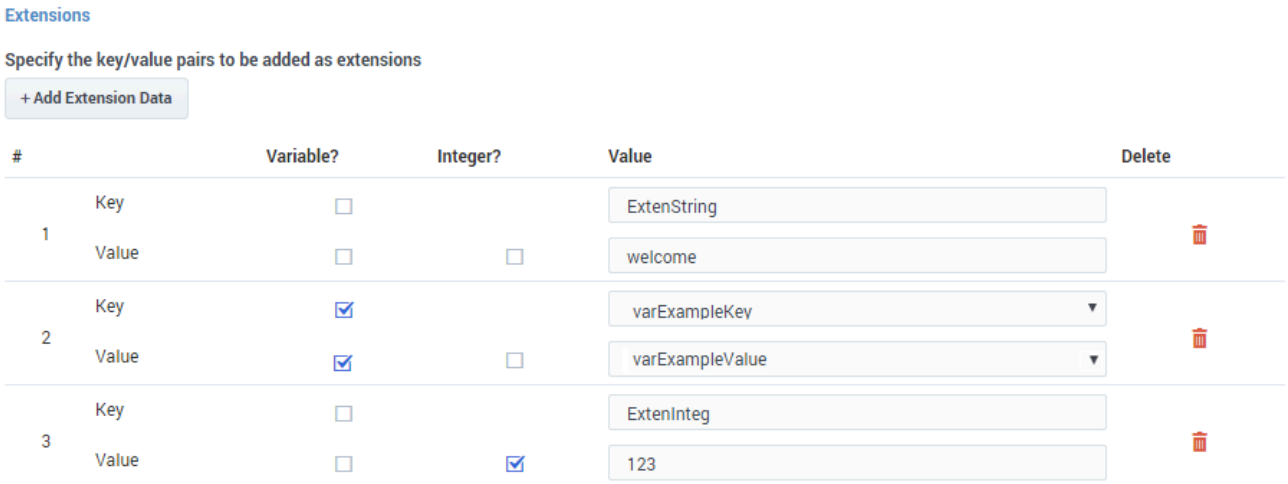
If this block is in the **Assisted Service** phase, use the **Advanced** tab to add an extension as a key-value pair to the key. Click **Add Extension Data** to add an extension as a key-value pair to this block. The value type can be a string or integer.

If you want to use a variable for the **Key** or **Value**, select the **Variable** checkbox and then select a variable from the drop-down menu. If the **Value** is an integer, select the **Integer** checkbox.

You do not need to enclose extension values in quotes. However, if the quote is part of the value, you must escape the quote character by using a preceding backslash (see Syntax requirements.)

> **Important**
> Designer displays an error message if **Extension Data** is added, but the **Key** and **Value** settings are not defined.

This image shows a few different ways that key-value pairs can be added as extensions:



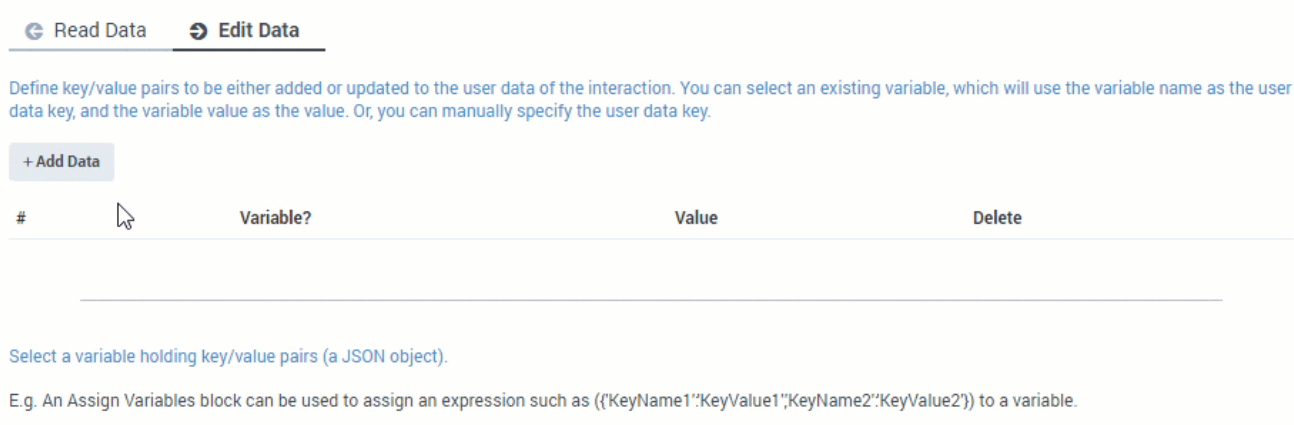
Examples

Self Service

For this example, we want to use the **Call Data** block to specify a key-value pair to be written to the call data. We've created a variable called **CustomerSegment** that Designer will use to store the segment detail about the customer (for example, **Bronze**) and we want to associate its value with the **segment** key in the call data.

On the **Edit Data** tab, we'll click **Add Data** to specify a new key-value pair. For the **Key** field, we'll enter **segment**. For the **Value**, we'll enable the **Variable** checkbox and select the **CustomerSegment** variable we created earlier.

Watch:

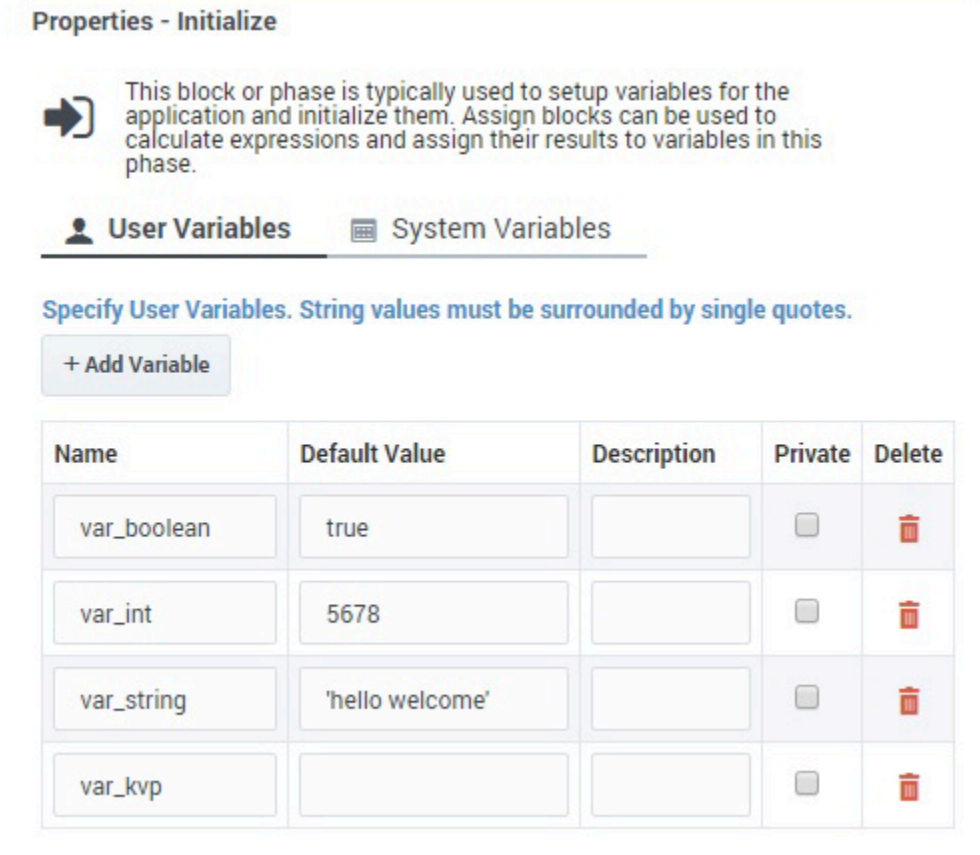


For more information about using the **Edit Data** tab in the Self Service phase, see Self Service phase.

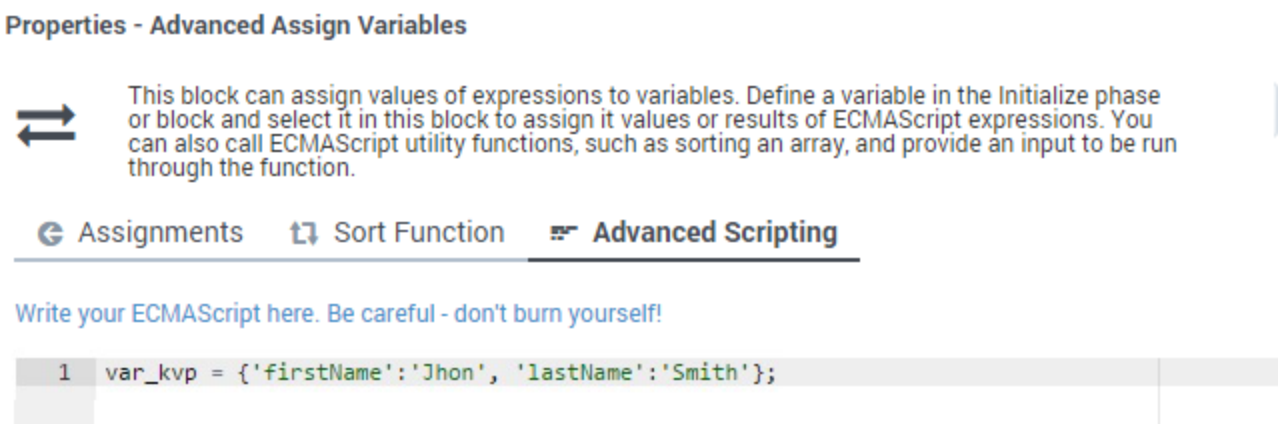
Assisted Service

This example shows how you can assign the value of a key-value pair in the **Call Data** block to a variable in the **Assisted Service** phase.

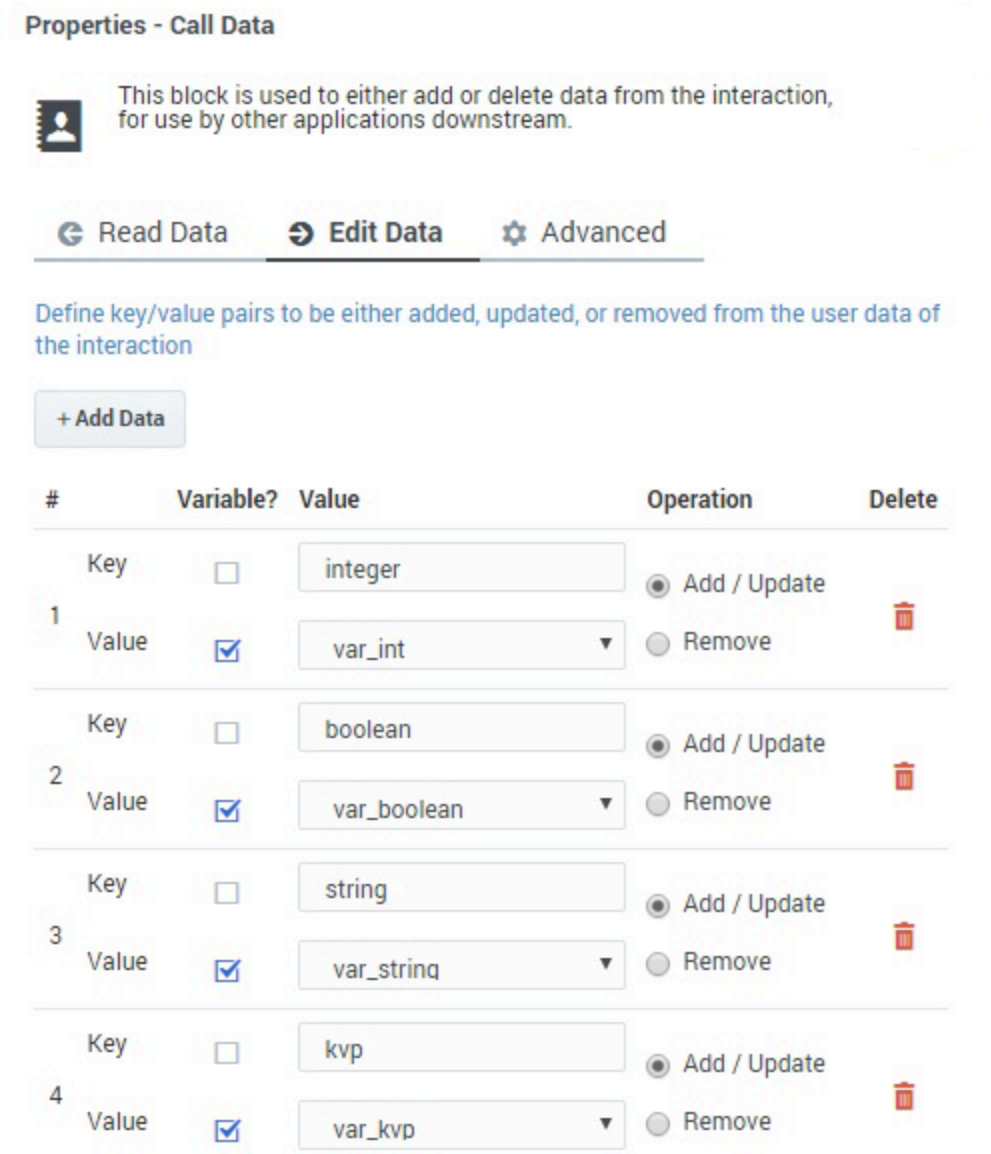
First, initialize your variables as various data types (integer, character string, binary [boolean], and so on), and create a variable for your KVP (in this example, we are using *var_kvp*):



Next, initialize a variable as a list of kv-pairs using the Assign Variables block. This example shows this using an ECMAScript on the **Advanced Scripting** tab:



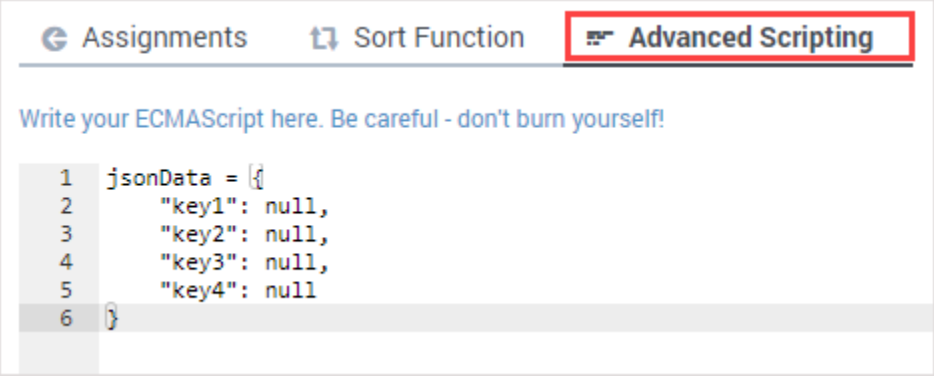
Finally, add the user data as a kv-pair in the Call Data block:


Properties - Call Data
This block is used to either add or delete data from the interaction, for use by other applications downstream.
Read Data
Edit Data
Advanced
Define key/value pairs to be either added, updated, or removed from the user data of the interaction
+ Add Data
# Variable? Value Operation Delete
1 Key integer Add / Update
Value var_int Remove
2 Key boolean Add / Update
Value var_boolean Remove
3 Key string Add / Update
Value var_string Remove
4 Key kvp Add / Update
Value var_kvp Remove

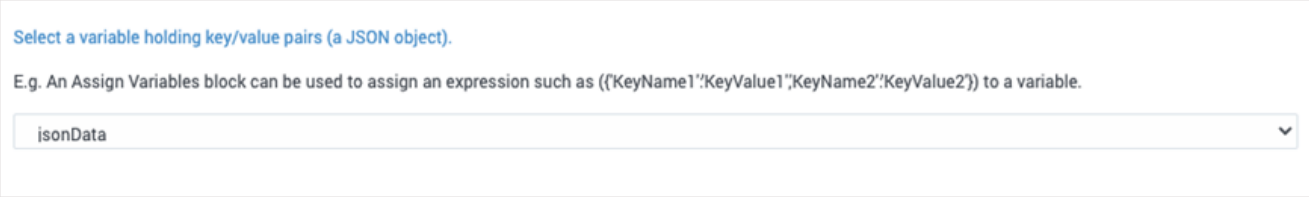
Adding or Removing Large Amounts of Key-Value Pairs

If you want to add or delete several (i.e. 30 or more) key-value pairs, you can use the Advanced Scripting tab of an **Assign Variables** block to create a JSON object that lists each of the keys you want to modify.

For example, to remove a large amount of key-value pairs, create a script that assigns each key a 'null' value. In this example, we've defined an object called **jsonData** and assigned each of its keys a value of **null**:



If you wanted to add/update the keys, simply use the script to assign the desired values to each key. You can then reference this JSON object as a variable in the **Call Data** block to perform the desired operation:



Note that the operations are processed in smaller batches, to prevent too many requests from being processed at one time.

> **Tip**
> This method can be used in conjunction with the standard key-value pair operations described in Edit Data.

Restricted Variable Names

The following variable names are used by the Designer application and must not be updated by the **Call Data** block.

- **_CB_N_CUSTOMER_ABANDONED_WHILE_WAITING_FOR_AGENT**
- **_CB_SERVICE_ID**
- **_CB_T_CALLBACK_ACCEPTED**
- **_CB_T_CUSTOMER_CONNECTED**
- **_CB_T_SERVICE_START**
- **_COMPLETE_CS**
- **_SEND_FINAL_UE**
- **CustomerSegment**
- **EXECUTION_MODE**
- **GMS_UserData**
- **gvp-tenant-id**
- **gsw-ivr-profile-name**
- **GSYS_IVR**
- **GSYS_SystemApplicationDisposition**
- **IApplication**
- **IApplicationVersion**
- **IPurpose**
- **orsurl**
- **orssessionid**

Warning

The addition, updating, and removal of Call Data do not happen instantaneously while the application is running. The application starts the operation to edit the Call Data, and then continues to the next block, without waiting for confirmation that the Call Data is successfully modified.

Thus, there is no guarantee that the "write" operation has finished—a subsequent "read" operation could potentially give you either an old value or an updated value on different runs through the application.

For this reason, Genesys recommends that you do not write a **key-value pair** and then attempt to read back the same **key** from the Call Data within an application. If you need to access a Call Data value that was already edited within the application, Genesys recommends that you use the corresponding application variable.

Statistic Block

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

You can use the **Statistic** block in the **Assisted Service** phase to get the statistic value on queues or agent groups. These statistics can be used in the segmentation logic to route calls.

Using this Block

Click **Add Assignment** to add a statistic.

Select a variable in the **Variable** drop-down menu.

Click the **Type** drop-down menu to select the type of statistical value: current (**default**), **minimum**, **maximum**, or **average**.

Click the **Statistic** drop-down menu to select a statistic. For more information, see Statistic Types.

Click the **Object** drop-down menu to select the Agent Group or Queue for which the statistic is needed. When using a variable, the variable value must be a string in the form `ObjectName@.Type`, where the type can be one of `Q` (Queue) or `GA` (Agent Group), (for example, `Billing@.Q`).

Statistic Types

The following statistic types are available on the Statistic block:

Agent Loading

This statistic selects agents within an agent group and calculates a vector based on three values:

- The number of busy DNs.
- The agents' time in Ready state (the same value as the **Time In Ready State** statistic)
- A random number.

How It Works

The following assumes you are routing to multiple agents who are available in a group, and the routing engine must select one agent. The engine always selects an agent in the group according to **Agent Loading** statistics. It does this as follows:

- Evaluating the number of busy DNs and selecting the agent with the lowest number of busy DNs.
- If the number of busy DNs is equal among the agents, the routing engine selects the agent with the longest time in Ready state.
- If time in Ready state is equal among the agents, the routing engine uses a random number.

> **Important**
> The **Agent Loading** statistic works for agents only - it is not used for Agent Groups.

Agent Loading Media

This statistic is similar to **Agent Loading**, but it also considers other media types, such as chat and email channels.

The **Agent Loading Media** statistic ensures that the routing engine always distributes interactions evenly among agents.

Agent Occupancy

This statistic enables the routing engine to route interactions to the least occupied agent (the agent with the lowest occupancy rate). The occupancy rate is determined by the ratio between the time the agent has been busy since last login compared to the agent’s total login time.

The **Agent Occupancy** statistic enables the routing engine to evaluate multiple available agents and select the least occupied agent, which balances the workload among available agents.

Agent Occupancy is defined and calculated when the agent logs in and can be used only in statistics that are applied for an agent. This statistic cannot be used with agent groups.

How it works

Consider the following scenario:

- After login, Agent 1 was on a call for five minutes and was in a Ready state for five minutes. His occupancy is 50 per cent (5 / (5+5)).
- After login, Agent 2 was on a call for one minute and was in a Ready state for two minutes. His occupancy is 33 per cent (1 / (1+2)).

When using the **Agent Occupancy** statistic, the routing engine distributes an incoming interaction to Agent 2, as he is the least occupied agent.

Agent State

This statistic provides the current status of agents.

How it works

You can use the **Agent State** statistic when you want to make a routing decision based on the status of your agents (for example, the number of agents logged out, ready, or not ready).

> **Important**
> To use this statistic, the application developer must first create and format the variable value for agent ID (for example, `<agentid>@.A.`)

Agents Available

This statistic provides the current number of agents in Ready State within one or more agent group at any given point of time.

How it works

You can use the **Agents Available** statistic when you are routing to multiple agent groups and you want to make a routing decision based on agent availability in each group.

Agents Busy

This statistic provides the number of busy agents in an agent group at any given point of time.

How it works

You can use the **Agents Busy** statistic when you are routing to multiple agent groups and you want to make a routing decision based on the number of busy agents in each group.

Agents in Queue Login

This statistic provides the current number of agents who are logged into a queue. This applies to agents logged into a virtual queue or an ACD queue.

How it works

You can use the **Agents in Queue Login** statistic when you are routing an interaction and you want make a routing decision based on the number of agents logged into one or more specific queues.

Agents in Queue Ready

This statistic provides the current number of agents who are logged into a queue and in Ready state. This applies to agents logged into a virtual queue or an ACD queue.

How it works

You can use the **Agents in Queue Ready** statistic when you are routing an interaction and you want to make a routing decision based on the number of agents who are logged into one or more specific queues and are also in Ready state.

Agents Total

This statistic provides the total number of agents who are logged into an agent group.

How it works

You can use the **Agents Total** statistic when you are routing an interaction and you want to make a routing decision based on the number of agents who are logged into a specific agent group.

> **Important**
>
> Capacity Rules might affect the number of available agents that is reported by the **Agents Total** statistic. For example, if an agent is already handling the maximum number of interactions, as defined by a Capacity Rule, then the **Agents Total** statistic does not include this agent.

Calls Answered

This statistic provides the total number of calls answered by an agent or agent group.

How it works

You can use the **Calls Answered** statistic when you are routing an interaction to multiple agent groups and you want to make a routing decision based on the total number of calls answered by a specific agent group and pick the agent group with the lower total.

Calls Completed

This statistic provides the total number of calls that have been completed by an agent or agent group. This includes all types of calls (inbound calls, outbound calls, internal calls, and so on).

How it works

You can use the **Calls Completed** statistic when you are routing an interaction to multiple agent groups and you want to make a routing decision based on the total number of calls that have been completed by a specific agent group and pick the agent group with the lower total.

Calls in Queue

This statistic provides the current number of calls that are waiting in a specific queue. This could be an ACD queue or a virtual queue.

How it works

You can use the **Calls in Queue** statistic when you are routing an interaction to multiple queues and you want to make a routing decision based on the total number of calls in a specific queue and pick the queue with the lower total.

Calls Waiting

This statistic provides the current number of interactions waiting for a specific target. It takes into account all the interactions that are waiting for the targeted skill level, both directly and through groups. This statistic applies to the following objects:

- Agent.
- Agent Group.
- Destination Label.
- Place.
- Place Group.
- Queue (virtual and ACD).
- Queue Group.
- Routing Point (virtual and ACD).

How it works

You can use the **Calls Waiting** statistic when you are routing an interaction to multiple targets and you want to make a routing decision based on the total number of calls waiting for a specific target and pick a target with the lower total.

Estimated Waiting Time

This statistic provides the estimated wait time for queued calls, based on the average talk time and average answering time. This statistic applies to queues such as ACD queues, virtual queues, and routing points.

> **Important**
> This statistic does not take into account calls that are in transition.

Stat Server calculates the **Estimated Waiting Time** statistic according to the formula below:

```
LB =

If (ALI= 0) /*no agents at all, wait forever*/ 10,000,000,000

Else if (AR <= CIQ) /*no available agents, use expected waiting time*/AHT * (CIQ - AR +1)/ALI

Else /*enough agents, waiting time is 0, use minus occupancy*/ – (AR - CIQ)/ALI
```

```
Where:
LB or LB(Q) = calculated StatLoadBalance
CIQ or CIQ(Q) = calls waiting in queue
AR or AR(Q) = number of ready agents logged into the queue
ALI or ALI(Q) = number of all agents logged into the queue
AHT or AHT(Q) = the average handling time for agents logged into the queue
```

Expected Waiting Time

Similar to the **Estimated Waiting Time** statistic, the **Expected Waiting Time** statistic also provides wait-in-queue estimates for the last interaction that entered a virtual queue. This statistic has been designed for the multimedia model but assumes agents cannot handle more than one simultaneous non-voice interaction at a time.

This statistic applies to all types of media and is calculated and refreshed internally for the last 600 seconds.

Load Balance

Load balancing between routing targets helps to ensure there is an equal distribution of interactions among queues, DNs, agent groups and queue groups. This statistic considers the number of calls distributed between objects, the percentage of busy agents, the expected waiting time, or other routing features.

When this statistic is defined, the routing engine automatically counts calls that are routed to different DNs and queues and adjusts the logic so there is an equal distribution of interactions across the specific DNs or queues.

Position in Queue

This statistic provides the exact position of an interaction in a queue.

Interactions can be moved based on their position in the queue. Interactions of high value can be moved to a different queue to provide a quicker response.

Service Factor

This statistic is a percentage/interval pair that specifies a certain percentage of interactions must be handled in a certain period of time (for example, 80 percent of interactions in 20 seconds). The statistic is applied against an agent group.

The routing engine calculates the **Service Factor** at an interval of every 50 interactions or 30 seconds. Based on this information, the routing engine decides whether to expand or contract the current set of agents available for the next interval.

The internal algorithm within the routing engine calculates and manages this scenario based on three values:

- SLReal = The percentage of calls distributed in *Y* seconds.
- SLWarn = The percentage of calls distributed in *3/4 * Y* seconds.

- AWT = The average waiting time for distributed calls.

The routing engine uses SLReal, SLWarn, and AWT to determine whether to adjust the current group of target agents as follows:

- If SLReal equals the **Service Factor** percentage, the routing engine makes no change to the current working agent group.
- If SLReal falls below the specified service factor (*SLReal* < *X*), the routing engine compares the current AWT with the AWT previously measured.
- If the current AWT is less than the previous AWT, the **Service Factor** is improving. The routing engine assumes that the service factor will continue improving and makes no change to the current working agent group.
- If the current AWT is greater than the previous AWT, the routing engine adds agents to the working agent group according to the formula *1/4*(N-M)*, where *N* equals the number of selected agents and *M* equals the ideal set of agents.
- If the **Service Factor** level is acceptable but SLWarn is less than the percentage that should be achieved in *3/4 * Y* seconds for *X* per cent that was set as the **Service Factor**, the routing engine initiates one of the following preventive actions:
  - If the current AWT is greater than or equal to the previous AWT, the routing engine adds agents to the working agent group according to the formula *1/8 * (N-M)*.
  - If the current AWT is less than the previous AWT, the routing engine tries to reduce the number of agents in the current working agent group according to the formula *1/8 * (N-M)* for only those agents who are not ideal.
  - If the service factor SLWarn is greater than or equal to *Y* and the current AWT is less than the previous AWT, the routing engine tries to reduce the number of agents in the current working agent group according to the formula *1/4 * (N-M)* for only those agents who are not ideal.

> **Important**
>
> Routing Platform cannot reduce the number of working agents if all agents are currently handling interactions.

Time in Ready State

This statistic provides the total current time of an agent in Ready state. It is calculated based on availability - that is, there is no activity currently in progress on a particular DN and the agent has placed the DN in Ready state. The **Time in Ready State** statistic aggregates the total time for the state and this information can be used to build the target list.

You can use the **Time in Ready State** statistic to route interactions to the agent that has been waiting in Ready state for the longest time.

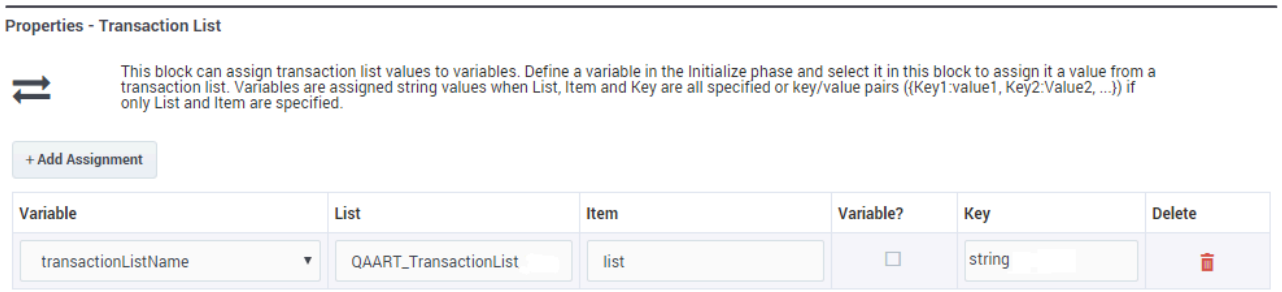
Transaction List Block

> **Important**
>
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

You can use the **Transaction List** block to assign transaction list values to variables. You can define a variable in the **Initialize** phase and then select it in this block and assign it a value from the transaction list.

Variables are assigned string values when **List**, **Item**, and **Key** are all specified, or key-value pairs if you only specify **List** and **Item** (for example, `{Key1:value1, Key2:Value2}`).

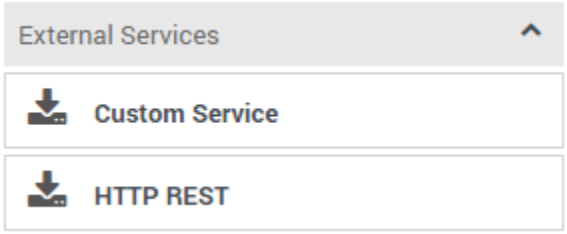
Example



External Services Blocks

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

These blocks allow your application to interact with an external service, such as a custom service that Genesys has provided to your company, or an external system that stores and exposes data through a REST web service.



Use the links below to learn more about each block.

Custom Service

Provides inputs to a custom service that Genesys created for your company.

Used in: **Initialize**, **Self Service**, **Assisted Service**, **Finalize**

HTTP REST

Accesses external systems using RESTful API (over HTTP).

Used in: **Initialize**, **Self Service**, **Assisted Service**, **Finalize**

Custom Service Block

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

You can use the **Custom Service** block to access a custom service that was created by Genesys for your company.

You can provide input to the service. The resulting variable from the block is true if the service request is successful, otherwise the result is false. This result is available for use in later blocks.

Service Details tab

Select the service name and action to use in this **Custom Service** block.

Select **Disable DTMF buffering** if you want to prevent any DTMF inputs made during fetch audio playback from being buffered and carried forward into subsequent User Input or Menu blocks.

If you enable the **Play fetch audio** check box, you can specify an audio resource to play to the caller while the custom service is being fetched.

> **Important**
> Only Announcements containing audio files are supported. TTS audio will not be played.

- Enable the check box beside the **Play fetch audio** check box to specify a variable.
- In the **Play fetch audio minimum for** field, you can enter the minimum length of time to play the audio, even if the custom service has arrived in the meantime.
- In the **Start fetch audio after** field, you can enter a period of time to wait before audio is played.

> **Important**
> In the **Self Service** phase, fetch audio playback stops when the end of the audio file

> is reached, even if the service request is still in progress. In the **Assisted Service** phase, fetch audio playback loops until the service request times out.

Input Parameters

In the **Input Parameters** tab, specify the input expected by the custom service.

- **Name** - Specify the name of the parameter expected by the custom service.
- **Type** - The type of parameter (variable or literal).
- **Value** - Specify the parameter value to pass to the input.

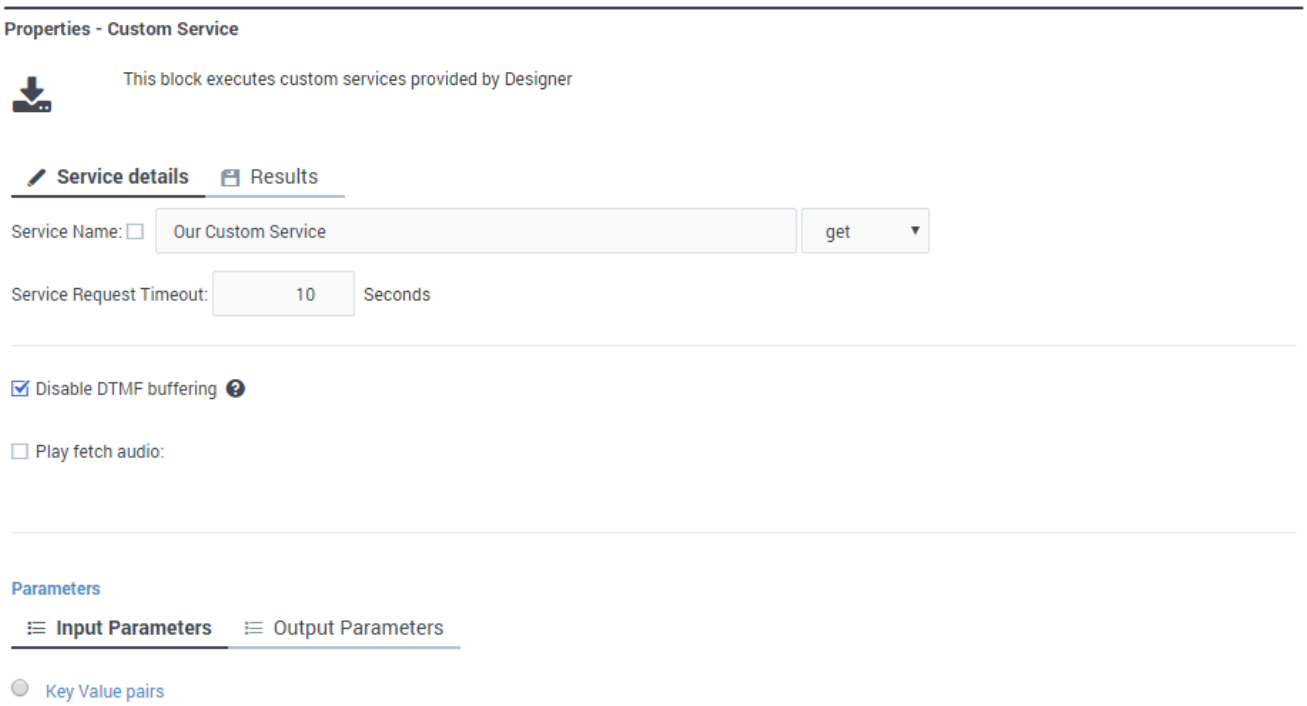
Output Parameters

In the **Output Parameters** tab, specify how and where to store the results of the custom service.

- **Variable Name** - Select the application variable in which to store the data.
- **JSON Expression** - Specify the key in which you expect the result to be in the response object. See the code sample and table below for an example.

```
{
        "thing": {
                "otherthing": "abc"
        },
        "arrayofthings": [
                "thing1", "thing2"
        ]
}
```

JSON Expression	Result
`thing.otherthing`	**abc**
`arrayofthings[1]`	**thing2**



Results tab

Select a variable to store the outcome status (**true** or **false**) of the Custom Service request.

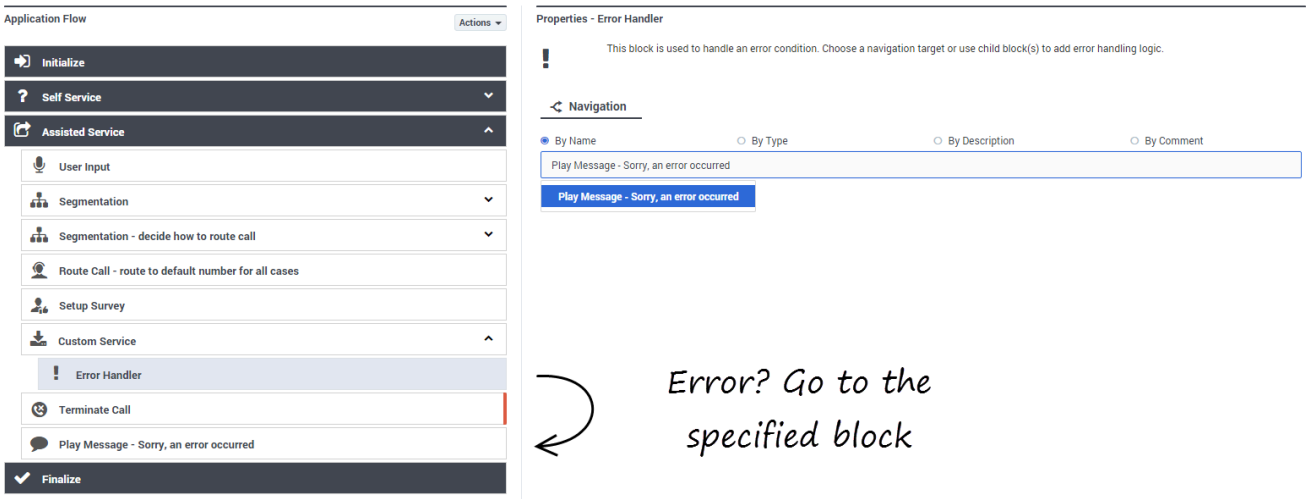
You must also select an action to take if the fetch operation is not successful. You can choose to "Continue with normal processing" or "Execute error handler blocks".

If you select "Execute error handler blocks", an **Error Handler** child block appears under the **Custom Service** block.

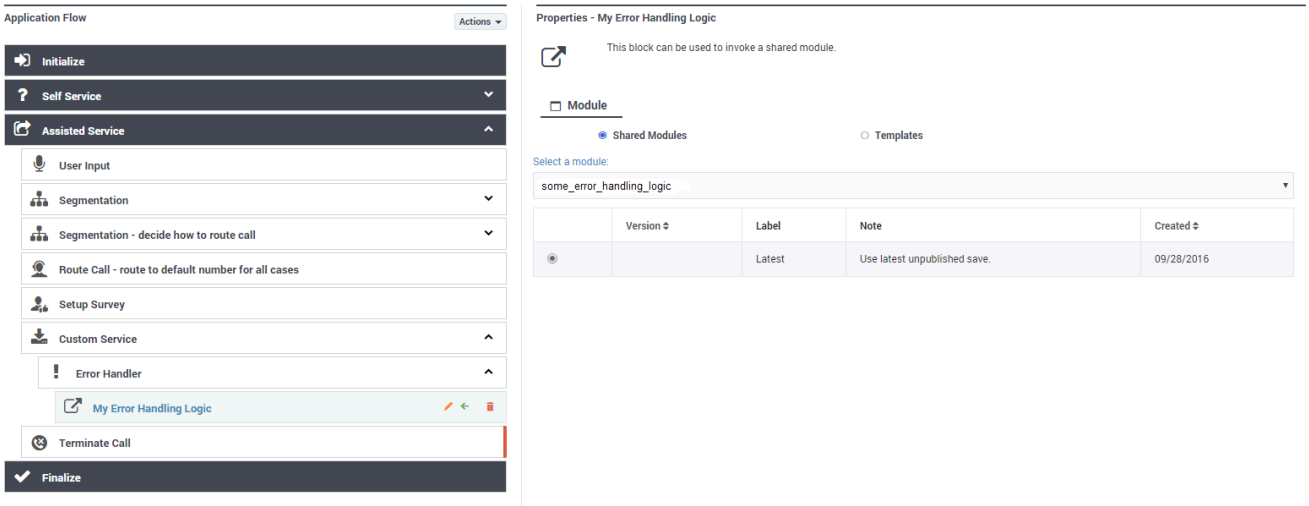
Use the **Error Handler** block to send the application to another target block that you select from the **Navigation** tab, or add child blocks that will perform the actual error handling.

Examples

In this example, the **Navigation** tab is used to specify a target block. If there is an error, the application will go to the **Play Message** block and play an error message:



In this example, a child block is used to invoke a module that will perform the error handling:



Tip

- If you select a target block from the **Navigation** tab, then any child blocks you've added to the **Error Handler** parent block are ignored.
- Standard validation rules still apply — any child blocks that you add to the **Error Handler** block must be valid for the application phase in which they are being used.

HTTP REST Block

> **Important**
>
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

You can use the **HTTP REST** block for accessing external systems using a RESTful API, over HTTP. You can read or write to these web services, although routing applications typically read from web services.

You can read or write to any external system that houses and exposes data through a REST web service. This could be a generic web service, such as one that returns the weather forecast for a specific location, or one that converts a monetary value from one currency to another. Or this could be a company's internal web service that fetches a customer's account details and billing history from the company's internal databases.

This block can be used in all four phases of the application.

> **Tip**
>
> - Check that the RESTful API you are accessing will return data in the format that you expect. While most web services typically return JSON data, there are some that may not. You may want to use an external tool to test the RESTful API outside of Designer to ensure it behaves the way you expect, before attempting to access it within your application.
> - If the request timeout period is reached and no response is received from the REST web service, the output variables have a value of **undefined**.

Service Details tab

Enter the URL of the RESTful web service in the **HTTP URL** field. Enable the check box to use a variable, or disable the check box to use a string.

In the drop-down menu beside the **HTTP URL** field, select the HTTP method to access the web service: **get**, **post**, **put**, or **delete**.

If you are using **post** or **put** as the HTTP method, select an **Encoding Type**. (Otherwise, you will not

see this option.)

In the **Request Timeout** field, enter the time, in seconds, that the application waits for a response from the web service before moving on to the next block.

If you want to post the results of a recording captured by the Record Utterance block to the specified URL, you can specify the variable that holds a recorded file in the **Upload Record Utterance** field. (This option is only supported in the Self Service phase, as the recording file captured by the Record Utterance block is no longer available after the Self Service phase.)

Select **Disable DTMF buffering** if you want to prevent any DTMF inputs made during fetch audio playback from being buffered and carried forward into subsequent User Input or Menu blocks.

Select **Play fetch audio** if you want to specify an audio resource to play to the caller while the data is fetched.

> ### Important
> Only Announcements containing audio files are supported. TTS audio will not be played.

- Enable the check box beside the **Play fetch audio** check box to specify a variable.
- In the **Play fetch audio minimum for** field, you can enter the minimum length of time to play the audio, even if the document arrives in the meantime.
- In the **Start fetch audio after** field, you can enter a period of time to wait before audio is played.

> ### Important
> In the **Self Service** phase, fetch audio playback stops when the end of the audio file is reached, even if the fetch request is still in progress. In the **Assisted Service** phase, fetch audio playback loops until the request times out.

Input Parameters

In the **Input Parameters** tab, specify the inputs to the web service. You can choose either:

- **JSON Payload** — Send a JSON value from a variable as an input to the web service. This option is applicable only for **put** and **post** methods.
- **Key Value pairs** — Click **Add Parameters** and enter the **Name** of the parameter expected by the web service, and the **Value** to pass to the input. You can toggle the **Value** between a string and a variable.

Output Parameters

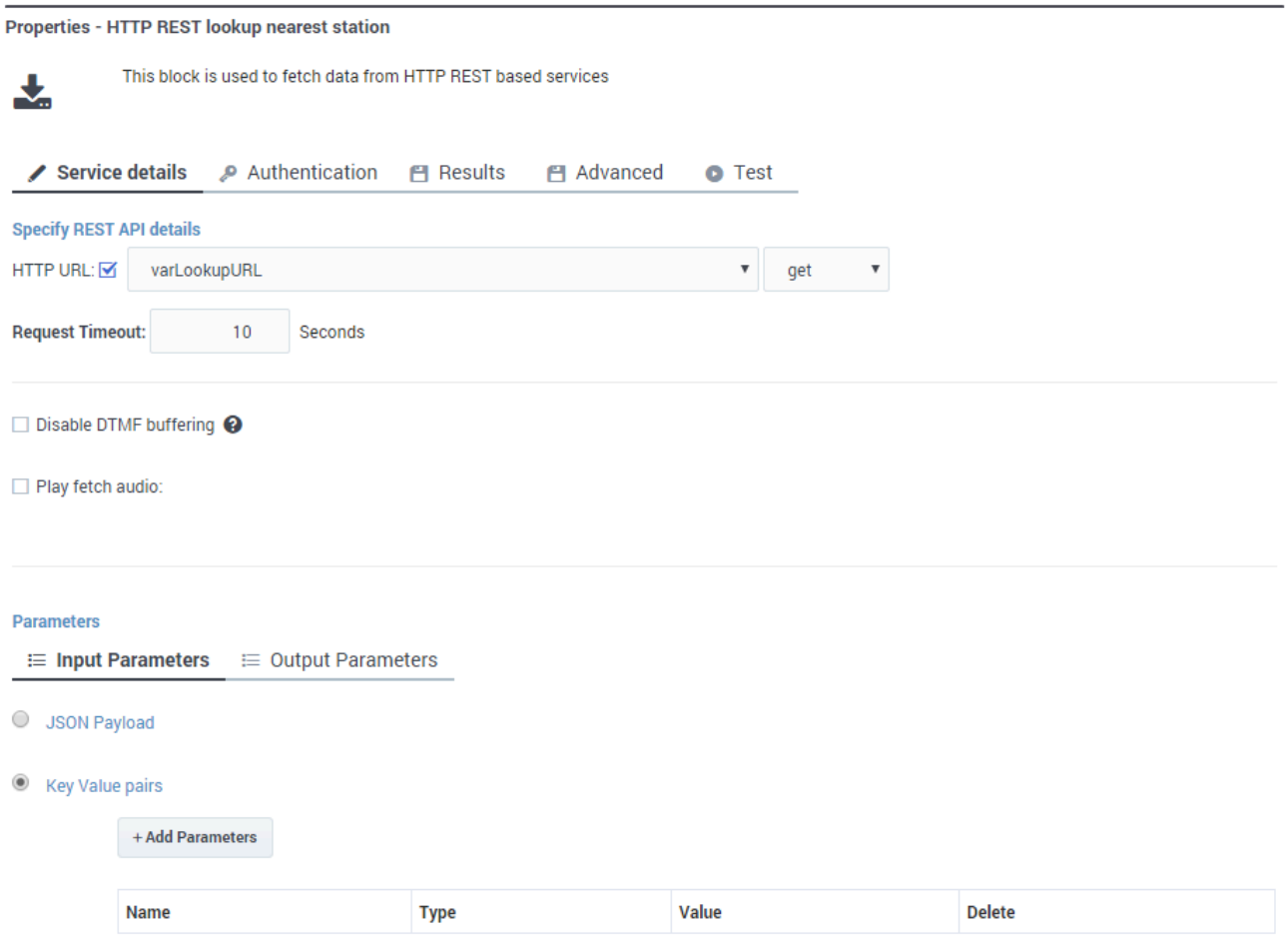
> **Important**
>
> You must only specify an output parameter if you are certain the web service will provide a consistent response. Otherwise, your application will generate an error if the web service provides a response that does not conform to what you have specified in the **JSON Expression** field (for example, a **400** error or a **200** code with no output). If the web service will not provide a consistent response, you can select a variable in the **Results** tab in which to store the entire HTTP response. Next, use an Assign Variables block to check for specific properties in the response and, if these properties are present, specify a JSON expression to assign to the variable.

In the **Output Parameters** tab, click **Add Parameters** to specify how and where to store the results of the web service call. The **Variable Name** is the application variable in which to store the data, and the **JSON Expression** is the key in which you expect the result to be in the response object.

See the code sample and table below for an example:

```
{
        "thing": {
                "otherthing": "abc"
        },
        "arrayofthings": [
                "thing1", "thing2"
        ]
}
```

JSON Expression	Result
`thing.otherthing`	**abc**
`arrayofthings[1]`	**thing2**



Authentication tab

Enable the **Enable Basic Authentication** check box to use HTTP basic authentication as part of the web service request. When enabled, the **User Name** and **Password** fields are displayed. Optionally, click the check box to select a variable for either of these fields.

Results tab

Select a variable to store the outcome status (**true** or **false**) of the HTTP fetch.

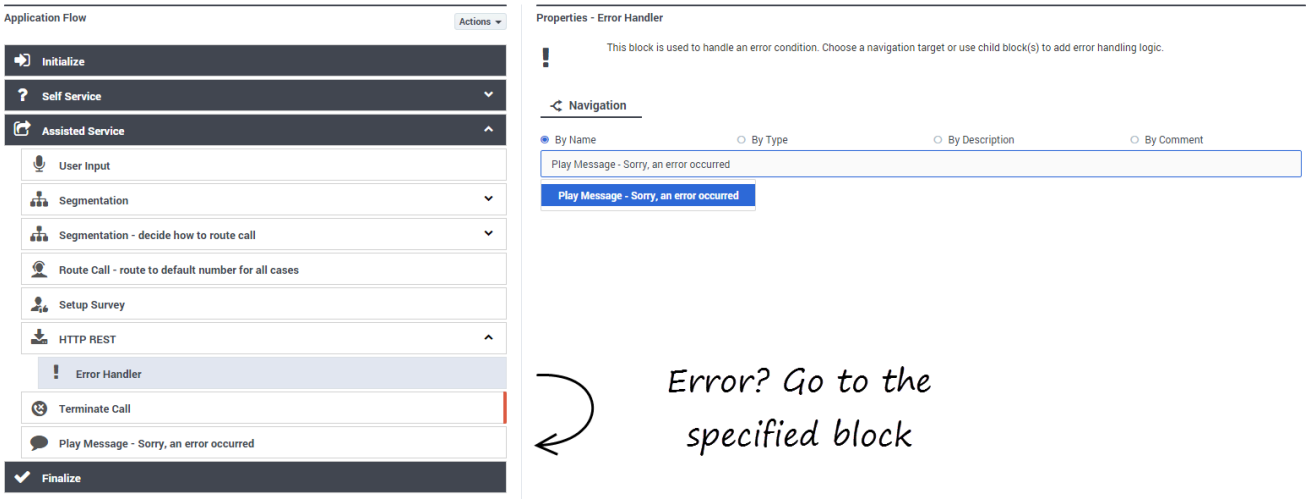
You can also select variables in which to store the data and headers of the HTTP response, and the HTTP error code if the operation failed.

You must also select an action to take if the fetch operation is not successful. You can choose to "Continue with normal processing" or "Execute error handler blocks".

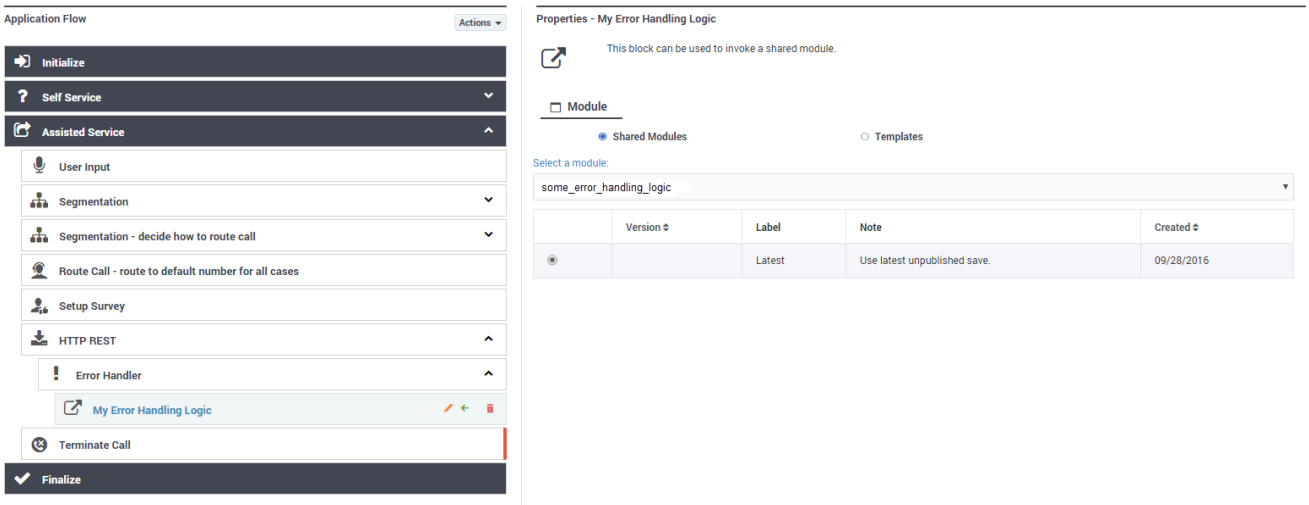
If you select "Execute error handler blocks", an **Error Handler** child block appears under the **HTTP REST** block.

Use the **Error Handler** block to send the application to another target block that you select from the **Navigation** tab, or add child blocks that will perform the actual error handling.

In this example, the **Navigation** tab is used to specify a target block. If there is an error, the application will go to the **Play Message** block and play an error message:



In this example, a child block is used to invoke a module that will perform the error handling:



Tip

- If you select a target block from the **Navigation** tab, then any child blocks you've added

to the **Error Handler** parent block are ignored.

- Standard validation rules still apply — any child blocks that you add to the **Error Handler** block must be valid for the application phase in which they are being used.

Advanced tab

The **Use Designer service to make this request** check box is enabled by default. This allows the fetch request to use a HTTP proxy, which is typically required when sending requests to external resources.

Select **Internal Genesys Service** if the application is sending a fetch request to an internal Genesys service. This type of request does not go through a HTTP proxy.

Click **Add Header** if you want to use a custom HTTP header.

Important

If this **HTTP REST** block is used in a **Self Service** phase or shared module, there might be a processing delay due to the use of a proxy to perform the HTTP fetch. This delay does not apply to **HTTP REST** blocks that are used in **Assisted Service** phases or shared modules.

Test tab

The **Test** tab lets you test an API call from the block without making an actual test call.

Select the variables to be used as Input Parameters (make sure you specify them in the requested format, using single quotes for strings and "()" for JSON values) and any other variables to be used.

If the variables had a default value set in the **Initialize** phase, you can choose to keep those values or provide your own. The application will remember the values used the next time you open the application.

Important

Any literal values stored in the block will also be used for the test request.

Click **Send Test Request** to run the test and generate the results.

Scenarios

If you want to:

- Play weather information for a customer for whom you have a profile and address:
  - This scenario assumes that the weather API expects two input parameters (**date** and **location**) and provides its output in JSON format, under the key **result**. The corresponding input information is stored in two variables: **currentdate** and **zipcode**.
  - Add the **HTTP REST** block to the **Self Service** portion of the application, in a position after you have retrieved the customer location.
  - In the **HTTP URL** field, enter the URL of the weather web service (for example, `http://sample.webservice.com/api/weather/`).
  - Select **get** as the HTTP method.
  - In the **Input Parameters** tab, click **Add Parameters** twice.
  - For the first parameter, use the following information:
    - **Name**: `date`
    - **Type**: `variable`
    - **Value**: `currentdate`
  - For the second parameter, use the following information:
    - **Name**: `location`
    - **Type**: `variable`
    - **Value**: `zipcode`
  - In the **Output Parameters** tab, click **Add Parameters** and use the following information:
    - **Variable Name**: `weather`
    - **JSON Expression**: `result`

Reporting Blocks

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

These blocks are used to manage certain reporting functions within an application, such as to start or stop an activity or to indicate certain points of an application's progress.

Activity

Starts or stops an activity.

Used in: **Self Service**, **Assisted Service**

Debug

Captures information used for debugging (Development stage only).

Used in: **Self Service**, **Assisted Service**

Milestone

Marks key moments within an application.

Used in: **Self Service**, **Assisted Service**

Activity Block

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

You can add the **Activity** block to Self Service or Assisted Service phases to start or stop activity in a report. You can also nest activities to provide additional details.

Do not use **Activity** blocks for modules, as Designer reports module activity automatically.

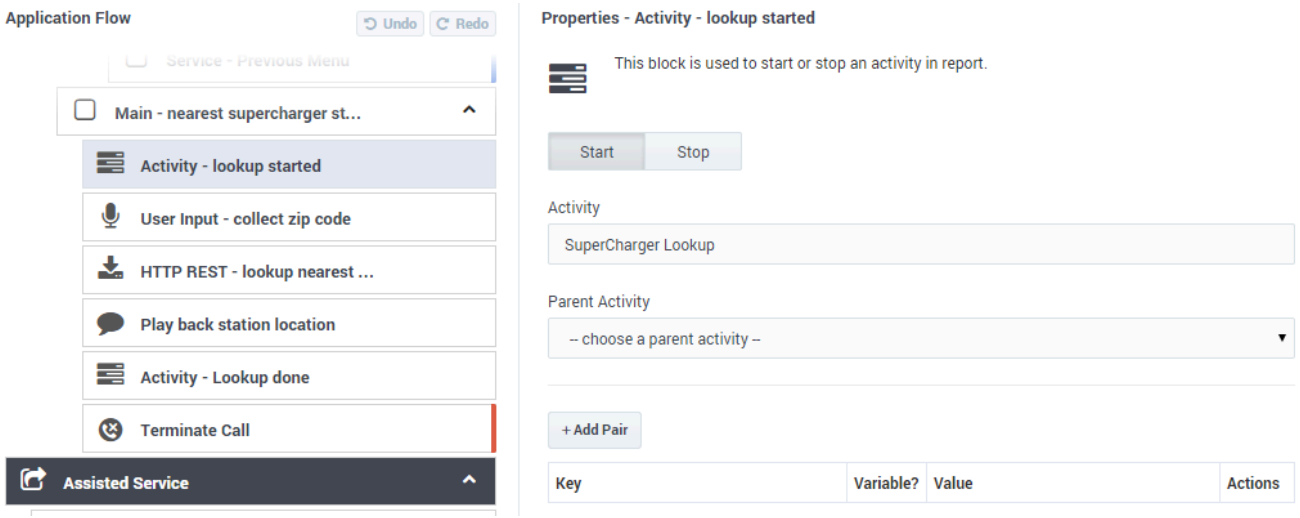
> **Important**
> VAR action IDs are stripped of spaces and pipe characters (|). This includes implicit actions that are generated when a caller enters a shared module.

Start tab

Click **Start** to indicate this block is the start of the activity.

Enter the name of the activity (IVR Action) in the **Activity** field. Optionally, you can select a parent activity by clicking the **Parent Activity** drop-down menu.

Click **Add Pair** to include data, values, or variables to store in the metric data of the activity.

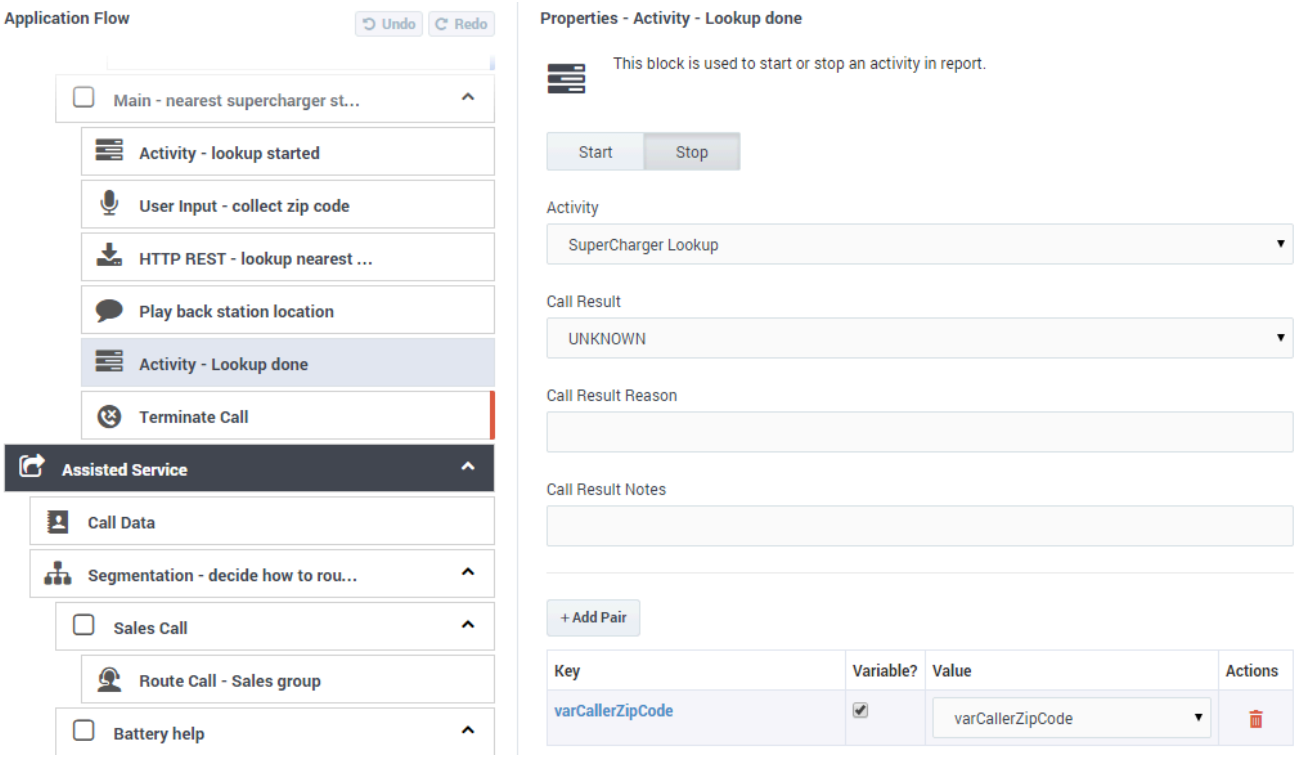


Stop Tab

Click **Stop** to indicate this block is the end of the activity.

Enter information in the following fields: **Call Result**, **Call Result Reason**, and **Call Result Notes**.

Next, click **Add Pair** to include data, values, or variables to store in the metric data of the activity.



Milestone Block

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

You can add the **Milestone** block to **Self Service** or **Assisted Service** phases to mark key moments while the application is running.

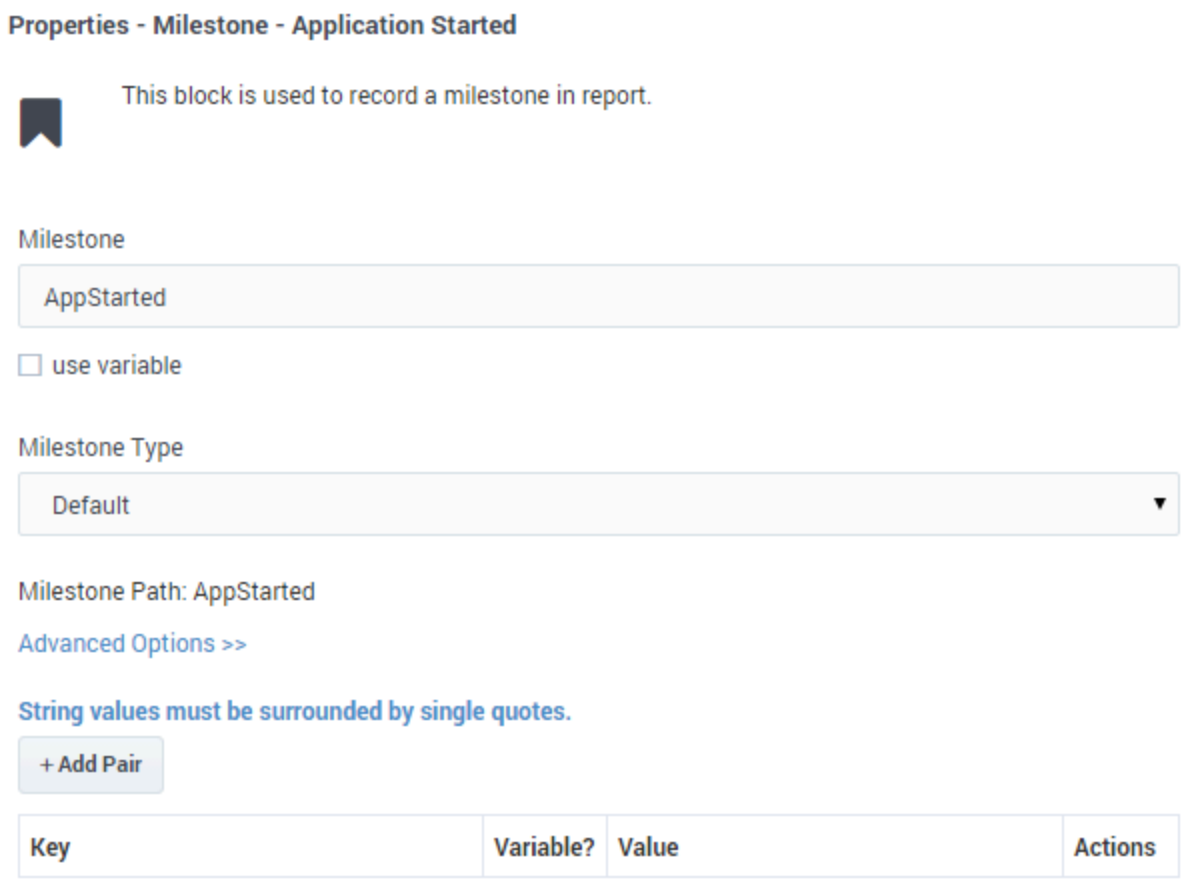
Using this Block

> **Important**
> VAR action IDs are stripped of spaces and pipe characters (|). This includes implicit actions that are generated when a caller enters a shared module.

Enter the name of the milestone in the **Milestone** field. Optionally, if this block is used in the **Self Service** phase, you can enable the **use variable** check box to use a variable for the milestone name.

In the **Milestone Type** menu, select **Default**. Only select **Survey** if this **Milestone** block will be used in a survey application (see the Survey section below for more information).

Optionally, enter additional information by clicking **Advanced Options** or **Add Pair**.



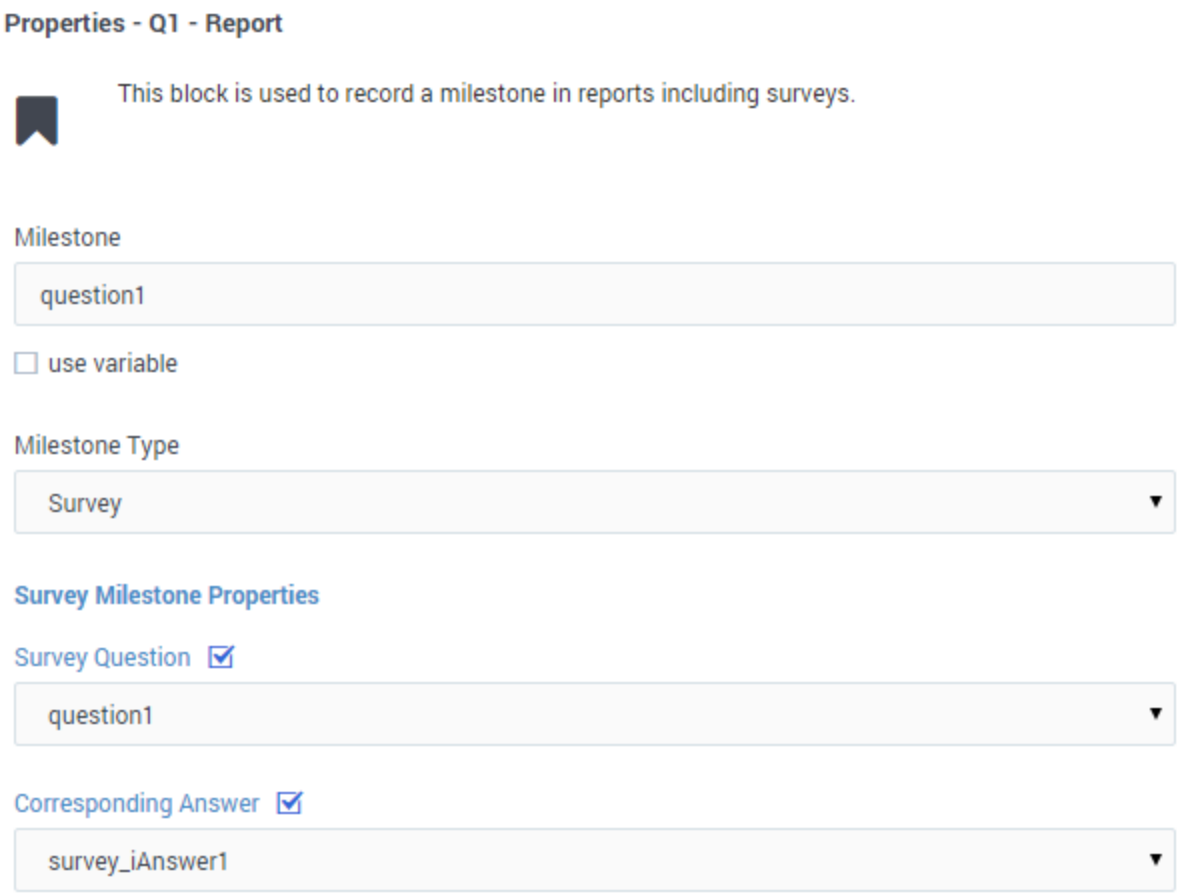
> **Important**
> When entering the payload **Key**, do not use single or double quotes.

Survey

The **Survey** type is reserved for survey applications.

Once you select **Survey**, the **Survey Milestone Properties** section appears. Configure the following:

- **Survey Question** - Select the variable that stored the survey question.
- **Corresponding Answer** - Select the variable that stored the survey answer.



Important

You can disable the check boxes if you would prefer to not use a variable for **Survey Question** and **Corresponding Answer**. However, Genesys recommends that you use variables for consistency and ease of use.

Debug Block

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

You can use the **Debug** block to define a specific checkpoint in a module or application. This is useful when you are developing an application and need to debug runtime logic that isn't running as intended.

For example, you might add some ECMAScript expressions to an Assign Variables block to assign values to certain variables, but discover that the logic isn't producing the desired result.

To assist in debugging this, you could add a **Debug** block immediately after the Assign Variables block to capture the values of the variables as they exist at that time. These values can then be reviewed in Designer Analytics, under the `debugcheckpoints` property of the Session Detail Record (SDR) .

> **Important**
> The **Debug** block is only processed in the Dev, QA, and UAT application streams. It is ignored in the Live stream.

Debug tab

Capture Checkpoint

Select this option to enable debugging for a specified checkpoint.

Checkpoint Name

Specify the name of the checkpoint to be captured. You can also specify a variable that holds the value of the checkpoint name.

Condition

Specify the condition as a script expression. For example:

```
var02 === 1
```

Data tab

This tab displays a list of variables that can be captured by the checkpoint. Select the variables that you want to include.

Advanced tab

Enable the **Write these statements to platform logs** option if you want to write the results of the specified ECMAScript expressions to platform logs that can be reviewed by Genesys support. Click **Add Log** to enter the ECMAScripts.

You can also specify an error message statement to add to the logs if the expression evaluations result in an error.

Callback Blocks

> **Important**
>
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

These blocks manage options, rules, and features for Callback.

> **Warning**
>
> **Use templates or modules — not both.**
>
> Genesys recommends that you avoid mixing templates and modules in callback applications. When planning your applications, decide whether you are going to use templates *or* modules, and then be consistent with your choice.
>
> If you need to make changes to a template, clone *all* of the callback templates into corresponding modules, and then use those modules in your applications.

Book ASAP Callback V2

Books an ASAP ("as soon as possible") Callback on Genesys Mobile Services (GMS).

Book Scheduled Callback V2

Books a scheduled Callback on Genesys Mobile Services (GMS).

Callback Availability V2

Retrieves the scheduled callback availability from Genesys Mobile Services (GMS).

Callback V2

Offers callback and reconnects to the customer when an agent is ready.

Cancel Callback V2

Cancels an existing callback.

Check for Existing Callback V2

Checks if the customer's phone number already has an existing callback scheduled or queued in a particular Callback service in Genesys Mobile Services (GMS).

Validate Phone Number

Provides phone number validation and international phone number support for Callback V2.

Callback VQ Watermark

Checks the number of active callbacks that are currently queued for a specific virtual queue (VQ).

Callback V2

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

You can use the **Callback V2** block in the **Assisted Service** phase of your **Default** type application for inbound calls. Designer supports the following types of callbacks:

- Immediate (or in-queue) callbacks, where the caller requests a callback when the next agent is available.
- Scheduled callbacks, when the caller selects a preferred time and date for the callback.
- Web-invoked callbacks, where the caller requests a callback using an HTTP request (such as a website or a mobile application to request a callback when an agent is available).

You can use the **Callback V2** block in the **Initialize** phase of your **Callback** type application for Scheduled callback. This block processes the Scheduled callback at the desired time.

Before using the **Callback V2** block, you must first create a variable for the callback virtual queue. Then, you can use Special Days, Business Hours, and Data Tables (under Business Controls) to specify your business requirements and associate those settings with the virtual queue.

The settings for callback virtual queues are stored in the Callback_Settings data table.

> **Important**
> If redirecting a caller to an application that contains IVR callback, only **1-step transfers** are supported.

Sample callback scenarios

The following scenarios describe sample call flows for immediate and scheduled callbacks.

Immediate callbacks

- The session starts when the customer's call arrives.
- The caller is offered immediate (or in-queue) callback. They accept, and confirm the number they want to be called at.

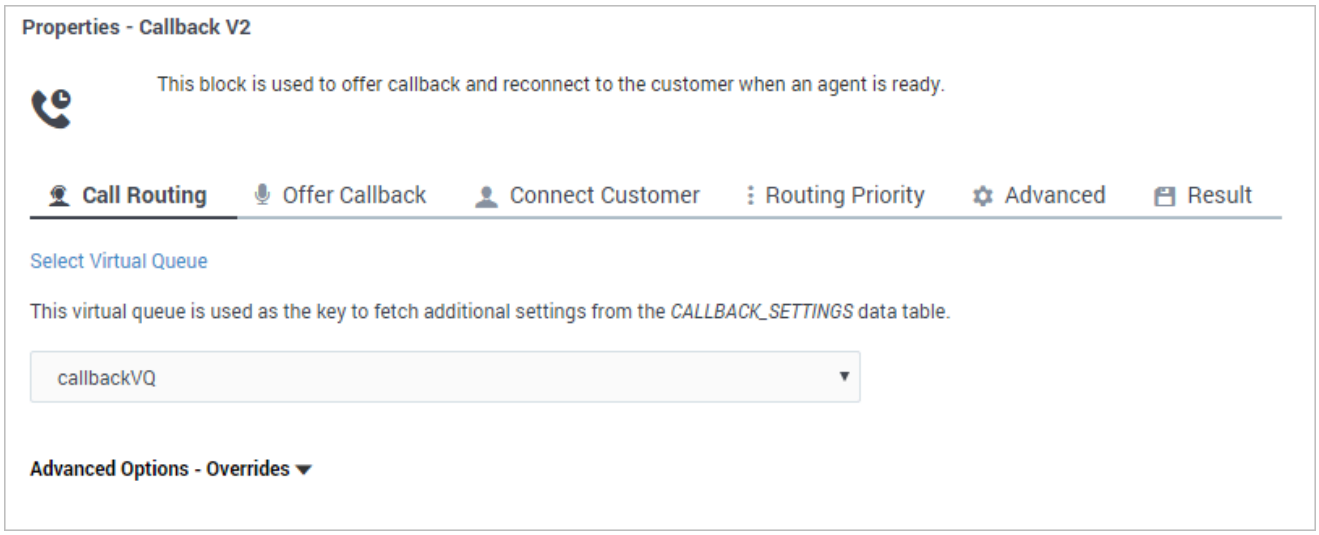
- At this point, the caller can hang up. The voice interaction is converted to a virtual call and added to the queue.
- While the virtual call waits in the queue, the session remains active and continues to monitor statistics for the call, such as the Estimated Wait Time (EWT) and it's position in the queue.
- When an agent that satisfies the required skill expression is ready, the customer is called.
- Music on hold plays while the call is being routed to the agent.
- Once the agent connects to the call, the virtual call is removed from the queue and the session ends.
- (Optional) If survey is enabled and the caller has agreed to take it, the caller is taken to the survey application after the agent disconnects.

Scheduled callbacks

- The session starts when the customer's call arrives.
- The caller is offered a scheduled callback. They accept, and confirm the number they want to be called at, along with the date and time when they would like to receive the callback.
- At this point, the caller can hang up.
- When an agent that satisfies the required skill expression is available, the customer is called.
- Music on hold plays while the call is being routed to the agent.
- Once the agent connects to the call, the virtual call is removed from the queue and the session ends.
- (Optional) If survey is enabled and the caller has agreed to take it, the caller is taken to the survey application after the agent disconnects.

Call Routing tab

Select the **Virtual Queue** that you are going to use for callback. Designer uses this Virtual Queue to fetch the associated configuration settings from the **CALLBACK_SETTINGS** data table.

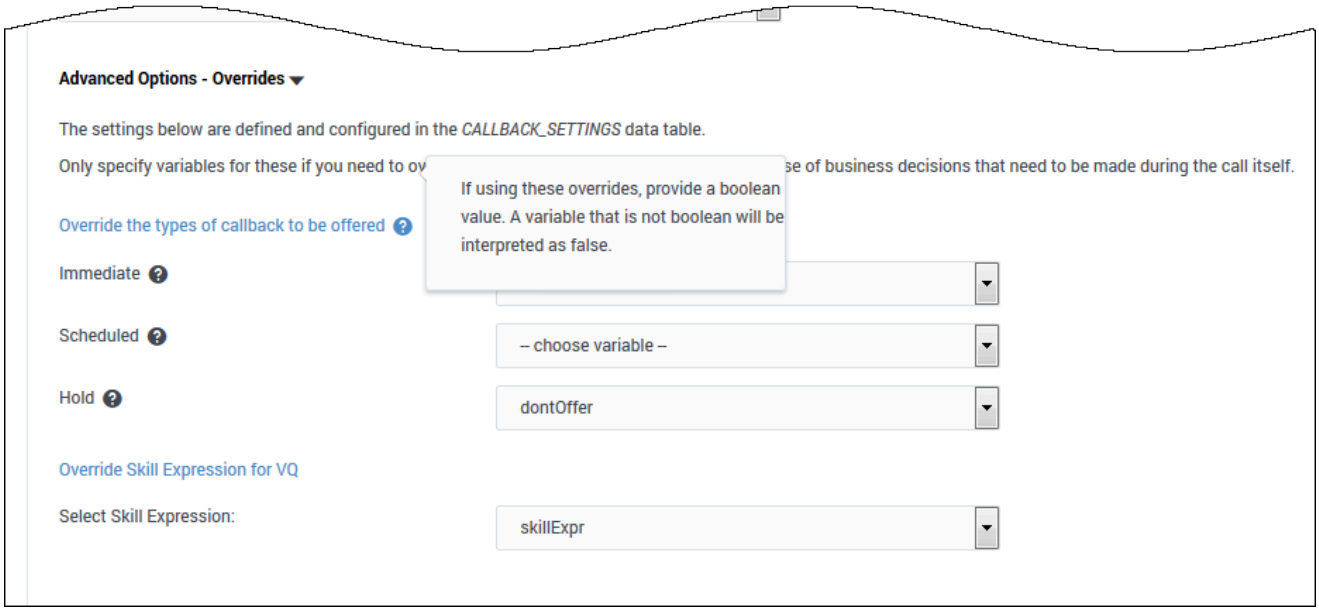


Advanced Options - Overrides

(Optional) You can expand the **Advanced Options - Overrides** section to select your own variables for certain parameters. For example, your business might require that an offer or skill expression parameter override the current setting in **CALLBACK_SETTINGS** with a different value.

Important

Variables used for overrides must be provided as *boolean* values (for example, `true`/`false`, or `0`/`1`). Otherwise, Designer interprets the variable lookup as `false`.



Offer Callback tab

In the drop-down menu, select **Callback V2 - Offer Callback** to use the pre-packaged template for callback.

The inbound callback feature is provided by a series of shared modules. The **Callback V2** block hands off the call to one main shared module that guides callers through the callback process. This shared module might rely on one or more supporting shared modules to extend its functionality (such as to collect a phone number or negotiate a time for **Scheduled** callback). When the callback process is complete, the main shared module returns the call to your application.

For ease of use, you can use shared module templates that provide pre-packaged callback functionality. The templates are read-only and cannot be edited or deleted. If you want to modify these templates, go to the **Shared Modules** list and click **Clone** beside a template to create a copy for editing.

> ### Warning
> Although you can copy a template to modify its prompts or behavior, you must not change its inputs or outputs. Doing so might cause unexpected behavior or application errors. If you want to change audio prompts only, you can modify audio resources in the **Callback V2 Audio** audio collection, which you can access by going to the Media Resources window.

Connect Customer tab

In the drop-down menu, select **Callback V2 - Calling Back** to use the pre-packaged template for callback.

The outbound callback feature is provided by a shared module.

For ease of use, you can use a shared module template that provides pre-packaged callback functionality. The template is read-only and cannot be edited or deleted. If you want to modify this template, go to the **Shared Modules** list and click **Clone** beside the template to create a copy for editing.

> ### Warning
> Although you can copy a template to modify its prompts or behavior, you must not change its inputs or outputs. Doing so might cause unexpected behavior or application errors. If you want to change audio prompts only, you can modify audio resources in the **Callback V2 Audio** audio collection, which you can access by going to the Media Resources window.

Routing Priority tab

Enable the **Use Priority during Routing** check box to use priority-based routing, which prioritizes your calls depending on your business requirements.

To prioritize calls, you should set the **Initial Priority** based on your business segmentation (for example, *Gold* customers start at Initial Priority = 50, *Silver* customers start at Initial Priority = 30, and *Bronze* customers start at Initial Priority = 0).

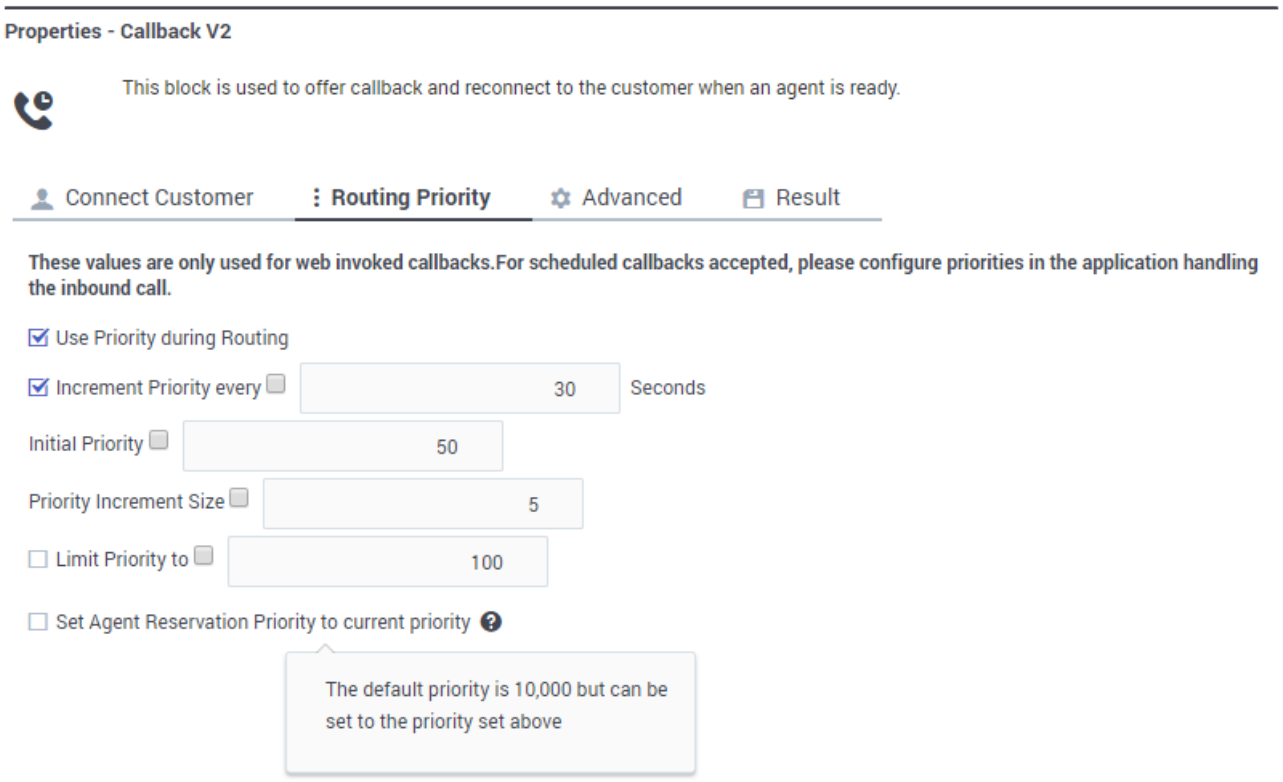
Enable the **Increment Priority every _____ seconds** check box to specify the time interval between priority increments. If you enable the other check box beside the field, you can select a variable that specifies the overall **Routing Timeout** and **Priority Increment Interval** properties.

If the **Increment Priority every _____ seconds** option is enabled, you can use the **Limit Priority to** option to set a maximum priority value. For example, if the initial priority is 50, you can use this option to not let the priority value increase beyond 100.

If you enable the other check box beside the field, you can select a variable for this option.

If **Use Priority during Routing** is enabled, you can also choose to enable the **Set Agent Reservation Priority to current priority** option. This will apply the current priority of the call at the time an agent was found for the callback to the agent reservation request. If you choose not to enable this option, the default priority value of 10,000 is used.

Example



> **Tip**
> Ideally, the Route Call block and **Callback V2** block should have their priorities synchronized, so that their rate of increase is the same. One way you can do this is by using variables for the **Initial Priority**, **Increment Priority every...**, **Priority Increment Size**, and the **Limit Priority to...** settings.

Advanced tab

Greetings

Enable the check box beside **Customer Greeting** and/or **Agent Greeting** to play an audio file to that person while the call is being connected.

For customers, you might use this feature to play a legal disclaimer, or to announce that the call might be recorded (if you use call recording in your contact center). For agents, you might use a variable to announce the customer name or other relevant information.

After you enable **Customer Greeting** and/or **Agent Greeting**, you can select an audio file to play by clicking the icon in the **Announcement** field. This is useful for customer greetings that play a static disclaimer audio file.

Optionally, enable the **Var?** check box to use a variable to dynamically select the audio file. This is useful for agent greetings that use a variable to provide call-specific information, such as the customer name.

Music on Hold

Enable **Music on hold** to select the music file that plays while callers are on hold.

Reporting

Enable **Put (re)connected call into a virtual queue** if you would like to place the real interaction in a separate virtual queue for reporting purposes, to differentiate between regular calls and calls routed to agents as a result of a callback.

You can either select a virtual queue that is defined in the application or you can add a suffix to the virtual queue that is used for the inbound call. For example, if the callback virtual queue is named *VQ1_cb*, and the suffix is *_out*, the reconnected virtual queue should be configured as *VQ1_cb_out*.

The following options enable you to specify the metrics to display in reporting:

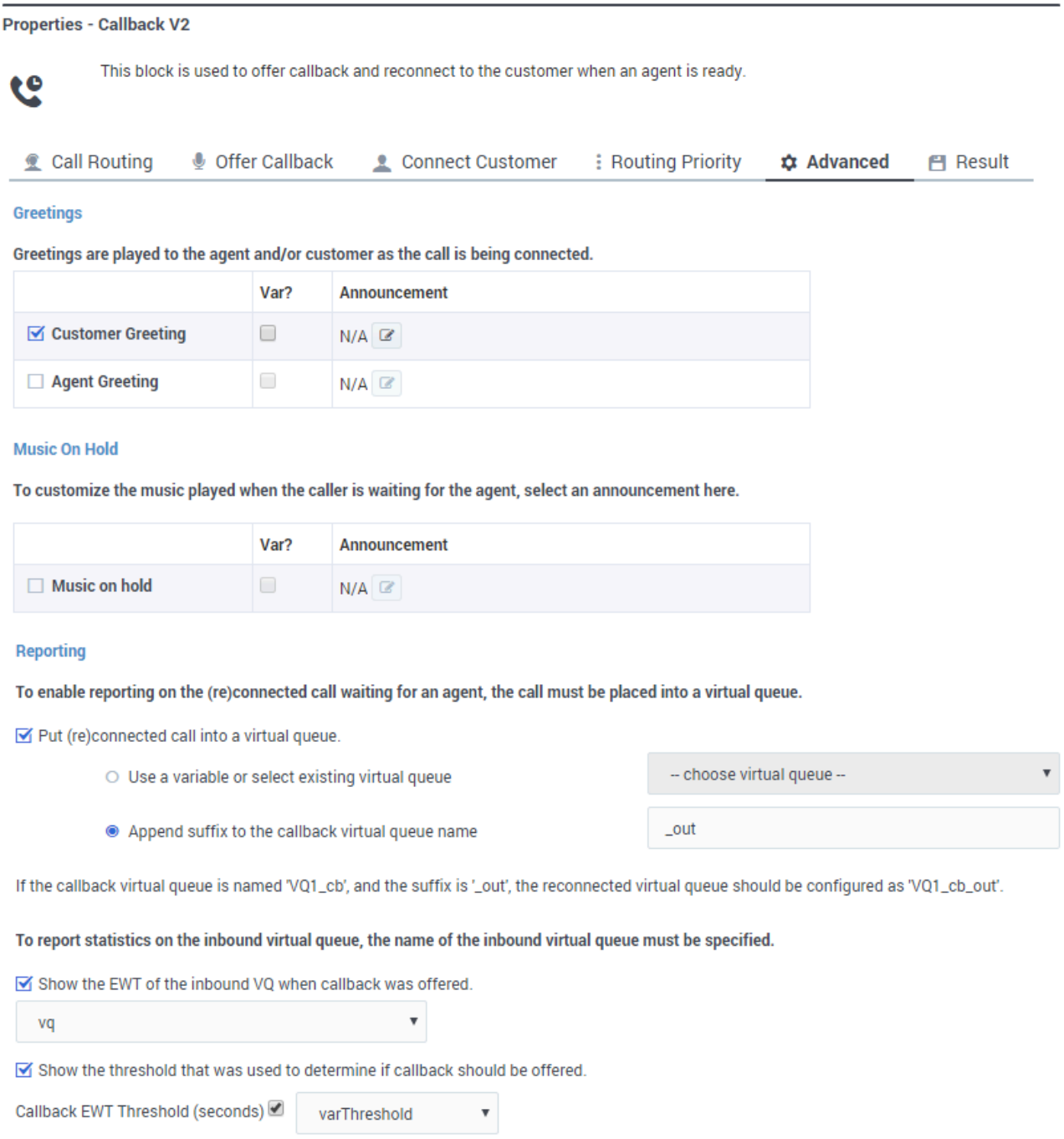
Enable **Show the EWT of the inbound VQ when callback was offered** to specify the name of the inbound virtual queue. You can select a variable or one of the virtual queues available in the drop-down menu.

Enable **Show the threshold that was used to determine if callback should be offered** to specify the **Callback EWT Threshold value** (in seconds). You can select a variable or enter an integer.

Survey

(If survey is enabled) Enable **Route to a different RP than previously specified in Setup Survey block** if you need to change the routing point to use for the survey application after the agent disconnects. Otherwise, the routing point configured in the Setup Survey block is used.

Example


Properties - Callback V2
This block is used to offer callback and reconnect to the customer when an agent is ready.
Call Routing
Offer Callback
Connect Customer
Routing Priority
Advanced
Result
Greetings
Greetings are played to the agent and/or customer as the call is being connected.
Var?
Announcement
Customer Greeting
N/A
Agent Greeting
N/A
Music On Hold
To customize the music played when the caller is waiting for the agent, select an announcement here.
Var?
Announcement
Music on hold
N/A
Reporting
To enable reporting on the (re)connected call waiting for an agent, the call must be placed into a virtual queue.
Put (re)connected call into a virtual queue.
Use a variable or select existing virtual queue
-- choose virtual queue --
Append suffix to the callback virtual queue name
_out
If the callback virtual queue is named 'VQ1_cb', and the suffix is '_out', the reconnected virtual queue should be configured as 'VQ1_cb_out'.
To report statistics on the inbound virtual queue, the name of the inbound virtual queue must be specified.
Show the EWT of the inbound VQ when callback was offered.
vq
Show the threshold that was used to determine if callback should be offered.
Callback EWT Threshold (seconds)
varThreshold

Result tab

(Optional) In the drop-down, select a variable to store the outcome of the callback interaction.

Callback Settings Data Table

The callback settings for each virtual queue are stored in a special data table called **CALLBACK_SETTINGS**. You can view the settings for this data table by selecting it on the Data Tables page.

The data table includes a default queue that is already populated with the recommended values. To add a new virtual queue, simply add a new row to the data table. Each parameter is automatically assigned the default setting, but you can edit the values to further refine and customize the callback settings for each virtual queue.

If you are making changes to this data table, note the following:

- If your application is not automatically detecting the caller's number (ANI), you might have to use the *Dial Prefix* setting to enter a country calling code, or use the audio prompts to ask callers to include their country code when manually entering their callback phone number.
- The value for *Slot Duration (minutes)* must be a divisor of 60. The recommended values are 15 (default), 20, 30, or 60, with 60 being the maximum value you can use.

Parameters

This data table contains the following parameters:

Setting	Key	Description	Default Value	Web Callbacks (for API – see note)
VQ	(section name)	Name of the Virtual Queue.	(none)	
Immediate Enabled	_immediate_enabled	Enables (or disables) the option to offer Immediate Callbacks.	true	✗
Scheduled Enabled	_scheduled_enabled	Enables (or disables) the option to offer Scheduled Callbacks.	true	✗
Hold Enabled	_hole_enabled	Enables (or disables) the option to Hold (or Reject) a callback.	true	✗
Logged In Check	_logged_in_check	Checks to see if any agents are logged in before offering Immediate Callback. If this feature is enabled and no agents are logged in, Immediate callback is not offered.	false	✗
Immediate Blackout (minutes)	_immediate_blackout	This value acts as a cut-off time (in minutes) before the end of the business day when Immediate callbacks won't be offered. For example, if the business closes at 5:00 PM and the **Immediate Blackout** value is set to 60 minutes (default), customers who call and receive an estimated waiting time that exceeds 4:00 PM (i.e. 60 minutes before closing) won't be offered an Immediate	60	✗

Setting	Key	Description	Default Value	Web Callbacks (for API – see note)
		callback. Important If **Immediate Offer Hours** is configured, this option is ignored.		
Callback Purge Time (minutes)	_callback_purge_time	Duration (in minutes) to keep a callback session alive before we make a courtesy call to reschedule or cancel the callback because no agents were found and the callback cannot be processed. However, the courtesy call will be made at the end of the business day if the business closes before the callback session alive time. Important You might want the courtesy outbound call to happen at the end of the business day. To do this, set the value of this parameter to a number that is greater than the total number of minutes the office is open. For example, an office with business hours of 09:00–17:00 would be open 480 minutes (or 8 hours). Setting this parameter to a higher value, such as 500, would initiate an outbound call to the customer after business hours (for example, to inform them that no	120	✓

Setting	Key	Description	Default Value	Web Callbacks (for API – see note)
		agents were available and to provide them with the option to reschedule or cancel the callback). |2		
Call Display Name	_call_display_name	Name to display for Caller-ID.		✓
Call Display Number	_call_display_number	Number to display for Caller-ID.		✓
Enable CPD	_cpd_enable	Enables (or disables) Call Progress Detection.	false	✓
CPD Timeout (seconds)	_cpd_timeguard	Specifies the maximum time (in seconds) allowed for Call Progress Detection after the call is connected.	3	✓
Dial Prefix	_prefix_dial_out	The prefix to add to the phone number for outbound dialing. (This should only be used to add the country code, if desired. The + should not be added here, since it should already be configured in the dial plan.) Important If your application is not automatically detecting the caller's number (ANI), you might have to use this setting to enter a country calling code, or use the audio prompts to ask callers to include their country code		✓

Setting	Key	Description	Default Value	Web Callbacks (for API – see note)
		when manually entering their callback phone number.		
Dial Retry Timeout (seconds)	_dial_retry_timeout	Time to wait (in seconds) before making another attempt to dial an outbound call, if the previous attempt failed.	30	✓
Max Dial Attempts	_max_dial_attempts	Maximum number of times to try dialing an outbound call.	3	✓
Min Time Before Callback (seconds)	_min_time_before_callback	Minimum time (in seconds) between the disconnection of the inbound call and the dialing of the outbound call.	60	✓
Snooze Duration (minutes)	_snooze_duration	Time to wait (in minutes) before dialing a caller who chose the "snooze" option from the menu.	5	✓
Pushed Callback Expiry Time (minutes)	_pushed_callback_expiry_time	Duration (in minutes) to keep user-originated callback sessions alive.	120	✓
Skill Expression	_target	The skill expression to use for targeting an agent. (Example: `Billing>0&Collections>0`)		✓
Attach Userdata	_attach_udata	Specifies the format in which the user data should be attached to the interaction before it is routed to an agent.	separate_keys	✓

Setting	Key	Description	Default Value	Web Callbacks (for API – see note)
		• Selecting **single_json** will attach all user data as one JSON object (key: GMS_UserData). • Selecting **separate_keys** will attach each user data as a separate key. (The name of the key will be the same as the user data key.)		
Business Hours	_business_hours_service	Name of the Business Hours entry for this VQ. This name must correspond to one of the entries in Business Hours. • If **Immediate Offer Hours** is not configured, these hours apply to both immediate and scheduled callbacks. • If **Immediate Offer Hours** is configured, these hours apply only to scheduled callbacks. • For immediate callbacks, these hours indicate when immediate callback is		✗

Setting	Key	Description	Default Value	Web Callbacks (for API – see note)
		to be offered (up until the time specified by **Immediate Blackout**, if configured). • For scheduled callbacks, these hours indicate when timeslots will be available for booking scheduled callbacks.		
Immediate Offer Hours	_immediate_offer_hours	Name of the Business Hours object that defines the hours when Immediate callback is to be offered. This name must correspond to one of the entries in Business Hours. However, this option uses the timezone of the Designer application. • Immediate callback will not be offered if the current time plus the **Estimated Wait Time** (EWT) is outside of the hours defined in **Business Hours** (_business_hours_service). For example, if Business Hours are set to Monday-Friday,		✗

Setting	Key	Description	Default Value	Web Callbacks (for API – see note)
		09:00 - 17:00, the EWT is 10 minutes, and the current time is Monday at 16:55, then Immediate callback will not be offered. Important • This option disables both **Immediate Enabled** and **Immediate Blackout (minutes)**. • If **Immediate Offer Hours** is configured, the override for offering Immediate callback configured in the **Callback V2** block is ignored.		
Call Direction	_call_direction	Determines who will initiate the call to the target.	USERTERMINATED	(n/a)
Slot Capacity	_max_request_by_time_bucket	How many callbacks can be offered for each slot.	5	✗
Slot Duration (minutes)	_request_time_bucket	Duration (in minutes) of the time slots for scheduled callbacks.	15	✗

Setting	Key	Description	Default Value	Web Callbacks (for API – see note)
		Important This value must be a divisor of 60. The recommended values are 15 (default), 20, 30, or 60, with 60 being the maximum value you can use.		
Routing Point	_route_point	Routing Point (RP) to use for making the outbound call.		✓
Callback Application	_service	The name of the Designer callback application.		✓
Application Stream	_stream	Specifies the stream of the application (of **Callback** type) that will be used.	Live	✓
Callback TTL (seconds)	_ttl	Specifies how long (in seconds) the callback record is stored on the system. This setting is fixed at 14 days (starting from the Desired Callback Time) and cannot be changed.	259200	✓

Important

Web Callbacks (API): If you are booking callbacks directly with the API, not all settings in the CALLBACK_SETTINGS data table are taken into account. The parameters that are included with API callbacks are marked with a checkmark (✓) in the **Web Callback** column. Parameters with a crossmark (✗) can be checked manually with the appropriate API calls prior to booking the callback. For more information, see Genesys Engage REST APIs and tutorials for Callback.

Book ASAP Callback V2

> **Important**
>
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

> **Important**
>
> - This block relies on callback functionality provided by Genesys Mobile Engagement (formerly known as Genesys Mobile Services). Read the Callback User's Guide for more information on how to implement this feature.
> - You must use this block in connection with the callback feature. See the Callback V2 block page for more information.

Use this block to book an immediate callback ("as soon as possible").

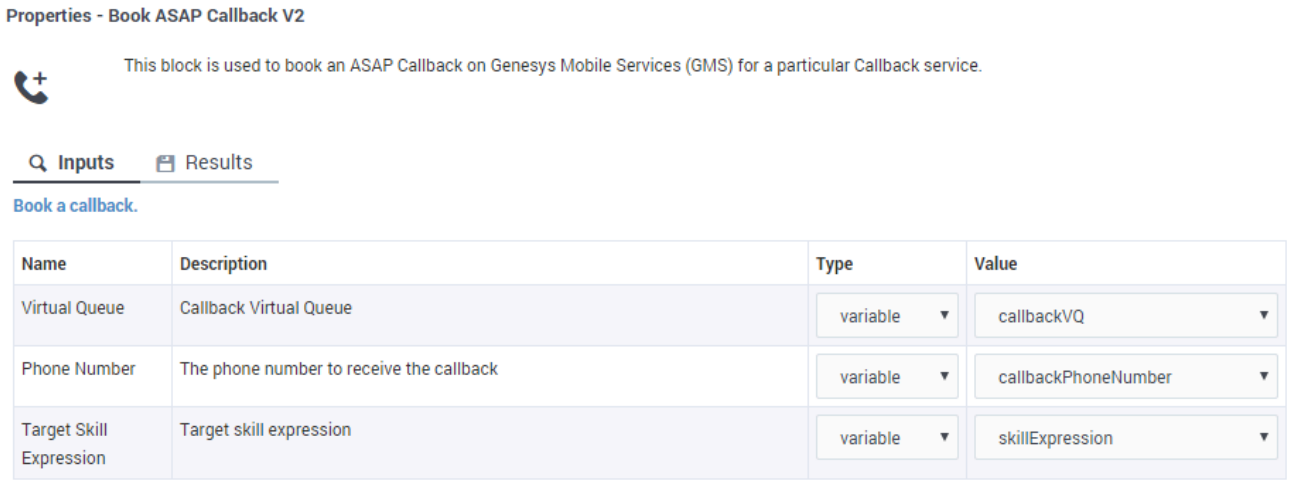
Inputs tab

Select the input **Type** and **Value** for the following parameters (this step is mandatory):

- Virtual Queue
- Phone Number
- Target Skill Expression

You can use literal or variable value types.

Example



Results tab

Select the variables that will store the results of the **Outcome** and **Callback ID** queries.

> **Tip**
>
> Genesys recommends that you use the system variable *GmsCallbackServiceID* to store the value of the **Callback ID**.

Example

Book Scheduled Callback V2

> **Important**
>
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

> **Important**
>
> - You must use this block in connection with the callback feature. See the Callback V2 block page for more information.

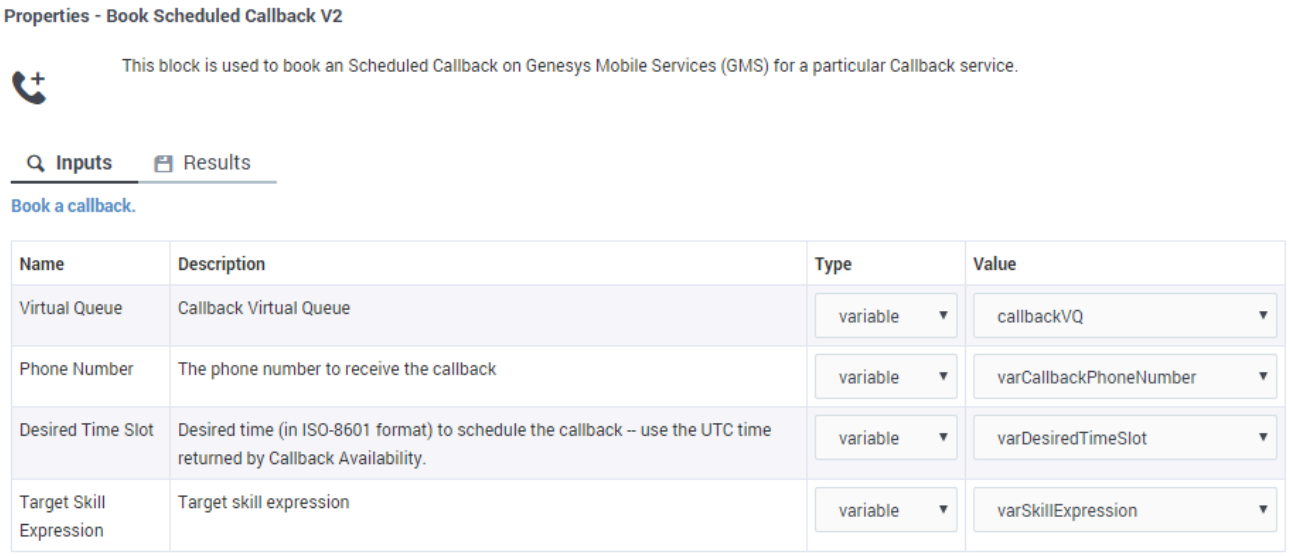
Use this block to book a scheduled callback.

Inputs tab

Select the input **Type** and **Value** for the following parameters (this step is mandatory):

- Virtual Queue
- Phone Number
- Desired Time Slot (this must be in ISO-8601 format, i.e. YYYY-MM-DD)
- Target Skill Expression

Example

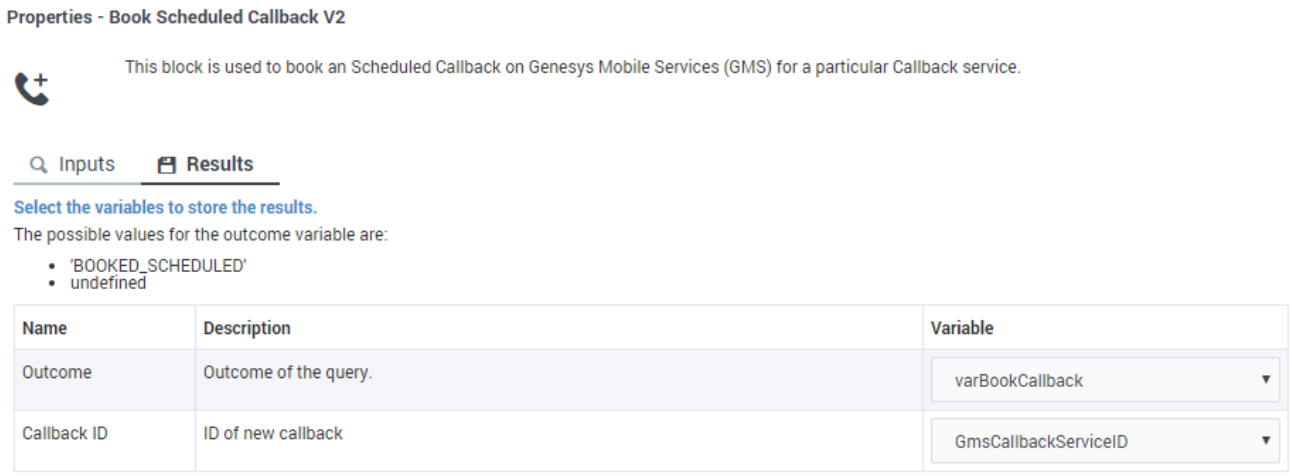


Results tab

Select the variables that will store the results of the **Outcome** and **Callback ID** queries.

> ### Tip
> Genesys recommends that you use the system variable *GmsCallbackServiceID* to store the value of the **Callback ID**.

Example


Properties - Book Scheduled Callback V2
This block is used to book an Scheduled Callback on Genesys Mobile Services (GMS) for a particular Callback service.
Inputs
Results
Select the variables to store the results.
The possible values for the outcome variable are:
'BOOKED_SCHEDULED'
undefined
Name
Description
Variable
Outcome
Outcome of the query.
varBookCallback
Callback ID
ID of new callback
GmsCallbackServiceID

Callback Availability V2

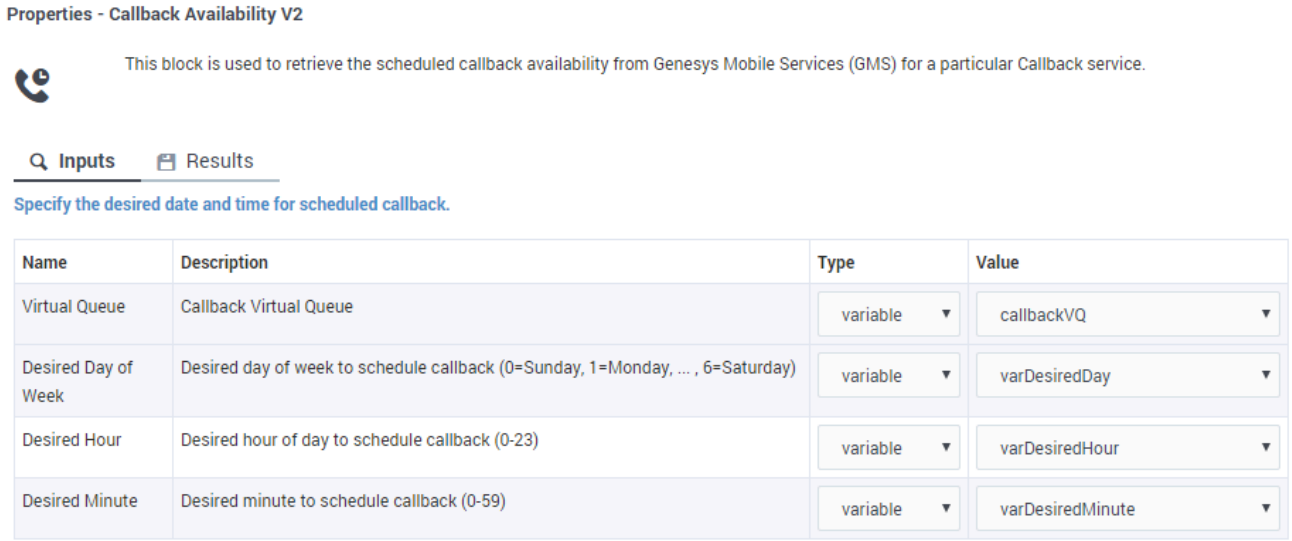
> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

This block checks to see if there are any available time slots for the caller's preferred callback date and time, and provides up to three possible time slots to choose from.

Inputs tab

Specify the desired date and time for the callback.

Example



Results tab

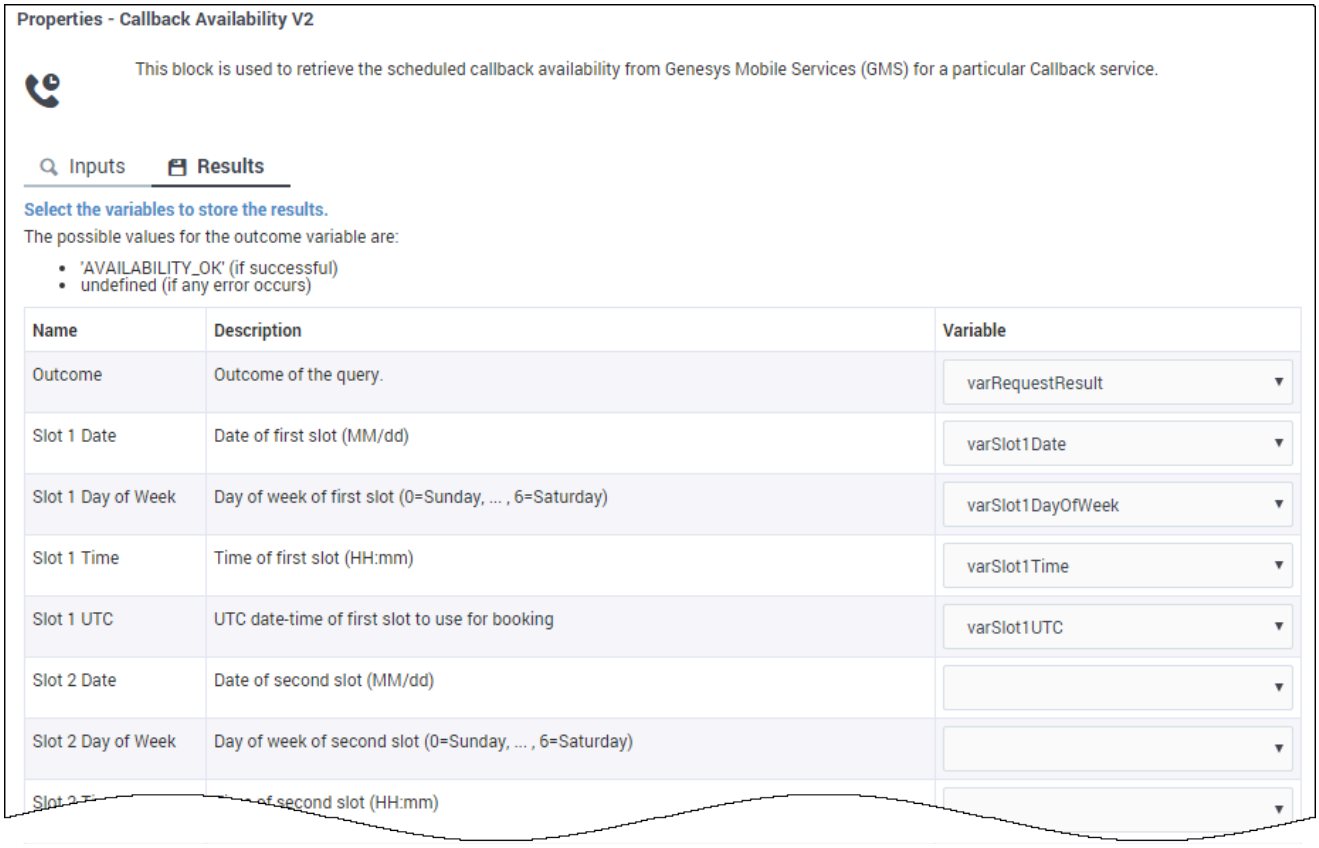
The closest available time slot to the requested callback date and time is returned, while also taking

into consideration the Estimated Wait Time (for example, if the current Estimated Wait Time is 30 minutes, the earliest time slot that could be offered is 30 minutes from now). In addition, up to two alternate time slots are also proposed.

> ### Important
> The desired day of week, hour, and minute should be collected from the caller in the time zone of the Designer application. For the three closest time slots that are returned, the **Slot *x* Date**, **Slot *x* Day of Week**, and **Slot *x* Time** are all in the time zone of the Designer application, so they can be played as prompts back to the customer to confirm the time, and **Slot *x* UTC** is in the UTC (Coordinated Universal Time) time zone.

You can use this tab to store the time slot results in variables.

Example



Cancel Callback V2

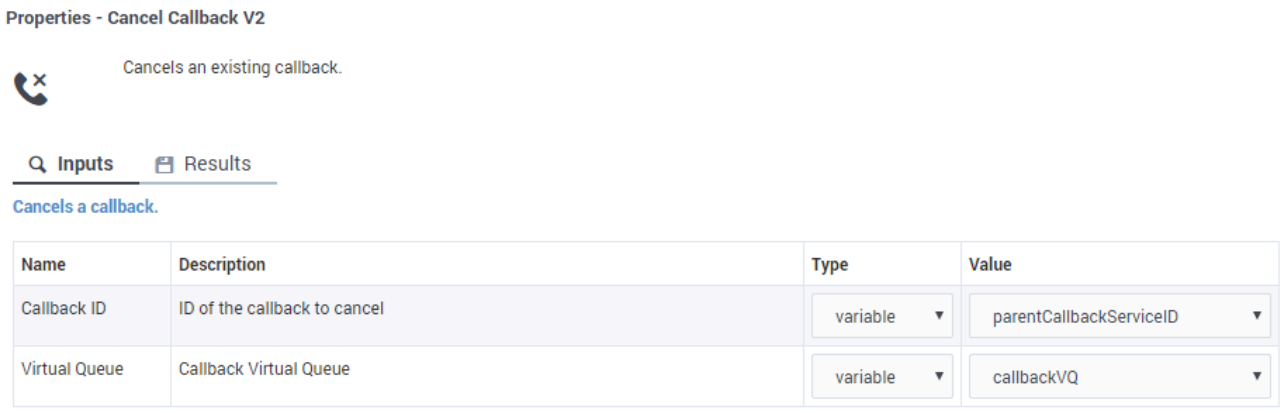
> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

This block allows you to cancel a callback.

Inputs tab

Provide the **Callback ID** and **Virtual Queue** of the callback to be cancelled.

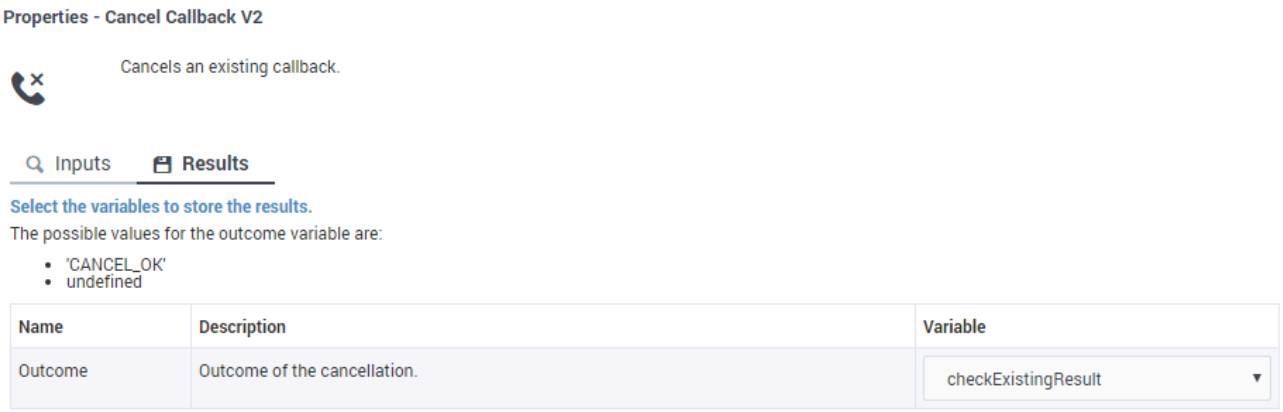
Example



Results tab

Select a variable to store the results of the cancellation request.

Example


Properties - Cancel Callback V2
Cancels an existing callback.
Inputs
Results
Select the variables to store the results.
The possible values for the outcome variable are:
'CANCEL_OK'
undefined
Name
Description
Variable
Outcome
Outcome of the cancellation.
checkExistingResult

Check for Existing Callback V2

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

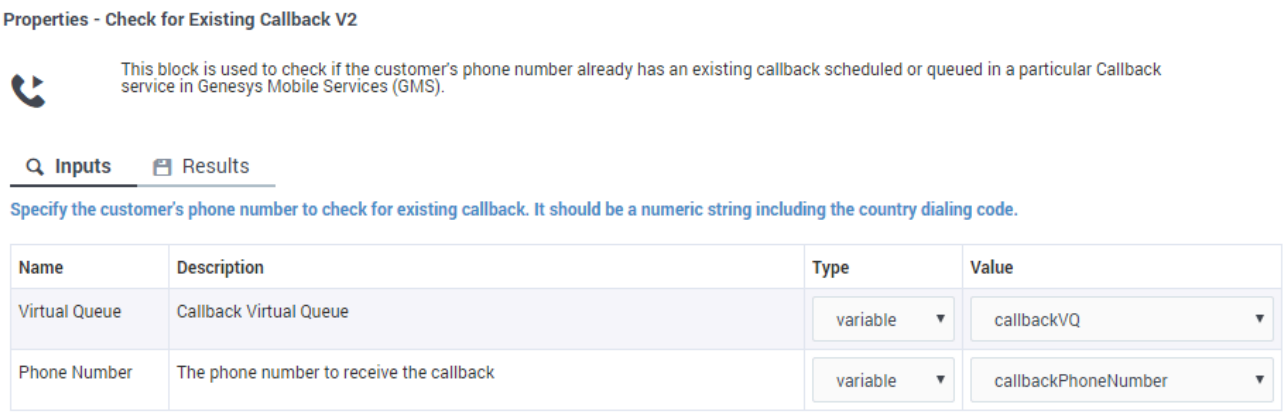
This block checks to see if an existing callback already exists for a caller's phone number in the specified virtual queue.

> **Important**
> This check is performed separately for each virtual queue. Keep in mind that if a caller is using different virtual queues, they could potentially book multiple callbacks with the same phone number.

Inputs tab

Provide the **Virtual Queue** and **Phone Number** that are to be checked for existing callbacks.

Example

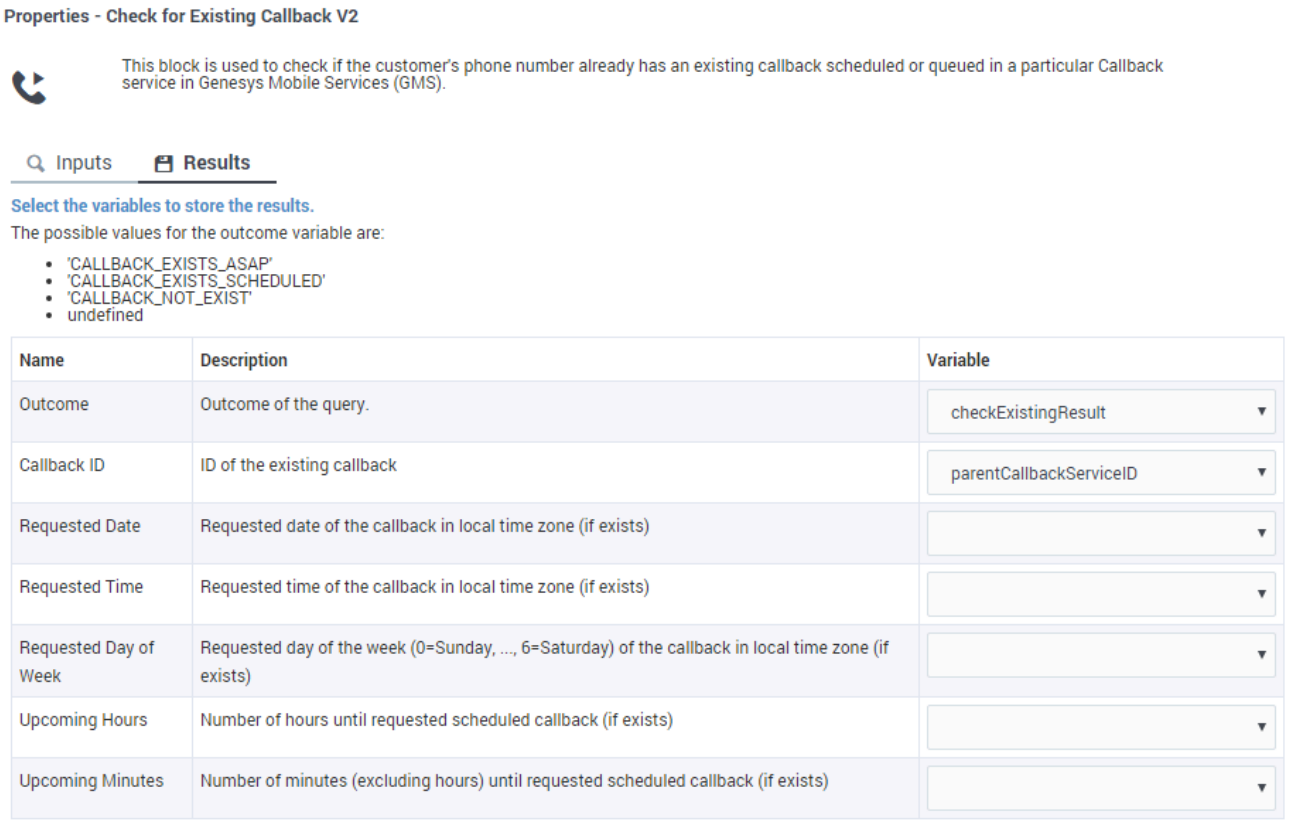


Results tab

If an existing callback with the same phone number is found in the same virtual queue, the **Callback Service ID** associated with the existing callback is returned, along with additional information such as the request date, request time, and so on.

You can use this tab to store this information in variables.

Example



Properties - Check for Existing Callback V2

This block is used to check if the customer's phone number already has an existing callback scheduled or queued in a particular Callback service in Genesys Mobile Services (GMS).

Inputs **Results**

Select the variables to store the results.

The possible values for the outcome variable are:

- 'CALLBACK_EXISTS_ASAP'
- 'CALLBACK_EXISTS_SCHEDULED'
- 'CALLBACK_NOT_EXIST'
- undefined

Name	Description	Variable
Outcome	Outcome of the query.	checkExistingResult
Callback ID	ID of the existing callback	parentCallbackServiceID
Requested Date	Requested date of the callback in local time zone (if exists)	
Requested Time	Requested time of the callback in local time zone (if exists)	
Requested Day of Week	Requested day of the week (0=Sunday, ..., 6=Saturday) of the callback in local time zone (if exists)	
Upcoming Hours	Number of hours until requested scheduled callback (if exists)	
Upcoming Minutes	Number of minutes (excluding hours) until requested scheduled callback (if exists)	

Validate Phone Number Block

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

This block provides phone number validation and international phone number support for Callback V2.

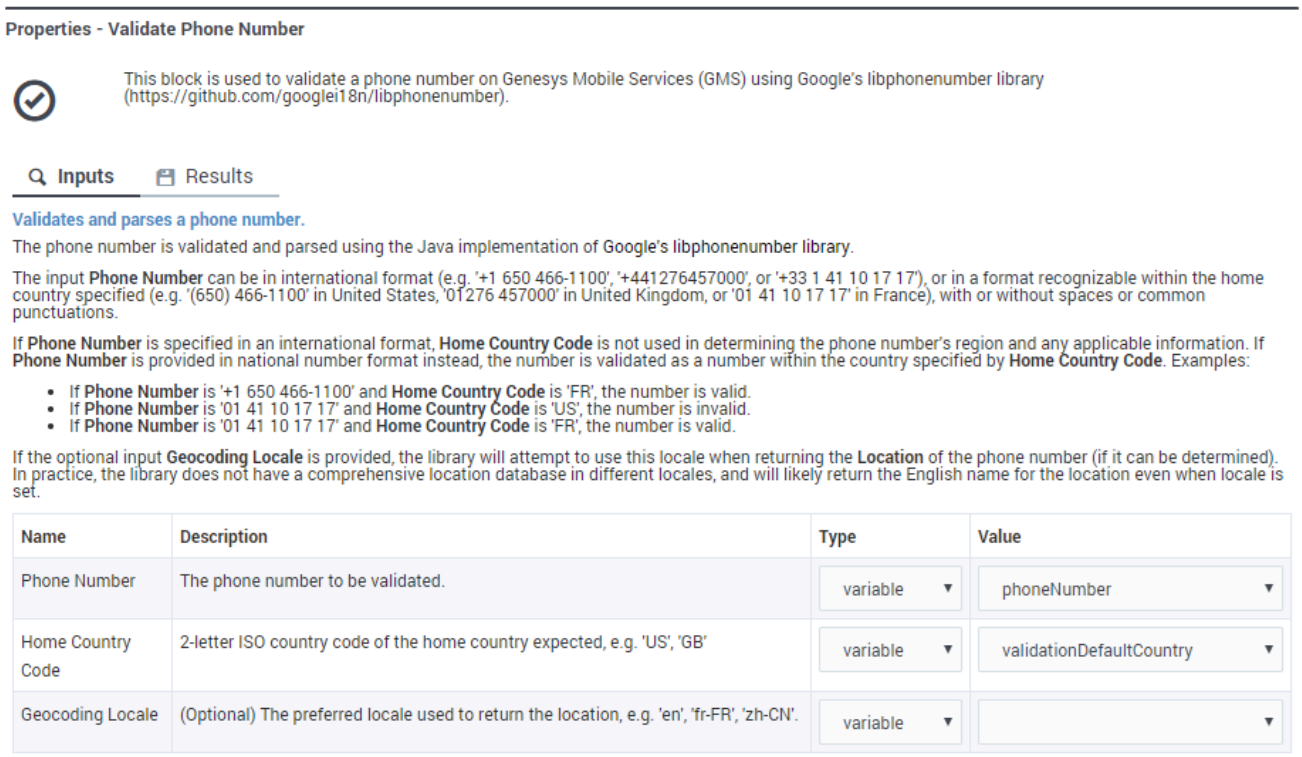
> **Important**
> This block is only supported for environments using shared GMS.

Inputs tab

The **Validate Phone Number** block has three inputs. Enter the values, or select the appropriate variables.

- **Phone number** – The phone number to be validated.
- **Home Country Code** – The 2-letter ISO code of the expected "home" country. For example, *US* or *GB*.
- **Geocoding Locale** – (Optional) The preferred locale in which to return the detected location. For example, *en* or *zh-CN*. The default is *en-US*.

Example



Results tab

Select the variables that will store the results of the phone number validation query.

All outputs are optional. If the phone number is not valid, all outputs (other than **Outcome**) will return null.

Number Validation Configurations Data Table

Callback V2 uses a special data table called NUMBER_VALIDATION_CONFIGURATIONS to provide support for phone number validation and international phone numbers.

You can view the settings for this data table by selecting it on the Data Tables page.

Parameters

This data table contains the following parameters:

Setting	Description
Config Name	Name of the configuration.
Validation Enabled	If number validation is enabled.
Default Country	The default country to be used for validation. (This should also be used for the **Home Country Code** in the Validate Phone Number block.)
Expects International Number	If callers will be prompted to include a country code when entering their phone number. **Important** If set to `true`, any countries involved in international shared-cost (e.g. +808) or toll-free (e.g. +800) numbers that are not the default country must be added to **Additional Countries Allowed**.
Always Say Country on Confirm	The country name will always be stated when confirming a phone number to the caller, even if it is the same as the home country. (It is always stated if the country name differs from the default country.)
Say Country When Not Allowed	The country name will be stated if a phone number is not from an allowed country, or use a generic message (such as "your country"). **Tip** You can set this option to `false` if you do not want your voice talent to record the names of all possible countries
Always Says International Number	When confirming a phone number to the caller, always state it in full international format. For example, say "1<xxx>5551100" instead of "<xxx>5551100"'.
Premium Rate Allowed	Whether premium rate numbers (e.g. 1-900 numbers in the U.S.) are allowed.
Toll Free Allowed	Whether toll-free numbers (e.g. 1-800 numbers in the U.S.) are allowed.
Shared Cost Allowed	Whether shared-cost numbers (e.g. +808 numbers) are allowed.
Voicemail Allowed	Whether voicemail numbers (if they can be determined) are allowed.
Pager Allowed	Whether pager numbers (if they can be determined) are allowed.
Additional Countries Allowed	Select the countries, in addition to the default country, which are allowed.

Callback VQ Watermark Block

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

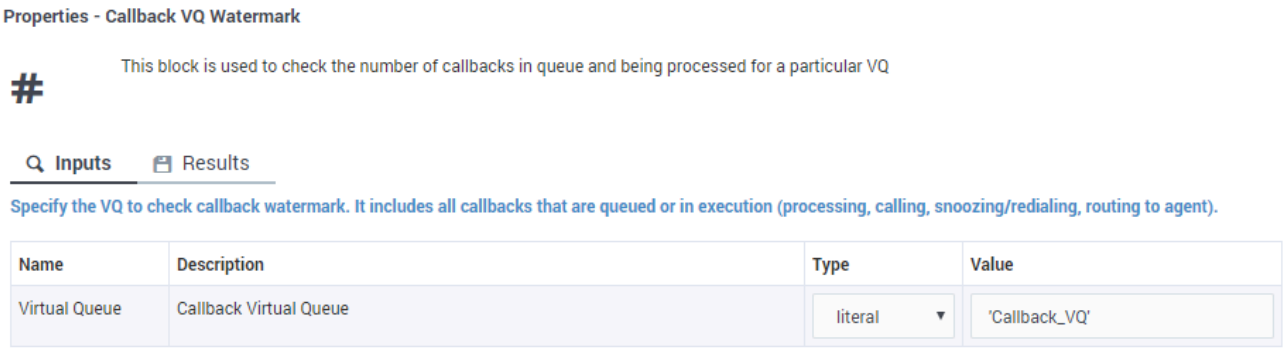
This block enables you to check the number of active callbacks that are currently queued for a specific virtual queue (VQ). The result is returned as a "watermark" value that represents the number of executed callbacks.

This can be useful for certain business scenarios, such as when you want to stop offering immediate callbacks if there are a certain number of active calls already in the queue.

Inputs tab

Specify the Callback VQ to check for callbacks that are waiting in the queue or in a state of execution (such as being dialed or routed to an agent).

Example



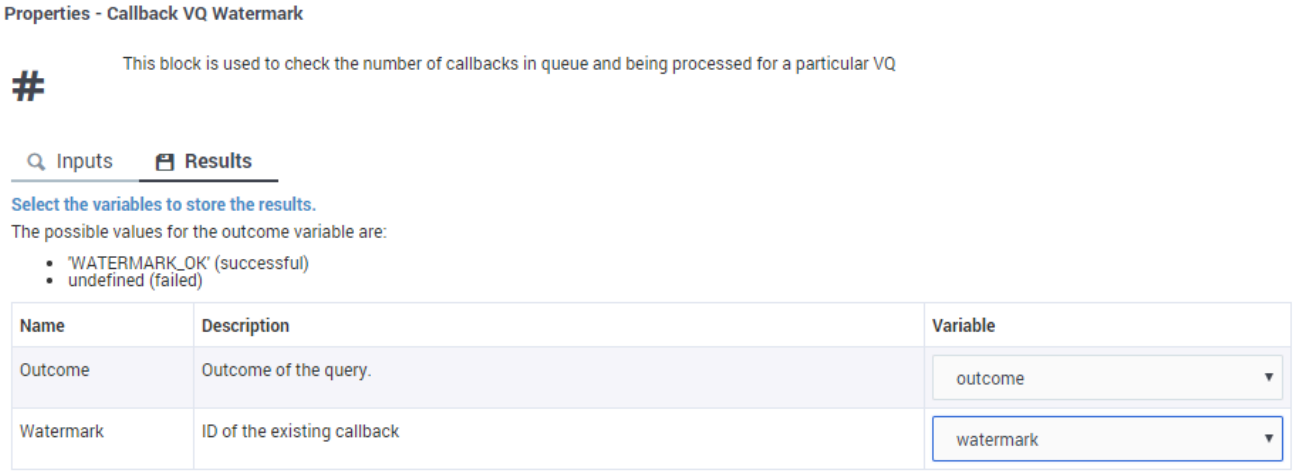
Results tab

Specify the variables that will store the results of the query.

The following outputs are possible:

- **outcome** (string)
  - **WATERMARK_OK** (if query OK)
  - **undefined** (failure)
- **watermark** (the number of executed callbacks)

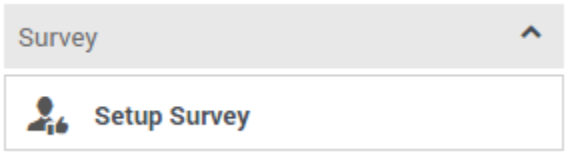
Example



Survey Blocks

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

These blocks are used to manage surveys offered to customers.



Use the links below to learn more about each block.

Setup Survey

Sets up a survey for the customer.

Used in: **Initialization**, **Self Service**, **Assisted Service** *

* Surveys are typically offered during the **Self Service** phase and completed by an accepting customer after they have finished interacting with an agent in the **Assisted Service** phase. But if you have set up your survey to be **Immediate**, the customer can complete the actual survey while still in the **Self Service** phase of the application.

Setup Survey Block

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

You can use the **Setup Survey** block in the **Assisted Service** phase to set up a survey for a customer.

Typically, you offer the survey earlier in the session, in either the **Self Service** phase or before routing begins in the **Assisted Service** phase. Then, once the customer has been served, place the **Setup Survey** block in the **Assisted Service** phase to provide the survey functions.

Once the block is set, you can choose to start the survey immediately (the customer completes the survey within the **Self Service** phase of the current application), or after the customer has finished talking to an agent (if they agreed earlier to take the survey, the customer is then sent to a survey application).

You can also choose to not start the survey if the customer rejects the offer or to not offer the survey at all.

This video demonstrates how to build a chat-based survey in Designer and how the Surveys dashboard can be used to view and track the results:

Link to video

Using this Block

In most applications, you will place a User Input block before the **Setup Survey** block and use prompts to ask the customer if they want to take a survey. You can then use a Segmentation block to segment the interaction based on the customer's response.

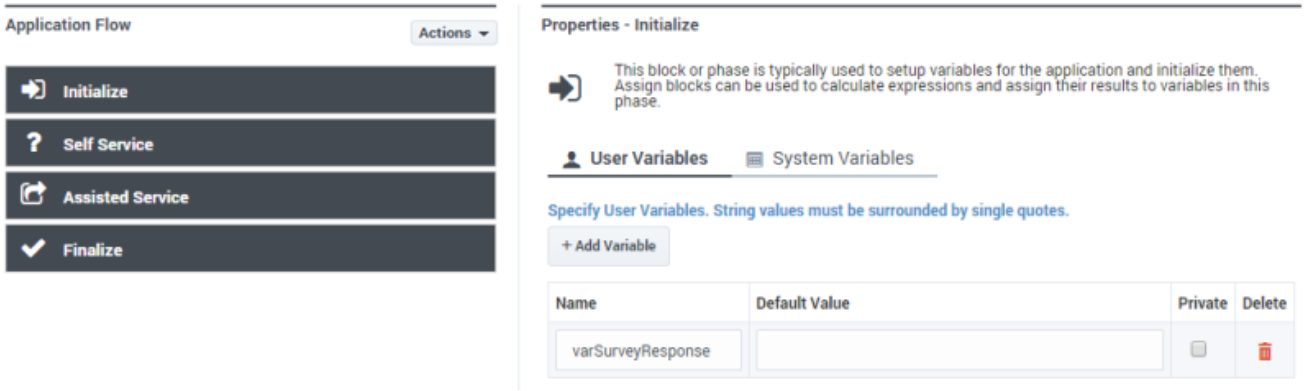
The sections below explain how to incorporate a survey into an existing application. Your application and User Variable names might differ.

> **Important**
> The examples below offer the survey in the **Assisted Service** phase, but it is also possible to offer the survey in the **Self Service** phase. In either case, the **Setup**

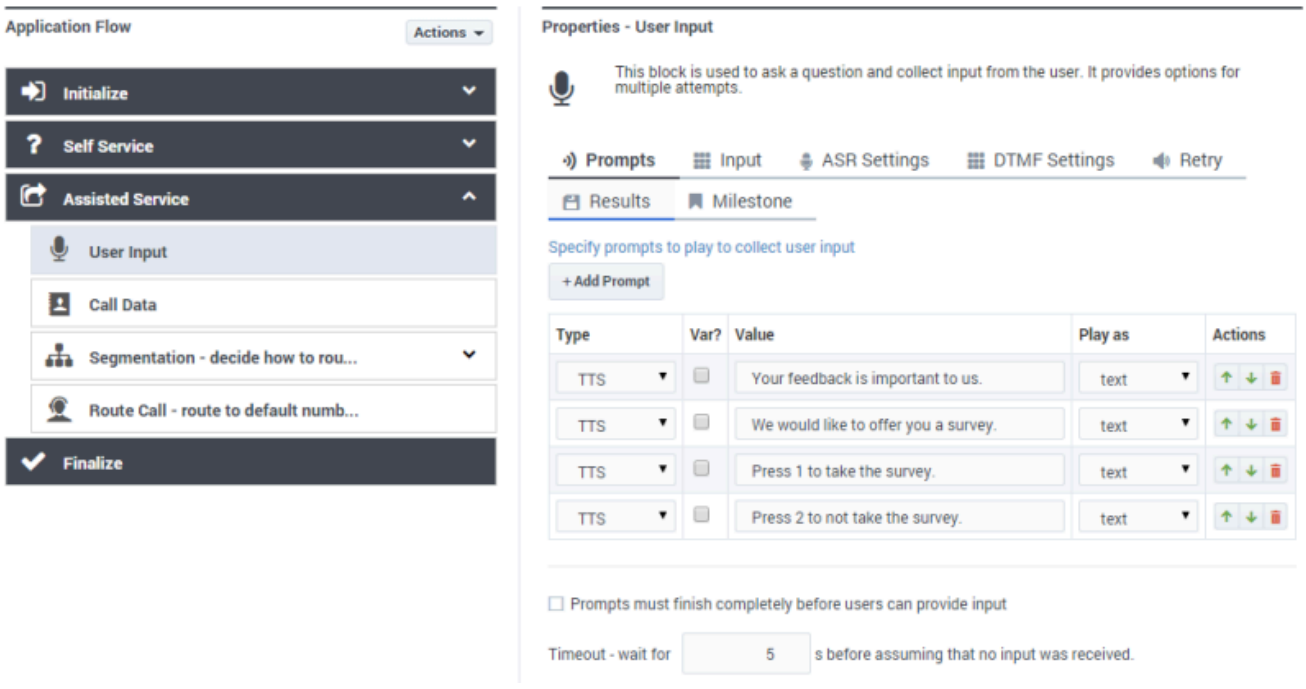
> **Survey** block must be placed in the **Assisted Service** phase.

Offer the Survey

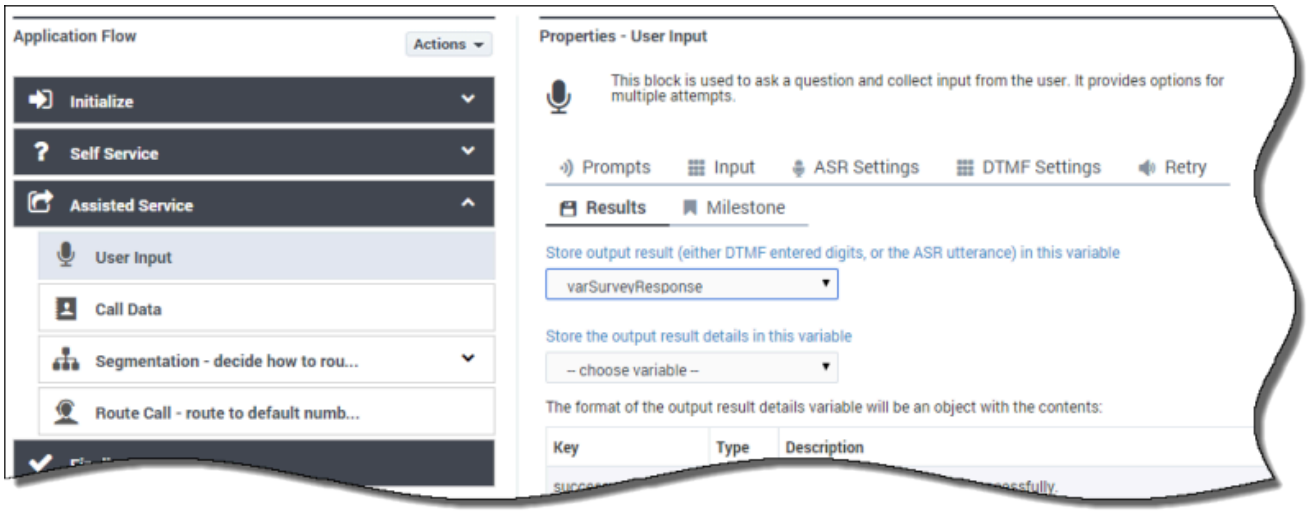
Click the **Initialize** phase and create a **User Variable** named **varSurveyResponse**.



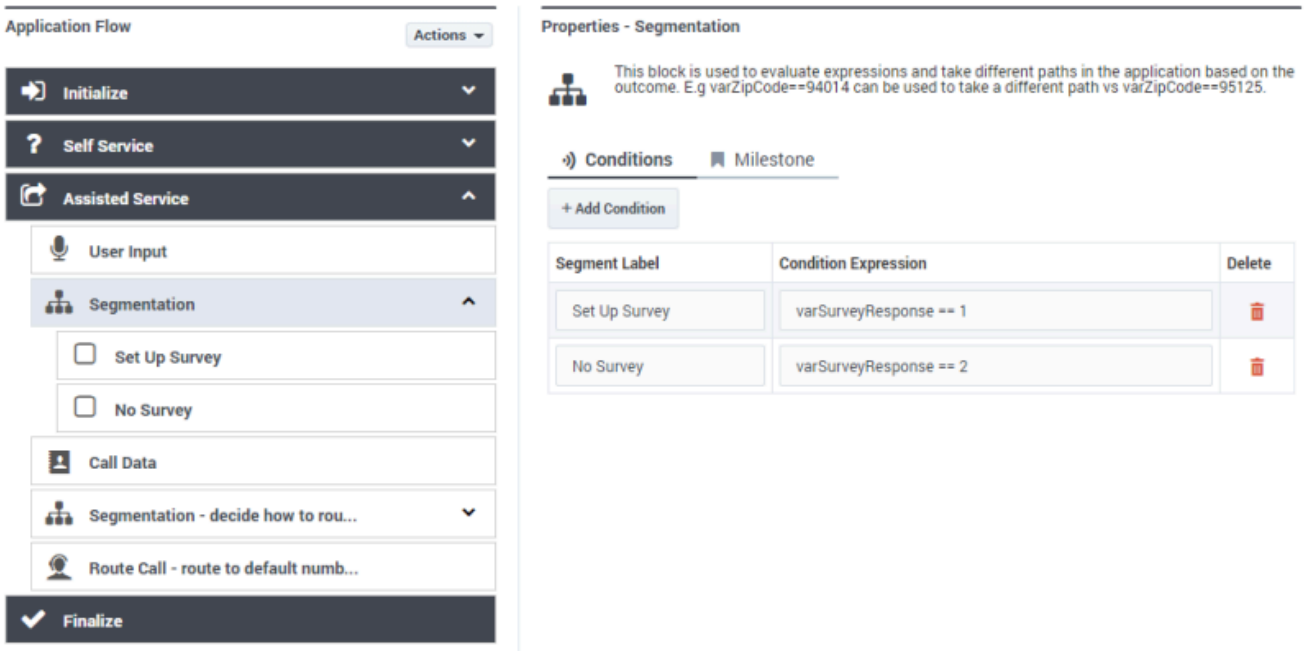
In the **Assisted Service** phase of your application, before the interaction is routed, add a **User Input** block and create a message in the **Prompts** tab. In this example, you can use the following values:



Next, in the **Results** tab, select the **varSurveyResponse** variable that you created earlier. This variable stores the input from the customer.

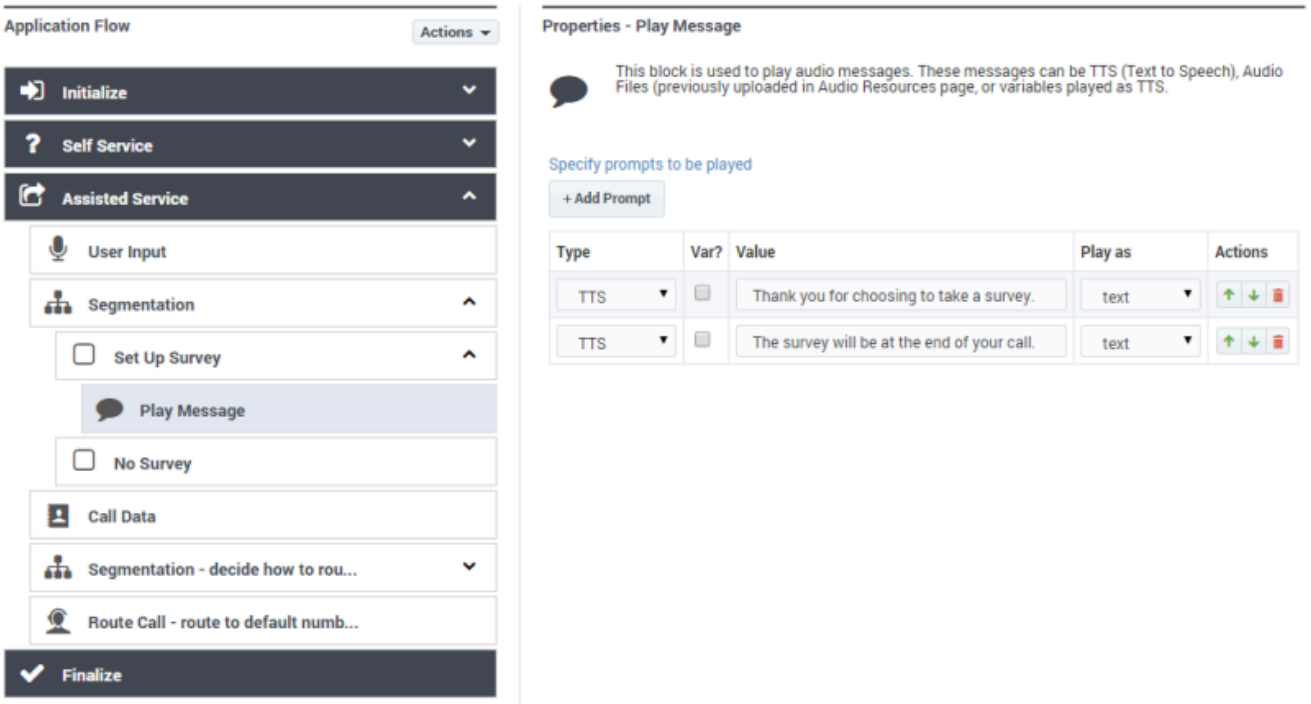


Next, place a Segmentation block to configure how your application responds to the result from the **User Input** block. In this example, configure the **Segmentation** block as shown below:

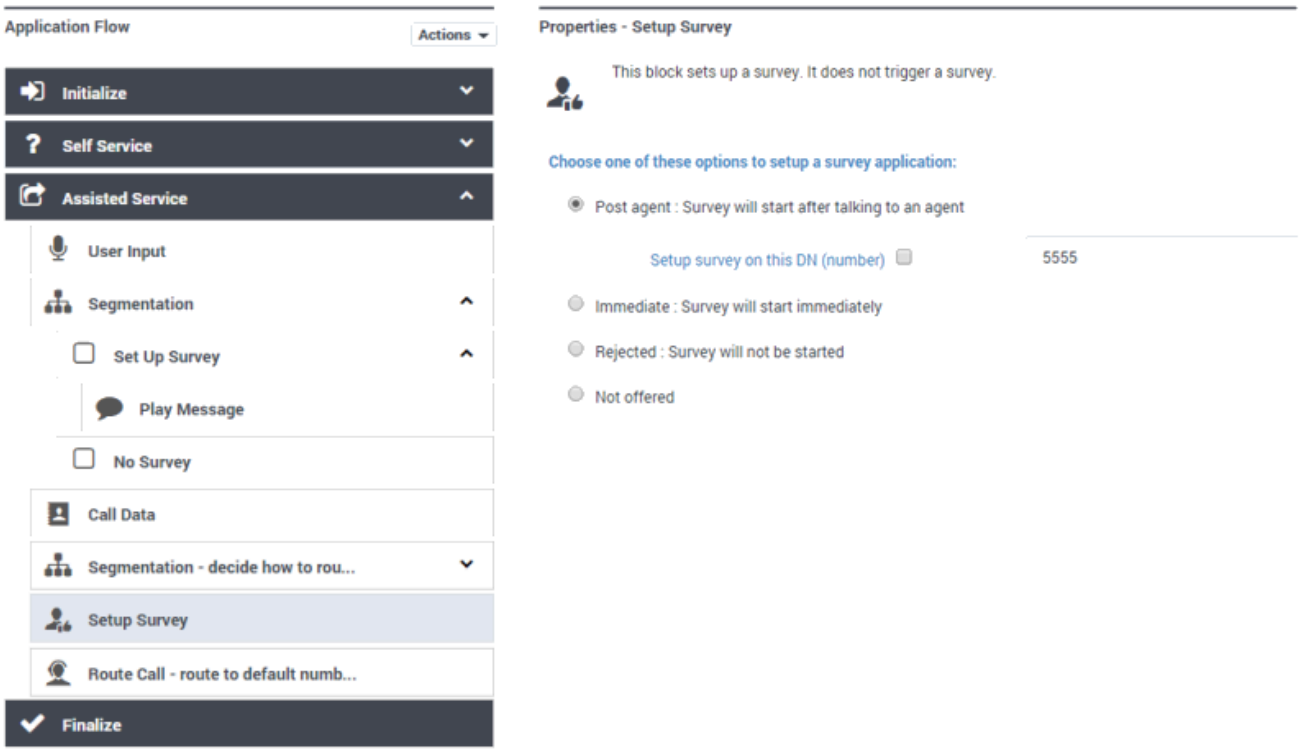


Set Up Survey Segment

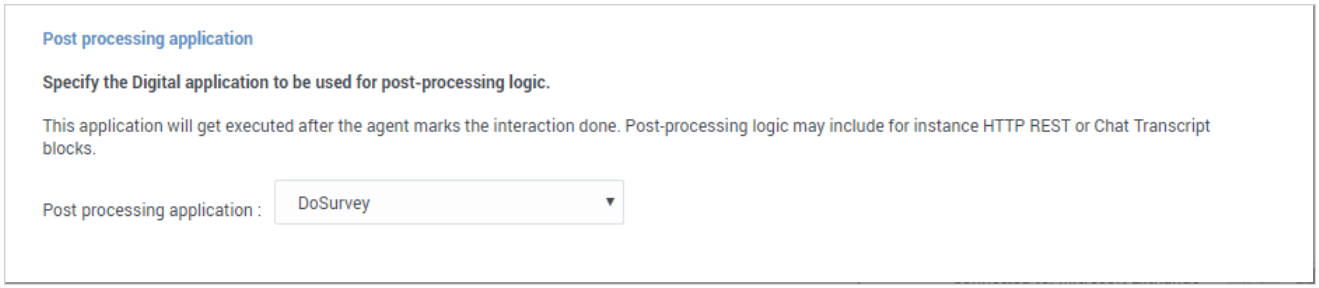
The application processes the **Set Up Survey** segment if the customer pressed **1** to accept the survey. Next, the application uses a Play Message block to thank the customer for taking the survey.



Next, place a **Setup Survey** block before the interaction is routed to an agent. For this example, we'll select the **Post agent: Survey will start after talking to an agent** option. Our example is based on a voice interaction, so we'll also enter the DN of the survey application. The example shown below uses the DN 5555, but your survey application might use a different DN. Optionally, you can enable the check box to specify the DN as a variable.



For a chat-based survey, leave the DN field blank. You can specify the name of the chat survey application in the **Advanced** tab of the Route Call (or Route Digital) block, under the **Post processing application** section.

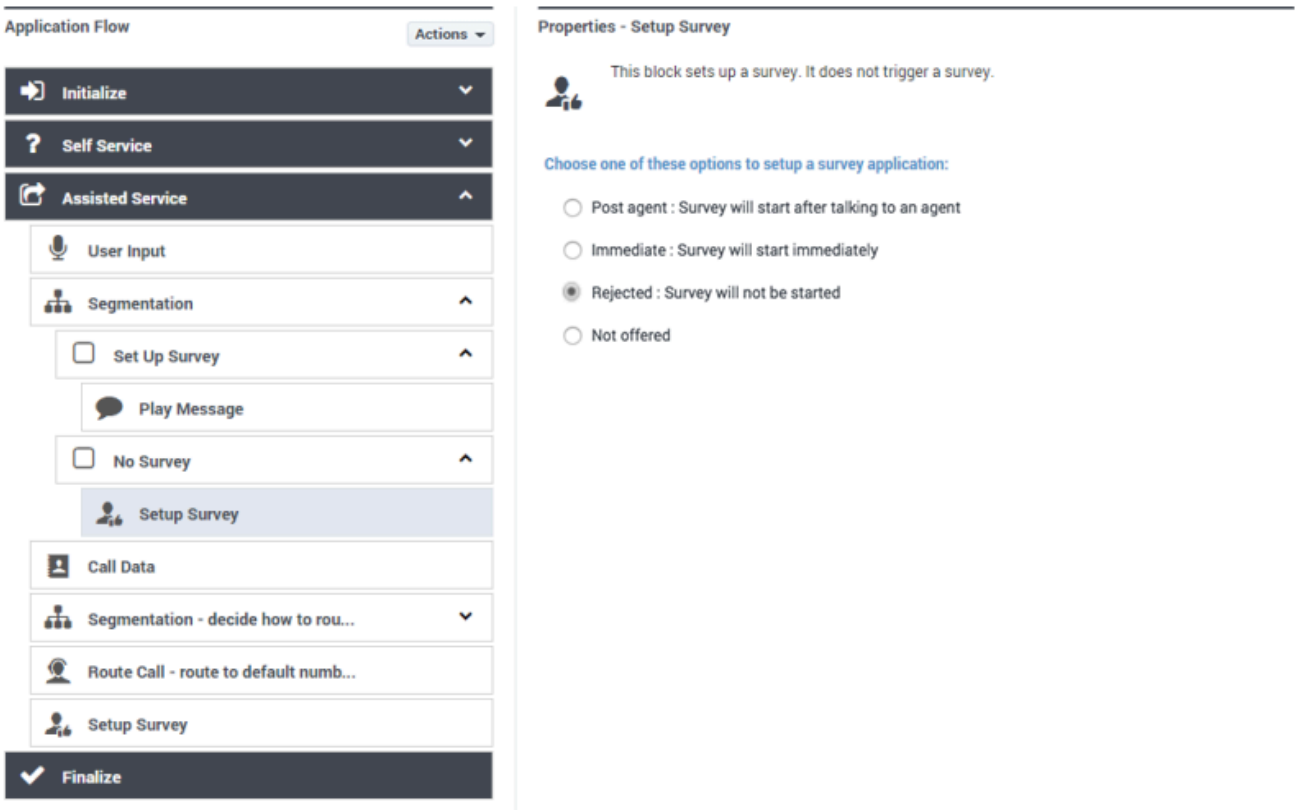


The interaction forwards to the survey application. See the Creating the Survey Application section for more information.

No Survey Segment

The application processes the **No Survey** segment if the customer pressed **2** to decline the survey.

Place a **Setup Survey** block and select **Setup was offered but it was rejected**.



Survey Not Offered

You might have noticed that a third option exists in the **Setup Survey** block - **Setup was not offered - no need to setup survey**.

For reporting, this option records that the customer was never offered a survey. This can happen for several reasons. For example, the customer might have ended the interaction early or in the **Self Service** phase, or your application might contain a segment in which it does not make sense to offer a survey.

To receive reporting in these scenarios, you must place a **Setup Survey** block in your application and select the **Setup was not offered - no need to setup survey** option to record that this interaction did not include a survey offer.

Creating the Survey Application

The actual survey takes place in a second application. For a voice-based survey, this application is loaded on the number that you specified in the **Setup Survey** block. If your survey is chat-based, it is the application specified in the **Post processing application** section of the **Route Call** (or **Route Digital**) block, which is under the **Advanced** tab.

A survey application is created with the application type **Default** and behaves in the same way as

other applications. You can use **User Input** blocks to ask questions and record responses. Each **User Input** block stores the response from the customer for reporting.

> **Tip**
>
> As survey applications are **Default** type applications, you can use Route Call and various other blocks to direct the interaction if the customer's survey responses meet certain criteria. For example, if the customer inputs a low satisfaction score, you can use a Segmentation block to check for low satisfaction scores and a **Route Call** block to route the interaction to an agent to follow up on the customer's concerns.

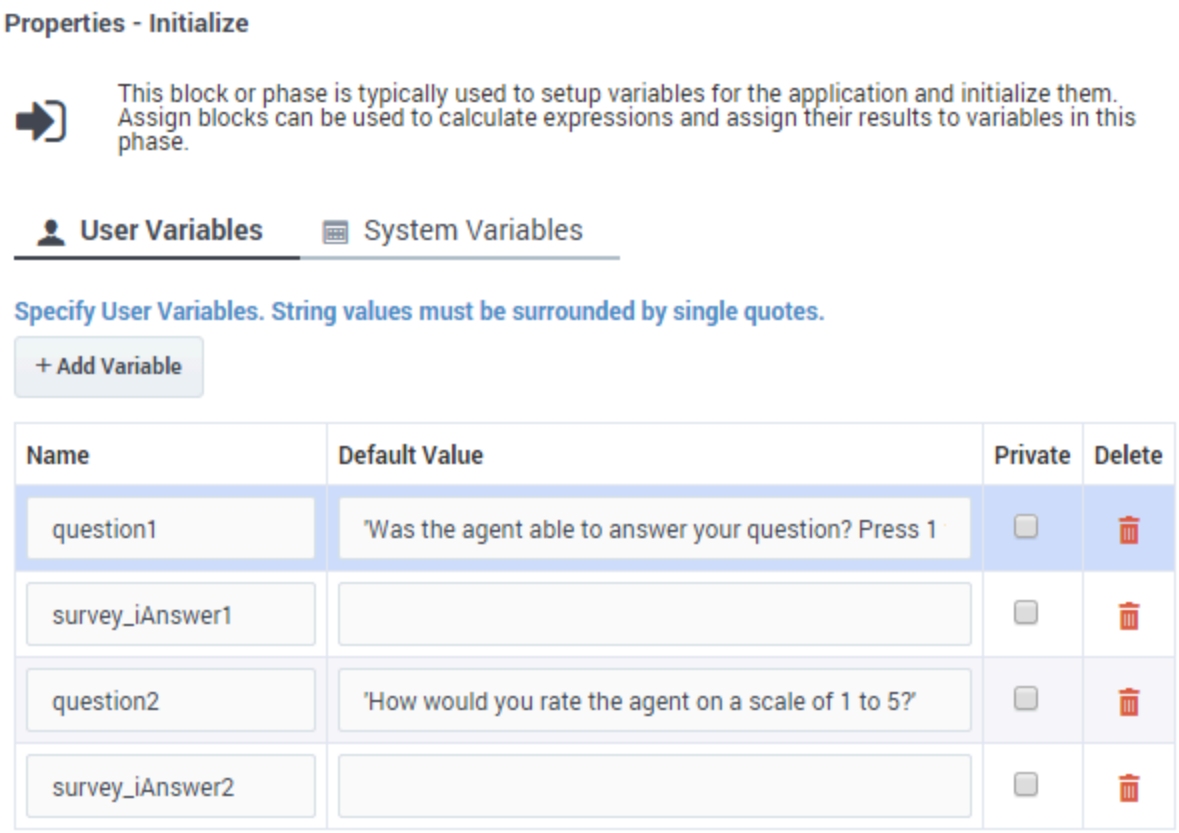
Example

The following example demonstrates how to create a simple survey application.

First, create a new application of type **Default** to provide the survey.

> **Important**
>
> Although a survey can be initiated from either a **Default** or **Digital** type application, the survey itself must be contained within a **Default** type application, as **Digital** type applications are not able to accept customer inputs).

In the application, create a series of variables to hold the questions and answers for your survey. In the example below, **question1** and **question2** hold the question that the survey asks the customer, and **survey_iAnswer1** and **survey_iAnswer2** holds the answer from the customer.



Designer also provides standard variables, which you can view in the **System Variables** tab, that you can use if your company uses standard reporting. For example, instead of using **survey_iAnswer2** to hold the feedback score for the agent, we could instead use **survey_iAgentScore**.

Variable	Editable	Purpose
survey_sOffer	No	Specifies whether a survey was offered, accepted, or rejected. This variable is set by the **Setup Survey** block.
survey_iRecommendScore	Yes	A rating (on a scale from 0 to 10) that indicates if the company, product, or service is recommended. This variable is used for calculating the Net Promoter Score (NPS).
survey_iAgentScore	Yes	Specifies a user satisfaction score for the agent (if this question is asked in the survey).
survey_iCompanyScore	Yes	Specifes a user satisfaction score for the company (if this question is asked in the survey).
survey_iCallScore	Yes	Specifies a user satisfaction

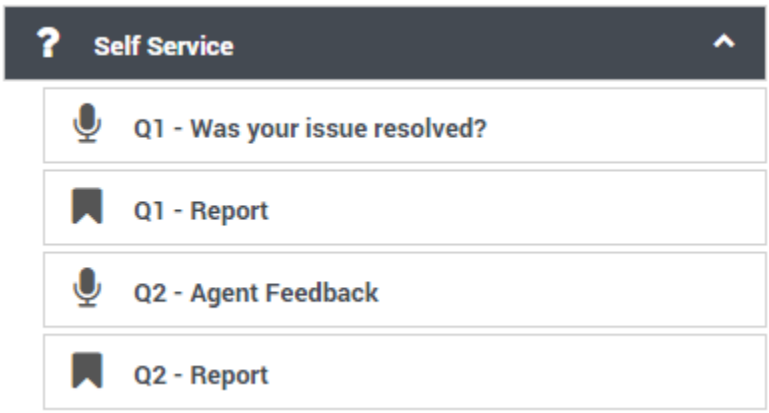
Variable	Editable	Purpose
		score for the entire session (if this question is asked in the survey).
survey_iProductScore	Yes	Specifies a user satisfaction score for the product (if this question is asked in the survey).
survey_sQ1..10	Yes	You can create these variables (1-10) to store **string**-type survey responses that will be used for reporting. (Use the naming convention as shown. For example, *survey_sQ1*, *survey_sQ2*, and so on.)
survey_iQ1..10	Yes	You can create these variables (1-10) to store **integer**-type survey responses that will be used for reporting. (Use the naming convention as shown. For example, *survey_iQ1*, *survey_iQ2*, and so on.)

Important

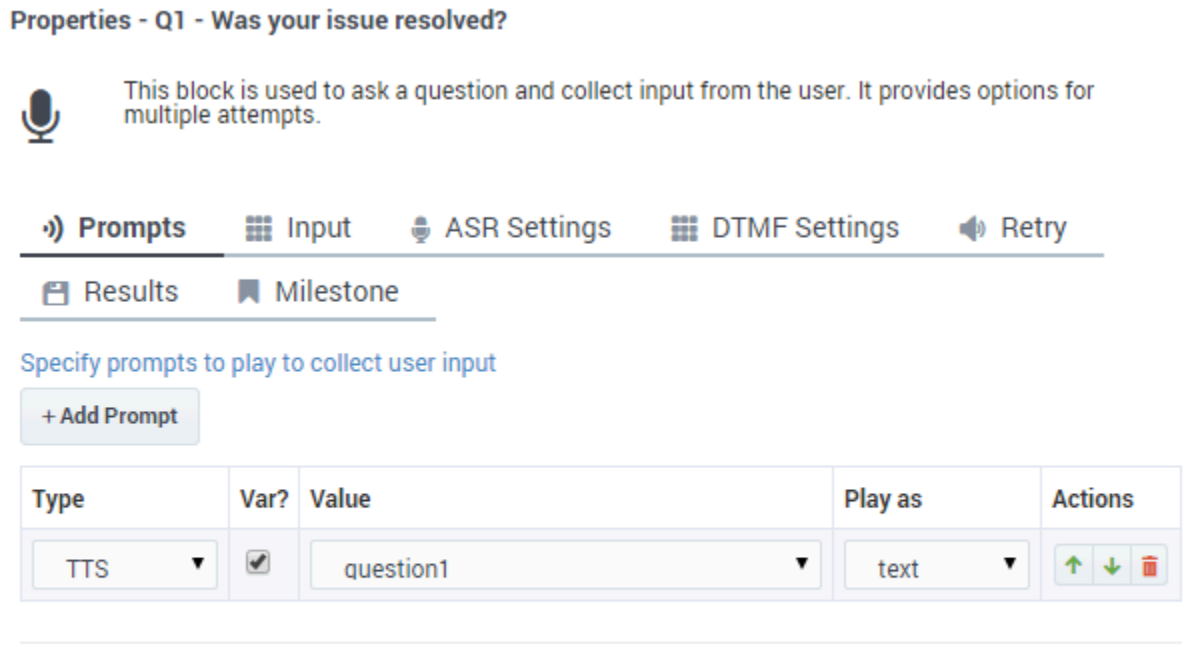
Survey answer variables must use the following naming convention:

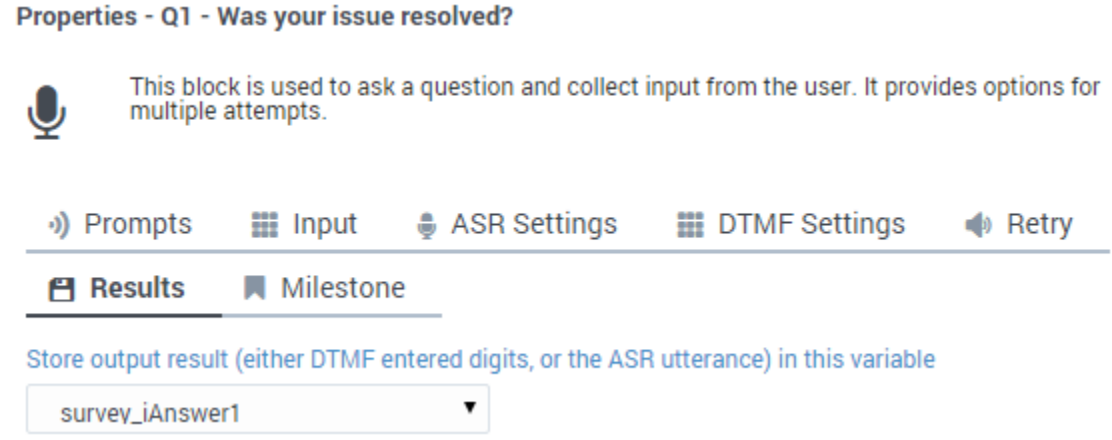
- The name must have the prefix `survey_`.
- The next character must indicate the data type (for example, `i` for integer or `s` for string).
- Example: `survey_iAnswer`.

Next, we add a series of User Input blocks and Milestone blocks to the **Self Service** phase. The **User Input** block asks the survey question and the **Milestone** block reports the survey answer.

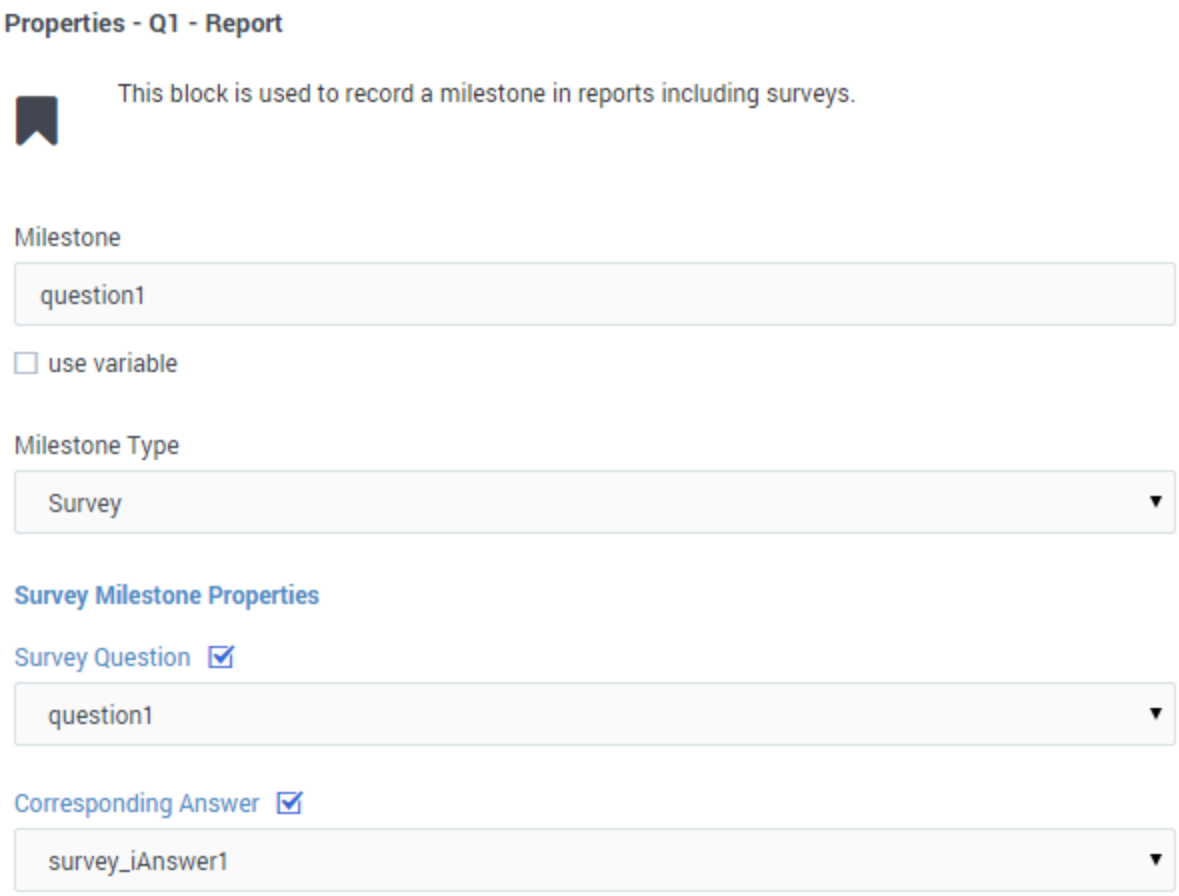


In each **User Input** block, select the question variable in the **Prompts** tab and answer variable in the **Results** tab.

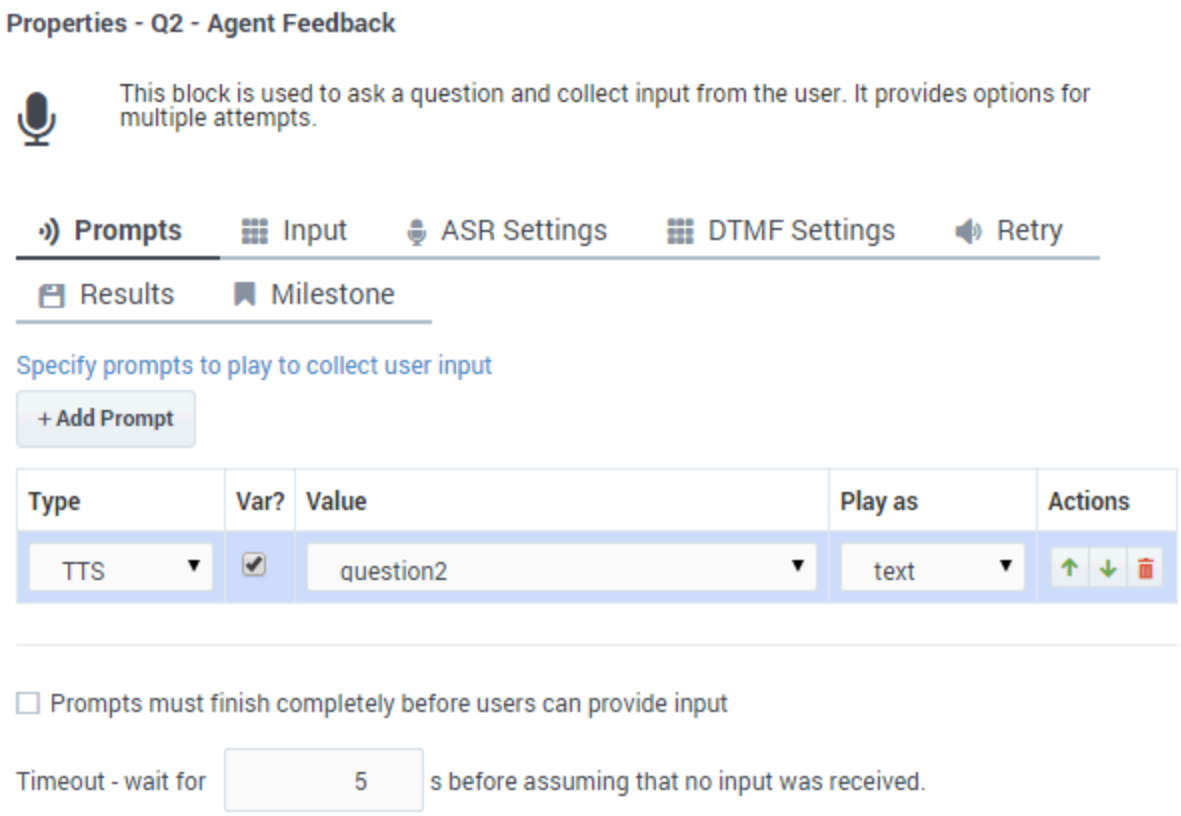


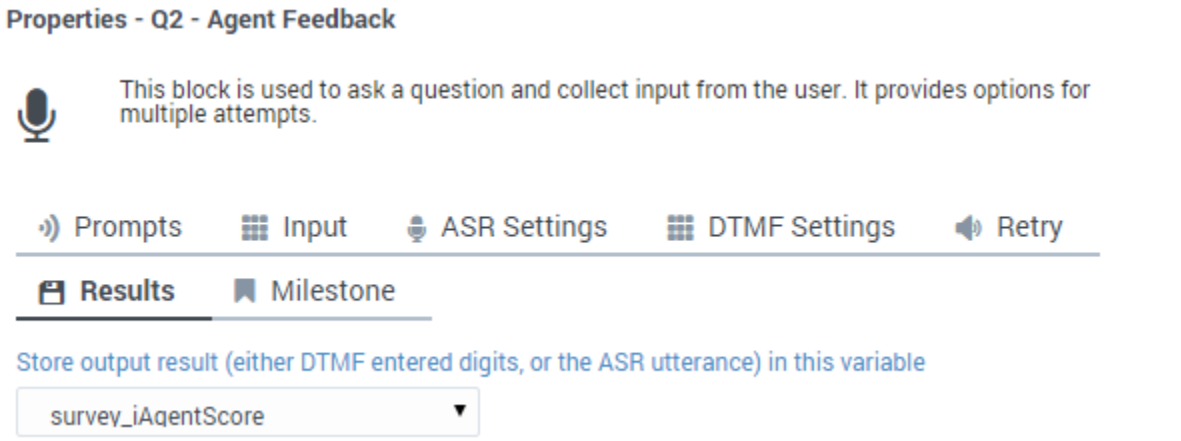


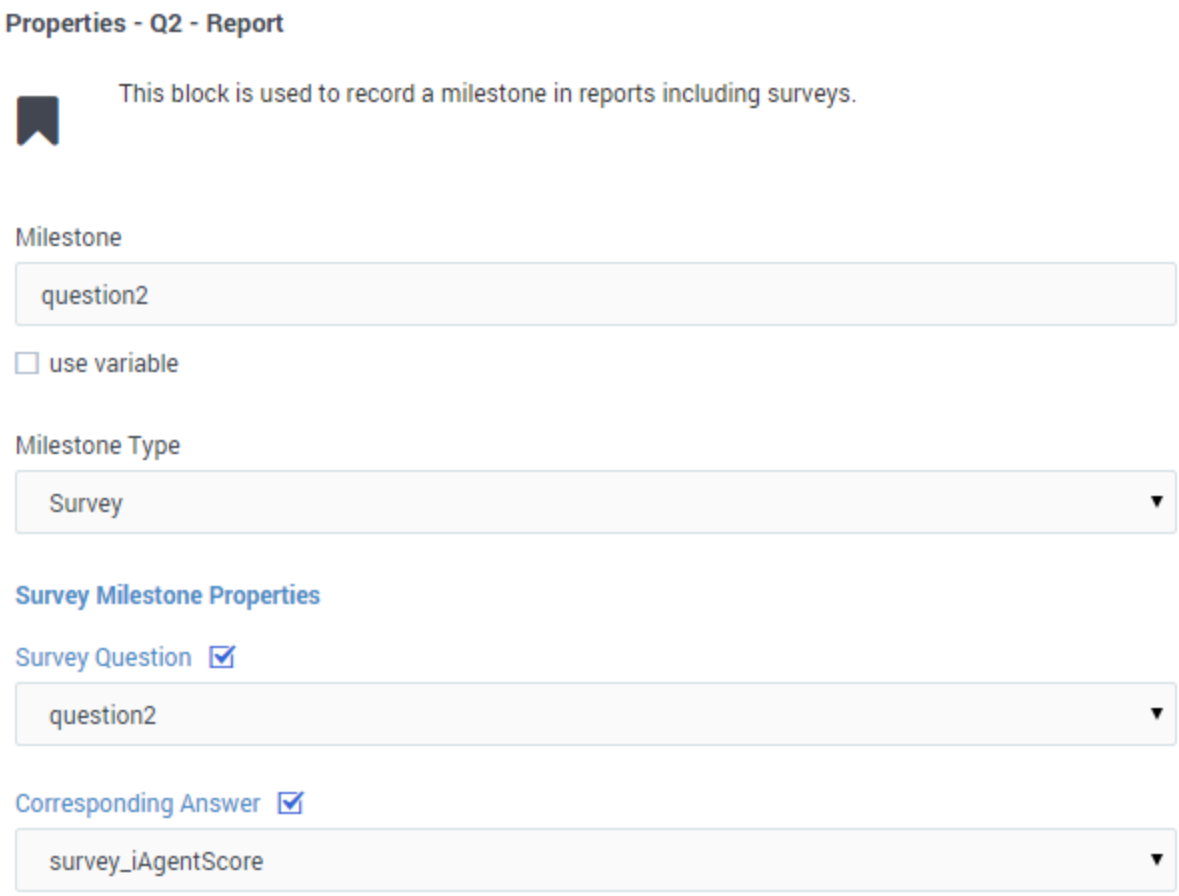
In each **Milestone** block, select the question and answer to send to reporting.



The following graphics show the process for survey question two, using the standard answer variable **survey_iAgentScore**.


Properties - Q2 - Agent Feedback
This block is used to ask a question and collect input from the user. It provides options for multiple attempts.
Prompts
Input
ASR Settings
DTMF Settings
Retry
Results
Milestone
Specify prompts to play to collect user input
+ Add Prompt
Type
Var?
Value
Play as
Actions
TTS
question2
text
Prompts must finish completely before users can provide input
Timeout - wait for
5
s before assuming that no input was received.


Properties - Q2 - Agent Feedback
This block is used to ask a question and collect input from the user. It provides options for multiple attempts.
Prompts
Input
ASR Settings
DTMF Settings
Retry
Results
Milestone
Store output result (either DTMF entered digits, or the ASR utterance) in this variable
survey_iAgentScore


Properties - Q2 - Report
This block is used to record a milestone in reports including surveys.
Milestone
question2
use variable
Milestone Type
Survey
Survey Milestone Properties
Survey Question
question2
Corresponding Answer
survey_iAgentScore

Designer Analytics

> **Important**
>
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

Designer Analytics is a powerful tool that provides a rich overview of your contact center operations. It features a series of informative dashboards, each of which offers a variety of visualizations and in-depth reporting panels that highlight specific aspects of your operations.

You can track calling trends, monitor how callers are interacting with the applications, and quickly notice and react to any potential issues with the applications or system platform. It can quickly answer questions you might have about your operations, such as:

- How long are customers waiting for an agent?
- How many customer interactions did we receive yesterday? Last week? Last month?
- How many of our customers are contacting us from North America? Europe? Asia?

Key features

- Almost real-time reporting means that as soon as an application session ends, Designer Analytics can start using the data to build reports.
- 90-day data retention, so you can see how your applications are performing over time.
- Advanced filtering options, so you can focus on the data you want to see.

Session Detail Records

The data contained in the Session Detail Records (SDR) is the "secret sauce" that Designer uses to generate the reporting data shown on the dashboards.

Each time an interaction is processed by an application, Designer creates a SDR. The fields within the SDR capture important details about the interaction, such as the starting time of the call, source and destination numbers, the block sequence (or path) that the caller took through the application, and the final status of the call (for example, the caller hung up or was connected to an agent).

Important

- Designer receives SDR information from a variety of sources. These details are often added to reports in the order they are received by Designer and not according to their individual timestamps. For example, a report showing a sequential block path might make it appear as if a block that starts a treatment was executed *after* an interaction had ended, but the detail about the treatment block might have been received by Designer after it had already added the information about the interaction being ended.
- A *session* is not the same thing as a *call*. Sessions are started each time a call (or interaction) is processed by an application. If an interaction is processed by multiple applications (or processed multiple times by the *same* application), multiple SDRs are created.

Designer assigns each interaction a unique ID that follows it through each session that is created, thus enabling you to track the entire journey of an interaction from start to finish, across each application that handles it. This makes SDRs useful for call flow analysis and troubleshooting.

Dashboards

Designer includes several system dashboards that you can start using right away.

Each dashboard contains reporting panels that focus on a particular aspect of your operations. For example, panels might display results based on milestones, system errors, or the paths that callers took through an application.

Many of the panels also have options for viewing additional details about the data displayed (such as the query used to generate the results) or for changing the panel properties.

For more information about working with the dashboards, see Dashboard management.

System dashboards

The following system dashboards are included with Designer and ready to use. Use the dashboard icons to quickly navigate between the different types:

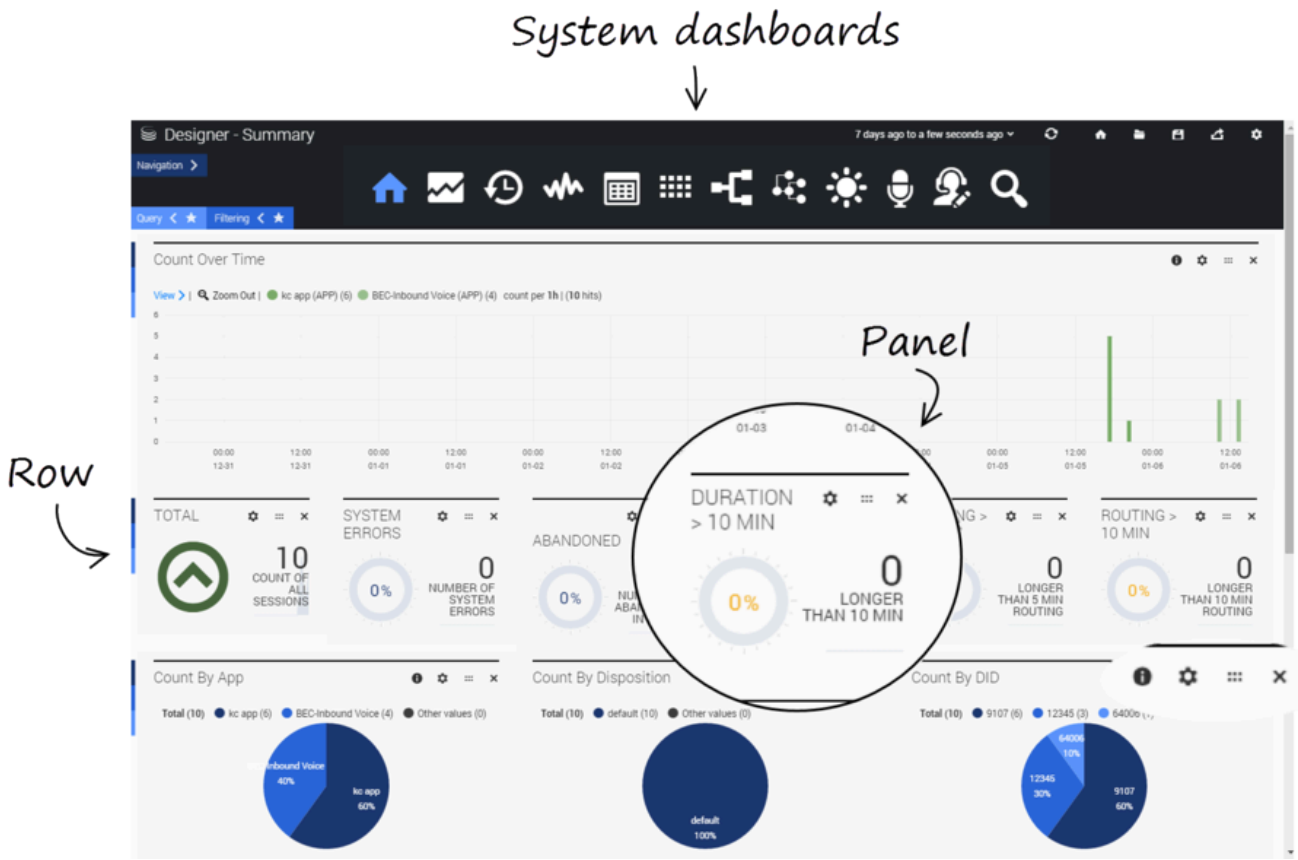
	Summary provides a high-level overview of how application sessions are being handled across the platform.
	Application Details provides how customers are moving through the application flows, such as milestones reached, activities completed, and paths taken.

	Durations shows how much time customers are spending in various parts of the applications.
	Data Tables displays disposition information for your applications in a spreadsheet-like table format.
	Spikes displays the peaks in your application session counts over a specific period of time.
	Heatmap provides the intensity in occurrence (or "heat") of a particular item or event.
	Path provides a visualization of how customers are moving through an application.
	Sankey Path Analysis is similar to Path, but generates the results based on milestones and menu inputs.
	Sunburst Path Analysis renders your reporting data as a sunburst graphic.
	Inputs provides information about how customers are responding to the various menu options.
	Surveys provides a deeper look into the performance of your survey applications.
	External Services provides information about external requests made by the applications.
	Routing Analysis provides details about routing sessions.
	Business Controls provides details about Business Controls objects.
	Bots provides details about the performance of your voice and chat bots.
	Session Detail Records displays some of the raw data contained in the Session Detail Records (SDRs).

Dashboard management

> **Important**
>
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

Each dashboard is made up of several rows, each containing one or more reporting panels. The buttons on the main dashboard toolbar let you quickly navigate between the various types of system dashboards.

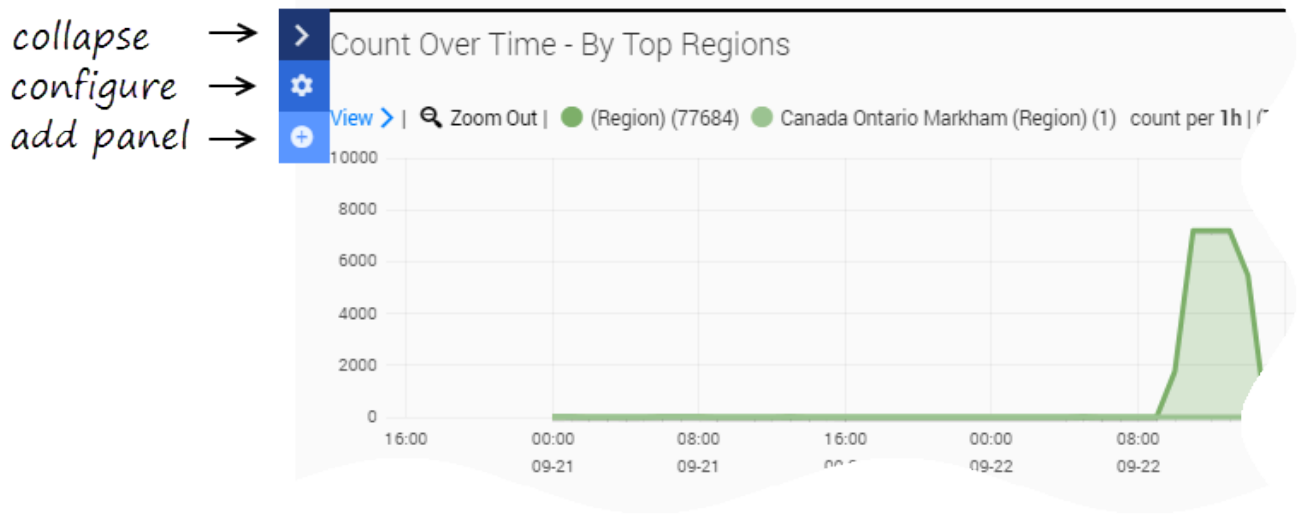


The settings for the built-in system dashboards cannot be changed.

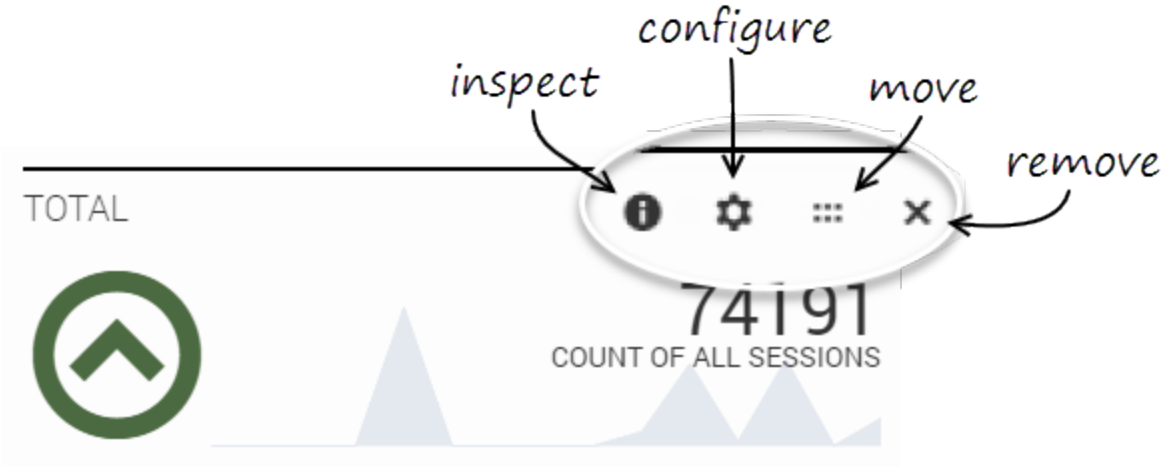
> **Tip**
> The Designer Dashboards are built on the Kibana plug-in (by elastic). For additional details that are not covered in this Help, please refer to the Kibana 3.0 documentation.

Rows and Panels

On the left side of each row is a sliding menu that remains hidden (or collapsed) when not in use. The menu slides out when you hover over it and has options to **collapse**, **configure**, or **add panels** to the row.



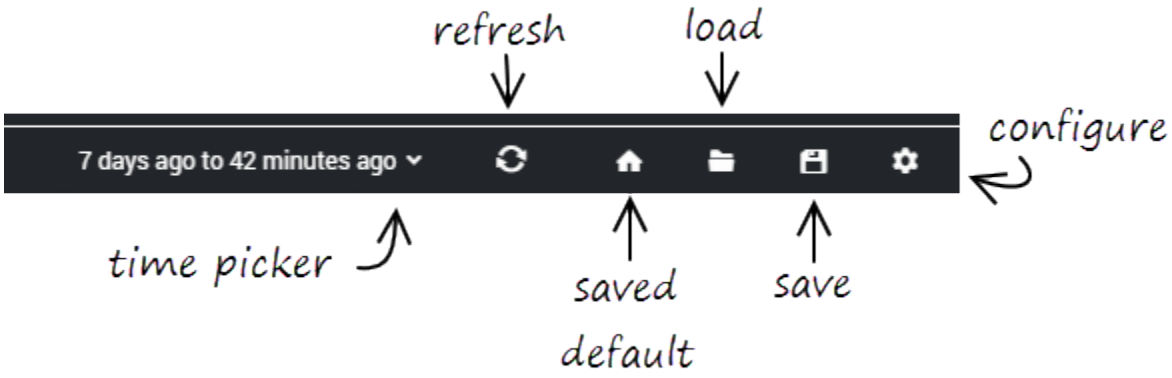
Each of the panels has options to **configure**, **move**, or **remove** the panel. Some panels also have an **inspect** icon that lets you view the query for the report being displayed.



> **Tip**
> Sometimes, after closing an **inspect** panel, it might look like the dashboard page has gone blank, especially if there was a lot of information to scroll through. If this happens, simply scroll back up to the top of the dashboard page. (This issue can be avoided by using **Esc** to dismiss an **inspect** panel instead of closing it.)

Dashboard controls

At the top of the screen are control icons for performing common tasks.



Time picker

Lets you select the time period for which you want to display data.

Refresh

Refreshes the dashboard with the most current data for the selected time period.

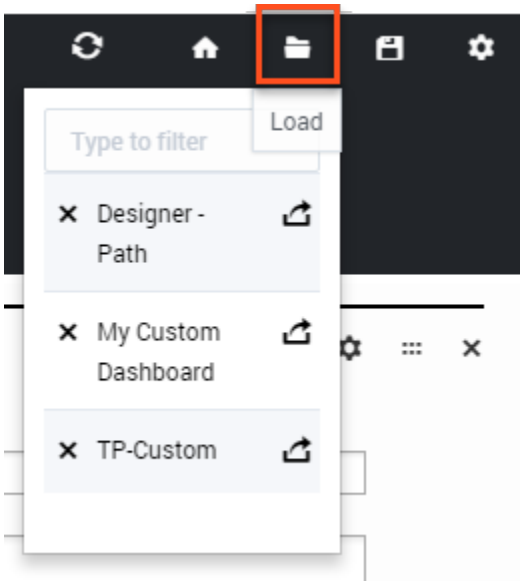
Home (saved default)

Returns to the dashboard that is currently saved as the default (or "home") dashboard.

> **Tip**
> This **Home** button is not the same as the **Home** button on the main toolbar (for the built-in dashboard types). That button will always return you to the built-in Summary dashboard, while this one will return you to the dashboard you've assigned to it.

Load

Lets you select and open any dashboards that you have saved.



Save

Saves the current dashboard. You will be prompted to give the dashboard a new name. (This option might not be available for all users or deployments.)

Configure

Customize certain properties and settings for the current dashboard.

Navigation, Query, and Filtering tabs

Toggling these tabs lets you show or hide their options.

Navigation tab

Shows or hides the dashboard icons.

Query tab

Toggling the **Query** tab lets you manually enter a query statement to search for specific terms, criteria, or conditions.

For example, you might enter the following query:

```
=applicationName=Test 2 AND finalDisposition=System Error AND ANI=703-123-1234
```

This query would search the **Test 2** application for sessions where the final disposition was **System Error** and the ANI (the caller's phone number) was **703-123-1234**.

Query statements are not automatically applied to all panels on the dashboard. To use a query statement on a panel, go to the panel's **Configure** settings and select it from the **Queries** tab.

Filtering tab

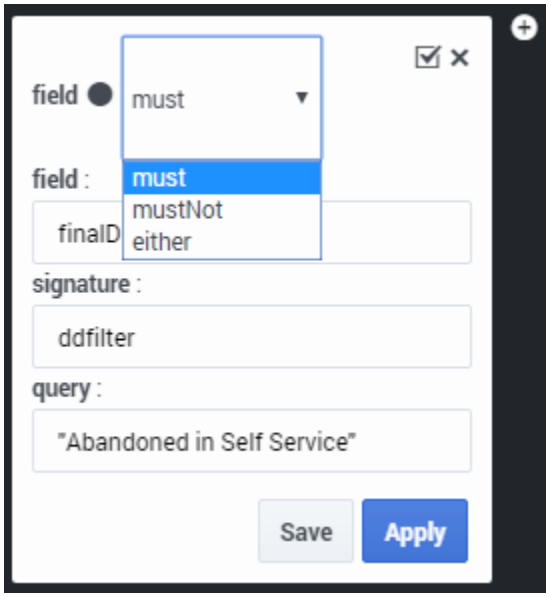
Toggling the **Filtering** tab lets you see any filters that are currently applied to the dashboards.

> ### Important
> In most cases, any filters that you define are applied across *all* of the system dashboards, not just the dashboard you are currently viewing.

Whenever you select a value from the Filter panel, or click a specific time range or section of a pie or bar chart, a filter is automatically created. In this example, we can see that an **Abandoned in Self Service** filter was applied:

With the filters visible, you can easily modify, remove, or turn a filter on or off (the solid dot indicates the filter is currently *on*).

You can also make a filter *inclusive* or *exclusive* by clicking the field and choosing one of the options:



Summary

> **Important**
>
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

The **Summary** dashboard gives you a quick overview of what's happening with your application sessions.

You can use the information provided here to see if there are any patterns that can give you insight into the sessions being initiated on Designer. For example, you might look at sessions over a period of time (such as the last 24 hours, a week, or a month) to see how sessions are being handled across various applications, what's happening to those sessions, and how much time callers are spending in various stages of applications.

Reports on this dashboard

Count Over Time

This report shows you the patterns of traffic that are coming onto the platform over a period of time. You can quickly adjust the range of time shown, say, for the last hour, 24 hours, or the last week, to see how sessions are being initiated on the platform.

Each bar indicates the total number of sessions for each application that took place during the given time period. The higher the bar, the more sessions that ran. You can easily see what time of the day (or what day of the week) that traffic is higher or lower, and organize your business accordingly.

Total

The total number of sessions or interactions that were processed.

System Errors

The percentage of sessions (out of all sessions) that had system errors. (A system error does not necessarily mean that your callers experienced any issues with the applications. This report helps you to notice any possible issues with the system platform.)

Abandoned

The percentage of sessions that are being abandoned while callers are in the Assisted Service (or routing) phase. A higher number might indicate that callers are waiting too long and hanging up before they can be connected to an agent.

Duration > 10 Mins

The percentage of sessions where a caller spent more than 10 minutes combined in both the Self-Service and Assisted Service phases. This gives you an overall look at how long it's taking for callers to be processed.

Routing > 5/10 Mins

The percentage of sessions where a caller spent more than 5/10 minutes in the routing (or Assisted Service) phase. This lets you see if there are any potential issues that might be causing callers to be stuck in the routing phase for too long.

Count by Apps

This report provides distributions of sessions across the various applications that were built on Designer. This data could be useful in allocating resources based on the traffic across applications.

Count by Disposition

This report provides distributions of sessions across the various disposition codes. A *disposition* represents the status of a call at the time it exited the call flow, such as whether it was routed to an agent or the caller hung up.

This data could be useful in finding out how your application sessions are actually performing. For example, if a high amount of sessions are getting abandoned in Self Service, you can check if there is an application error or some other reason why those calls aren't getting sent to agents.

Common dispositions:

default
This code is used when no other disposition code is applicable. For example, it could indicate that a call was not routed, not terminated (by any party), and was likely still active when the session and/or application completed.

System Error
There was an unexpected error in the application (such as a script validation error).

Application Timeout
The application got stuck in a loop and reached the timeout limit.

Terminated - <reason>
The call was ended due to a certain condition, such as:

- **Terminate Call** — the application reached a Terminate Call block.
- **Business Hours** — the call came in outside of regular business hours (if set up this way in the Business Hours block).
- **Special Days** — the call came in on a special day (if set up this way in the Special Day block).
- **Emergency** — the emergency flag was set (if set up this way in the Emergency block).
- **Menu Option** — the caller chose a menu option to exit or end the call.

Abandoned in Self Service
The caller hung up before completing the Self Service phase of the application.

Abandoned in Queue
The caller completed the Self Service phase, but hung up while waiting to speak with an agent.

Completed in Self Service
The caller successfully completed their call in the Self Service phase, i.e. the application did not need to go to Assisted Service and went directly to Finalize. If the call ended in Self Service due to some other condition or event (such as the call being received outside of business hours or the customer hanging up), then the appropriate disposition code for that condition or event is used (**Abandoned in Self Service**, **Terminated**, etc.).

Routed to Agent
The interaction was successfully delivered to an agent.

Routed to DN
The call was successfully delivered to a direct number.

Count by DID

This report provides details about the distributions of sessions across various Dialed Numbers (DNIS). Typically, a DNIS represents a department or line of business.

Dispositions by App

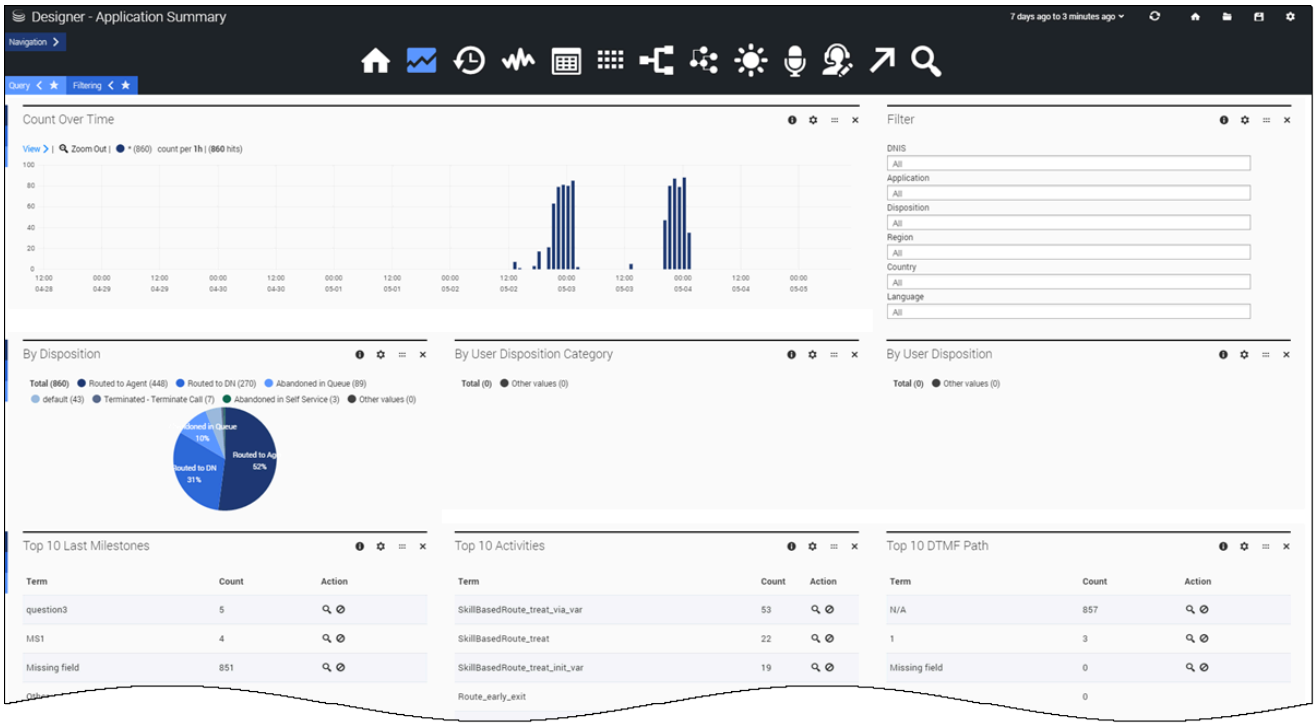
This report provides distributions of sessions across two parameters, **Disposition Code** and **Applications**. This give you a quick look into how sessions are distributed across the various applications and what their disposition codes are.

This could be very handy if you want to compare sessions with certain disposition codes across several applications. For example, if **Abandoned in Queue** for Application A is higher than that of Application B, you might want to think about adjusting the resources assigned to handle calls coming into Application B.

Application Details

> **Important**
>
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

The **Application Details** dashboard gives you a closer look at how callers are moving through the application flows.



Most of the reports on this dashboard are focused on the milestones reached, activities completed, and paths taken by callers as they move through the application flows. You can quickly see if there are any unexpected deviations (such as a sudden rise or drop in certain milestones) that might require further investigation.

Reports on this dashboard

Count Over Time

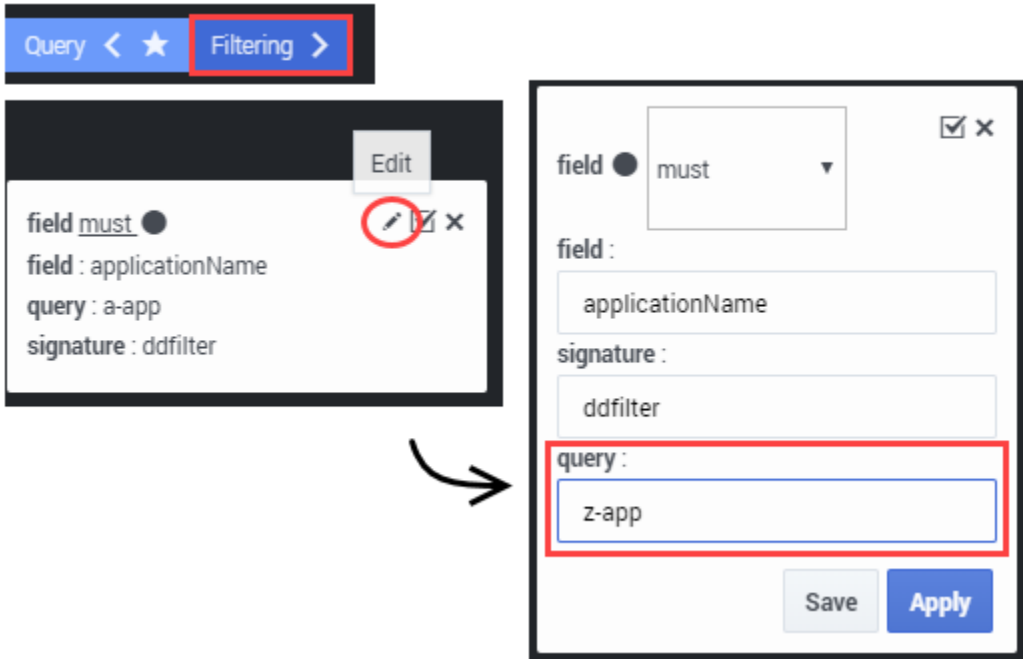
(See the Summary dashboard for a description of this report.)

Filter

The **Filter** panel appears on many of the dashboards (notable exceptions include the **Session Detail Records** and **Summary** dashboards). Use it to select the specific values you want to filter for, such as **Application** or **Disposition**.

> ### Important
> Any filters you select are applied across *all* of the dashboards, not just the one you are viewing. You can toggle the Filtering tab to see the filters that are currently being applied to the dashboard.

The drop-downs on the **Filter** panel only list the top 100 values for each item. If the value you are looking for isn't in the list, you can toggle the Filtering tab to expose the filter queries that are currently being applied to the dashboard. You can then edit the filter query to change the value to the one you are looking for:



By Disposition

This report shows the distributions of **final dispositions** across all calls received. A *final disposition* is the status assigned to a call at the time it exited the call flow, such as whether it was routed to an agent, terminated due to it being a special day or outside of regular business hours, or the caller hung up. (Learn more about dispositions here.)

By User Disposition Category

This report shows the distributions of final dispositions across all calls received, by the disposition category. The *disposition category* is the high-level status of the call when the caller exited the call flow, such as **Transfer** or **Abandoned**.

By User Disposition

This report shows the distributions of final dispositions across all calls received, by user disposition. The *user disposition* is the status assigned to a call when the caller exited the call flow. (Learn more about dispositions here.)

Top 10 DTMF Path

Designer automatically tracks various DTMF paths (how callers are responding to things like menu options), so if the top paths here are not the ones you expect, it might indicate that callers are not following the intended flow.

Top 10 Last Milestones

This report tracks the last milestones that callers reached before the call ended. A *milestone* is a custom benchmark (or checkpoint) that you've defined in an application to indicate that a significant point in the application flow was reached. For example, you might set up a milestone to mark when callers have made a successful payment, and another for when they've agreed to certain terms and conditions.

There are also other milestone-related reports:

- **Top 10 Bailout Milestones** — when the caller requests an agent (for example, by pressing **0**)
- **Top 10 Self-Helped Milestones** — when the caller is able to complete their call in self-service
- **Top 10 Deflection Milestones** — when the caller is not able to complete their call in self-service and is sent elsewhere (for example, to an agent)

Top 10 Activities

An *activity* is a task that you've defined in an application as a series of steps with a starting point and stopping point. For example, you might set up an activity for making a payment that starts with the caller being asked for their credit card details and then ends with the system sending those details to a payment processor and receiving the approval.

Each activity has a *start* and *end* point, and can be complete or incomplete, with success or failure.

Top 10 Milestones

This report displays the top ten milestones that were successfully reached by callers.

Top 10 Deflection Messages

These messages are generated when a caller can't complete their call in self-service and is redirected elsewhere.

Performance by Activities

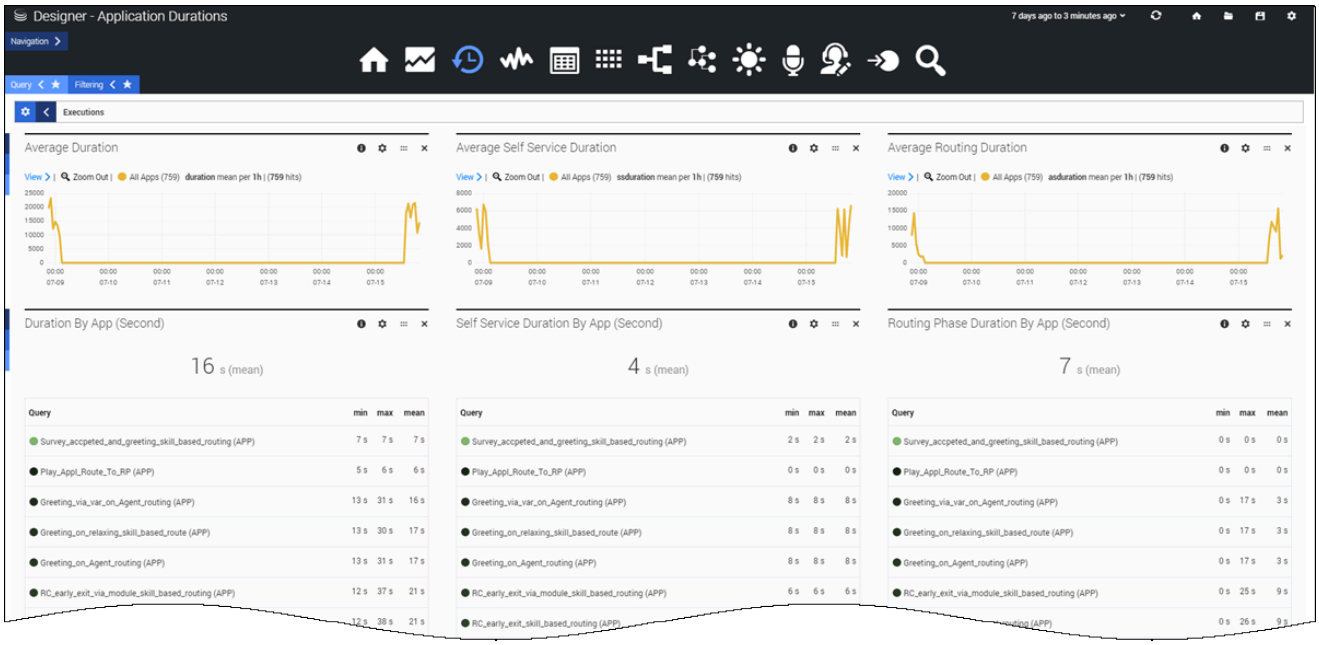
This report shows a breakdown of performance data for each activity over a given time period, grouped by Type (**user** or **system**). Note that activities with the same name are grouped together, even if they belong to different applications.

Clicking an activity will show (or hide) the call results for that activity. You can also export the results to a CSV file.

Durations

> **Important**
>
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

The **Durations** dashboard shows you how much time (in seconds) your callers are spending in the different phases of the application flow.



This dashboard can quickly show you:

- if callers are spending more time than expected in the **Self Service** or **Assisted Service** phases of an application
- if the amount of time callers are spending in certain sections of an application varies at different times
- any other unusual peaks in time being spent

The reports on this dashboard can help you determine if there are any possible issues with the application or its flow design, such as callers finding it difficult to navigate.

Reports on this dashboard

Count Over Time

(See the Summary dashboard for a description of this report.)

Filter

(See the Summary dashboard for a description of this panel.)

Average Duration

The average duration of time (in seconds) for all interactions across all applications. This data is captured for every 5-minute interval and then plotted as a line graph for the last 7 days.

Average Self Service Duration

The average time (in seconds) that callers are spending in the **Self Service** phase of the application, across all applications.

Average Routing Duration

The average time (in seconds) that callers are spending in the **Assisted Service** phase of the application, across all applications.

Routing Phase Duration by App

The amount of time (in seconds) callers are spending in the **Assisted Service** phases of the application, broken down by application.

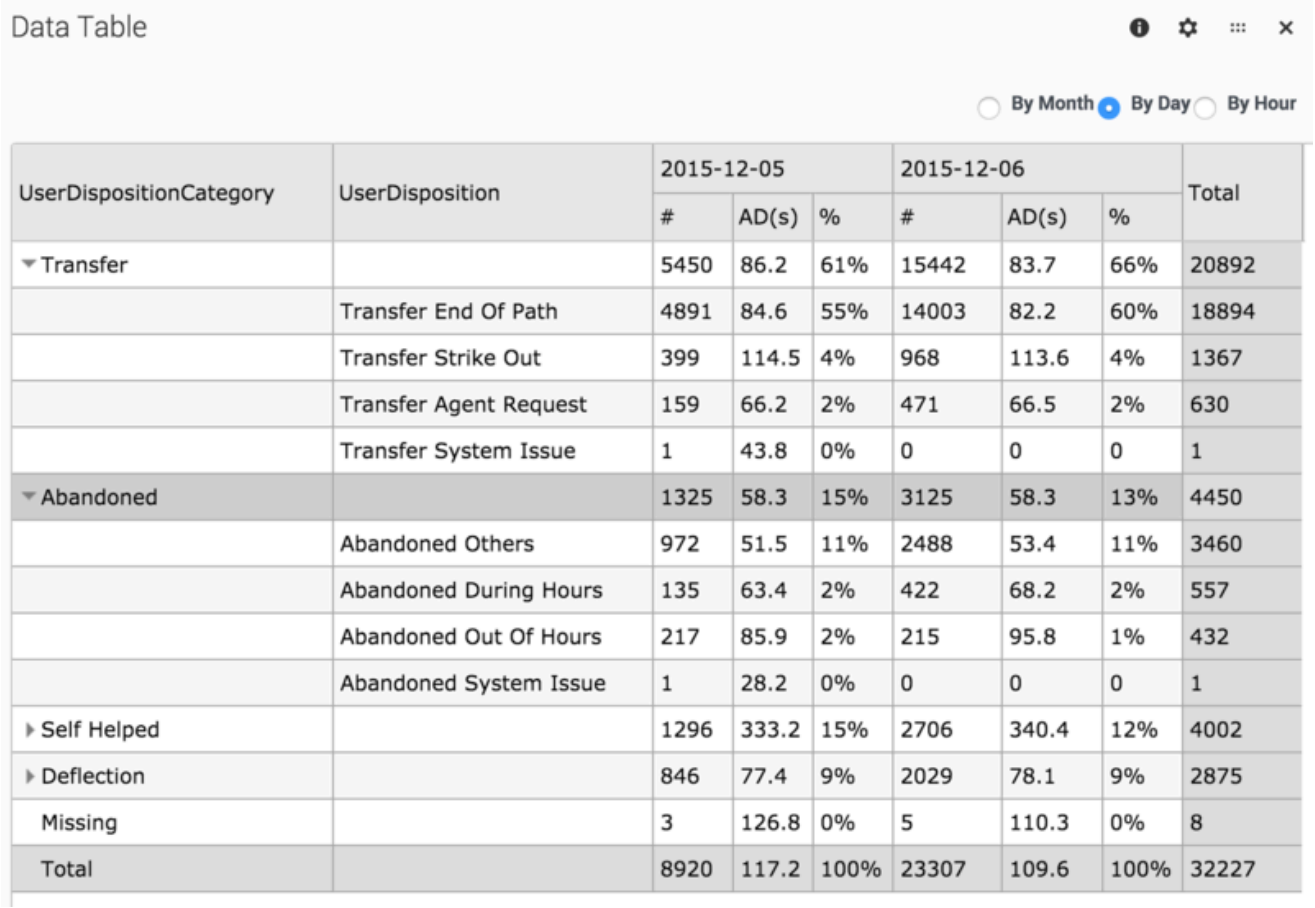
Self Service Duration by App

The amount of time (in seconds) callers are spending in the **Self Service** phases of the application, broken down by application.

Data Tables

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

The **Data Tables** dashboard arranges the disposition data of your applications in a tabular format.



UserDispositionCategory	UserDisposition	2015-12-05			2015-12-06			Total
		#	AD(s)	%	#	AD(s)	%	
Transfer		5450	86.2	61%	15442	83.7	66%	20892
	Transfer End Of Path	4891	84.6	55%	14003	82.2	60%	18894
	Transfer Strike Out	399	114.5	4%	968	113.6	4%	1367
	Transfer Agent Request	159	66.2	2%	471	66.5	2%	630
	Transfer System Issue	1	43.8	0%	0	0	0	1
Abandoned		1325	58.3	15%	3125	58.3	13%	4450
	Abandoned Others	972	51.5	11%	2488	53.4	11%	3460
	Abandoned During Hours	135	63.4	2%	422	68.2	2%	557
	Abandoned Out Of Hours	217	85.9	2%	215	95.8	1%	432
	Abandoned System Issue	1	28.2	0%	0	0	0	1
Self Helped		1296	333.2	15%	2706	340.4	12%	4002
Deflection		846	77.4	9%	2029	78.1	9%	2875
Missing		3	126.8	0%	5	110.3	0%	8
Total		8920	117.2	100%	23307	109.6	100%	32227

Reports on this dashboard

Count Over Time

(See the Summary dashboard page for a description of this report.)

Filter

(See the Application Details dashboard page for a description of this panel.)

Data Table

This report organizes the application disposition information into a table view. It groups the dispositions by category so you can see:

- the number of sessions that took place (**#**)
- the average duration (in seconds) for each session (**AD**)
- the percentage count of sessions (**%**)

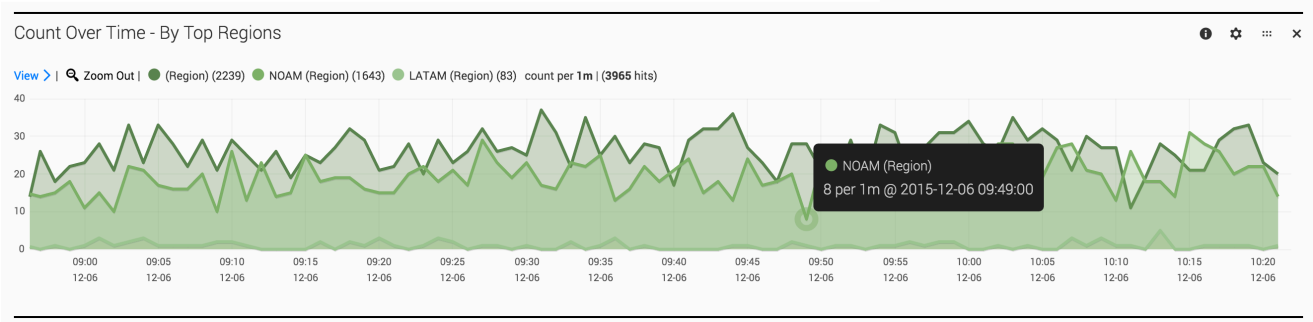
You can then use the options to toggle the results **By Month**, **By Day**, or **By Hour** to get a more detailed look at the final results of your calls.

> ### Important
> **By Hour** is only available if the given time window is within two days.

Spikes

> **Important**
>
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

This dashboard provides a "spikes" view of application sessions over a given period of time. It lets you easily visualize your call volumes by breaking down the total number of sessions by **Top Regions**, **Top Countries**, and **Top Languages**.



Reports on this dashboard

Count Over Time

(See Count Over Time.)

Filter

(See Filter.)

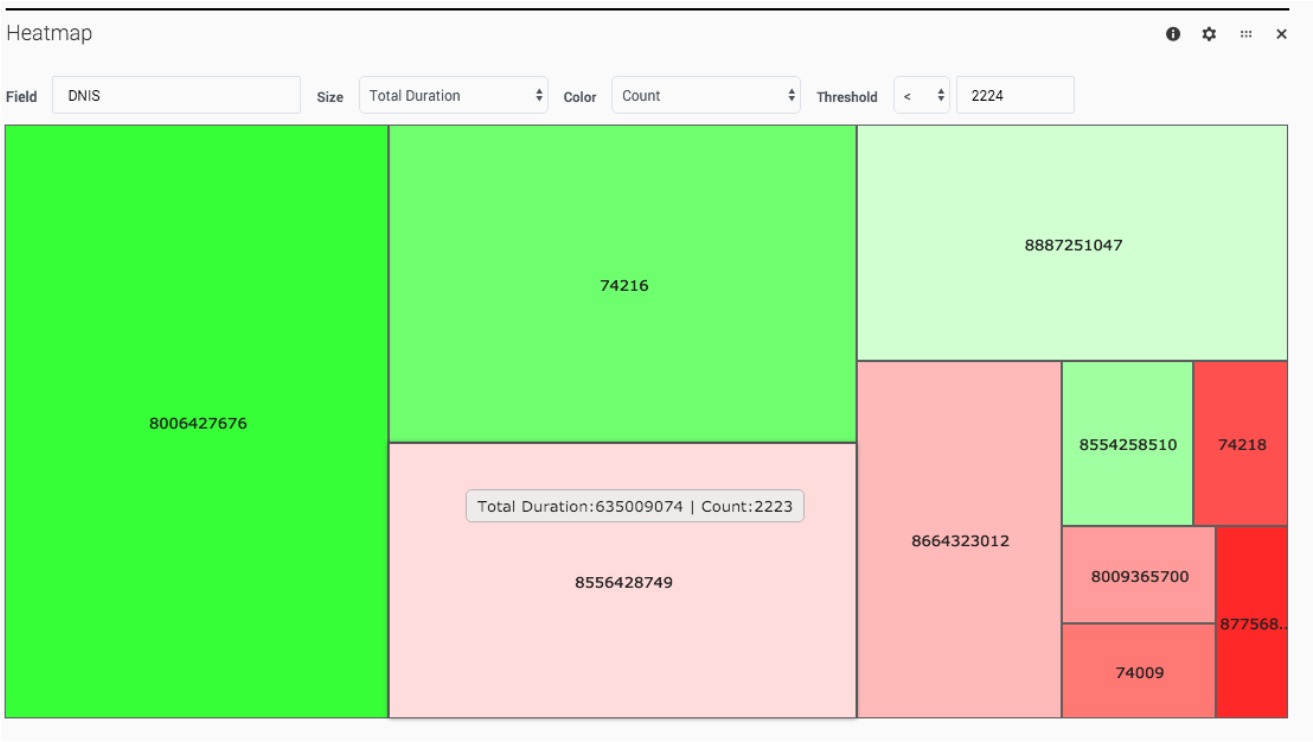
Count Over Time - By Top Regions/Countries/Languages

These panels take the total number of sessions received during the given time period and break them down into individual reports for each item.

Heatmap

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

The Heatmap dashboard uses colors to show a graphical representation of individual values. The graph changes colors when certain counts (or thresholds) for the selected field are reached.



Reports on this dashboard

Count Over Time

(See Count Over Time.)

Filter

(See Filter.)

Heatmap

Basically, the heatmap is a collection of colored rectangles that represent the session counts for a particular field.

If you look at the example above, the Heatmap panel shows the session counts for the **DNIS** field, with the size and colors of the rectangles changing as the **Total Durations** values increase or decrease. The longer the **Total Durations**, the larger the rectangle, and the color progression goes from green to red based on the **Count**, with the color intensity set by the **Threshold**.

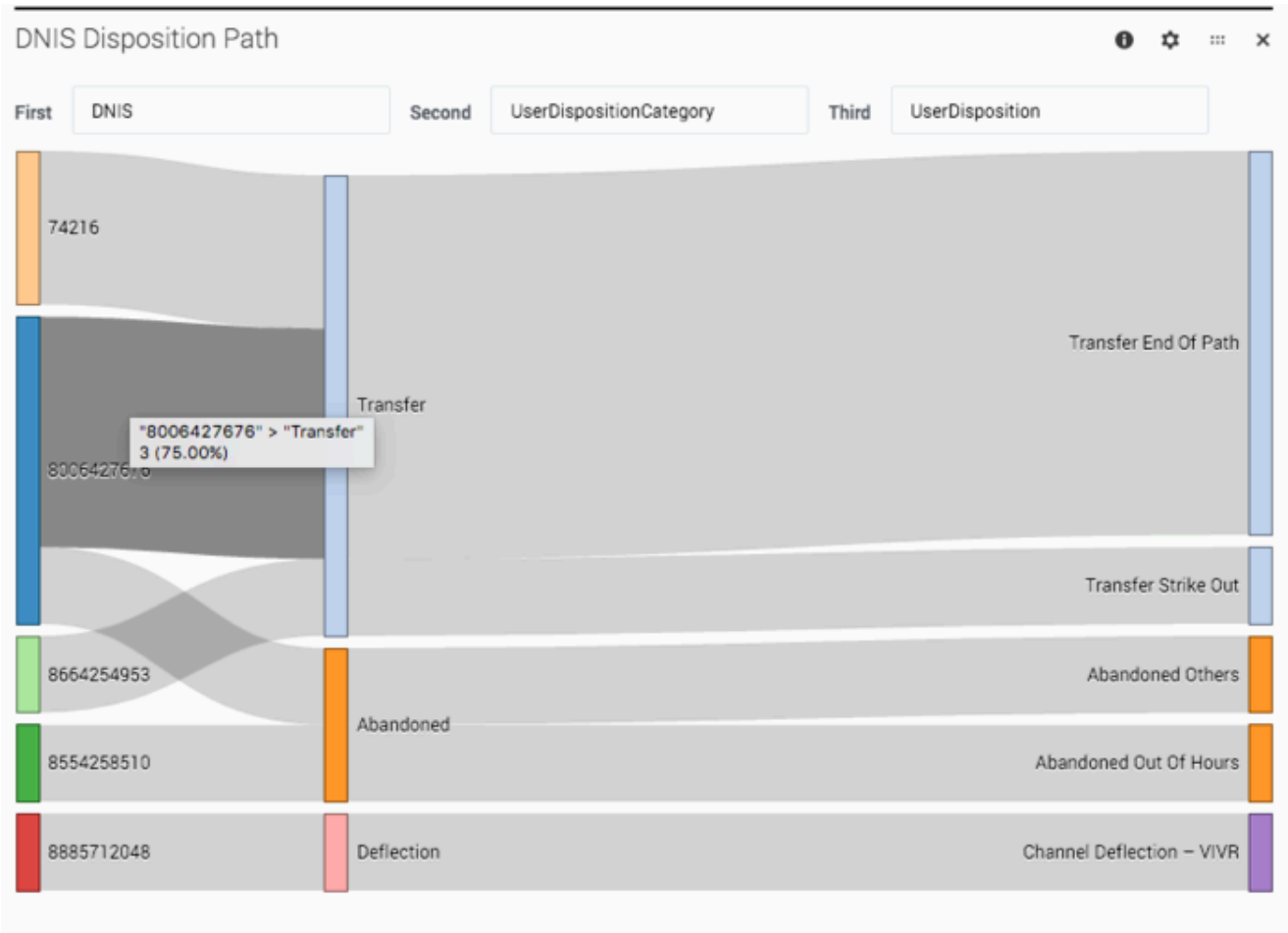
So, in this example, the largest rectangle (DNIS=8006427676) had the longest **Total Duration**, while the smallest (DNIS=87568) had the shortest. As the **Threshold** is set to **< 2224**, rectangles where the count is less than that are in red. The deeper the red, the further away the count is from the set threshold.

Path

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

This dashboard provides a visualization of how callers are moving through the applications by tracing their path through various nodes.

> **Tip**
> The Sankey panel might not be able to render correctly if there are several nodes or links to display. When this happens, you'll be prompted to increase the height of the panel.



Looking at this sample report, there are three nodes selected:

- First = *DNIS*, which is the number that callers dialed
- Second = *UserDispositionCategory*, which is the top-level disposition category (for example, **Transfer**)
- Third = *UserDisposition*, which is how the call finally ended up within that disposition category (for example, **Transfer End of Path**)

For each application session that took place during the selected time, a line is drawn between each node to represent the path that callers took. As more sessions share a common path, the path gets thicker.

You can easily see how callers are navigating through the application flows and quickly adjust the selected nodes to build paths for other values and categories.

Reports on this dashboard

> **Tip**
> You can select **Final Disposition** or **End Call Path** as an option to track callers in the application.

Count Over Time

(See Count Over Time.)

Filter

(See Filter.)

DNIS Disposition Path

This path shows how the sessions are tracking by dialed number through the high-level disposition categories (such as **Transfer** or **Abandoned**) to a final disposition (such as **Transfer End of Path** or **Abandoned Out of Hours**).

DNIS Call Type Exit Point

This path shows how sessions are tracking from each dialed number by call-type, through to the exit point of the call (the block where the caller hung up).

CallType Disposition Path

This path shows how the sessions are tracking for each call type, through the high-level disposition categories to a final disposition.

Call Type Disposition Exit Point

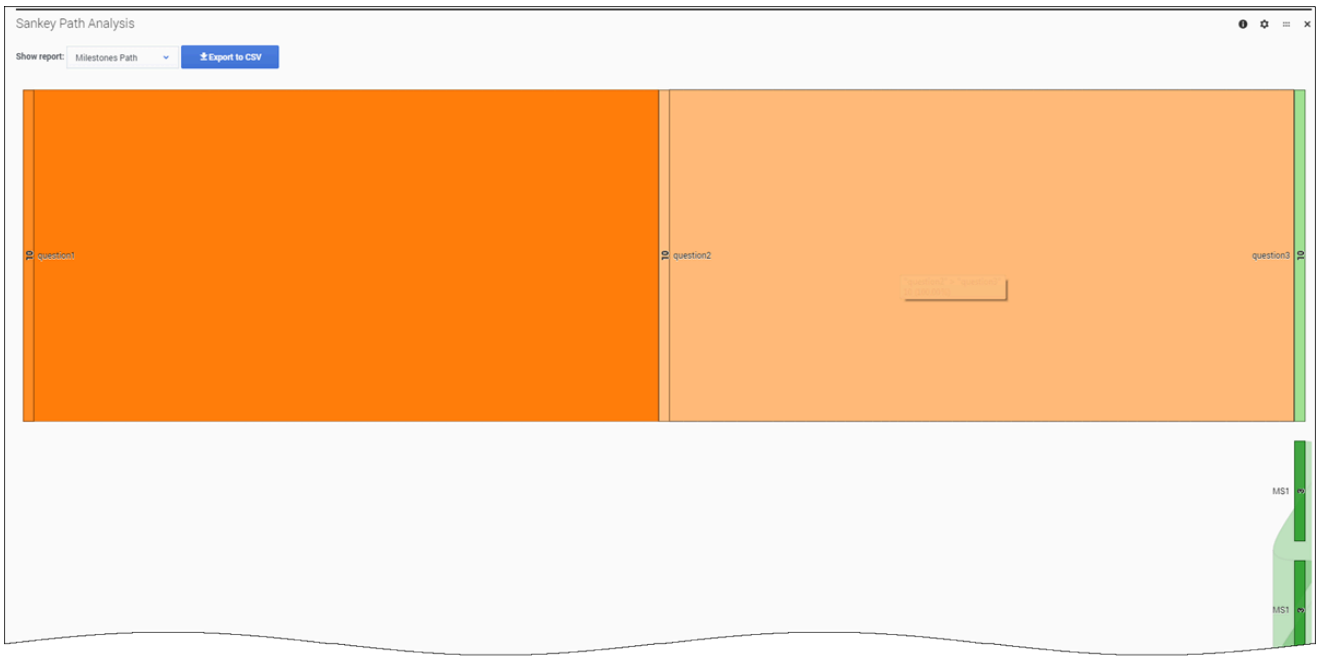
This path shows how the sessions are tracking by call type, through the high-level disposition categories to the exit point of the call (the block where the caller hung up).

Sankey Path Analysis

> **Important**
>
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

This dashboard is similar to the Path dashboard, except that it generates the results based on milestones and menu inputs.

> **Tip**
>
> The Sankey panel might not be able to render correctly if there are several nodes or links to display. When this happens, you'll be prompted to increase the height of the panel.



Reports on this dashboard

Count Over Time

(See Count Over Time.)

Filter

(See Filter.)

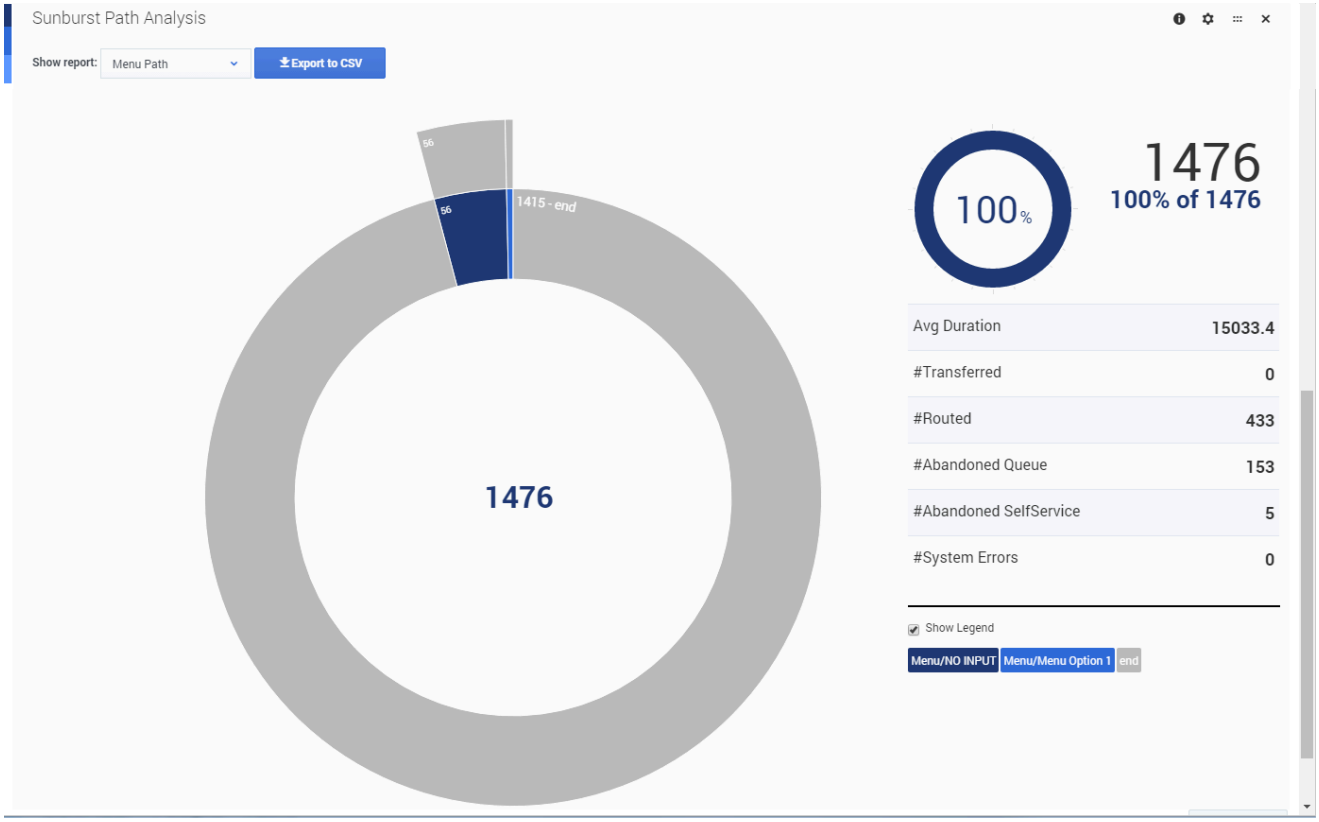
Sankey Path Analysis

This diagram shows how sessions are tracking through a series of milestones or menu inputs. You can toggle between the different reports using the **Show report** menu, and export the results to a CSV file.

Sunburst Path Analysis

> **Important**
>
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

The **Sunburst Path Analysis** dashboard shows a visual representation of the menu and milestone paths for your application sessions.



Reports on this dashboard

Count Over Time

(See Count Over Time.)

Filter

(See Filter.)

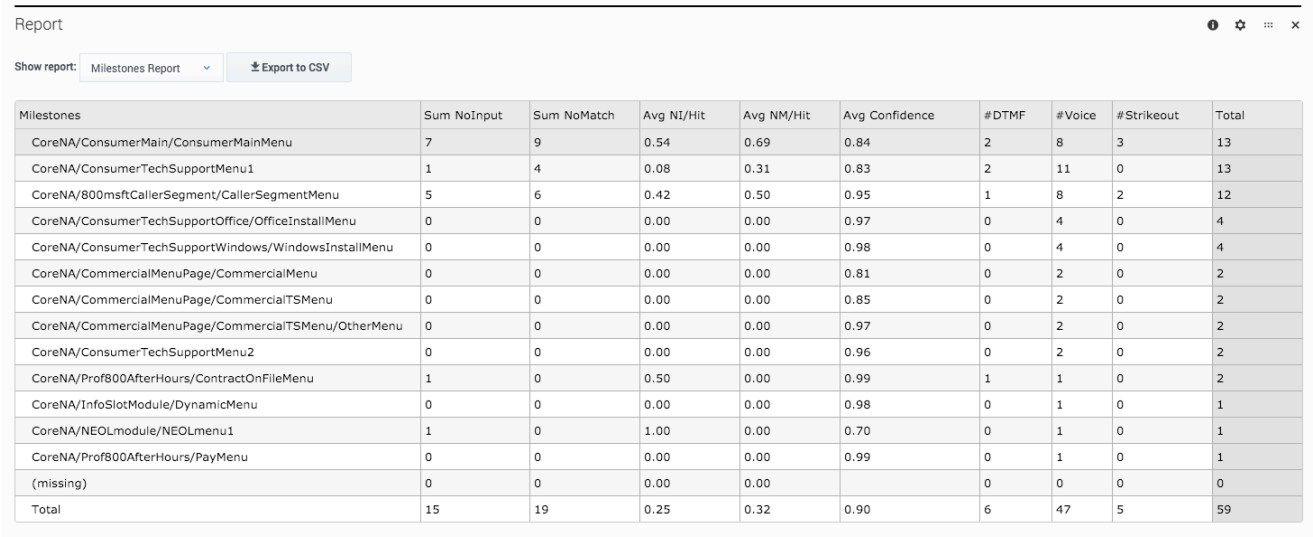
Sunburst Path Analysis

By default, this report displays the first seven nodes of a menu or milestones path, but you can change this in the configuration settings for the panel. The center of circle shows the total count, and you can double-click any of the partitions to drill down for more information.

Inputs

> **Important**
>
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

The **Inputs** dashboard shows you information about the menu inputs for application sessions that were active during the given time window. For example, you can see the number of sessions where there was **No Input**, **No Match**, or a **Strikeout**.


Report

Show report: Milestones Report Export to CSV

Milestones	Sum NoInput	Sum NoMatch	Avg NI/Hit	Avg NM/Hit	Avg Confidence	#DTMF	#Voice	#Strikeout	Total
CoreNA/ConsumerMain/ConsumerMainMenu	7	9	0.54	0.69	0.84	2	8	3	13
CoreNA/ConsumerTechSupportMenu1	1	4	0.08	0.31	0.83	2	11	0	13
CoreNA/800msftCallerSegment/CallerSegmentMenu	5	6	0.42	0.50	0.95	1	8	2	12
CoreNA/ConsumerTechSupportOffice/OfficeInstallMenu	0	0	0.00	0.00	0.97	0	4	0	4
CoreNA/ConsumerTechSupportWindows/WindowsInstallMenu	0	0	0.00	0.00	0.98	0	4	0	4
CoreNA/CommercialMenuPage/CommercialMenu	0	0	0.00	0.00	0.81	0	2	0	2
CoreNA/CommercialMenuPage/CommercialTSMenu	0	0	0.00	0.00	0.85	0	2	0	2
CoreNA/CommercialMenuPage/CommercialTSMenu/OtherMenu	0	0	0.00	0.00	0.97	0	2	0	2
CoreNA/ConsumerTechSupportMenu2	0	0	0.00	0.00	0.96	0	2	0	2
CoreNA/Prof800AfterHours/ContractOnFileMenu	1	0	0.50	0.00	0.99	1	1	0	2
CoreNA/InfoSlotModule/DynamicMenu	0	0	0.00	0.00	0.98	0	1	0	1
CoreNA/NEOLmodule/NEOLmenu1	1	0	1.00	0.00	0.70	0	1	0	1
CoreNA/Prof800AfterHours/PayMenu	0	0	0.00	0.00	0.99	0	1	0	1
(missing)	0	0	0.00	0.00		0	0	0	0
Total	15	19	0.25	0.32	0.90	6	47	5	59

Reports on this dashboard

No Input

The total number of sessions with **No Input**.

No Match

The total number of sessions with **No Match**.

Strikeout

The total number of sessions where the maximum number of inputs for **No Input** or **No Match** was reached.

Report

This report lists the milestones and their various input counts. You can use **Show Report** to toggle the following report views:

- Milestones Report
- Milestones and Utterances Report
- Milestones and Interpretations

You can also export the results to a CSV file.

Report by Block

This report lists the blocks and their various input counts. You can use **Show Report** to toggle the following report views:

- Blocks and Utterances Report
- Blocks and Interpretations Report

You can also export the results to a CSV file.

> ### Tip
>
> An **utterance** is what Designer believes the caller has *said*. An **interpretation** is what Designer thinks the caller actually *meant* (in a context that is meaningful to the application).
>
> For example, a caller might say "I need to speak with *someone.*" This becomes an utterance value that Designer could map to an interpretation value of "I need to speak with an *agent*", which enables the application to respond appropriately.

Blocks By NoMatch/NoInput Count

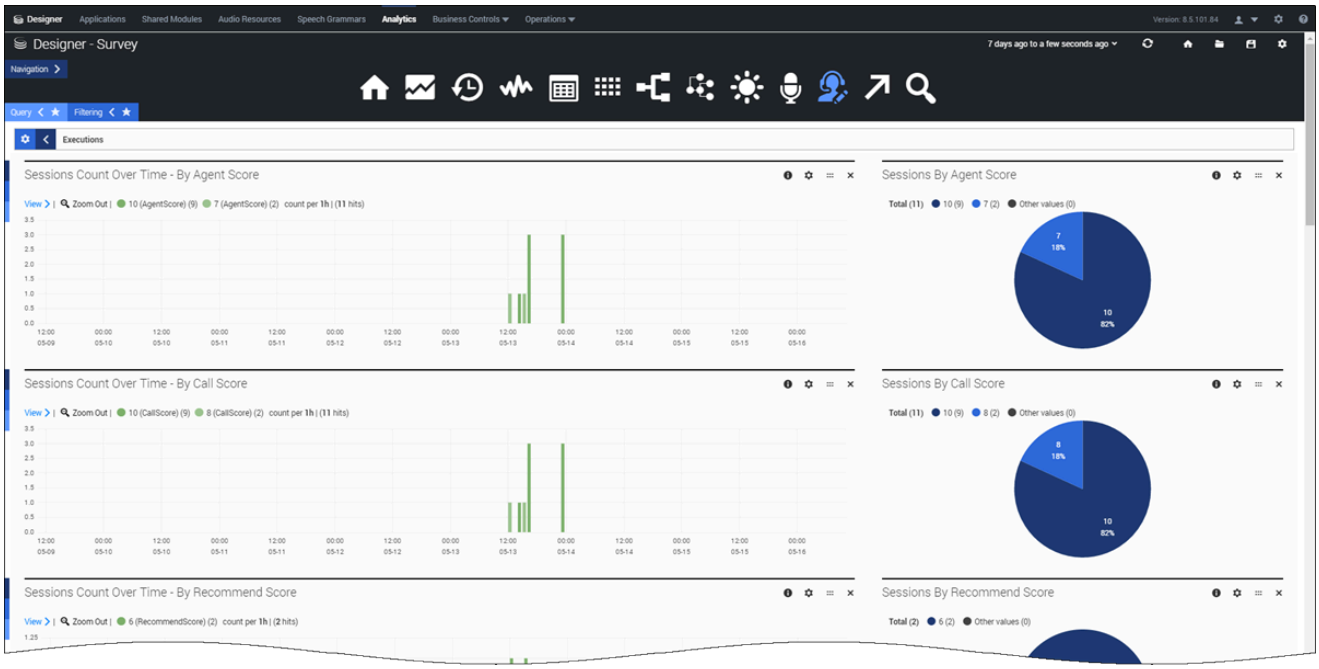
This report shows the number of times an input field encounters a No Match or No Input per call.

Use **Show Report** to toggle the **Blocks By NM/NI Count** report. You can also export the results to a CSV file.

Surveys

> **Important**
>
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

The **Surveys** dashboard gives you information related to your survey applications.



The **Surveys** dashboard contains many of the same panel types that are shown on other dashboards, but also includes data that is specific to the surveys in your applications (mostly pulled from the variables you initialized when the survey was set up).

Reports on this dashboard

Survey Count Over Time

Similar to Count Over Time, but specific to the number of surveys that were offered, accepted, or

rejected.

Filter

(See Filter.)

Sessions Count Over Time

These histogram reports break the number of session counts down by:

- **Product Score**
- **Company Score**
- **Recommend Score**

These are based on the ratings callers gave during the survey.

Sessions By Score

These pie graph reports are in pie graph format and break the number of session counts down by:

- **Product Score**
- **Company Score**
- **Recommend Score**

These are based on the ratings callers gave during the survey.

Survey Answer Distribution

This report lists the questions that were presented to callers, along with the average scores for each question, the total number of sessions in which the question was presented, and the total number of sessions where the question was answered.

Sunburst Path Analysis

This report gives you a quick snapshot of how many surveys were offered, accepted, or rejected.

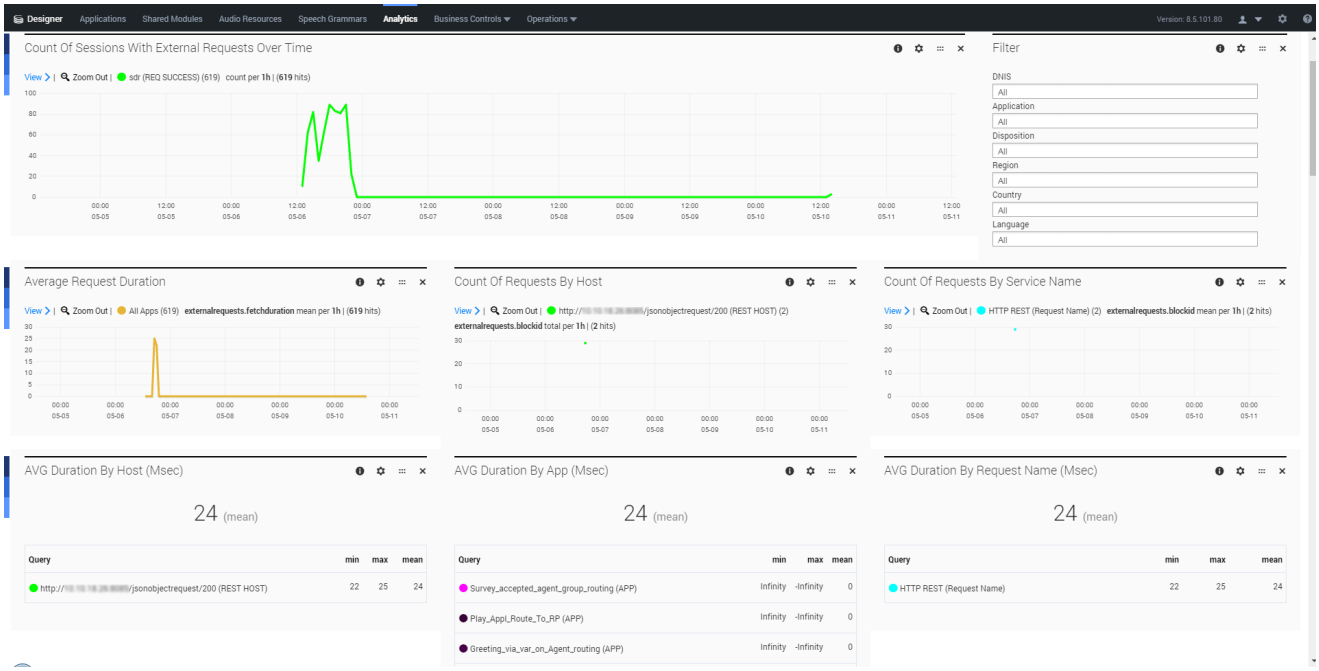
External Services

> **Important**
>
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

Designer applications can request a connection to a web-accessible external system, such as Enterprise Resource Planning (ERP) or Customer Relationship Management (CRM), to fetch or update data about a call. These integration points between applications and external services can play an important role, such as determining whether a call stays within **Self Service** or is routed to an agent through **Assisted Service**.

Currently, there are two blocks that can make these types of requests: Custom Service and HTTP REST.

The reports on this dashboard will help you to analyze the external requests being made by your applications.



Reports on this dashboard

Count of Sessions With External Requests Over Time

This report captures the number of external service requests that were made by the applications during the given time period. You can quickly see how the number of requests compares to the traffic patterns, and spot any trends or unexpected deviations.

For example, if you see that web service requests are increasing during certain times, you can check to make sure that the external services can handle that volume of requests.

Filter

(See Filter.)

Average Request Duration

You can use this report to see how long (on average) it is taking for the external services to respond to requests from the applications.

For example, if the response time trajectory is flat, it typically means that data is being retrieved or updated as expected. But if the response times are increasing significantly as your call volumes rise (it's normal to see a slight increase in response times whenever there is a spike in call volumes), it might indicate that the external system is becoming strained while trying to handle so many requests at the same time.

Count Of Requests By Host and Service Name

These reports provide the number of external service requests (or hits) for each Host of a particular Service Name.

- *Host* is the domain name of the URL that is receiving a request from the HTTP REST or Custom Service block.
- *Service Name* is the name of the block used within the application.

You can use this information to identify which hosts or services are receiving the most hits, and then plan the external requests accordingly. If you are using third party integrations (such as payment gateways or location services), this data can provide insight into the consumption of those types of services.

AVG Durations By Host, App, and Request Name

These reports tell you how long (on average) it is taking the external web services to respond to requests. This average is calculated by adding the response times of all service requests and dividing them by the number of requests.

Analyzing the average durations based on host, URL, or application can help you identify if services are responding within the expected timeframe.

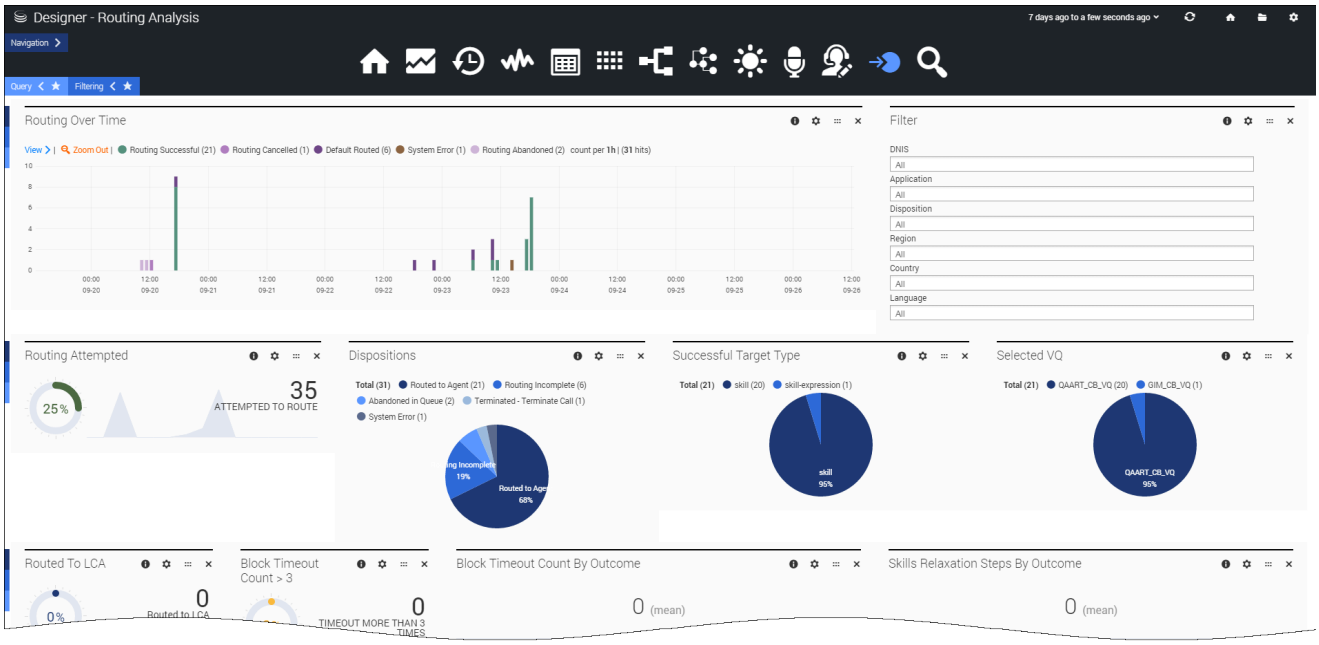
External Requests Status

This report shows the status of all external requests that were made during the given time period.

Routing Analysis

> **Important**
>
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

This dashboard gives you a deeper look into those sessions that have entered the Assisted Service phase of the applications.



These reports can help to answer questions like:

- How many sessions were successfully routed to their destination?
- How long are callers waiting in the queue before hanging up?
- Which applications had the lowest abandoned rates?
- Which blocks had the least number of timeouts?

Reports on this dashboard

Routing Over Time

The total number of routing sessions that happened during the given time period, broken down by status (such as, if the routing session was abandoned, successful, cancelled, default routed, or had an error).

Filter

(See the Summary dashboard for a description of this panel.)

Routing Attempted

The total count and percentage of sessions that entered the routing phase (based on the number of all sessions processed during the given time period).

Routed to LCA

The number of sessions that were successfully routed to the last called agent (that is, the agent that the customer last spoke with).

Timeout Block Count > 3

The total count and percentage of all routing sessions where the routing blocks had three (3) or more timeouts.

Predictive Routing Attempted Count > 3

The total count and percentage of sessions where three (3) or more predictive routing attempts were made.

Dispositions

The total number of routing attempts made, broken down by final disposition.

Successful Target Type

The number of sessions that were successfully routed to the target, broken down by target type (such as a specific skill).

Selected VQ

The number of sessions that were successfully routed to the selected virtual queue.

Predictive Dispositions

The total number of predictive routing attempts made, broken down by final disposition.

Successful Predictive Target Type

The number of sessions that were successfully routed to the target by predictive routing, broken down by target type (such as a specific skill).

Predictive Selected VQ

The number of sessions that were successfully routed to the selected virtual queue by predictive routing.

Block Timeout Count by Outcome

The total count, including the minimum, maximum, and mean, of routing block timeouts experienced by routing sessions, broken down by routing outcome (positive/negative).

Skills Relaxation Steps by Outcome

The total count, including the minimum, maximum, and mean, of skill relaxation steps, broken down by routing outcome (positive/negative).

Predictive Routing Attempted By Outcome

The total count, including the minimum, maximum, and mean, of predictive routing attempts, broken down by routing outcome (positive/negative).

Average Time in Queue (in seconds)

The average amount of time that sessions waited in the queue before being routed.

Abandoned in Queue Stats

Details about routing sessions that were abandoned while still in the queue, such as how long callers waited before hanging up.

Performance Report by Applications

This table shows the performance breakdown by individual applications, as based on a variety of metrics. This panel lets you see which applications have the best (or worst) performance.

Performance Report by Blocks within Applications

This table shows the performance breakdown by individual blocks, as based on a variety of metrics. This panel lets you see which blocks have the best (or worst) performance for any given application.

Skills Path

This panel provides a sankey visualization of the skills path. (By default, the depth of this panel is set to 7.)

Abandoned in Queue Stats

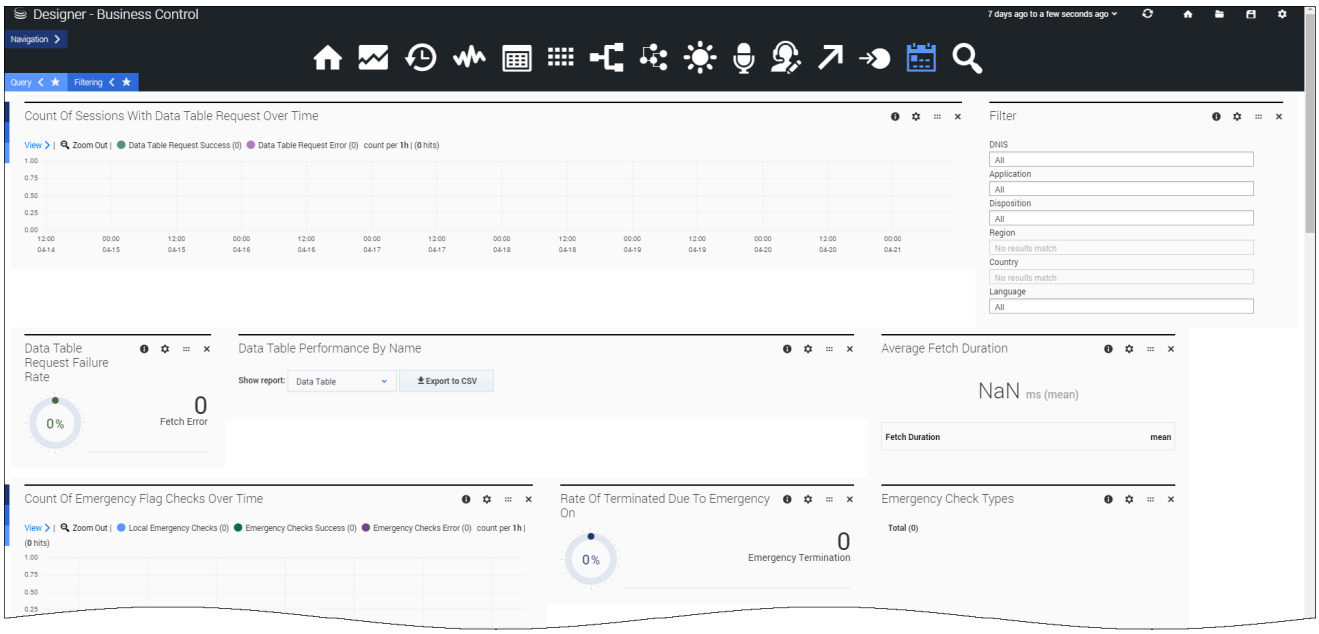
This panel provides a sunburst visualization of the Abandoned in Queue stats.

Business Control Dashboard

> **Important**
>
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

The Business Control dashboard provides near real-time metrics on the runtime usage of Emergency Flags, Business Hours, Special Days, and Data Tables.

> **Important**
>
> To populate the reports on this dashboard with data, you must first set the value of the **SdrTraceLevel** system variable in the application to `100` (debug level). For more information, see System Variables.



Reports on this dashboard

> **Tip**
> If a business control object that was checked in a session is deleted, the report adds the suffix `(deleted)` to the object ID.

Filter

(See the Application Details dashboard page for a description of this panel.)

Data Tables

Count of Sessions With Data Table Request Over Time

This report shows you the number of data table retrieval requests that were received over a period of time. You can quickly adjust the range of time shown, say, for the last hour, 24 hours, or the last week, to see the trends for those specific time periods.

Each bar indicates the total number of data table retrieval requests for each application that took place during the given time period. The higher the bar, the higher the number of requests. You can easily see what time of the day (or what day of the week) that requests are higher or lower, and organize your business accordingly.

Data Table Request Failure Rate

The failure rate (as a percentage) of data table requests across all sessions. It also shows the total number of Fetch Errors.

Data Table Performance by Name

A breakdown of outcomes for the most requested data tables. You can export the results to a CSV file.

Data Table Request Fetch Duration

The average duration (ms) of data table fetch requests, grouped by outcome (success or failure).

Emergency Flags

Count of Emergency Flag Checks Over Time

The total number of Emergency Flag checks that were sent during the selected period of time.

Rate of Terminated Due to Emergency On

The total rates of sessions (where an emergency flag was checked) that were terminated due to the

Emergency flag being on.

Emergency Check Types

A breakdown of Emergency checks by type.

Emergency Check Error Rate

The rate of errors (as a percentage) across all external emergency checks. It also shows the total of Check Errors.

Shared Emergency Flag Performance

A breakdown of outcomes for the most checked Emergency flags. You can export the results to a CSV file.

External Emergency Check Fetch Duration

The average duration (ms) of Emergency check fetch requests, grouped by outcome (success or failure).

Business Hours

Count of Business Hours Checks Over Time

The total number of Business Hours checks that were sent during the selected period of time.

Rate of Terminated Due to Business Closed

The total rates of sessions (where Business Hours were checked) that were terminated due to the business being closed.

Business Hours Check Types

A breakdown of Business Hours checks by type.

Business Hours Check Error Rate

The rate of errors (as a percentage) across all external Business Hours checks. It also shows the total of Check Errors.

Shared Business Hours Check Performance

A breakdown of outcomes for the most checked Business Hours. You can export the results to a CSV file.

Business Hours Check Fetch Duration

The average duration (ms) of Business Hours check fetch requests, grouped by outcome (success or

failure).

Special Days

Count of Special Days Checks Over Time

The total number of Special Days checks that were sent during the selected period of time.

Rate of Terminated Due to Special Days

The rate of sessions (where Special Days were checked) that were terminated due to it being a Special Day.

Special Days Hours Check Error Rate

The rate of errors (as a percentage) across all external Special Days checks. It also shows the total of Check Errors.

Special Days Check Type Breakdown

A breakdown of Special Days checks by type.

Shared Special Days Performance

A breakdown of outcomes for the most checked Special Days. You can export the results to a CSV file.

Special Days Check Fetch Duration

The average duration (ms) of Special Days check fetch requests, grouped by outcome (success or failure).

Top Hit Holidays

A list of the top holidays where a Special Days check returned *true*.

Errors

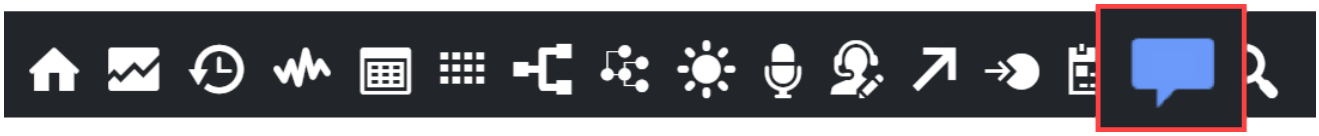
Count of Sessions With Business Control Errors Over Time

The total counts of sessions where errors in Data Table, (shared) Emergency Flag, (shared) Special Days and (shared) Business Hours checks were encountered, over the selected period of time.

Bots Dashboard

> **Important**
>
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

You can view the **Bots** dashboard by selecting the **Bots** icon:



The reports available on this dashboard include visualizations and detailed breakdowns of your bot sessions, such as how many sessions were successful or failed, the top intents that were detected, and how many bot sessions had intents detected that were not successfully fulfilled.

Understanding the results

This section describes some of the bot-related terminology used by Designer.

In some cases, bot service providers may use similar terms but with different applications of meaning. For example, a bot services provider might consider an intent to be *fulfilled* as soon as it is detected, while Designer doesn't consider an intent to be fulfilled until the additional processing related to the intent is successfully completed.

To better understand the results displayed on this dashboard, you should be familiar with how these terms are used within Designer.

Bot Session

Each **Bot Session** represents a single conversation between a customer and the bot service that was invoked by Designer via the Bot block.

A bot session starts as soon as the **Bot** block receives voice or chat input from the customer and ends when Designer either moves to an intent block or to an **Error Handler** block. If the same **Bot** block is executed again at a later point in the application, it is considered to be the start of a new bot session.

Bot Sessions can be classified as **Success** or **Failed**:

- **Success** indicates that Designer was able to invoke the bot. A conversation with the customer took place and the bot was able to successfully identify an intent and return it to the Bot block.
- **Failed** indicates that there was a condition that triggered the **Error Handler** block, such as Designer not being able to communicate with the bot.

Utterance

During a conversation, the bot will ask the customer to provide some information. Each voice or chat reply that the customer provides is an **utterance**. For example, it could be a request ("I want to book a hotel room") or a simple reply to a question ("Yes").

Intent

During its conversation with a customer, the bot attempts to identify the intent, or what it is that the customer wants to do. For example, the bot might detect that the customer wants to buy a ticket. It will then proceed to fill the required slots (or "entities") that are associated with that intent, such as the name of the show the customer wants to buy a ticket for, the date and time they want to attend, and so on, by asking the customer to provide those details.

Intent fulfillment

After the bot has successfully gathered all of the information it needs to satisfy an intent, Designer consider the intent to be fulfilled.

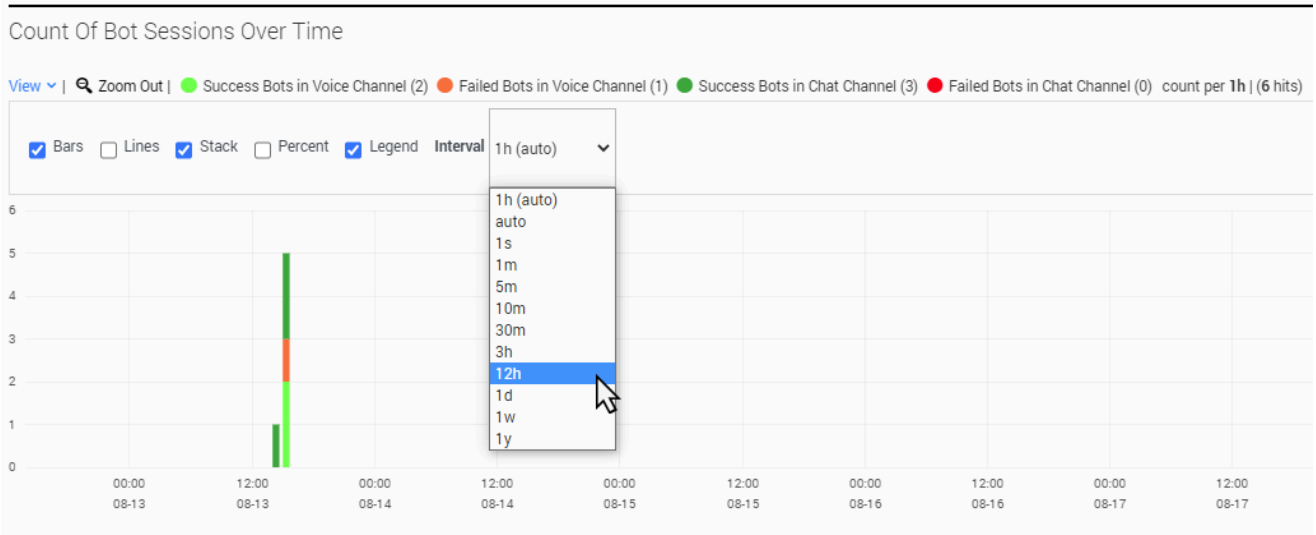
Reports on this dashboard

The following reports are available on this dashboard:

Count of Bot Sessions Over Time

This panel displays the total number of Bot Sessions that took place over a specified period of time for both Voice and Chat channels.

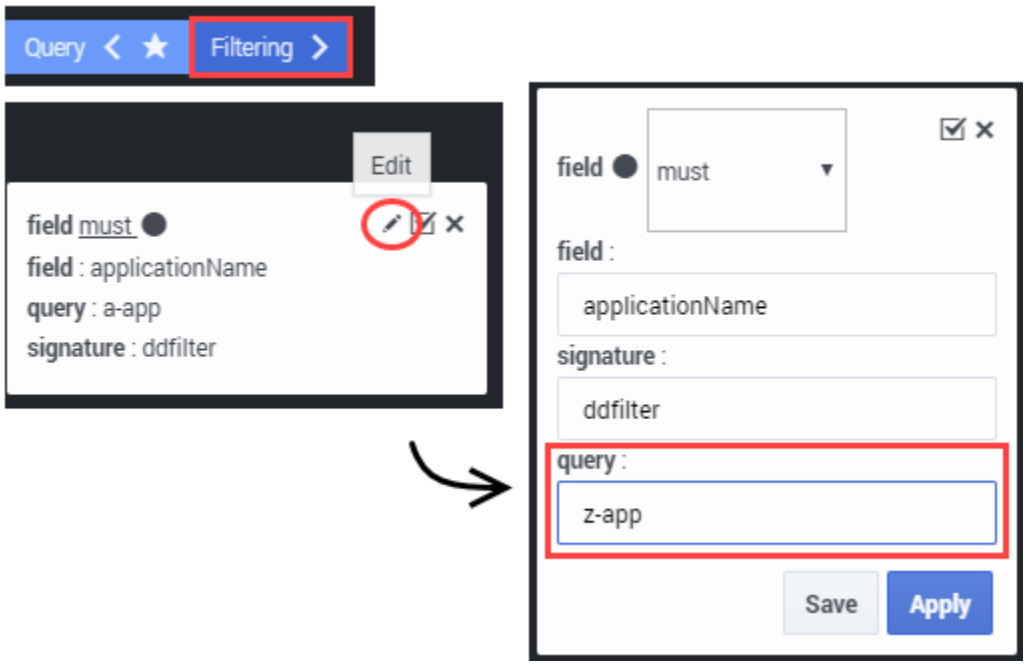
You can expand the **View** menu to select a different visualization option, or use the **Interval** setting to adjust the range of time shown, such as for the last hour, 12 hours, or the previous week.



Filter

You can use the **Filter** panel to select specific values you want to filter the results for, such as **Application** or **Disposition**.

The drop-downs on the **Filter** panel only list the top 100 values for each item. If the value you are looking for isn't in the list, you can toggle the Filtering tab to expose the filter queries that are currently being applied to the dashboard. You can then edit the filter query to change the value to the one you are looking for:

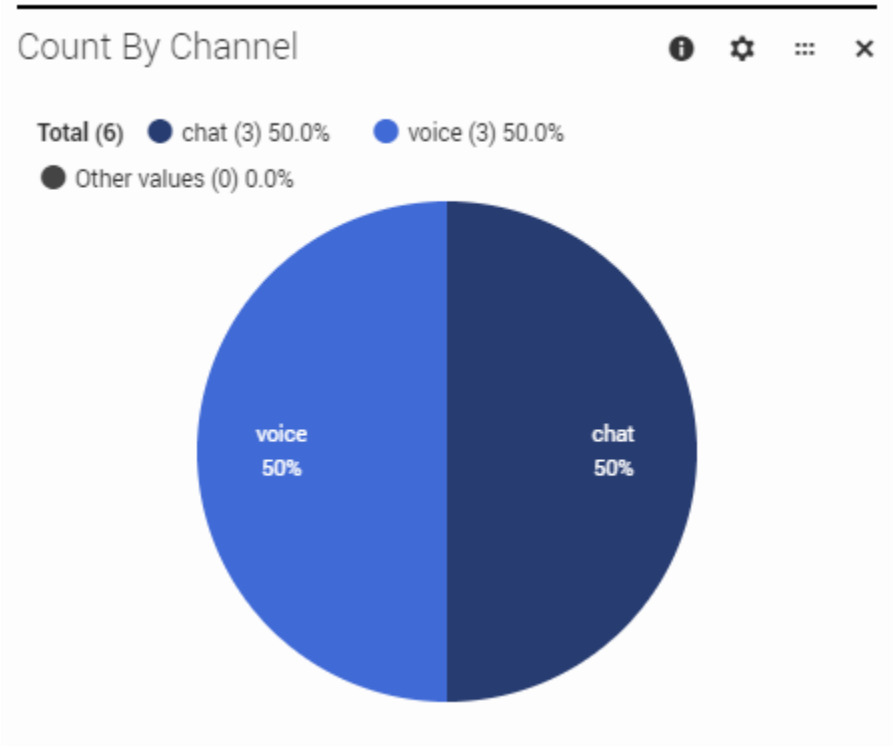


Any filters you select are applied across *all* of the dashboards, not just the one you are viewing. You

can toggle the Filtering tab to see the filters that are currently being applied to the dashboard.

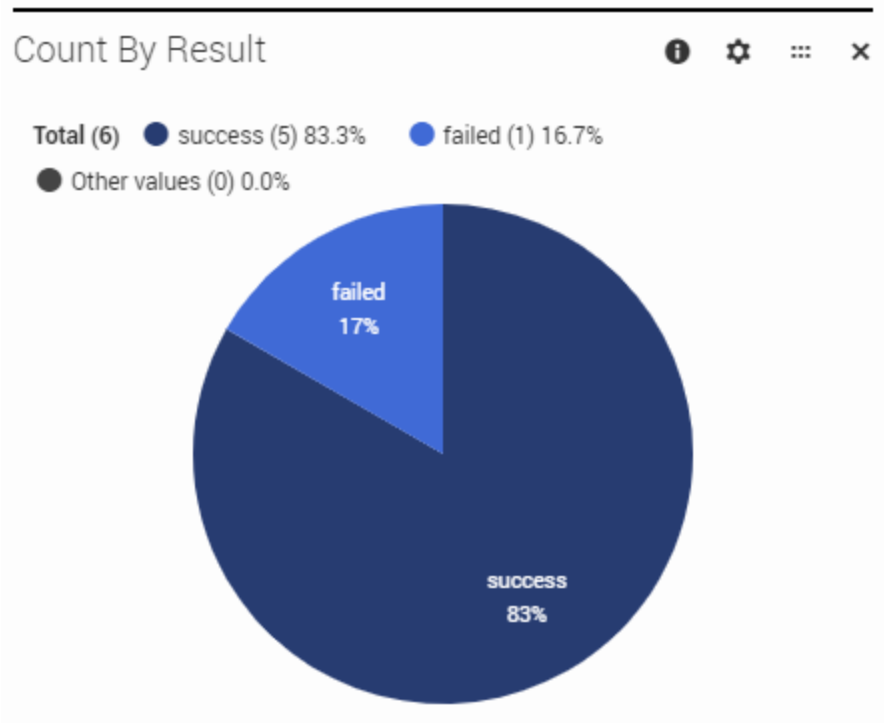
Count by Channel

This panel displays a breakdown of total counts and percentages for both Voice and Chat channel sessions.



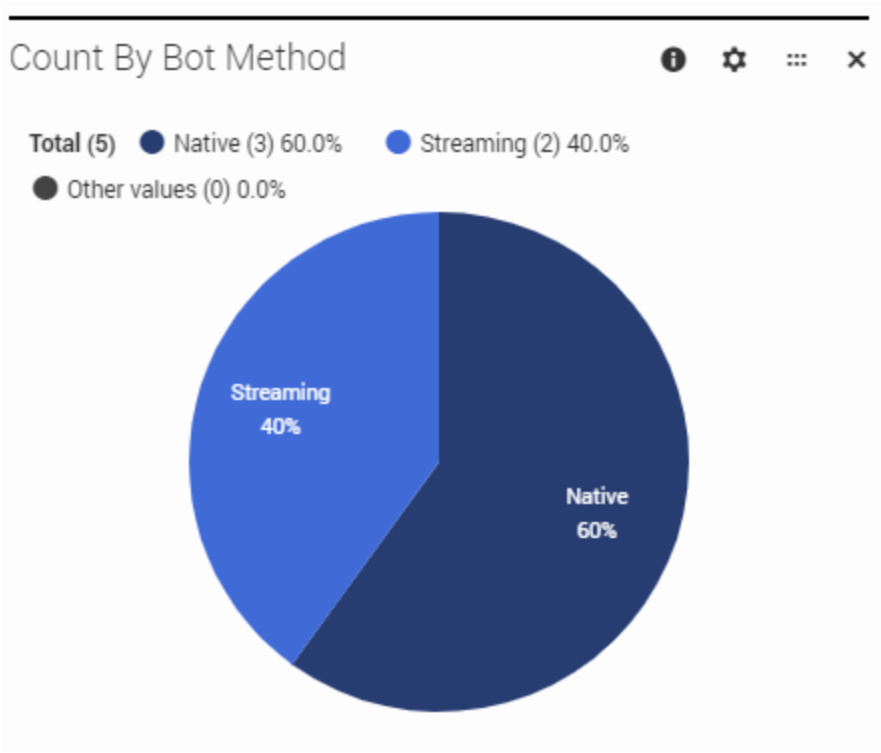
Count by Result

This panel provides total counts and percentages for Bot Session results, broken down by status. For details about **Success** and **Failed** sessions, see Understanding the results.



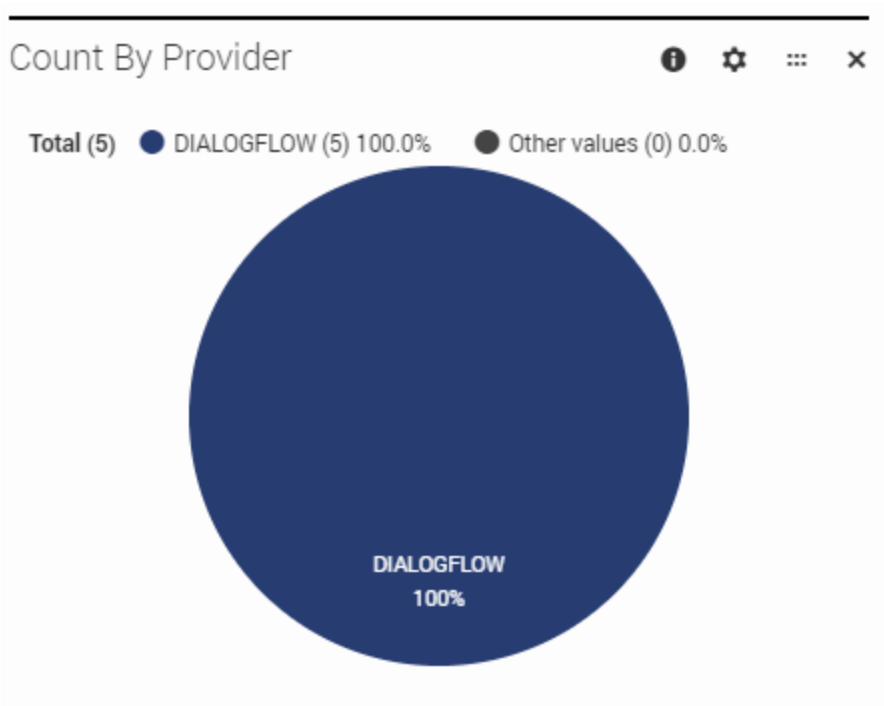
Count by Bot Method

This panel displays a breakdown of all Bot Sessions by the method used by the bot service (default is **Native**).



Count by Provider

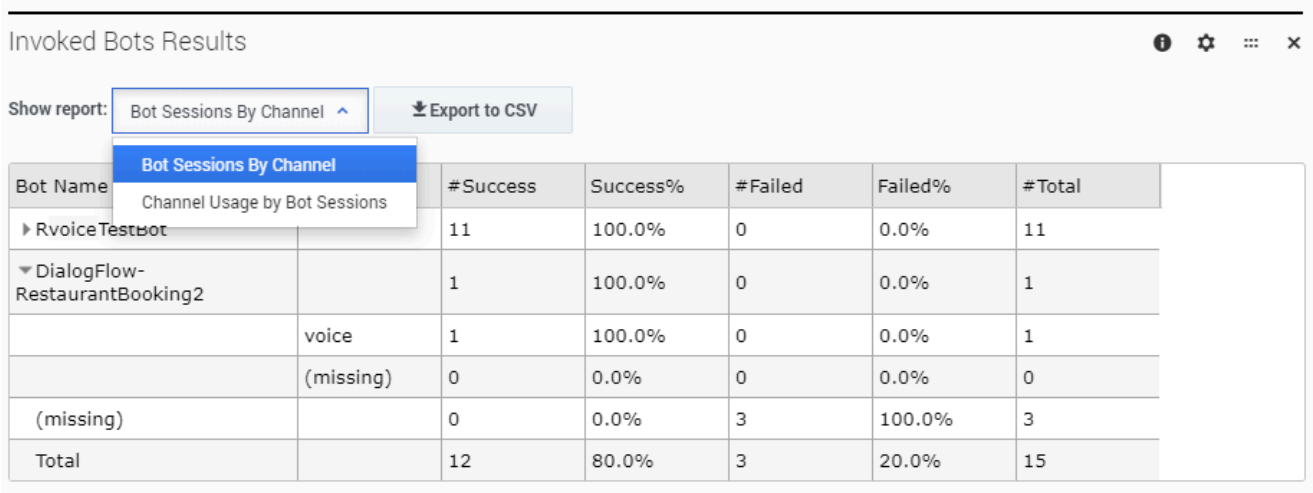
This panel displays a breakdown of all Bot Sessions by the bot services provider that was invoked (e.g. **Dialogflow**, **Lex**, **Dialog Engine**).



Invoked Bot Results

This panel lists the names of all bots that were invoked during the selected time period, along with the total counts and percentages for sessions where the attempt to invoke the bot was successful or failed. Clicking the arrow beside the bot name expands the row to show the breakdown of these totals by channel.

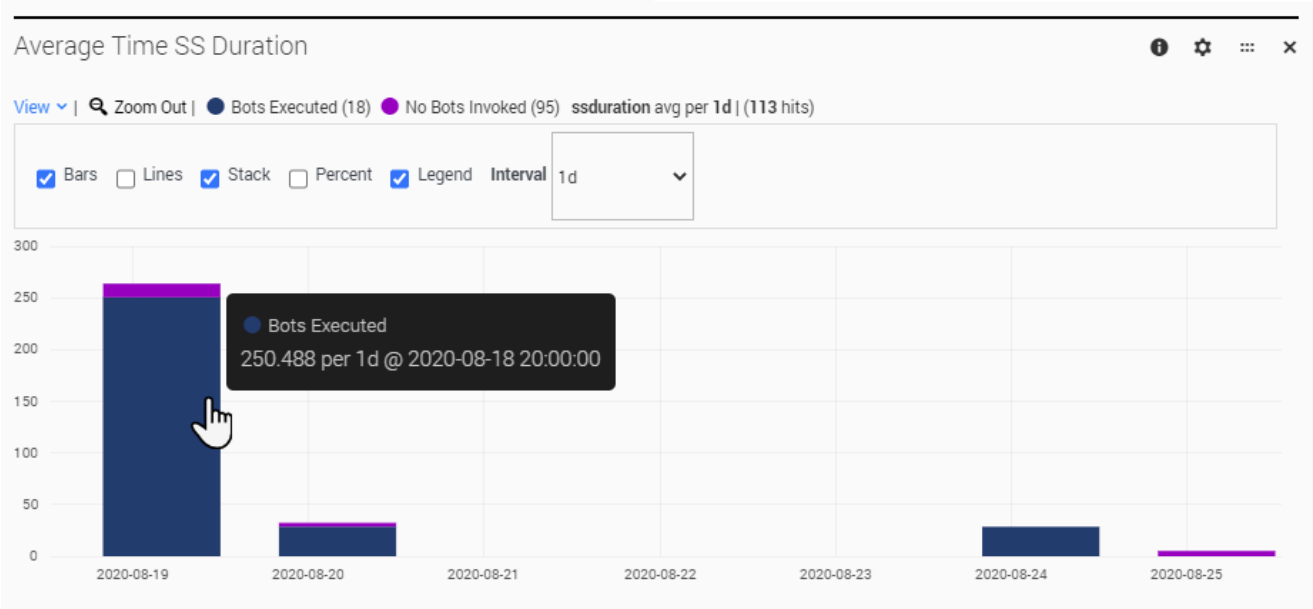
You can change the reporting view to show either the **Bot Sessions by Channel** or **Channel Usage by Bot Sessions**. If desired, you can export the results to a CSV file.



Average Time SS Duration

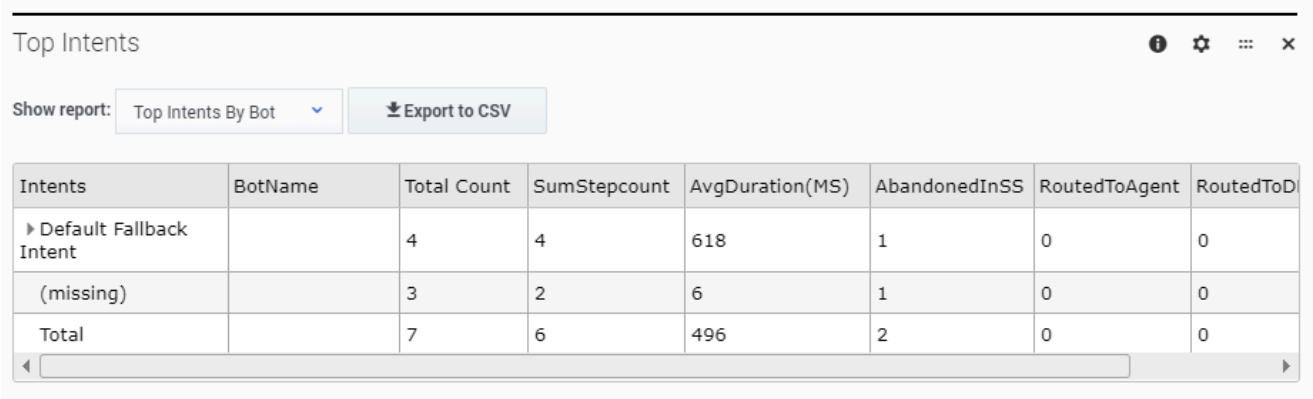
This panel displays the average duration of time (in seconds) that all sessions spent in the **Self Service** phase, broken down by total counts of sessions where bots were invoked (**Bots Executed**) and all other sessions (**No Bots Invoked**).

You can expand the **View** menu to change the visualization options or select a specific **Interval** to display (such as 1 day). Hovering over the results displays the details for that item.



Top Intents

This panel lists the top intents for all bots that were invoked during the selected time period. As described in Understanding the results, an **intent** is detected by the bot during its conversation with the customer and returned to Designer so that it can then be fulfilled (e.g. book a car, reserve a room).



Clicking an intent expands the row to display the names of the bots that detected that intent. Other

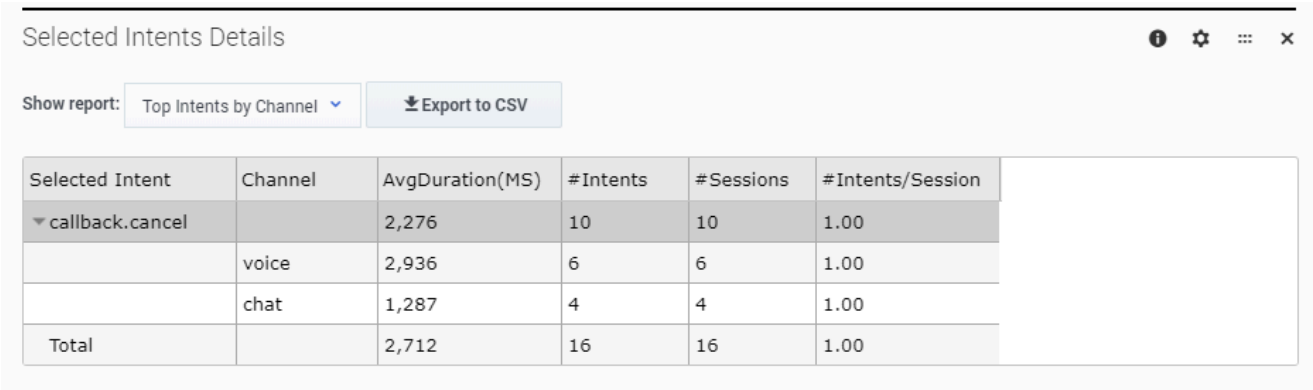
details provided in this panel include:

- **AvgDuration(MS)** — Average duration of time (in milliseconds) that the Bot Sessions were active, based on the period of time between the first request sent to the invoked bot service and the last response that was received.
- **SumStepcount** — Total number of requests that were sent to the bot service.
- **AbandonedInSS** — Total number of sessions that ended with a final disposition of **Abandoned in Self Service**.
- **RoutedToAgent** — Total number of sessions that ended with a final disposition of **Routed to Agent**.
- **RoutedToDN** — Total number of sessions that ended with a final disposition of **Routed to DN**.
- **AbandonedInQ** — Total number of sessions that ended with a final disposition of **Abandoned in Queue**.

If desired, you can also choose to export the results to a CSV file.

Selected Intents Details

This panel lists details for the top selected intents, broken down by channel type (voice or chat). If desired, you can export the results to a CSV file.



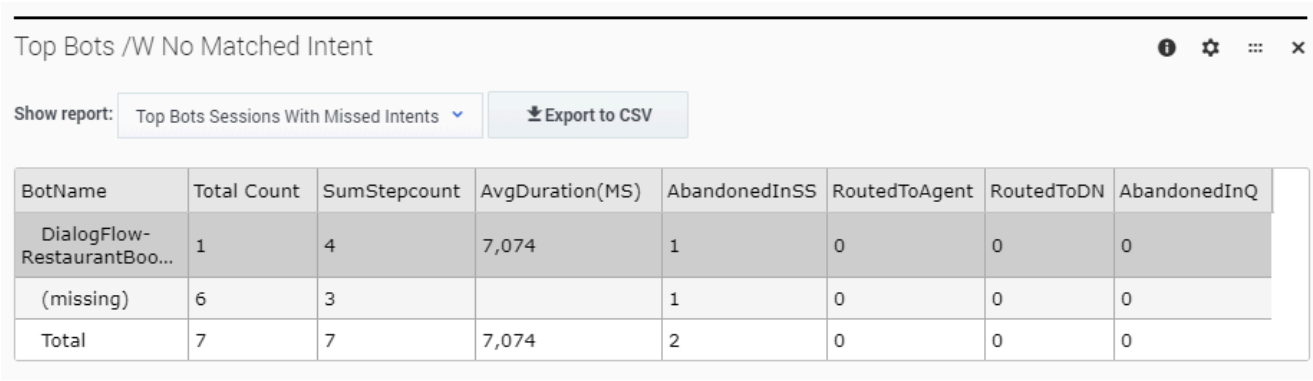
Selected Intent	Channel	AvgDuration(MS)	#Intents	#Sessions	#Intents/Session
callback.cancel		2,276	10	10	1.00
	voice	2,936	6	6	1.00
	chat	1,287	4	4	1.00
Total		2,712	16	16	1.00

Clicking an intent expands the row to display the totals by **Channel** type. Other details provided in this panel include:

- **AvgDuration(MS)** — Average duration of time (in milliseconds) that the Bot Sessions were active, based on the period of time between the first request sent to the invoked bot service and the last response that was received.
- **#Intents** — Total number of selected intents.
- **#Sessions** — Total number of sessions.
- **#Intents/Session** — Average total of selected intents per session.

Top Bots /W No Matched Intent

This panel provides information about sessions where a bot was invoked, but no intent was detected.


Top Bots /W No Matched Intent

Show report: Top Bots Sessions With Missed Intents | Export to CSV

BotName	Total Count	SumStepcount	AvgDuration(MS)	AbandonedInSS	RoutedToAgent	RoutedToDN	AbandonedInQ
DialogFlow-RestaurantBoo...	1	4	7,074	1	0	0	0
(missing)	6	3		1	0	0	0
Total	7	7	7,074	2	0	0	0

Details provided in this panel include:

- **AvgDuration(MS)** — Average duration of time (in milliseconds) that the Bot Sessions were active, based on the period of time between the first request sent to the invoked bot service and the last response that was received.
- **SumStepcount** — Total number of requests that were sent to the bot service.
- **AbandonedInSS** — Total number of sessions that ended with a final disposition of **Abandoned in Self Service**.
- **RoutedToAgent** — Total number of sessions that ended with a final disposition of **Routed to Agent**.
- **RoutedToDN** — Total number of sessions that ended with a final disposition of **Routed to DN**.
- **AbandonedInQ** — Total number of sessions that ended with a final disposition of **Abandoned in Queue**.

If desired, you can export the results to a CSV file.

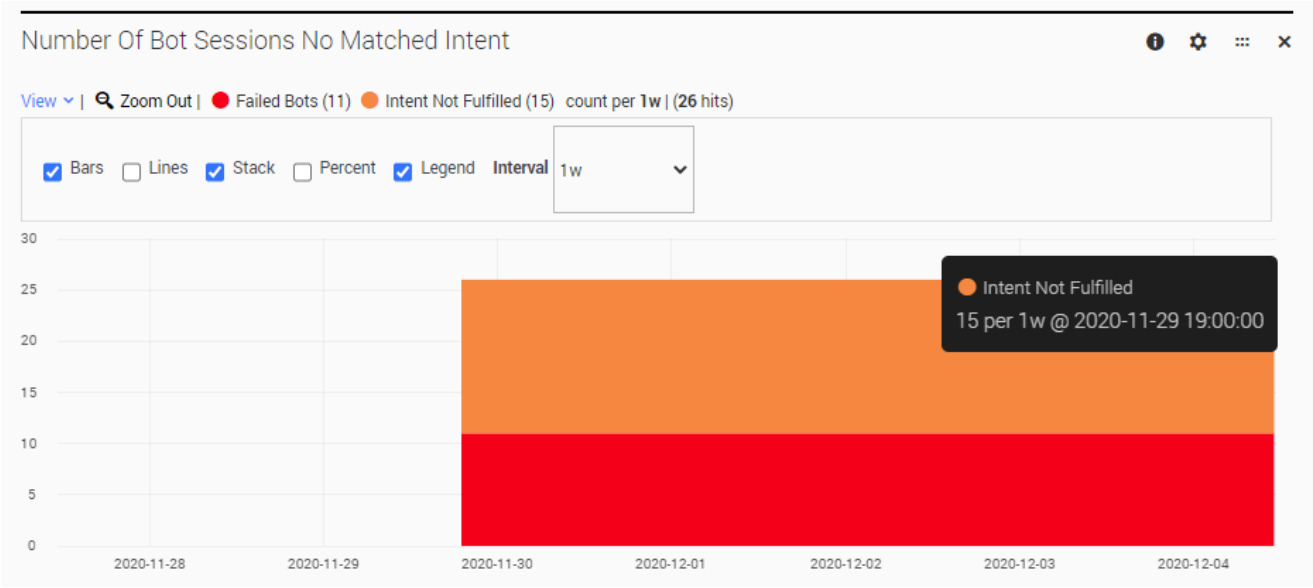
Number of Bot Sessions No Matched Intent

This panel expands on the results provided in the above report. It displays a total of all bot sessions broken down by **Failed Bots** and bot sessions that ended with **Intent Not Fulfilled**.

For a **Failed Bot**, the bot session experienced an error that prevented the bot service from functioning properly.

For **Intent Not Fulfilled**, the bot service was not able to match the customer's input to an intent. Everything worked, but the bot was not able to determine what the customer wanted to do.

You can expand the **View** menu to change the visualization options or select a specific **Interval** to display (such as 1 day). Hovering over the results displays the details for that item.



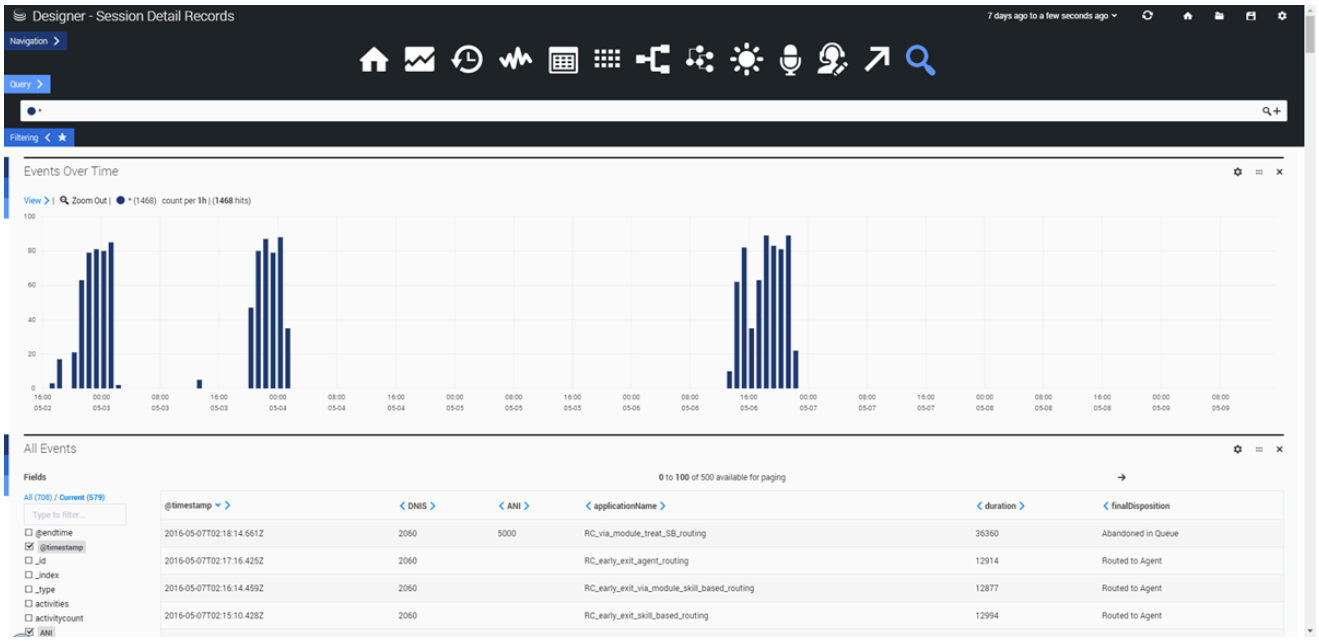
Important

In some cases, the reporting results can include a value of **(missing)**. This indicates a value was missing from one of the fields included in the data query.

Session Detail Records

> **Important**
>
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

The **Session Detail Records** dashboard lets you view (and query) some of the raw data contained in the Session Detail Records (SDRs).



Reports on this dashboard

Events Over Time

Similar to Count Over Time, this report shows the number of events that were logged during the given time period.

All Events

Basically, this is a table showing raw data information for all application sessions that were active during the given time period. You can use the **Fields** check boxes to select which columns to display. You can then search the results for the selected columns by toggling the **Query** option and entering your own query.

Searching the Session Detail Records

You can use the **Query** tab to enter a custom search query. For example:

```
applicationName:Joules\ Coulomb\ Direct\ Sales AND finalDisposition:Abandoned\ in \Self\
Service AND ANI:7031231234
```

This query would search the **Joules Coulomb Direct Sales** application for sessions where the final disposition was **Abandoned in Self Service** and the ANI was **7031231234**. Note that operators (such as **AND** or **OR**) are in caps and that a backward slash (\) is needed when using terms that include spaces.

> **Important**
>
> If you don't specify the SDR field(s) you are searching, Designer only matches the search terms against the following fields: **ANI**, **DNIS**, **SessionID**, **InteractionID**, **ConnectionID**, and **childIxns.id**. If you want Designer to search any other field, you must specify it in the query.

You can also use operators to search for fields that do (or do not) have values. For example, to include all SDRs where the **DNIS** field contains a value, you could use the following expression in your query:

```
DNIS:*
```

In this expression, the asterisk (*) acts as a wildcard to represent any value.

Or, if you wanted to include SDRs where the **DNIS** field does *not* contain a value (is blank):

```
-(DNIS:*)
```

In this expression, the minus sign (-) acts as a negative operator. The asterisk (*) again acts as a wildcard to represent any value, even one that is empty.

After the results are generated, click an event to expand it. You can then view its details in **Table**, **JSON**, or **Raw** format. When an event is expanded, you can also use the **Action** menu to add/remove the item from the filter, or to add/remove that column from the table.

Important

Available for Paging: This message might not display correctly. For example, you might see an inaccurate total provided (e.g. ***0 to 2*** *of 2 available for paging*, when it should say ***1 to 2*** *of 2...*), or asterisks (*) inserted where they shouldn't be (e.g. *2*00**). These anomalies are caused by an issue with Kibana (the third-party engine that drives the Analytics reports) and are not related to Designer.

Session Detail Record (SDR) Fields Reference

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

This page lists some of the most commonly used Session Detail Record (SDR) fields. It is not intended as a comprehensive list of all SDR fields used by Designer.

In most cases, these fields are automatically populated with system-generated values. But in some cases, the values are based on information provided by the application developer, such as the name of the country and the language of the application.

The SDR fields are organized according to the block category (or classification) they are most typically associated with.

> **Tip**
> Not sure which category your field belongs to? Use the search function in your browser to see if it's listed on this page. For example, in Chrome, press **CTRL + F** to open the search tool.

Session Data

The following SDR fields contain values that are sourced from data collected during various phases of the application session. They are mostly used for generating reports on the Designer Dashboards but can also be used for troubleshooting.

Values set by Designer

Field	Description
`@endtime`	Timestamp of when the interaction ended.

Field	Description
	Example: 2017-03-08T01:56:26.085Z
@timestamp	Timestamp of when the interaction started. Example: 2017-03-08T01:56:12.037Z
ANI	The caller's phone number (or Caller ID).
asduration	Length of time that the call was in the Assisted Service phase.
asend	Timestamp of when the Assisted Service (or IVR) phase of the interaction ended. If the interaction was routed, this value represents the time that the session exited the Self Service phase. Example: 2017-03-08T01:56:20.937Z
asstart	Timestamp of when the interaction entered the Assisted Service phase. Example: 2017-03-08T01:56:20.933Z
callEndReason	Indicates why a call ended, such as whether it was terminated in Self Service, Assisted Service, or due to a System Error.
DNIS	The phone number that the caller dialed.
duration	Duration of the total session, in milliseconds (ms). **Tip:** To calculate the duration of blocks execution and exclude session wrap-up time, subtract the value of the operationalOverheadDuration field from this value.
operationalOverheadDuration	Total time that the application was in an idle state (typically, this occurs just before the session finalizes).
finalDisposition	Status of an interaction at the time it exited the application flow, such as whether it was routed to an agent or the caller hung up. (See Count by Disposition for more information about dispositions.)
InteractionCategory	Channel type used for the interaction (voice, chat, or email).
InteractionID	Unique ID of the interaction. This ID can be used to track an interaction across multiple application sessions (or multiple instances of the *same* application).
ssduration	Amount of time that the caller spent in the Self Service phase of the application (in milliseconds).
ssend	Timestamp of when the Self Service (or IVR) portion of the interaction ended.
ssstart	Timestamp of when the Self Service phase of the application started.

Values set by application developer

These fields are populated with values that were provided by the application developer.

Field	Description
activities	List of activities encountered in the interaction. Activities are defined in an Activity block or automatically captured when an interaction enters or exits a shared module.
activitycount	Number of activities that were referenced in the interaction.

Field	Description
CountryName	Name of the country.
LanguageName	Language of the application.

Business Controls

These fields contain values sourced from Busines Controls blocks.

Field	Description
businesshourserrcount	Number of times a Business Hours block was accessed during the interaction.
businesshoursextcount	Total number of external business hours checks that occurred within a session.
businesshoursreqcount	Total number of requests for business hours checks that occurred within a session.
emergencieserrcount	Total number of emergency flag checks within a session.
emergenciesextcount	Total number of external emergency flag checks within a session.
specialdayserrcount	Total number of special days checked within a session (all checks in a single Special Day block count as one check).
specialdaysextcount	Total number of special days checked externally within a session (all checks in a single Special Day block count as one check).
specialdaysreqcount	Total number of exceptions encountered when checking special days (all errors in a single Special Day block count as one check).

Routing Data

These fields contain values sourced from Routing blocks.

Field	Description
routingBlockCount	Number of routing blocks that were hit within a session.
routingBlockTimeoutCount	Number of times that routing blocks timed out. Tip A high number here can indicate that customers are waiting too long in the queue for some reason. For example, skill levels might be set too high or there are not enough agents available.
routingCallHandlingType	Indicates the type of routing used for the call (default or consult).
routingLCAAttempted	Indicates if **Last Called Agent** routing was attempted (this option can be enabled on the Route Agent block).
routingSkillRelaxationCount	Number of routing blocks that used skill relaxation.

User Interaction

These fields contain values sourced from User Interaction blocks.

Field	Description
inputcount	Total number of User Input blocks executed during a session.
nicount	Total count of **No Input** instances that occurred for Input class blocks (User Input and Menu).
nmcount	Total count of **No Match** instances that occurred for Input class blocks (User Input and Menu).
strikeoutcount	Count of inputs where the maximum number of permitted attempts was exceeded.

External Requests

These fields contain values sourced from External Services blocks.

Field	Description
extreqcount	Total number of external requests.
extreqerrorcount	Total number of failed external requests.

User Milestone

These fields contain values sourced from user-defined milestones (such as those defined in a Milestone block or Menu block) and Survey blocks.

Field	Description
FinalUserMilestone	The last milestone in the milestones array at the end of the application session.
LastMilestone	Tracks the most recent milestone encountered.
milestones	List containing the system milestones that were encountered. Milestones indicate special points or transitions in the application, such as phases starting, phases ending, or an application terminating. **Note:** These values can be auto-populated by Designer or provided by the application developer.
userMilestones	Milestones that were defined in Milestone blocks or set in other blocks, such as Menu. **Note:** These values can be auto-populated by Designer or provided by the application developer.
usermilestonecount	Number of user-defined milestones hit in the session.
userMilestonesPath	Names of all milestones hit during the call.

Activity

These fields contain values sourced from Activity blocks.

Field	Description
activities	List of activities encountered in the interaction. Activities are defined in an Activity block or automatically captured when an interaction enters or exits a shared module.
activitycount	Number of activities that were referenced in the interaction.

SMART Inbound Voice Routing Application

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

The **SMART Inbound Voice Routing** application is a baseline application designed to operate with multiple **profiles** that you can define and manage using Business Controls.

At a basic level, the **SMART Inbound Voice Routing** application works as follows:

- A customer calls your contact center.
- The application checks the defined business controls (hours of business, special days, and emergency flags) and plays a corresponding message.
- The customer selects an option from a menu, which the application then uses to route the call to an appropriate agent.

You can use the application "as is", with the default profile, or create and customize additional profiles to offer a unique experience for a particular customer segment or line of business.

The application package includes several Designer object resources, such as shared modules, data tables, and media resources. You can learn more about the resources that are used by the SMART application here.

Creating a new application profile

The application profiles are contained in the M1 Profiles data table. While the default profile can be used for many business scenarios, the **SMART Inbound Voice Routing** application is structured to let you quickly create new application profiles that can be customized for your own business needs.

Keep in mind that the new application profile can only support settings that are available for SMART-type applications. It's best if the new application serves a similar business purpose or requires a similar type of flow as the original baseline application.

To create a new application profile, add a new row to the **M1 Profiles** data table and configure the settings for each of the columns. To edit the data table, go to **Business Controls** > **Data Tables** and select the **M1 Profiles** data table for editing.

Here are some recommendations and guidelines:

Profile ID and **Profile Name**
These must be unique. The ID could be a line of business (LOB), department name, DNIS, or another unique value that has business significance (for example, don't use the department name as an ID if you are setting up more than one application profile for it). The profile name will be used for reporting purposes, so you should give it a name that distinguishes it from any other profiles.

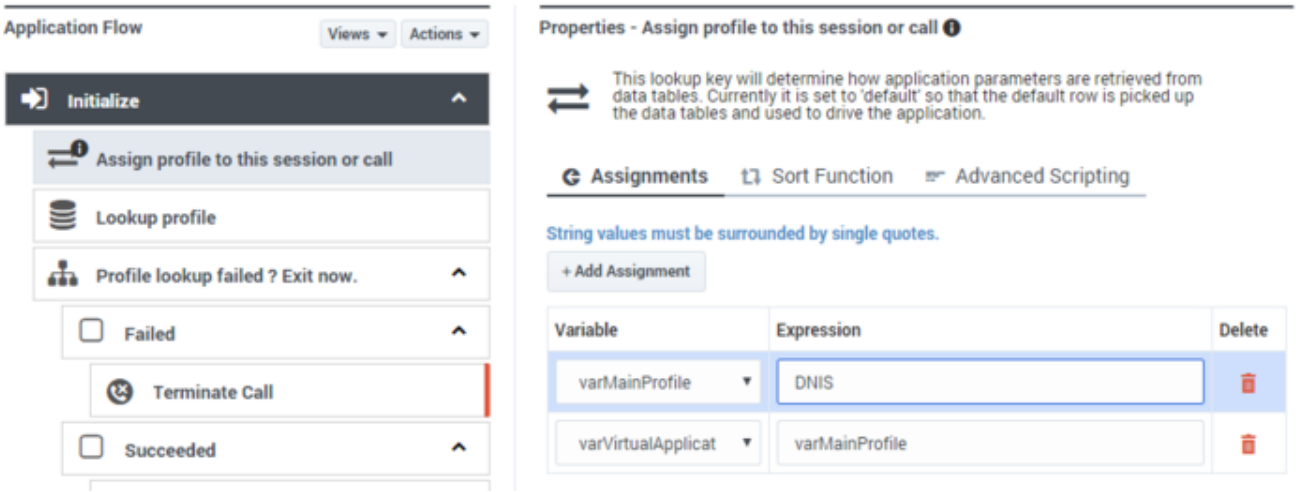
HOOPS Service ID
If the profile shares HOOPS (Hours of Operation) checks with another profile, you can just reuse the same HOOPS row and specify the ID. If not, add a new HOOPS row to the M1 HOOPS Services data table and specify its ID.

IVR Menu Service ID
If offering a menu, specify its ID from the M1 IVR Menus table. If not, leave it blank.

Routing Service ID
Specify the ID of a Routing Service from the M1 Routing Services table to use if the IVR Menus do not specify a routing service.

Assuming the new application profile uses existing HOOPS, IVR Menus, and Routing Services, it is ready to go. But you will also need to change the application initialization logic to select this newly created profile based on your criteria.

For example, if the profile ID is a DNIS, select the **Assign profile to this session or call** block and set `varMainProfile = DNIS`, as shown here:



For reporting purposes, in the Analytics dashboards, the application profiles are associated with the baseline SMART application, but each profile has reporting data associated with its assigned name.

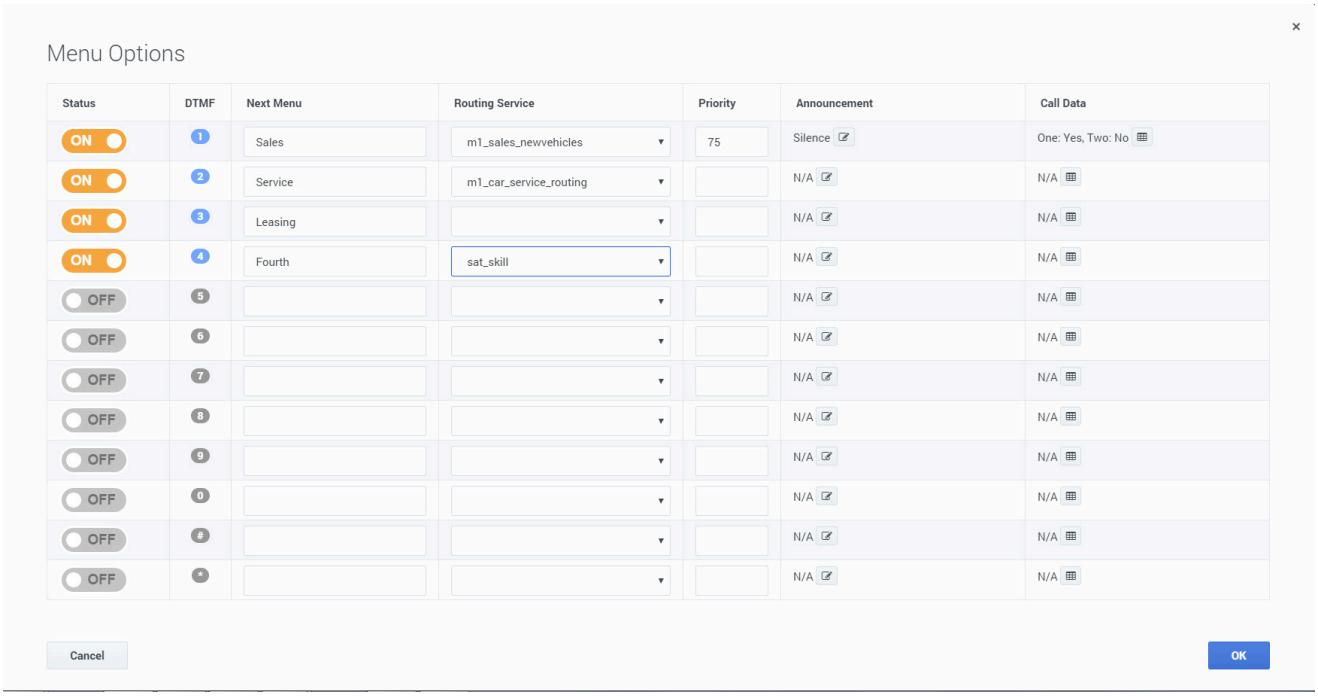
Creating or editing IVR menu options

To create a new menu service, go to **Business Controls** > **Data Tables** and select the M1 IVR Menus data table for editing.

Add a new row and specify the **Menu Name**, **Menu Audio**, and **Menu Level**. Set the other values according to how the menu will be used.

For example, if this menu is to be offered if the caller presses 2 from the first-level menu, you would set it up like this:

- In the first-level menu row, set **Menu Options** to use the new menu name. Let's say we've called it **Leasing**.
- In the example below, the **Next Menu** for **DTMF** option 3 has been set to **Leasing**, which becomes the second-level menu. If a caller presses 3 when the first-level menu is offered, they are offered the **Leasing** menu next.



- For this particular scenario, we don't want to specify a **Routing Service** for the **Leasing** option. If we do, the **M1 IVR Menus** module will end and the application will start routing the call according to the specified service. So it is left blank.
- All related menus must have the same **IVR Service ID**, so make sure that both the first-level menu option and the newly-defined second-level menu option have the same **IVR Service ID** value. Otherwise, they will not be treated as related menu levels.

Application resources

The following Designer objects are included as part of the SMART Inbound Voice Routing application package:

Application

- **SMART M1**
  This is the main application. The flow has been structured to provide some of the most common functions of a Designer application, such as calling modules to check for various conditions (like holidays or emergencies), offer self-service menus, and route calls to designated targets, while still allowing for some customization.

Shared Modules

Shared modules are smaller application "chunks" that work with their related data tables to perform various functions within the main application, such as checking for certain conditions, offering menus, or routing calls.

- **M1 HOOPS Checks**
  Checks for various conditions that can trigger an early-exit from the application. For example, it checks if there is an emergency flag, a holiday, or if the call is outside the scheduled hours of operation.

- **M1 IVR Menus**
  Presents a series of menu choices to a caller and then routes the call according to how the caller responds.

- **M1 Routing**
  Used for routing a call to an intended target (such as a **Skill Expression**, **Agent Group**, or **Direct Number**).

Data Tables

Data tables contain data values that are used by the related modules and other application objects. You can edit the data tables directly to add or make changes to various options.

M1 Profiles

Each row in this data table is a *profile* that influences how the application behaves and processes the incoming call. You can think of a profile as being like a *virtual application*, as each application profile will handle calls in a different way. To add a new application profile, simply add a new row to this table (see Creating a new application profile).

[+] Show table details

Display Name	Column ID	Description
Profile ID**	profile_id	Internal ID for uniquely identifying a profile. This can be a string, a VQ name, or the DNIS.

Display Name	Column ID	Description
		(There is a **default** profile already created.)
Profile Name	profile_name	The application profile name. This name helps to distinguish the various application profiles for reporting purposes.
Profile Description	profile_description	Description for this profile.
HOOPS Service ID	hoops_service_id	ID of the HOOPS Service defined in the HOOPS data table.
Default Menu Service ID	menu_service_id	ID of the IVR Menu Service defined in the IVR Menus data table.
Default Routing Service ID	routing_service_id	ID of the default Routing Service defined in the M1 Routing data table. This is used if the selected IVR Menus don't define their own routing service.

M1 HOOPS Services

This data table contains early-exit criteria objects, typically related to Business Controls settings. For example, it specifies which message to play when a call is received during an emergency, a holiday, or outside of the defined business hours.

[+] Show table details

Display Name	Column ID	Description
HOOPS Service ID**	hoops_service_id	Internal ID for uniquely identifying this service.
Emergency Flag	emergency_flag	Boolean value that indicates whether or not there is an emergency.
Emergency Message	emergency_message	Emergency message.
Open Hours	business_hours	Points to a business hours object.
Closed Hours Message	business_hours_message	Specifies the message to be played if the

Display Name	Column ID	Description
		current time is outside the defined business hours.
Holidays	special_days	Points to a special days object.
Holidays Messages	special_days_message	Message to be played if the current date is a holiday.
Queue Music	music_in_queue	(Reserved for future use.)

M1 IVR Menus

This data table contains multiple levels of nested menus that are offered to callers in the Self Service (IVR) phase. Each menu can enable DTMF options 1-9 and take one of the following actions for the selected option:

- Go to another menu, as per the target menu you've specified in the option settings, or
- Start routing the call, as per the routing service you've specified in the option settings.

[+] Show table details

Display Name	Column ID	Description
IVR Service ID**	ivr_service_id	ID for the IVR Menu Service.
Menu ID**	menu_id	Unique ID of a Menu within a Menu Service.
Menu Name	menu_name	User-friendly name for the Menu.
Menu Description	menu_description	Description of what the Menu does.
Menu Level**	menu_level	Numeric value indicating the nesting level of a Menu. The first (or top) level is 1.
Menu Audio	menu_audio	Announcement to be played that will offer the Menu.
Menu Options	menu_options	Options and actions for this menu.
Menu Type	menu_type	(Reserved for future

Display Name	Column ID	Description
		use.)

M1 Routing

This data table controls routing services, and contains definitions of multiple routing attempts, each of which can route to a different type of target (such as a **Skill Expression**, **Agent Group**, or **Direct Number**).

[+] Show table details

Display Name	Column ID	Description
Routing Service**	routing_service_id	Unique name for the routing service. Keep in mind that multiple attempts will have the same service if the attempts are related.
Routing Attempt**	routing_attempt	Integer indicating the attempt number. Attempts are made in increasing order.
Routing Option	routing_type	Type of routing. This must be one of **Skill**, **Direct Number**, or **Agent Group**. The exact string is required.
Skill Expression	routing_skill	Skill expression for routing.
Agent Group	routing_agent_group	Agent Group for routing.
Direct Number	routing_direct_number	Agent Group for routing.
Voice Mailbox	routing_voicemail	(Reserved for future use.)
Virtual Queue	virtual_queue	Virtual Queue for routing.
Queue Music	music_in_queue	In-queue audio.
Duration	routing_duration	Duration for routing attempt.

Business Hours

- **Main Office Hours**

The business hours are predefined as Mon-Fri, 9 AM to 5 PM.

Special Days

- **Office Holiday**
  The special days are predefined with a list of US holidays.

Media Resources

These include multiple audio files, as contained within the **Shared Audio** collection found under Media Resources.

> **Important**
> The SMART M1 application does not include any Speech Grammars.

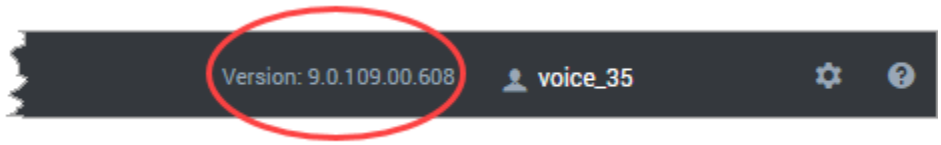
Troubleshooting

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

When troubleshooting an issue with Designer, you might be asked to provide Genesys with additional information that can be used to help resolve the issue you are having.

How to check your Designer version

Your Designer version is displayed in the top-right corner of the interface:



How to capture a network log

Most browsers allow you to capture an HTTP Archive (HAR) file, which contains details about how a particular web page has been handled by the browser. If you are asked to provide a HAR file to Genesys, follow the steps for your browser.

> **Tip**
> For best results, Genesys recommends that you generate the HAR file while using your browser's **private** (or **incognito**) browsing mode.

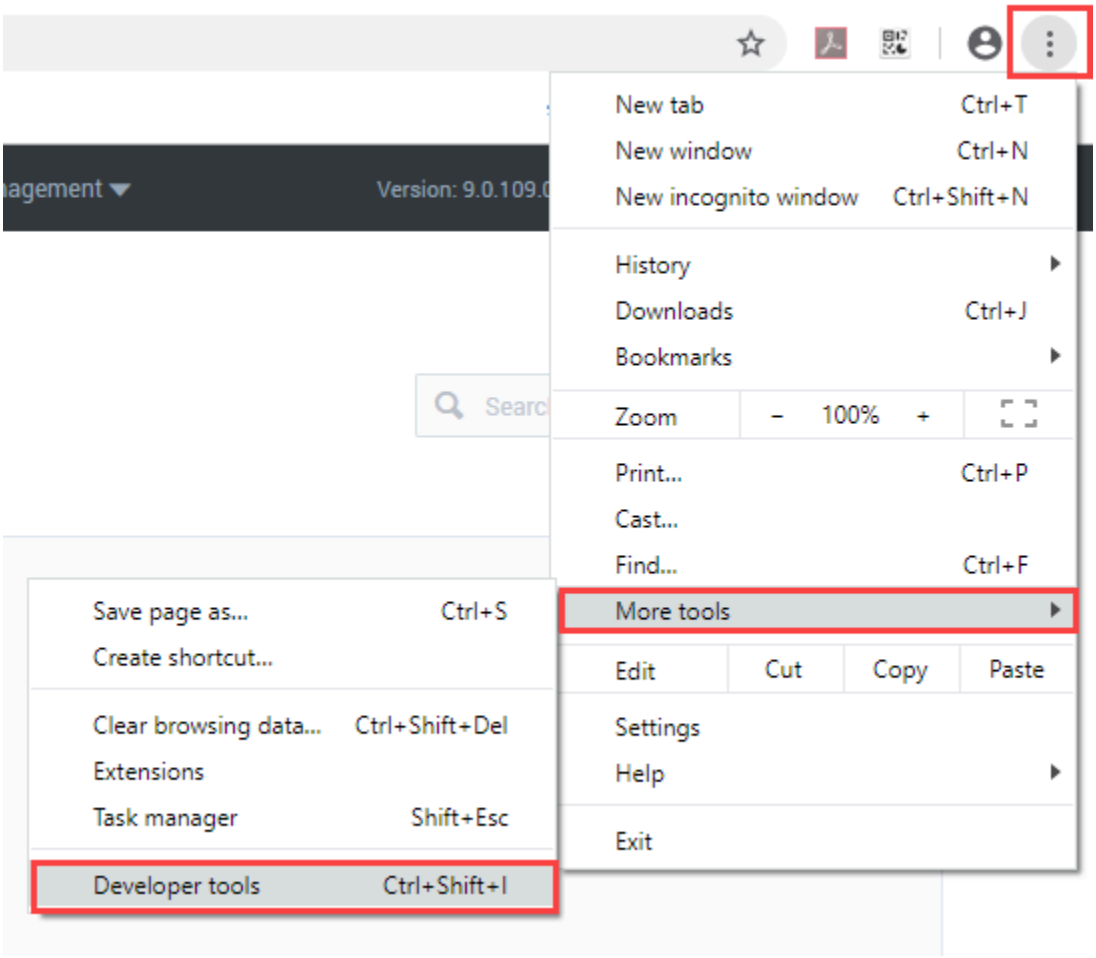
- Chrome
- Firefox
- Safari

- Edge

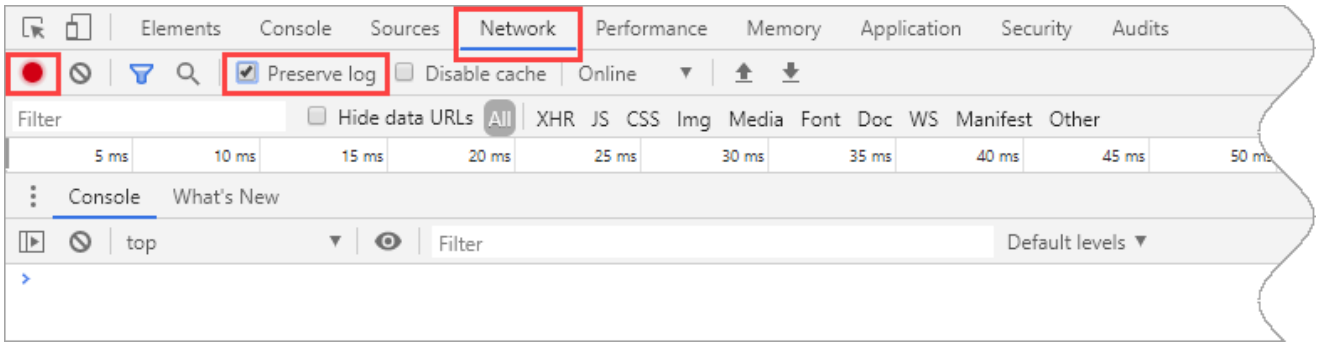
How to generate a HAR file using Chrome

Launch a new **incognito** window (you can do this from the Chrome settings menu, which appears as 3 vertical dots in the top right-side corner of the browser screen) and go to the Designer page where you are experiencing the issue.

Next, from the Chrome settings menu, go to **More tools** > **Developer tools** to launch the developer tools console.

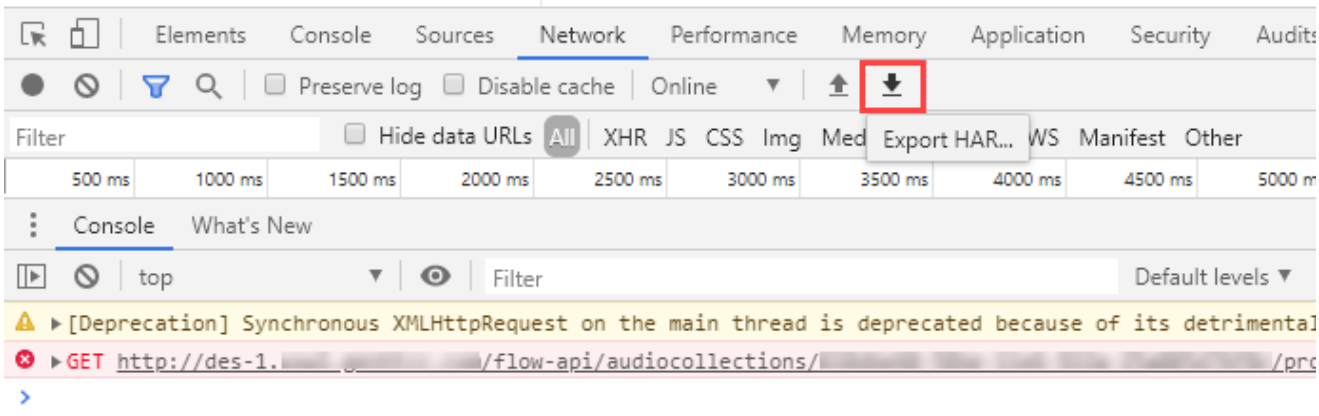


In the developer tools console, go to the **Network** tab and select **Preserve log**. Once you click the box, the red circle on the left indicates that the network activity is now being captured. (If the circle is still black, you can click it to toggle capturing on.)



While the capture is running, refresh the page and try to reproduce the issue you are having with the Designer page. When you are finished, you can stop the capture by deselecting the **Preserve log** box, or by clicking the red circle.

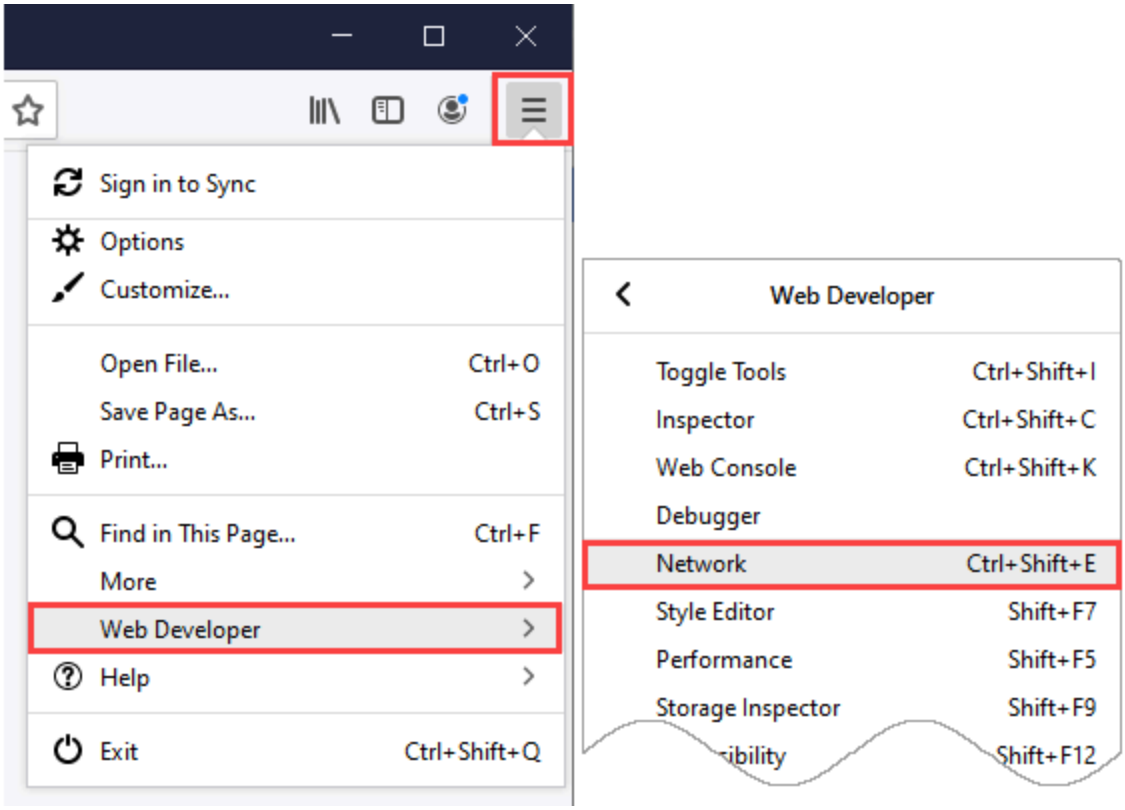
Click the **Export HAR** button to download the file to your computer.



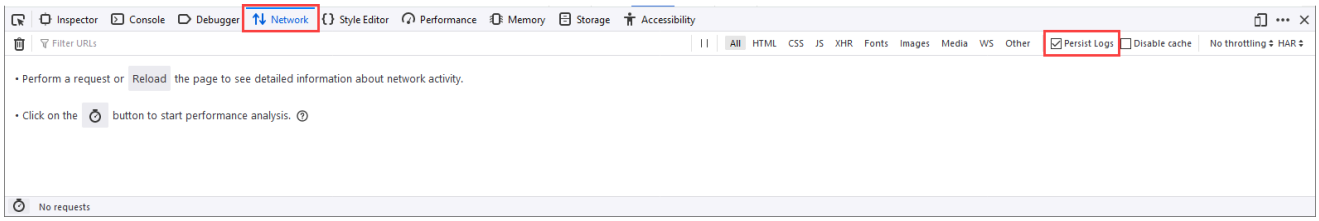
How to generate a HAR file using Firefox

Open a new **Private** window (you can do this from the Firefox settings menu, which appears as 3 vertical lines in the top right-side corner of the browser screen) and go to the Designer page where you are experiencing the issue.

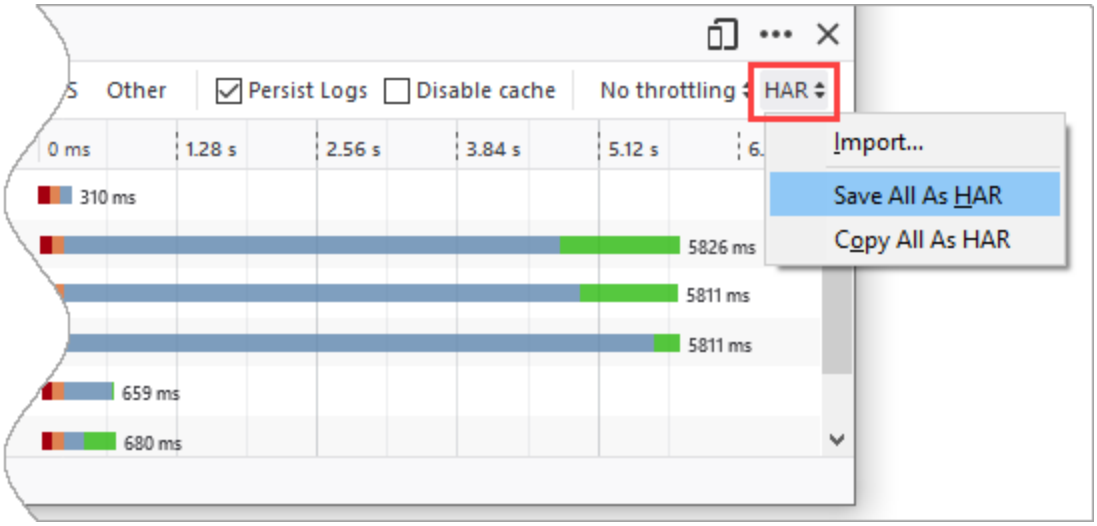
Next, from the Firefox settings menu, click **Web Developer** and select **Network** to launch the developer console.



In the developer console, go to the **Network** tab and select **Persist Logs**.



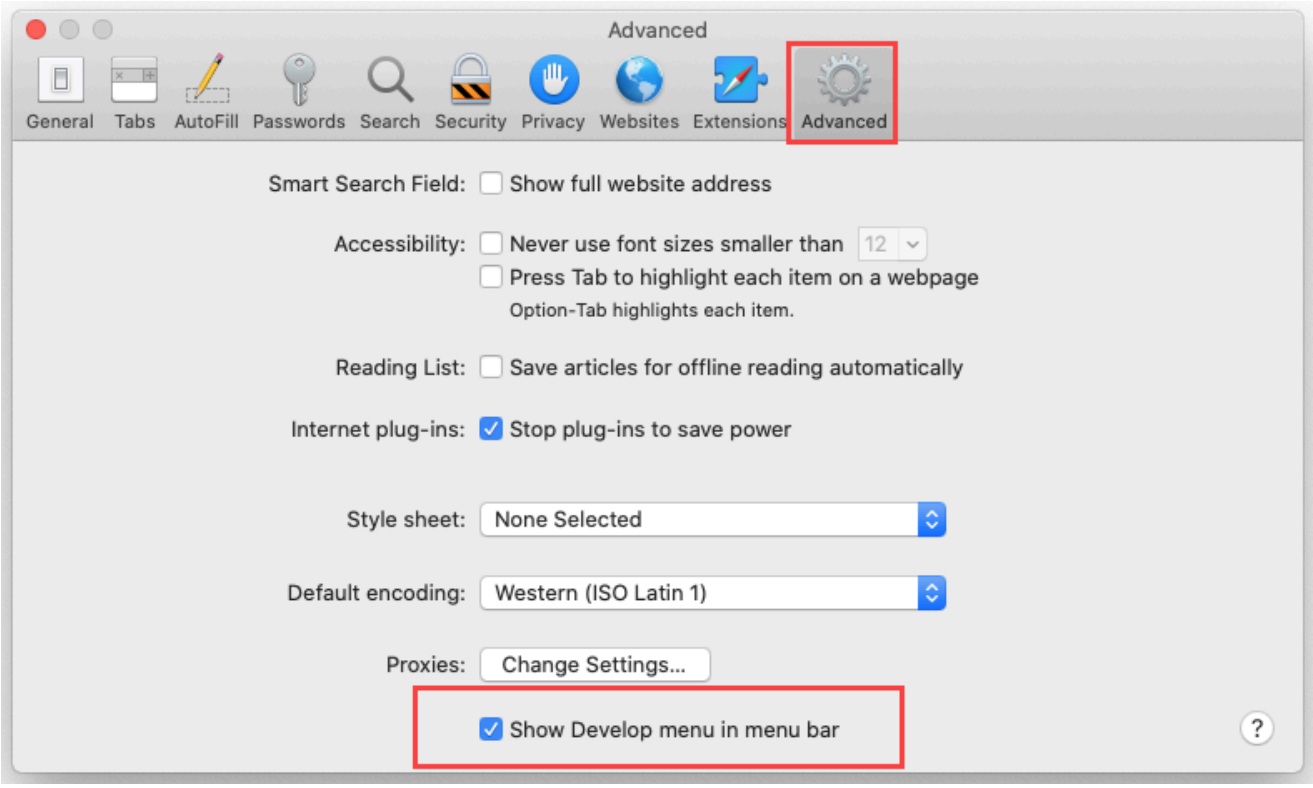
Refresh the Designer page and/or perform the action that isn't working correctly. When you are finished, click the **HAR** button and select **Save All As HAR** to download the file to your computer.



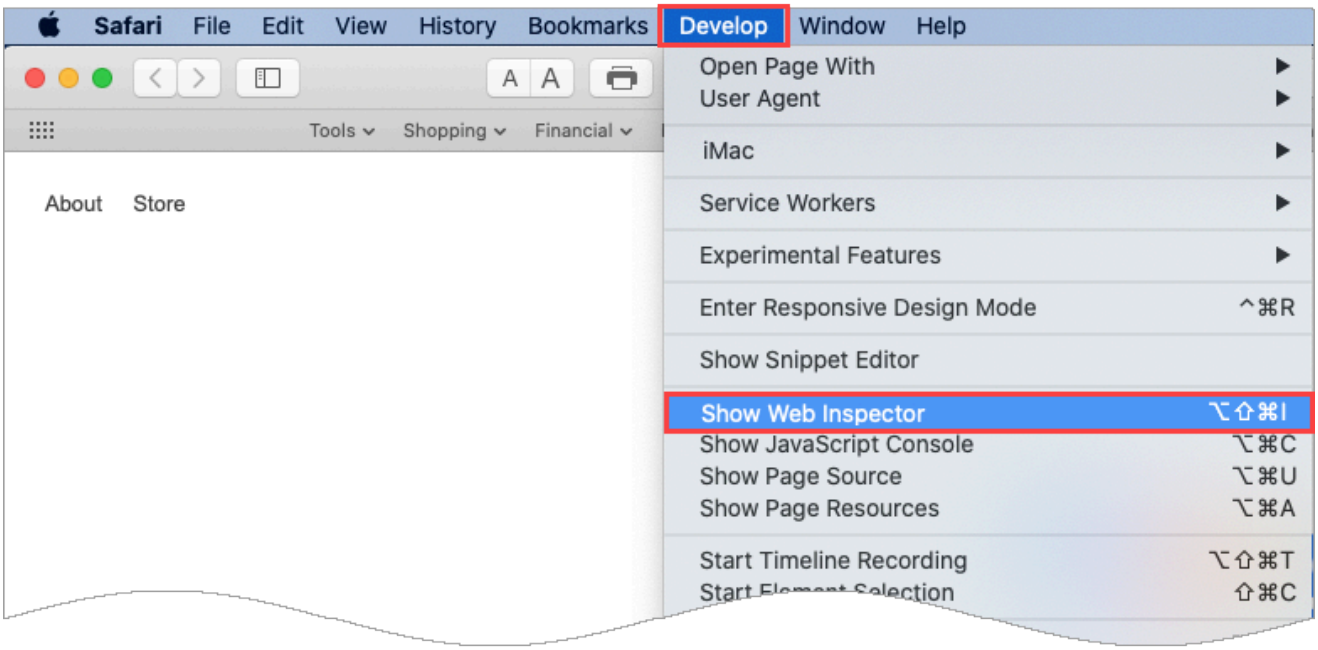
How to generate a HAR file using Safari

Open a new **Private** window (you can do this from the **File** menu) and go to the Designer page where you are experiencing the issue.

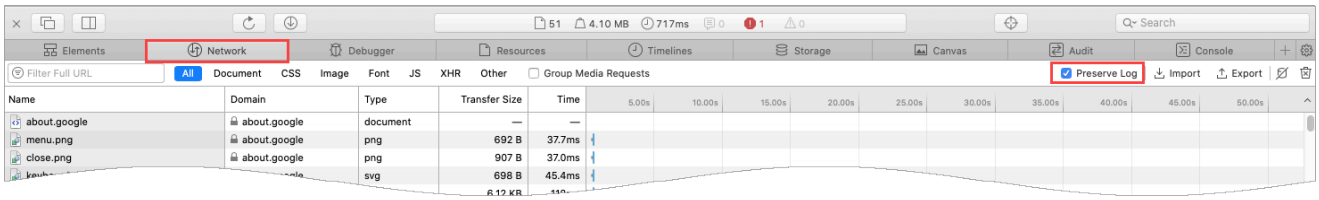
Next, from the **Safari** menu, select **Preferences**. Click the **Advanced** tab and enable the **Show Develop menu in menu bar** option.



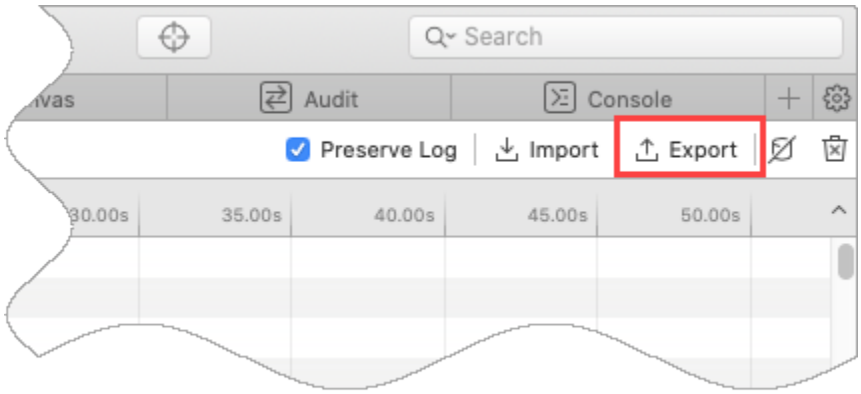
From the **Develop** menu, select **Show Web Inspector**.



In the **Web Inspector** console, go to the **Network** tab and select **Preserve Log**.



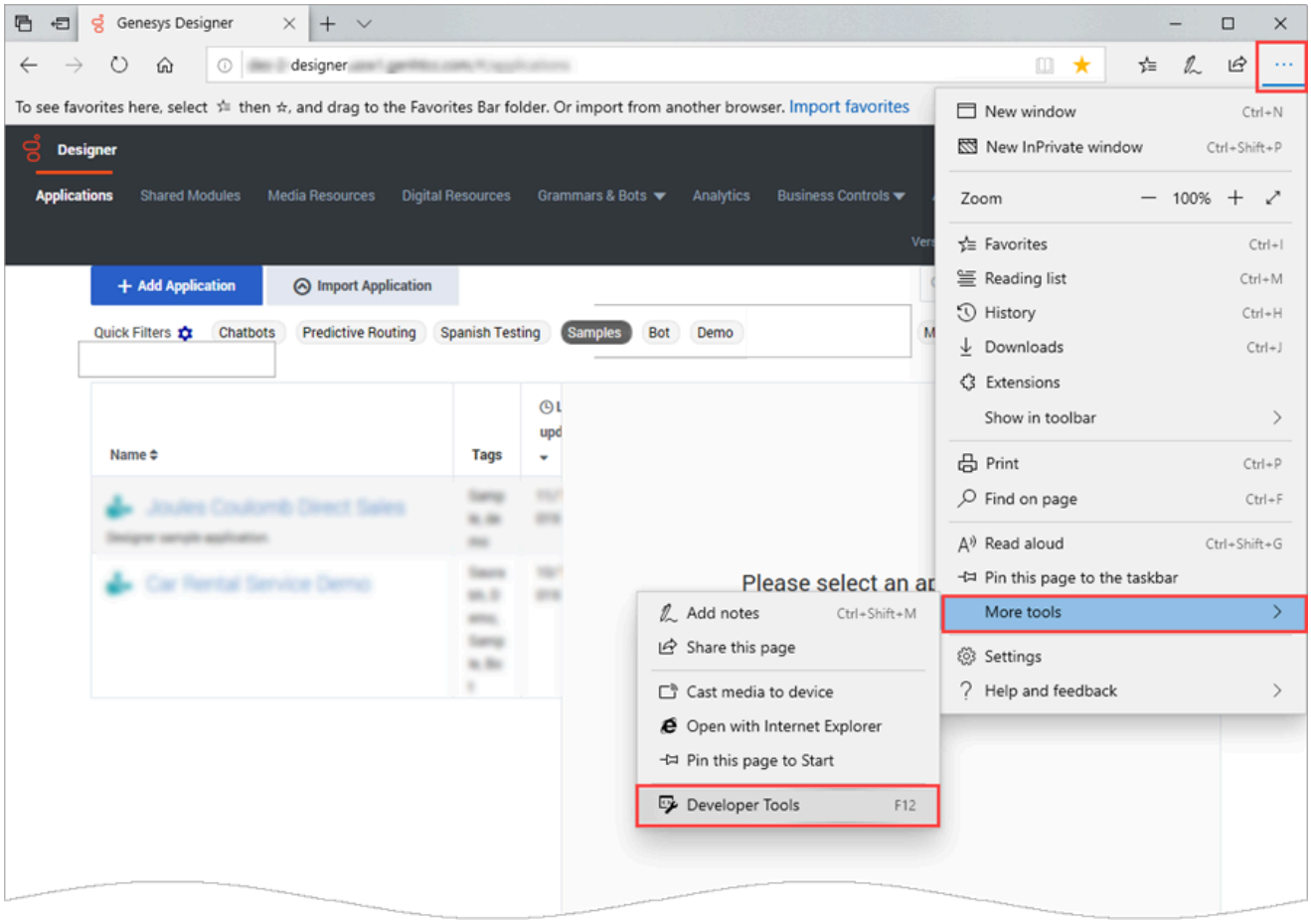
Refresh the Designer page and/or perform the action that isn't working correctly. When you are finished, click the **Export** button to download the file to your computer.



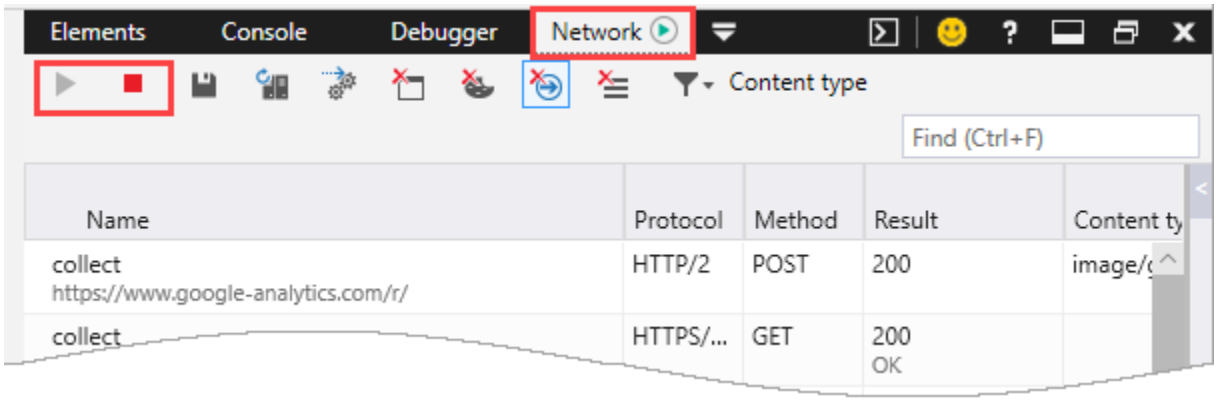
How to generate a HAR file using Edge

To start, open a new **InPrivate** window (you can do this from the settings menu, which appears as 3 horizontal dots in the top right-side corner of the browser screen) and go to the Designer page where you are experiencing the issue.

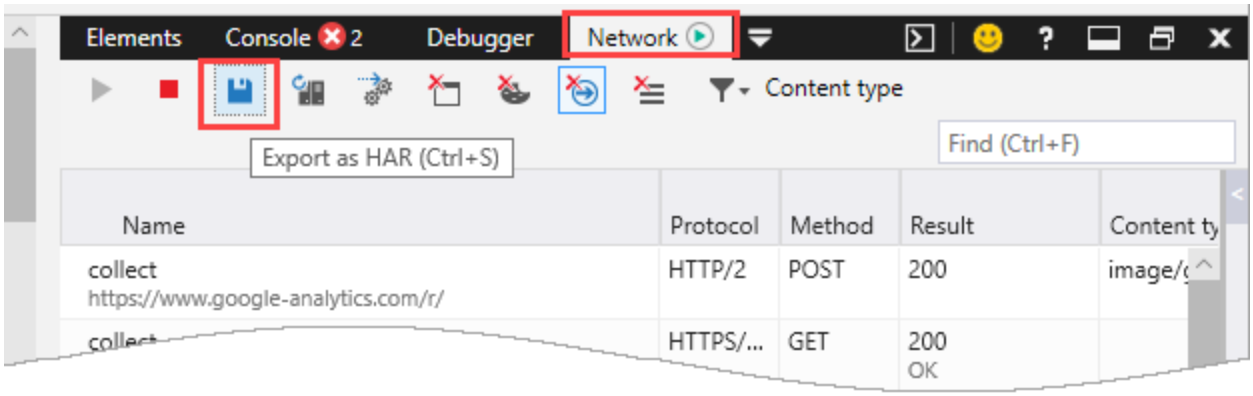
Next, from the Edge settings menu, go to **More tools** > **Developer tools** to launch the Developer Tools console.



In the Developer Tools console, go to the **Network** tab. You can toggle the capture on/off using the **Play** (triangle) and **Record** (square) buttons. If the record button is not red (i.e. active), click it to start capturing data.



Refresh the Designer page and/or perform the action that isn't working correctly. When you are finished, click the **Export HAR** button to download the file to your computer.


Elements
Console 2
Debugger
Network
Content type
Export as HAR (Ctrl+S)
Find (Ctrl+F)
Name
Protocol
Method
Result
Content ty
collect
https://www.google-analytics.com/r/
HTTP/2
POST
200
image/
collect
HTTPS/...
GET
200
OK

Digital Channels setup

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to the Digital Channels Administrator's Guide.

Digital Channels powers your customer interactions across channels such as chat, SMS, messaging and social media. It provides a platform that enables you to grow sales, create more targeted marketing campaigns, and deliver exceptional customer service. The Digital Channels service processes, manages and archives customer and agent interactions across media.

Enabling digital channels in your contact center opens up Agent Desktop to chats from Facebook, Twitter, WhatsApp and SMS that are treated just like regular Genesys interactions. When customers communicate with your company on one of these channels, Genesys matches them against customers already in the contact database. If there's a match, the agent handling the interaction has access to all previous interactions with the contact. Until the interaction is marked Done, agents can also return to the chat conversation at any time in the future — for example, they might need to take time to find additional information for the contact or initiate a business process in your company.

Facebook

Using the Facebook channel, agents can monitor your business presence and identify and respond to online comments. Facebook Messenger direct messages and comments and replies on your company's Facebook page are monitored through Genesys Hub and then automatically routed to agents across your enterprise who are using Workspace Agent Desktop.

For more about enabling Facebook in your contact center, see Set up Facebook and Twitter.

Twitter

Using the Twitter channel, agents can monitor your business presence and identify and respond to online comments. Twitter Direct Messages and Tweets and replies on your company's Twitter page are monitored through Genesys Hub and then automatically routed to agents across your enterprise who are using Workspace Agent Desktop.

For more about enabling Twitter in your contact center, see Set up Facebook and Twitter.

WhatsApp

WhatsApp is a messaging service that enables the exchange of text messages, emojis, images, and Highly Structured Messages (HSM) between agents and customer contacts. Typically, a contact will send an agent a message from a mobile device or computer. These messages are captured by Genesys Engage through Genesys Hub and then automatically routed to agents across your enterprise who are using Workspace Agent Desktop.

For more about enabling WhatsApp in your contact center, see Set up WhatsApp.

SMS

Short Message Service (SMS) is a messaging service that enables the exchange of text messages between agents and customer contact. Typically, a contact will send an agent a message from a mobile device. These messages are captured by Genesys Engage and then automatically routed to agents across your enterprise who are using Workspace Agent Desktop.

For more about enabling SMS in your contact center, see Set up SMS.

Secure Messaging

Consumers often try to solve problems through email due to its convenience. Sometimes these problems involve sensitive information that should only be delivered to the consumer, such as transaction history or an account number. In order to make sure this information goes only to the consumer, you must have a secure login and email client that are either built in-house or provided through a third party.

Genesys' role is not to provide this secure messaging platform, but to allow our solution to work with one that you provide. Our solution sends and receives messages from your secure messaging platform and notifies the consumer when they have a new message in the secure messaging platform.

For more information about enabling Secure Messaging in your contact center, see [GSESM|Set up Secure Messaging]].

Set up Facebook and Twitter

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Set up Facebook and Twitter.

Genesys Social Engagement enables you to receive and send messages, posts, comments, likes, tweets, and retweets on the world's top two social engagement platforms, Facebook and Twitter.

After you complete the steps on this page, you can treat social media communications like any other Genesys interaction. For example, you can route tweets with keywords to the best-skilled agent using a Designer application. Reporting is also available through Genesys CX Insights.

1. Configure channels in Genesys Hub

Contact your Genesys representative to access Genesys Hub and then complete the steps on Integrate with Genesys Hub to setup your connections to Facebook and Twitter.

2. Enable Digital Shared Services

Work with your Genesys representative to enable service channels in Agent Setup to connect to the channels you created in Genesys Hub. Later when you create a Designer application to route social media interactions, it will pull the data from these channels.

3. Configure Agent Desktop

You should go through the normal process to configure any necessary desktop options for the agents in your contact center, in particular Facebook and Twitter options. See the Agent Setup documentation for more information.

4. Create a Designer application

Create a Designer application to route social media interactions to the right agents. **Note:** You should see the service channel you created in step 2 listed under **Manage Digital Endpoint** in Designer.

Related documentation

See how users handle social media interactions in Agent Desktop:

- Facebook and Twitter for agents.
- Facebook and Twitter for supervisors.

Set up WhatsApp

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Set up WhatsApp.

Genesys Digital Channels enables you to integrate WhatsApp into Agent Desktop. The WhatsApp messaging service enables the exchange of text messages, emojis, images, and Highly Structured Messages (HSM) between your agents and customers.

After you complete the steps on this page, you can treat WhatsApp communications like any other Genesys interaction. For example, you can messages to the best-skilled agent using a Designer application. Reporting is also available through Genesys CX Insights.

1. Configure channels in Genesys Hub

Contact your Genesys representative to access Genesys Hub and then complete the steps on Integrating with Genesys Hub to setup your connection WhatsApp.

2. Enable Digital Shared Services

Work with your Genesys representative to enable service channels in Agent Setup to connect to the channels you created in Genesys Hub. Later when you create a Designer application to route social media interactions, it will pull the data from these channels.

3. Configure Agent Desktop

You should go through the normal process to configure any necessary desktop options for the agents in your contact center, in particular Chat options. See the Agent Setup documentation for more information.

4. Create a Designer application

Create a Designer application to route WhatsApp interactions to the right agents. **Note:** You should

see the service channel you created in step 2 listed under **Manage Digital Endpoint** in Designer.

Related documentation

See how users handle WhatsApp interactions in Agent Desktop: WhatsApp for agents and WhatsApp for supervisors.

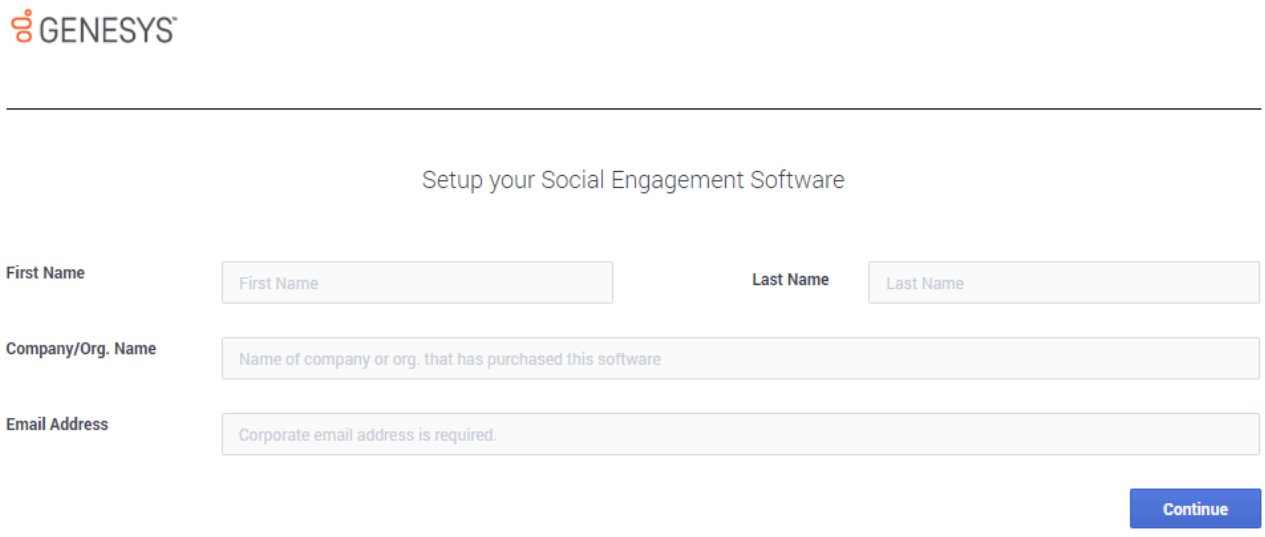
Integrate with Genesys Hub

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Integrate with Genesys Hub.

Genesys Hub is our centralized platform that connects to Twitter, Facebook and WhatsApp. Complete the steps below to set up an account with Genesys Hub and enable channels and services, then return to Set up Facebook and Twitter or Set up WhatsApp to finish the rest of the setup.

> **Important**
> Contact your Genesys representative to get access to Genesys Hub.

Set up your account



Go to https://socialanalytics.genesyscloud.com/gse/signup and complete the form.

You should receive a confirmation email titled "Confirmation required for Genesys Social Engagement

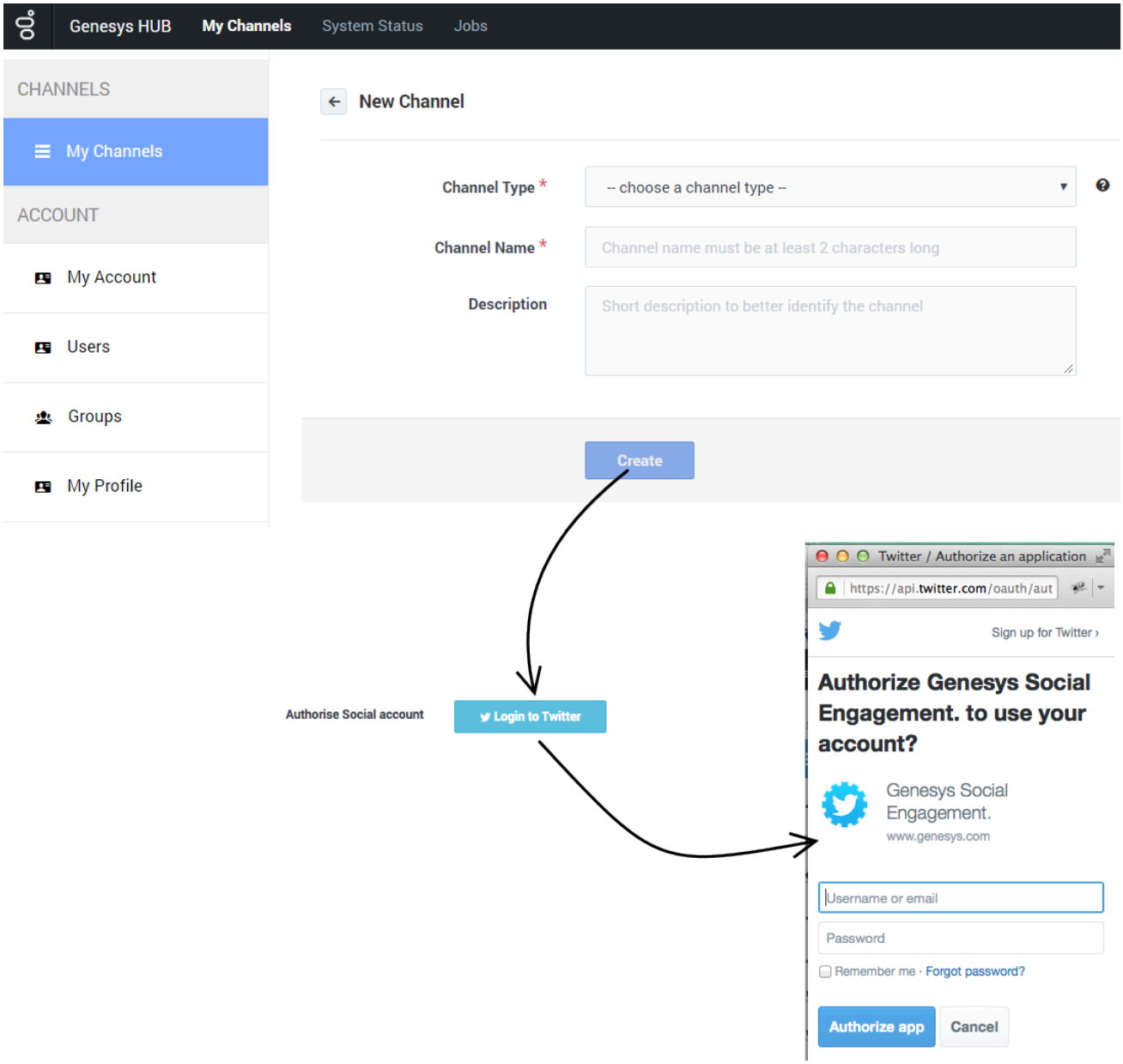
deployment" (don't forget to check your spam folder). Follow the steps in the email and login to Genesys Hub.

Note: You can always use https://socialanalytics.genesyscloud.com to access Hub. If you prefer a static URL, you can use https://static.socialanalytics.genesyscloud.com instead.

Set up Twitter

Complete the steps below to set up Twitter account and service channels in Genesys Hub.

Create a Twitter account channel



Create a Twitter account channel to authenticate with Twitter using your organization's Twitter handle (login) and password.

Under **My Channels** in the left navigation menu, click **+Add Channel**.

- Choose *Twitter:Account* for the **Channel Type**.
- Set the **Channel Name** to `<your channel's Twitter handle> - Twitter Account`. Example: `@Genesys - Twitter Account`.

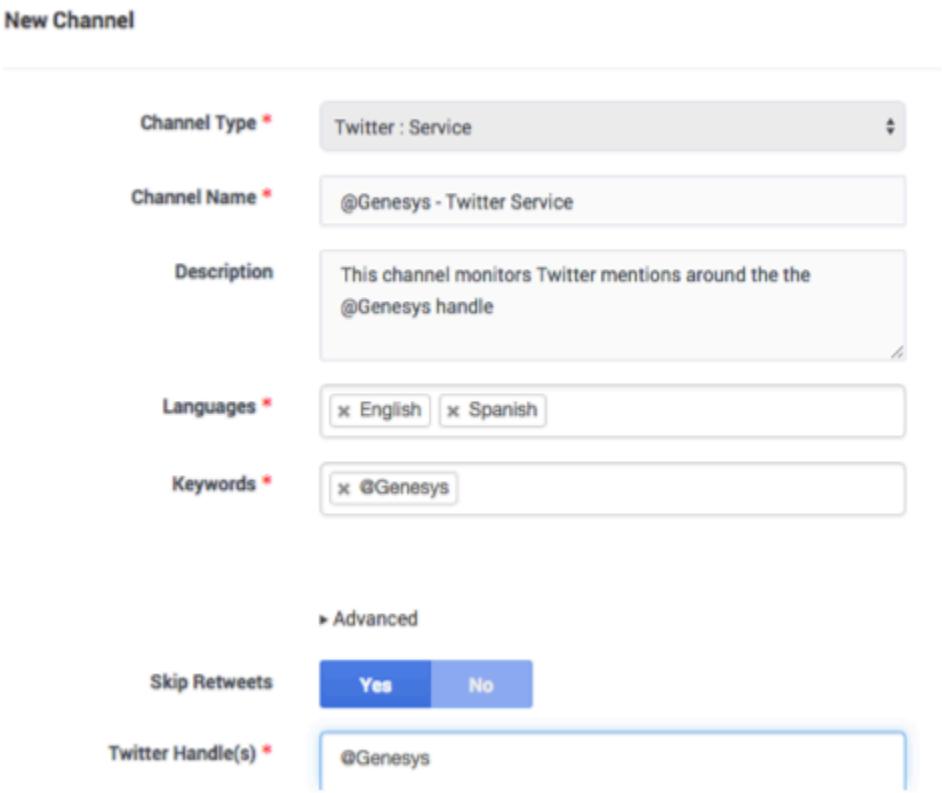
- Add an optional description of how you will use the channel.

Save your channel and click **Login to Twitter**.

Enter the Twitter credentials for your organization: username (handle) or email, and the password associated with your Twitter account. **Important:** Also check **Remember me**.

If you have successfully authenticated, you will see the following message: **Access_token successfully saved**.

Create a Twitter service channel



Create a Twitter service channel to listen for mentions of your keyword(s) on Twitter.

Under **My Channels** in the left navigation menu, click **+Add Channel**.

- Choose *Twitter:Service* for the **Channel Type**.
- Set the **Channel Name** to `<your channel's Twitter handle> - Twitter Service`. Example: `@Genesys - Twitter Service`.
- Add an optional description of how you will use the channel.

After saving your channel, the application displays an extended view where you can configure additional settings:

- Add any applicable languages.
- For **Keywords**, enter the handle you want to monitor for mentions, for example `@Genesys`, then hit Return or Tab. You can include other words or phrases to listen for, such as `#Genesys`. These keywords are the way Hub gathers mentions from Twitter -- there is no mechanism for excluding posts with certain mentions. See Best Practices for Managing Keywords in Twitter Service Channels below for more information about using a broader range of words and phrases.
- If you choose **Skip Retweets**, your system will not source tweets that have been retweeted, which may help cut out clutter and noise. You may want to select **No** if you want to analyze retweets and viral activity.
- In **Twitter Handle(s)**, just type the Twitter handle that you will be using to respond, for example `@Genesys`. You can only respond with one handle per channel.

Best practices for managing keywords in Twitter service channels

When setting up a Twitter:Service channel, you can add any text in the **Keyword** field. For example, if we set up a channel to monitor mentions of Genesys on Twitter, some appropriate keywords might be *@Genesys*, *#Genesys* and *Genesys software*.

Make sure to use keywords that are specific to your business or use case and not too broad or ambiguous. You can check this using Twitter's search functionality at https://twitter.com/search-home. Enter each keyword, one at a time, and assess the search results. For example, if your company is called "United Ultra Local Bank Inc.", you might want to use the keyword *#United.* This would be a mistake, as it would return a flood of posts related to topics as broad as *#United Nations*, *#United States*, *Manchester #United*, and so on. Even worse would be *#Check, savings account, #balance,* or *overdraft* as these are even more generic.

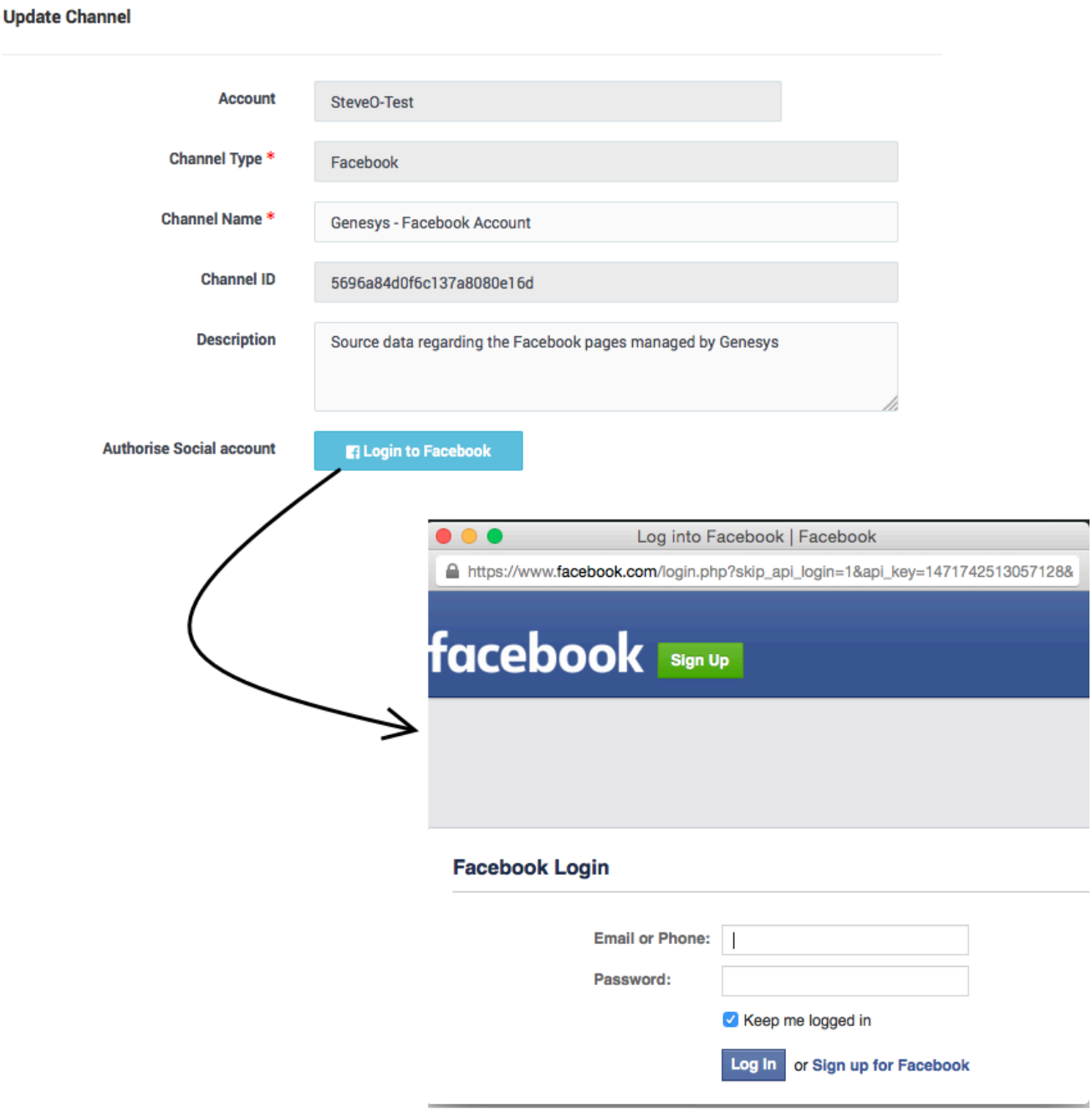
Ambiguous keywords can clutter your inbound feed and, in extreme cases, Twitter might cut off access to data if the volumes are too high. If a channel's inbound traffic exceeds 300 tweets per minute for a certain keyword, Genesys Hub automatically removes the keyword from the channel to keep data volumes from exceeding Twitter's thresholds. After three days, Hub unblocks the keyword and you can add it back to your channel. If you need to add the keyword before the three days are up, contact your Genesys representative.

After you re-add the keyword, you can run a recovery task to get the data that was not monitored while the keyword was blocked. **Note:** Hub will not re-block the keyword while data recovery is running.

Set up Facebook

Complete the steps below to set up Facebook account and service channels in Genesys Hub.

Create a Facebook account channel



Create a Facebook account channel to authenticate with Facebook using your organization's Facebook administrator login credentials. These user credentials must have admin access to all the pages you want to associate with this channel. For simplicity, make sure you are logged into Facebook with the same browser you're using for this setup.

Under **My Channels** in the left navigation menu, click **+Add Channel**.

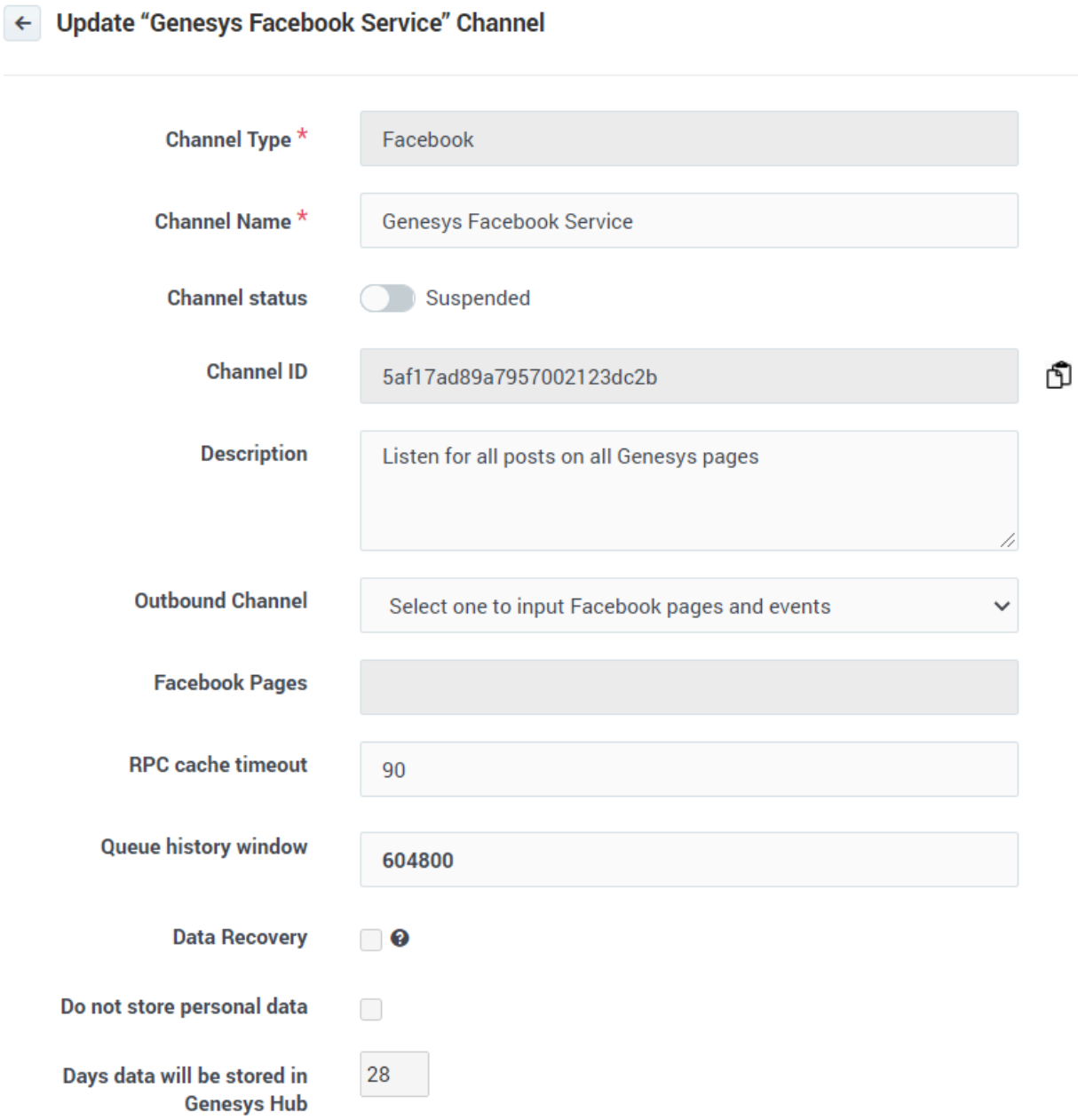
- Choose *Facebook:Account* for the **Channel Type**.
- Set the **Channel Name** to `<company name> - Facebook Account`. Example: `Genesys - Facebook Account`.
- Add an optional description of how you will use the channel.

After saving your channel, the application displays an extended view where you can **Login to Facebook**.

Use the Facebook admin credentials for your organization. **Important:** Also check **Keep me logged in**.

If you have successfully authenticated, you will see the following message: **access_token successfully saved, close window**.

Create a Facebook service channel



Create a Facebook service channel to listen for posts, comments and messages on Facebook. Under **My Channels** in the left navigation menu, click **+Add Channel**.

- Choose *Facebook:Service* for the **Channel Type**.
- Set the **Channel Name** to `<company name> - Facebook Service`. Example: `Genesys - Facebook Service`.

- Add an optional description of how you will use the channel.

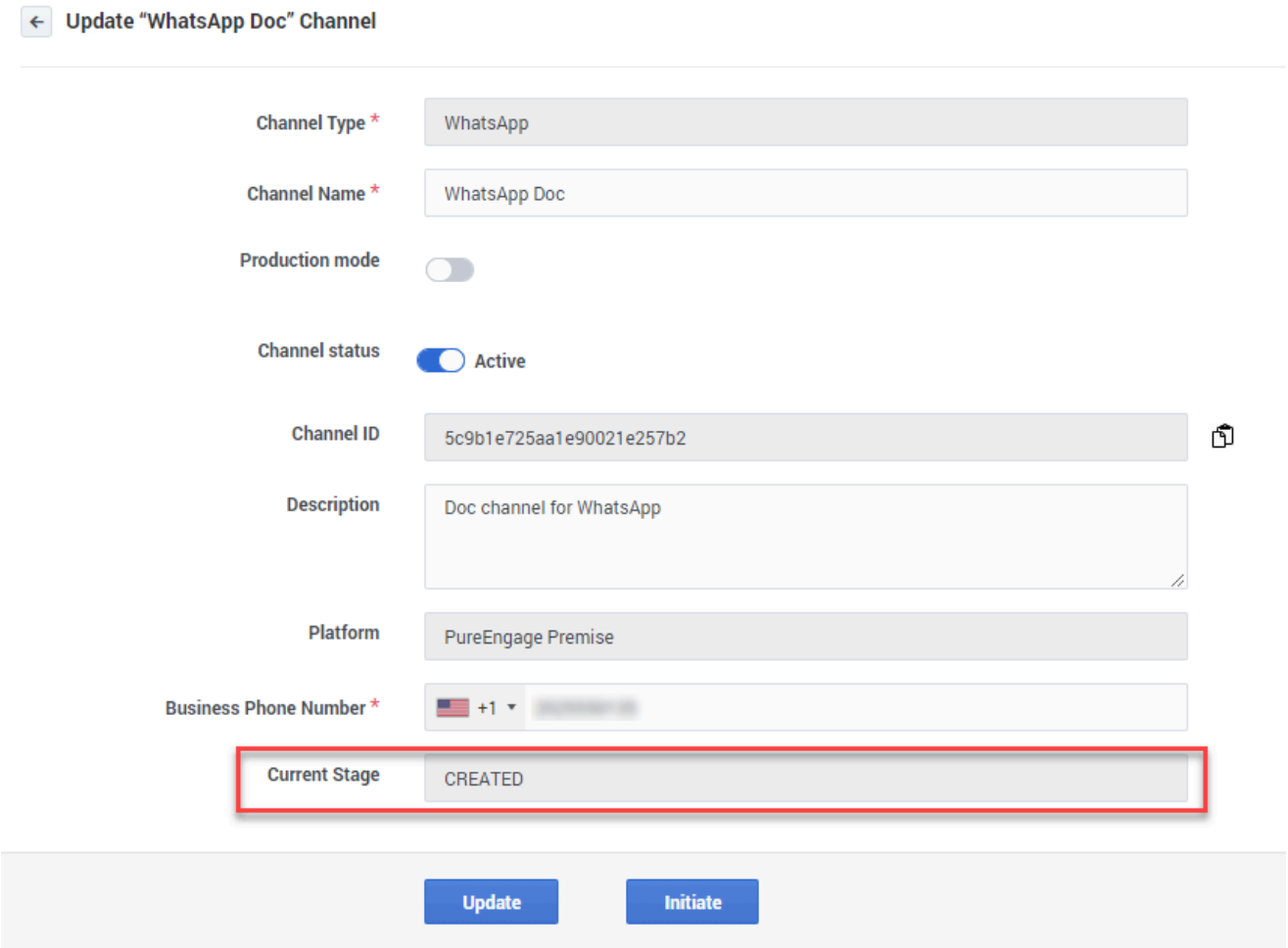
After saving your channel, the application displays an extended view where you can configure additional settings:

- Choose the pages you want to monitor from the list under **Facebook Pages**. This list is made up of all the pages your login account can administer.

Set up WhatsApp

Complete the steps below to set up a WhatsAPP account and service channels in Genesys Hub.

Create a WhatsApp service channel



Create a WhatsApp service channel to communicate with your customers.

Under **My Channels** in the left navigation menu, click **+Add Channel**.

- Choose *WhatsApp: Service* for the **Channel Type**.
- Choose *Genesys Engage cloud* for the **Platform**.
- Enter your business phone number. **Note:** If you want to create more than one WhatsApp channel, you must use a unique phone number for each channel.
- Enter a channel name. This is used as the phone display name in WhatsApp and is visible to consumers interacting with the brand in WhatsApp Messenger. You must ensure that the name complies with WhatsApp policies: https://developers.facebook.com/docs/whatsapp/guides/vname.
- Add an optional description of how you will use the channel.

After you create the channel, it goes through the following stages in the activation process:

1. CREATED - The first stage after channel creation.
2. INITIATED - The channel is under review by Genesys.
3. VERIFIED - The channel information was reviewed and is correct.
4. REGISTERED or REJECTED - The channel can be registered or rejected by the Genesys team based on the provided channel data.
5. ACTIVATED - The channel is activated and is enabled for WhatsApp Business.

Register a WhatsApp service channel

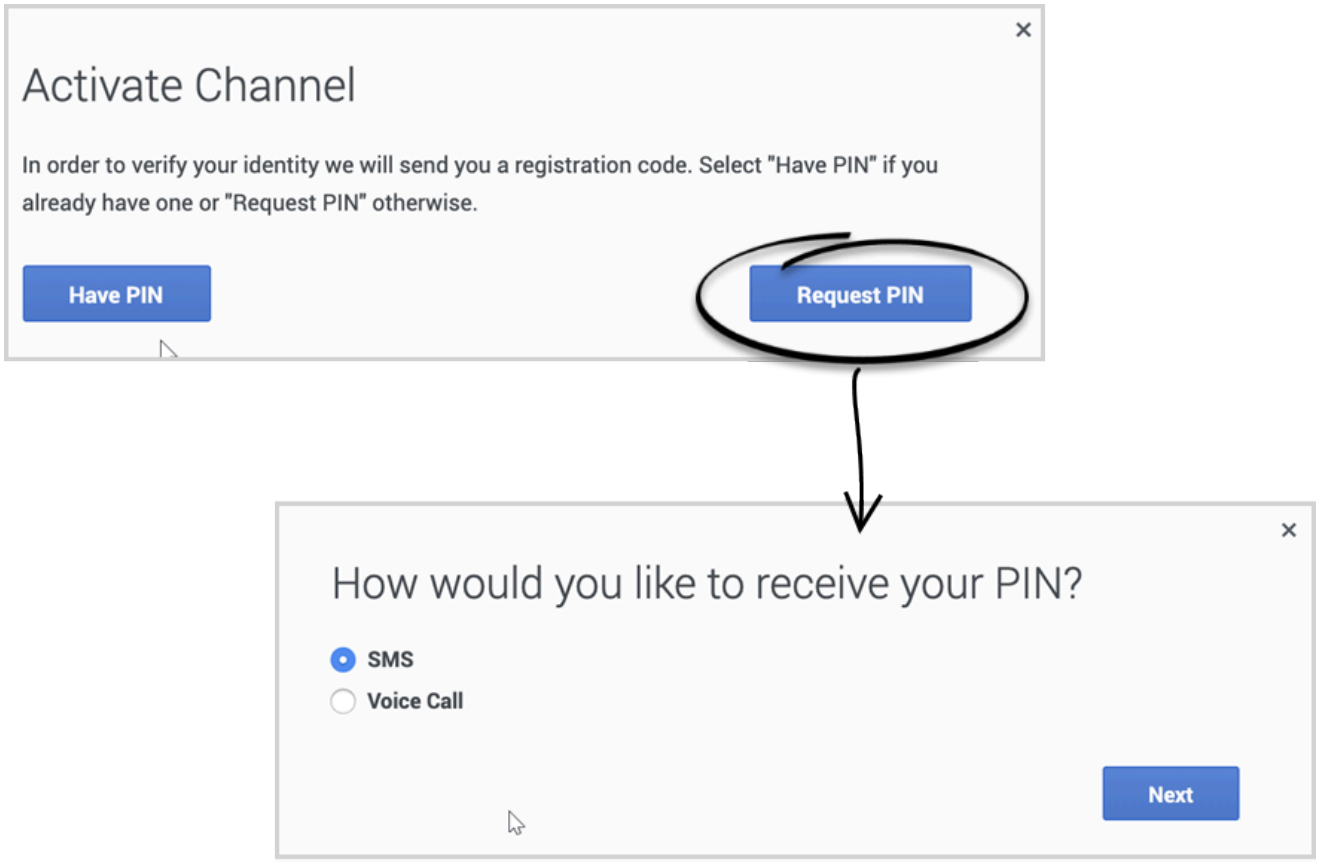
Register your channel by sending a request to Genesys. The Genesys team verifies, provisions and configures all the required infrastructure for your business contact number to function and service your consumers. **Note:** After registering, you must contact your Genesys representative if you want to change your business phone number or channel name.

Under **My Channels** in the left navigation menu, edit the WhatsApp service channel you created previously and click **Initiate** to send a review request to Genesys. Your channel is now in the INITIATED stage. You may be contacted by your Genesys representative to verify your WhatsApp channel, so make sure that all email addresses for the users registered in your account are valid. Once the initial validity has been checked, Genesys will set the channel state to VERIFIED.

After Genesys completes the full review, you'll receive an email indicating whether your request was approved. Your channel will now have one of the following states:

- REGISTERED - You can now activate the WhatsApp service channel.
- REJECTED - Contact your Genesys Representative for details about why your channel was rejected.

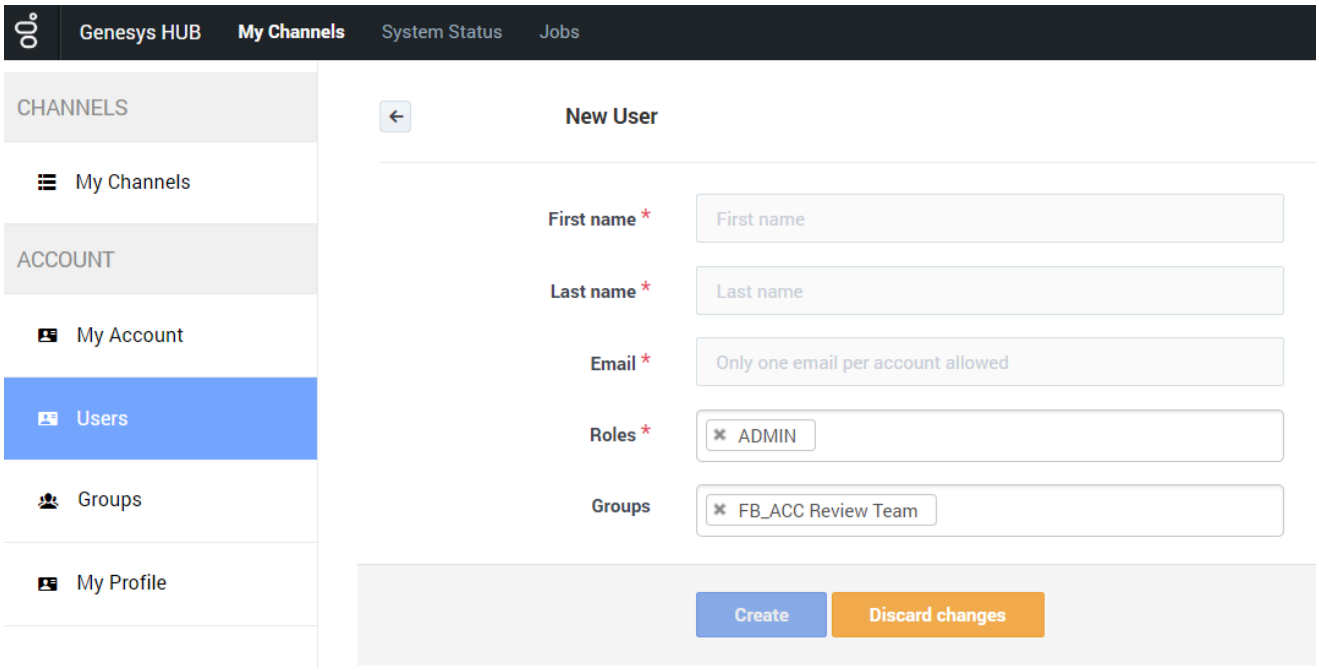
Activate a WhatsApp service channel



Activate your WhatsApp channel once you receive a registration confirmation email from Genesys.

Under **My Channels** in the left navigation menu, edit the WhatsApp service channel you created previously and click **Activate**. Now, click **Request PIN** and choose your contact method. Enter your PIN in the form and click **Activate**.

Create an administrator user for the account

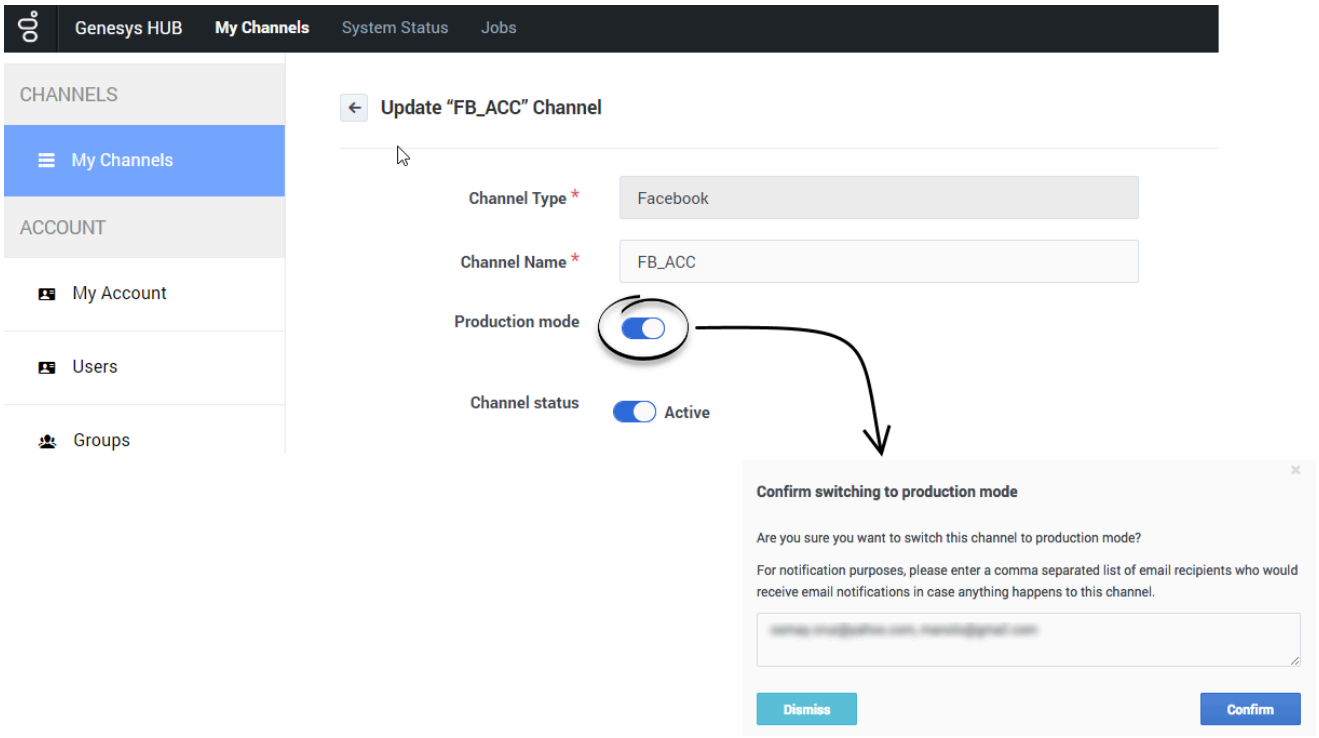


Create an administrator user account in Genesys Hub. Under **Users** in the left navigation menu, click **+Add User**.

- For **First Name**, use your company's name followed by a 1; for example, `Genesys1`. For **Last Name** field, type `admin`.
- In the **Email** field, type an email address using a format like the following: `Genesys1_Admin@GenSocialEngage.com`.
- Make sure to choose **Admin** for the role.

After you create this user, return to the **Users** menu. You should see a list of the users associated with the account that you have set up. Lastly, reset the password for the administrator user to finish the setup.

Production toggle



You can toggle between production and non-production (lab, staging, test) modes for using a switch on each channel.

Under **My Channels** in the left navigation menu, edit a channel and slide the toggle to the right to engage **Production mode**. On the confirmation window, add any email address that should receive notifications for the channel. This creates a distribution list in the Hub Status page.

Note the following:

- Internal Service Level Agreements (SLAs) apply only to accounts in production mode.
- Genesys carries out recovery, or assists with it, for production accounts only.
- Production mode is indicated in reports.
- Production mode requires all required licenses to be in place.

Set up SMS

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Set up SMS.

Genesys Digital Channels enables you to integrate Short Message Service (SMS) into Agent Desktop so that agents can exchange text messages with customers.

After you complete the steps on this page, you can treat SMS communications like any other Genesys interaction. For example, you can messages to the best-skilled agent using a Designer application. Reporting is also available through Genesys CX Insights.

1. Configure SMS number

Contact your Genesys representative to set up your SMS number in Genesys SMS Aggregator.

2. Enable Digital Shared Services

Work with your Genesys representative to enable service channels in Agent Setup to connect to the channels you created. Later when you create a Designer application to route SMS interactions, it will pull the data from these channels.

3. Configure Agent Desktop

You should go through the normal process to configure any necessary desktop options for the agents in your contact center, in particular Chat options. See the Agent Setup documentation for more information.

4. Create a Designer application

Create a Designer application to route SMS interactions to the right agents. **Note:** You should see the service channel you created in step 2 listed under **Manage Digital Endpoint** in Designer.

eServices Manager

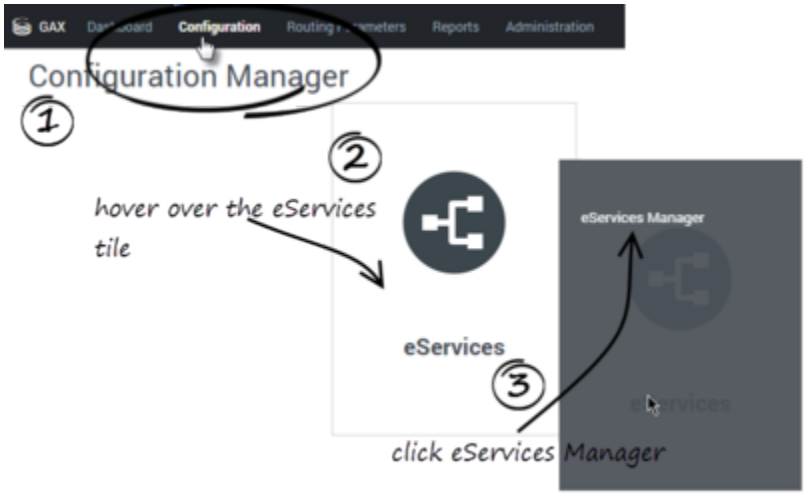
> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

The eServices Manager Plug-in gives you the tools to:

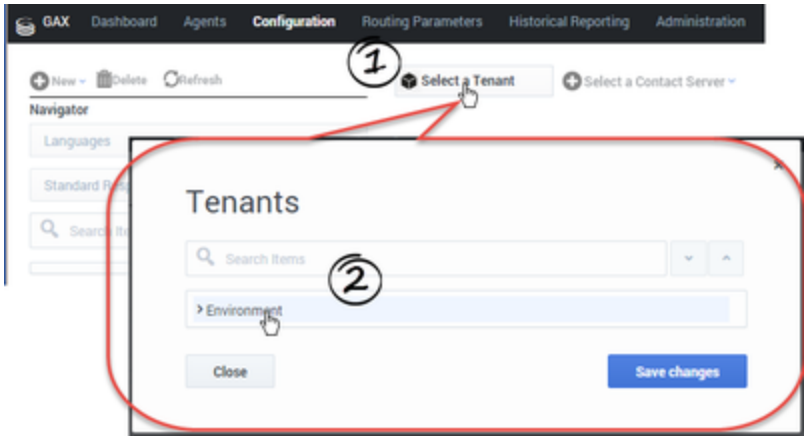
- Respond to incoming interactions using pre-written Standard Responses.
- Customize the Standard Responses using Field Codes to add a personal touch.
- Create Screening Rules in order to screen interactions for specific words or phrases, which you can then use to decide how to handle the interaction.

Let's get started

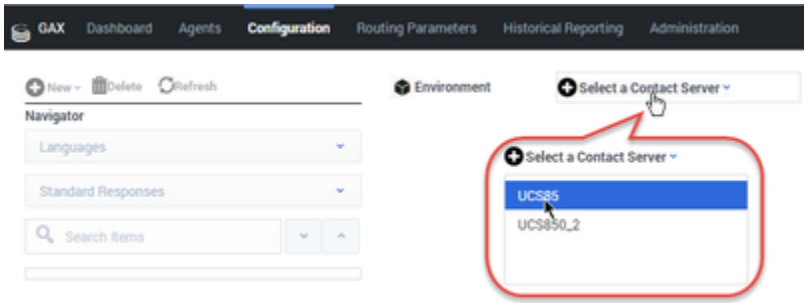
Start Platform Administration and open the eServices Manager Plug-in:



Now, select your Tenant...



and your Contact Server or Contact Server Proxy:



The Knowledge Manager window includes the following areas:

What next?

- Working with Standard Responses explains how to create and edit Standard Responses and how to create the category tree structure you will use to organize your Standard Responses, Field Codes, and Screening Rules.
- How to create and edit Field Codes, which enable you to insert personal information, such as names, into your Standard Responses.
- Working with Screening Rules explains how to create and edit Screening Rules, which enable you to analyse what incoming messages are about and to handle them accordingly.

Standard Responses

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

Standard Responses enable you to send welcoming, helpful answers to frequently-submitted queries from customers.

The high-level process

To create Standard Responses you do the following high-level steps:

1. Plan and build your Category Tree structure. A Category Tree provides the framework for organizing your Standard Responses.
2. Create your Standard Responses.
3. Create Field Codes and then add them to the Standard Responses. Field Codes enable you to personalize your Standard Responses.

You can cut, copy, paste, and delete Standard Responses and other Knowledge Management objects.

Plan and Build a Category Tree

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

A Category Tree consists of one or more *root categories*, each of which can have subcategories under it. Standard Responses are nodes under categories or subcategories in the Category Tree.

Step 1: Planning

We'll start by planning your Category Tree. To make your Standard Responses useful, you need root categories and subcategories that make sense in your business.

For example, you might create root categories for business units such as Sales, Service, and Billing. Then create useful subcategories. You might want to separate out customer type, such as Platinum, Gold, and Silver customers; or maybe it would work better for you to separate types of products, such as Pet Food, Pet Toys, and Cleaning Supplies; or you might want to respond differently to customers in different locations.

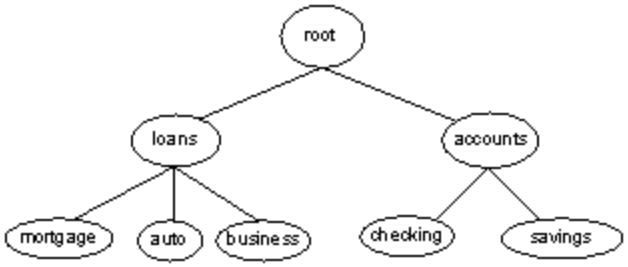
- Make sure that all the Standard Responses you need will fit within the categories you create.

> **Tip**
> Plan your category structure ahead of time. You can edit it, but a little thought now will save time a frustration later.

To read more about how category structures work, see

[+] How Category Structures Work

In general terms, a *category* is a unit of knowledge. Categories are organized in a tree structure; "Example Category Tree" shows an example.



Example Category Tree

Genesys eServices uses category trees to organize and provide access to the library of standard responses. Each standard response must be associated with one category. One category can have zero or many standard responses associated with it.
Categories with no associated standard responses may be of use in grouping other categories together.

Note these definitions:

- A *terminal* category is one that has no subcategories: a leaf on the category tree.
- A *nonterminal* category is one that has subcategories.
- *Child* is another term for subcategory. For example, in "Example Category Tree", **savings** is a child of **accounts**, and accounts has the two children **checking** and **savings**.

A category tree is specific to a tenant and a language. Each tenant/language pair can have multiple category trees.

You can design different sets of screening rules (for example) for different languages within a single tenant. But the screening rules operate the same way regardless of which language they are grouped under.

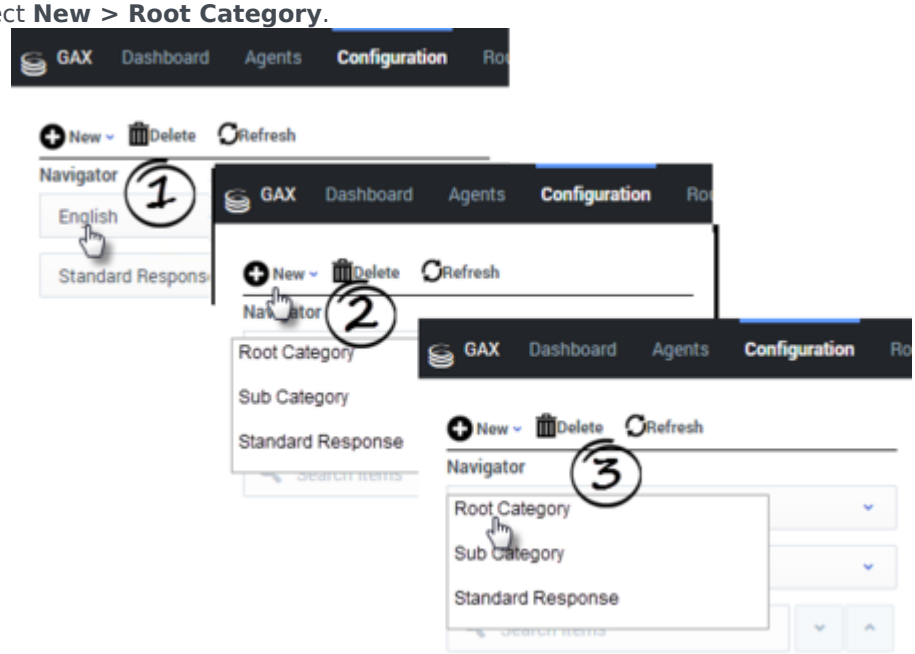
Category membership is inherited. That is, if Category 1 includes Categories 10 and 11, and Category 10 includes Categories 100 and 101, then Category 1 also includes Categories 100 and 101.

Step 2: Build your Category Tree

You've done your planning. Now let's make it happen. For our example, we are going to create two new root categories, Cats and Dogs, with subcategories, and sub-subcategories:

First we'll create our two root categories:

1. Make sure you have selected the right Tenant and Contact Server.
2. Select the language. Note that in this release, the language choice does not affect how messages are handled.
3. Select **New > Root Category**.



4. Then enter the root category name, `Cat`, and click **Save.** Repeat the same process to create the Dog root category.

5. Now create the subcategories. For each subcategory, highlight the correct root category and then follow the same process as above, but select **New > Sub Category** instead of **New > Root Category**.

> **Tip**
> You can cut, copy, paste, and delete categories as well as other Knowledge Management objects.

More About Categories

- To change a category name, select it, edit the name in the text box, and click **Save**.
- To delete a category, check its checkbox and then click **Delete**.
  - Be careful to select only the category check boxes you want to delete!
  - Deleting an upper-level category also deletes all the categories under it.
- There is a special type of category called Service Category that the system may create for you.
- To open the root category and show the categories beneath it, click a caret mark (>).

Characters Allowed in Names

Knowledge Manager Objects

Names of categories, like those of all Knowledge Manager objects, can consist only of the alphanumeric characters supported in UTF-8, plus the characters shown in "Additional Characters Allowed in Object Names".

Additional Characters Allowed in Object Names

Name	Character	Name	Character
Hyphen	-	Exclamation point	!
Number sign, pound	#	Dollar sign	$
Caret	^	Asterisk	*
Underscore	_	Curly brackets	{ }
Angle brackets	< >	Period, full stop	.

Names can be no more than 64 characters long.

Language Names

Names of the Business Attribute called **Language** can consist only of Latin characters and numbers (A–Z, a–z, 0–9), plus the characters in "Additional Characters Allowed in Object Names."

Custom Variables

Names of custom variables can consist only of Latin characters and numbers (A–Z, a–z, 0–9).

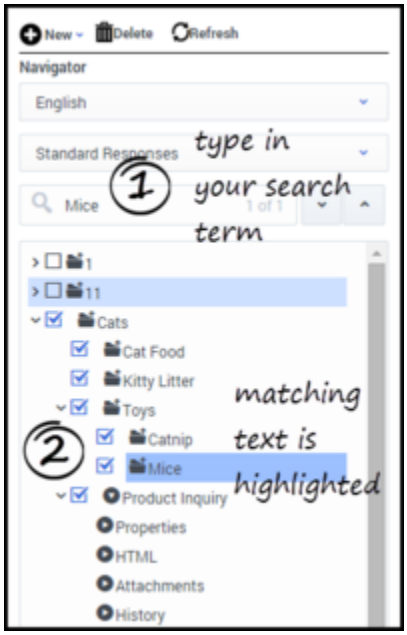
Text Direction

The default direction for text is left to right. To change the direction, right-click the text field and select `writing direction`.

Search the Category Tree

To locate a category, Standard Response, Field Code, or Screening Rule:

- Select the type of object you want to search for and then enter your search term.



Next Steps

- Creating Standard Responses
- Personalizing Standard Responses with Field Codes

Create a Standard Response

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

A Standard Response is an item in the Standard Response Library, which stores prewritten responses for use as suggestions to agents, acknowledgments, and/or autoresponses. Each standard response is assigned to exactly one category in the system; however, a category may have zero or many standard responses assigned to it.

You can use Standard Responses for any of the eServices channels: eMail, Chat, Social Engagement, or SMS.

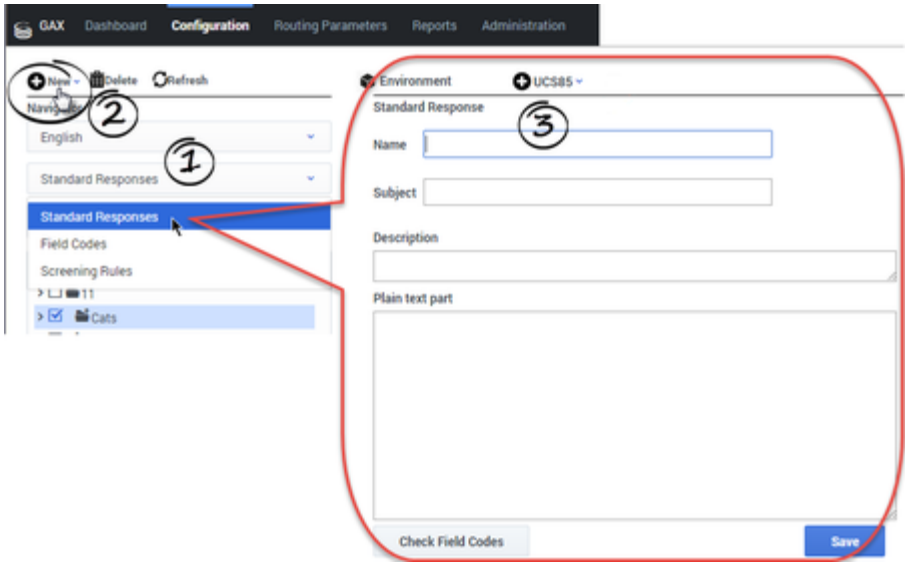
It takes just a few steps to create a Standard Response. The Process Overview below lists the main steps. We'll take each one in sequence, or you can skip to the information you need right now.

Process Overview

- Create a Standard Response
- Configure the properties for your Standard Response
- Enter the HTML version
- Add attachments
- Create additional versions and retrieve prior versions
- Create and add Field Codes

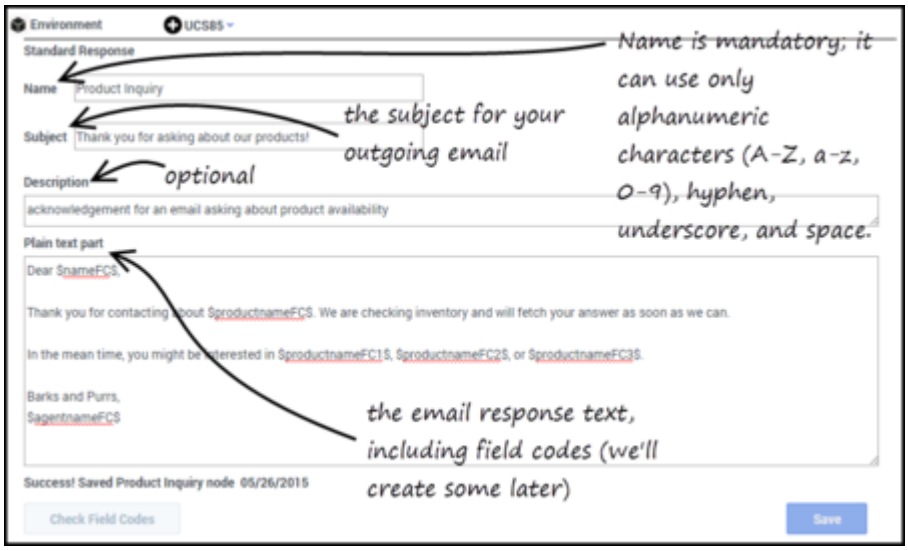
To create a Standard Response:

1. Select the correct Category Tree node, and then select **New > Standard Response**. The first Standard Response configuration page appears on the configuration pane.

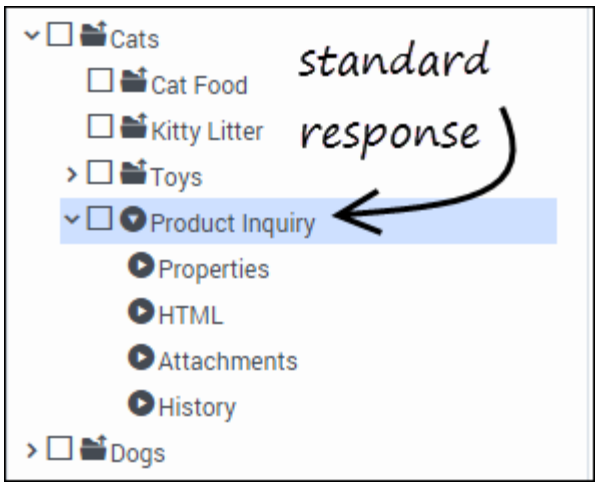


2. Enter the basic information for your Standard Response on the configuration pane.

> **Important**
>
> The Standard Response name can consist only of alphanumeric characters (a-z, A-Z, 0-9), hyphen, underscore, and space. For more information about characters you can use in Knowledge Manager objects, see Characters Allowed in Names.



3. Click **Save** and the new Standard Response appears in the Category Tree.



4. To edit the standard response, select it, make your changes, and then click **Save**.

Field Codes

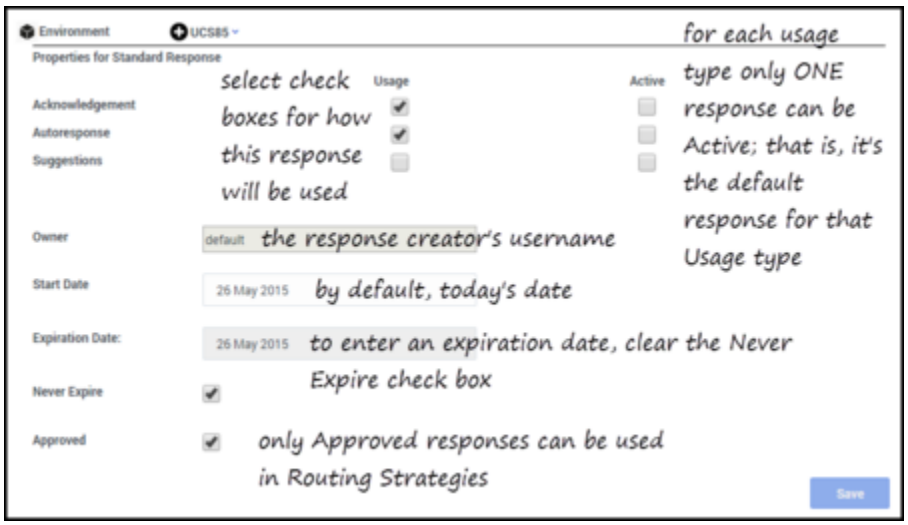
You will probably want to use Field Codes to personalize your Standard Responses. We'll be creating Field Codes after creating some Standard Responses, and then insert the Field Codes into them.

Set the Standard Response Properties

> **Important**
>
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

To configure the properties of your Standard Response:

- Click **Properties** under your Standard Response in the Category Tree to open the Properties configuration page.



The possible Usages are:

- Acknowledgment—The standard response may be sent to acknowledge receipt of an incoming interaction.
- Autoresponse—The standard response may be used as an automatic response to an incoming interaction.
- Suggestions—The standard response may be offered to agents as suggested wording to use in their own replies to interactions.

Each category in the Category Tree may have multiple standard responses of each Usage type.

About the Active Parameter

For each Usage type, you must specify whether this standard response is the Active one. Only one standard response of a given Usage type can be Active. When the system needs to send a Standard Response of a specific Usage type automatically, it sends the one marked Active.

If you attempt to select Active for a Standard Response (either a new one or an existing one), and there is already an Active Standard Response with that usage type for that category, Knowledge Manager offers to take the previously Active Standard Response out of Active status.

About the Expiration Date

If a standard response's expiration date has been reached, it has the following effects:

- The standard response is not shown in IRD, so it cannot be used in a new or modified strategy.
- If this standard response was saved in a strategy before the expiration date was reached, E-mail Server does not send the standard response, but returns an error message.

Next Steps

- Create an HTML version of the standard response.
- Add an attachment to the standard response.
- Maintain multiple versions of the standard response.
- Create Field Codes to use in your standard responses.

Create the HTML Version

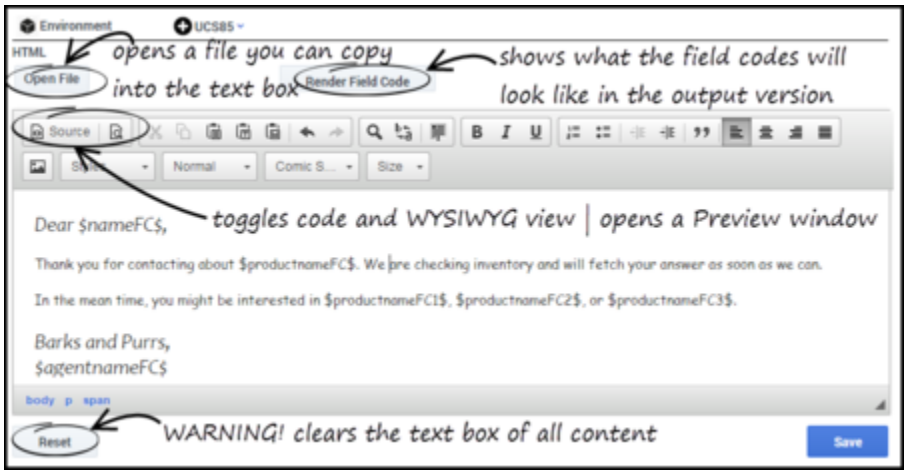
> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

To create the HTML version of your Standard Response:

- Click **HTML** under your Standard Response in the Category Tree to open the HTML configuration page. The plain text you entered appears automatically in the text box.

Most of the buttons provide commonly-used editing functionality, including the option to insert a link to an image. If you aren't sure what a button does, hover over it to open a tooltip.

The figure below explains buttons with more specialized functionality.



Paste an Already-Configured Response

You have a number of options for pasting content in from another document, including pasting:

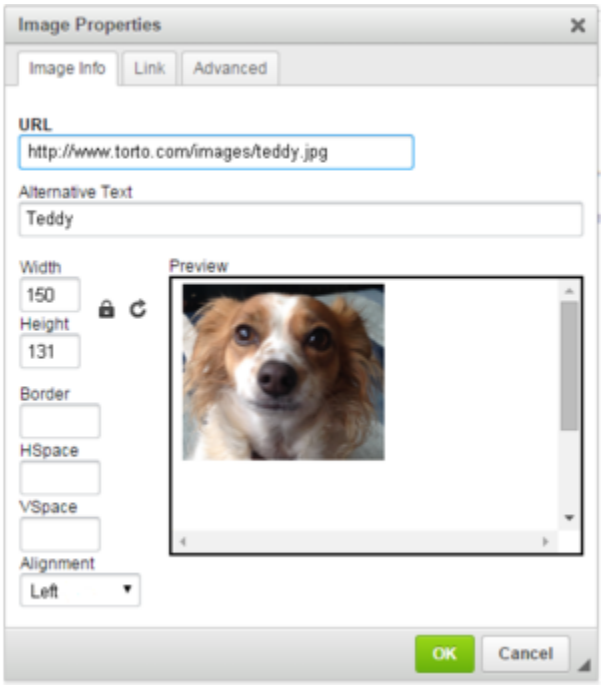
- HTML code into the code view.
- Any text in as plain text.

- Formatted content from Microsoft Word.

> **Important**
>
> - Links to resources that are used in the content, such as image files, must use absolute URLs; embedded graphics or relative links are not supported. Also, these resources must be available on the web through an HTTP server at the time that the standard response containing this HTML is sent to the customer.
> - Make sure that the content of the HTML version, including field codes, matches the plain text version that you created on the main configuration page tab. This is important because e-mail clients may display multipart e-mails in varying ways. For example, if Microsoft Outlook has AutoPreview turned on, the preview may show the plain text version whereas the full display shows the HTML version. For this reason you should be careful that the plain text and HTML versions have identical content.

Add an Image

To add an image, click the  button, then configure the dialog box that opens:



Next Steps

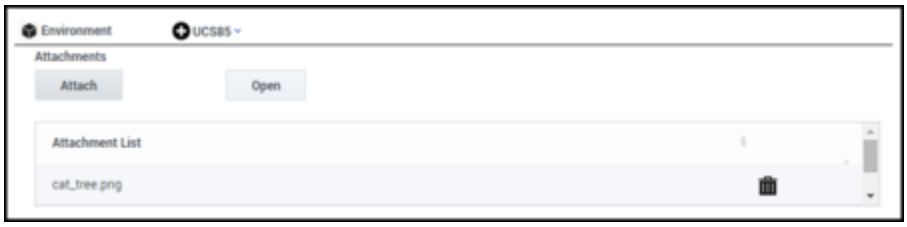
- Add an attachment
- Review the history and manage versions
- Create Field Codes

Add Attachments

> **Important**
>
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

To add an attachment to your Standard Response:

Click **Attachments** under your Standard Response in the Category Tree to open the **Attachments** configuration page.



- To add an attachment, click **Attach**.
- To remove an attachment, select it, then click the trash can icon.
- To view an attachment, select it, then click **Open**. This downloads the file to your environment.

Next Steps

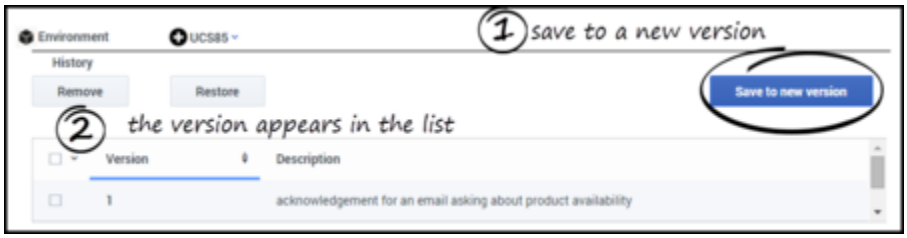
- Review the history and manage versions
- Create Field Codes

Create and Manage Multiple Versions

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

To create multiple versions of your Standard Response:

Click **History** under your Standard Response in the Category Tree to open the History configuration page.



- When you save a new version, it retains everything except for attachments.
- After you save multiple versions, use the **Remove** and **Restore** buttons to manage them.
- To choose the version that appears on the Standard Response pane, select the check box for the version you want and click **Restore**.

Field Codes

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

Although Field Codes are used mostly in standard responses, they are the most complex and powerful aspect of standard responses, so they are described in this separate section.

Knowledge Manager enables you to create a wide range of Field Codes types, from simple Field Codes that function similarly to a Mail Merge-type word processor feature to complex Field Codes that include multiple objects, formulas, and constants (see Using a Complex Field Code for an example).

Once you create a Field Code, you can use it in multiple standard responses.

The interface for creating Field Codes is simple; creating really useful Field Codes requires a deeper understanding of how Field Codes can be constructed. How to Create and Insert Field Codes provides step-by-step instructions, with links to detailed reference information at the relevant places.

The reference information consists of the following topics:

- Field Code Variables
- Using Formulas in Field Codes
- Field Code Examples

> **Tip**
> You can cut, copy, paste, and delete Field Codes as well as other Knowledge Management objects.

Field Codes Overview

The main use of field codes is to particularize standard responses.

For example, you can use the field code `<$Contact.FirstName$>` in a response beginning `Dear <$Contact.FirstName$>`, which you send to dozens of recipients. In each message, `<$Contact.FirstName$>` is replaced by the first name of the addressee of the message (the contact) as listed in the Universal Contact Server database.

More generally, a "field code" is a formula that you insert into an outgoing text object, such as an e-mail that E-mail Server generates when triggered to do so by a routing strategy object.

The most common type of such text object is a standard response (triggered by an Autoresponse or Acknowledgement object), but you can also insert field codes into other types, such as chat transcripts, SMS messages, and forwarded or redirected e-mails. In some cases, the only place you can insert a field code is in the Subject line using the **Format** tab in a strategy object.

The following is a complete list of the strategy objects that can use field codes either in a standard response or in the Subject line:

- Acknowledgement
- Autoresponse
- ChatTranscript
- Create EmailOut
- Create Notification
- Create SMS
- Forward
- RenderMessageContent

The following is a complete list of the strategy objects that can use field codes only in the Subject line:

- Redirect
- Reply from External Resource
- Send

When a text object containing such a formula is processed, the following happens:

1. The formula performs an operation, which produces a result.
2. The result replaces the field code in the text object.

This process of performing an operation and substituting its result is called "rendering."

> **Important**
> Field codes can be used in outgoing text objects only.

A complete reference list of field codes is available in the Genesys eServices Field Codes Reference Manual.

How to Create and Insert Field Codes

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

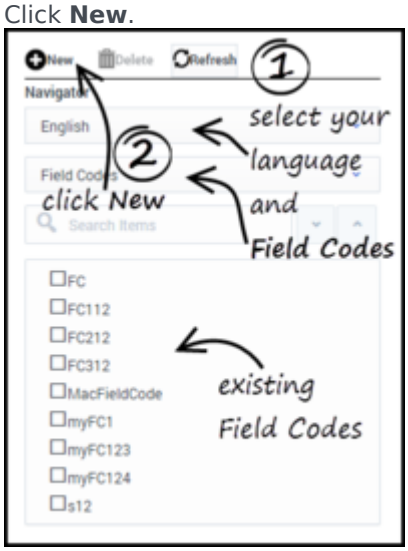
Using Field Codes includes:

- Creating Field Codes
- Inserting Field Codes into a Standard Response

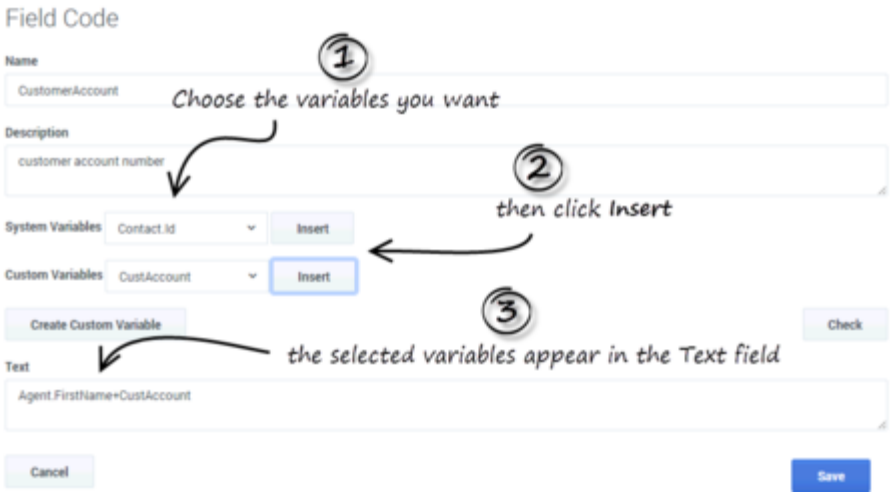
Creating Field Codes

From the Knowledge Manager browser window:

1. Select the language and **Field Codes** from the drop-down menus.
2. Click **New**.



3. To configure your new Field Code, enter the properties you need in the Field Code configuration pane and then click **Save**.



The new Field Code appears in the Category Tree pane. You can edit a Field Code anytime by double-clicking it in the Category Tree.

4. Enter a name and (optionally) a description for the field code.
5. Select System and/or Custom variables from the drop-down lists.
6. Click **Insert** to populate the text box with the selected variables.
   See a detailed discussion of variables in Knowledge Manager.

   If the custom variable you need is not in the list, you can **[+] create a custom variable.**

   a. Click **Create Custom Variable**. The Custom Variable dialog box appears.
   b. Enter a name (required) and description (optional).

   > **Important**
   > The name must consist only of alphanumeric characters or underscores.

   c. Select **String** or **Integer** for the type.
   d. Enter a default value. This is mandatory.
   e. Click **Add**. The new custom variable appears in the list.

   > **Important**
   > You can also use this dialog box to edit and delete (Remove) existing custom variables.

   f. Click **OK**.
7. Enter any other desired text in the Text field. This text must conform to the rules described in Using Formulas in Field Codes.

8. Click **Check** to verify that the field code is well-formed (that is, that it has no typographical errors, missing parentheses, and so on).

For detailed help constructing Field Codes, see Using Formulas in Field Codes. For examples showing the use of a custom variable and of a complex Field Code, see Field Code Examples.

Inserting Field Codes into a Standard Response

To insert Field Codes:

1. Select a Standard Response from the Category Tree or else create a new Standard Response.

> **Important**
> Standard Responses that are intended for use in FAQ objects should not contain field codes.

You can insert Field codes in two ways:

- Plain text version:
  1. Click in the plain text area. A table of field codes displays to the right (if you click elsewhere the table disappears).
  2. Drag and drop the field code from the table into the standard response.
- HTML version:
  1. Move the cursor to the right to make the field code table display.
  2. Double-click a field code to insert it in the standard response.

1. Click **Insert Field Code** to display a list of all the available field codes.
2. Select a field code and click **OK** to insert it, together with its required delimiters (<$ $>), into the Standard Response.
3. Click **Check Field Codes** to see the standard response with the Field Codes rendered, showing the default values for each Field Code.

Field Code Variables

Using Field Code Variables includes:

- Using UCS Data as System Variables in Standard Responses
- Custom Variables
- Using Your Own Data in Standard Responses

Using UCS Data as System Variables in Standard Responses

In the example given in Field Codes Overview, the `Contact.FirstName` retrieves a piece of data about the interaction. The ability to access interaction data is perhaps the most frequent use of field codes. Although field code formulas can be very complicated, many simply retrieve a single piece of data, such as a contact's name.

You access Universal Contact Server data using predefined variables, called "system variables."

These variables access three predefined objects. Each object has a name and a set of properties. In the example, `Contact` is an object and `FirstName` is one of its properties. The system variable `Contact.FirstName` retrieves the value of the `FirstName` property of the `Contact` object.

In similar fashion, there is a system variable for each `object+property` pair. The objects and properties that you can use in field code formulas are described in the following sections.

Interaction

This object represents the particular interaction being worked on, such as an inbound e-mail. These are its properties:

- `Id`
- `DateCreated`
- `Subject`
- `ToAddress`
- `FromAddress`
- `AttachedData`
- `TimeZone`

Contact

This object represents the contact associated with the interaction being worked on. These are its properties:

- Id
- Title
- FirstName
- LastName
- FullName
- PrimaryPhoneNumber
- PrimaryEmailAddress

Agent

This object represents the agent working on the interaction. These are its properties:

- FirstName
- LastName
- FullName
- Signature

> **Important**
> Automated responses use the default agent. Create the default agent as a Person object just like any other in Configuration Manager. Then select this Person on the Automated Reply Agent screen of the E-mail Server configuration wizard (or set this Person as the value of the autobot-agent-login-name option in the E-Mail Processing section of the E-mail Server Java application). Since this is the Person who the automated response appears to be from, you may want to name it after your company or institution.

Custom Variables

In addition to the system variables, you can use Knowledge Manager to create custom variables. Custom variables have the following properties:

- Their values are assigned by strategy objects.
- Therefore, standard responses that use field codes containing custom variables must have the usage type Autoresponse or Acknowledgment.

For an example of the use of a custom variable in a standard response, see Using a Custom Variable.

> **Important**
>
> The names of custom variables must begin with an alphabetic character or underscore, and the remainder of the name must consist only of alphanumeric characters or underscores. This differs from the requirements for the names of other Knowledge Manager objects, which may also contain hyphen and space. For example, 5-usercode is not an acceptable name for a custom variable, but it is acceptable as the name of a screening rule or category.

Using Your Own Data in Standard Responses

It is possible to incorporate data that you keep external to Universal Contact Server into your standard responses (including automated responses). This data could include case numbers, account information, and so on. Remember that attached data always consists of key-value pairs.

Incorporating external data into standard responses is a two-step process:

1. Retrieve the external information and add it to the interaction as attached data. One place to do this is in a routing strategy.
2. Now that you have attached the data to the interaction, you can use the `AttachedData` property of the Interaction object to access the data and incorporate it into your standard response. The `AttachedData` property requires one argument, which is the key name. The result of the following formula is the value associated with the `OrderStatus` attached-data key:

```
<$Interaction.AttachedData("OrderStatus")$>
```

Using Formulas in Field Codes

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

In addition to system variables such as `Contact.FirstName`, field codes may contain formulas. This section provides an outline of formula usage. Details on many of these topics are provided in the Genesys eServices Field Codes Reference Manual.

You must always delimit field codes by using `<$ … $>`. If you type a field code directly into the body of a standard response, then you must enter the delimiters yourself. If you select from the list of field codes in Knowledge Manager, then the delimiters are added automatically.

The text that appears inside the delimiters is a formula. Field code formulas are very similar to formulas in other applications, such as Microsoft Excel.

A *formula* is a sequence of one or more operands (such as numbers and text strings), separated by operators (such as + and -).

For example, in the following formula, 2 and 3 are operands and + is an operator:

```
<$2 + 3$>
```

Operands can be values that do not change (constants), or values that vary based on the context. In the previous formula, all the operands are constants, so the formula always evaluates to `5`. The next formula, on the other hand, evaluates to a different value for each agent who uses it:

```
<$Agent.Signature$>
```

Field Code Syntax

To summarize field code syntax:

- A field code must be delimited by `<$ ... $>`.
- Alphabetic strings, whether constants in formulas or elsewhere in a field code, must be enclosed in double quotes.
- Numeric constants require no special treatment.
- You must use special characters for some purposes. For example, for your field code to render with a line break, you cannot simply type a carriage return. Instead, you must insert the expression `\n`. A list of these special characters is available.

HTML in Field Codes

With special configuration, field codes can contain HTML markup; for example, you could have a field code <$my.agent.signature$> defined as

```
Sam Agent<BR />
Acme Products<BR />
29 Exterior Blvd<BR />
Springfield, CX 09090<BR />
```

To enable this, you must use the Java property `-Dsrl-field-code-allow-html=true`, in one of the following ways:

- Add it to the `JavaArgs` section of `ContactServerDriver.ini`
- Add it as an argument to the startup command line in `contactServer.sh`.

Operator Precedence

If you use more than one operator in a formula, the order in which they are evaluated depends on their relative *precedence* (higher precedence operators are evaluated first). For example, multiplication (`*`) has a higher precedence than addition (`+`), so that the formula below evaluates to `14`, not `20`:

```
<$2 + 3 * 4$>
```

You can use parentheses to override the default precedence. The formula below evaluates to `20`:

```
<$(2 + 3) * 4$>
```

For a complete list of operators and their relative precedence, see "Operator Precedence” in the Genesys eServices Field Codes Reference Manual.

Data Types

Operands of several different types may appear in formulas:

- Number
- String (text)
- Date/time
- Boolean (true/false)
- Object (Contact, Interaction, and Agent)

Each data type behaves differently in formulas, and the operators have different meanings when you use them with different data types. For example, the + operator means “add” when used with numbers, but “concatenate” (paste together) when used with strings. This formula evaluates to *Uncle Sam Wants You*

```
<$"Uncle Sam " + "Wants You"$>
```

In addition, some operators cannot be used with some data types at all. For example, you cannot use the multiplication (*) operator on two strings.
All formulas, regardless of their final data type, are converted to strings before being merged into your standard response. This conversion follows a set of default rules that depend on the data type. For example, the default rules for numbers round them off to integers. This formula causes 2 to be inserted into your standard response, even though the real result is `2.25`:

```
<$9 / 4$>
```

You can use the `Text` function (see below) or `format operator`:) to override the default formatting. Either of the following formulas inserts `2.25` into your standard response:

```
<$Text(9 / 4, "#.##")$>
```

```
<$(9 / 4):"#.##"$>
```

For a detailed list of data types and how you can use them, see "Data Types" in the Genesys eServices Field Codes Reference Manual..

Functions

When composing formulas, you can use many built-in functions. *Functions* are predefined formulas that perform calculations using values, called *arguments,* which you supply. To use a function, write its name, followed by an opening parenthesis, the arguments for the function separated by commas, and a closing parenthesis.

Function arguments may be of any data type, although individual functions may place restrictions on their arguments. Function arguments may be constants or formulas. The `Length` function, for example, takes a single string argument and returns its length in characters. This formula evaluates to `13`:

```
<$Length("Hello, world!")$>
```

As another example, the `Date` function takes individual date components (year, month, day, and so on), and constructs a date/time value. The formula below evaluates to `1965-11-23 09:03:10`:

```
<$Date(1965, 11, 23, 9, 3, 10)$>
```

Functions may act as arguments to other functions. The `WeekdayName` function takes a single date/time argument and returns the day of the week as a string. The formula below evaluates to `Tuesday`:

```
<$WeekdayName(Date(1965, 11, 23, 9, 3, 10))$>
```

This formula evaluates to `7`:

```
<$Length(WeekdayName(Date(1965, 11, 23, 9, 3, 10))$>)
```

For detailed descriptions of all available functions, see "Functions" in the Genesys eServices Field Codes Reference Manual.

Using Objects

All object/property pairs are also available in the `Variables` drop-down menu in the Knowledge Manager `Field Code Editor`.
Object properties can be of any data type. `Agent.FullName`, for example, is a string, but `Interaction.DateCreated` is a date/time.
The data type of an object property can even be another object. For example, `Contact.EmailAddresses` yields another object called a `ContactEmailAddressList`. In cases such as this, you can access the properties of the resulting object by entering a period (.), followed by the property name, just as before. For example, the formula below evaluates to the number of e-mail addresses assigned to the contact:

```
<$Contact.EmailAddresses.Count$>
```

Some object properties require arguments just as functions do. For these properties, write the arguments, enclosed in parentheses after the property name, just as before.

For example, the ContactEmailAddressList object has a property named `Exists`, which you can use to test whether a particular e-mail address is assigned to a contact. The data type of this property is Boolean (true/false), and it takes one argument, the e-mail address to test. For example:

```
<$Contact.EmailAddresses.Exists("samd@acme.com")$>
```

For detailed descriptions of all objects and their properties, see "Objects" in the Genesys eServices Field Codes Reference Manual.

Field Code Examples

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

This section presents examples of the use of field codes.

Using a Custom Variable

Purpose: This is a simple example of the use of a custom variable in a standard response.

Prerequisites
This example assumes a category tree that includes categories called `Cookbooks`, `Mysteries`, and `Reference`.

1. In Knowledge Manager:
   a. Create a custom variable called QueryTopic (see See Creating field codes).
   b. Create a field code called `Query_Topic` that consists of the variable `QueryTopic`.
   c. Create a standard response of type `Autoresponse` called `AUTO` that includes the sentence `Thank you for your inquiry about <$ Query_Topic $>`.
   d. Create the following screening rules:
   e. `Cookbook: RegExFind("cook") || RegExFind("recipe") || RegExFind("food") || RegExFind("cuisine")`
   f. `Mystery: RegExFind("murder") || RegExFind("crime") || RegExFind("case of the") || RegExFind("detective")`
   g. `Reference: RegExFind("dictionar") || RegExFind("encyclopedia") || RegExFind("almanac")`

8. In Interaction Routing Designer, create a strategy that applies these screening rules one after the other, assigning a different value to the custom variable for each screening rule:
   a. Create a variable called var_screen.
   b. Create a strategy. Start the strategy with a `Screen` object. On the `General` tab of the `Screen` object, select the `Cookbook` rule.
   c. On the `Result` tab, click `Assign values of the key-value pairs`. Then under `Output values` select `var_screen` for `Variable` and enter `ScreenRuleMatch` for `Key from output`.

d. Connect the Screen object to a Generic Segmentation object. Create two segments: var_screen = true and var_screen = false.

e. Connect an Autoresponse object to the top green port (the one corresponding to true) of the Segmentation object. In the Autoresponse, select the Select standard response radio button and select AUTO in the associated drop-down list.

f. Still in this Autoresponse, go to the General tab and in the Field Codes area (bottom of the tab) click the New icon, enter QueryTopic under Key, and enter cookbooks under Value. This will generate an e-mail that includes the sentence *Thank you for your inquiry about cookbooks.*

g. Return to the Generic Segmentation and connect a new Screen object to its second green port (the one corresponding to false).

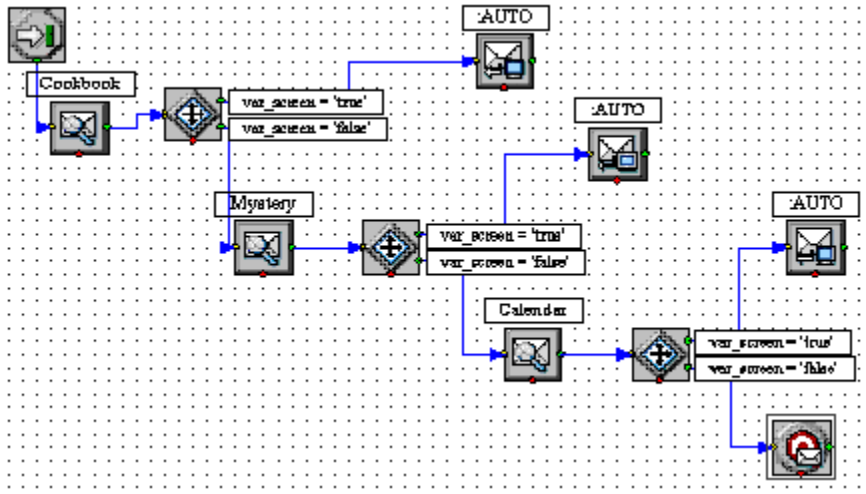
h. On the General tab of the new Screen object, select the Mystery rule. On the Result tab, click Assign values of the key-value pairs. Then under Output values select var_screen for Variable and enter ScreenRuleMatch for Key from output .

i. Proceed as in Steps d–f: Connect this Screen object to a new Generic Segmentation, again with segments for var_screen = true and var_screen = false.

j. As in Step e, connect a new Autoresponse object to the green port for true, select the AUTO standard response, and enter QueryTopic under Key. This time enter mysteries under Value.

k. Return to the second Generic Segmentation's green port for false and repeat Steps g–j, creating a third Screen object and Generic Segmentation. In the Screen object, select the Reference rule; in the Segmentation object, set the custom variable to reference .

The figure "Strategy Using Custom Variable" shows the strategy as described. The single standard response AUTO generates three e-mails, each with a different word filling the blank in *Thank you for your inquiry about ___.*



Using a Complex Field Code

The following is an example of a complex field code:

```
<$ If (Time() - Interaction.DateCreated > 14, “Please accept our apologies for not having replied sooner. ”, “”) $>
```

This field code inserts a tardiness apology if more than 14 days have elapsed since the interaction first entered the system. It uses the function `If`, which has these properties:

- Its syntax is `If (Boolean, TrueResult, FalseResult)`
- If `Boolean` evaluates to `True`, it returns the second argument.
- If `Boolean` evaluates to `False`, it returns the third argument.

In this example, the three arguments of `If` are as follows:

1. `Time() - Interaction.DateCreated > 14` A formula that returns `True` if the difference between the date created and the current system time is more than 14 days. (The result of a mathematical operation on dates is given in days.)
2. `“Please accept our apologies for not having replied sooner. ”` A text string apologizing for tardiness, inserted if the formula evaluates to `True`.
3. The null string: if the reply is not late (the formula evaluates to `False`), nothing is inserted in it.

Next Step

- Go on to create and manage Screening Rules.

Screening Rules

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

Screening rules scan an interaction and try to match either a destination address (who the message is going to, whether that is identified by an email address, a cell phone number, or some other parameter), a regular expression, or both. Screening is performed by Classification Server when it is triggered by a Screen Interaction object in a routing strategy.

A screening rule can optionally be associated with a category.

> **Important**
> Screening can operate on any interaction that has text somehow associated with it, whether as the body of the interaction (e-mail, chat), or otherwise (as user data, for example). In practice, it is expected that most interactions which are screened will be e-mail messages; therefore, the terms *e-mail* and *message* are used interchangeably here, to refer to these interactions. In fact, whatever is said here about e-mail applies to any interaction that has associated text.

Screening Rules topics include:

- How to Create and Test Screening Rules
- Screening Rules Reference—This section includes:
  - What they check: What Screening Rules Check
  - The functions, arguments, and operators used in them: Functions and Arguments
  - Regular expressions to use in them: Regular Expressions
- Examples of Screening Rules

You can cut, copy, paste, and delete Screening Rules as well as other Knowledge Management objects.

How to Create and Test Screening Rules

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

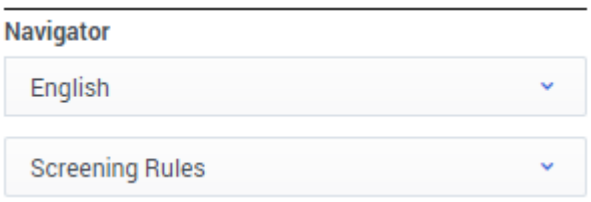
The step by step procedure for creating a Screening Rule is quite simple. The power of Screening Rules lies in the many ways you can configure them. We'll start with the high-level procedure, then drill down as needed into more detailed explanations of the various parameters.

This page contains the following sets of instructions:

The High Level Steps

Note that the order of these steps matters!

1. Start on the Knowledge Manager browser window. Select the correct:
   - Tenant
   - UCS Server
   - Language
2. On the **Navigator** drop-down menu, select **Screening Rules**.

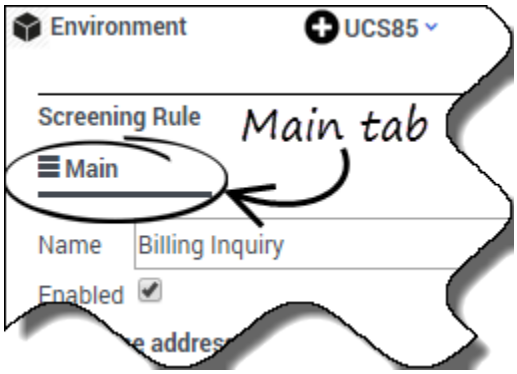


3. On the Category Tree, check the boxes for the categories this Screening Rule should apply to.
4. Click **New**.

5. Select **Screening Rule**.
6. After configuring the Screening Rule—which will be explained shortly—click **Save**. The new Screening Rule appears in the Category Tree under the selected Root Category.
7. To edit a Screening Rule, highlight it in the Category Tree and then make the desired changes. Don't forget to click **Save** when you are done.

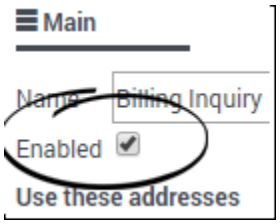
Configuring a Screening Rule

Configuring a Screening Rule is mostly done on the Main tab.



When you name your Rule, remember that you can only use permitted characters in object names.

Check the **Enabled** check box to make this Rule available when you add a Screen object to a routing strategy.



Once a strategy includes a Screen object that uses a particular Screening Rule, the strategy will continue to use the Rule whether it is disabled or enabled.

Do you need to use the Order text box?

if the Screening Rule is used in a Multiscreen routing object in which **All rules** is selected, use the **Order** text box to specify where in the sequence of rules this particular one should be applied.

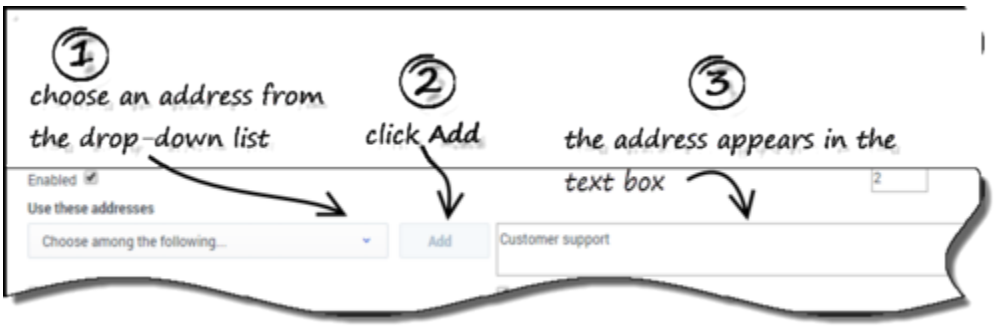
In other cases, the **Order** text box is unused.

Do you need to screen the addresses the email is sent to?

- If so, in the **Use these addresses** area, select an address from the left-hand window, then click **Add** to copy it to the right-hand window. You can add as many addresses as you need. If the address you need isn't there yet, you can
  - Directly type an address in the right-hand window.
  - Define additional addresses in the **EmailAccounts** configuration database object:

    GAX: **Configuration > Routing/eServices > Business Attribute Values > Business Attributes > EmailAccounts > Attribute Values**

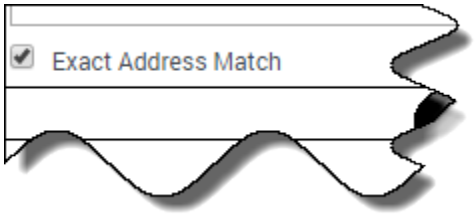
    Config Manager: **Tenant > Business Attributes > EmailAccounts > Attribute Values**



- If you don't need to screen addresses, leave this section empty.

Does it matter if the address is exact?

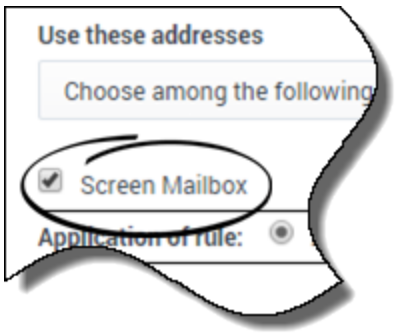
- If so, select the **Exact address match** box. The screening rule looks for messages having that exact address as a destination. For example, **xyz@domainname.com** matches **xyz@domainname.com** but not **abc.xyz@domainname.com.** This match is not case sensitive.



- If not, clear the **Exact address match** box. Then the rule looks for messages having that address as a substring of their destination address. For example, **xyz@domainname.com** matches **abc.xyz@domainname.com** and **xyz@domainname.com.**

Do you want to screen for the exact POP mailbox the email is sent to?

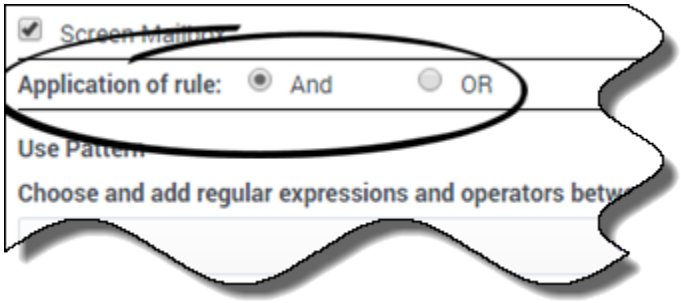
- If so, select **Screen mailbox** to make the rule match the POP box from which the e-mail entered the eServices system, rather than the **To** field of the e-mail itself. The difference is that each e-mail enters the system from exactly one mailbox, while the **To** field can contain multiple addresses.



> ## Important
> For this feature to work as expected, the E-mail Server **enable-same-mail-from-mailboxes** option must be set to `true`. With this setting, E-mail Server creates a separate interaction for each address in the **To** field (that is, for each mailbox that it pulls the e-mail from when it creates the interaction).

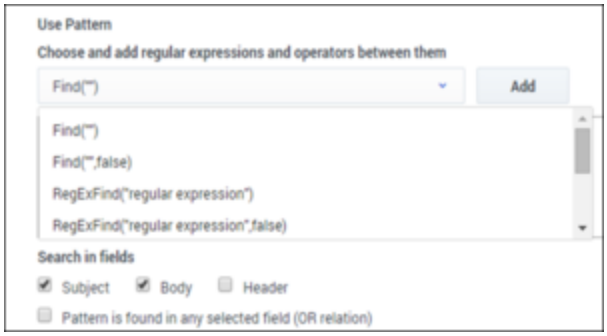
How to use the AND and OR radio buttons

- **AND**
  Choosing **AND** makes Knowledge Manager use *both* the addresses selected in **Use these addresses** *and* match the pattern defined in **Use Pattern**.
- **OR**
  Choosing **OR** makes Knowledge Manager us *either* the addresses selected in **Use these addresses** *or* match the pattern defined in **Use Pattern**.

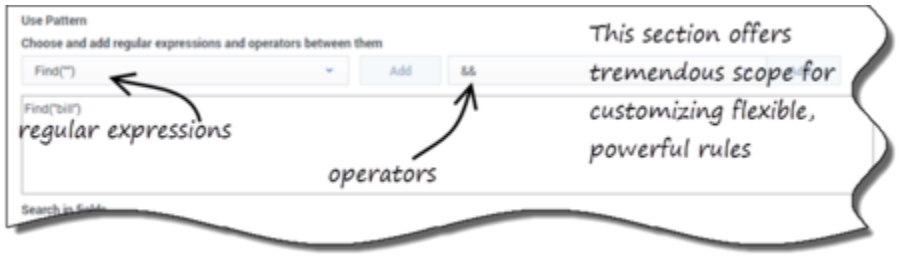


How to configure the Use Pattern area

This is the section where you can get really specific about what the Screening Rule should match. The Plug-in includes a pattern builder that offers the choice of each function type in all possible forms, with and without optional arguments, for a total of eight, as shown below.



The drop-down lists in the **Use Pattern** are contain regular expressions (functions) and operators you can select to create your Rule. Select the regular expression you want, and then click **Add**. The expression appears in the text box. Do the same to add operators. You can also enter text manually.



After you select an expression and click **Add**, you must put text between the quotation marks. More specifically, you must:

- For **Find**, put text between the empty quotation marks.
- For **RegExFind** and **RegExMatch**, substitute your desired text for `regular expression` and/or `key`.

> **Important**
>
> See Screening Rules Reference for an in-depth explanation of how to construct rules.

What message sections should the Screening Rule apply to?

Use the check boxes to have the Screening Rule apply to the message body, subject, header, or any combination. You must select at least one.

If you check multiple boxes, the Screening Rule can behave in one of two ways. See Email Sections to Screen for an explanation.

Assign Categories and Remove Category Assignments

You must select a category from the Category Tree to create a Screening Rule. However, you can remove or add to the categories associated with the rule after you create it.

Click the **Categories** tab to open the **Choose a Category** dialog box.

When an interaction matches this screening rule, it receives a category name and a confidence level. The confidence level indicates that the system is *X* percent confident that this interaction belongs in this category. *X percent* is the Relevancy value you set here.

Choose the desired categories and then click **Save**.

Service Categories

When using the Knowledge Manager interface, it is possible to create screening rules that are not associated with any category. In eServices Manager, any such screening rules are assigned to a category called <tenant><language>; for example, **<AcmeProducts><English>**.

You can add rules to an existing Service Category in any of the following ways:

- By using a rule (created in Knowledge Manager) that has no associated category
- By removing all categories from an existing rule.
- By deleting a category that has a rule associated with only that category.

After any of these, click **Refresh** to see the rule appear in the Service Category.

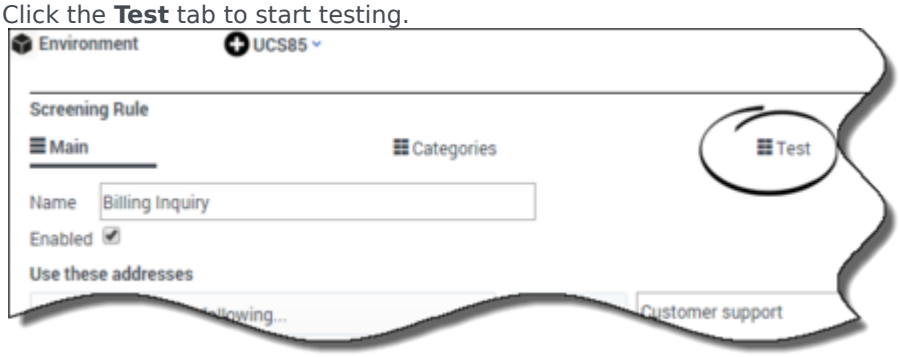
Other details about Service Categories:

- Rules under the service category can be *cut* and *pasted* under other categories. This deletes the rule from the Service Category and associates it with the new category. However, you cannot *copy* rules from a Service Category.
- You cannot paste a rule into a Service Category.
- You cannot cut, copy, or paste the Service Category itself.
- If you delete a Service Category, you also delete all of the rules associated with it and they cannot be recovered.

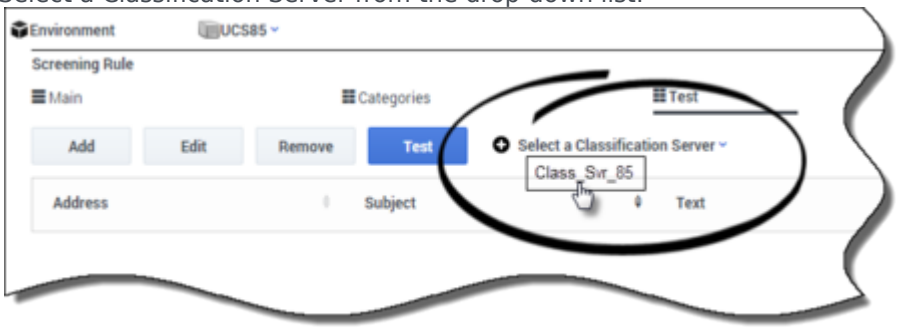
Test a Screening Rule

You will be testing your Screening Rule against some specially created text to see if it works as desired.

1. Click the **Test** tab to start testing.

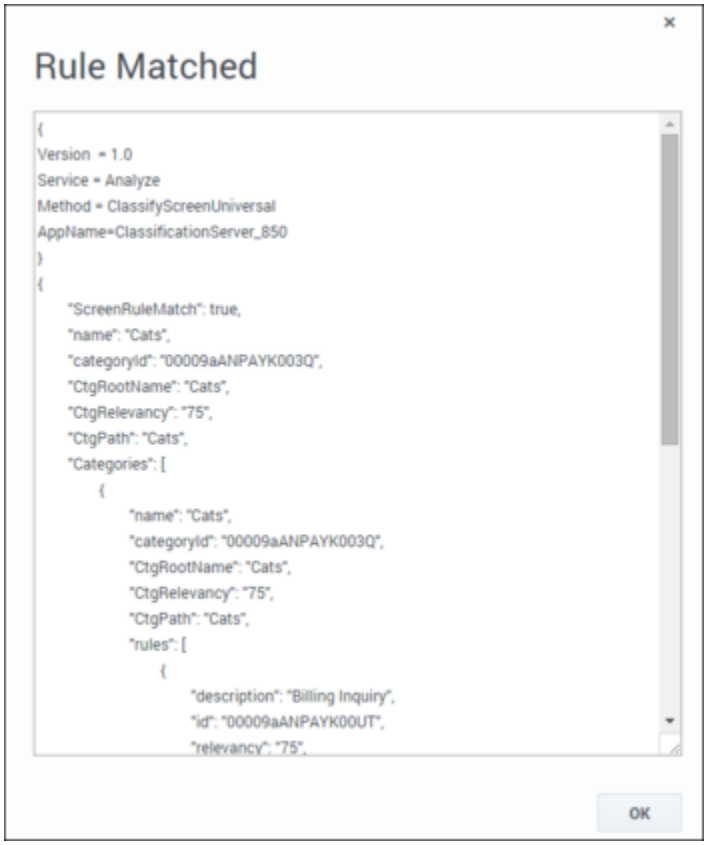


2. Select a Classification Server from the drop-down list.



3. Choose a test message from the list of those already created or click **Add** to create a new one.
4. Click **Test**.

If your test message fulfills the constraints you created for your rule, you will receive a detailed message indicating success.

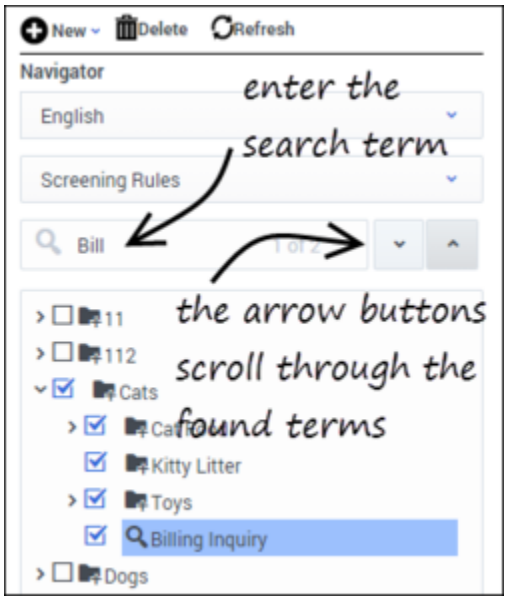


If you get a **Rule Not Matched** message, check the message for clues to the problem. If necessary, analyze your Screening Rule step by step to determine what needs to be edited to make it work the way you want.

> **Important**
>
> The test messages created by each user are stored separately, and are not synchronized between multiple Knowledge Manager instances. Test messages that you create are stored on your local machine under the default Documents folder of the logged-in Windows User folder, in the `UserInfo` file. This file is binary and must not be edited manually.

Finding an Existing Screening Rule

To search for a Screening Rule, make sure **Screening Rules** is selected in the **Navigator** drop-down list. Then start typing in the name of the rule in the **Search** field. The matching names are highlighted as you begin typing.



Next Steps

- Learn more about how screening rules work:
  - Screening Rules Reference
  - Screening Rules Examples

Screening Rule Reference

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

This section provides reference information on the following topics relating to screening rules:

- What they check: What Screening Rules Check
- The functions, arguments, and operators used in them: Functions and Arguments
- Regular expressions to use in them: Regular Expressions

What Text Do Screening Rules Check?

Screening rules check the following parts of an interaction, depending on what you select in the Screening Rule Editor, and on the settings in the IRD screening objects:

- The subject, if you select that check box.
- The body, if you select that check box.
- The header, if you select that check box. See also "Subject, Body, and Header" below, on how screening rules behave if two or more of the preceding are selected.
- The destination address, if you have put anything in the right-hand box in the **Use these addresses** area of the **Edit Screening Rule** dialog box.
- The value of any key in the user data, if both of the following are true:
  - In the Multiscreen or Classify strategy object, you select a key in the **User data key if specified** drop-down list under **Get screened data from**.
  - In the Screening Rule Editor, you select the **Body** check box in the **Use pattern** area. Use the check boxes to have the screening rule apply to the message body, subject, header, or any combination. You must select at least one.

User data is first associated with the interaction by the media server when it creates that interaction. As an example, E-mail Server associates the following user data with the interaction:

- FirstName (from Contact information)

- LastName (from Contact information)
- Mailbox (value of the **address** option in the **[pop-client]** section of the E-mail Server Application object)
- To (MIME header field)
- Subject (truncated to 512 characters)
- FromAddress (personal part of From header field)
- FromPersonal (e-mail address part of From header field)
- All Header fields (except Received, Return-Path, X-MIMETrack, Subject, Sender, From, To, Cc, Bcc) prefixed by `Header_`
- All parent attached data (originally created by E-mail Server) which can be inherited; that is, all parent attached data:
  - not starting with `Header_`
  - not starting with _ (underscore)
  - not equal to `GEM_Failure`
  - not equal to `GEM_FailureMsg`
  - not equal to `GEM_FailureArgs`

User data may then be added or modified by a routing strategy.

Email Sections to Screen

If you select more than one of the Subject, Body, and Header areas, a screening rule can behave in the following two ways:
The default behavior is for the rule to apply to each area in turn; for example, with `Subject` and `Body` selected, the rule applies first to the Subject, then to the Body.

The alternative behavior is for all selected items to first be concatenated so that the rule applies to all at once. There are two ways to achieve this alternative behavior.

- To enforce it for all screening rules, set the `subject-body-header` option for both Knowledge Manager and Classification Server to `true`.
- To enforce it for a particular screening rule:
  1. Leave subject-body-header set to false.
  2. Open the rule in the Screening Rule Editor.
  3. Select the `Merge sources checked above` check box.

> **Important**
> Setting subject-body-header to `true` overrides any selection of the Merge sources checked above check box for a particular rule.

Functions, Arguments, and Operators

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

Functions

Screening rules can use three basic functions:

- `Find("<text>")`, where `<text>` is a text string. It returns the result `true` if the interaction contains the exact string between quotes, ignoring case.
- `RegExFind("<regular expression>")`, where `<regular expression>` is a regular expression (see Regular Expressions). It returns the result `true` if the interaction contains any string that matches the regular expression between quotes.
- `RegExMatch("<regExp>")`, where `<regular expression>` is a regular expression. It returns the result `true` only if the entire content of the interaction matches the regular expression between quotes.

> **Important**
> RegExFind and RegExMatch are the same except that RegExFind looks for a match anywhere in the body of the interaction, whereas RegExMatch demands that the entire body of the interaction match the regular expression.

Arguments

All functions have one required argument, which must appear between double quotation marks, as represented above (`<text>`) or (`<regular expression>`). This required argument can be followed by one or two optional arguments, depending on the function. The full form of each function, including

all arguments, is as follows:

- `Find("<text>", <IgnoreCase>)`
- `RegExFind("<regular expression>",<"key">,<IgnoreCase>)`
- `RegExMatch("<regular expression>",<IgnoreCase)`

IgnoreCase

The IgnoreCase argument must be a Boolean value (*true* or *false*). All three functions ignore case in searches unless you include the IgnoreCase argument with a value of `false`.

For example:

- `Find("pacific")` finds *Pacific* and *pacific.*
- `Find("Pacific",false)` finds *Pacific* but not *pacific.*

You can also substitute `true` for `false`—for example, `Find("Pacific",true)`—which means that case is ignored. So `Find("Pacific",true)` is the same as `Find("Pacific")`.

Key

The key argument must be a string. If this argument is present, the system creates a key-value pair with the following characteristics:

- The key name is the string specified by the key argument, prefixed by `ScrKey_`.
- The value is the material that the screening rule matches.

The system then adds this key-value pair to the interaction's attached data. For example, `RegExFind("[A-Z]\d\d\d","ID_code",false)`:

1. Finds strings consisting of a capital letter followed by three digits (see Regular Expressions).
2. Attaches to the interaction a key-value pair called `ScrKey_ID_code` whose value is A123, X005, M999, or whatever the function found in this interaction to match the regular expression.

Operators

Operators are of two types:

- Binary operators join two functions.
- Unary operators operate on a single function.

The operators are as follows:

&& is the binary operator "and". For example,

```
Find("interest rate") && Find("APR",false)
```

matches a message only if it includes both "interest rate" and "APR."

|| is the binary operator "or." For example,

```
Find("station wagon") || Find("convertible")
```

matches any message that includes either "station wagon" or "convertible" (or "Station Wagon" or "station Wagon" or "Convertible").

! is the unary operator "not." For example,

```
!Find("windows")
```

matches any message that does not include the word "windows."

You can combine `!` with a binary operator. For example,

```
Find("bird") && !Find("goose")
```

matches any message that includes "bird" but does not include "goose."

Operator Precedence

`p && q || r` is parsed as `(p && q) || r`. For example, consider:

```
Find("debt") && Find("income") || Find("profit")
```

To paraphrase, this screening rule is basically "find X or find Y," where X is "debt" and "income," and Y is "profit." It matches both "debt exceeds income" and "profits are fantastic".

You can modify the default precedence by the explicit use of parentheses; for example:

```
Find("debt") && (Find("income") || Find("profit"))
```

This screening rule is basically "find X and find Y," where X is "debt" and Y is either "income" or "profit." It matches both "debt exceeds income" and "debts impact profit."

Regular Expressions

> **Important**
>
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

A regular expression stands for, not one particular character string, but a class of character strings. For example, suppose that you want to find all interactions with U.S. Zip codes in them. U.S. Zip codes are five-digit numbers, so you could in theory write about 9,000 screening rules (`Find("00000")`, `Find("00001")`, `Find("00002")`, and so on).

Fortunately, you can use the special symbol `\d`, which stands for any digit, to write a screening rule using a regular expression: `RegExFind("\d\d\d\d\d")`. This screening rule matches any sequence of five digits.
There are often several different ways of writing the same regular expression.

For instance, two items separated by a hyphen and enclosed in square brackets denotes a range of which the two items are endpoints. So `[a-d]` matches a, b, c, or d, and `[5-8]` matches any digit between 5 and 8; hence `\d` is the same as `[0-9]`.

> **Important**
>
> In general usage, apart from Genesys eServices, regular expressions are case sensitive. However, in the eServices Manager Plug-in, regular expressions are not case sensitive unless you add `,false` as described in IgnoreCase.

The table "Elements of Regular Expressions" lists some of the most commonly-used elements of regular expressions:

Elements of Regular Expressions

Symbol	Meaning	Example
.	Any character, including space	`b.t` matches *bat, bet, bit,* and *but.*
\d	Any digit	`\d\d` matches any pair of digits from 00 to 99.

Symbol	Meaning	Example
\s	Space	\d\s\d matches 1 0, 5 9, and so on.
*	Zero or more instances of the preceding expression	o*f matches *oof, of,* and *f.* me.*d matches *med, mead,* and *meed.*
+	One or more instances of the preceding expression	bre+d matches *bred, breed* and*breeed.*
?	Zero or one instances of the preceding expression	c?rude matches *rude* and *crude.*
{x}	X instances of the preceding expression	st.{2}k matches *steak, stork* , and *stink.*
^	Any character except the following	s[^e]t matches *sat, sit,* and *sot,* but not *set.*
[ ]	Any characters or ranges within the brackets	Any characters: b[aeiou]at matches *boat* but not *brat.* Any range(s): [0-9]th matches *5th, 6th, 7th* [a-z] matches any lowercase letter; [A-Z] matches any uppercase letter.
\	Turns off the special meaning of the following symbol	* matches the character * (asterisk);\. matches the character . (period or full stop).
|	Or	[b|p]ig matches *big* and *pig.* Do not be confused: | means *or* in regular expressions, but || means *or* as one of the Operators used in screening rule formulas.

Here are some other points to keep in mind:

- Space is just another character. The regular expression `savings account` contains a space, and so it does not match the string *savingsaccount.*
- Word boundaries are not considered. The regular expression `read` matches not only *read,* but also *reader, ready, spread, bread,* and so on.
- Use parentheses to group parts of regular expressions together. For example, `RegExFind("(\d{3}\.){2}")` puts `\d{3}\.` in parentheses so that the number-of-instances item `{2}` applies to the all of `d{3}\.`, not just to `\.` This expression matches any group of three digits plus period plus any three digits plus period (for example, 198.351.). Further examples are provided in Examples of Screening Rules.
- Regular expressions make use of many more special characters and operators than those listed in the table "Elements of Regular Expressions." Much documentation on regular expressions is available on the Web. Because Genesys Knowledge Manager uses Java classes for regular expressions, it is best to

consult documents describing the particular version of regular expressions used in Java.

Examples of Screening Rules

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

This section provides examples of screening rules.

Credit Card Number

To find text that includes a typical credit card number, you need to match a sequence of four groups of four digits, each group separated by `-`(hyphen):

```
\d\d\d\d\-\d\d\d\d\-\d\d\d\d\-\d\d\d\d
```

> **Important**
> This regular expression also works without the `\` (backslash) before the hyphens. However, it is better practice to write `\-` for the character hyphen, because the hyphen also has a special use in range expressions like `[a-z]`.

Or if you want to allow for the possibility that some people will omit the hyphens, use? to make the hyphen optional:

```
\d\d\d\d\-?\d\d\d\d\-?\d\d\d\d\-?\d\d\d\d
```

You could also use the repetition notation to shorten each `\d\d\d\d` to `\d{4}`.

North American Phone Number

North American phone numbers consists of ten digits, grouped into two groups of three and one of four. There are a number of ways for the groups to be separated:

```
203-555-1234
```

(203) 555-1234

(203)555-1234

203 555-1234

203.555.1234

The following regular expression matches all of the above:

```
(\d\d\d|\(\d\d\d\))[\s\.\-]?\s*\d\d\d[\-\.]\d\d\d\d
```

The table "Phone Number Regular Expression" analyzes this regular expression.

Phone Number Regular Expression

Symbols	Meaning	Remarks
\d\d\d	Three digits	
\d\d\d|\(\d\d\d\)	Three digits, or three digits enclosed in parentheses	\ turns off the special meaning of the character (
[\s\.\-]?	Space or period or hyphen or zero	Any one of the items enclosed in square brackets, either once or not at all
\s*	Zero or more spaces	
\d\d\d	Three digits	
[\-\.]	Hyphen or period	Note again the need to use \
\d\d\d\d	Four digits	

Telltale Words

To screen for interactions from dissatisfied customers, you might try a regular expression like the following:

```
(not\s([a-z]+\s)*(pleased | satisfied)) | unhappy | complain
```

The first part of this expression matches *not* followed by zero or more words followed by *pleased* or *satisfied*; for example, *not* very pleased, not satisfied, not at all satisfied (but it also matches strings like *can not believe how pleased I am*). The rest matches the single words "unhappy" and "complain."

Cut/Copy/Paste Objects

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

You can cut, copy, or paste categories, standard responses, field codes, and screening rules.

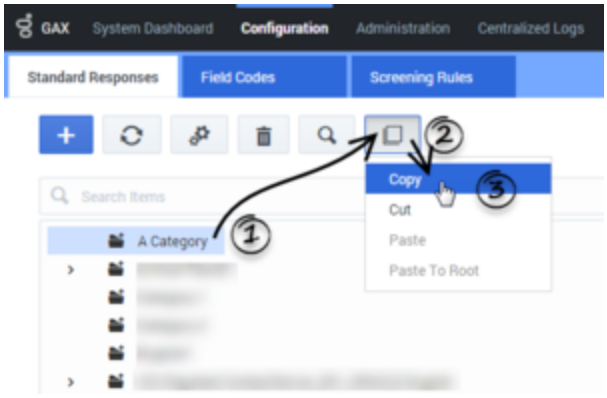
The general procedure is:

1. Select the target object(s).
2. Click  and select an action (for example, **Copy**).

> **Important**
> - When you cut an object, it disappears from its original location only after you paste it somewhere else—except that if you cut a category that has a linked screening rule, the screening rule remains in its original location.
> - You can also delete any object by selecting it, then clicking  .
> - Root category names must be unique across all languages.
> - If you paste an object into a location that includes an object with the same name, the pasted object is renamed **Copy_of_<name>**.

Categories

You can cut, copy, or paste one category at a time.



Copying a category in eServices Manager.

1. Click the category that you want to cut or copy.
2. Click  and select an action (for example, **Copy**).
3. Do one of the following:
   - To paste inside a category: Click the target category, and then click  and select **Paste**.

     > **Important**
     > You cannot use the **Paste** option if you previously cut a source category and it is still selected.

   - To paste to the root category: Click  and select **Paste to Root**.

Standard responses, field codes, and screening rules

You can cut, copy, or paste one or more objects at a time. See Important notes about screening rules for information pertaining only to screening rules.

Copying two standard responses in eServices Manager.

1. Click the check box beside each object that you want to cut or copy.

> **Tip**
> You can click the check box in the header row to select all objects in the table.

2. Click  and select an action (for example, **Copy**).
3. Do one of the following:
   - For standard responses and screening rules: Go to the category into which you want to paste the objects. Click  and select **Paste**.
   - For field codes: Click  and select **Paste**.

> **Important**
> Field codes are shared across all languages.

Important notes about screening rules

Screening rules can be cut, copied, and pasted using the same method as standard responses and field codes. However, there are additional options and restrictions to consider:

- You cannot use the **Paste** option if you cut a screening rule and you are still in the same category. You must go to a different category to use the **Paste** option.
- The **Paste as link** option links a copied screening rule to a target category, but it does not create a separate copy of the screening rule. This is useful if you want to use a screening rule in multiple categories, but you still want to manage the rule from a central location.
- You cannot use the **Paste as link** option in the List view, or if you cut a screening rule in the List view and subsequently switched to the Tree view.
- You can use the **Unlink from Category** option (Tree view) or **Unlink from all Categories** option (List view) to unlink selected screening rule(s) from the category (Tree view) or all categories (List view).

Importing and Exporting Objects

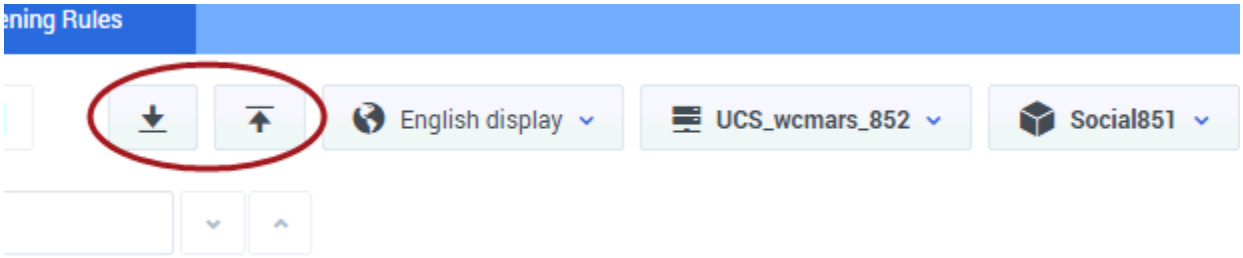
> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

You can export and import categories, standard responses, field codes, screening rules, training data objects, and models (training data objects and models are handled by Genesys Content Analyzer).

> **Important**
> There are two exceptions: test messages and custom variables cannot be imported or exported

Use exported files as backups or to transfer objects between environments.

The import and export buttons are at the upper right of the eServices Manager window, next to the dropdowns for language, UCS, and tenant. Export is the down arrow, and Import is the up arrow.



Export

The process of exporting is simple. After you click the Export icon, series of windows allows you to select:

- Root categories (and their associated category trees)
- Training Data Objects
- Models
- Screening rules

At the beginning, you decide whether the category tree(s) that you select will also include their associated Standard Responses and Field Codes.

The resulting file has the extension `.kme`.

Import a KME file

Clicking the Import icon takes you through these windows:

1. **Import - Select file and options**. Click **Select** to navigate to the KME file you want to import. See below for explanations of the options **Update screening rules** and **Create new UCS ids** (selected by default).
2. **Import - Edit category names**
   If the import file includes any category that has the same name as an existing category, eServices Manager asks you to rename the incoming category.
3. **Import - Process import file**
   This simply shows the progress of the import process.

Note the following:

- The check box **Create new UCS Ids** controls whether the imported records receive new database IDs. Genesys strongly recommends that you leave this check box selected; otherwise, the imported records keep their old IDs and there is a risk of creating uniqueness conflicts. It is only safe to keep old IDs when you are importing into an empty database. One reason to keep the old IDs would be to preserve compatibility with non–eServices Manager components (such as a routing strategy) that need to refer to them.

  > **Important**
  > If the **Create new UCS Ids** check box is cleared and an imported object comes in with an ID that is identical with an existing object's ID, the import process cancels.

- The check box **Update screening rules** controls whether imported screening rules overwrite existing screening rules with the same name. If this check box is not selected, screening rules are treated like all other objects, as described in the preceding paragraph. If the check box is selected, screening rules are treated differently from all other objects: If the names match, the imported screening rule replaces the existing one.
- If an imported screening rule's name does not match any existing rule, but its database ID happens to match that of an existing rule, then eServices Manager creates a new ID for the imported rule.
- If a root category being imported has the same name as an existing category, eServices Manager asks you to change the name of the category being imported.
- If other objects have the same name as existing objects, eServices Manager appends _<hms> to the name of the imported object. <hms> is a timestamp where `h` is the hour (using a 12-hour clock), `m` is the minutes, and `s` is the seconds. Each unit may be one or two digits; there is no padding. For example, if at 4:25:07 PM you import a screening rule called `Sales`, and there is also an existing rule called `Sales`, the new name of the imported rule is `Sales_4257`.

> **Important**
>
> This adds between four and seven characters to the name of the object. You should be especially careful of this if any imported object's name is more than 58 characters long: the added characters may produce a new name that violates the 64-character limit on names of eServices Manager objects. Importing may fail on objects with names that are too long.

- If a Training Data Object contains no training messages, it cannot be imported.

> **Warning**
>
> Once you import an archive that includes any Training Data Object, you cannot import the archive again to a different language. If you attempt to do so, you will receive a message asking you to change the root category name, but when you do that you receive an error message saying that a Training Data Object with that name already exists.
>
> As a workaround, you can rename the Training Data Object before importing the second time.

Import an Excel file

You can perform a bulk import of standard responses from Excel files, including *.xls* (Excel 1997-2003) and *.xlsx*. Other formats are not supported.

The Excel file must use the columns listed in the table below, and these columns must appear in the first row of the first worksheet. All other data is ignored.

Column name	Description	Valid values	Default value	Comments
TheName	Populates in the **Name** field in the UI.	Any string	Mandatory	128 chars max
Body	Populates the body text in the **General** tab. Uses plaintext characters only.	Any string	Empty string	unlimited
StructuredBody	Populates the body text in the **HTML** tab. Uses plaintext characters and HTML.	Any string	Empty string	unlimited
Description	Populates the **Description** field in the UI.	Any string	Empty string	254 chars max

Column name	Description	Valid values	Default value	Comments
Subject	Populates the **Subject** field in the UI.	Any string	Empty string	512 chars max
MimeType	Defines the mime type of the standard response.	Not verified during the import process. Can be left blank.	Empty string	256 chars max
StartDate	Populates the **Start Date** field in the **Properties** tab.	Cell value must be of type **DATE** in Excel.	NULL	
ModifiedDate	Populates the **Modified Date** field in the **Properties** tab.	Cell value must be of type **DATE** in Excel.	Current time	
ExpirationDate	Populates the **Expiration Date** field in the **Properties** tab.	`NeverExpire` or cell value of type **DATE** in Excel.	NULL	

The following example shows a valid table with data:

TheName	Description	Subject	Body	StructuredBody	MimeType	StartDate	ModifiedDate	ExpirationDate
Sales followup	Sent to customers who request contact from Sales	Thank you for contacting us	Thank you for contacting your company. We are happy to assist you with your order.					

Important

- Columns can appear in any order.
- Default values are assigned if a cell value is missing.
- The following rules apply to dates:
  - If both **StartDate** and **ExpirationDate** are empty, eServices Manager assigns **StartDate** to current date and **ExpirationDate** to `NeverExpire`.
  - If **StartDate** is empty and **ExpirationDate** is not empty, eServices Manager assigns **StartDate** a value equal to 30 days before **ExpirationDate**.
  - If **StartDate** is not empty and **ExpirationDate** is empty, eServices Manager sets

ExpirationDate to `NeverExpire`.

- An error message appears if eServices Manager fails to process one or more standard responses. You can check the GAX logs for more details about the cause of the error.
- eServices Manager assumes that all date properties in your UCS environment are stored in the GMT time zone. These properties are always converted to the local time zone by eServices Manager. However, when importing date properties from an Excel file, dates are not converted to local time. For consistency, Genesys recommends you specify dates in the GMT time zone.

To import the Excel file using eServices Manager:

1. In the **Standard Responses** tab, select a category into which the standard responses will be imported.
2. Click  .
3. Select the Excel file to import.
4. Click **Next**.
5. A prompt appears to state whether the import was successful or not. If not successful, the prompt states how many errors were encountered. You can check the GAX logs for more information on the cause of the error(s).
6. Click **Finish**.

Search

> **Important**
>
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

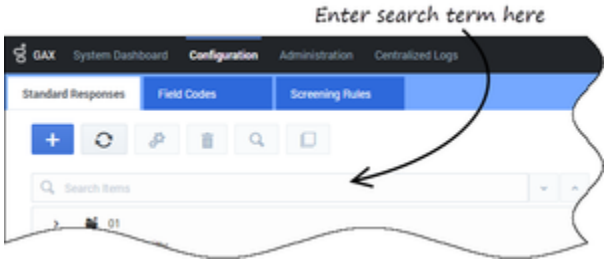
You can search for categories, standard responses, field codes, and screening rules, either by name or by specifying various parameters and properties. Some searches also allow the use of regular expressions.

> **Tip**
>
> Need help with regular expressions? Many possible parameters are documented on the Regular Expressions page in this guide.

Categories

You can search for categories in the **Standard Responses** or **Screening Rules** tabs.

Click the search bar and enter a search term to search the list of categories.



Standard responses

In the **Standard Responses** tab, select a category and click  to search for one or more standard responses.

> **Tip**
>
> Are you unable to click  to search? Remember that you must first select a category in which to search.

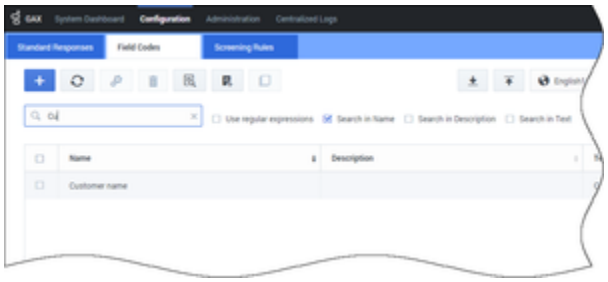
A dialog box appears so you can enter parameters for the search. You can define various parameters for the search by selecting a tab above the search field. In the graphic below, no tabs have a search parameter, as shown by the **0** beside each tab name.

After you define a search parameter in a tab, the value changes to show the number of active search parameters in the tab. For example, in the graphic below, the Text tab has a value of **1**, meaning it has one search parameter, but all other tabs have a value of **0**, meaning these tabs do not have search parameters.

After you define one or more search parameters, click **Find** to begin the search. The **Results** section appears at the bottom of your screen.

Field codes

In the **Field Codes** tab, click  to start a search. Enter a search term in the provided field. You can also specify whether you want to look for the search term in field code names, descriptions, or text. You can also use regular expressions.



You can also search when inserting a field code into a standard response.

Custom variables

You can search for custom variables when you are viewing the **Custom Variables** list.

Screening rules

In the **Screening Rules** tab, select a category and click  to search for one or more screening

rules.

> **Tip**
>
> Are you unable to click  to search? If you are in the Tree view, you must first select a category in which to search.

A dialog box appears so you can enter a search term. You can select or de-select check boxes to specify whether you want to search in the name or pattern of screening rules. You can also use regular expressions.

After you define one or more search parameters, click **Find** to begin the search. The **Results** section appears at the bottom of your screen.

Genesys Co-browse

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Getting Started with Co-browse.

Sometimes your customers need help when they are browsing your website. Maybe they can't tell exactly where to click—or perhaps they need help filling out a complex form. Genesys Co-browse lets your agents use Agent Desktop to do some of the driving for them, by showing them what the customer sees in their browser window (not the whole screen) and allowing the customer to give them control of the web page.

Here are some of the main features of Genesys Co-browse:

- Provides the ability for an agent and the customer to browse and navigate the same web page at the same time.
- Active participation—both the agent and the customer have the ability to take control.
- Browsing always happens on the customer side.
- Pointer Mode and Write Mode—Co-browse sessions begin in Pointer Mode where the agent cannot enter information or navigate for the customer. The agent may send the customer a request to enter Write Mode where the agent can enter information for the customer. The customer must agree to enter Write Mode.
- DOM (Document Object Model) restrictions—hide sensitive data from agents and restrict control of elements (buttons, check boxes, and so on) in a co-browse session.

You can find some more details about Co-browse sessions here.

Getting started

There are a few steps to take in order to get up and running with Genesys Co-browse. Here are the details:

What Genesys does for you

We configure Co-browse for you, based on the following information that you provided to us:

- Allowed origins (your domains and sites from which Co-browse can start)

- Allowed external domains (your domains where static resources for the site are stored)

What you need to do

DOM restrictions

You must configure your DOM restrictions for your website. The Genesys Co-browse DOM Restrictions Editor streamlines this process for you.

Genesys Widgets

You must install and configure Genesys Widgets on your website, in order for the end customer to start a co-browse session with an agent (associated with a chat or a voice session). Everything you need to know about Genesys Widgets is in the following pages:

- Genesys Widgets Deployment
- Genesys Widgets Reference

Here are some examples of Genesys Co-browse enabled in different Genesys Widgets.

Using Co-browse without Genesys Widgets

While Genesys Widgets is the preferred and recommended way to add Co-browse to your website, you may have some scenarios where Genesys Widgets cannot be used. In this case, see the Genesys Co-browse JavaScript API for details on how to deploy Co-browse to your website without using Genesys Widgets. Note that this is considered advanced functionality.

Genesys Co-browse DOM Restrictions Editor

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Co-browse DOM Restrictions Editor.

The Genesys Co-browse DOM (Document Object Model) Restrictions Editor makes it easier for you to hide sensitive customer data from agents and restrict control of elements in a co-browse session. The Editor handles most of the configuration complexity behind the scenes and allows you to focus on your restrictions. You can implement two types:

- **DOM control**—the agent sees the content but won't be able to interact with it. For example, clicking a button or a link won't work for the agent, even when Co-browse is in Write Mode. The agent will see a green border surrounding content with DOM control.
- **Data masking**—the agent sees masked content as asterisks (******) instead of characters, and masked images will be grayed out. The agent will see a purple border surrounding masked characters and images.

> **Important**
> Since masked content is also DOM controlled (non-interactive), the DOM Restrictions Editor shows Data Masking as DOM Control and Data Masking.

With the DOM Restrictions Editor, you can easily

- create, edit, or delete a restriction.
- have an optional description.
- view a list of existing restrictions.
- logically group your restrictions.
- apply a restriction to all web pages, or to the page or set of pages that match the regular expression.
- view all restrictions on the current web page.

To allow users to log into the DOM Restrictions Editor, add them to the **<Tenant name> Administrators** Access Group.

Link to video

Watch the video tutorial on how to use the DOM Restrictions Editor.

Genesys Co-browse sessions

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Co-browse sessions.

A session is initiated when a customer requests to co-browse. The session stays idle until the agent joins, then the session is considered to be active. The session ends when one of the parties (the customer or the agent) exits. It is not possible to re-join a co-browse session. If one party exits accidentally, a new session must be initiated. An agent is limited to handling one co-browse session at a time.

Session identifiers

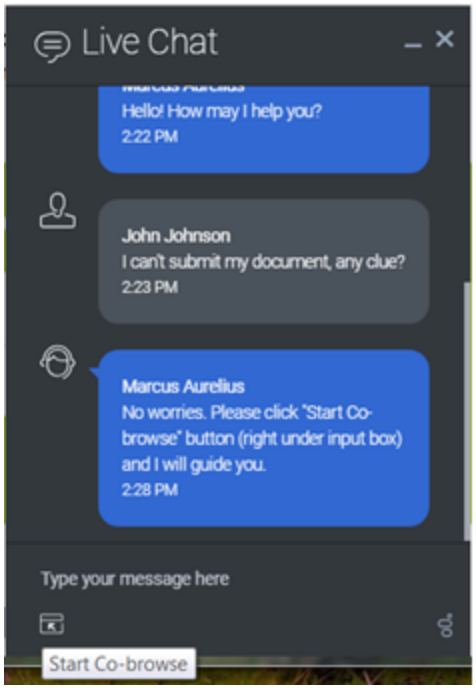
Each live session has an identifier that can be used to track the session. This session ID is a sequence of nine digits that is applicable only to live sessions.

Starting a session

A co-browse session can only be initiated by a customer. An agent does not have the option or ability to send a co-browse request to a customer. This provides greater security to the customer. In order to initiate a co-browse session, the customer must already be engaged in an interaction with an agent, be it a voice call or a chat.

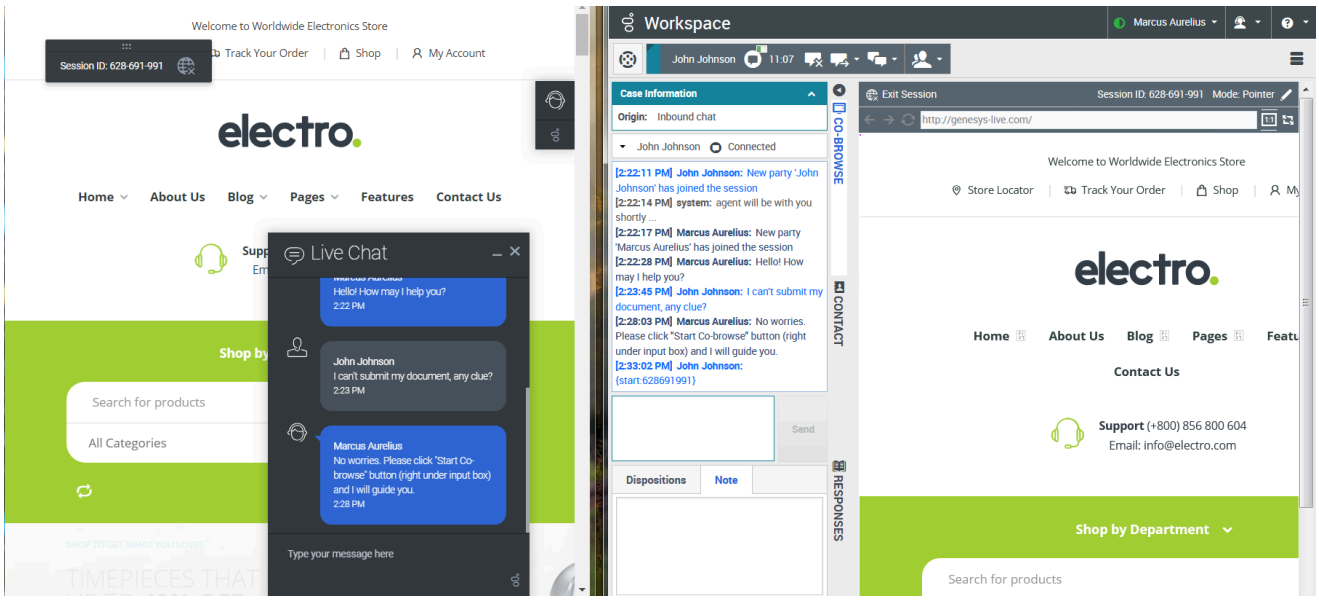
When the session is established, the agent's browser displays a view of the customer's browser. This view that the agent sees is loaded from Genesys Co-browse. The agent is not a client of the website. All actions taken by the agent are passed onto and "replayed" on the customer's side.

Starting a co-browse session from Genesys Widgets

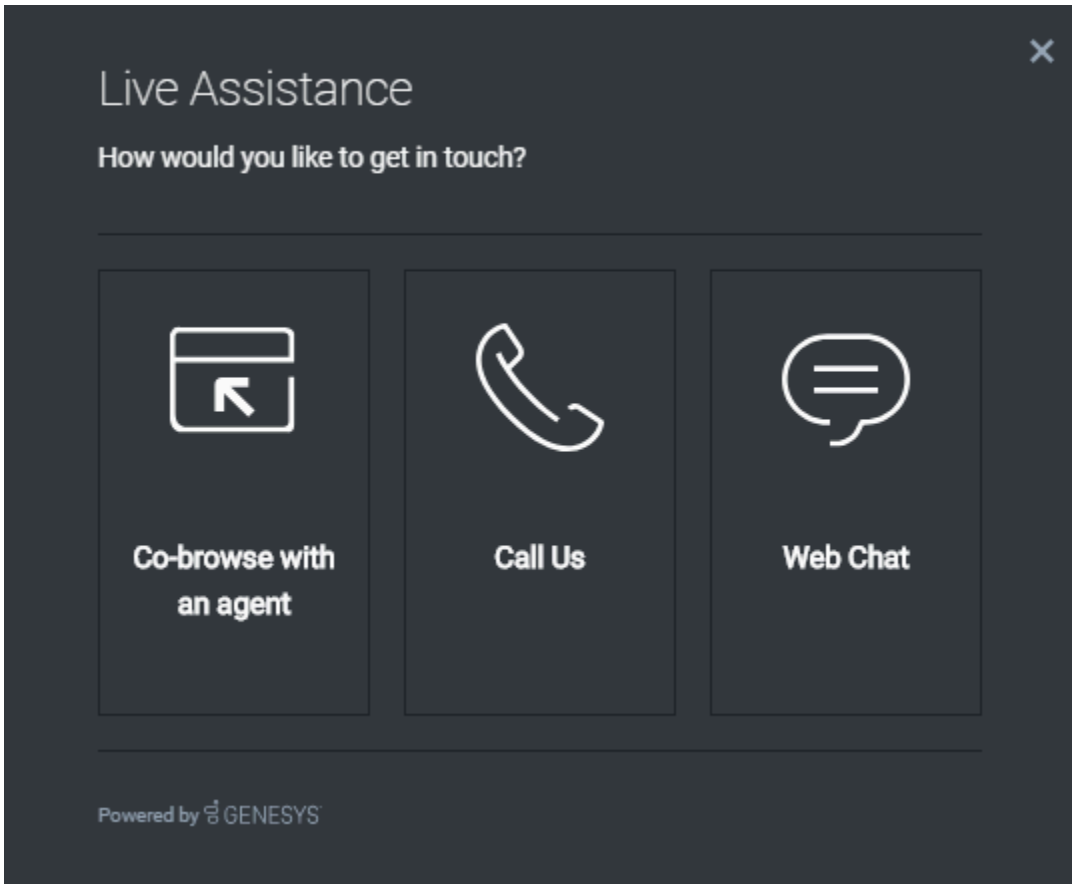


A customer initiates a co-browse session through a Genesys Widget integrated into the website. You can enable Genesys Co-browse in several Genesys Widgets.

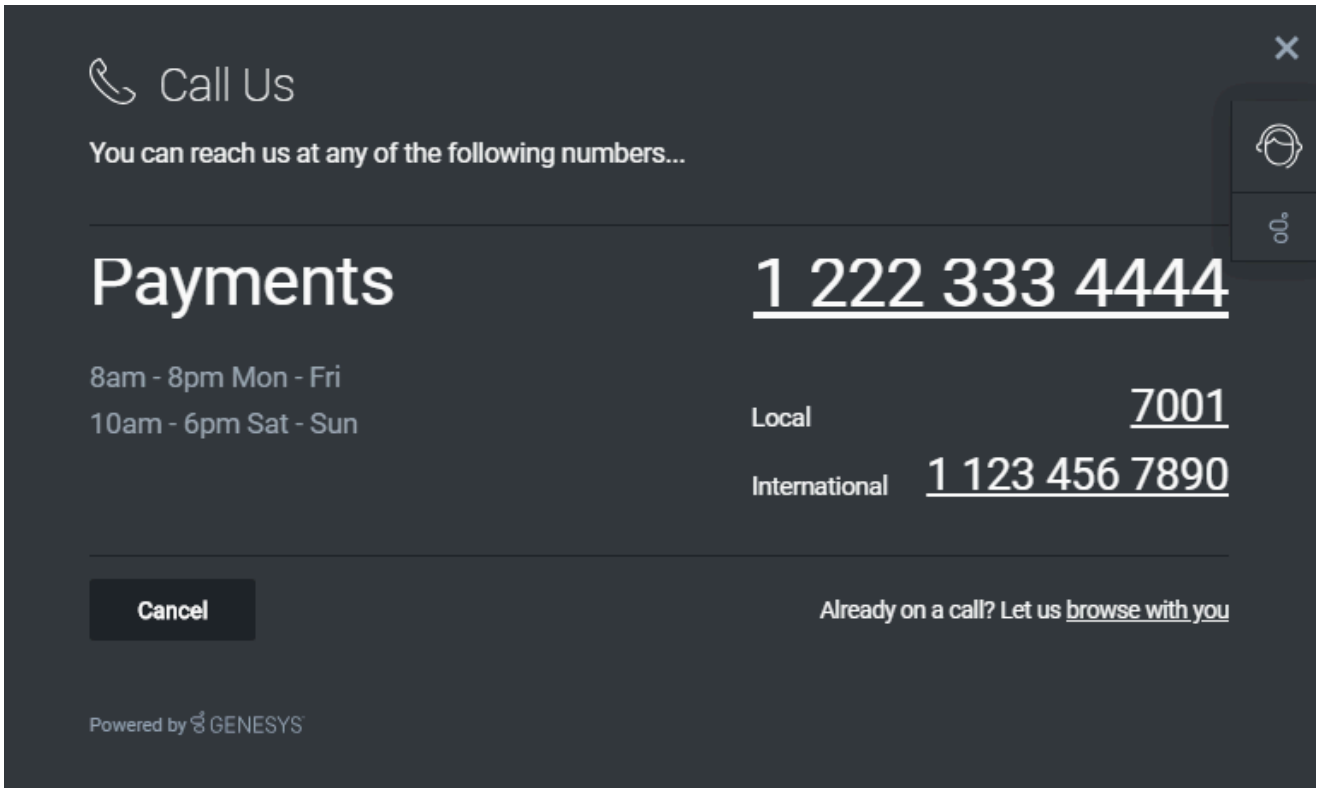
Here is an example of starting a co-browse session from the WebChat Widget.



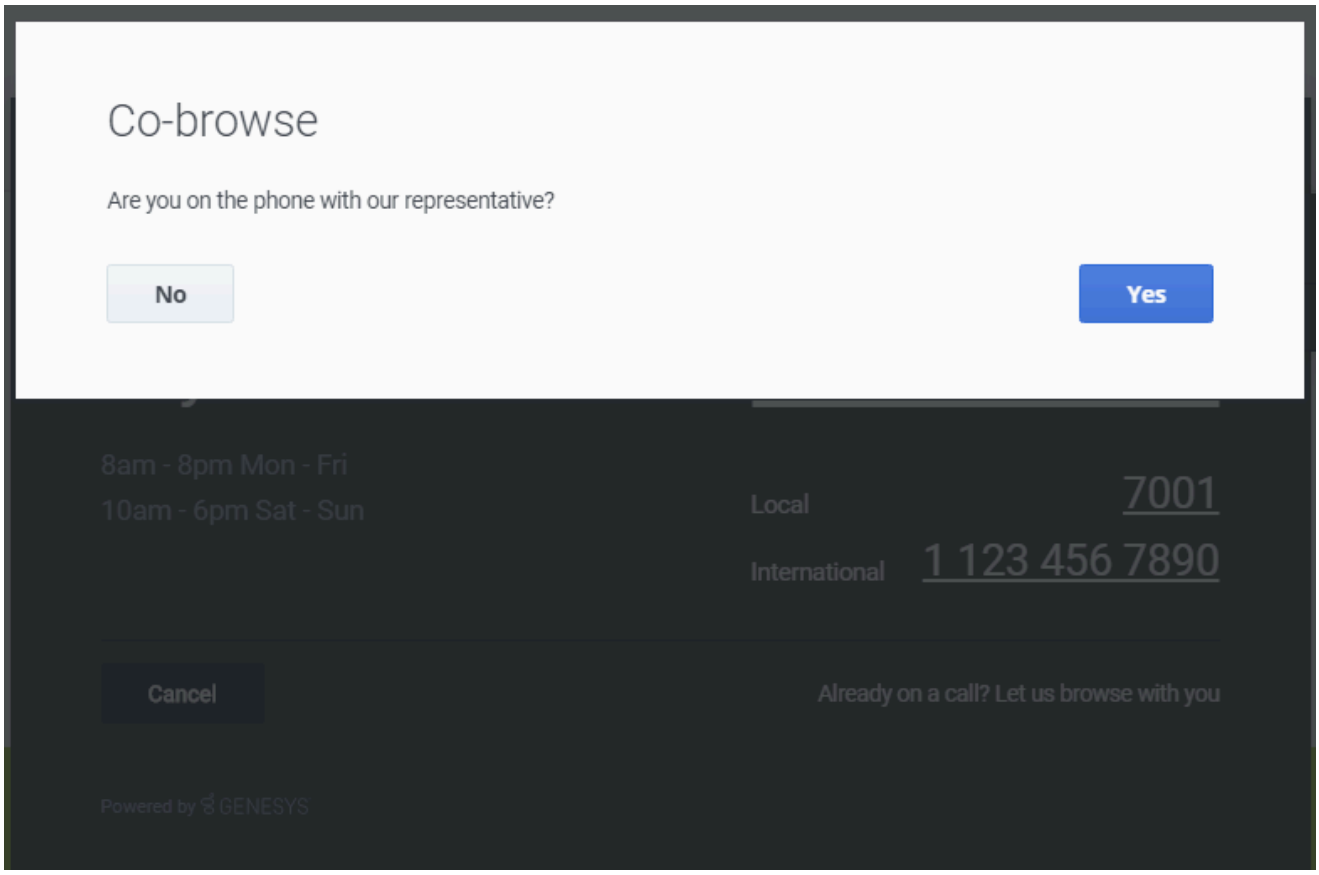
This example shows a customer and agent view of a co-browse session started from the WebChat Widget.



And here is an example of starting a co-browse session from the ChannelSelector Widget.

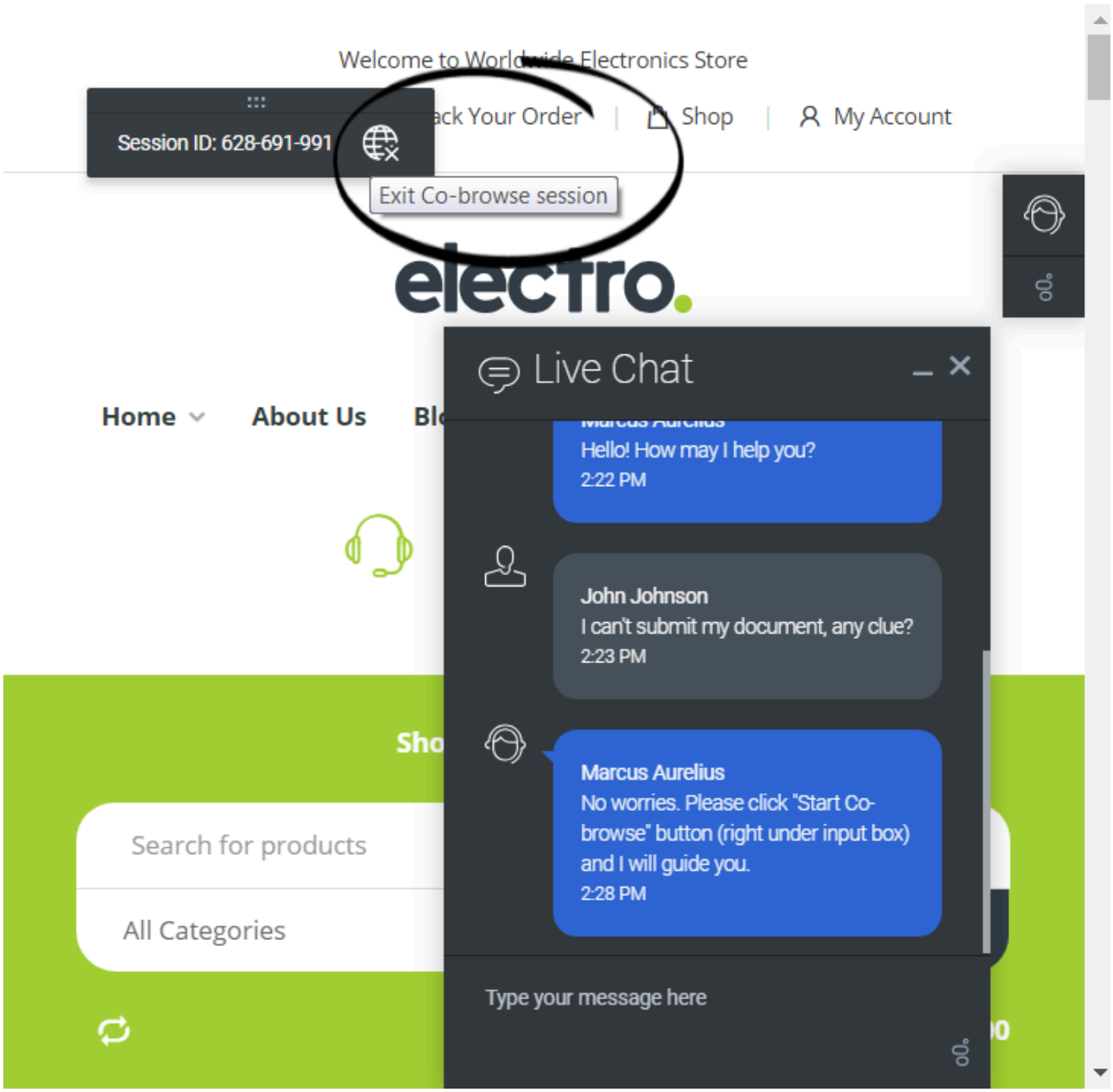


This example shows starting a co-browse session from the CallUs Widget with the **browse with you** link as the co-browse option.



After clicking **browse with you** in the CallUs Widget, the customer will see this dialog.

Stopping a co-browse session from Genesys Widgets



Once a co-browse session has been established, both parties have the ability to terminate the session. At any time, either party may click the **Exit Co-browse session** icon next to the session ID. The agent can also exit by clicking **Exit Session** in Agent Desktop.

The other party will be notified that the session has ended, and the agent's browser will no longer display a view of the customer's browser. Also, if the primary interaction (chat or voice call) is terminated, the co-browse session terminates automatically. Sessions can also terminate due to

inactivity, after a pre-configured timeout expires. Likewise, if the agent closes their browser, or navigates to a third-party website, the session will terminate if the agent does not return to the session page within the pre-configured timeout.

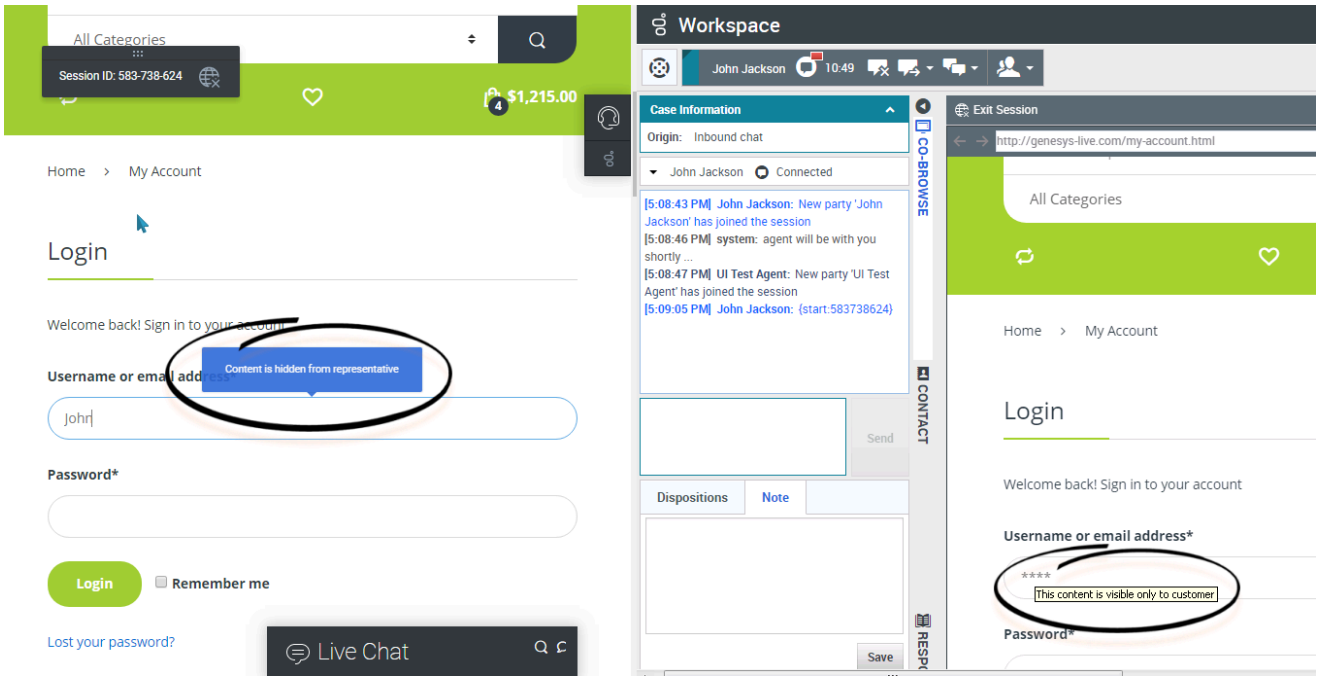
Once a session has been terminated, it cannot be reactivated. If the session was deactivated accidentally, a new session has to be initiated, with a new session identifier.

Participating in a co-browse session

Once a co-browse session begins, the agent can use his or her mouse pointer to guide the customer through the web site. Agents start co-browse sessions in Pointer Mode. In Pointer Mode, the customer and the agent can see each other's mouse pointer but the agent can not enter any information into the web page, click buttons, or navigate the customer's browser. If the agent needs to enter information into the web page or to navigate the browser, he or she can send the customer a request to switch the co-browse session to Write Mode.

All actions (mouse clicks, key presses, and so on) are actually performed on the customer side. Any actions taken by the agent are sent to the customer's browser. This ensures a secure approach, as all browsing is done on one side—the customer's side. This approach also provides for greater performance and a more seamless customer experience. Each participant can see the other participant's mouse movements as well. This enables an agent to point to specific sections on the web page to help direct the customer through their task.

Restricting visibility of sensitive data



You can limit which fields are visible to and editable by the agent and which elements are controlled by agents. This configuration task is made much easier for you by using the Co-browse DOM Restrictions Editor.

Some fields can have the data masked. For example, you might choose to hide the customer's user name, email address, password, Social Security information, and so on, from the agent. The end user can easily identify which information is hidden (data masked) from the agent. By default, all passwords are masked.

At the same time, control for some elements can be disabled. By default, all **Submit** buttons are deactivated for the agent. If he or she clicks on a **Submit** button, nothing happens. The customer always has permission to submit any web forms, just as they would while browsing normally.

Genesys Co-browse Localization

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Co-browse localization.

To localize the end-user Co-browse UI, configure the **localization** option in the global configuration object in **`window._genesys.widgets.cobrowse`**.

> **Important**
> The Co-browse agent-facing UI inherits localization from the Agent Desktop locale settings.

To set the end-user Co-browse UI language to German or French, add `de` or `fr` to the instrumentation:

Example

```
window._genesys.widgets.cobrowse = {

    src: APIGEE_GCB_URL+'/cobrowse/js/gcb.min.js?apikey=APIGEE_GCB_KEY',
    url: APIGEE_GCB_URL+'/cobrowse/',
    apikey: 'APIGEE_GCB_KEY'

    lang: 'de'
};
```

If both `lang` and `localization` are provided, `localization` takes priority. This way it is possible to use built-in German or French localization and override some fields as necessary.

Example

```
window._genesys.widgets.cobrowse = {
    ...
    lang: 'de',
    localization: {
        "modalYes": "Natürlich!"
    }
};
```

If you are using Co-browse outside of Genesys Widgets, use **`window._genesys.cobrowse.localization`** instead of **`window._genesys.cobrowse.widgets.localization`**.

Example

```
if(!window._genesys)window._genesys = {};

window._genesys.cobrowse= {
    src: 'https://www.website.com/cobrowse/js/gcb.min.js',
    url: 'https://www.website.com/cobrowse/',
    localization: {
      // Here we're just changing default "Session ID" to "Session token"
      // You can use this mechanism to adjust default localization to your taste, not
necessarily just for other languages
      'toolbarContent': 'Session token: {sessionId}'
  }
};
```

> **Tip**
>
> You don't have to list all key-value pairs. Ones not listed are inherited from the defaults.

Default values

```
{
  "agentJoined": "Representative has joined the session",
  "youLeft": "You have left the session. Co-browse is now terminated.",
  "sessionTimedOut": "Session timed out. Co-browse is now terminated.",
  "sessionInactiveTimedOut": "Session timed out. Co-browse is now terminated.",
  "agentLeft": "Representative has left the session. Co-browse is now terminated.",
  "sessionError": "Unexpected error occured. Co-browse is now terminated.",
  "sessionsOverLimit": "Representative is currently busy with another Co-browse session. Co-
browse is now terminated.",
  "serverUnavailable": "Could not reach Co-browse server. Co-browse is now terminated.",
  "sessionStarted": "Your co-browse session ID is {sessionId}. Please spell it to our
representative to continue with co-browsing.",
  "navRefresh": "Representative has refreshed the page. Reloading.",
  "navBack": "Representative has pressed the \"Back\" button. Reloading page.",
  "navForward": "Representative has pressed the \"Forward\" button. Reloading page.",
  "navUrl": "Representative has requested navigation. Reloading page.",
  "navFailed": "Navigation request by representative has failed.",
  "toolbarContent": "Session ID: {sessionId}",
  "contentMasked": "Content is hidden from representative",
  "contentMaskedPartially": "Some content is hidden from representative",
  "exitBtnTitle": "Exit Co-browse session",
  "areYouOnPhone": "Are you on the phone with our representative?",
  "areYouOnPhoneOrChat": "Are you on the phone or chat with our representative?",
  "connectBeforeCobrowse": "You need to be connected with our representative to continue with
co-browsing. Please call us or start a live chat with us, and then start Co-browse again.",
  "sessionStartedAutoConnect": "Co-browse session started. Waiting for representative to
connect to the session…",
  "browserUnsupported": "Unfortunately, your browser is not currently supported.<br><br>
Supported browsers are: <ul><li><a target='_blank' href='http://www.google.com/chrome'>Google
Chrome</a></li><li><a target='_blank' href='http://www.firefox.com/'>Mozilla
Firefox</a></li><li><a target='_blank' href='http://microsoft.com/ie'>Internet Explorer 11
and above</a></li><li><a target='_blank' href='https://www.apple.com/safari'>Safari 6 and
above</a></li></ul>",
  "modalTitle": "Co-browse",
  "modalYes": "Yes",
  "modalNo": "No",
```

```
  "writeModeInProgress": "Agent has control over the page.",
  "downgradeMode": "Revoke control",
  "modeUpgraded": "Co-browse session was upgraded. Agent has control over the page.",
  "modeDowngraded": "Co-browse session was downgraded. Agent has no control",
  "modeUpgradeRequested": "Agent requests upgrading Co-browse session to \"write\" mode. In
\"write\" mode agent will have control over the page."
}
```

Genesys Co-browse JavaScript API

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Co-browse JavaScript API.

> **Important**
> This article contains advanced functionality and assumes that you are not using Genesys Widgets to add Co-browse to your website. Genesys Widgets is recommended in most cases.

Deploying Co-browse to your website

Prerequisite: you must have Co-browse provisioned and an API Key provided to you.

A good starting point is this script:

```
<script>
if(!window._genesys)window._genesys = {};
window._genesys.cobrowse = {
  apikey: <APIKEY>
};
</script>
<script>(function(d, s, id, o) {
  var fs = d.getElementsByTagName(s)[0], e;
  if (d.getElementById(id)) return;
  e = d.createElement(s); e.id = id; e.src = o.src;
  e.setAttribute('data-gcb-url', o.cbUrl);
  fs.parentNode.insertBefore(e, fs);
})(document, 'script', 'genesys-js', {
  src: "https://api.genesyscloud.com/gcb/cobrowse/js/gcb.min.js?apikey=<APIKEY>",
  cbUrl: "https://api.genesyscloud.com/gcb/cobrowse"
});</script>
```

Configuring Co-browse

Co-browse is configured via a `global _genesys.cobrowse` variable.

The following options are configurable as properties of an object passed to `_genesys.cobrowse`:

debug

Default: false

Set to `true` to enable debugging console logs.

```
window._genesys.cobrowse = {
  debug: true;
};
```

disableBuiltInUI

Default: false

Set to `true` to use a custom Co-browse UI. Use the Co-browse JavaScript API to implement a custom UI.

```
window._genesys.cobrowse = {
  disableBuiltInUI: true
};
```

You can still start the Co-browse session with just the configuration above, but the main components of the UI, such as the toolbar and notifications, will be missing.

primaryMedia

Default: null

Used to pass an object implementing an external media adapter interface.

Example:

```
var myPrimaryMedia = {
  initializeAsync: function(done) { /* initialize your media here and then call done() */ },
  isAgentConnected: function() { /* return true or false depending on whether an agent is
connected */ },
  sendCbSessionToken: function(token) { /* send the Co-browse session token to agent */ }
};

window._genesys.cobrowse = {
  primaryMedia: myPrimaryMedia
};
```

See Integrating Co-browse with Chat for more details.

If Co-browse does not detect any primary media or detects that the agent is not connected with the primary media, Co-browse will still ask the user, "Are you on the phone with representative?" before starting the Co-browse session.

setDocumentDomain

Default: false

Determines if Co-browse sets the `document.domain` property. If set to `true`, Co-browse modifies the `document.domain` property. If set to `false`, Co-browse does not modify `document.domain`.

Co-browse modifies `document.domain` to support cross-subdomain communication between iframes and the topmost context. Setting `setDocumentDomain` to `false` stops synchronization of subdomain iframes from working.

```
window._genesys.cobrowse = {
  setDocumentDomain: true
};
```

disableBackForwardCache

Default: true

By default, Co-browse disables Safari's Back/Forward cache, which can stop Co-browse sessions from functioning.

Setting `disableBackForwardCache` to `false` can make Co-browse unusable in Safari when users click the back or forward browser buttons.

```
window._genesys.cobrowse = {
  disableBackForwardCache: false
};
```

Accessing the API

Since the main Co-browse JavaScript file is added to the page asynchronously, you cannot instantly access the Co-browse APIs. Instead, you must use a function that will accept the APIs as an argument.

For that, add a special `.onReady` property in `_genesys.cobrowse` configuration and set it to empty array.

```
if(!window._genesys)window._genesys = {};
window._genesys.cobrowse = {
  apikey: <APIKEY>,
  onReady: []
};
```

Now, you can use `_genesys.cobrowse.onReady.push(callbackFn)` anywhere in your code. When the Co-browse JavaScript is loaded and the API is available, Co-browse will call back the `callbackFn` with the reference to the API object.

```
_genesys.cobrowse.onReady.push(function(cobrowseApi) {
  // use the API here
});
```

Using the API

This API provides methods and callbacks to work with Co-browse and can be used to implement a custom UI for co-browsing.

Co-browse in iframes

Co-browse synchronizes the content of any iframes that exist on the page, given:

- the iframe is on the same domain as the page
- the page in the iframe has Co-browse JavaScript

Some Co-browse UI elements, such as the toolbar, should not appear in an iframe. Common Co-browse UI elements (such as notification that an element is masked) should appear whether or not Co-browse is in an iframe. As such, there are two contexts for the Co-browse JavaScript API:

- Top context, available when Co-browse is not in an iframe but in "top" context.
- Child context, used when a page is rendered in an iframe. For the child context, a subset of the top context API is available.

isTopContext

You can use the `isTopContext` variable to determine which context Co-browse is rendered in. `isTopContext` is passed to the onReady callback and equals `true` if Co-browse is rendered in the top context and `false` in iframe.

Example:

```
_genesys.cobrowse.onReady.push(function(cobrowseApi, isTopContext) {
  // common functionality
  cobrowseApi.onMaskedElement.add(function() {/* deal with masked elements here*/});
  if (!isTopContext) {
    return;
  }
  // top context functionality goes below
});
```

See Accessing the API if you are unfamiliar with the onReady syntax above.

Signals and Callbacks

The Co-browse API exposes a number of **signals** in both the top and child contexts. Each signal is an object with the following three methods:

- `add(function)`—adds a callback
- `addOnce(function)`—adds a callback that will be executed only once
- `remove(function)`—removes a callback

The naming convention for signal names begins with "on" and follows the format **onSomethingHappened**.

> **Important**
>
> **Signals** act similar to **deferred** objects. If you add a callback to an event that has already happened, the callback will be called immediately. For example, if you add a callback to the `onAgentJoined` signal when the event has already happened, the callback will be called immediately.

Session Object

Many callbacks receive a `session` object as an argument. This object has the following properties:

- `token`—String containing the session token shared with the agent and possibly shown in the UI. The token is a 9 digit string such as "535176834".
- `agents`—Array of connected agents. Each element in the array is an object with no properties.

Common API

The following elements and properties are available from both the top and child Co-browse contexts:

VERSION

String containing current JS version. For example, `9.0.002.02`.

```
console.log(_genesys.cobrowse.VERSION);
```

onSessionStarted

This signal is dispatched when a Co-browse session is successfully started such as when the `Co-browse` button is pressed or when `startSession()` is called.

Arguments:

- `session`—Session `object` representing the ongoing session.

Example:

```
function notifyCobrowseStarted(session) {
    alert('Co-browse has started. Spell this session token to our representative: ' +
session.token);
}
cobrowseApi.onSessionStarted.add(notifyCobrowseStarted);
```

onSessionEnded

This signal is dispatched when a Co-browse session ends.

Arguments:

- `details`—Object with the follwing field:

- `reason`—Field with value of a string or undefined. Possible string values:
  - `self`—The user has exited the session by clicking the `Exit` button or calling the `exitSession()` API method.
  - `external`—The agent has closed the session. Some server errors may also result in this value.
  - `timeout`—The session has timed out such as when a user reopens a page with an expired Co-browse cookie.
  - `intactivityTimeout`—The agent did not join a session in the configured amount of time.
  - `serverUnavailable`—The Co-browse server was unreachable.
  - `sessionsOverLimit`—Agent is busy with another co-browse session and is prohibited from starting another session at the same time.
  - `error`—There is an error such as a critical misconfiguration.

Example:

```
var cbEndedMessages = {
    self: 'You exited Co-browse session. Co-browse terminated',
    external: 'Co-browse session ended',
    timeout: 'Co-browse session timed out',
    inactivityTimeout: 'Agent did not join. Closing Co-browse session.',
    serverUnavailable: 'Could not reach Co-browse server',
    sessionsOverLimit: 'Agent is busy in another Co-browse session'
}
cobrowseApi.onSessionEnded.add(function(details) {
    alert(cbEndedMessages[details.reason] || 'Something went wrong. Co-browse terminated.');
    showCobrowseButton();
});
```

markServiceElement(element)

Service elements do not show up in the agent's view. This function is used to mark service elements in a custom Co-browse UI.

Arguments:

- `element`—HTML element that will be masked.

> **Important**
>
> Elements must be marked as **service** elements **before** the Co-browse session begins. If the Co-browse session has already started, **service** elements should be marked before they are added to the DOM. It is also possible to mark elements as **service** without using this function. Doing so is useful for static HTML content. To do so, add an attribute `data-gcb-service-node` with value `true`.

> **Important**

> The `markServiceElement()` method should not be used to hide sensitive information. Business functions like DOM Control and Data Masking should be used for sensitive content such as private user data.

Example:

```
function createCustomCobrowseUI(cobrowseApi) {
    var toolbar = document.createElement('div');
    toolbar.className = 'cobrowseToolbar';
    toolbar.textContent = 'Co-browse is going on';
    cobrowseApi.markServiceElement(toolbar); // don't show the toolbar to agents
    cobrowseApi.onConnected.add(function() {
        document.body.appendChild(toolbar);
    })
}
```

Static content example, without JS API usage:

```
<div id="myChatWidget" data-gcb-service-node="true">...</div>
```

onMaskedElement

This signal is dispatched when Co-browse encounters an element that is subject to data masking.

Arguments:

- `element`—HTML Element

This signal is dispatched multiple times when Co-browse initiates and can be dispatched again if a masked element is added to the page dynamically.

Example:

```
cobrowseApi.onMaskedElement.add(function(element) {
    element.title = 'Content of this elements is masked for representatives.';
});
```

Top Context API

The following methods and properties are available only when Co-browse is rendered in the **top** context.

isBrowserSupported()

This method checks for the presence of MutationObserver and a few other required APIs, not for browser type and version. It returns `true` when the browser supports required APIs and `false` otherwise.

startSession()

This method instantiates a new Co-browse session. It will throw an error if the browser is not

supported.

exitSession()

This method exits and ends an ongoing Co-browse session.

downgradeMode()

This method immediately switches the current session from Write Mode to Pointer Mode. The built-in Co-browse UI executes this method when an end user clicks "Revoke Control" while in Write Mode.

See related signals: onModeUpgradeRequested and onModeChanged.

onInitialized

This signal is dispatched after the page is loaded and the Co-browse business logic is initialized.

Arguments:

- `session`— Session object representing the ongoing session or `null` if there is no ongoing session.

Example:

```
cobrowseApi.onInitialized.add(function(session) {
    if (!session) {
        showCobrowseButton();
    } else {
        showCobrowseToolbar(session);
    }
})
```

onAgentJoined

This signal is dispatched when an agent successfully joins a session.

Arguments:

- `agent`—Object representing the new agent. This object has no properties.
- `session`—Session `object` representing the ongoing session.

Example:

```
cobrowseApi.onAgentJoined.add(function(agent, session) {
    alert('Representative has joined the session');
});
```

onAgentNavigated

This signal is dispatched when the agent initiates navigation such as refresh, back, forward, or enters a URL into the agent Co-browse UI. Signal is dispatched a few seconds before the navigation happens. This can be used, for example, to send a warning to the user or disable the `Exit session` button before navigation.

Arguments:

- `details`—Object containing the following navigation detail fields:
  - `command`—String with the value of `back`, `refresh`, `forward`, or `url`.
  - `url`—Optional string that is present only if the `command` field has the value of `url`.

Example:

```
cobrowseApi.onAgentNavigated.add(function(details) {
    if (details.url) {
        alert('Representative has navigated to the page: ' + details.url);
    } else {
        alert('Representative has pressed the "' + details.command + '" button. Page will be
refreshed');
    }
});
```

onNavigationFailed

This signal is dispatched when the navigation request from the agent fails to execute such as when the agent navigates forward when there is no forward history. You can use this signal to to re-enable the `Exit` button and/or show a notification.

The callback receives no arguments.

Example:

```
cobrowseApi.onNavigationFailed.add(function() {
    alert('Navigation request by representative has failed');
});
```

onModeUpgradeRequested

This signal is dispatched when an agent requests upgrading the Co-browse session to Write Mode.

Arguments:

- `done`—The function passed by the Co-browse code. Call it with `true` to allow the transition to Write Mode, or with `false` to prohibit.

Example:

```
cobrowseApi.onModeUpgradeRequested.add(function(done) {
  if (confirm('Representative requests control over the web page. Allow?') {
    done(true); // allow upgrading to Write Mode
  } else {
    done(false); // disallow and stay in Pointer Mode
  }
});
```

onModeChanged

This signal is dispatched when the Co-browse session Mode changes, either to Pointer or Write.

Arguments:

- `mode`—An object with two boolean properties:
  - `pointer`—This is `true` if the session has switched from Write to Pointer Mode. Otherwise, it's `false`.
  - `write`—This is `true` when the session has switched from Pointer to Write Mode.

Example:

```
cobrowseApi.onModeChanged.add(function(mode) {
    if (mode.write) {
        alert("Representative has now control over the page");
    } else if (mode.pointer) {
        alert("Representative can no longer control the page").
    }
});
```

Integrating Co-browse with Chat

External chat can be connected to Co-browse via an adapter object assigned to the `_genesys.cobrowse.primaryMedia` option. Such object may implement the following methods:

initializeAsync(done)

Use this only if your chat initializes asynchronously and you cannot be sure it is ready before Co-browse.

If the `initializeAsync` method is implemented, the Co-browse JavaScript will call the method and pass it a done callback. You must call the done callback when your media finishes initialization.

```
var myChatAdapter = {
  initializeAsync: function(done) {
    waitForChatInitialization(function() {
      // tell Co-browse chat is now ready
      done();
    });
 }
};
```

sendCbSessionToken(token)

Co-browse will use it to pass the session token to the agent. The Co-browse session token is a string consisting of nine digits.

Example:

```
myChatAdapter = {
```

```
    sendCbSessionToken: function(sessionToken) {
        myChat.sendMessage('User has started Co-browse session: ' + sessionToken);
    }
};
```

If you use Genesys Agent Desktop, wrap the Co-browse token in a `{start:<TOKEN>}`, then the agent will join a Co-browse session as soon as he or she receives the token.

```
// For example:
myChatAdapter.sendCbSessionToken = function(token) {
  myChat.sendMessage('{start:' + token + '}');
};
```

isAgentConnected()

This method must return a `true` or `false`.

Co-browse calls this method before calling the `sendCbSessionToken`. If `isAgentConnected` returns `true`, Co-browse will call the `sendCbSessionToken` method. If `isAgentConnected` returns `false`, the user will be asked to connect with an agent via phone before starting Co-browse. If the method is absent, the user will be asked to connect with an agent via phone or chat before starting Co-browse.

Genesys Co-browse Realtime API (CometD and REST)

> **Important**
>
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

This content is restricted. Please contact your Genesys representative for access permissions.

> **Important**
>
> To access this content:
>
> - **Customers**: Log in to My Support and select *Documentation*.
> - **Partners**: Log in to Partner Portal and select *Genesys Technical Docs*.
> - **Employees**: Go to the internal access point.

Click here to access this content (login required).

Genesys Predictive Routing

> **Important**
>
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

What is Genesys Predictive Routing?

Genesys Predictive Routing (GPR) draws on accumulated agent, customer, and interaction data, enabling you to analyze omnichannel interactions and outcomes and generate models to predict outcomes. From this analysis, combined with machine learning, you can determine the best possible match between waiting interactions and available agents to improve your chosen KPIs, and then route the interactions accordingly.

In addition, you can report on the predicted versus actual outcomes. The actual outcome is also used to further train the machine-learning model, improving the accuracy of predicted outcomes between similar customer profiles and agent profiles.

You can:

- Review the Customer Profile schema the Agent Profile schema, and interaction and other data that is automatically collected from Genesys Info Mart or uploaded from prepared CSV files using the Data Loader.
- Use the Feature Analysis report to identify which factors most strongly affect various KPIs. The results enable you to create more effective predictors and models.
- Use the Agent Variance report to determine where differences between agent effectiveness in different scenarios offers potential for improved outcomes.
- Create predictors and models based on your imported data.
  - Model quality report: Provides an analysis of how well the model is performing.
  - Agent Coverage report: Indicates how many agent models were built, as a function of the total agents available.
- Review and test the performance of your predictors and models, as well as viewing your customer and agent distribution and details.
- Monitor jobs that you are running or have run.

Access the complete Help.

> **Important**
> Some features described in the Help are active for only certain user roles.

For deployment instructions, system requirements, and other information related to the on-premise components—Data Loader and URS Strategy Subroutines—see the Deployment and Operations Guide.

Genesys Softphone

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

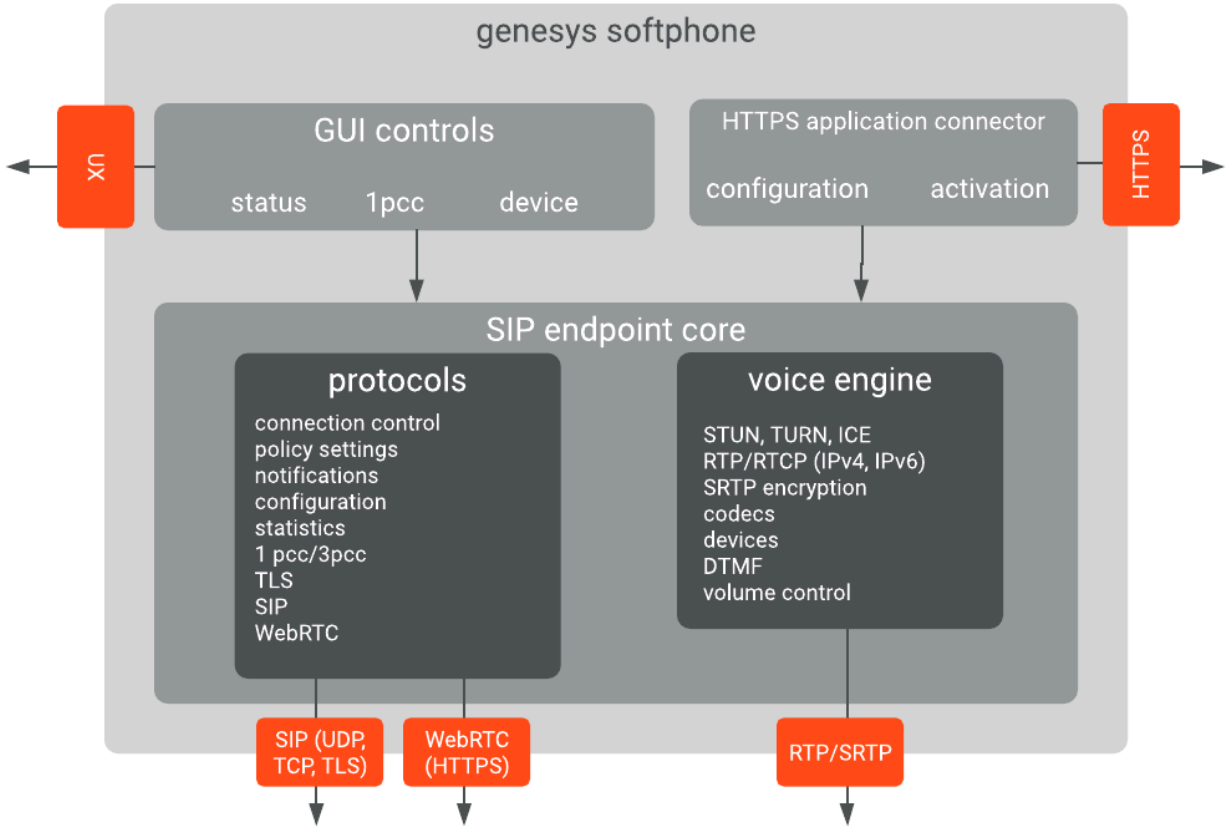
The architecture and features of Genesys Softphone when deployed in your environment.

Architecture

Genesys Softphone embeds the Genesys SIP Endpoint Core Library to take advantage of the SIP-based third-party call control functionality.

Standard Architecture

The following diagram illustrates the Genesys Softphone architecture when it is installed on a physical workstation as a standard executable gathering all product functionalities:



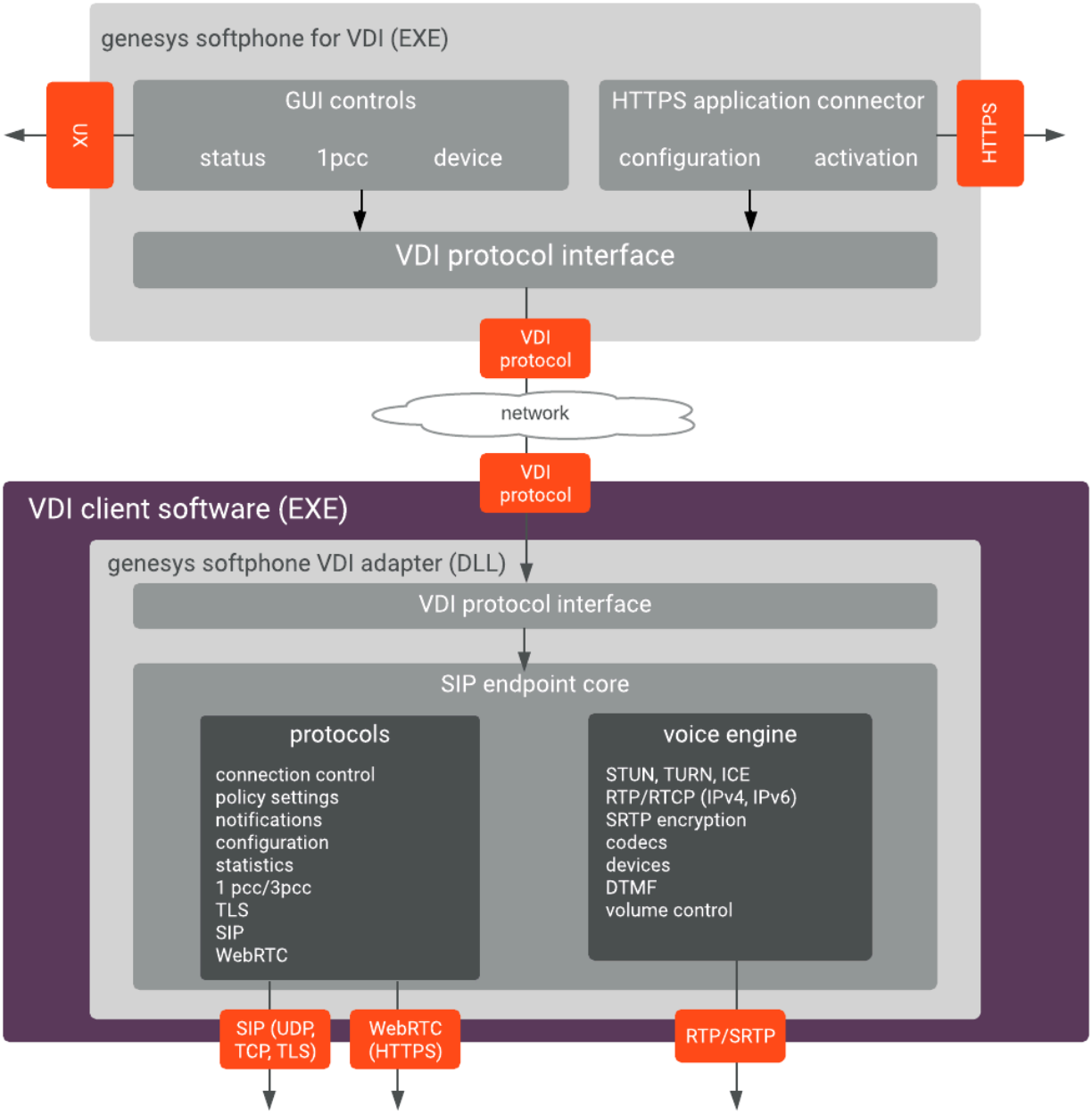
Architecture in VDI environments

Genesys Softphone supports the Citrix Virtual Desktop Infrastructure (VDI). When deployed in this kind of environment, the Genesys Softphone software is divided into two parts:

1. The application layer, running in the Virtualized system. This is the Genesys Softphone executable. The user interface runs here as well as connectivity with other applications, such as Agent Desktop. You install this through the Genesys Softphone installation package by selecting the Citrix installation option.
2. The Signaling Protocols, the Media Protocols, and the Audio Device management. These are off-loaded to the physical workstation to optimize call quality and ensure network and data center scalability. It is a plug-in (DLL) to the VDI Client run-time (Citrix Workspace app, previously known as Citrix Receiver). It is deployed by the Genesys Softphone VDI Adapter installation package.

The two Software parts communicate over the Citrix ICA proprietary protocol already established for standard Citrix operations; therefore, there is no need for any extra connectivity settings.

The following diagram illustrates the Genesys Softphone architecture in the Citrix VDI environment:



Features and functionality

Genesys Softphone media stack is based on Google's open source WebRTC Native Code package. Softphone includes an adaptive jitter buffer, Packet Loss Concealment (PLC), echo cancellation, and noise reduction. For more information refer to SDK for .NET.

The following are the standard features and functions of Genesys Softphone.

DTMF

The Genesys Softphone supports Dual-Tone Multi-Frequency (DTMF) signalling according to the RFC 2833 standard for third-party call control.

After receiving a NOTIFY with DTMF event, the Softphone Endpoint generates DTMF signals.

DTMF can be sent by using one of the three possible methods:

- InbandRTP
- RFC 2833
- SIP INFO message

Third-party call control

When the Genesys Softphone Endpoint has registered on the Genesys SIP Server, it will support the following third-party call control scenarios:

- Make a call
- Answer a call
- Hold and retrieve a call
- Single-step and two-step transfers
- Participate in a conference that is provided by the GVP
- Play DTMF signals.

SIP Voice

The Genesys Softphone supports the following codecs for SIP signaling:

- PCMU/8000 (G.711/mu-law)
- PCMA/8000 (G.711/A-law)
- G722/16000
- iLBC/8000 (iLBC — internet Low Bitrate Codec)
- iSAC/32000 ((iSAC/32kHz) — internet Speech Audio Codec)
- iSAC/16000
- G729/8000
- OPUS/48000/2

WebRTC Voice

The Genesys Softphone supports the following codecs for WebRTC signaling:

- OPUS
- G711

WebRTC and TLS in Windows 7

When Genesys Softphone is used for WebRTC communication, TLS 1.2 is used; however, Windows 7 does not support TLS 1.2 by default; therefore, you must enable TLS 1.2 in Windows 7 before you can use Genesys Softphone in WebRTC mode.

Refer to the following Microsoft document for the procedure to enable TLS 1.2: *Update to enable TLS 1.1 and TLS 1.2 as default secure protocols in WinHTTP in Windows*.

WebRTC and OAuth Support

WebRTC with OAuth is supported in Agent Desktop/Softphone connector mode only. If you use the standalone Softphone mode, you must migrate to connector mode. To enable this feature in Agent Desktop you must configure the value of the `sipendpoint.enable_webrtc_auth` option to `true`.

Virtual Desktop Infrastructure (VDI)

Softphone supports Virtual Desktop Infrastructure (VDI) to enable agents to use Softphone in a VDI environment.

Softphone can be deployed in a Citrix virtual environment.

- Prerequisites for installing Softphone in a Citrix VDI environment
- Installing the Genesys Softphone VDI Adapter

Localization

Starting from release 9.0.012.04, Genesys Softphone can be presented in various languages.

In Connector Mode, some agent applications like Workspace Web Edition can automatically align the language with the one selected in the controlling application. In other cases, the agent can select the language using the appropriate menu.

Deploying Genesys Softphone

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

How to deploy and configure the Genesys Softphone in your environment, including both standard and Virtual Desktop Infrastructure (VDI) installations.

> **Important**
> You should receive access to the Genesys Softphone download when you purchase Softphone. Contact your Genesys representative if you did not receive access to the installation package.

Environment prerequisites

Ensure that your environment meets the prerequisites described in this section.

Supported operating systems

Refer to the Genesys Softphone and the Virtualization Platform Support topics in the *Genesys Supported Operating Environment Reference Manual* for a list of the latest supported operating systems.

Prerequisites for deployment on a physical workstation

To work with Genesys Softphone, you must ensure that your system meets the software requirements established in the *Genesys Supported Operating Environment Reference Manual*, as well as meeting the following minimum software requirements:

- Visual C++ Redistributable Packages for Visual Studio 2013 (32 bits version): The Genesys Installation Package installs this redistributable package on the workstation where it is executed.
- .NET Framework 4.0 or higher: This is used at installation time only when the Administrator installs Genesys Softphone with HTTPS connector based on a *self-signed certificate*.
- QoS requirement for voice, either one-to-one or on a conference connection capability, require the

following:

- ≤ 150 ms of one-way latency from mouth to ear (per the ITU G.114 standard)
- ≤ 30 ms jitter
- ≤ 1 percent packet loss
- 17 to 106 kbps of guaranteed priority bandwidth per call (depending on the sampling rate, codec, and Layer 2 overhead)
- 150 bps (plus Layer 2 overhead) per phone of guaranteed bandwidth for voice control traffic

- A headset or other microphone and speaker audio device that is supported by Windows OS installed on the workstation.

Prerequisites for deployment in a VDI environment

To work with Genesys Softphone in a VDI environment, you must ensure that your system meets the software requirements established in the *Genesys Supported Operating Environment Reference Manual*, as well as meeting the following minimum software requirements:

1. On the workstation running Citrix Workspace (formerly Citrix Receiver):
   - Visual C++ Redistributable Packages for Visual Studio 2013 (32 bits version): The Genesys Installation Package installs this redistributable package on the workstation where it is executed.
   - QoS requirement for voice, either one-to-one or on a conference connection capability, require the following:
     - ≤ 150 ms of one-way latency from mouth to ear (per the ITU G.114 standard)
     - ≤ 30 ms jitter
     - ≤ 1 percent packet loss
     - 17 to 106 kbps of guaranteed priority bandwidth per call (depending on the sampling rate, codec, and Layer 2 overhead)
     - 150 bps (plus Layer 2 overhead) per phone of guaranteed bandwidth for voice control traffic
   - A headset or other microphone and speaker audio device that is supported by the OS installed on either the client host.
2. On the VDI environment (XenApp or XenDesktop server) that runs the application layer of the VDI runtime:
   - Visual C++ Redistributable Packages for Visual Studio 2013 (32 bits version): The Genesys Installation Package installs this redistributable package on the workstation where it is executed.
   - .NET Framework 4.0 or higher: This is used at installation time only when the Administrator installs Genesys Softphone with HTTPS connector based on a *self-signed certificate*.

Installing Genesys Softphone

(For information on installing Genesys Softphone in a VDI environment see Installing the Genesys Softphone VDI Adapter)

To install Genesys Softphone, follow these steps:

1. Double-click the **setup.exe** file that is located in the **<Genesys Softphone Install Package Directory>\windows** directory to open the **Genesys Installation Wizard**.
2. In the **Welcome to the Installation** window, click **Next**.
3. In the **Choose Destination Location** window, click **Next** to accept the default destination folder, or click **Browse** to select another destination location.
4. In the **Deployment Type** window, click **Standard** or **Citrix** (for Virtualization deployments only) then click **Next**.
5. In the **Startup and Secure Connection options** window, you may choose one or more of the following options, and then click **Next**:
   - Auto Startup: Specifies that Genesys Softphone launches when Windows starts up. This means that agents do not have to manually launch Genesys Softphone before they launch Agent Desktop.
   - Enable Dynamic Configuration Connector: Specifies that Agent Desktop is allowed to dynamically configure Genesys Softphone when it is launched.

     If you choose the Enable Dynamic Configuration Connector option, the **Dynamic Configuration Connector Parameters** window is displayed.

     a. Specify the Connector Port for Genesys Softphone. This port must be compliant with the value specified by the sipendpoint.uri option.

     b. Select **HTTPS secure connections**. You must then indicate the type of security certificate you are using:
        - Self-signed Certificate: In this mode, the IP creates a self-signed certificate, installs it in the Personal Certificate section of the workstation where **setup.exe** is executed and also installs it as a root certificate authority at Machine level in the workstation where **setup.exe** is executed.
        - Certificate Authorities from the Windows Certificate Store

     > **Important**
     >
     > To properly install the self-signed certificate, .NET Framework 4.0 or higher is mandatory.

6. In the **Ready to Install** window, select **Install**. The wizard installs Genesys Softphone and all associated files in the selected directory and displays the **Installation Status** window. The installation might take several minutes.
7. In the **Installation Complete** window, select **Finish**.

> **Important**
>
> For more information about Genesys Softphone deployment for Agent Desktop, see Single sign on with Agent Desktop.

Installing Genesys Softphone in Silent mode

To install Genesys Softphone in Silent mode, use the Installation Wizard **Silent** arguments as follows:

1. Update the **genesys_silent.ini** file by making the following modifications:
   - Add the path to the Genesys Softphone directory. For example, **InstallPath=C:\GCTI\Genesys Softphone**.
   - Specify if Genesys Softphone is a physical workstation ("Std") or a Citrix enviornment ("Citrix") by using the **DeploymentType=<Std or Citrix>** parameter.
   - Specify whether Genesys Softphone starts automatically when Windows starts up by using the **Startup=<Std or Auto>** parameter.
   - Specify whether Agent Desktop can dynamically modify the Genesys Softphone configuration by using the **Connector=<Disable or Enable>** parameter.
   - If you are *deploying* Softphone for Agent Desktop dynamic configuration:
     - If the Connector is enabled, specify the Connector Port by using the **ConnectorPort=<port number>** parameter.
     - Specify that the connector uses an HTTPS secure connection by using the **HTTPS=Used** parameter.
     - If you are using a secure connection, specify the certificate type to be used by using the **CertificateType=<SelfSigned or WindowsStore>** parameter.
     - If you assign the value **WindowsStore** to the **CertificateType** option, specify the certificate thumbprint by using the **CertThumbPrint=<certificate thumbprint>** parameter.
   - If you are *upgrading* Genesys Softphone specify:
     - **IPVersion= <current version of Genesys Softphone on this box (before upgrade)>**
     - **IPBuildNumber= <current build number of Genesys Softphone on this box (before upgrade)>**
2. Execute the following command:
   `setup.exe /s /z"-s 'FullPathToGenesysSilentConfigurationFile' -sl 'FullPathToGenesysSilentResultFile'"` where:
   - `/s` specifies that the installation is running in InstallShield Silent Mode.
   - `/z` passes the Genesys Silent mode silent parameters to the installation.
   - `-s` specifies the full path to the silent configuration file. The **<Full path to Genesys Silent Configuration file>** is optional. If the **<Full path to Genesys Silent Configuration file>** parameter is not specified, the installation uses the **genesys_silent.ini** file in the same directory where the **setup.exe** is located.

     > **Important**
     >
     > Enclose the value of the **<Full path to Genesys Silent Configuration file>** parameter by apostrophes (') if the parameter contains white symbols.

   - `-sl` specifies the full path to the installation results file. If the **<Full path to Genesys Installation**

Result file> parameter is not specified, the installation creates the **genesys_install_result.log** file in the **<System TEMP folder>** directory.

> **Important**
>
> Enclose the value of the **<Full path to Genesys Installation Result file>** parameter in apostrophes (') if the parameter contains white space characters.

The **InstallShield setup.exe** installation starter requires that:

- There is *no* space between the `/z` argument and quotation mark. For example, `/z"-s"` is valid, while `/z "-s"` is not valid.
- There *is* a space between the -s,-sl parameters and quotation mark. For example, `/z"-s c:\temp\genesys_silent.ini"` is valid, while `/z "-sc:\temp\genesys_silent.ini"` is not valid. For example,
  `setup.exe /s /z"-s 'C:\8.5.000.05\windows\b1\ip\genesys_silent.ini' -sl 'C:\GSP\silent_setup.log'"`.

3. After executing this command, verify that Genesys Softphone is installed in the **C:\<Genesys Softphone Directory>**, and that the **silent_setup.log** file has been created in the **C:\GSP** directory.

Installing the Genesys Softphone VDI Adapter (Windows)

If you installed Genesys Softphone in a VDI environment, you must install the Genesys Softphone VDI Adapter on each workstation by following these steps:

1. Double-click the **setup.exe** file that is located in the **<Genesys Softphone VDI Adapter Install Package Directory>\windows** directory to open the **Genesys Installation Wizard**.
2. In the **Welcome to the Installation** window, click **Next**.
3. In the **Select Operating System** window, select **Windows** and click **Next**.
4. In the **Ready to Install** window, select **Install**. The wizard installs Genesys Softphone VDI Adapter and displays the **Installation Status** window.
5. In the **Installation Complete** window, select **Finish**.

Installing the Genesys Softphone VDI Adapter (eLux)

If you installed Genesys Softphone in a VDI environment, you must install the Genesys Softphone VDI Adapter on each eLux workstation by following these steps:

1. Double-click the **setup.exe** file located in the **<Genesys Softphone VDI Adapter Install Package Directory>\windows** directory to open the **Genesys Installation Wizard**.
2. In the **Welcome to the Installation** window, click **Next**.

3. In the **Select Operating System** window, select **eLux**, specify the destination to install the installation package, and click **Next**.
4. In the **Ready to Install** window, select **Install**. The wizard installs Genesys Softphone VDI Adapter and displays the **Installation Status** window.
5. In the **Installation Complete** window, select **Finish**.
   The installation package installs the following items:
   - a Virtual Driver for Citrix shared object
   - a **libgsecurity** module
   - a startup script to update the Citrix **module.ini** config file.

   These files are packaged into an EPM/FPM pair, each with a separate signature file with four files for the VD package and three files with certificates used for signing:
   - **genesys_vd-<ip-version>.UC_RP5-1.0.fpm**
   - **genesysvd-<ip-version>.UC_RP5-1.0.epm**
   - **genesys_vd-<ip-version>.UC_RP5-1.0.fpm.sig**
   - **genesysvd-<ip-version>.UC_RP5-1.0.epm.sig**

   - **0-VeriSign-RootCA.cer**: VeriSign Universal Root Certification Authority
   - **1-Symantec-intermediate.cer**: Symantec Class 3 SHA256 Code Signing CA
   - **2-Genesys-codesign.cer**: Genesys certificate used for signing packages
6. Import the package files to the existing container and add them to the client image using the Unicon Scout Enterprise ELIAS tool:
   1. Using the **Security / Manage certificates** menu option, import the certificates as trusted.
   2. If the client is configured with **signature check**, the VeriSign Root CA certificate must be installed on each client in the **/setup/cacerts** folder.
   3. To add packages to the container, in ELIAS select the **Container / Import Package** menu option, then select the files with .the **epm** extension.
   4. To update the image definition file (IDF), open it in ELIAS, then add the new package by selecting **Genesys VD for Citrix, <ip-version>** in the right pane and press the **<==** button.
   5. Update the client workstation using the Scout Enterprise Console and perform these steps:
      - Check the firmware configuration of the relevant Thin Clients by selecting **Device configuration** and then choosing **Firmware**.
      - Update the device by selecting the **Commands / Update** option to initiate the update and force a device restart.

Installing the Genesys Softphone VDI Adapter in Silent mode

To install Genesys Softphone VDI Adapter in Silent mode, use the Installation Wizard **Silent** arguments as follows:

1. Update the **genesys_silent.ini** file by making the following modifications:
   - Specify if Genesys Softphone VDI Adapter should be installed for Windows ("citrix_windows") or eLux5 ("citrix_elux_5") by using the **DeploymentType** parameter. For example, **DeploymentType=citrix_windows**.
   - In case of installation on eLux5, add the path to the Genesys Softphone VDI Adapter directory using the **InstallPath** parameter. For example, **InstallPath=C:\GCTI\Genesys SoftphoneVDIAdapter**.
2. If you are *upgrading* Genesys Softphone VDI Adpater specify:
   - **IPVersion= <current version of Genesys Softphone VDI Adapter on this box (before upgrade)>**
   - **IPBuildNumber= <current build number of Genesys Softphone VDI Adapter on this box (before upgrade)>**
3. Execute the following command:
   `setup.exe /s /z"-s 'FullPathToGenesysSilentConfigurationFile' -sl 'FullPathToGenesysSilentResultFile'"` where:
   - `/s` specifies that the installation is running in InstallShield Silent Mode.
   - `/z` passes the Genesys Silent mode silent parameters to the installation.
   - `-s` specifies the full path to the silent configuration file. The **<Full path to Genesys Silent Configuration file>** is optional. If the **<Full path to Genesys Silent Configuration file>** parameter is not specified, the installation uses the **genesys_silent.ini** file in the same directory where the **setup.exe** is located.

     > **Important**
     > Enclose the value of the **<Full path to Genesys Silent Configuration file>** parameter by apostrophes (') if the parameter contains white symbols.

   - `-sl` specifies the full path to the installation results file. If the **<Full path to Genesys Installation Result file>** parameter is not specified, the installation creates the **genesys_install_result.log** file in the **<System TEMP folder>** directory.

     > **Important**
     > Enclose the value of the **<Full path to Genesys Installation Result file>** parameter in apostrophes (') if the parameter contains white space characters.

   The **InstallShield setup.exe** installation starter requires that:
   - There is *no* space between the `/z` argument and quotation mark. For example, `/z"-s"` is valid, while `/z "-s"` is not valid.
   - There *is* a space between the -s,-sl parameters and quotation mark. For example, `/z"-s c:\temp\genesys_silent.ini"` is valid, while `/z "-sc:\temp\genesys_silent.ini"` is not valid. For example,
     `setup.exe /s /z"-s 'C:\9.0.007.03\windows\b1\ip\genesys_silent.ini' -sl 'C:\GSP\silent_setup.log'"`.

4. After executing this command, verify that Genesys Softphone VDI Adpater is installed in the expected directory, and that the **silent_setup.log** file has been created in the **C:\GSP** directory.

Configuring Genesys Softphone

Genesys Softphone installation includes a configuration file (**<Genesys Softphone Directory>/Softphone.config>**) with configuration settings that are applied to the Softphone when it starts.

> ### Important
> You can make changes to the configuration file, but you must restart the Softphone before any of the changes take effect.

The configuration file is organized into *containers*. Each container is divided into *domains* that are further divided into *sections* that hold the *settings* for a group of parameters. The following configuration file examples describe the settings in each container:

For the description and valid values of each parameter, see Configuration Options Reference.

Basic container

The Basic container sets the Genesys Softphone user's DNs and the protocol used.

```
<Container name ="Basic">
    <Connectivity user ="DN0" server="Server0:Port0" protocol="Protocol"/>
    <Connectivity user ="DN1" server="Server1:Port1" protocol=" Protocol"/>
  </Container>
```

> ### Important
> If Single sign on is used with Agent Desktop, these parameters in configuration file are not taken in account.

Genesys container

The Genesys container sets the policy, endpoint, session, device, connector, codecs, proxy, mailbox, system, and security parameters.

Configuring the agent's DN

Set the following TServer section option for the DNs of the Place to which the agent is logging in:

- `sip-cti-control = talk,hold,dtmf`

> **Important**
> This option is mandatory to use third-party call control on the SIP device.

For information about configuring DN objects, see the Platform Administration help.

Configuring SIP Server

Genesys recommends setting the following SIP Server options:

- `dual-dialog-enabled=true` (default value)
- `make-call-rfc3725-flow=1` (allows for better and/or simpler codec negotiation)
- `ring-tone-on-make-call=true` (default value)
- `use-register-for-service-state=true`

For more information about these options, see the SIP Server Deployment Guide.

Suppressing the ringtone

The ringtone is generated for all incoming calls to Genesys Softphone. To suppress the ringtone for third-party call control for the originating DN, configure the following SIP Server option:

- `make-call-alert-info=<urn:alert:service:3pcc@genesys>`

or

- `make-call-alert-info=<file://null>;service=3pcc`

> **Important**
> If at least one Genesys Softphone in the contact center is configured with the `ringing_enabled` option set to 1, the SIP Server `make-call-alert-info` option should be set to one of the values specified above.

Single sign on with Agent Desktop

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

How to configure Genesys Softphone for single sign-on with Workspace Agent Desktop.

Genesys Softphone includes an HTTPS connector to simplify using Genesys Softphone with Agent Desktop:

- Single sign-on—Agent Desktop controls the SIP settings for Softphone based on explicit Agent Desktop centralized options and agent login credentials (Place and DN).
- Simplified deployment—each agent workstation runs the same application and configuration files, avoiding workstation specific configuration.
- Password authentication—Agent Desktop passes the DN password as one of the parameters through the Genesys Softphone connector to allow the Softphone to securely login to SIP Server and avoid the need for MPLS.

Configuring Softphone for Agent Desktop

The **softphone.config** configuration file contains a **connector** section in the **policy** domain. This section is populated automatically by the Installation Package based on the options specified in the wizard or in the silent installation configuration file.

```
<Container name ="Genesys">
...
        <domain name="policy">
        ...
                <section name="connector">

                <!-- Activates HTTP or HTTPS communication.
                Requires a port defined in the port option. -->
                  <setting name="protocol" value="https"/>

                  <!-- Specifies the port used when communicating in HTTP or HTTPS -->
                  <setting name="port" value="8000"/>

                  <!-- Activates the SESSIONID in cookies -->
                <setting name="enable_sessionid" value="1"/>

                <!-- Gives a thumbprint string value Workspace
                uses to select a certificate if the 'protocol' option
```

```
                is set to HTTPS. -->
                <setting name="certificate_search_value" value="55 20 C3 64 79 ED 13 87
FB C3 6C 99 54 D2 6C AC 11 61 BE CE"/>

                <!-- Specifies if the Softphone application is auto started
                or started by the client application.-->
                <setting name="standalone" value="1"/>

                </section>
        ...
        </domain>
...
</Container>
```

> **Important**
>
> You can enable Agent Desktop options for Softphone through Agent Setup.

Codec priority

Use the **enabled** section of the **codecs** domain in the **Softphone.config** configuration file to specify the order in which audio codecs are given priority.

For example:

```
<domain name="codecs">
   <section name="enabled">
     <setting name="audio" value="opus,pcmu,pcma,G722,iSAC/16000,G729"/>
   </section>
   <section name="PCMU/8000"/>
   <section name="PCMA/8000"/>
   <section name="G722/16000"/>
```

> **Warning**
>
> Any codec that is not explicitly included in the **enabled** section will not be used, even if the section for that codec is present in the configuration file.

To use the **enabled** section of the **codecs** domain, follow these guidelines:

- Codec names are *case-insensitive*. You can omit the clock rate portion of the section name unless needed to discriminate between two sections with the same name. The clock rate portion must be provided for **iSAC**.
- Specify codec parameters as a comma-separated list in parenthesis after an equals sign. You can use abbreviations such as "pt" for "payload_type".
- If there are codec conflicts, the value in the **enabled** section takes precedence over value in corresponding codec section, regardless of whether those values come from the configuration file. For example:

```
<setting name="audio" value="g729=(fmtp='annexb=no'),opus=(pt=125),pcmu,pcma"/>
<setting name="video" value="h264=(pt=120,fmtp='profile-level-id=420028')"/>
```

- If codec parameters are specified in-line (or a particular codec does not require any parameters, such as the PCMU and PCMA codecs), then a separate codec section is not necessary. In any case, codecs specified in the "enabled" section do not require presence of corresponding section to take effect.

Signing on with Agent Desktop

Before starting Agent Desktop, agents need to have Softphone running. Administrators can specify that Softphone starts automatically when the Windows user logs in or agents can startup Softphone

User interface and call controls

When using Softphone with Agent Desktop, Softphone disables its default user interface. Instead, agents can use the Agent Desktop user interface for call controls, mute, and volume control. For information on the Agent Desktop user interface, see the Agent Desktop Help.

Configuration options reference

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

Configuration settings by container and domain found in the Softphone.config file in the Genesys Softphone Installation Directory.

For an example of the configuration file, see Configuring Genesys Softphone.

The **Softphone.config** file is installed, along with **genesys_softphone.exe**, by either the **Genesys Installation Wizard** or silently by command line. The contents of the **Softphone.config** file is generated by the choices specified in the wizard or by modifications made to the **genesys_silent.ini** file.

In the **Softphone.config** file, the following attributes of the **Connector** section are set by **setup.exe**: `protocol`, `port`, and `certificate_search_value`, while `enable_sessionid`, `auto_restart` are not. The default value of these attributes are designed to address most business deployments. However, if you want to adjust their values, follow these steps to make a custom deployment:

1. Install Genesys Softphone on an administrator's machine.
2. Edit the **Softphone.config** file to change the values of the attributes in the **Connector** section.
3. Repackage Genesys Softphone with the custom **Softphone.config** file through an IT-controlled installation.
4. Push the custom package to the agent workstations.

Basic Container

The first container ("Basic") holds the basic connectivity details that are required to connect to your SIP Server. This container has at least one connection (Connectivity) element with the following attributes:

```
<Connectivity user="DN" server="SERVER:PORT" protocol="TRANSPORT"/>
```

If you are using a configuration that supports Disaster Recovery and Geo-Redundancy, there may be multiple connection elements present with each specifying a separate possible connection.

You must make the following changes and save the updated configuration file before using Genesys

Softphone:

- `user="DN"`: Supply a valid DN for the user attribute.
- `server="SERVER:PORT"`: Replace SERVER with the host name where your SIP Server is deployed, and PORT with the SIP port of the SIP Server host. The default SIP port value is 5060. For SRV resolution, specify the SRV record without including the port number in the server's URI. Also see SRV Resolution below.
- `protocol="TRANSPORT"`: Set the protocol attribute to reflect the protocol being used to communicate with SIP Server. Possible values are UDP, TCP, or TLS.

SRV resolution

When using an SRV record for the **server** parameter, note the following:

- Do not specify the port in the server URI.
- Genesys Softphone does not take into account the **weight** field of an SRV record.
- You cannot combine IPv4 and IPv6 for a single FQDN.
- The maximum number of targets (SRV records) per service is 20.
- You can only specify SRV records in the **server** parameter of the **Connectivity** element. You cannot use SRV records for the mailbox section or the **vq_report_collector** setting.

WebRTC

You will have to make the following changes and save the updated configuration file before using the Genesys Softphone:

- `user="DN"`—Supply a valid DN for the user attribute.
- `server="WEBRTCGATEWAY_SERVER:WEBRTCGATEWAY_PORT?sip-proxy-address="SIPPROXY_SERVER:SIPPROXY_PORT"`—Replace WEBRTCGATEWAY_SERVER with the host name where the WebRTC Gateway is deployed, and PORT with the HTTPS port of the WebRTC Gateway. Also, replace SIPPROXY_SERVER and SIPPROXY_PORT (optional) with the connectivity parameters of the SIP Proxy that need to be contacted by the WebRTC Gateway to register this DN.
- `protocol="TRANSPORT"`—Set the protocol attribute to reflect the protocol being used to communicate with the WebRTC Gateway: HTTPS.

> **Important**
> Your environment can have up to six SIP URIs (Connectivity sections) that represent six endpoint connections with SIP Server.

Domain	Section	Setting	Default Value	Description
	Connectivity	user		The first user's DN extension as configured in the configuration

Domain	Section	Setting	Default Value	Description
				database. Included in the SIP URI—for example, <sip:**DN0**@serverHostName0:por
		server		The SIP Server or Proxy location for the first user. Included in the SIP URI—for example, <sip:DN0@**serverHostName0:p**
		protocol		The transport protocol for the first user. For example, UDP, TCP, or TLS.

Genesys Container

The second Container ("Genesys") holds a number of configurable settings that are organized into domains and sections. You don't have to change these settings but you can customize them.

An overview of the settings in this container and the valid values for these settings is provided here:

Domain	Section	Setting
policy		
	endpoint	
		audio_qos
		include_os_version_in_user_agent_header
		include_sdk_version_in_user_agent_header
		ip_versions
		public_address
		include_mac_address
		refer_to_proxy
		rtp_inactivity_timeout
		rtp_port_min
		rtp_port_max
		tcp_port_min
		tcp_port_max
		signaling_qos
		sip_port_min
		sip_port_max
		sip_transaction_timeout
		video_max_bitrate
		video_qos
		vq_report_collector
		vq_report_publish
		vq_alarm_threshold
		webrtc_audio_layer
		answer_sdp_priority

Domain	Section	Setting
		sip_port_binding
		defer_device_release
	session	
		agc_mode
		auto_accept_video
		auto_answer
		auto_answer_delay
		dtmf_method
		echo_control
		noise_suppression
		dtx_mode
		reject_session_when_headset_na
		sip_code_when_headset_na
		vad_level
		ringback_enabled
		ringback_file
		ringing_enabled
		ringing_timeout
		ringing_file
		restart_audio_if_stuck
		reject_session_when_busy
		number_sessions_for_busy
		sip_code_when_busy
		rx_agc_mode
	device	

Domain	Section	Setting
		audio_in_device
		audio_out_device
		capture_device
		headset_name
		use_headset
codecs — See SIP Endpoint SDK for .NET 9.0.0NET—Working with Codec Priorities.		
proxies		
	proxy''	
		display_name
		domain
		password
		reg_interval
		reg_match_received_rport
		reg_timeout
		mailbox (sub-section of proxy)
		server
		timeout
		transport
		user
		nat (sub-section of proxy)
		ice_enabled
		stun_server
		stun_server_port

Domain	Section	Setting
		turn_password
		turn_relay_type
		turn_server
		turn_server_port
		turn_user_name
		system
	diagnostics	
		enable_logging
		log_file
		log_filter
		log_level
		log_options_provider
		log_options_endpoint
		logger_type
		log_segment
		log_expire
		log_time_convert
		log_time_format
	security	
		certificate
		tls_enabled
		tls-target-name-check
		use_srtp
	media	
		SPOptions#ringing_file

policy Domain

endpoint Section

audio_qos

Valid Values: Integer

Integer value representing the DSCP bits to set for RTP audio packets. **Note:** QoS is not supported for Windows Vista, Windows 7, or higher.

include_os_version_in_user_agent_header

Valid Values: `0`, `1`

Default Value: `1`

If set to `1`, the user agent field includes the OS version the client is currently running on.

include_sdk_version_in_user_agent_header

Valid Values: `0`, `1`

Default Value: `1`

If set to `1`, the user agent field includes the SDK version the client is currently running on.

ip_versions

Valid Values: `IPv4`, `IPv6`, `IPv4`, `IPv6`, `IPv6`, `IPv4`, or empty

Default Value: `IPv4`, `IPv6`

- IPv4—the application selects an available local IPv4 address; IPv6 addresses are ignored.
- IPv6—the application selects an available local IPv6 address; IPv4 addresses are ignored.
- IPv4,IPv6 or an empty—the application selects an IPv4 address if one exists. If not, an available IPv6 address is selected.
- IPv6,IPv4—the application selects an IPv6 address if one exists. If not, an available IPv4 address is selected.

Note: This parameter has no effect if the **public_address** option specifies an explicit IP address.

public_address

Valid Values: See description below

Default Value: Empty string which is fully equivalent to the `$auto` value

Local IP address or Fully Qualified Domain Name (FQDN) of the machine. This setting can be an explicit setting or a special value that the SDK uses to automatically obtain the public address.

Valid Values:

This setting may have one of the following explicit values:

- An IP address. For example, `192.168.16.123` for IPv4 or `FE80::0202:B3FF:FE1E:8329` for IPv6.
- A bare host name or fully qualified domain name (FQDN). For example, `epsipwin2` or `epsipwin2.us.example.com`.

This setting may have one of the following special values:

- $auto—The SDK selects the first valid IP address on the first network adapter that is active (status=up) and has the default gateway configured. IP family preference is specified by the policy.endpoint.ip_versions setting.
- $ipv4 or $ipv6—Same behavior as the $auto setting but the SDK restricts the address to a particular IP family.
- $host—The SDK retrieves the standard host name for the local computer using the gethostname system function.
- $fqdn—The SDK retrieves the fully qualified DNS name of the local computer. The SDK uses the GetComputerNameEx function with parameter ComputerNameDnsFullyQualified.
- An adapter name or part of an adapter name prefixed with `$`. For example, `$Local Area Connection 2` or `$Local`. The specified name must be different from the special values `$auto`, `$ipv4`, `$host`, and `$fqdn`.
  If the value is an explicit host name, FQDN, or $fqdn, the Contact header includes the host name or FQDN for the recipient of SIP messages (SIP Server or SIP proxy) to resolve on their own. For all other cases, including $host, the resolved IP address is used for Contact. The value in SDP is always the IP address.
- $net:subnet - The SDK will select the IP address matching the given network (from any local interface). The *subnet* is the full CIDR name as per RFC 4632. For example, `$net:192.168.0.0/16`.

include_mac_address

Valid Values: `0`, `1`

Default Value: `0`

If set to 1, the MAC address is included in the Contact header of the REGISTER message of the host's network interface in a format compatible with RFC 5626.

refer_to_proxy

Valid Values: `0`, `1`

Default Value: `0`

Specifies the destination of a referred INVITE.

- `0`—Send the INVITE to the URL specified in the Refer-To header of the REFER message.
- `1`—Send the INVITE to your configured SIP Proxy.

rtp_inactivity_timeout

Valid Values: `5-150`

Default Value: `150`

Suggested Value: `30`

Timeout interval in seconds for RTP inactivity.

rtp_port_min

Valid Values: `9000-65535`

The integer value representing the minimum value for an RTP port range. Must be within the valid port range of `9000` to 65535. If the minimum and maximum values are not specified or are set to an invalid value, the default minimum (9000) and maximum (minimum value + 999) are used. Setting the minimum to a value that is larger than the maximum is considered an error and will result in a failure to initialize the endpoint.

rtp_port_max

Valid Values: `9000-65535`

The integer value representing the maximum value for an RTP port range. Must be within the valid port range of `9000` to 65535. If the minimum and maximum values are not specified or are set to an invalid value, the default minimum (9000) and maximum (minimum value + 999) are used. Setting the maximum to a value that is less than the minimum is considered an error and will result in a failure to initialize the endpoint.

tcp_port_min

Valid Values: `0-65535`

The integer value representing the minimum value for a TCP client-side port range. Must be within the valid port range of 1 to 65535. If set to `0` (default) or if the configured range is not valid, SIP connections over TCP and TLS use ephemeral ports, assigned by the operating system.

tcp_port_max

Valid Values: `0-65535`

The integer value representing the maximum value for a TCP client-side port range. Must be within the valid port range of 1 to 65535. If set to `0` (default) or if the configured range is not valid, SIP connections over TCP and TLS use ephemeral ports, assigned by the operating system.

If the value is non-zero and greater than the *tcp_port_min* value, this value specifies the maximum value for a TCP client-side SIP port range that will be used for all outgoing SIP connections over TCP and TLS transport.

signaling_qos

Valid Values: Integer

The integer value representing the DSCP bits to set for SIP packets. **Note:** QoS is not supported for Windows Vista, Windows 7, or higher.

sip_port_min

Valid Values: `1-65535`

The integer value representing the minimum value for a SIP port range. Must be within the valid port range of 1 to 65535. If the minimum and maximum values are not specified or are set to an invalid value, the default minimum (5060) and maximum (minimum value + 6) are used. Setting the minimum to a value that is larger than the maximum is considered an error and will result in a failure to initialize the endpoint.

sip_port_max

Valid Values: `1-65535`

The integer value representing the maximum value for a SIP port range. Must be within the valid port range of 1 to 65535. If the minimum and maximum values are not specified or are set to an invalid value, the default minimum (5060) and maximum (minimum value + 6) are used. Setting the maximum to a value that is less than the minimum is considered an error and will result in a failure to initialize the endpoint.

sip_transaction_timeout

Valid Values: `1-32000`

Default Value: `4000`

SIP transaction timeout value in milliseconds. Valid values are 1 through 32000, with a default value of 4000. The recommended value is 4000.

video_max_bitrate

Valid Values: Integer

Integer value representing the maximum video bitrate.

video_qos

Valid Values: Integer

The integer value representing the DSCP bits to set for RTP Video packets. **Note:** QoS is not supported for Windows Vista, Windows 7, or higher.

vq_report_collector

See SIP Endpoint SDK for .NET—Producing RTCP Extended Reports.

vq_report_publish

See SIP Endpoint SDK for .NET—Producing RTCP Extended Reports.

vq_alarm_threshold

Valid Values: `0` or a number from `1.0` to `5.0`

Default Value: `0`

Specifies the MOS threshold for generating Voice Quality Alarms. A `0` value disables the alarms. The recommended threshold value is `3.5`. Genesys recommends that you avoid using values above `4.2` as an MOS that high might not be obtainable with some codecs, even in perfect network conditions.

webrtc_audio_layer

Valid Values: `0`, `1`, `2`, `1000`, `2000`, `3000`

Default Value: `0`

Specifies which audio layer is used for WebRTC.

- `0` — The audio layer is defined by the `GCTI_AUDIO_LAYER` environment variable — Core audio is used if this environment variable is not specified.
- `1` — Wave audio layer is used.
- `2` — Core audio layer is used.
- `1000` — Instructs the audio layer to open the microphone channel when the endpoint starts up, using the audio layer type defined by option `0`, and to keep it open until the endpoint is terminated.
- `2000` — Opens the speaker channel for the life of the endpoint, using the audio layer type defined by

option `0`. Eliminates any delay in opening the audio device when an incoming or outgoing call is connected, for example in environments where audio device startup is slow due to a required restart of the Windows MMCSS service.

- `3000` — Opens the microphone and speaker channels for the life of the endpoint, using the audio layer type defined by option `0`.

> **Important**
>
> Keeping the audio channels permanently open eliminates any delay in connecting audio device to the call works around any issues with device occasionally not starting (or stopping) properly, at the cost of very small performance penalty.

answer_sdp_priority

Valid Values: `config`, `offer`

Default Value: `config`

- `config`—the endpoint selects the first codec from the codec configuration listed in both the codec configuration and the SDP offer.
- `offer`—the endpoint selects the first codec in the SDP offer listed in both the codec configuration and the SDP offer.

sip_port_binding

Valid Values: `0`, `1`

Default Value: `0`

- `0`—open the SIP port to listen on any interface.
- `1`—the SIP port binds to the interface specified by the public_address setting and listens only on this IP address.

defer_device_release

Valid Values: Any integer

Default Value: `200`

If set to a non-zero value, releasing of audio devices will be deferred for a given time (in milliseconds) after the audio stream has been stopped, to avoid any potential service interruptions when the audio is going to be quickly restarted, and if audio device operations are too slow on the user workstation or have other problems with restart. A zero value disables the deferred device release.

session Section

agc_mode

Valid Values: `0`, `1`

Default Value: `1`

If set to 0, AGC (Automatic Gain Control) is disabled; if set to 1, it is enabled. Other values are reserved for future extensions. This configuration is applied at startup, after which time the **agc_mode** setting can be changed to 1 or 0 from the main sample application.

Note: It is not possible to apply different AGC settings for different channels in multi-channel scenarios.

auto_accept_video

Valid Values: `0`, `1`

This setting is only used in auto-answer scenarios when `auto_answer=1`.

If auto_accept_video is set to 1, both audio and video streams are accepted, otherwise incoming calls are answered as audio only, even if video is present in the offer.

auto_accept_video applies to a 3pcc answer when make-callrfc3275 is configured to 1 on the originating DN and a video codec is configured in the endpoint. auto_accept_video is not applied to a 3pcc answer when make-call-rfc3275 is configured to 2 on an originating DN, even if auto_accept_video is set to 1 and a video codec is configured in the endpoint.

auto_answer

Valid Values: `0`, `1`

If set to 1, all incoming calls should be answered automatically.

auto_answer_delay

Valid Values: Number in milliseconds

Default Value: `1500`

Time in milliseconds to wait before auto-answering. The recommended and default value is 1500 milliseconds.

dtmf_method

Valid Values: `Rfc2833`, `Info`, `InbandRtp`

Method to send DTMF.

echo_control

Valid Values: `0`, `1`

If set to `1`, echo control is enabled.

noise_suppression

Valid Values: `0`, `1`

If set to `1`, noise suppresion is enabled.

dtx_mode

Valid Values: `0`, `1`

If set to `1`, DTX is activated.

reject_session_when_headset_na

Valid Values: `0`, `1`

If set to `1`, the SDK should reject the incoming session if a USB headset is not available.

sip_code_when_headset_na

Valid Values: SIP Error Code

Default Value: `480`

If a valid SIP error code is supplied, the SDK rejects the incoming session with the specified SIP error code if a USB headset is not available.

vad_level

Valid Values: `0-3`

Sets the degree of bandwidth reduction, from `0` for conventional VAD to `3` for aggressive high.

ringback_enabled

Valid Values: `0`, `1`, `2`, `3`, `4`, `6`

Default Value: `2`

Specifies whether the ringback tone is enabled for outgoing calls.

- `0` — The ringback is not played when the INVITE dialog is not yet established. In scenarios where ringback is provided by Media Server, the ringback tone would be still present.
- 1 — The incoming media stream is played if provided by the Media gateway in a reliable provisional response with SDP.
- 2 — A local file is used for the ringback.
- 3 — The ringback is always played using either a local file or media provided by the gateway, if the provisional response is reliable.
- 4 — Same as 1, but the incoming media stream is played even if the provisional response from Media gateway is not reliable.
- 6 — The ringback is always played using either a local file or media provided by the gateway (regardless of whether the provisional response is reliable or not).

ringback_file

Valid Values: Empty or the path to the ringback sound file. The path can be a file in the current directory or the full path to the sound file.

Default Value: Empty

Specifies the audio file that is played when the ringing tone is enabled with the `ringing_enabled` option.

WebRTC does not support MP3 playback. The ringtone file for built-in ringback must be a RIFF (little-endian) WAVE file using one of the following formats:

- kWavFormatPcm = 1, PCM, each sample of size bytes_per_sample
- kWavFormatALaw = 6, 8-bit ITU-T G.711 A-law (8 KHz sampling rate)
- kWavFormatMuLaw = 7, 8-bit ITU-T G.711 mu-law (8 KHz sampling rate)

Uncompressed PCM audio must be 16-bit mono or stereo with a sampling rate of 8, 16, or 32 KHz.

ringing_enabled

Valid Values:

- `0`: None, disable ringtone.
- 1: (default) Play ringtone through system default device only. Configure media in system.media.ringing_file.
- 2: Play ringtone through communication device (headset) only. Configure media in policy.session.ringing_file.
- 3: Play ringtone through both devices at the same time (the combination of values 1 and 2).
- 4: Play ringtone through a separate ringer device, specified by policy.device.ringer_device.
- 5: Play ringtone through system default device and lay ringtone through a separate ringer device (the combination of values 1 and 4).
- 6: Play ringtone through the communication device (headset) once only for the full duration

(policy.session.ringing_timeout is ignored, and ringing does not stop when the call is answered). Configure media in policy.session.ringing_file.

- 7: Play ringtone once for the full duration through both system default device and communication device (headset) (policy.session.ringing_timeout is ignored, and ringing does not stop when call is answered). Configure media in system.media.ringing_file and policy.session.ringing_file.

Default Value: 1

Specifies whether to enable the ringtone and on which device to play the media file. This option applies to both calls that are auto-answered by Softphone or by other applications such as Workspace Web Edition and by calls that are manually answered by an agent.

Suppressing the Ringtone

The ringtone is generated for all incoming calls to the Genesys SIP Endpoint SDK. To suppress the ringtone for third-party call control for the originating DN, configure the following SIP Server option:

- `make-call-alert-info=<urn:alert:service:3pcc@genesys></urn:alert:service:3pcc@genesys>`

or

- `make-call-alert-info=;service=3pcc`

> **Important**
> If at least one application based on SIP Endpoint in the contact center is configured with the `ringing_enabled` option set to a non-zero value, the SIP Server `make-call-alert-info` option should be set to one of the specified values.

ringing_timeout

Valid Values: Empty, `0`, or a positive number

Default Value: `0`

Specifies the duration, in seconds, of the ringing tone. If set to `0` or if the value is empty, the ringing time is unlimited.

ringing_file

Valid Values: Empty or the path to the ringing sound file. The path may be a file name in the current directory or the full path to the sound file.

Default Value: ringing.wav

Specifies the audio file that is played when the ringing tone is enabled with the `ringing_enabled` option.

Note that WebRTC does not support MP3 playback. The ringtone file for built-in ringing should be a RIFF (little-endian) WAVE file using one of the following formats:

- kWavFormatPcm = 1, PCM, each sample of size bytes_per_sample
- kWavFormatALaw = 6, 8-bit ITU-T G.711 A-law
- kWavFormatMuLaw = 7, 8-bit ITU-T G.711 mu-law

Uncompressed PCM audio must 16 bit mono or stereo and have a frequency of 8, 16, or 32 KHZ.

restart_audio_if_stuck

Valid Values: Empty, `0`, `1`

Default Value: `0`

- `0` or Empty—disable auto restart for stuck audio
- `1`—enable auto restart for stuck audio

reject_session_when_busy

Valid Values: Empty, `0`, `1`

Default Value: `0`

- `0` or Empty—disable rejection of a session when busy
- `1`—enable rejection of a session when busy

number_sessions_for_busy

Valid Values: Positive integer

Default Value: 1

Sets the number of sessions before busy. Must be a positive integer.

sip_code_when_busy

Valid Values: Empty, 4xx, 5xx, 6xx

Default value: Empty

SIP error response code to use when busy. Can be set to any valid SIP error response code in the 4xx, 5xx, or 6xx range, for example, 486.

rx_agc_mode

Valid Values: `0`, `1`

Default value: `0`

When set to `1`, the SDK enables the receiving-side AGC allowing the volume of the received RTP stream to be adjusted automatically. When set to `0` (default), the feature is disabled.

device Section

audio_in_device

Valid Values: A regex that matches the ECMAScript standard.

Microphone device name.

audio_out_device

Valid Values: A regex that matches the ECMAScript standard.

Speaker device name.

capture_device

Valid Values: A regex that matches the ECMAScript standard.

Capture device name.

headset_name

Valid Values: A regex that matches the ECMAScript standard.

The name of the headset model.

ringer_device

Valid Values: A valid ringer device name: can be either the device proper name or a regular expression. This option is applicable when ringing_enabled = 4

use_headset

Valid Values: `0`, `1`

If set to `0`, the audio devices specified in audio_in_device and audio_out_device are used by the SDK.

If set to 1, the SDK uses a headset as the preferred audio input and output device and the audio devices specified in audio_in_device and audio_out_device are ignored.

codecs Domain

See SIP Endpoint SDK for .NET 9.0.0NET—Working with Codec Priorities.

proxies Domain

Configure a proxy section for each connectivity line. For example, for three connectivity lines, configure sections for `proxy0`, `proxy1`, and `proxy2`.

When the proxy section does not exist in the configuration file for a particular connectivity line, the framework takes the configurations settings from the `proxy0` section. You can use this feature in use cases where the proxy sections are the same for all connectivity lines.

proxy Section

display_name

Valid Values: String

Proxy display name.

domain

Valid Values:Any valid SIP domain

Defalut Value: Empty

A SIP domain is an application layer configuration defining the management domain of a SIP proxy. The configured value should include hostport and may include uri-parameters as defined by RFC 3261. The scheme, userinfo, and transport URI parameters are included automatically.

If set to an empty string, SIP Endpoint SDK for .NET uses the parameters from the Connectivity section to construct the SIP domain value as it did in previous versions.

password

Valid Values: String

Proxy password.

reg_interval

Valid Values: Integer

Default Value: `0`

The period, in seconds, after which the endpoint starts a new registration cycle when a SIP proxy is down. Valid values are integers greater than or equal to `0`. If the setting is empty or negative, the default value is 0, which means no new registration cycle is allowed. If the setting is greater than 0, a new registration cycle is allowed and will start after the period specified.

reg_match_received_rport

Valid Values: `0` or `1`

Default Value: `0`

This setting controls whether or not SIP Endpoint SDK should re-register itself when receiving an IP address (in the `received` parameter of a REGISTER response) that is different from the address supplied in the Contact header and does not match any local network interfaces. A value of `0` (default) disables this feature and a value of 1 enables re-registration.

Starting from 9.0.003, this setting is deprecated and is not recommended for use, unless suggested by Genesys Technical Support to fix specific problems. When the `received` parameter of a REGISTER response matches a local IP address, changing the IP address and re-registering is now done automatically.

reg_timeout

Valid Values: Number in seconds

The period, in seconds, after which registration should expire. A new REGISTER request will be sent before expiration. Valid values are integers greater than or equal to 0. If the setting is `0` or empty/null, then registration is disabled, putting the endpoint in standalone mode.

mailbox Sub-section

> **Important**
> **mailbox** is a sub-section of the **proxy<n>** section.

password

Valid Values: String

Mailbox password.

server

Valid Values: String

Proxy server address and port for this mailbox.

timeout

Valid Values: Number in seconds

Default Value: `1800`

Subscription expiration timeout in seconds. If the setting is missing or set to `0`, the SDK uses a default timeout of `1800` seconds (30 minutes).

transport

Valid Values: `udp`, `tcp`, `tls`

Transport protocol to use when communicating with the server.

user

Valid Values: String

Mailbox ID for this mailbox.

nat Sub-section

> **Important**
> **nat** is a sub-section of the **proxy<n>** section.

ice_enabled

Valid Values: Boolean

Enable or disable ICE.

stun_server

Valid Values: String

STUN server address. An empty or null value indicates this feature is not used.

stun_server_port

Valid Values: Valid port number

Default Value: 3478

STUN server port value.

turn_password

Valid Values: String

Password for TURN authentication.

> **Warning**
> Starting from 9.0.012.02, this setting is deprecated and is not recommended for use, unless suggested by Genesys Technical Support to fix specific problems. Use the GCTI_TURN_PASSWORD environment variable to set the password for TURN authentication.

turn_relay_type

Valid Values: `0`, `udp`, `1`, or `tcp`

Type of TURN relay.

- `0` or `udp` for TURN over UDP.
- `1` or `tcp` for TURN over TCP.

turn_server

Valid Values: String

TURN server address. An empty or null value indicates this feature is not used.

turn_server_port

Valid Values:Valid port number

Default Value: 3478

TURN server port value.

turn_user_name

Valid Values: String

User ID for TURN authorization

> **Warning**
> Starting from 9.0.012.02, this setting is deprecated and is not recommended for use, unless suggested by Genesys Technical Support to fix specific problems. Use the GCTI_TURN_USERNAME environment variable to set the username for TURN authentication.

`system` Domain

diagnostics Section

enable_logging

Valid Values: `0` or `1`

Default Value: `1`

Disable or enable logging.

log_file

Valid Values: String

Log file name, for example, `SipEndpoint.log`.

log_filter

Valid Values: *Empty*, `dtmf`

Default Value: *Empty*

Specifies the list of log filters to be applied to hide sensitive data from the endpoint log. Currently the only supported filter is `dtmf`, which hides all occurrences of DTMF data from the log (by replacing entered digits with 'x').

log_level

Valid Values: `0-4`

Default Value: 3

Log levels: `0` = "Fatal"; 1 = "Error"; 2 = "Warning"; 3 = "Info"; 4 = "Debug"

log_options_provider

Valid Values: Valid values for webrtc, `warning`, `state`, `api`, `debug`, `info`, `error`, `critical`

Example value: `gsip=2, webrtc=(error,critical)`

log_options_endpoint

Valid Values: `0-4`, same as **log_level**

Default Value: 2

Log levels: `0` = "Fatal"; 1 = "Error"; 2 = "Warning"; 3 = "Info"; 4 = "Debug"

5 = Logging disabled.

This setting should not be set higher than log_level setting.

logger_type

Valid Values: file

If set to `file` the log data will be printed to the file specified by the **log_file** value.

log_segment

Valid Values: `false`, number, or number in KB,MB, or hr

Default Value: `10 MB`

- false: No segmentation is allowed
- or KB: Size in kilobytes
- MB: Size in megabytes
- hr: Number of hours for segment to stay open

Specifies the segmentation limit for a log file. If the current log segment exceeds the size set by this option, the file is closed and a new one is created. This option is ignored if log output is not configured to be sent to a logfile.

log_expire

Valid Values: `false`, number, number `file`, number day

Deafult Value: `10` (store 10 log fragments and purge the rest)

- false: No expiration; all generated segments are stored.
- or file: Sets the maximum number of log files to store. Specify a number from 1—1000.
- day: Sets the maximum number of days before log files are deleted. Specify a number from 1—100

Determines whether log files expire. If they do, sets the measurement for determining when they expire, along with the maximum number of files (segments) or days before the files are removed. This option is ignored if log output is not configured to be sent to a log file.

log_time_convert

Valid Values: `local`, `utc`

Default Value: `local`

- `local`: The time of log record generation is expressed as a local time, based on the time zone and any seasonal adjustments. Time zone information of the application's host computer is used.
- `utc`: The time of log record generation is expressed as Coordinated Universal Time (UTC).

Specifies the system in which an application calculates the log record time when generating a log file. The time is converted from the time in seconds since the Epoch (00:00:00 UTC, January 1, 1970).

log_time_format

Valid Values: `time`, `locale`, `ISO8601`

Default Value: time

- `time`: The time string is formatted according to the HH:MM:SS.sss (hours, minutes, seconds, and milliseconds) format
- `locale`: The time string is formatted according to the system's locale.
- `ISO8601`: The date in the time string is formatted according to the ISO 8601 format. Fractional seconds are given in milliseconds.

Specifies how to represent, in a log file, the time when an application generates log records. A log record's time field in the ISO 8601 format looks like this: 2001-07-24T04:58:10.123.

security Section

> **Important**
> SIP Endpoint SDK no longer uses the **tls_enabled** setting.

certificate

Valid Values: String

Thumbprint value of the Public endpoint certificate file which is used as a client-side certificate for outgoing TLS connections and server-side certificate for incoming TLS connections. For example, `78 44 34 36 7a c2 22 48 bd 5c 76 6b 00 84 5d 66 83 f5 85 d5`

This option replaces the **cert_file** option from previous versions. For backwards compatibility, the SDK accepts **certificate** or **cert_file**.

tls-target-name-check

Valid Values: `no`, `host`

Default Value: `host`

Specifies if the Common Name in the subject field and/or the Subject Alternate Names of the server's certificate will be compared to the target host name (option value `host`). If they are not identical, the connection fails. If the option is set to `no`, a comparison is not made, and the connection is allowed.

use_srtp

Valid Values: `optional`, `allowed`, `disabled`, `off`, `elective`, `both`, `enabled`, `force`, `mandatory`

Indicates whether to use SRTP:

- `optional` or `allowed`—do not send secure offers, but accept them
- `disabled` or `off`—do not send secure offers and reject incoming secure offers
- `elective` or `both`—send both secure and non-secure offers and accept either
- `enabled`—send secure offers, accept both secure and non-secure offers
- `force` or `mandatory`—send secure offers, reject incoming non-secure offers

Adding either `',UNENCRYPTED_SRTCP'` (long form) or `',UEC'` (short form) to any value (for example, "*enabled,UEC*""), would result in the **UNENCRYPTED_SRTCP** parameter being added to that offer. When this parameter is negotiated, RTCP packets are not encrypted, but are still authenticated.

media Section

ringing_file

Valid Values: Empty, String file name

Default Value: `ringing.mp3`

The Ringing sound file name in the current directory or the full local path to the ringing sound file.

Audio device settings

This section describes how to set up Genesys Softphone to work with your audio devices, such as headsets.

Genesys Softphone uses the following criteria to select its audio input and output devices:

- Basic settings for audio input and output devices.
- Selection rules used to choose an audio device, auto-answer a call, and reject a call.
- Combinations of settings that affect audio device selection, auto-answer, and call rejection.

Basic settings

Use the following parameters to configure headsets and other audio input devices:

- **headset_name**
- **audio_in_device**
- **audio_out_device**

If none of the audio devices that are accessible to the endpoint, match the device names in the configuration file; Genesys Softphone picks up the first available devices from the WebRTC list for audio devices.

> **Tip**
>
> The **headset_name**, **audio_in_device**, and **audio_out_device** options support both device proper names and regular expressions.

Audio device selection rules

The following rules are used to select an audio device, auto-answer a call, and reject a call.

The following audio device selection procedure is applied on startup and every time any changes are made to device presence (such as when a new device is plugged in or an existing device is removed):

1. The first device in the applicable list that is present in the system is selected when possible. This device (or devices) will either be specified by the **headset_name** parameter or by the **audio_in_device** and **audio_out_device** parameters, depending on whether the **use_headset** parameter has been enabled.
2. If none of the configured devices are present (or if the configuration list is empty), then Genesys Softphone selects the audio devices using the priority provided by WebRTC, based on the order of the available devices in its device list.

Auto-answer

When either of the following conditions is met, the SDK blocks the auto-answer functionality (a policy of **should answer** returns the value **unknown**; a manual answer is still possible):

- the **use_headset** parameter is set to **1**, and none of the devices listed in the **headset_name** settings are present (but session rejection is not applicable, that is, the **reject_session_when_headset_na** parameter has been set to **0**).
- Genesys Softphone was unable to find any usable microphone or speaker device (applicable to cases where the **use_headset** parameter is set to **0**).

Finally, if the **auto_answer** parameter is set to 1 and the auto-answer functionality is not blocked (and the call was not already rejected), Genesys Softphone answers the incoming call automatically (the **should answer** policy returns the value **true**).

Rejecting a call

For backward compatibility with previous releases, a call can only be rejected when both of the following conditions are met (a policy of **should answer** returns the value **false**):

- Both the **use_headset** and **reject_session_when_headset_na** parameters are set to 1.
- None of the devices listed in the **headset_name** settings is present on the workstation.

When these conditions are both met, an incoming call is rejected with the SIP response code that is configured in the **sip_code_when_headset_na** setting. If the setting is missing or the value is not in the valid range of **400** to **699**, then the default value of **480** (**Temporarily Unavailable**) is used.

In addition, when these conditions are met, Genesys Softphone refuses to initiate any new calls; it rejects all outgoing call attempts.

The availability of a fallback device (selected by Step 2 in the Audio device selection section) does not affect call rejection.

Audio setting combinations

Sometimes combinations of settings that you make can have unexpected results. Before adjusting your settings, review this section. The following combinations of settings affect audio device selection, auto-answer, and call rejection in the ways described below:

use_headset=1

Headset is available Genesys Softphone considers a headset to be available if a headset is found by name in the list of headset names stored in the **headset_name** parameter. (The highest priority device in the list is selected). Outgoing calls can be initiated.	**auto_answer=1**	Incoming calls are answered automatically.
	auto_answer=0	Incoming calls are answered manually.
Headset is not available Genesys Softphone determines that no headset is available if a headset is not found by name in the list of headset names stored in the **headset_name** parameter. An audio device is still assigned if any supported devices are present in the system, using the first available audio input and output devices from the list compiled by WebRTC.	No auto-answer is possible in this subcase, so the **auto_answer** setting is not used.	**reject_session_when_headset_na=1** • Incoming calls are automatically rejected. • Outgoing calls are blocked.
		reject_session_when_headset_na=0 • Incoming calls can be answered manually. It is assumed that the agent will plug in the headset (or use an available non-headset device, if applicable) before answering the call. • Outgoing calls can be initiated. It is the agent's responsibility to ensure that the appropriate audio devices are available before the call is answered by the remote side.

use_headset=0

Audio devices are configured using the names from the **audio_in_device** and **audio_out_device** settings. Genesys Softphone selects the highest-priority input and output devices from that list or, if no valid devices are found in that list, from the first available devices in the list compiled by WebRTC. Outgoing calls can be initiated.

Both microphone and speaker are available	**auto_answer=1**	Incoming calls are answered automatically.
	auto_answer=0	Incoming calls are answered manually.
Either microphone or speaker is not available • Incoming calls can be answered manually. It is assumed that the agent will plug in the headset (or use an	No auto-answer is possible in this subcase, so the **auto_answer** setting is not used.	Auto-rejection is not applicable, so the **reject_session_when_headset_na** setting is not used.

available non-headset device, if applicable) before answering the call. • Outgoing calls can be initiated. It is the agent's responsibility to ensure that the appropriate audio devices are available before the call is answered by the remote side.		

Gplus Adapter for Salesforce Lightning (v9)

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Gplus Adapter for Salesforce Lightning (v9).

Genesys Engage cloud offers an integrated Agent Desktop experience within the Salesforce Lightning environment to handle Genesys contact center interactions. The integrated solution presents a complete customer view allowing your contact center agents to service your customers.

See the Web Services and Applications topic in the *Supported Operating Environment Reference* for a list of the browsers supported for Gplus Adapter.

> **Important**
> The Gplus Adapter URL in Salesforce Call Center follows this format: `https://<your company name>/ui/crm-workspace/index.html`

How do I setup Gplus Adapter in Salesforce Lightning?

You can setup Gplus Adapter by following the procedures described in Integrating Gplus Adapter in Salesforce.

How do I migrate Gplus Adapter from an earlier version?

Follow this procedure to migrate your existing Gplus Adapter 8.5 to Gplus Adapter 9.0 for Salesforce Lightning.

How do I configure Gplus Adapter functionality?

First set up corresponding Salesforce objects and configurations, then enable Gplus options in Agent Setup.

You can configure the following Salesforce functionality:

- SSO
- Click-to-dial
- Screen pop
- Activity History

How do I access Gplus Adapter?

You can access Gplus Adapter by logging into Salesforce and clicking the phone icon at the bottom-left corner of the window. You will see a pop-up window where you can login with your Genesys credentials and start using the complete set of Genesys contact center functionalities.

Integrating Gplus Adapter

Important

This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Integrating Gplus Adapter.

Important

1. Ensure that Agent Desktop release 9 has been enabled in your environment by confirming that the Agent Desktop 9 application icon is displayed in your Genesys Portal.
2. In the Portal, click the **Agent Desktop** icon to launch Agent Desktop and then copy the domain name from the Agent Desktop URL to build Gplus Adapter URL. For example, copy `gwa-<region>.genesyscloud.com` from the Agent Desktop URL. The domain in the Agent Desktop URL should be used in Gplus Adapter URL.
3. The Adapter URL will look like this:

   ```
   https://gwa-<region>.genesyscloud.com/ui/crm-workspace/index.html
   ```

Important

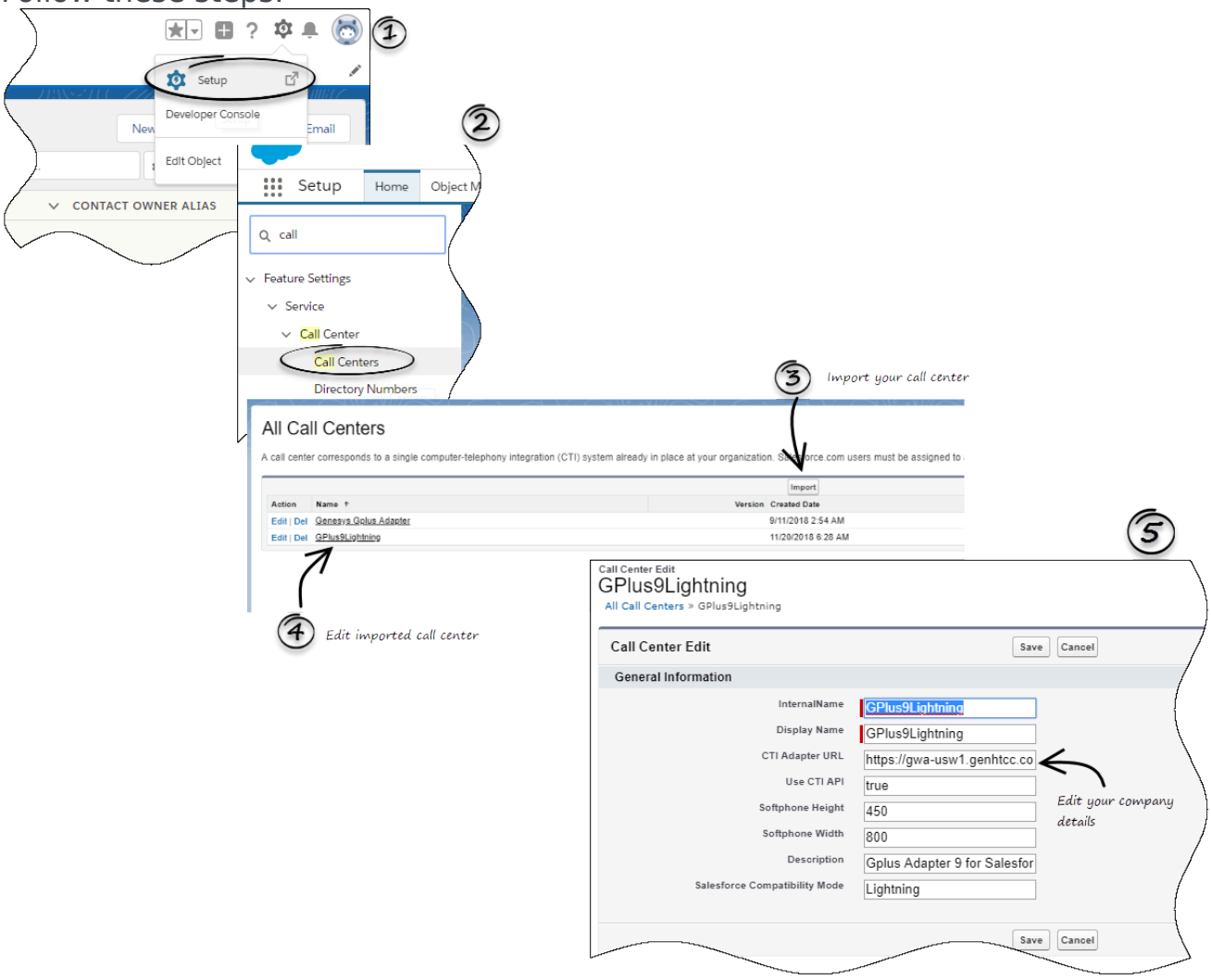
If you are migrating from Adapter 8.5, you must first follow the migration procedure in this article.

Here's a summary of the steps you need to take to setup and access Gplus Adapter in Salesforce.

1. Enable **Lightning** in your Salesforce environment.
2. Integrate Gplus Adapter in Salesforce Lightning.
3. Add users to your Call Center.
4. Configure the Utility bar for Gplus Adapter for the agents to easily access the application.
5. Configure Gplus Adapter SSO.

Integrating Gplus Adapter in Salesforce Lightning

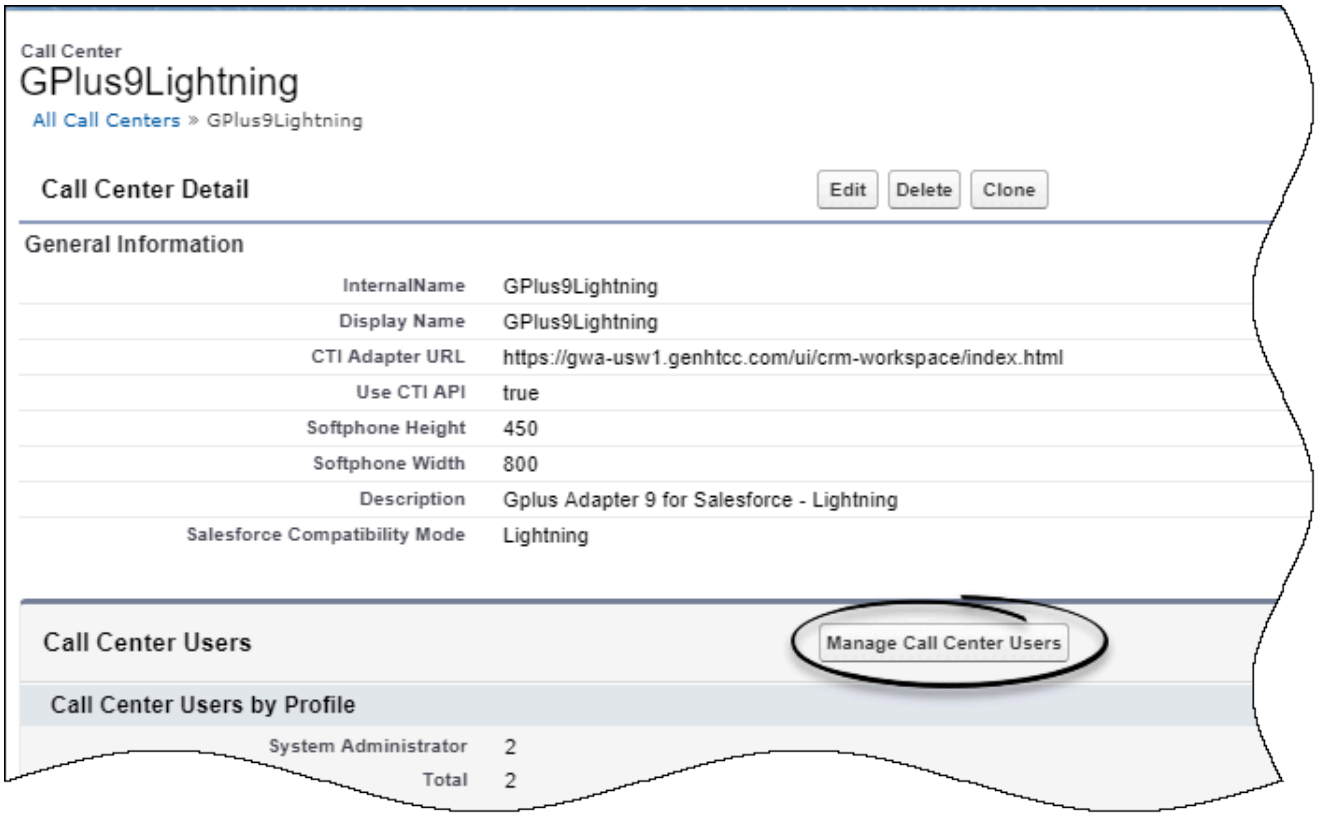
Follow these steps:



1. Click the gear icon in the top right corner and then click **Setup**.
2. Using the **Quick Find** field, search for and access the **Call Centers** setup page.
3. Using the **Import** functionality, import the lightning-callcenter.xml file from your computer. If you have not already downloaded the file, download it from here.
4. From the **All Call Centers** list, click the Call Center you just imported. For example, **GPlus9Lightning**.
5. Take the Adapter URL that you built at the beginning of this article and enter it in the **CTI Adapter URL** field. The domain name can be in one of two formats. If you do not have a unique tenant, the domain name includes a region:
   `https://gwa-<region>.genesyscloud.com/ui/crm-workspace/index.html`.
   If you have a unique tenant, your URL will include your company name:
   `https://<your company name>.genesyscloud.com/ui/crm-workspace/index.html`.

6. Save the changes.

Adding users to Call Center

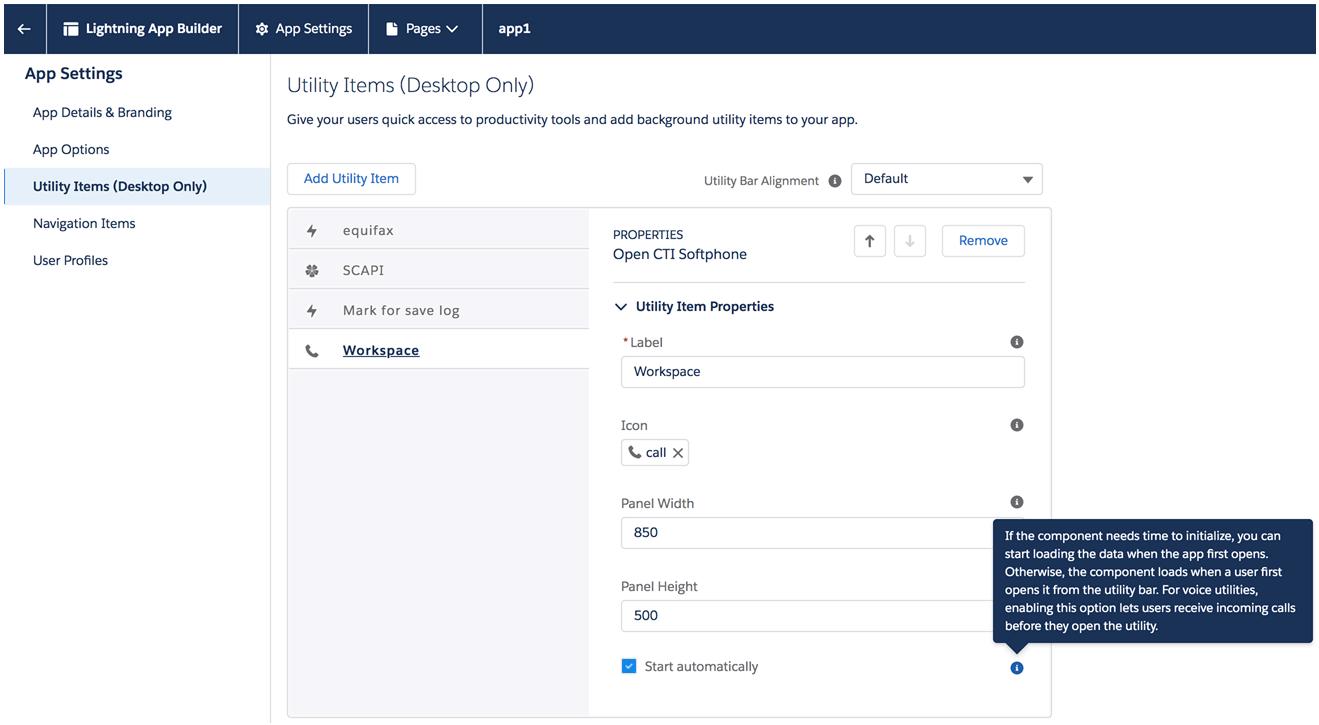


The next step after you setup your Adapter is to add users to your call center. You must add at least one user to your call center.

1. In the **Call Centers** setup page, click **Manage Call Center Users**.
2. Click **Add More Users**.
3. Search the interface to find the users you want to add to Gplus Adapter.
4. Select the users you want to add and click **Add to Call Center**.

> ### Tip
> From the **Manage Users** page, select the users you want to remove from a call center and click **Remove Users**.

Configuring the Utility bar



1. In the setup page, using the **Quick Find** field, search for and access the **App Manager** settings page.
2. Create a new application by clicking **New Lightning App**. Follow the steps in New Lightning App creation procedure in Salesforce documentation.
3. When creating a new Lightning app, follow the additional steps below to configure the utility items for your Gplus Adapter.
   - In the **App Options** step, ensure that you select **Console Navigation**. Gplus Adapter does not support **Standard Navigation**.
   - In the **Utility Items** step, add `Open CTI Softphone` by clicking **Add Utility Item**.
   - For the soft phone, you can modify the display properties such as **Label, Panel Width** and **Panel Height**.
4. Specify **Navigation Items** and **User Profiles** for the new Lightning app by referring to Salesforce documentation.
5. Save the application.

Accessing Gplus Adapter

You can access the Gplus Adapter application by following the procedure below.

1. In the top-left corner, click the **App Launcher** icon  .
2. Select the app that you created when setting up the Gplus Adapter application.
3. Click the phone icon from the bar at the bottom-left corner to open the Gplus Adapter application.
4. Log in to the application to start using the Genesys contact center functionalities.

Migrating to Gplus Adapter (v9)

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Migrating to Gplus Adapter (v9).

Follow this procedure to migrate your existing Gplus Adapter 8.5 to Gplus Adapter 9.0 for Salesforce Lightning.

> **Important**
> - Gplus Adapter 9.0 supports only Salesforce Lightning. Hence, this migration procedure assumes that you're already using Salesforce Lightning.
> - You can run Gplus Adapter 8.5 and 9.0 in parallel during the migration testing phase.

Migration procedure

> **Important**
> Ensure that Agent Desktop release 9 has been enabled in your environment by confirming that the Agent Desktop 9 application icon is displayed in your Genesys Portal.

Genesys recommends that you complete steps 1 to 3 in your Salesforce Lightning test environment first.

1. Follow the steps in Integrating Gplus Adapter in Salesforce Lightning to download the **lightning-callcenter.xml** file, import into the Salesforce Call Centers configuration, and modify the imported Salesforce Call Center.
2. Gplus Adapter 9.0 provides two single sign-on (SSO) enabled options. If you are migrating Gplus Adapter 8.5 with SSO enabled, or are planning to start using SSO with Gplus Adapter 9.0, you can choose one of these two options:

- Gplus Adapter configured with SSO and a third-party Identity Provider (IDP).
- Gplus Adapter integrated with Salesforce SSO as the IDP. This option provides an improved experience over the first option.

3. Add users to the new Salesforce Call Center you just imported and modified. See Adding users to Call Center for more detailed steps. **Tip**: For testing purposes, you can either add a new test user or remove an existing user from the Salesforce Call Center with Gplus 8.5 and add them to the Salesforce Call Center with Gplus 9.0.
4. Test the Gplus Adapter 9.0 functionalities with the new Salesforce Call Center configuration.
5. If you're ready to migrate to the Salesforce Lightning production environment, follow the steps below:
   a. Repeat step 1 for the production Salesforce Call Center.
   b. Remove users from the Salesforce Call Center with Gplus 8.5 and add them to the Salesforce Call Center with Gplus 9.0.

Configuring SSO

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Configuring SSO.

You can configure Gplus Adapter to use either your own IDP or Salesforce as an IDP. You can choose one of these two options:

1. **Gplus Adapter configured with SSO and your own Identity Provider (IDP)**.
   - Ensure that SSO for Agent Desktop release 9 has been enabled in your environment by following the instructions in the Agent Setup SSO article and the Single Sign-On article.
   - To ensure that your IDP does not block access from the genesyscloud.com domain, use the **Content-Security-Policy** header setting (refer to CSP: frame-ancestors).
     For example: **Content-Security-Policy: frame-ancestors 'self'** https://*.lightning.force.com;

     > **Warning**
     > Do not set **X-Frame-Options** to **SAMEORIGIN**

   - For SSO logout configuration, use the following Agent Setup Desktop Global Login options:
     - **Invalidate Auth SSO session on Workspace logout**
     - **Show Change Account Link**
   - Adapter is now ready to use with SSO.

2. **Gplus Adapter integrated with Salesforce SSO as the IDP**.
   This option provides an improved experience over the first option.

Salesforce as SSO and IDP

You can simplify your agent log in process by integrating Gplus Adapter with Salesforce to use Salesforce as your single sign-on (SSO) identity provider (IDP). This means that your agent only has

to provide their Username and Tenant to log in to Gplus Adapter after they have logged into Salesforce using Salesforce as your SSO identity provider.

Follow these steps to set up SSO with Salesforce as the identity provider:

1. Enable Salesforce as an Identity Provider
2. Define Gplus Adapter as a Connected App in Salesforce

Enable Salesforce as an Identity Provider

Prerequisites

- You must have an Admin role in your organization's Salesforce account
- User email address (username) that you use to login to Salesforce. **Note:** Username email addresses must be the same in both Salesforce and Gplus Adapter.

Enable Salesforce as an Identity Provider

1. Follow the steps in the Enable Salesforce as an Identity Provider article in the **Salesforce Help**.
2. In the **Identity Provider** view, click **Download Metadata** to obtain a copy of the IDP metadata XML.

Enable SSO on Genesys tenants

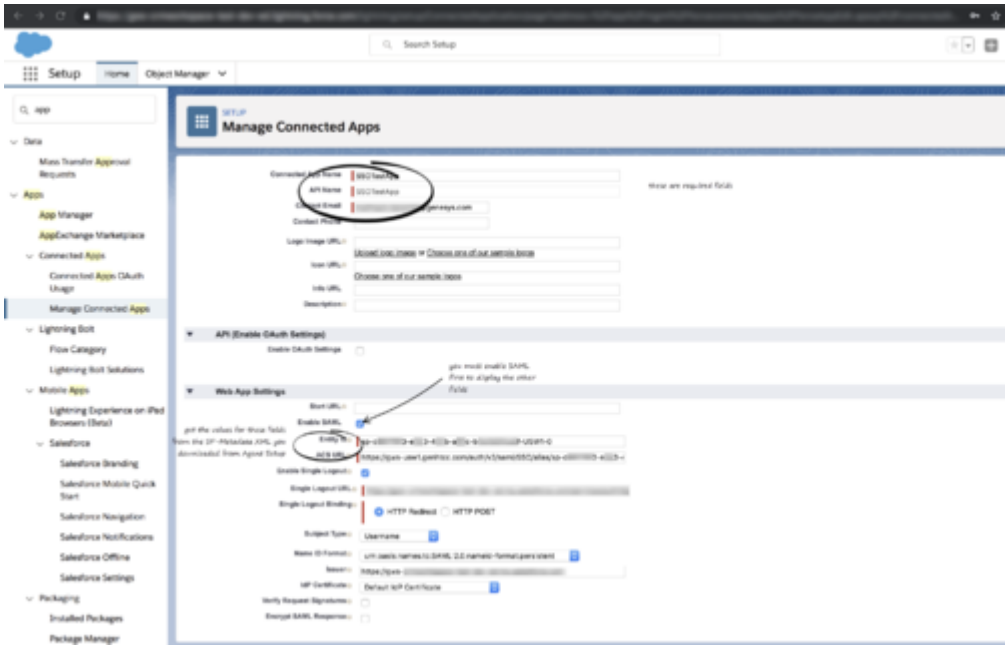
1. Employing SAML for SSO requires two parts, a Service Provider (Genesys Auth Service) and an Identity Provider (Salesforce). To complete your Service Provider configuration, you must upload the Salesforce Identity Provider IDP-metadata XML file you created in the previous procedure to Agent Setup (refer to the Agent Setup SSO article).
2. Download and open the SP-Metadata XML file in an XML viewer and find the **Location** parameter. You will need this parameter for the Entity ID and the ACS URL when you define Gplus Adapter as a Connected App in Salesforce. The **Location** URL looks something like this: `https://gws-usw1.genhtcc.com/auth/v3/saml/SSO/alias/<string representing the Entity ID>`.
3. For SSO logout configuration, use the following Agent Setup Desktop Global Login options:
   - **Invalidate Auth SSO session on Workspace logout**
   - **Show Change Account Link**

Define Gplus Adapter as a Connected App in Salesforce

1. To configure a new Connected App follow the instructions in the Salesforce Help Defining Service Providers as SAML-Enabled Connected Apps document.
2. Specify the required fields under **Basic Information**:
   - Connected App Name (for example, Gplus Adapter)
   - API Name (for example, GWS)

- Contact email (the email address of the Admin user)

3. Under Web App Settings, select **Enable SAML**
4. Use the **Location** string from the SP-Metadata xml file of the Enable Salesforce as an Identity Provider procedure to provide Entity ID and ACS URL:
   - Entity ID: The long string of numbers and letters at the end of the **Location** parameter URL after the last "/".
   - ACS URL: The URL from the **Location** parameter that you obtained from the SP-Metadata in step 2 of the *Enable SSO on Genesys tenants* procedure.



5. Use the identity provider event log to troubleshoot errors when trying to log in to Gplus Adapter.

Agent SSO login workflow

This is the general workflow when Gplus Adapter initiates the login process and uses Salesforce to identify the user:

1. An agent logs in to Salesforce.
2. The agent provides their Username and Tenant name in Gplus Adapter and clicks **Next**.
3. GPlus Adapter sends a request to Salesforce to authenticate the user.
4. Salesforce identifies the agent specified in the request and sends an authentication response.
5. Gplus Adapter authenticates the response sent by Salesforce.
6. If the agent is authenticated, they are logged in to Gplus Adapter while logged in to Salesforce.

Useful links

- Salesforce guide
- Video example from Salesforce

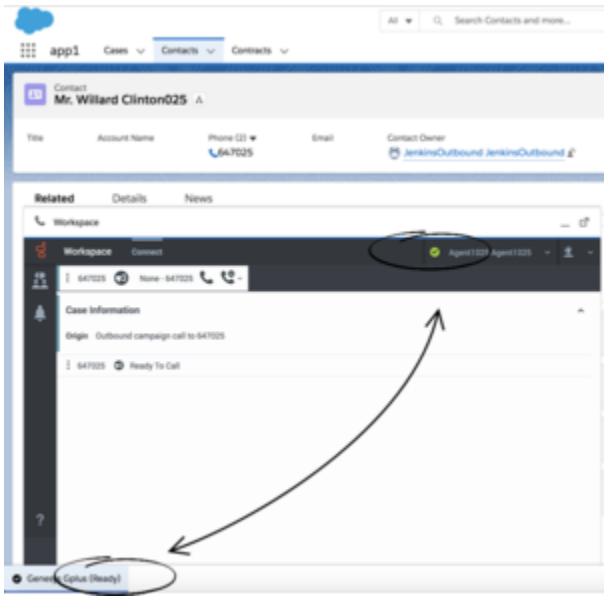
Configuring and Using General Features

> **Important**
>
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Configuring and Using General Features.

Gplus Adapter integrates Genesys interaction handling capabilities into Salesforce. The Adapter window is accessed through the Adapter button in the Salesforce Utility Bar. The Adapter window displays agent status at all times. The Adapter window can be docked, undocked, moved, resized, minimized, and maximized. Some Genesys interactions events cause the Adapter window to be maximized automatically.

Agent Status Display in Gplus Adapter

Gplus Adapter displays agent status in the Utility Bar whether the Adapter window is minimized or maximized. Agent status is obtained from Workspace Agent Desktop. This feature enables agents to monitor their status even when the Adapter window is minimized.



Dock, undock, move, and resize Adapter

By default, the Adapter window launches in the bottom-left corner in the Salesforce Utility Bar.

Agents can undock the Adapter window then move it to a different location. To undock (pop out) the window, they click the  icon in the top-right corner of the Adapter window and drag it where they want to place it. Once the window is undocked, agents can move or resize the window as needed.

Agents can dock (restore) the window back to its original position by clicking the  icon.

> **Tip**
> When Adapter is undocked and an agent clicks a Salesforce link to open another site within Salesforce and exits Lightning, an undocked Adapter window should close automatically. However, if it doesn't, as a workaround, the agent should manually close the Adapter window. For more information, refer to the following: Salesforce issue.

Interaction Events causing the Adapter window to Maximize

If an agent has minimized Adapter to handle other tasks in Salesforce, the Adapter window is automatically maximized by the following events:

- Receiving voice, chat, email, or workitem interactions
- Releasing interactions
- Marking Done interactions
- Receiving connections error notifications
- Receiving confirmation that an interaction has been added to the Genesys Universal Contact Server (UCS) database

Configuring Gplus to run Salesforce Apex

If you want to customize Salesforce by running Salesforce Apex on Genesys interaction events, you can configure Gplus Adapter to execute Salesforce runApex() initiated by the following Genesys interaction events: **DIALING** (outbound call), **RINGING** (inbound call), **ESTABLISHED**, and **RELEASED**.

Examples of how to use the Run Salesforce Apex on interaction events functionality include:

- Create Salesforce Case with user data from an interaction.
- Provide Salesforce Alert to Agent for reminders on how to handle a workflow in Salesforce.

Configure which events you want to trigger runApex code by using the **Run Salesforce Apex on Interaction Events** option in Agent Setup.

Configure the name of the custom Apex Class to execute by using the **Apex Class Name** option in Agent Setup.

Support for Genesys Interaction Recording

The Gplus Adapter console supports Genesys Interaction Recording (GIR). For an overview of Genesys Interaction Recording, see About Genesys Interaction Recording.

Genesys Interaction Recording is only supported for Voice channel and is not supported when Gplus Adapter is popped out of the Salesforce window.

To configure Genesys Interaction Recording for the Gplus Adapter, see Enable Screen Recording in the *Genesys Interaction Recording Solution Guide*. In this procedure, refer to the Integrating with Workspace Web Edition section.

To manage the call recording features of an agent, see Recording setup.

Support for Service Client API

Gplus Adapter supports the Service Client API, which you can use for custom features such as invoking click-to-email, starting or stopping a recording when a sensitive field is on the screen, and attaching data from Salesforce to Genesys key/value pairs.

Complete the following steps in Salesforce:

1. Deploy your Salesforce domain.
2. Create and deploy a message channel using Salesforce CLI or a third-party tool such as Workbench.
3. Download **gplus-service-client-api.js** from `https://[YOUR_HOST]/ui/crm-workspace/build/service-client-api/gplus-service-client-api.js`.

1. Add gplus-service-client-api.js to static resources.
2. Create VF page according to example.
3. Add VF page to your dashboard.
4. Contact your Genesys representative who can configure Gplus Adapter to use your message channel.

Click-to-Dial

> **Important**
>
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Click-to-Dial.

Click-to-dial enables agents to dial a phone number directly from Salesforce phone fields. Based on the pre-processing rules defined in Agent Setup, Adapter formats the selected phone number and then dials it. These configurations are often helpful while dialing an international number. Agents see the formatted phone number in the screen pop.

Pre-processing rules for international dialing

You can configure Adapter to apply custom pre-processing rules on click-to-dial requests to adjust the format of phone numbers or ANI for international dialing.

To support click-to-dial the pre-processing rules you setup for international numbers, Adapter examines the phone number it receives from Salesforce to determine if a prefix should be added before making the dial request. This is helpful when local numbers are stored in Salesforce records, but dialing prefixes are required to dial the call. You can configure pre-processing rules with the **Click to Dial Preprocessing Rules** setting in Agent Setup. For example, you can configure a pre-processing rule to add a "+1" prefix to the phone number being dialed.

You can also define pre-processing rules for the screen pop to replace a portion of the Automatic Number Identification (ANI) with a dialable number — for example, to remove a leading "+1". This is helpful in cases where local phone numbers are stored in Salesforce, but inbound customer calls might include international prefixes. You can configure these rules with the **Screen Pop Preprocessing Rule** setting in Agent Setup.

Screen Pop

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Screen pop.

Screen pop displays a relevant record or a search results list from Salesforce for an agent to choose when they are performing an inbound, outbound, or open media interaction.

You can setup how you want Adapter to select the screen pop record from Salesforce for different type of interactions.

This involves:

- Configuring screen pops in Salesforce and
- Configuring screen pops in Agent Setup

Configuring screen pops in Salesforce

To set up this functionality in Salesforce, login and go to **Setup > Customize > Call Center > SoftPhone Layouts** to create a SoftPhone Layout. Check out the Salesforce documentation for details about configuration.

When you set up a SoftPhone Layout for Adapter, the Gplus Adapter for Salesforce ignores the SoftPhone Layout settings that control call-related fields. Instead, Adapter gets this information from toast and case data you configure in the Genesys environment.

> **Important**
> Make sure you configure the **Screen Pop Settings** in the "CTI 2.0 or Higher Settings" section. These settings control whether the screen pop opens in a new window, tab, or Visualforce page.

Configuring screen pops in Agent Setup

In Agent Setup, you can configure Adapter screen pop settings for calls, chats, emails, and open media interactions and how Adapter displays screen pop when a call or chat is transferred or merged in a conference.

How screen pop records are retrieved from Salesforce

Before you proceed with selecting the screen pop settings, get familiar with how screen pop records are selected and displayed to an agent. This section helps you understand how Adapter determines the matching screen pop record from Salesforce.

Here's the default way Adapter tells Salesforce to display in a screen pop:

- If the UserData for the call has key(s) with the prefix "id_" (such as "id_SalesforceCaseId"), then the first value Adapter finds is sent in the screen pop request to Salesforce.
- If the UserData does not include any "id_" keys, Adapter builds a search string using a combination of the ANI or the DNIS, if available, and any UserData values with keys that have the "cti_" prefix (such as "cti_FirstName" or "cti_PhoneNumber"). You can specify whether the search should include the ANI by selecting the **Use ANI in Screen Pop Search** setting or DNIS (the number dialed by the caller) by selecting the **Use DNIS in Screen Pop Search** setting. These are dependent on the **Screen Pop Preprocessing Rule** in Agent Setup.
- If there are no "id_" keys, no ANI, no DNIS, and no "cti_" keys, then Adapter doesn't send a screen pop request.

Alternatively, you can configure the expressions Adapter uses to match screen pop UserData keys for the ID and search fields by selecting the **RegEx to Match UserData** setting and specifying **RegEx to Match UserData** option. Note that these options only change the criteria by which keys are matched from UserData. The existing logic and order between ID-based screen pop and search are unchanged (see the bullet points above). If you use these regular expression options to match your own ID key, then you should also ensure you set the **Object ID UserData Key** to **id_transfer_object** so that screen pop works with in-focus page transfers.

Calls

When an agent receives an external call, Adapter initiates a screen pop that causes Salesforce to show an appropriate record for the caller. By default, Adapter initiates the screen pop when the call is established, but you can select the **Screenpop On Ringing** setting to initiate a screen pop when the call is ringing instead. Note that Adapter also initiates screen pop for calls established through Genesys Softphone in Connector mode.

> ### Important
> You can also select screen pops for internal calls with the **Screen Pop For Internal Calls** setting and consult calls with the **Enable Screen Pop For Consult** setting.

For Outbound campaign calls, the screen pop appears immediately after the agent clicks **Get Record**

when the notification is displayed. An agent can then choose to make or decline the Outbound campaign call. For Push Preview campaigns, the screen pop appears after the agent clicks **Accept**. For more information on setting up outbound campaigns, case and toast data, see Outbound campaigns in this Administrator Guide.

Integrating Genesys Universal Contact Server (UCS)

You can enable agents to use Team Communicator to call a contact that is stored in the Universal Contact Server (UCS). You can view the detailed contact information from the **Contacts** tab. For more information, see Global Login Options in the Agent Setup Guide.

Chat

When an agent receives a chat invite from a contact, Adapter initiates a screen pop based on whether the **Screen Pop On Chat Invite** setting is selected.

- When selected, Adapter displays the screen pop on chat invite.
- When not selected or if the option is not available, Adapter displays the screen pop only when the agent accepts the invitation for the chat interaction.

Email

When an agent receives an email invite from a contact, Adapter initiates a screen pop based on whether the **Screen Pop On Email Invite** setting is selected.

- When selected, Adapter displays the screen pop on email invite.
- When not selected or if the option is not available, Adapter displays the screen pop only when the agent accepts the invitation for the email interaction.

Open Media items

When an agent receives an Open Media invite from a contact, Adapter initiates a screen pop based on whether the **Screen Pop On Open Media Invite** setting is selected.

- When selected, Adapter displays the screen pop on open media invite.
- When not selected or if the option is not available, Adapter displays the screen pop only when the agent accepts the invitation for an open media interaction.

Configuring screen pop by Salesforce object type

You can use the **Screen Pop object type** option in Agent Setup to specify what type of object you want Salesforce to screen-pop. **SOBJECT** is the default value, other available object types are:

- FLOW: Pops to the target flow. The flow is specified by the **Screen Pop Flow section name** option. Additional arguments can be passed to the flow, for example, the caller's phone number or a list of matching records. Specify the flow name and "flowArgs", valid flow arguments in JSON format, or the

`$attach_data_flow_name$` and `$attach_data_flowArgs_name$` parameters to take the flow name and arguments from the attached data.

- URL: Opens the URL in a new browser or tab. The URL is specified by the **Screen Pop URL section name** option. Specify the URL or use the `$attach_data_key$` parameter to take the URL from the interaction attached data.
- OBJECTHOME: Pops to the home of an object or entity, such as a Case or an Account. The object home is specified by the **Screen Pop Object Home section name** option. Specify the object or the `$attach_data_key$` parameter to take the object name from the interaction attached data.
- LIST: Pops to the specified list of Salesforce objects, such as Contacts or Accounts. The list is specified by the **Screen Pop List section name** option. Specify the user data keys or the `$attach_data_list_id$` and `$attach_data_list_scope$` parameters to take the List from the interaction attached data.
- SEARCH: Pops to the Top Results section of the search page. The search string is specified by the **Screen Pop Search section name** option. Specify the search string or the `$attach_data_search$` parameter to take the string from the interaction attached data.
- NEW_RECORD_MODAL: Pops a modal form for creation of a new object, such as Account, Case, Contact, or Lead. The new record modal is specified by the **Screen Pop New Record section name** option. Specify the object name or the `$attach_data_entityname$` parameter to take the new record type from the interaction attached data.

Configuring screen pops for conference or transfer interactions

When an agent performs a chat conference or transfer, Adapter can update the UserData with the ID of the object the agent is looking at in Salesforce. This ensures that the agent who receives the conference or transfer gets a screen pop for the most relevant object.

This can be helpful when:

- There are multiple matches for a screen pop search initially and the agent has to select between them.
- There are no matches for the initial screen pop search and the agent creates a new record.

You can enable this feature by selecting the **Object ID UserData Key** to **id_transfer_object**.

When Adapter updates the UserData, any existing UserData keys that begin with "id_" or "cti_" are removed and a new key, called "id_transfer_object", is added with the object ID of the focused page. If the focused page doesn't correspond to a Salesforce object, Adapter does not make an update and the existing UserData is sent with the conference or transfer request.

If you configured your environment to use a key other than "id_" or "cti_" (basically, if you need to set the **screenpop.id-key-regex** option as described in How screen pop records are retrieved from Salesforce, then you must also set the **screenpop.transfer-object-key** to your custom key for screen pop to work. Adapter adds the key you specify in this option instead of **id_transfer_object**.

Attached data

You can configure Adapter to save the ID, name, and type of the focused Salesforce object to

attached data when an agent clicks **Mark Done**. To do this, set **Salesforce Object Key**, **Salesforce Object Name**, and **Salesforce Object Type** to the keys you want Adapter to use when saving the ID, name, and type. Set these options to ensure the ID, name, or type is added to attached data when agents create a new object during a call. **Also, ensure the Workspace Web Edition option** interaction.case-data.is-read-only-on-idle **is set to `false`.** For more information about setting up Case Data, refer to this article.

Activity History

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Activity History.

Activity history entries are created by Gplus Adapter in Salesforce at the end of a call, chat, email, open media interaction, or an outbound campaign call. You can control which types of interactions are saved to a Salesforce activity history entry in Agent Setup by selecting the following settings:

- **Voice Activity Log** for call types
- **Chat Activity Log** for chat types
- **Open Media Activity Log** for open media (workitem) types
- **Activity Log on Screen Pop** to create activity on screen pop or mark done
- **Salesforce Activity Log Status** for the custom task status to be applied

Gplus Adapter also enables you to configure the subject of the activity history record with custom templates.

Task and activity creation in Salesforce when agent handles an interaction

Adapter creates activity logs in Salesforce for every interaction handled by an agent. When the agent marks an interaction as **Done**, Adapter searches for a corresponding contact in Salesforce and creates a task/activity for the interaction and associates it with the Salesforce object. Agents can update a Salesforce activity log **Comment** field only after marking **Done**. Any changes before marking **Done** are overwritten by Adapter. You must ensure that all the fields in the attached data exist in Salesforce. If one or more fields do not exist in Salesforce, an error will occur when Adapter tries to update the activity.

The rule that Adapter uses to select the correct record for activity log creation is the same as for screen pop record. The ANI phone value is used as the search main criterion. Additionally, if the call data contains a key name that is prefixed with **cti_** or other optionally defined prefix, it includes those key values in the search request using **OR** logic.

If there is more than one possible Salesforce object to which the interaction can be associated, or if there are no Salesforce objects to which the interaction can be associated, the task is created in the Task Pool. The agent can then associate the task to the correct object.

Agents sometimes handle calls or other interactions without having a Salesforce data object open, for example when they are on the **Homepage**. In this scenario, Adapter uses the same search record rule as described above, linking the selected record to a created task or not, depending on uniqueness of the search result.

> **Important**
>
> In Agent Desktop, if the **Mark Done** button does not disappear after an agent clicks it to close an interaction and activity recording fails, the likely cause is pre-processing options or an Adapter option is misconfigured with invalid regular expression value.

If you select the **Activity Log on Screen Pop** setting, Adapter creates the activity in Salesforce when the screen pop is displayed, and modifies logged activity when interaction is marked **Done**. Activity on screen pop is created for inbound and internal calls. Any changes in **Comments** field before marking **Done** are overwritten by Adapter.

Activity History fields

By default, Adapter updates certain Salesforce activity fields when it saves an activity history entry. Take a look at those fields and its corresponding Adapter field in the Salesforce activity fields table.

You can also create custom activity fields to store additional values as part of activity creation. See Customizing Activity Fields for more details.

Salesforce Activity Field	Genesys
Assigned To	Agent accepting Call
Created By	Agent accepting Call followed by timestamp.
Last Modified By	Agent accepting Call followed by timestamp.
Subject	Text (including the type of call, such as inbound, outbound to #Phonenumber). You can also define the subject by referring Configuring Subject fields.
Email Content	Text from the **Email Body** field for inbound and outbound replies. Stored as part of **Comments** block in Salesforce.
Due Date	Current Date
Related To	Record currently in context when an agent clicks **Mark Done**
Comments	Text with information from the Subject, Case Data (the full set that comes from Adapter), and Note (note data added by the agent) fields.
Status	Values are preset in the Salesforce configuration
Priority	Values are preset in the Salesforce configuration
Call Duration	Duration in seconds

Salesforce Activity Field	Genesys
Call Object Identifier	Interaction Id
Call Type	Type of call, such as inbound, outbound, or internal

Customizing activity fields

Gplus Adapter also allows you to create your own custom activity fields, and populate those fields with data specific to your organization. For example, you can create custom activity fields that can populate call, chat, or email related data in your Salesforce reports.

The first step is to create a custom activity field in Salesforce as described in the Salesforce documentation.

Custom Status

Prerequisite: Create custom values for the Status selection drop-down list in Salesforce. Refer to Add Custom Task Types and Customize Tasks in the Salesforce Help.

Next, you can define an activity log custom status to by applied by Gplus Adapter when an interaction is ending, such as **Completed**, **Answered**, and others that fit your business needs by specifying custom statuses as values of the **Salesforce Activity Log Status** option.

Configuring Subject field

You can configure what Adapter saves for the **Subject** field in the Salesforce activity by using custom templates with the **Templates Salesforce <interaction type> Subject** option, where, **<interaction type>** is either **Inbound-Voice**, **Outbound-Voice**, Transfer-Voice**,** Chat**, or** Email**.**

The value is a string that contains any text, along with templating variables wrapped in the reserved "{" and "}" characters.

You can use as many instances of this option as you need to handle multiple interactions types. For example, your configuration could include options for each of the supported interaction types:

- Templates Salesforce Inbound-Voice Subject
- Templates Salesforce Outbound-Voice Subject
- Templates Salesforce Transfer-Voice Subject
- Templates Salesforce Chat Subject
- Templates Salesforce Email Subject

Gplus Adapter supports **only** the following templating variables:

Variable	Description
interaction.ani	The number that originated the call. This variable identifies the caller for inbound calls and is best used in inbound templates.
interaction.callType	The type of call (inbound/outbound).
interaction.caseId	The unique ID of the related case.
interaction.contact	The first and last name of contact.
interaction.dnis	The last call dialed (useful for call transfer). This variable identifies the outbound location for outbound calls and is best used in outbound templates.
interaction.endDate	The date and time when interaction ended.
interaction.isConsultation	This is true if the interaction is a consultation.
interaction.startDate	The date and time when the interaction started.
userData.<key>	This can be any UserData key available for the interaction. You might also find some of the following UserData keys useful (they're included by default by Workspace Web Edition on transfers): • userData.GCS_TransferringAgentName — The name of the transferring agent. • userData.GCS_TransferringDate — The date and time of transfer. • userData.GCS_TransferringEmployeeId — The ID of the transferring employee. • userData.GCS_TransferringReason — The reason for the transfer. This is an empty string if no reason exists.
contact.EmailAddresses	A list of email addresses associated with the contact.
contact.PhoneNumbers	A list of phone numbers associated with the contact.

For example, you might want the activity for all inbound voice calls to have details about the customer's name and phone number. In this case, here's how you would configure the option:

Templates Salesforce Inbound-voice Subject = Customer: {interaction.contact}, Phone Number: {interaction.ani}

If a call comes in from Willard Clinton at 123-456-7890, when the call ends Adapter creates the related activity in Salesforce with following subject field: `Customer: Willard Clinton, Phone Number: 123-456-7890`

Another common use case might be to include information from UserData for all inbound calls. This could come from your own UserData keys or from UserData provided by Workspace Web Edition (see the table above). For example, you want the activity history's subject to include information from your own UserData key, PurposeOfCall, and the contact's name. Here's how you should configure the option:

Templates Salesforce Inbound-voice Subject = Call from {interaction.contact} about {userData.PurposeOfCall}

In this scenario, if Willard Clinton calls into the contact center and chooses the Technical Support option in the IVR, that information is saved in the PurposeOfCall UserData key. When the call ends, Adapter creates the related activity in Salesforce with the following subject field: `Call from Willard Clinton about Technical Support`

Storing email content in Activity History

The body of an email can be stored as a text version in the **Email Content** field in the **Comments** block of an Activity History in Salesforce. You can turn on or off this feature by using the **Email Include in Activity Description** option.

Open Media Activity History

Open Media items are documents (for example, faxes or PDFs) that might be directed to agents for handling while they are working on interactions of another type, such as email. For the Open Media interactions, the activity type is updated as **inbound** in the **Subject** and the **Call Type** fields of an Activity History. Other fields are updated as explained in the Salesforce activity fields table.

Use the **Open Media Activity Log** option in Agent Setup to specify whether activity histories are created in Salesforce for **Inbound** open media workitems.

Interaction History in Genesys Universal Contact Server (UCS)

Agents can view and manage their previous interactions with a Genesys UCS contact from the Agent Desktop My History tab.

Gplus Adapter for Microsoft Dynamics 365

> **Important**
>
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

The Gplus Adapter for Microsoft Dynamics 365 provides integrated management of communication channels within the Microsoft Dynamics 365 CRM. For Adminstrator documentation, see the Administrator's Guide. Also, see Gplus Adapter for Microsoft Dyanamics 365 documentation welcome page.

Gplus Adapter for ServiceNow

> **Important**
>
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

The Gplus Adapter for ServiceNow provides integrated management of communication channels within the ServiceNow CRM. For Administrator documentation, see the Administrator's Guide. Also, see Gplus Adapter for ServiceNow documentation welcome page.

Genesys Task Routing setup

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Set up Genesys Task Routing.

Genesys Task Routing allows you to submit workitems to Genesys from an external source, such as Salesforce or NetSuite. Once these external workitems are entered in the Genesys system, you can treat them like other Genesys interactions. For example, you can route workitems to the best-skilled agent using a Designer application. You can also get interactions that were processed by an agent in Agent Desktop, including automated third-party application screen pop-ups. Reporting is also available through Genesys CX Insights.

Genesys Task Routing is integrated with other Genesys solutions and it can support blended agents handling chat, email, voice and so on, at the same time they manage workitems.

At a high level, you'll need to complete the following steps to implement Genesys Task Routing:

- Use the Genesys Task Routing API to submit workitems to Genesys. You can also use this API to get, update and stop submitted workitems.
- Contact your Genesys representative to create a Capture Point.
- Define the routing application in Designer.
- Associate the Capture Point and routing application in Designer. **Note:** You can associate more than one Capture Point with the same routing application.

Complete the steps below to implement Genesys Task Routing for your contact center.

> **Important**
> You should be able to complete most of the steps on your own, but sometimes you'll need help from Genesys. We'll let you know when a task requires extra help from a Genesys representative.

1. Define agents and agent groups

Use Agent Setup to define agents and agent groups.

Next, work with your Genesys representative to enable an Open Media channel and define a capacity rule, then assign it in Agent Setup.

2. Configure Agent Desktop

You should go through the normal process to configure any necessary desktop options for the agents in your contact center. You'll also need help from your Genesys representative to set a few options related to Open Media.

3. Create a Capture Point

Contact your Genesys representative to create a Capture Point.

4. Create a routing application

Genesys Task Routing includes a sample Designer application to show one of the possible scenarios. Check out the sample application for more information.

For more details about creating a new routing application in Designer, see the documentation.

Once you have a routing application, use Designer to link it to the Capture Point created in step 3.

5. Integrate third-party applications

A third-party application can interact with Genesys Task Routing through the Genesys Task Routing API and the Service Client API.

You'll need to develop a third-party system adapter that can create and stop workitems using the Genesys Task Routing API. It should also update and retrieve information about a workitem, if needed.

The typical workflow is as follows:

1. The third-party application creates a workitem through the Genesys Task Routing API.
2. The third-party application updates and/or retrieves the interaction through the Genesys Task Routing API (if needed).
3. The third-party application stops the interaction.

There are two ways for the third-party application to stop an interaction:

- Implicit (recommended) — stop by invoking Mark Done using the Service Client API. For this method, complete the following steps:
  1. Set up Agent Desktop to use the API.
  2. Disable the Mark Done button in Agent Desktop (as mentioned earlier in Configuring Agent Desktop) and instead invoke Mark Done through the Service Client API.
  3. You can also use the Service Client API to set the Disposition Code.
- Explicit — stop by using the Genesys Task Routing API. This requires additional configuration from your Genesys representative.

A sample application that demonstrates using the Genesys Task Routing API and the Service Client API is also available through your Genesys representative.

Configure third-party application screen pop-up

You can configure Genesys Task Routing to enable a third-party application screen pop-up inside Agent Desktop when an agent accepts a task. There are two ways to do it:

1. Pop-up a third-party application screen for all media types. This approach works well when an agent is handling only workitems. See the Agent Setup documentation for details.
2. Pop-up a third-party application screen for only workitems. This approach works well when an agent handles multiple media types and the third-party pop-up should only happen for workitems. You'll need to work with your Genesys representative to configure this approach.

Sample Designer application for Genesys Task Routing

> **Important**
>
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Explore the sample Designer application.

> **Important**
>
> Contact your Genesys representative for access to the sample application.

Genesys Task Routing (GTR) includes a sample Designer application that you can use as a starting point to help meet your custom requirements for task routing. The application also includes a simplified Open Media Routing Data Table (called OPEN_MEDIA_ROUTING_CONFIG) to map the routing tasks based on agent, skill, agent group, and so on.

The sample application routes tasks to agents based on attributes you specify on the workitem interaction and information included in the Open Media Routing Data Table. It has four routing cascades that run in the following order:

1. GTR routes the task to the requested agent.
2. If not, GTR routes the task to an agent with the requested skill.
3. If not, GTR routes the task to the requested agent group.
4. If not, GTR routes the task to the default agent group.

The application works based on the assumption that all workitems are pre-classified in the third-party system before integrating with GTR. You can read more about this in Input values.

Using the Open Media Routing Table, you can configure specific values for agents, skills, and agent groups, or set a default mapping instead. If the interactions have more than one row, you can define a default row as routing precedence. You can also override the default routing table by specifying the requested agent, skill, or agent group directly in the interaction user data.

Prerequisites

Complete the following steps before you get started:

1. Create a default agent group named **workitems**. The sample application uses **workitems** as the default agent group, so if you choose another name, you must update the application to use the new name. The details about how to do this are explained in Import the application.
2. Configure the third-party pop-up as described in Integrate third-party applications.

Import the application

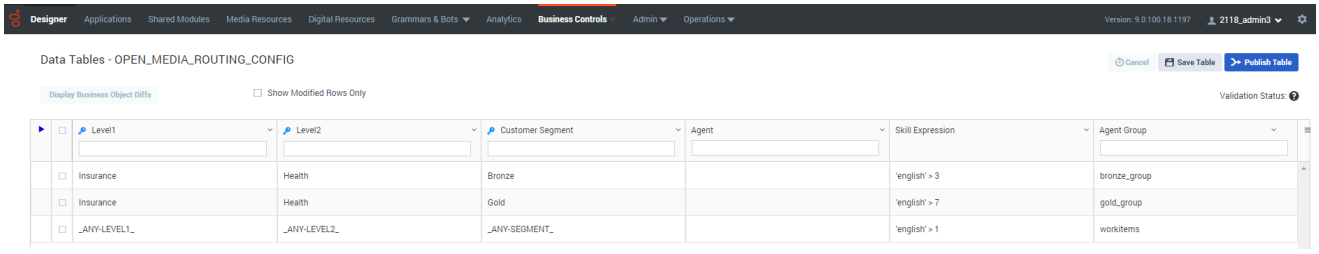
Contact your Genesys representative to get the sample Designer application and the Open Media Routing Data Table. Once you have the application, follow the steps in the Designer documentation to import the application.

Next, edit the sample application and click the **Initialize** header. Set the default value for the **overrideKeyName** variable — you can get this value from the Transactions Override Attached Data Key in Agent Setup.

If you are not using **workitems** as the default agent group, open the **Route to the default Agent Group** block and select another agent group instead.

Configure the routing table

Once the application is imported, you can find the OPEN_MEDIA_ROUTING_CONFIG table under **Business Controls > Data Tables**.



In the image above:

- **Level1**, **Level2**, and **Customer Segment** form a compound key.
- **Agent**, **Skill Expression**, and **Agent Group** are searchable values in the table.
- The row with *_ANY-LEVEL1_*, *_ANY-LEVEL2_*, and *_ANY-SEGMENT_* enables the default mapping. If you change the values in this compound key, the application won't be able to use the default mapping. If you don't need the default mapping, you can leave the **Agent**, **Skill Expression**, and **Agent Group** fields blank in this row.

Input values

To work with the sample application, make sure your third-party system adapter adds the following user data keys on the workitem interaction:

- GTR_level1 — The value for **Level1**. Use this field to add any kind of classification that makes sense for your business. For example, this could represent a particular category in your business structure or a severity/escalation level.
- GTR_level2 — The value for **Level2**. This field provides another level of classification.
- CustomerSegment — The value for **Customer Segment**. This is a third classification level that represents how important the customer is for your business. For example, you could have values such as Bronze, Silver, and Gold.

Together these three attributes form a compound key that is used to get the **Agent**, **Skill Expression**, and **Agent Group** values from the routing table. If you don't provide values for these attributes, the sample application uses the default mapping (see Configure the routing table above).

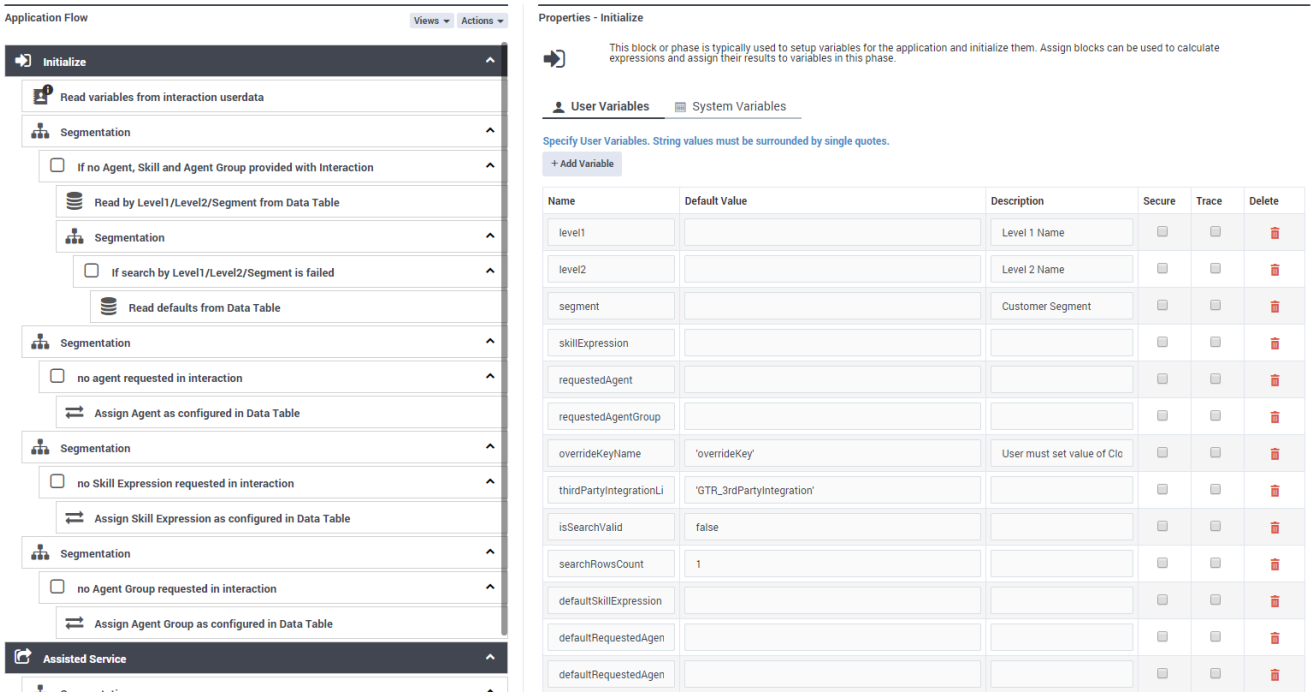
You can also override the default routing table values for **Agent**, **Skill Expression**, and **Agent Group** by specifying values for these fields with the following user data keys:

- GTR_requestedAgent
- GTR_skillExpression
- GTR_requestedAgentGroup

Application flow

The sample application goes through two phases when routing tasks: Initialize and Assisted Service.

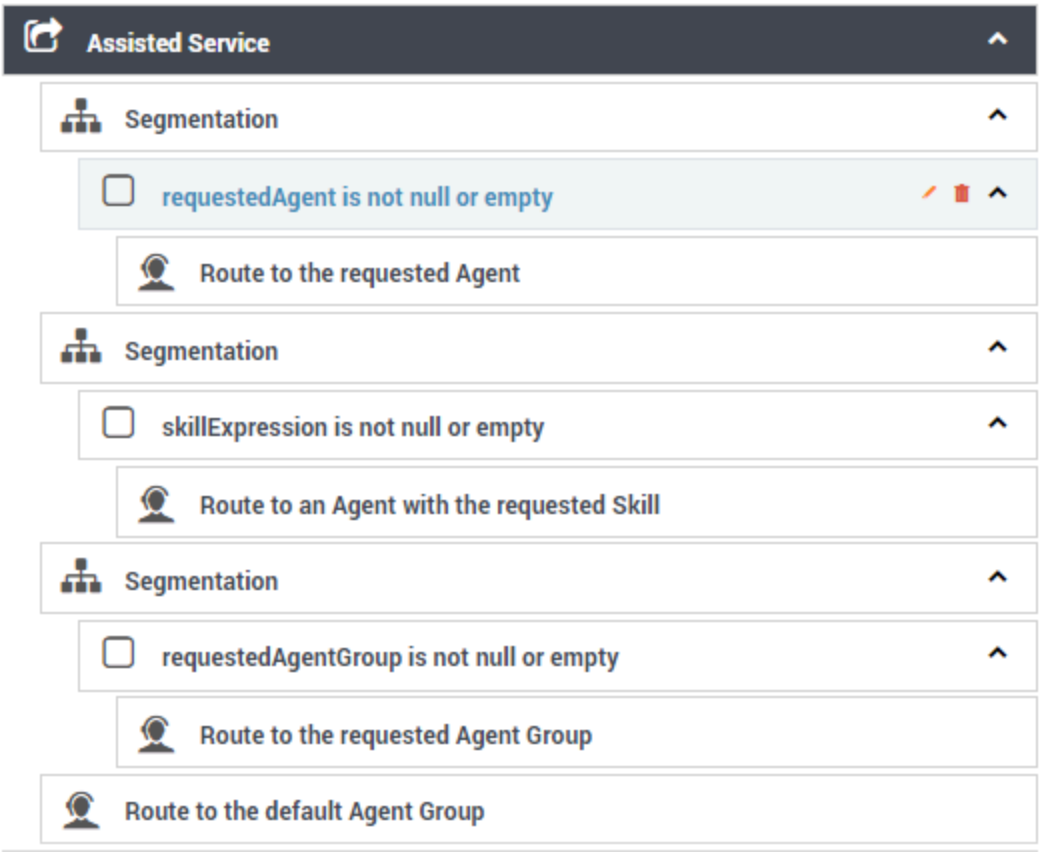
Initialize phase



In the Initialize phase, the sample application goes through the following setup:

1. The **Read variables from interaction userdata** block reads GTR_level1, GTR_level2, and CustomerSegment from the user data on the interaction. Next, the application writes a key/value pair to the interaction user data to support the media type for third-party application pop-up in Agent Desktop:
   - `overrideKeyName = thirdPartyIntegrationListObject` (the default is `overrideKey = GTR_3rdPartyIntegration`).
2. The **Segmentation** block checks to see **If no Agent, Skill or Agent Group provided with an interaction**.
   - If none exist, the application tries to read the missing parameter from the OPEN_MEDIA_ROUTING_CONFIG data table in the **Read by Level1/Level2/Customer Segment from Data Table** block.
   - If the key is not available or more than one row is present, the application sets the **isSearchValid** and **searchRowsCount** variables.
3. If the previous iteration is not successful, the application tries to **Read defaults from Data Table** by getting the values from the *_ANY-LEVEL1_*, *_ANY-LEVEL2_*, and *_ANY-SEGMENT_* default row.
4. After going through the parameters, the application runs three independent **Segmentation** blocks that check if there was an agent, skill, or agent group requested in the interaction user data (see Input values). If not, the application sets the values configured in the data table.

Assisted Service phase



In the Assisted Service phase, the application follows four routing cascades in this order:

1. Route to the requested Agent
2. Route to an agent with the requested Skill
3. Route to the requested Agent Group
4. Route to the default Agent Group

The first three routing cascades are a sequence of **Segmentation** blocks that perform null checks and route to the appropriate agent or agent group. The last cascade, **Route to the default Agent Group**, uses the **workitems** agent group you created in Prerequisites. As mentioned previously, you can also set another default group name here to suit your custom needs.

All **Route** blocks use a 120-second timeout by default. You can adjust this value and enhance the routing data table and application to specify timeouts for the **Level1**, **Level2**, and **Customer Segment** fields.

> **Warning**
> Genesys does not recommend values below 120 seconds.

Genesys Engage cloud APIs

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Developers.

The Genesys Engage cloud APIs are a collection of web APIs you can use to create your own custom applications that integrate with Genesys. For example, you can:

- Configure your Genesys environment, create users, and retrieve configuration data.
- Subscribe to statistics and get notifications when they change.
- Securely export your contact center data.
- Manage agent state.
- Perform basic call control, conferences and transfers, and call recording.
- Allow supervisors to monitor agent calls.

All documentation for the web APIs is hosted on the Genesys Engage cloud Developer Center. This is where you can find the reference information, articles, and tutorials you need to start working with the APIs.

The Genesys Engage cloud API suite also includes the Service Client API, which you can use to extend the UI integration when Agent Desktop is configured to embed other web client business applications. This JavaScript API is based on window.postMessage and provides methods your application can use to communicate cross domain with Agent Desktop while maintaining secured isolation. For example, you can use the API to get and set an agent's state from your application. See Service Client API for reference information and how to get started with the API.

Service Client API

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Service Client API Reference.

> **Important**
> You must contact your Genesys representative to configure Agent Desktop to use the Service Client API.

API overview

You can use the Service Client API to customize how your web application or website integrates with Agent Desktop. This JavaScript API is based on window.postMessage and provides methods your application can use to communicate cross domain with Agent Desktop while maintaining secured isolation.

You can use the Service Client API to perform the following actions:

- Controlling call recording from a third-party application
- Embedding multiple third-party applications in Agent Desktop
- Updating attached data from a third-party application
- Enabling click-to-dial from a third-party application
- Enabling Service Client API to invoke toast in Agent Desktop
- Controlling Case Selection from a Third Party Application

Controlling call recording from a third-party application

Review the following methods for details about call recording control:

- pauseCallRecording
- resumeCallRecording
- startCallRecording

- stopCallRecording

The call recording state is stored in the `recordingState` attribute on the interaction.Interaction object.

Embedding multiple third-party applications in Agent Desktop

You can now set the interaction.web-content option to a list of option section names that correspond to web extension views. This means that you can configure Agent Desktop to include more than one third-party web application, displayed as either a tab, a popup window, in the background at the interaction level, or hidden.

You should also make sure that the service-client-api.accepted-web-content-origins option references all the websites that should use the Service Client API.

Contact your Genesys representative to enable embedding multiple third-party applications in Agent Desktop.

Updating attached data from a third-party application

Review the following methods for details about updating attached data:

- deleteUserData
- getByInteractionId
- getInteractions
- setUserData

The user data is stored in the `userData` attribute on the interaction.Interaction object.

You should also be sure to configure the options related to user data in the Service Client section of Agent Setup to enable read and write access to user data.

Enabling click-to-dial from a third-party application

If you configure Agent Desktop to display your web application in a new tab in the Agent Desktop user interface, then the service API only gives access to the dial operation.

Enabling Service Client API to invoke toast in Agent Desktop

Review the following methods for details about enabling and updating toast:

- system.popupToast
- system.updateToast
- system.closeToast

Controlling case selection from a third-party application

Review the following method for details about case selecting control:

- selectCaseByCaseId

The case selection state is stored in the *isCaseSelected* attribute and the *isCaseExpanded* attribute on the **interaction.Interaction** object.

Getting started

Here's an overview of the steps you should to follow to access the API:

1. You have a web application that you've integrated in Agent Desktop—contact your Genesys representative to enable integration of web applications in Agent Desktop.
2. Download the sample application from GitHub.
3. Copy the **wwe-service-client-api.js** file in the sample application to a location your web application can access.
4. Set configuration options related to security—contact your Genesys representative to provision this security configuration.
5. Review Working with the API for more information about how to use the API.
6. Review the methods and types available in each namespace:
   - Agent Namespace
   - Configuration Namespace **Note:** You must work with your Genesys representative to enable and use this part of the Service Client API.
   - Email Namespace
   - Interaction Namespace
   - Media Namespace
   - System Namespace
   - Voice Namespace

Working with the API

After you've completed the setup and security steps, you're ready to start working with the Service Client API. The first thing you need to do is add a `<script>` tag to your web application that points to the **wwe-service-client-api.js** file (remember, you stored it somewhere accessible in Step 3 above).

Now you can access the API through the **genesys.wwe.service** namespace. For example:

```
<html>
    <head>
```

```
        <script src="wwe-service-client-api.js"></script>
        <script>

            function test() {
                genesys.wwe.service.sendMessage({
                    request: "agent.get"
                }, function(result) {
                    console.debug("SUCCEEDED, result: " + JSON.stringify(result, null, '\t'));
                }, function(result) {
                    console.debug("FAILED, result: " + JSON.stringify(result, null, '\t'));
                });
            }

            function eventHandler(message)
            {
                console.debug("Event: " + JSON.stringify(message, null, '\t'));
            }

            genesys.wwe.service.subscribe([ "agent", "interaction" ], eventHandler, this);

        </script>
    </head>
    <body>
        Hello world
    </body>
</html>
```

Here's an example of how you could modify attached data:

```
genesys.wwe.service.interaction.setUserData("1",
{
        MyKEY1: "MyValue1",
        MyKEY2: "MyValue2"
})
```

In the above example, the request is interaction.setUserData and the parameters are the `interactionId` of 1 and the keyValues of `MyKEY1` and `MyKEY2`.

All methods provided in the Service Client API are asynchronous, so to get the successful or failed result, just add the matching callback:

```
genesys.wwe.service.interaction.setUserData("1",
{
        MyKEY1: "MyValue1",
        MyKEY2: "MyValue2"
}, function(result){
    console.debug("SUCCEEDED, result: " + JSON.stringify(result, null, '\t'));
}, function(result){
    console.debug("FAILED, result: " + JSON.stringify(result, null, '\t'));
})
```

The global template for a service call is:

```
genesys.wwe.service.<Service name>.<Service function>(<... function parameters ...>,
[<optional done() callback>, [<optional fail() callback>]]);
```

The `done()` callback is called when a request is successfully sent without an error.

The `fail()` callback is called when a request generates an error or an exception.

The result of these functions is provided in a JSON object as a unique parameter.

Notifications

You can use the following code to subscribe to **agent** and **interaction** notifications:

```
function eventHandler(message)
{
	console.debug("Event: " + JSON.stringify(message, null, '\t'));
}

genesys.wwe.service.subscribe([ "agent", "interaction" ], eventHandler, context);
```

In the above example, `eventHandler` is the event handler function and `context` is an optional contextual object.

Here's an example with an agent `STATE_CHANGED` to Ready:

```
{
	"event": "agent",
	"data": {
		"eventType": "STATE_CHANGED",
		"mediaState": "READY"
	}
}
```

Here's an example with an agent `STATE_CHANGED` to Not Ready with a reason:

```
{
	"event": "agent",
	"data": {
		"eventType": "STATE_CHANGED",
		"mediaState": "NOT_READY_ACTION_CODE",
		"reason": "Break",
		"reasonCode": "1511"
	}
}
```

Finally, here's an example with an `ATTACHED_DATA_CHANGED` event on a voice interaction:

```
{
	"event": "interaction",
	"data": {
		"eventType": "ATTACHED_DATA_CHANGED",
		"media": "voice",
		"interaction": {
			"interactionId": "1",
			"caseId": "4dda1ab6-aeab-4a33-f5d0-0153c9fdb43b",
			"userData": {
				"IWAttachedDataInformation": {
					"DispositionCode.Label": "DispositionCode",
					"Option.interaction.case-data.header-foreground-
color": "#FFFFFF",
					"CaseDataBusinessAttribute": "CaseData",
					"DispositionCode.Key": "ChooseDisposition",
					"Option.interaction.case-data.frame-color": "#17849D"
				},
				"IW_CaseUid": "4dda1ab6-aeab-4a33-f5d0-0153c9fdb43b",
				"IW_BundleUid": "dfaca66c-4149-42a1-7244-337e949a12b5"
			},
			"parties": [
				{
```

```
                                "name": "5001"
                            }
                        ],
                        "callUuid": "4L6JGNEE9H7DT671FRPTKE6CQ000000G",
                        "state": "DIALING",
                        "previousState": "UNKNOWN",
                        "isConsultation": false,
                        "direction": "OUT",
                        "callType": "Internal",
                        "dnis": "5001",
                        "isMainCaseInteraction": true
                    }
                }
            }
```


Event Type references

The system eventType field can be one of the following:

eventType	Description
CUSTOM_TOAST_BUTTON_CLICK	Uses the following parameters: • **customToastId**: The identifier of the toast where the button has been clicked. The identifier is returned by the popupToast method. • **buttonIndex**: The index of the clicked button. The index starts by 0.

The interaction eventType field can be one of the following:

eventType	Description
Common events to all interaction types	
UNKNOWN	An unknown event occurs.
ADDED	The interaction has been added in the list of interactions.
REMOVED	The interaction has been removed from the list of interactions.
ATTACHED_DATA_CHANGED	The attached data have changed in the interaction.
CASE_OR_BUNDLE_ID_CHANGED	The case or the bundle identifier of this interaction has changed.
NEW_MESSAGE	This event represents a new message.
ERROR	An error occurs in the interaction.
Voice events	
CALL_RECORDING_STATE_CHANGED	The call recording state changed.
DIALING	The outbound call starts ringing.

eventType	Description
ESTABLISHED	The call has been established.
HELD	The call has been held.
PARTY_CHANGED	The list of party has been changed in the interaction.
RELEASED	The call has been released.
RINGING	The inbound call starts ringing.
OpenMedia events	
ACCEPTED	The open media interaction is accepted.
COMPLETED	The open media interaction has been completed (Mark as done).
COMPOSING	The open media interaction is in composing mode.
CREATED	The open media interaction has been created.
INSERT_STANDARD_RESPONSE	A standard response has been inserted in the interaction.
INVITED	The open media interaction is an invitation.
INVITED_CONFERENCE	The open media interaction receive a conference invitation.
IN_QUEUE_FAILED	The place in queue has failed.
IN_WORKBIN	The interaction has been placed in the work-bin.
IN_WORKBIN_FAILED	The place in work-bin has failed.
LEFT_CONFERENCE	The open media interaction has left the conference.
PULLED	The open media interaction has been pulled from a work-bin.
PULL_FAILED	The pull from the queue has failed.
PULL_WORKBIN_FAILED	The pull from the work-bin has failed.
REVOKED	The open media interaction has been revoked.
TRANSFER_COMPLETED	The open media interaction has been transferred and the transfer has been completed.
Chat events (inherit from OpenMedia events)	
ENDED	The chat has been ended.
JOIN_FAILED	The connection with the chat server failed.
JOIN_PENDING	The interaction is trying to join the chat session.
Outbound email events (inherit from OpenMedia events)	
CANCELLED	The outbound email has been cancelled.
SENT	The outbound email has been sent.

Agent namespace

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Agent Namespace.

Methods

The Agent namespace includes the following methods:

- get
- getState
- getStateList
- setState

get

Signature	<static> get() → {agent.Agent}
Description	Gets the agent's attributes.
Returns	agent.Agent

getState

Signature	<static> getState() → {media.State}
Description	Gets the agent's state.
Returns	media.State

getStateList

Signature	<static> getStateList() → {Array.<media.State>}
Description	Gets the list of possible agent states.
Returns	Array.<media.State>

setState

Signature	<static> setState(*stateOperationName*)
Description	Sets the agent's state.
Parameters	**Name** / **Type** / **Description** stateOperationName / string / An `operationName` from the agent states list. See State.

Type Definitions

The agent namespace includes the following object types:

- Agent

Agent

Description	Represents the JSON structure of the agent.
Type	Object
Properties	**Name** / **Type** / **Description** employeeId / string / The agent's unique identifier used for routing purposes. firstname / string / The agent's first name. lastname / string / The agent's last name. username / string / The agent's username. This is a global unique ID.

Configuration namespace

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Configuration namespace.

> **Important**
> You must work with your Genesys representative to enable and use this part of the Service Client API.

Methods

The Configuration namespace includes the following methods:

- getOption

getOption

Signature	<static> getOption(*options*) → {Array.<Section>}
Description	Get configuration options and values for a specific option name or a subset of options from the **[interaction-workspace]** section or a custom section.
Parameters	**Name** / **Type** / **Argument** / **Description** options / string / / An array of configuration options or sections to return. Unless otherwise specified, the API will return options

Signature	<static> getOption(*options*) → {Array.<Section>}

Name	Type	Argument	Description
			from the **[interaction-workspace]** section by default. You can specify any of the following: • A single option: `genesys.wwe.service.configuration.getOption('voice.auto-answer', succeeded, failed)` • A single option in a specific section: `genesys.wwe.service.configuration.getOption('CustomSection/option.cus`

Signature	<static> getOption(*options*) → {Array.<Section>}

Name	Type	Argument	Description
			tom.customer.code', succeeded, failed) • Multiple options: genesys.wwe.service.configuration.getOption(['voice.auto-answer', 'privilege.email.can-mark-done'], succeeded, failed) • Multiple options in different sections: genesys.wwe.service.configuration.getOption([

Signature	<static> getOption(*options*) → {Array.<Section>}

Name	Type	Argument	Description
			'privilege.*', 'CustomSection/option.custom.customer.code'], succeeded, failed) You can use an asterisk '*' as a wildcard, but only at the end of each word. For example: • voice.* • voice.auto* • sipendpoint.* • CustomAPI/test.* You cannot use an asterisk at the start of an option or section. For example, the following values are not allowed: • *.mark-done • *.auto

Returns	Array.<Section>

Type definitions

The Configuration namespace includes the following object types:

- Section

Section

Description	Represents the JSON structure of a configuration section. Each section includes a list of key/value pairs for the matching option(s).
Type	Object
Properties	**Name** / **Type** / **Description** name / string / The name of the configuration option. value / string or array of strings / The value of the configuration option.

Email namespace

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Email namespace.

Methods

The Email namespace includes the following methods:

- create

create

Signature	<static> create(*destination*, *userData*)			
Description	Creates a new empty email.			
Parameters	**Name**	**Type**	**Argument**	**Description**
	destination	string		The destination address for the email.
	userData	object	<optional>	The attached user data key/value object that is updated with each interaction event.

Interaction namespace

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Interaction namespace.

Methods

The Interaction namespace includes the following methods:

- deleteUserData
- getByInteractionId
- getInteractions
- selectCaseByCaseId
- setUserData
- markdone
- blockMarkdone
- unblockMarkdone
- accept
- reject

deleteUserData

Signature	<static> deleteUserData(*interactionId*, *key*)
Description	Deletes the user data attached to the interaction. The List of User Data Write Allowed option in Agent Setup might restrict the allowed key/value pairs.
Parameters	**Name** / **Type** / **Description** interactionId / string / The unique identifier for the interaction. key / string / The key to

Signature	<static> deleteUserData(*interactionId*, *key*)

Name	Type	Description
		delete from the attached data.

getByInteractionId

Signature	<static> getByInteractionId(*interactionId*) → {interaction.Interaction}
Description	Gets an interaction by its unique identifier.
Parameters	

Name	Type	Description
interactionId	string	The unique identifier for the interaction.

Returns	interaction.Interaction or null if the interaction doesn't exist.

getInteractions

Signature	<static> getInteractions() → {Array.<interaction.Interaction>}
Description	Gets all the interactions.
Returns	Array.<interaction.Interaction>

selectCaseByCaseId

Signature	<static> genesys.wwe.service.interaction.selectCaseByCaseId(caseId, succeeded, failed)
Description	Select the case in the UI by case identifier. If you subscribe to the "interaction" events (genesys.wwe.service.subscribe(["interaction"], eventHandler, this);), you will receive the following event:

```
Received interaction event: {
        "event": "interaction",
        "data": {
                "eventType":
"CASE_COLLAPSED",
                "selectedCaseId": "4401820b-
c4e6-4994-69c2-6ae7fdbc4905"
        },
```

Signature	<static> genesys.wwe.service.interaction.selectCaseByCaseId(caseId, succeeded, failed)

	<pre>"userAgent": "WWE Server",
	"protocolVersion": 2
	}
	Received interaction event: {
	"event": "interaction",
	"data": {
	"eventType": "CASE_EXPANDED",
	"selectedCaseId": "4401820b-c4e6-4994-69c2-6ae7fdbc4905"
	},
	"userAgent": "WWE Server",
	"protocolVersion": 2
	}
	Received interaction event: {
	"event": "interaction",
	"data": {
	"eventType": "CASE_SELECTED",
	"selectedCaseId": "d4187b87-9fe1-4db8-0515-6a91e666e22d"
	},
	"userAgent": "WWE Server",
	"protocolVersion": 2
	}</pre>

Parameters	**Name** / **Type** / **Description** caseId / string / The unique identifier for the case.

setUserData

Signature	<static> setUserData(*interactionId*, *keyValues*)
Description	Sets the user data on the live interaction (for voice, this means the interaction is not in the IDLE state). This request overwrites any existing keys on the user data. The List of User Data Write Allowed option in Agent Setup might restrict the allowed key/value pairs.
Parameters	**Name** / **Type** / **Description** interactionId / string / The unique identifier for the interaction. keyValues / object / The key value pairs to set on the user data.

markdone

Signature	<static> markdone(*interactionId*)
Description	Mark done the selected interaction.
Parameters	**Name** / **Type** / **Description** interactionId / string / The unique identifier for the interaction.

blockMarkdone

Signature	<static> blockMarkdone(*interactionId*, *warningMessage*)
Description	Block the mark done operation on the selected interaction. The "markdone" event must be subscribed to receive the event which informs that there is a delay in blocking the markdone operation with this method.
Parameters	**Name** / **Type** / **Description** interactionId / string / The unique interaction identifier of the interaction to prevent the mark done operation. warningMessage / string / The warning message.

unblockMarkdone

Signature	<static> unblockMarkdone(*interactionId*)
Description	Unblock the mark done operation on the selected interaction that was previously blocked.
Parameters	**Name** / **Type** / **Description** interactionId / string / The unique interaction identifier of the interaction to prevent the mark done

Signature	<static> unblockMarkdone(*interactionId*)
	Name / **Type** / **Description**: operation.

accept

Signature	<static> accept(*interactionId*, *succeeded*, *failed*)
Description	Accept an interaction when it is ringing in Agent Desktop.
Parameters	**Name** / **Type** / **Description**: interactionId / string / The unique interaction identifier of the interaction to be accepted. succeeded / string / The callback function to use if the operation succeeded. failed / string / The callback function to use if the operation failed.

reject

Signature	<static> reject(*interactionId*, *succeeded*, *failed*)
Description	Reject an interaction when it is Ringing in Agent Desktop.
Parameters	**Name** / **Type** / **Description**: interactionId / string / The unique interaction identifier of the interaction to be rejected. succeeded / string / The callback function to use if the operation

Signature	<static> reject(*interactionId*, *succeeded*, *failed*)

Name	Type	Description
		succeeded.
failed	string	The callback function to use if the operation failed.

Type definitions

The Interaction namespace includes the following object types:

- Interaction
- Party

Interaction

Description	Represents the JSON structure of an interaction. Attributes specific to voice interactions are: `callUuid`, `direction`, `callType`, `ani`, `dnis` and `recordingState`.
Type	Object

Properties

Name	Type	Description
interactionId	string	The unique identifier for the interaction. **Note:** This is a client-side ID that is lost on the next session or refresh.
parentInteractionId	string	The unique identifier for the parent interaction. **Note:** This is a client-side ID that is lost on the next session or refresh.
caseId	string	This identifier

Description	Represents the JSON structure of an interaction. Attributes specific to voice interactions are: `callUuid`, `direction`, `callType`, `ani`, `dnis` and `recordingState`.

Name	Type	Description
		targets the case that this interaction is part of.
userData	object	The attached user data key/value object that is updated with each interaction event.
state	string	The current state of the interaction. Possible values are: • UNKNOWN — An unknown state. • IDLE — Specifies a non-active interaction which could be closed. • RINGING — The inbound call is ringing. • DIALING — The outbound call is ringing. • TALKING — The call is established. • HELD — The call is on hold.

Description	Represents the JSON structure of an interaction. Attributes specific to voice interactions are: `callUuid`, `direction`, `callType`, `ani`, `dnis` and `recordingState`.

Name	Type	Description
		• PREVIEW — The interaction is a call preview. • INVITED — The open media interaction is inviting. • ACCEPTED — The open media interaction is accepted. • CREATED — The open media interaction has been created. • PULLED — The open media interaction has been pulled from a workbin. • REVOKED — The open media interaction has been revoked. • COMPLETED — The open media interaction

Description	Represents the JSON structure of an interaction. Attributes specific to voice interactions are: `callUuid`, `direction`, `callType`, `ani`, `dnis` and `recordingState`.

Name	Type	Description
		has been completed (Mark as done). • ERROR — The open media interaction has an error. • SAVED — The open media interaction has been saved. • TRANSFERRING — The open media interaction is being transferred. • TRANSFER_COMPLETED — The open media interaction has been transferred and the transfer has been completed. • INVITED_CONFERENCE — The open media interaction receives a conference invitation. • LEFT_CONFERENCE — The

Description	Represents the JSON structure of an interaction. Attributes specific to voice interactions are: `callUuid`, `direction`, `callType`, `ani`, `dnis` and `recordingState`.

Name	Type	Description
		open media interaction has left the conference. • USER_DATA_ATTACHED — Data has been attached to the interaction. • USER_DATA_UPDATED — The attached data has changed in the interaction. • JOIN_PENDING — Trying to join the chat session. • JOIN_FAILED — The connection with the chat server failed. • HISTORY_IN_PROGRESS — Loading the content of the chat interaction. • HISTORY_DONE — The content of the chat interaction has been loaded.

Description	Represents the JSON structure of an interaction. Attributes specific to voice interactions are: `callUuid`, `direction`, `callType`, `ani`, `dnis` and `recordingState`.

Name	Type	Description
		• CANCELLED — The outbound email is cancelled. • SENT — The outbound email is sent. • READY — The call preview is ready. • CANCELED — The call preview is cancelled. • REJECTED — The call preview is rejected.
previousState	string	The previous state of the interaction.
parties	Array.<interactionParty>	A collection of all the parties involved in the interaction.
isConsultation	boolean	This property is true if the interaction is a consultation; otherwise, it's false.
isMainCaseInteraction	boolean	This property is true if the interaction is the main interaction in the customer

Description	Represents the JSON structure of an interaction. Attributes specific to voice interactions are: `callUuid`, `direction`, `callType`, `ani`, `dnis` and `recordingState`.

Name	Type	Description
		case; otherwise, it's false. In Workspace Web Edition, the main interaction is related to Case Information, Disposition, Note, Contact Profile, and so on.
callUuid	string	The UUID of the call. This attribute is only on voice interactions.
direction	string	The call direction. Possible values are: IN, OUT or UNKNOWN. This attribute is only on voice interactions.
callType	string	The call type. Possible values are: INTERNAL, INBOUND, OUTBOUND, CONSULT or UNKNOWN. This attribute is only on voice interactions.
ani	string	The Automatic Number Identification service. This attribute is only on voice interactions.

Description	Represents the JSON structure of an interaction. Attributes specific to voice interactions are: `callUuid`, `direction`, `callType`, `ani`, `dnis` and `recordingState`.

Name	Type	Description
dnis	string	The Dialed Number Identification Service. This attribute is only on voice interactions.
recordingState	string	The call recording state. Possible values are: STOPPED, RECORDING or PAUSED. This attribute is only on voice interactions.
isCaseSelected	boolean	Is true if the case containing this interaction is selected, otherwise is false.
isCaseExpanded	boolean	Is true if the case containing this interaction is expanded, otherwise is false.
interactionUUID	string	The `attr_itx_id` for a multimedia interaction or the `callUuid` for a voice interaction.

Party

Description	Represents the JSON structure of a party.
Type	Object

Description	Represents the JSON structure of a party.
Properties	**Name** / **Type** / **Description** name / string / The name of the party.

Media namespace

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Media namespace.

Methods

The Media namespace includes the following methods:

- getMediaList
- setState

getMediaList

Signature	<static> getMediaList() → {Array.<media.Media>}
Description	Get the media attributes.
Returns	Array.<media.Media>

setState

Signature	<static> setState(*name*, *stateOperationName*)
Description	Sets the media state.
Parameters	**Name** / **Type** / **Description** name / string / The media name. stateOperationName / string / An `operationName` from the agent states list. See State.

Type definitions

The Media namespace includes the following object types:

- Media
- State
- Device

Media

Description	Represents the JSON structure of a media.		
Type	Object		
Properties	**Name**	**Type**	**Description**
	name	string	The media name.
	state	media.State	The media state object.

State

Description	Represents the JSON structure of a media state.		
Type	Object		
Properties	**Name**	**Type**	**Description**
	type	string	The type of operation. Possible values are: • LOGOUT • READY • PARTIAL_READY * • NOT_READY • NOT_READY_ACTION_CODE • NOT_READY_AFTER_CALLW • NOT_READY_AFTER_CALLW • DND_ON • OUT_OF_SERVICE *

Description	Represents the JSON structure of a media state.

Name	Type	Description
		• LOGOUT_DND_ON * • UNKNOWN *
displayName	string	The display name of the state.
operationName	string	The operation name to use with agent.setState and media.setState.

* *States that are limited to an event and can't be applied by code*

Device

Description	Represents the JSON structure of a media.
Type	Object

Properties

Name	Type	Description
number	string	The phone number configured for an agent – the physical DN. **Note**: This property is applicable only for voice data.
dynamicPhoneNumber	string	The dynamic phone number configured for the agent for the session. **Note**: This property is applicable only for voice data. This property is applicable only when there is an alternate phone number and

Description	Represents the JSON structure of a media.
	Name / **Type** / **Description** applicable for the current session only.

System namespace

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to System namespace.

Methods

The System namespace includes the following methods:

- getAllowedServices
- triggerActivity
- closeToast
- popupToast
- updateToast

getAllowedServices

Signature	<static> getAllowedServices() → {Array.<string>}
Description	Gets the list of allowed services, as determined by the security configuration for Agent Desktop. If the domain of the web application that calls this method isn't listed in the service-client-api.accepted-web-content-origins option, then this method fails. Contact your Genesys representative for details about this configuration.
Returns	Array.<string>

triggerActivity

Signature	<static> triggerActivity()
Description	Triggers a fake activity to prevent the inactivity timer from closing the agent session.

popupToast

<table>
<tr><th>Signature</th><td colspan="3"><static> popupToast(parameters) → {string}</td></tr>
<tr><th>Description</th><td colspan="3">Pops up a new custom toast.</td></tr>
<tr><th>Parameters</th><td colspan="3">
<table>
<tr><th>Name</th><th>Type</th><th>Description</th></tr>
<tr><td>title</td><td>string</td><td>The title</td></tr>
<tr><td>iconUrl</td><td>string</td><td>The URL of the icon you want to display in the title bar of the custom toast popup.</td></tr>
<tr><td>subject</td><td>string</td><td>Optional. The subject</td></tr>
<tr><td>message</td><td>string</td><td>Optional. The message</td></tr>
<tr><td>keyValues</td><td>string</td><td>Optional. JSON object used to fill the key value pair list. For example: <code>{"key1" ; "value one","key2" ; "value two","key3" ; "value three"}</code>.</td></tr>
<tr><td>buttons</td><td>Array.<string></td><td>Optional. Each character string in this array becomes a button.</td></tr>
<tr><td>buttonShowDismiss</td><td>boolean</td><td>Optional. If set to <code>true</code>, displays the Show and Dismiss buttons and pops up the current iframe if the Show button is pushed. If set to <code>false</code> , displays '"OK"' or custom buttons based on the parameter's</td></tr>
</table>
</td></tr>
</table>

Signature	<static> popupToast(parameters) → {string}
	<table><tr><th>Name</th><th>Type</th><th>Description</th></tr><tr><td></td><td></td><td>buttons.</td></tr><tr><td>autoCloseTimeout</td><td>object</td><td>Optional. If set to greater than <code>0</code>, the popup is automatically closed after the specified milliseconds.</td></tr><tr><td>sendToMyMessage</td><td>object</td><td>Optional. If set to <code>true</code>, sends the subject, iconUrl, title, keyValues, and message parameters to the MyMessage panel.</td></tr></table>
Returns	A unique identifier

updateToast

Signature	<static> updateToast(id, parameters) → {boolean}
Description	Updates the specified toast.
Parameters	<table><tr><th>Name</th><th>Type</th><th>Description</th></tr><tr><td>id</td><td>string</td><td>The identifier of the toast to update. The identifier is returned by the popupToast method.</td></tr><tr><td>parameters</td><td>object</td><td><table><tr><th>Name</th><th>Type</th><th>Description</th></tr><tr><td>title</td><td>string</td><td>The title</td></tr><tr><td>iconUrl</td><td>URL string</td><td>The URL of the icon you want</td></tr></table></td></tr></table>

Signature	<static> updateToast(id, parameters) → {boolean}

Name	Type	Description

Name	Type	Description
		to display in the title bar of the custom toast popup.
subject	string	Optional. The subject.
message	string	Optional. The subject.
keyValues	object	Optional. JSON object used to fill the key value pair list. For example: {"key1" : "value one","key2" : "value two","key3" : "value three"}.
buttons	Array.<string>	Each character string in this array becomes a button.
buttonSlideToDismiss	boolean	

<table>
<tr><th>Signature</th><td><static> updateToast(id, parameters) → {boolean}</td></tr>
<tr><td></td><td>
<table>
<tr><th>Name</th><th>Type</th><th>Description</th></tr>
<tr><td></td><td></td><td>
<table>
<tr><th>Name</th><th>Type</th><th>Description</th></tr>
<tr><td></td><td></td><td>set to <code>true</code>, displays Show and Dismiss buttons and pops up the current iframe if the Show button is pushed. If set to <code>false</code>, displays '"OK"' or custom buttons based on the parameter's buttons.</td></tr>
</table>
</td></tr>
</table>
</td></tr>
<tr><th>Returns</th><td><code>true</code> if the toast has been updated; <code>false</code> if the toast identifier has not been found.</td></tr>
</table>

closeToast

<table>
<tr><th>Signature</th><td><static> closeToast(id) → {boolean}</td></tr>
<tr><th>Description</th><td>Closes the specified toast.</td></tr>
<tr><th>Parameters</th><td>
<table>
<tr><th>Name</th><th>Type</th><th>Description</th></tr>
<tr><td>id</td><td>string</td><td>The identifier of the toast to close. The</td></tr>
</table>
</td></tr>
</table>

<table>
<tr><th>Signature</th><td><static> closeToast(id) → {boolean}</td></tr>
<tr><td></td><td>
<table>
<tr><th>Name</th><th>Type</th><th>Description</th></tr>
<tr><td></td><td></td><td>identifier is returned by the popupToast method.</td></tr>
</table>
</td></tr>
<tr><th>Returns</th><td><code>true</code> if the toast has been updated; <code>false</code> if the toast identifier has not been found.</td></tr>
</table>

Voice namespace

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Voice namespace.

Methods

The Voice namespace includes the following methods:

- answer
- dial
- hangUp
- hold
- resume
- pauseCallRecording
- resumeCallRecording
- startCallRecording
- stopCallRecording
- isMicrophoneMute
- muteMicrophone
- unmuteMicrophone
- isSpeakerMute
- muteSpeaker
- unmuteSpeaker

answer

Signature	answer('interactionId')
Description	Answers the incoming call.

Signature	answer('interactionId')			
Parameters	Name	Type	Argument	Description
	interaction	string		The interaction identifier

dial

Name	Type	Argument	Description
destination	string		The call destination number.
userData	object		The attached user data key/value object that is updated with each interaction event.

hangUp

Signature	hangUp('interactionId')			
Description	Releases the incoming call.			
Parameters	Name	Type	Argument	Description
	interaction	string		The interaction identifier

hold

Signature	hold('interactionId')			
Description	Holds the incoming call.			
Parameters	Name	Type	Argument	Description
	interaction	string		The interaction identifier

resume

Signature	resume('interactionId')			
Description	Resumes the held call.			
Parameters	**Name**	**Type**	**Argument**	**Description**
	interaction	string		The interaction identifier

pauseCallRecording

Name	Type	Description
interactionId	string	The unique identifier for the interaction.

resumeCallRecording

Name	Type	Description
interactionId	string	The unique identifier for the interaction.

startCallRecording

Name	Type	Description
interactionId	string	The unique identifier for the interaction.

stopCallRecording

Name	Type	Description
interactionId	string	The unique identifier for the interaction.

isMicrophoneMute

Signature	isMicrophoneMute()
Description	Get the mute state of the microphone of the SIP Endpoint.
Parameters	None.

muteMicrophone

Signature	muteMicrophone()
Description	Mute the microphone of the SIP Endpoint.
Parameters	None.

unmuteMicrophone

Signature	unmuteMicrophone())
Description	Unmute the microphone of the SIP Endpoint.
Parameters	None.

isSpeakerMute

Signature	isSpeakerMute()
Description	Get the mute state of the speaker of the SIP Endpoint.
Parameters	None.

muteSpeaker

Signature	muteSpeaker()
Description	Mute the speaker of the SIP Endpoint.
Parameters	None.

unmuteSpeaker

Signature	unmuteSpeaker())
Description	Unmute the speaker of the SIP Endpoint.
Parameters	None.

Gplus Adapter (v8) for Salesforce

Genesys Engage cloud offers two different ways for your Salesforce users to handle contact center interactions seamlessly within Salesforce. Both methods provide a single integrated agent desktop experience that presents complete customer information at a glance to more effectively serve customers in Salesforce.

- Gplus Adapter for Salesforce
- Agent Desktop in Salesforce Console

Gplus Adapter for Salesforce

Available in Salesforce Classic and Salesforce Console, the adapter provides voice and chat functionality, along with Salesforce-specific features such as updating activity history, screen pops, and click-to-dial.

Browser Support

The adapter supports the following web browsers:

- Microsoft Internet Explorer 10 and 11
- Google Chrome 38+
- Firefox 34+

To install and configure Gplus Adapter for Salesforce in your Salesforce environment, complete the procedure explained here.

Agent Desktop in Salesforce Console

Available in Salesforce Console, it provides Salesforce-specific features such as updating activity history, screen pop, and click-to-dial, along with the full Agent Desktop user interface and the following features:

- Voice
- Chat
- Email
- Outbound Preview
- Voice and Chat Supervision (monitoring, coaching, barge-in)

Browser Support

Agent Desktop in Salesforce Console supports the following web browsers:

- Microsoft Internet Explorer 11+
- Google Chrome 38+
- Firefox 34+

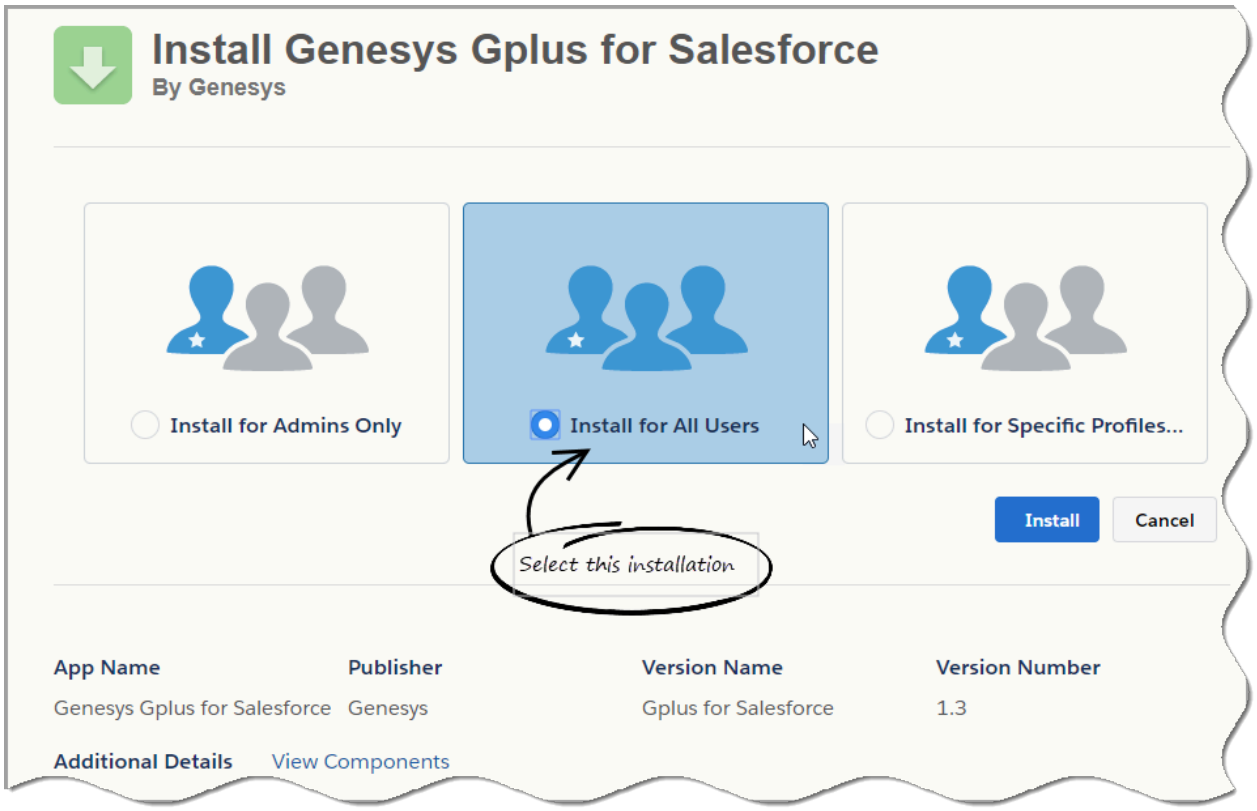
To deploy Agent Desktop in Salesforce Console, complete the procedure explained here.

Installing and configuring the Adapter (v8) in Salesforce

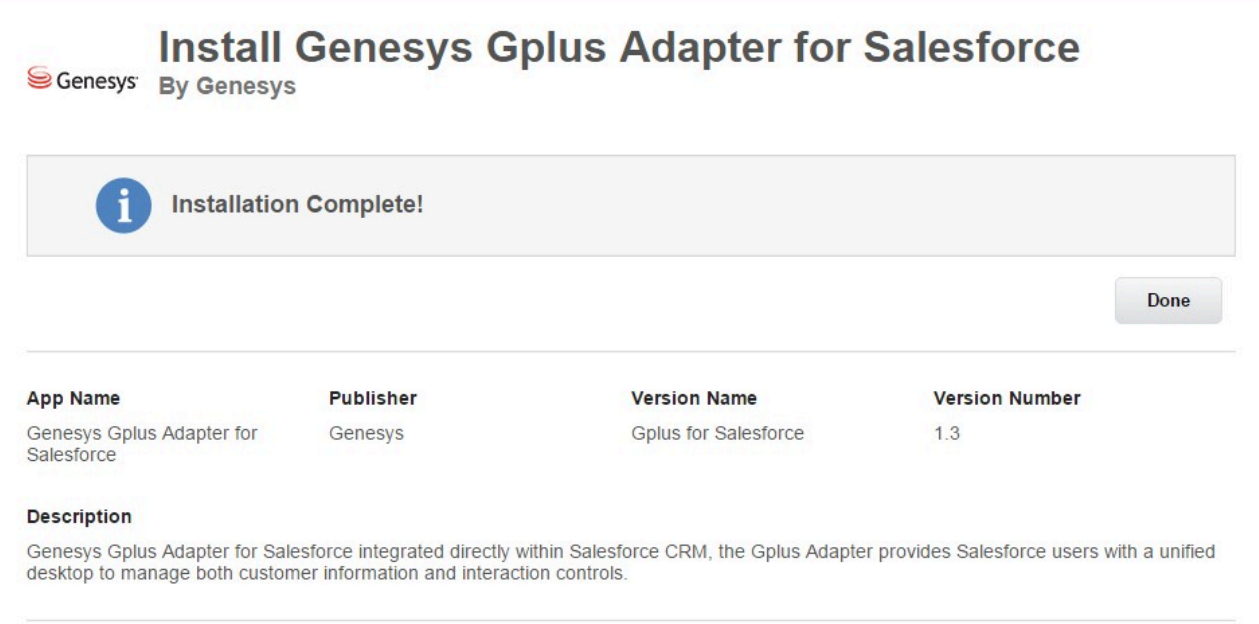
Complete the procedures on this page to install and configure the Gplus Adapter for Salesforce in your Salesforce environment.

Installing the Adapter in Salesforce

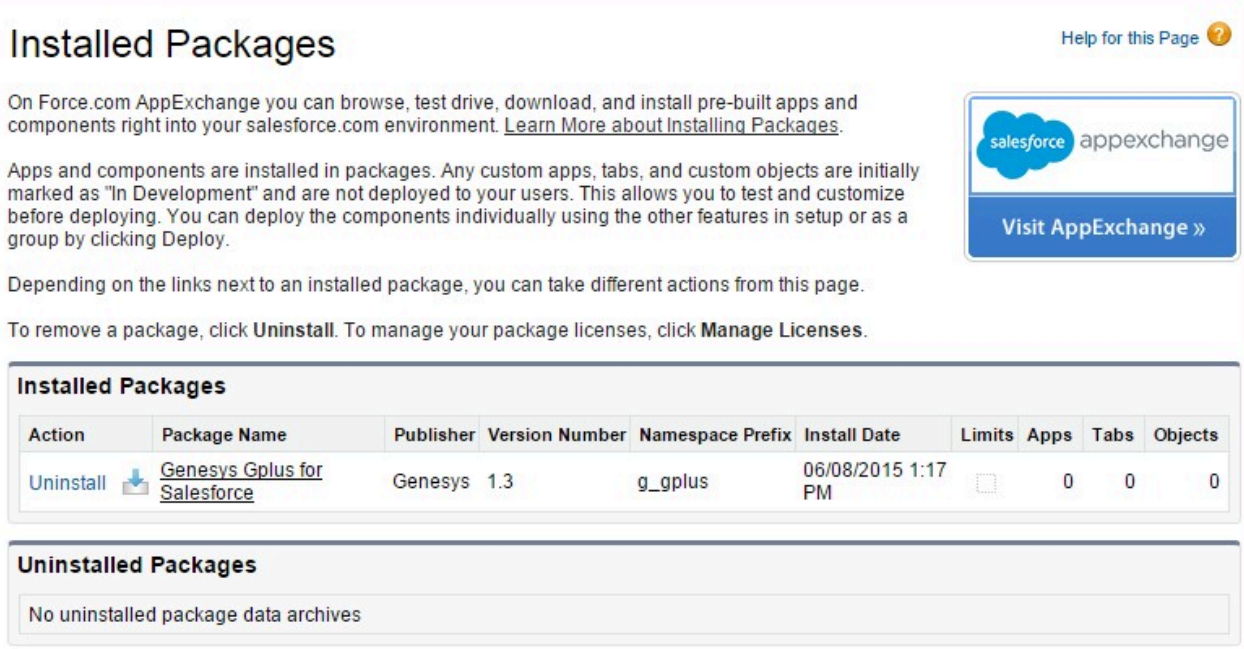
1. Open the following URL to install the latest Gplus Adapter for Salesforce package in Salesforce:
   https://login.salesforce.com/packaging/installPackage.apexp?p0=04to0000000C3VD
   If you're not logged in, Salesforce prompts for your username and password.
2. Now you should see the **Install Genesys Gplus Adapter for Salesforce** page. Select an installation type. Generally, you should select **Grant access to all users**, but if you want to limit access to the adapter to specific profiles, then you can choose **Install for Specific Profiles ...**. Click **Install**.



3. When you see the "Installation Complete!" message, click **Done**.



You should be redirected to the **Installed Packages** page, with "Genesys Gplus for Salesforce" included in the list.



Configuring the Adapter in Salesforce

Complete this procedure to define your call center in Salesforce. The call center was created when you installed the Gplus Adapter for Salesforce package as part of Installing the Adapter in Salesforce.

> ## Warning
> If no CRM is specified, screen pop and other integration features will be inactive.

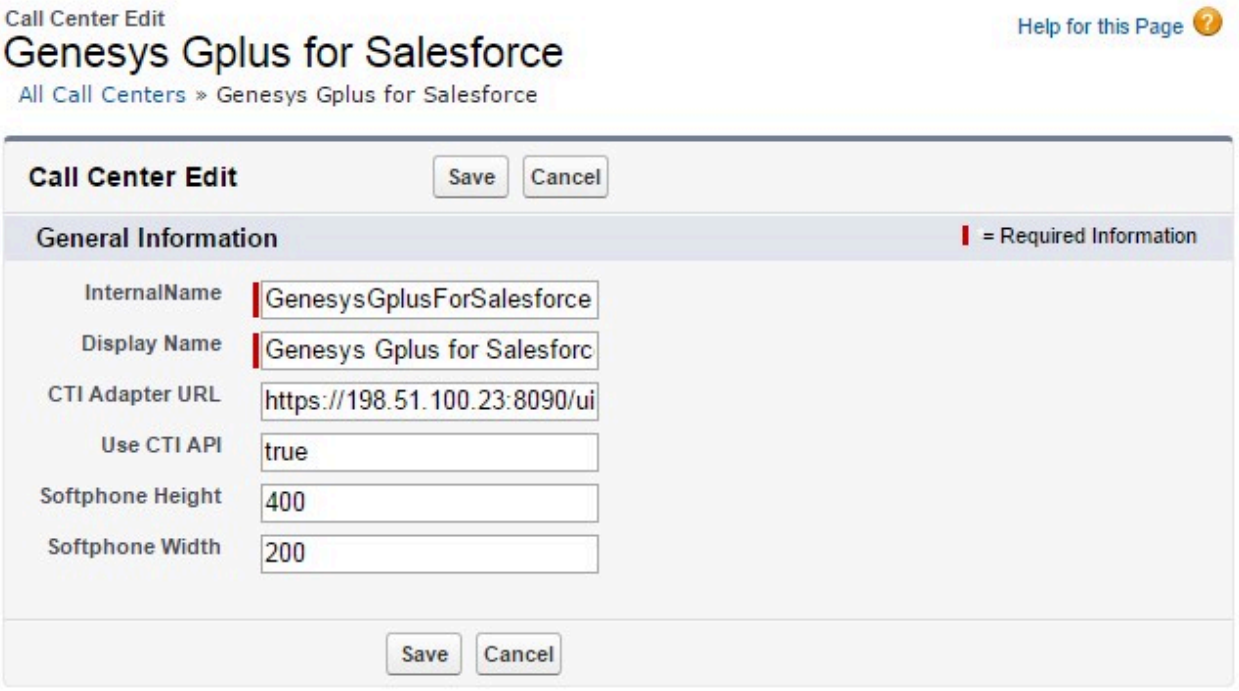
1. If you haven't already, login to Salesforce and go to **Setup > Build > Customize > Call Center > Call Centers**. Or, you can search for "Call Centers" in the **Search All Setup** field and select the "Call Centers" result. You should see the **Introducing Salesforce CRM Call Center** page. **Note:** You must have administrator privileges.
2. You can select **Don't show me this page again** if you want to hide the page in the future, and click **Continue**.
3. On the **All Call Centers** page, click **Edit** next to the Genesys Gplus for Salesforce entry.
4. In the **CTI Adapter URL** field, replace `GWS_HOST:GWS_PORT` with the correct host and port provided by Genesys. For example:

   ```
   https://198.51.100.23:8090/ui/crm-adapter/index.html?crm=salesforce
   ```

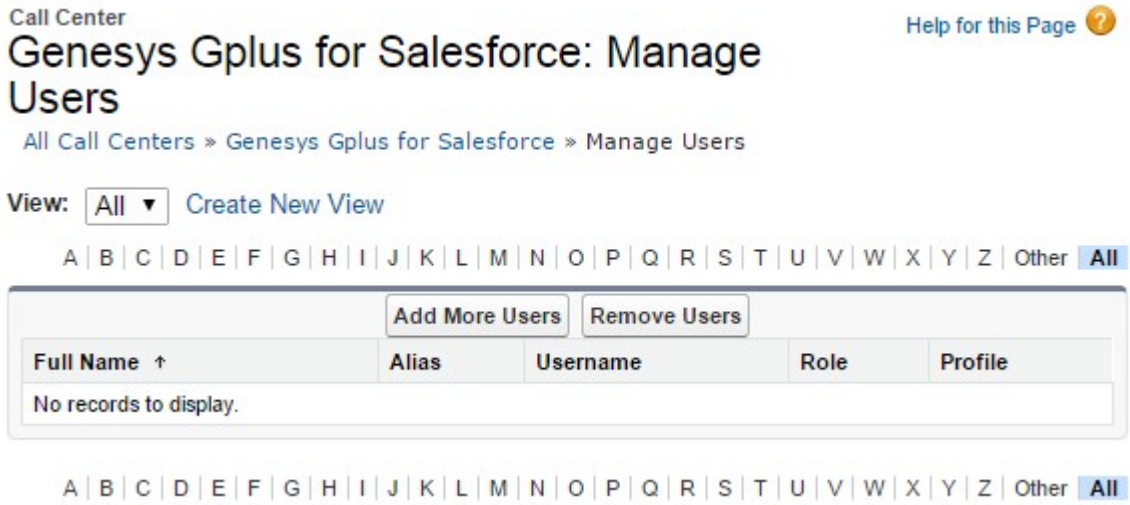
   If you're enabling single sign-on in the adapter, add the `authType=saml` parameter to the **CTI Adapter URL**. For example:

   ```
   https://198.51.100.23:8090/ui/crm-adapter/index.html?crm=salesforce&authType=saml
   ```

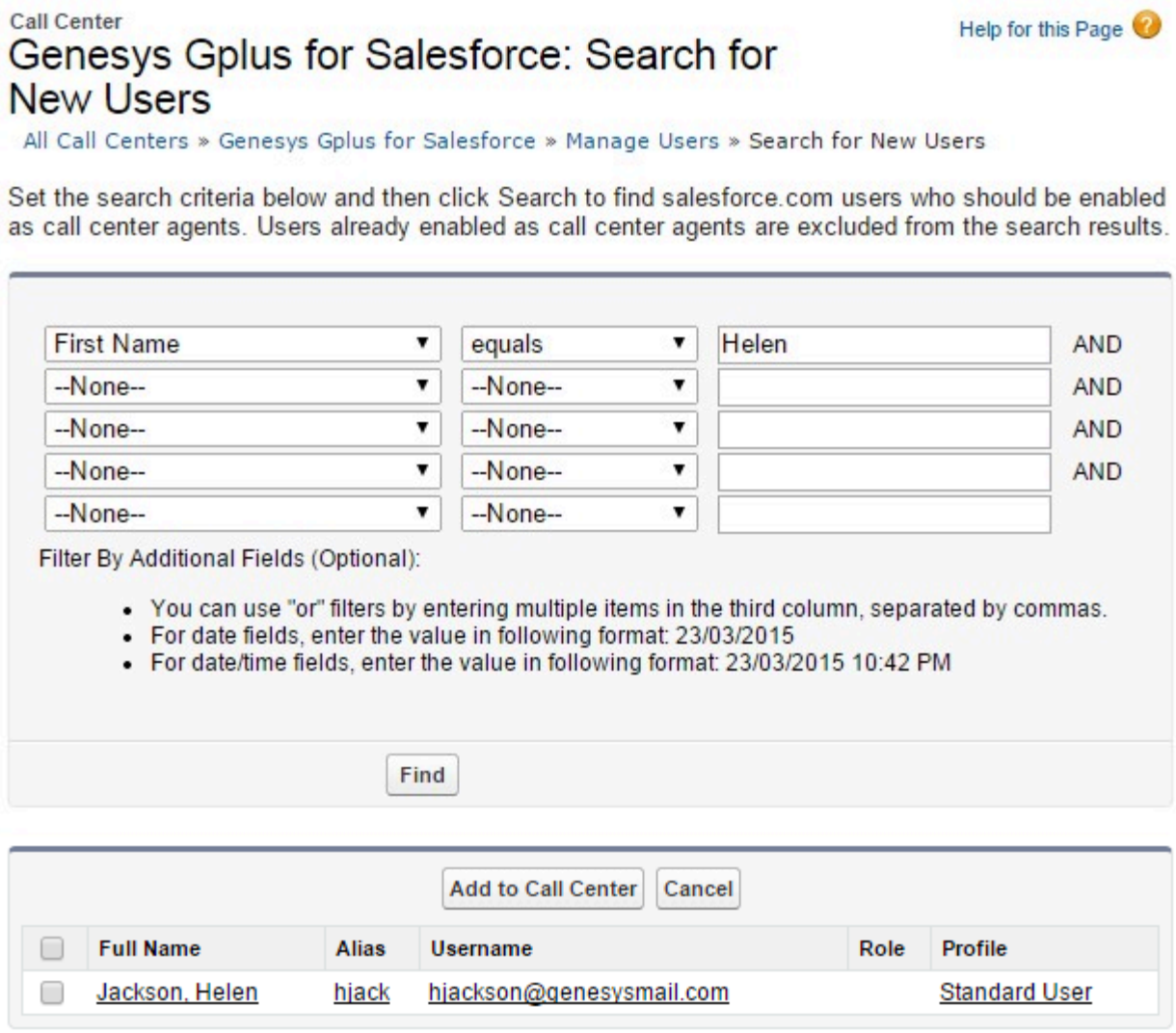
   You should leave the other options at their default values so the adapter works correctly in Salesforce.



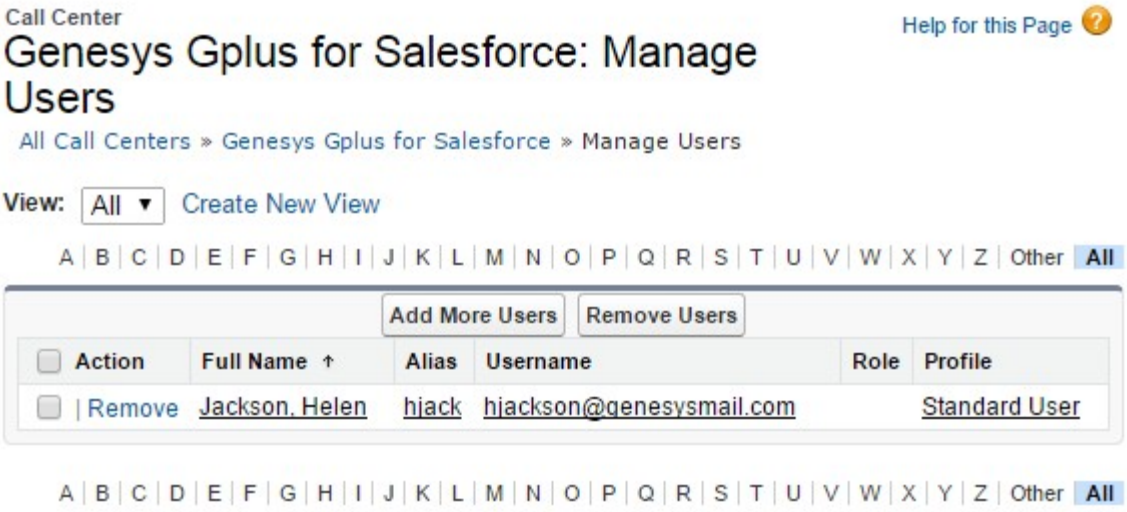
5. Click **Save**.
6. Click **Manage Call Center Users** and then click **Add users**.



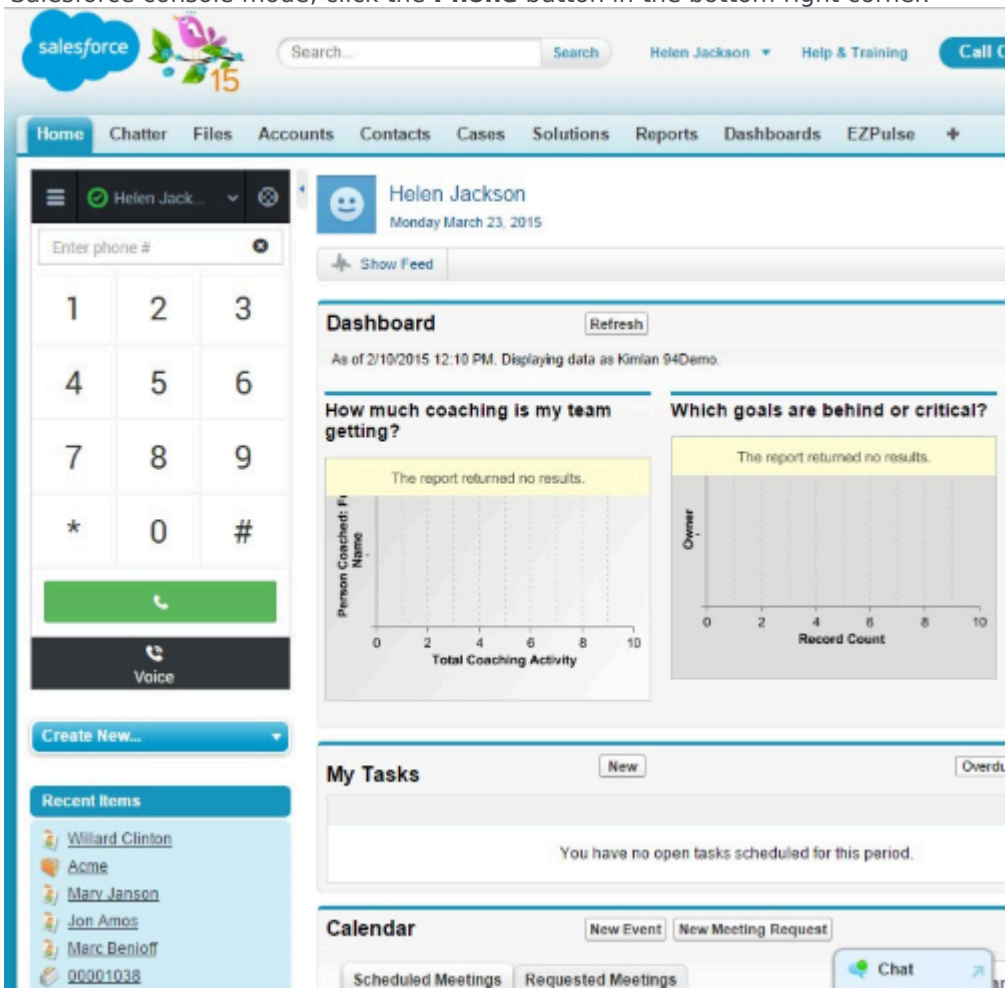
7. On the **Search for New Users** page, you can enter search criteria to find users. Select the ones you want to be able to use the adapter and click **Add to Call Center**.



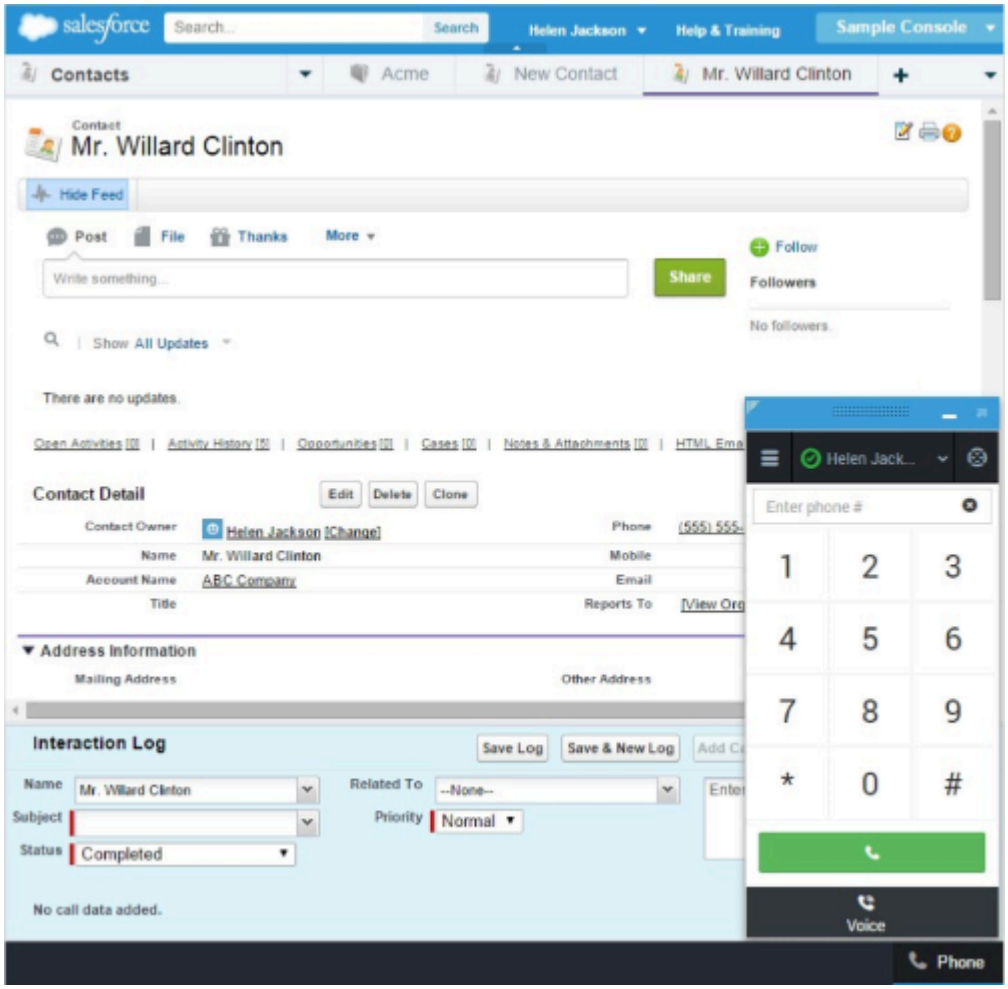
Your selected users are added to the list. You can remove a user on this page at any time.



8. To access the adapter in Salesforce standard mode, look for it in the left pane of your browser; in Salesforce console mode, click the **Phone** button in the bottom right corner.



The adapter in Salesforce standard mode.



The adapter in Salesforce console mode.

Configuring screen pops in Salesforce

When an agent receives an external call, the adapter can initiate a screen pop that causes Salesforce to show an appropriate record for the caller. To set up this functionality in Salesforce, login and go to **Setup > Customize > Call Center > SoftPhone Layouts** to create a SoftPhone Layout. Check out the Salesforce documentation for details about configuration.

In general, there are a couple of things to consider when you set up a SoftPhone Layout for the adapter:

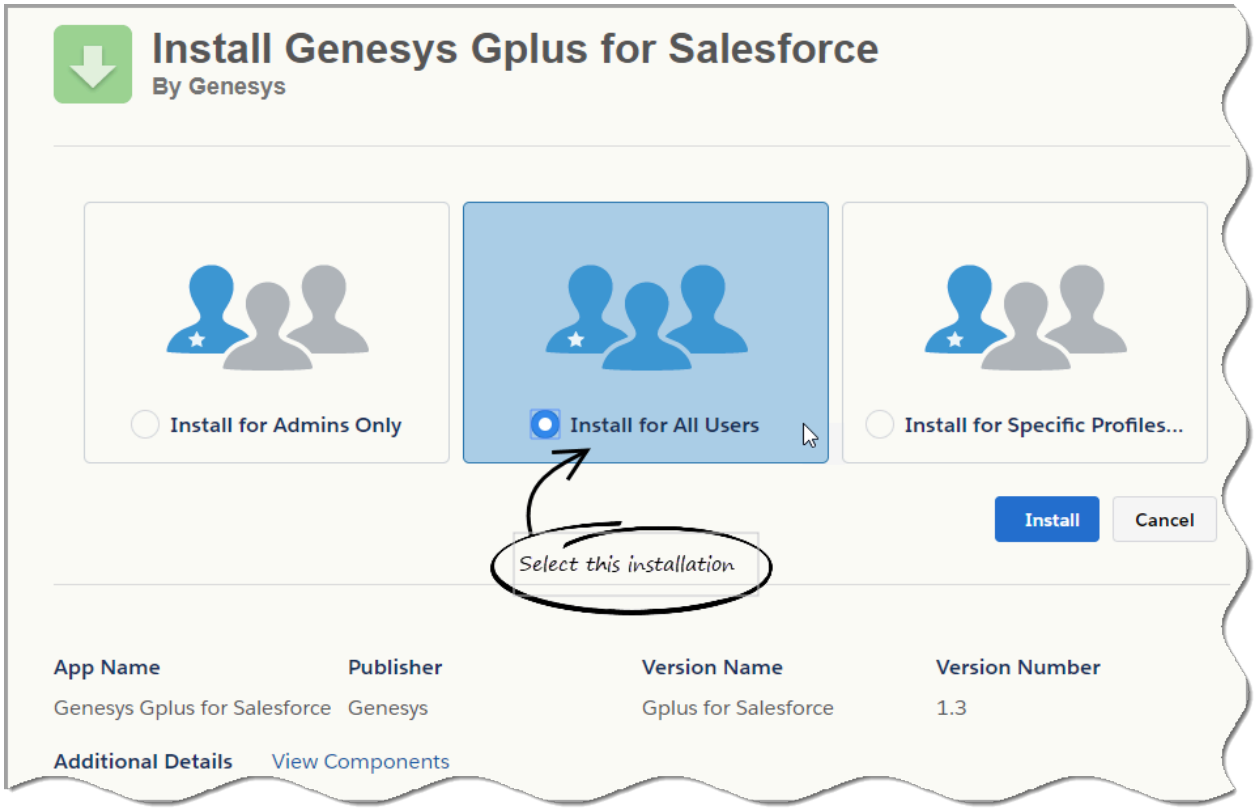
- The Gplus Adapter for Salesforce ignores the SoftPhone Layout settings that control call-related fields. Instead, the adapter gets this information from Toast and Case Data configured in your Genesys environment.
- Make sure you configure the **Screen Pop Settings** in the "CTI 2.0 or Higher Settings" section. These settings control whether the screen pop opens in a new window, tab, or Visualforce page.

Installing and Configuring Agent Desktop (v8) with Salesforce Console

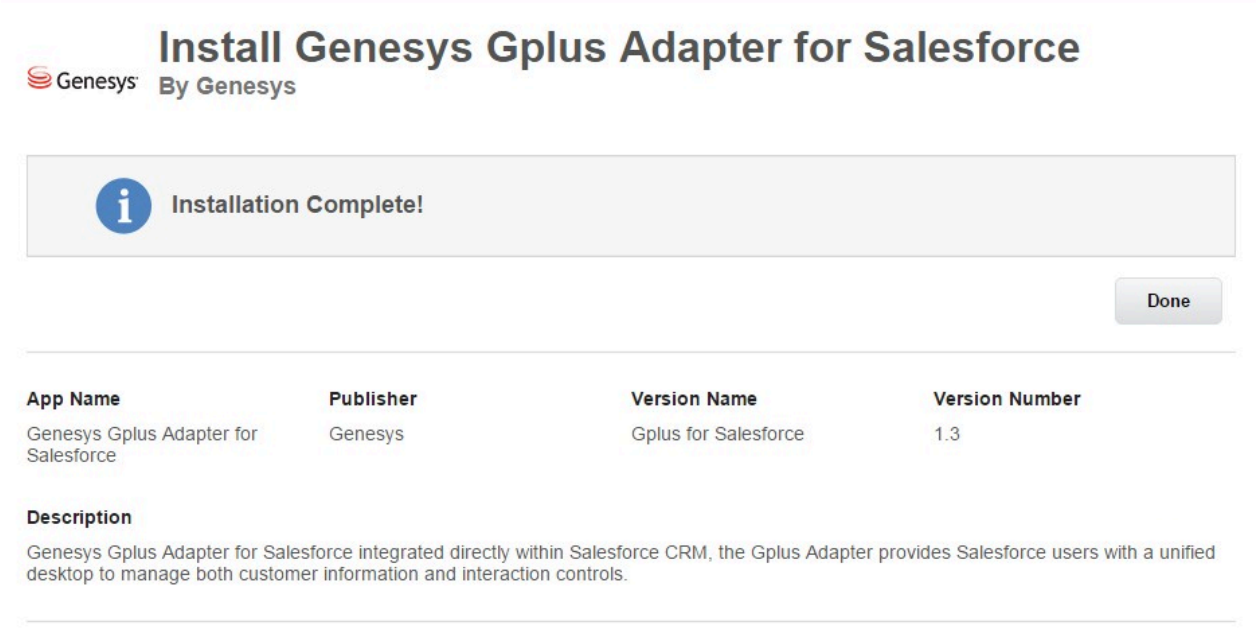
Complete the procedures on this page to install and configure Agent Desktop (v8) in Salesforce Console.

Installing Agent Desktop in Salesforce

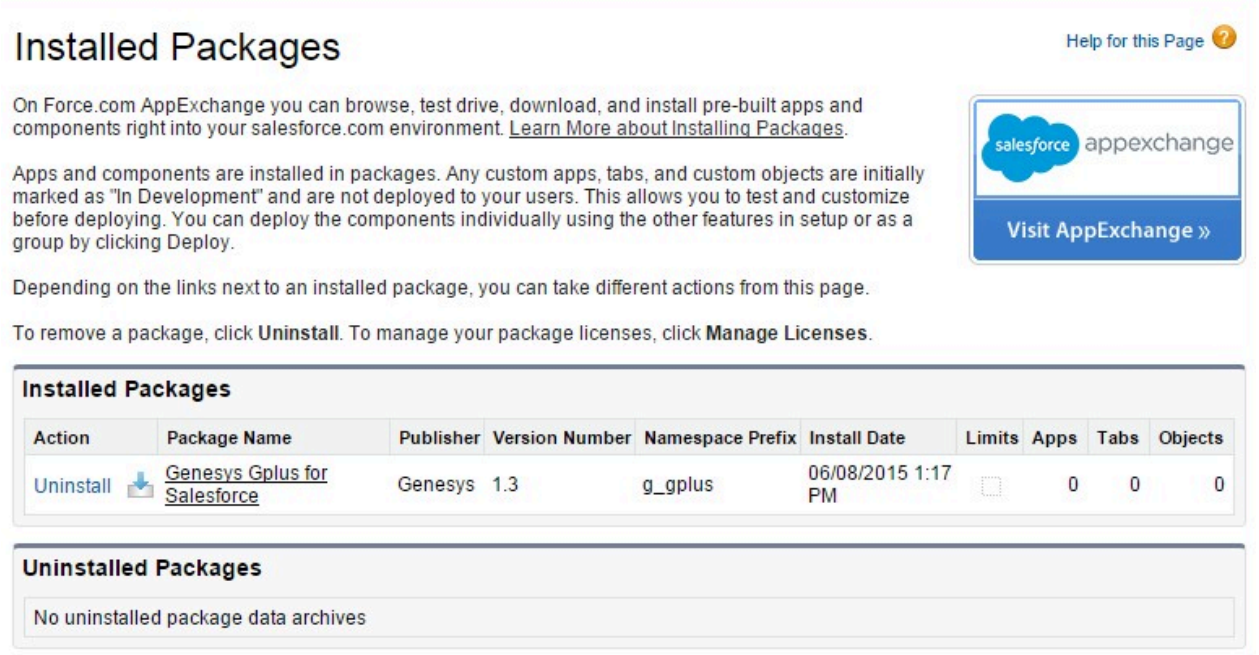
1. Open the following URL to install the latest Genesys Gplus Adapter (v8) for Salesforce package (this package is used for both the Agent Desktop and the Gplus Adapter) in Salesforce:
   https://login.salesforce.com/packaging/installPackage.apexp?p0=04to0000000C3VD
   If you're not logged in, Salesforce prompts for your username and password.
2. Now you should see the **Install Genesys Gplus Adapter for Salesforce** page. Select an installation type. Generally, you should select **Grant access to all users**, but if you want to limit access to Agent Desktop to specific profiles, then you can choose **Install for Specific Profiles ...**. Click **Install**.



3. When you see the "Installation Complete!" message, click **Done**.



You should be redirected to the **Installed Packages** page, with "Genesys Gplus for Salesforce" included in the list.



Configuring Salesforce for Agent Desktop

Complete this procedure to define your call center in Salesforce. The call center was created when you installed the Gplus Adapter for Salesforce package as part of Installing Agent Desktop in Salesforce.

> ## Warning
> If no CRM is specified, screen pop and other integration features will be inactive.

1. If you haven't already, login to Salesforce and go to **Setup > Build > Customize > Call Center > Call Centers**. Or, you can search for "Call Centers" in the **Search All Setup** field and select the "Call Centers" result. You should see the **Introducing Salesforce CRM Call Center** page. **Note:** You must have administrator privileges.
2. You can select **Don't show me this page again** if you want to hide the page in the future, and click **Continue**.
3. On the **All Call Centers** page, click **Edit** next to the Genesys Gplus for Salesforce entry.
4. In the **CTI Adapter URL** field, replace the text with the following URL:

   ```
   https://''GWS_HOST:GWS_PORT''/ui/crm-workspace/index.html
   ```

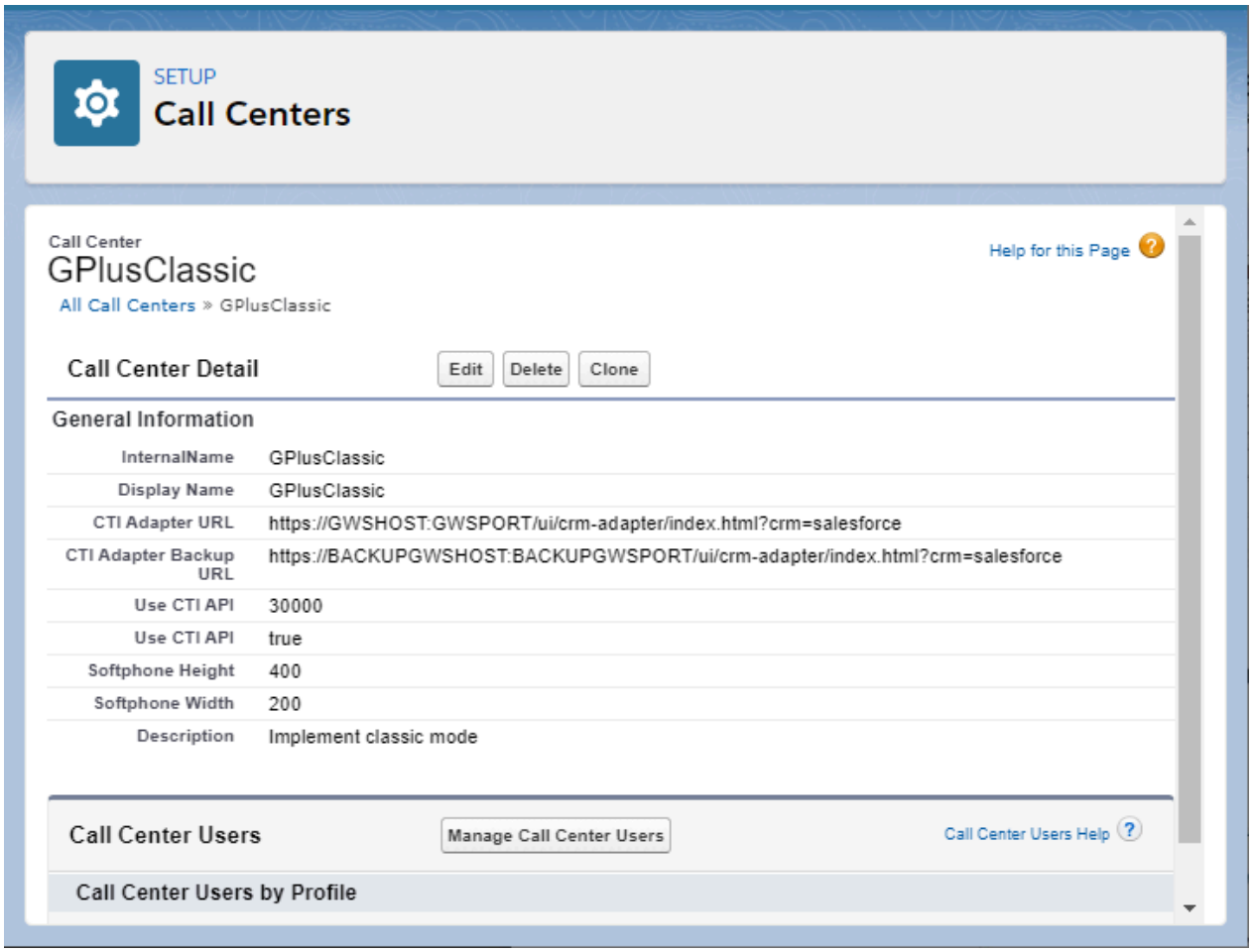
   You'll need to change *GWS_HOST:GWS_PORT* to the correct host and port provided by Genesys. For example:

   ```
   https://198.51.100.23:8090/ui/crm-workspace/index.html
   ```

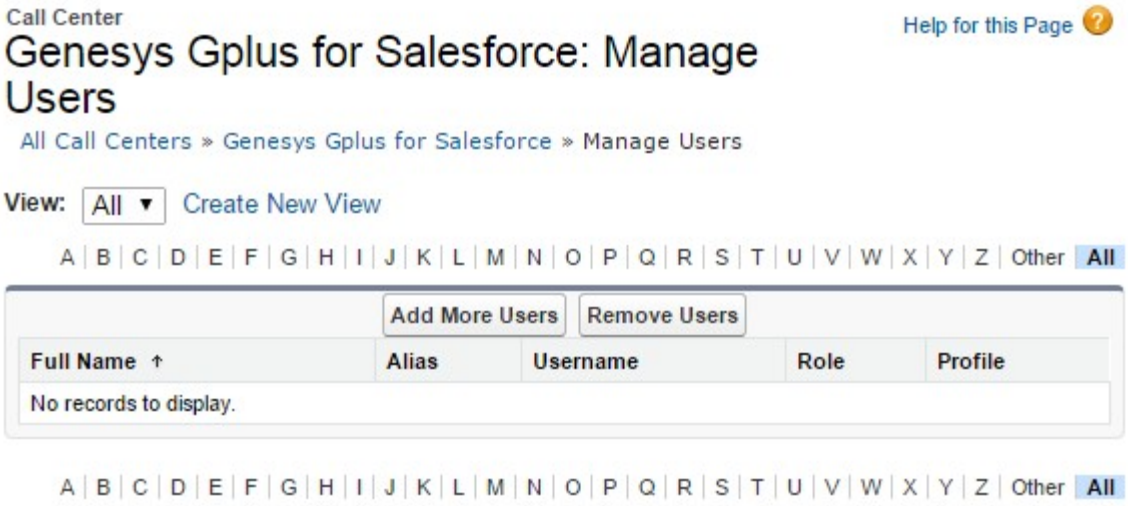
   If you're enabling single sign-on in the adapter, add the `authType=saml` parameter to the **CTI Adapter URL**. For example:

   ```
   https://198.51.100.23:8090/ui/crm-workspace/index.html&authType=saml
   ```

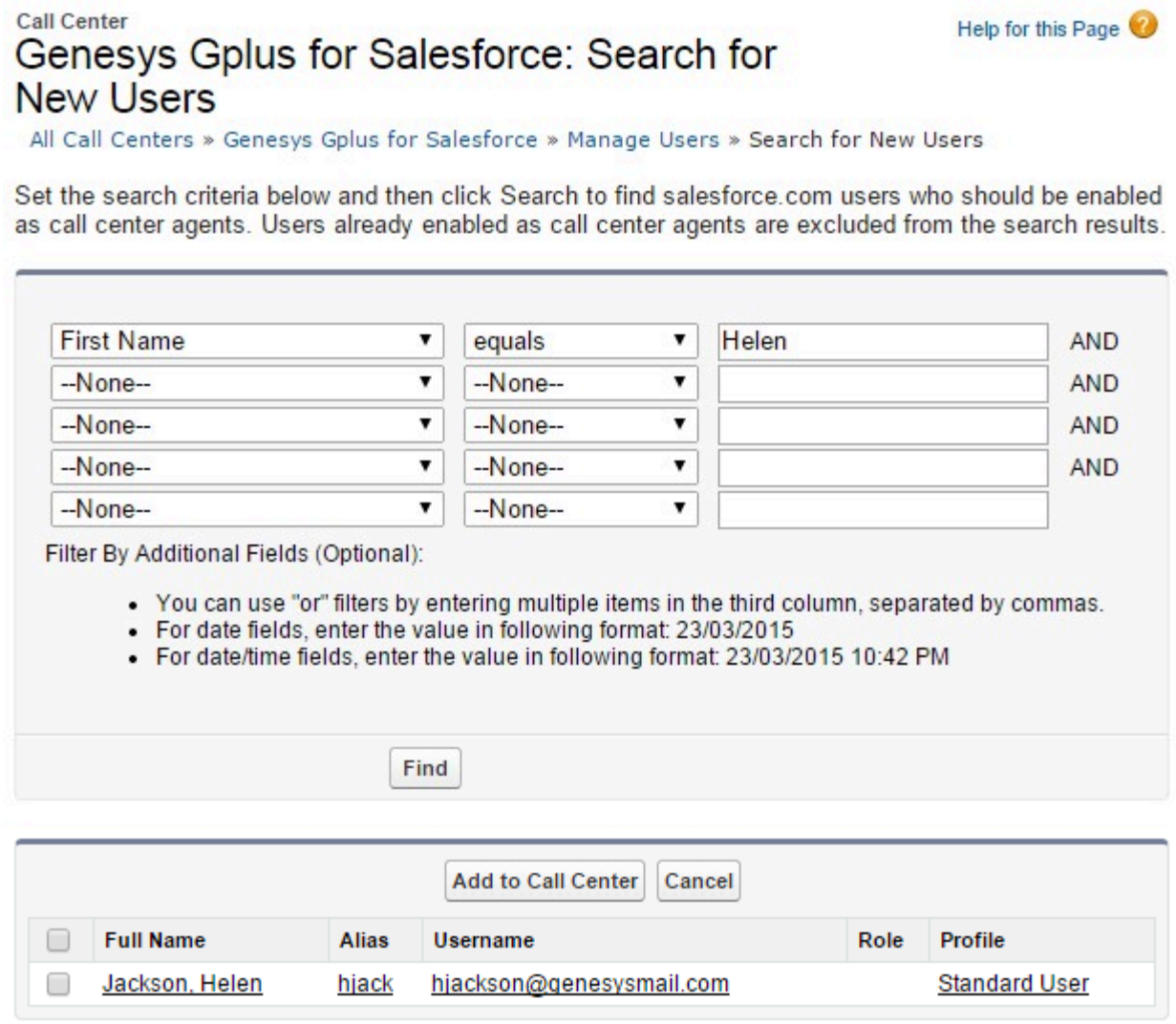
5. You might also want to adjust **Softphone Height** and **Softphone Width** to larger numbers (in pixels) so that Agent Desktop displays at an adequate size by default.
   You should leave the other options at their default values so Agent Desktop works correctly in Salesforce.



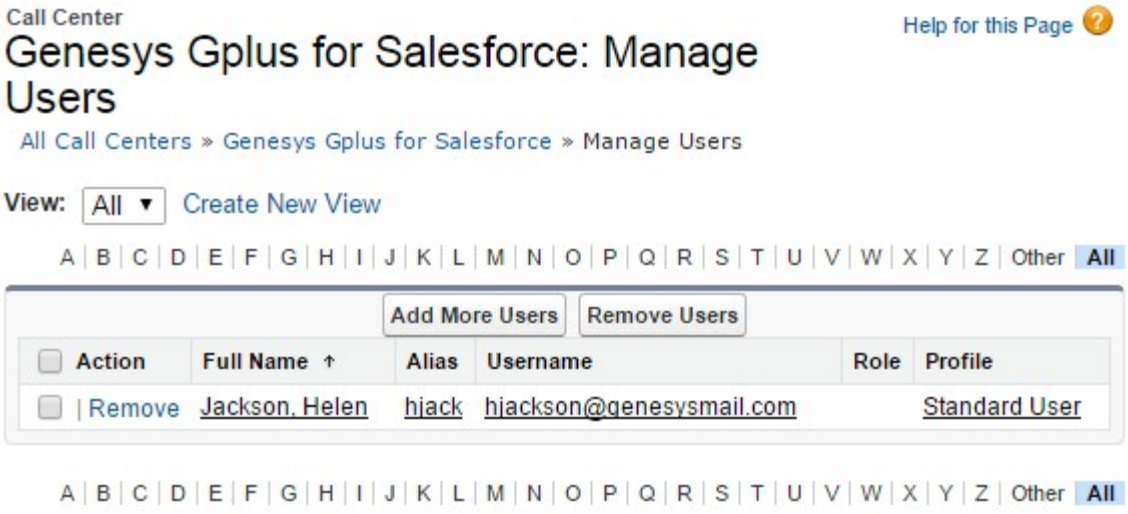
6. Click **Save**.
7. Click **Manage Call Center Users** and then click **Add users**.



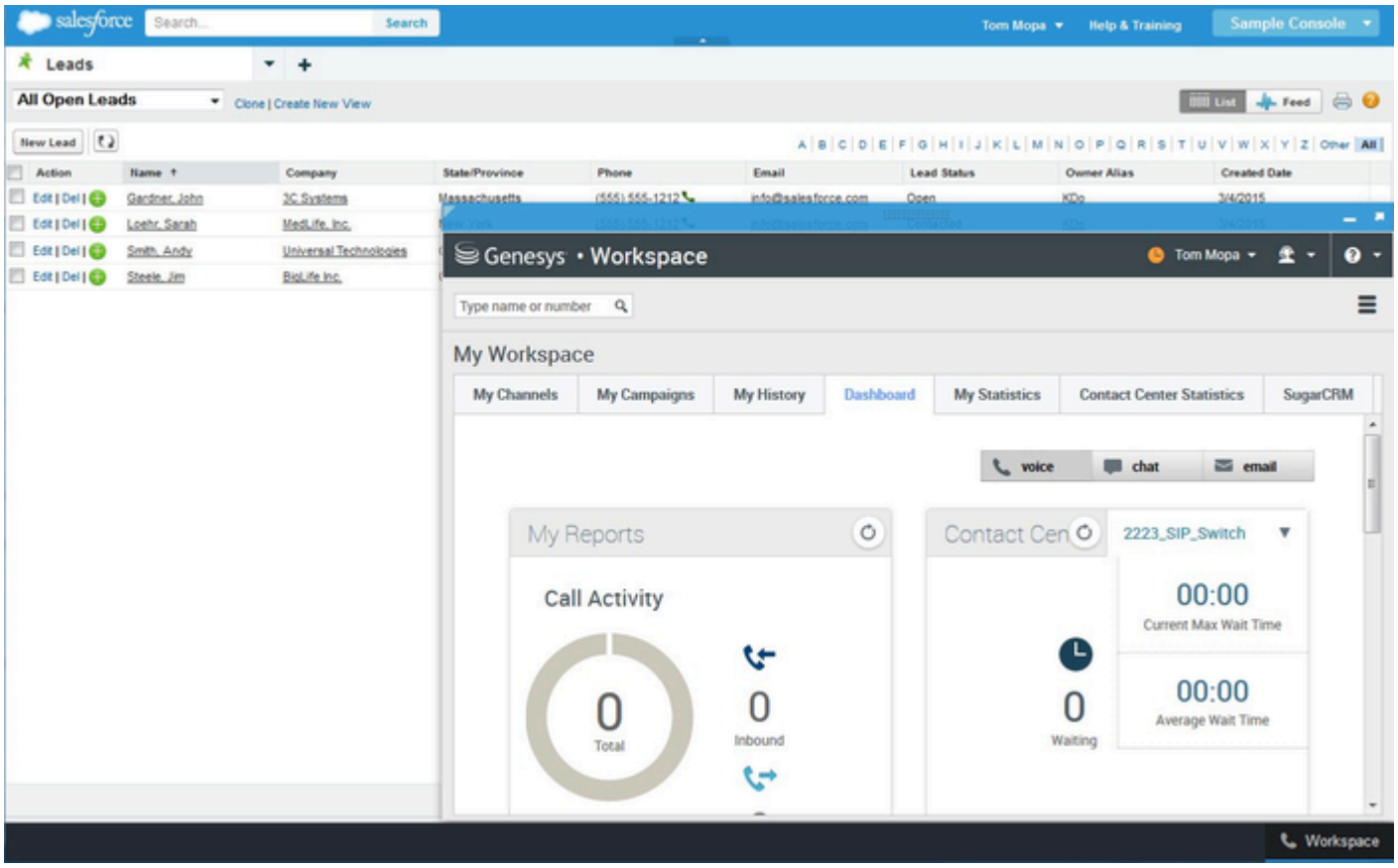
8. On the **Search for New Users** page, you can enter search criteria to find users. Select the ones you want to be able to use Agent Desktop and click **Add to Call Center**.



Your selected users are added to the list. You can remove a user on this page at any time.



9. To access Agent Desktop in Salesforce Console, click the **Workspace** button in the bottom right corner.

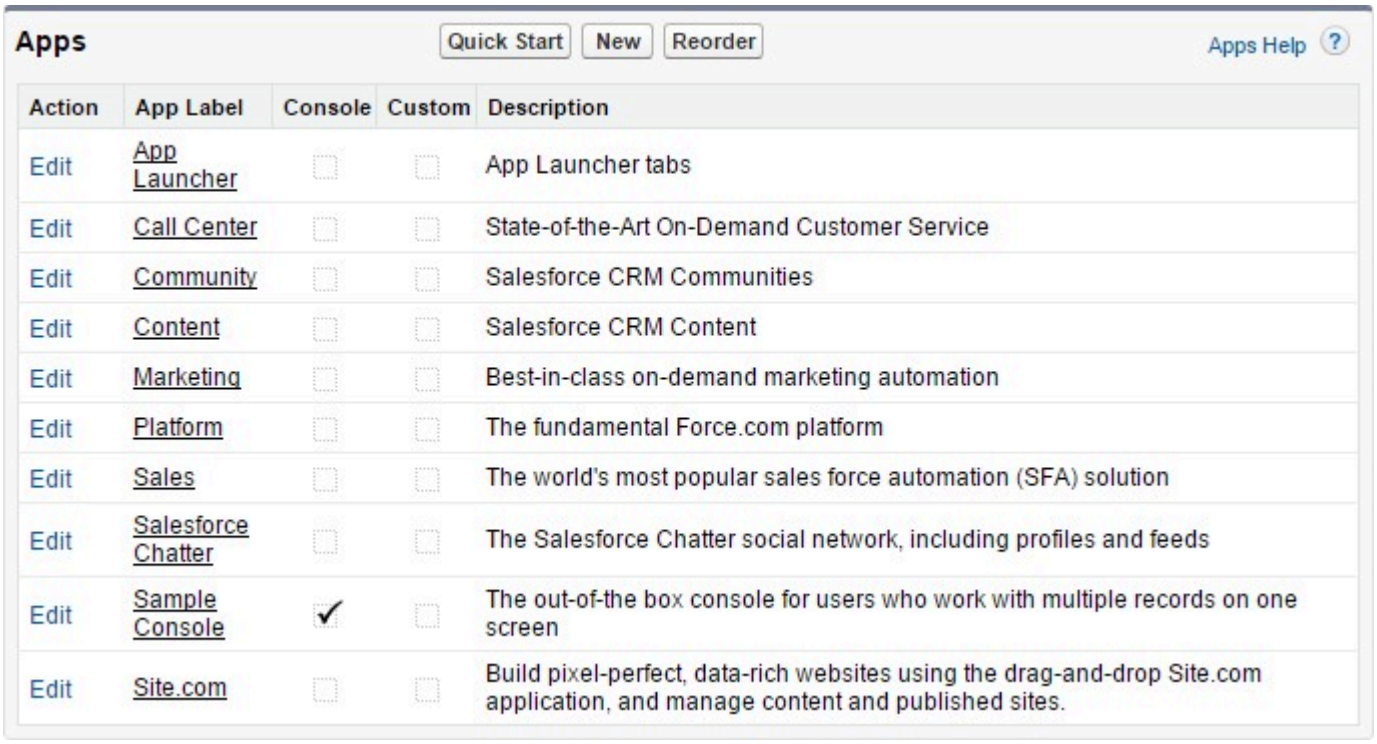


Agent Desktop in Salesforce Console.

Configuring the Whitelist Domain for your Salesforce Console

Complete this procedure to add the Genesys domain to the whitelist domains for your Salesforce Console. You need to complete this procedure to allow your users to access Agent Desktop in Salesforce Console in a separate browser window.

1. If you haven't already, login to Salesforce and go to **App Setup > Create > Apps** and select your console app — "Sample Console" in the image below:



2. Click **Edit**. In **Whitelist Domains**, add the host and port provided by Genesys in Step 4 of Configuring Salesforce for Agent Desktop. For example: `198.51.100.23:8090`
3. Click **Save**.

Configuring Screen Pops in Salesforce

When an agent receives an external call, Agent Desktop can initiate a screen pop that causes Salesforce to show an appropriate record for the caller. To set up this functionality in Salesforce, login and go to **Setup > Customize > Call Center > SoftPhone Layouts** to create a SoftPhone Layout. Check out the Salesforce documentation for details about configuration.

In general, there are a couple of things to consider when you set up a SoftPhone Layout for the Agent Desktop:

- Agent Desktop ignores the SoftPhone Layout settings that control call-related fields. Instead, the Agent

Desktop gets this information from Toast and Case Data configured in your Genesys environment.

- Make sure you configure the **Screen Pop Settings** in the "CTI 2.0 or Higher Settings" section. These settings control whether the screen pop opens in a new window, tab, or Visualforce page.

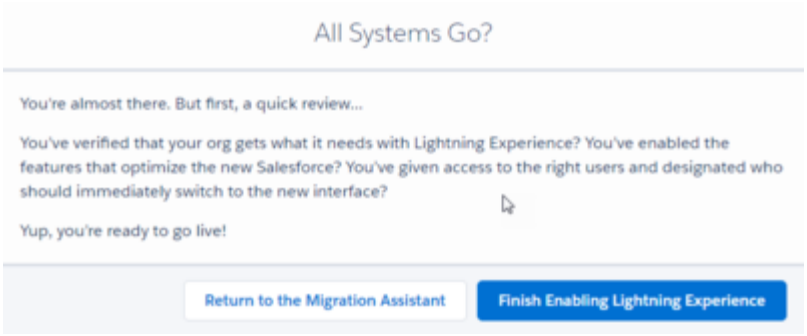
Enabling Lightning Experience (Adapter v8)

If you're using the Agent Desktop (v8) option, complete the procedures on this page to enable, set up, and access Lightning in your Salesforce environment.

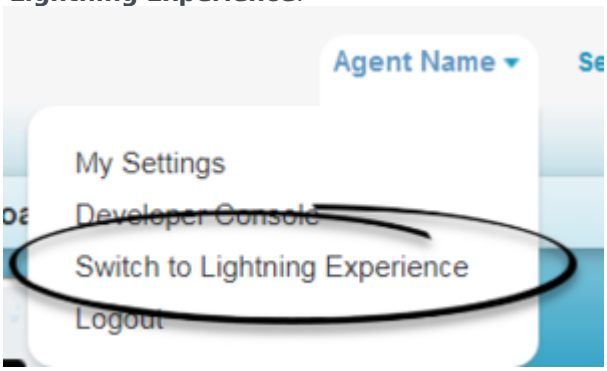
Enabling Lightning in Salesforce

1. Log into the Salesforce environment.
2. From the **Setup** page, select **Lightning Experience** in the left-hand navigation bar. **Note:** If in Salesforce Classic mode, click the **Setup** menu and then the **Get Started** button found in the left-hand navigation bar.
3. In the **Lightning Experience** window, select **Turn It On**.
4. Move the toggle to the **Enabled** state.

5. A modal will pop up; click the **Finish Enabling Lightning Experience** button in the modal.

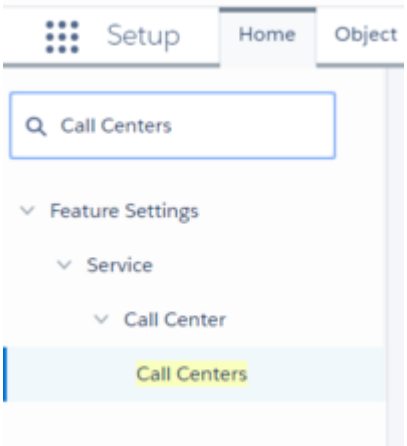


6. In the dropdown labeled with the agent's name at the top of the Salesforce classic view, click **Switch to Lightning Experience**.



Setting Up The Adapter In Lightning

1. Download the **lightning-callcenter.xml** file on your computer by right-clicking the link here and selecting the **Save link as...** option on the popup menu.
2. Go to the **Setup** page by clicking on the gear icon in the top right corner and clicking **Setup**.
3. Using the quick find field, search for and access the **Call Centers** settings page.



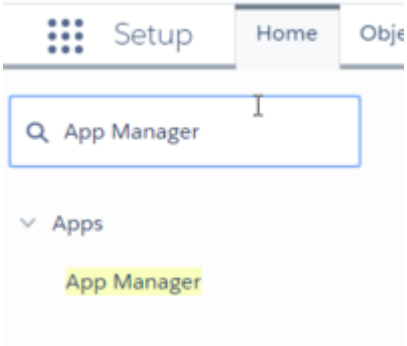
4. Click **Import**.

5. Click **Choose File**.
6. Select the **lightning-callcenter.xml** file downloaded on your computer. If you have not already downloaded the file, right-click the link here and select the **Save link as...** option to download.
7. Click **Import**.
8. From **All Call Centers** list, click the call center you just imported. For example, **GPlusLightning**.
9. Click **Edit**.
10. In the **CTI Adapter URL** field, replace 'GWSHOST' and 'GWSPORT' with the host and port details of the adapter in your environment. For example, an updated URL will look like this: `https://bec135-gws.live.genesys.com/ui/crm-workspace/index.html?crm=lightning`
    Note: If you are deploying the adapter with Single-Sign-On (SSO) capability, ensure that you add the `&authType=saml` parameter at the end of the CTI Adapter URL. For example, an updated URL with SSO capability will look like this: `https://bec135-gws.live.genesys.com/ui/crm-workspace/index.html?crm=lightning&authType=saml`
11. Click **Save**.
12. Click **Manage Call Center Users**.
13. Click **Add More Users**.
14. Search the interface to find the users you want to add to the Lightning adapter.

> **Important**
> A user cannot be added to both the Lightning and non-Lightning adapters

.

15. Select the users you want to add and click **Add to Call Center**.
16. Using the quick find field, access the **App Manager** settings page.



17. In the apps list, click the **Show more actions** drop down on the far right side of the adapter app you wish to use.
    Note: If you do not see any apps in the list, you can create one by clicking **New Lightning App**.
18. Click **Edit**.
19. Click **Utility Bar**.
20. From the Utility Bar window, click **Add** and select "Open CTI Softphone".

21. Change the **Label** field to "Workspace".
22. Click **Done**.

Accessing the Adapter

1. In the top-left corner, click the **App Launcher** icon:
   
2. Select the app that you created when setting up the adapter.
3. Click **Workspace** from the bar at the bottom-left to open the adapter.
4. Log in to the Adapter.

Reporting

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

What kind of reports are available in Genesys Engage cloud?

To help you and the Team Leads who supervise your agents make informed, timely business decisions, Genesys Engage cloud offers both real-time **dashboard** views and historical operational performance **reports**. These tools are described in the *Reporting in the cloud Guide*.

Genesys Engage cloud reporting tools provide easy-to-read visual summaries of the activity in your contact center, including information about agent performance, queues, and detailed information about interactions.

- Current events — To see information about customer interactions that are currently taking place, you can use *Genesys Pulse*: Real-time reporting with Genesys Pulse
- Both Current and Historical — *GVP* provides both historical and up to the minute reporting on the state of your contact center: GVP reporting
- Historical information — *Genesys Customer Experience Insights* provides historical reporting — detailed information about what happened with customer interactions over specified time periods.
  - Historical reporting with Genesys CX Insights
  - Genesys Info Mart historical database

Both real-time and historical reports are available using Web-based technology, so you can access contact center performance data anywhere, anytime.

Real-time reporting with Pulse

Genesys Pulse provides at-a-glance views of real-time contact center statistics on dashboards within the user interface.

GVP reporting

Genesys Voice Platform has many moving parts—individual components that perform different tasks during a call. GVP reports give you a diagnostic look at your contact center's performance, by monitoring how the components perform — by themselves and with each other. The goal is to improve your contact center's efficiency.

This page discusses how you can perform the following tasks with Genesys Voice Platform reports:

- Browse the Call Detail Records (CDRs) of in-progress and completed calls, filtering for the different components that process them. You can observe how each component is performing.
- Gather statistics on events such as call arrivals, call durations, and peak call volume— then sort or filter by IVR Profile, or by GVP component.
- Observe the success and failure rates for calls, and the IVR Actions that handle them.

Historical reporting with Genesys CX Insights

Use historical reports to visualize the performance of your contact center over time (as contrasted against real-time reports, which show what is going on right now. Read more: a discussion of the difference.)

This page discusses historical reporting in Genesys CX Insights. For more information about Genesys CX Insights:

- To learn how to use Genesys CX Insights, including how to access, run, read, or customize historical reports, see the Genesys CX Insights page in the *Reporting in the cloud* guide.
- To learn how to manage Genesys CX Insights user accounts, see the Genesys CX Insights User Management page in this guide, which describes the steps you can take to create an account for a new user, change your password, or to change another user's password. Only Administrator users can manage credentials for other Genesys CX Insights (Historical Reporting) users, which are managed separately from other Genesys Engage cloud components.

Genesys Info Mart historical database

Genesys Info Mart is the enterprise-level Genesys application behind the historical reports in your cloud deployment. Genesys Info Mart receives interaction data from various upstream enterprise-level Genesys applications, then processes the low-level data to produce a data mart that Genesys Engage cloud uses for contact center historical reporting. Genesys CX Insights relies on Genesys Info Mart.

This page introduces the Genesys Info Mart Historical Database Reference.

How do I learn more about data behind reports?

The detailed data behind historical reports is fairly complex. If Data Export capability is available in your cloud deployment, use the following resources to learn about the dimensional model of the Genesys historical database (called the Info Mart database), from which fact and dimension data is

exported.

What is Data Export?

Data Export capability periodically copies the data that is stored in the Info Mart database into local .csv files, one file per table, so that the historical reporting data is available for further import into a data warehouse. Data Export capability is available in certain Genesys Engage cloud deployments; contact your Genesys representative for more information or see the Genesys Info Mart Physical Data Model document.

Historical reporting for Genesys Designer applications

Some Genesys Designer Analytics tool data is stored in the Genesys Info Mart database for historical reporting purposes. In Genesys Designer Analytics tool, this data is displayed on the Session Detail Records dashboard (login required).

This data is stored as Session Detail Records (SDR) in Genesys Info Mart. If the Data Export feature is available in your cloud deployment, the data from the **SDR_*** tables is included in the export.

Genesys CX Insights also provides IVR reports based on the SDR data.

Genesys CX Insights User Management

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys CX Insights User Management.

Genesys Customer Experience Insights (GCXI or Genesys CX Insights) provides Historical Reporting tools.

This page describes the steps you can take to create an account for a new user, change your password, or to change another user's password.

Managing your own account

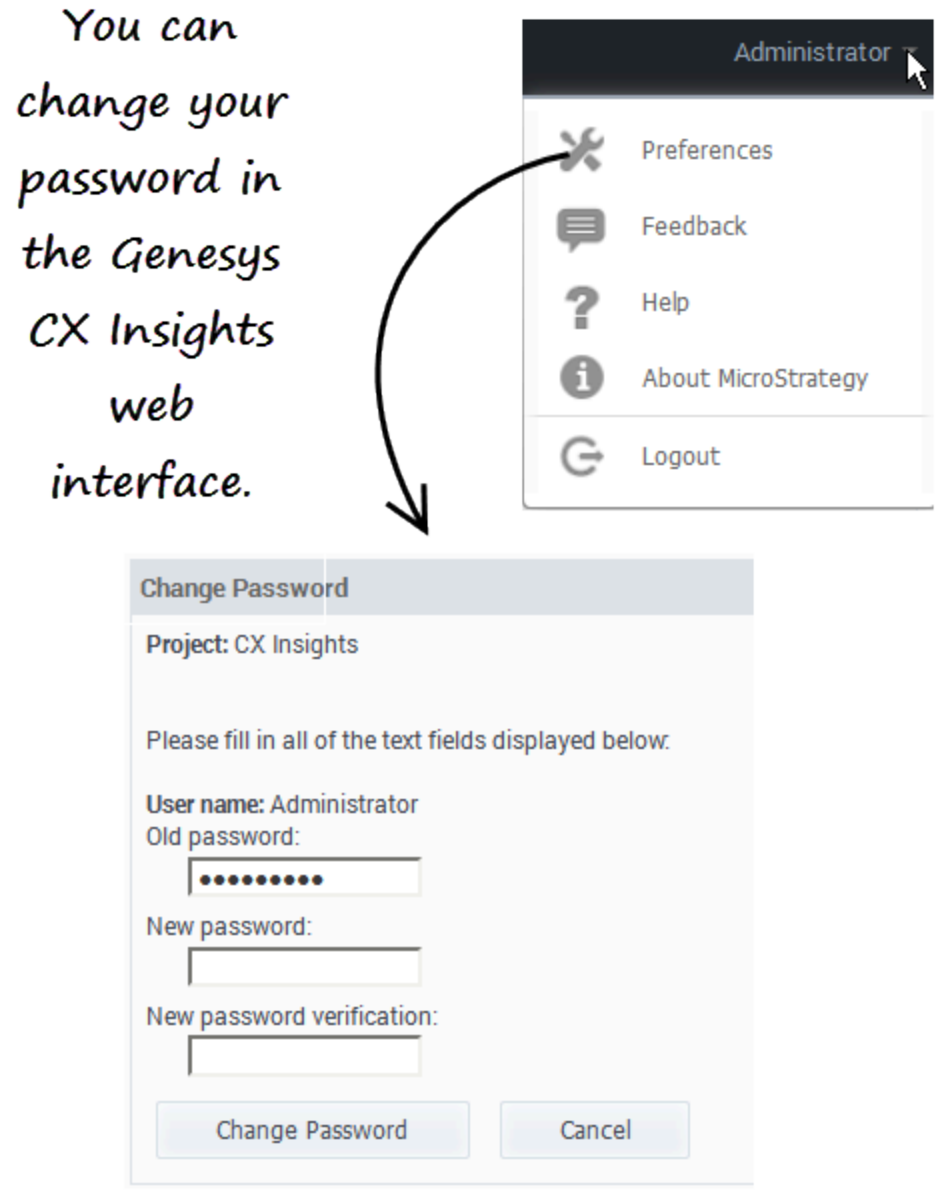
This section provides information about managing your own account.

Video: Changing your own password

Link to video

This video describes how to change your own password, if your permissions allow it.

Changing your own password



Use the following steps to change your password. Not all users are permitted to change their password; contact your administrator to find out if the functionality described on this page is available for your use.

> **Important**
> If you have forgotten your password, or otherwise cannot log in in, contact your

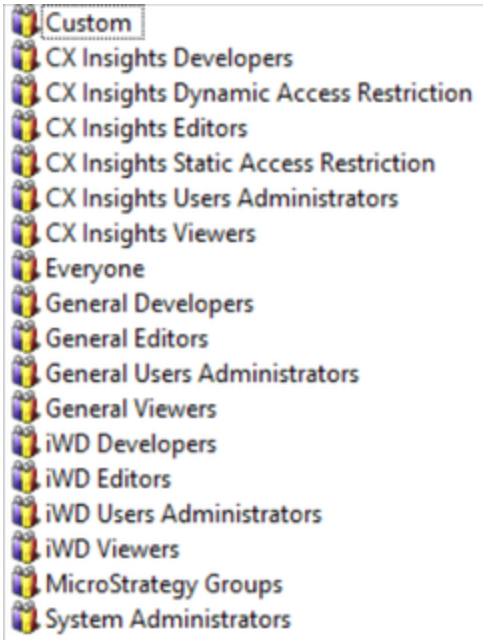
> administrator / next level of support.

1. On the Genesys Engage cloud Portal, click **All Apps** > **Reporting**.
2. At the Genesys CX Insights login screen, enter your user name and current password.
3. On the Genesys CX Insights menu, click your user name, and click **Preferences**.
4. Click **Change Password**.
5. In the **Old Password** field, type your current password.
6. In the **New Password** field, type your new password, and re-type it in the **New Password Verification** field.
   Passwords must be at least 8 characters, have at least one uppercase character, one lowercase character, and one number.
7. Click **Change Password**.

Managing other users

Most users cannot manage their own accounts. As an administrator, you can do so for them, as described in this section.

Predefined User Groups



Predefined User Groups

When you create a user, you must add them to at least one user group, thereby controlling the user's ability to work with reports and dashboards. Normally, you can do this by assigning users to the predefined user groups described in this section.

Predefined user groups and privileges

If you require permissions different from those assigned to the predefined groups, Genesys recommends that you avoid modifying privileges for the predefined user groups, because these user groups are overwritten during upgrades. Instead, create custom groups by duplicating the user group you wish to modify, and edit the duplicate group. Note that the group structure in release 9.0.011 and later is unlike earlier releases of the software, and each group contains only users (and does not contain other groups).

{The **User Groups** table lists and describes the predefined GCXI user groups.

User groups

Groups	Summary	Project Access Level*
Custom	Customer-defined user groups.	User-defined
CX Insights Developers	Members of this groups can create, edit, or view objects in the Genesys CX Insights project.	Genesys CX Insights
CX Insights Dynamic Access Restrictions	Security Filter you can use to restrict access to data based on	Genesys CX Insights

	user name, geographical location, line of business, or organizational role.	
CX Insights Editors	Members of this groups can edit or view objects in the Genesys CX Insights project.	Genesys CX Insights
CX Insights Static Access Restrictions	Security Filter you can use you prevent members of specified user groups from viewing data for a list of objects you specify.	Genesys CX Insights
CX Insights User Administrators	Members of this group can manage users in the Genesys CX Insights project.	Genesys CX Insights
CX Insights Viewers	Members of this groups can view objects in the Genesys CX Insights project.	Genesys CX Insights
Everyone	The *Everyone* group provides a way for you to easily apply privileges, security role memberships, or permissions to all users. All users are automatically members of this group.	none
General Developers	Members of this group can create, edit, and view objects in any project.	all
General Editors	Members of this group can edit and view objects in any project.	all
General User Administrators	Members of this group can manage users in any project.	all
General Viewers	Members of this group can view objects in any project.	all
iWD Developers	Members of this group can create, edit, and view objects in the CX Insights for iWD project.	iWD
iWD Editors	Members of this group can edit and view objects in the CX Insights for iWD project.	iWD
iWD User Administrators	Members of this group can manage users in the CX Insights for iWD project.	iWD
iWD Viewers	Members of this group can view objects in the CX Insights for iWD project.	iWD
MicroStrategy Groups	Built-in groups that are included in all MicroStrategy deployments.	none
System Administration	Members of this group have unrestricted management capabilities.	all

- Some groups provide access only to a specific project:
  - 'CX Insights' — membership in groups with this prefix allows users to work within the Genesys CX Insights project only.
  - 'General' — membership in groups with this prefix allows users to work in any project.
  - 'iWD' — membership in groups with this prefix allows users to work within the iWD project only.

Single sign-on

When enabled, Single-sign on (SSO) allows logged-in users to navigate across multiple applications without re-entering their credentials.

The **Configuration Server Access Groups** table shows how Genesys CX Insights user groups map to Configuration Server Access Groups:

Configuration Server Access Groups

Parent folder	Config Server Access Group	Mapping to MicroStrategy Users Groups
environment/Access Groups/$customer/	$customer Administrators	CX Insights Developers, CX Insights Users Administrators
environment/Access Groups/$customer/	$customer Managers	CX Insights Editors
environment/Access Groups/$customer/	$customer Supervisors	CX Insights Viewers
environment/Access Groups/$customer/GCXI	CX Insights Users Administrators	CX Insights Users Administrator
environment/Access Groups/$customer/GCXI	CX Insights Developers	CX Insights Developers
environment/Access Groups/$customer/GCXI	CX Insights Editors	CX Insights Editors
environment/Access Groups/$customer/GCXI	CX Insights Viewers	CX Insights Viewers

To learn more about Configuration Server Access Groups, talk to your Genesys representative.

Permissions needed to manage other users

To manage the accounts of other users, you must be a member of one of the *Administrator* user groups described in the following table, which describes the types of accounts each of the Administrator types can manage, and the actions they can carry out on each.

The following table describes the permissions needed to manage users.

Table: User management capabilities

Managing Group	Administrator	General Users Administrators	CX Insights Users Administrators	iWD Users Administrators
Managed Group				
Custom	Full Control	Full Control	Full Control	Full Control
MicroStrategy Groups	Full Control	No Access	No Access	No Access
System Administrators	Full Control	No Access	No Access	No Access
Everyone	Full Control	View / Modify / Modify children	View / Modify / Modify children	View / Modify / Modify children
General Developers	Full Control	View / Modify / Modify children	No Access	No Access
General Editors	Full Control	View / Modify / Modify children	No Access	No Access
General Viewers	Full Control	View / Modify / Modify children	No Access	No Access
General Users Administrators	Full Control	View / Modify / Modify children	No Access	No Access
CX Insights Static Access Restriction	Full Control	View / Modify / Modify children	View / Modify children	No Access
CX Insights Dynamic Access Restriction	Full Control	View / Modify / Modify children	View / Modify children	No Access
CX Insights Developers	Full Control	View / Modify / Modify children	View / Modify / Modify children	No Access
CX Insights Editors	Full Control	View / Modify / Modify children	View / Modify / Modify children	No Access
CX Insights Viewers	Full Control	View / Modify / Modify children	View / Modify / Modify children	No Access
CX Insights Users Administrators	Full Control	View / Modify / Modify children	View / Modify / Modify children	No Access
iWD Developers	Full Control	View / Modify / Modify children	No Access	View / Modify / Modify children
iWD Editors	Full Control	View / Modify / Modify children	No Access	View / Modify / Modify children
iWD Viewers	Full Control	View / Modify / Modify children	No Access	View / Modify / Modify children
iWD Users Administrators	Full Control	View / Modify / Modify children	No Access	View / Modify / Modify children

Subtractive access rights

In Genesys CX Insights / Microstrategy, access rights are controlled using a *subtractive* model. User access rights are restricted based on group membership. If a user is a member of more than one group, then the access restrictions of the most restrictive group are enforced. For example, a user who is a member of the groups Tenant Developers and Tenant Editors, has the privileges of Tenant Editors only. For this reason, Genesys recommends that you make each user a member of exactly one Admin Group and / or one Reporting Group.

If you are accustomed to historical reporting in Genesys Interactive Insights (GI2) / Business Objects, note that GCXI uses an entirely different approach to limiting access, and while the term *access restrictions* in GCXI refers to access rights granted to user groups, the same term in GI2 refers to restrictions placed on objects, rows, query types, and connections in the GI2 universe.

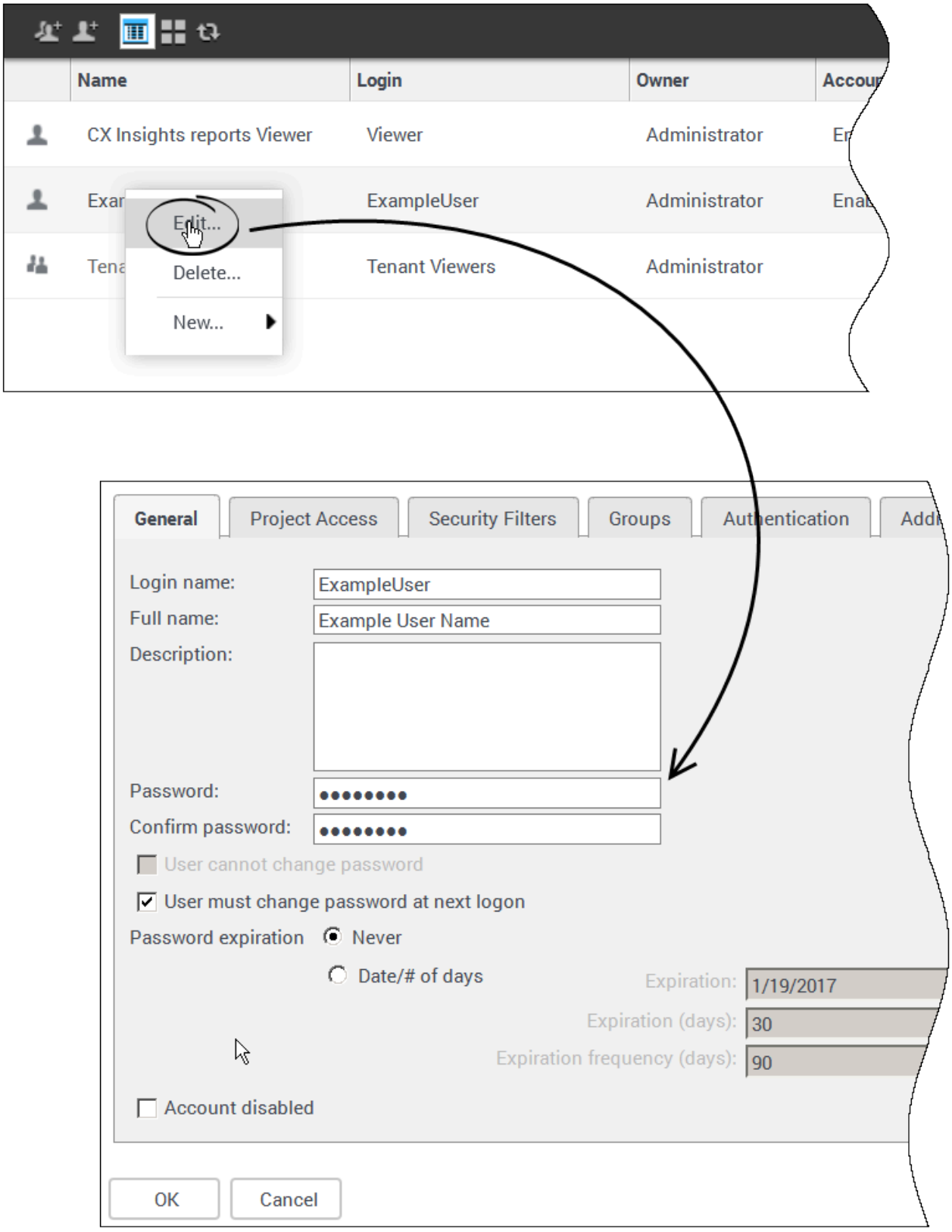
Video: Managing users

Link to video

This video describes how to manage users, including how to:

- create users
- delete users
- change users' passwords
- change users' permissions

Changing another user's password


Name
Login
Owner
CX Insights reports Viewer
Viewer
Administrator
ExampleUser
Administrator
Edit...
Delete...
New...
Tenant Viewers
Administrator
General
Project Access
Security Filters
Groups
Authentication
Login name:
ExampleUser
Full name:
Example User Name
Description:
Password:
Confirm password:
User cannot change password
User must change password at next logon
Password expiration
Never
Date/# of days
Expiration:
1/19/2017
Expiration (days):
30
Expiration frequency (days):
90
Account disabled
OK
Cancel

Use the following steps to change a password for another user (for example when they have forgotten their password) or to otherwise manage an existing user account.

To edit another user's account, you must log in as a member of a group that has the **Create And Edit Users And Groups** privilege.

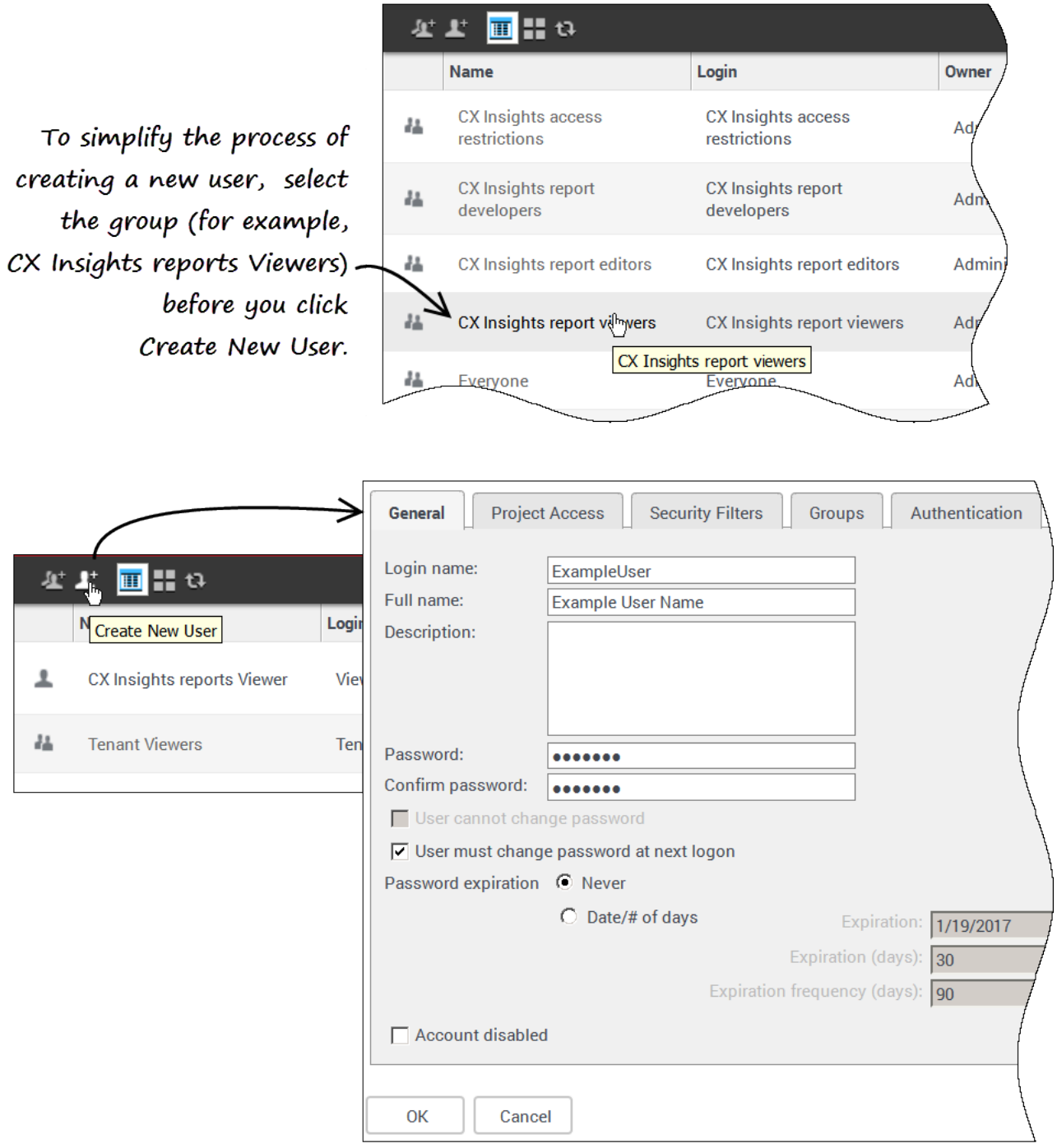
> **Tip**
> For users who are created without membership in any group other than Everyone, only the administrator who created the user can change the user's password. New users must always be members of at least one group, other than Everyone.

1. In your web browser, open the MicroStrategy Web Administrator page:

   `http://<hostname>:<port>/MicroStrategy/servlet/mstrServerAdmin`

2. On the page that appears, select your server.
3. On the MicroStrategy Web Administrator login screen, enter your user name and current password, and click **Login**. The **Tools** page opens.
4. Click **User Manager**.
5. Click a group of which the user is a member. A list appears, showing all the users in that group.
6. Right-click the user's name, and in the menu, click **Edit**.
7. In the **Password** field, enter the new password, and enter it again in the **Confirm Password** field.
8. Select **User must change password at next login**, and make any other changes if required.
9. Click **OK**.

Creating a new user



Use the following steps to create a new user account.

To edit another user's account, you must log in as a member of a group that has the **Create And**

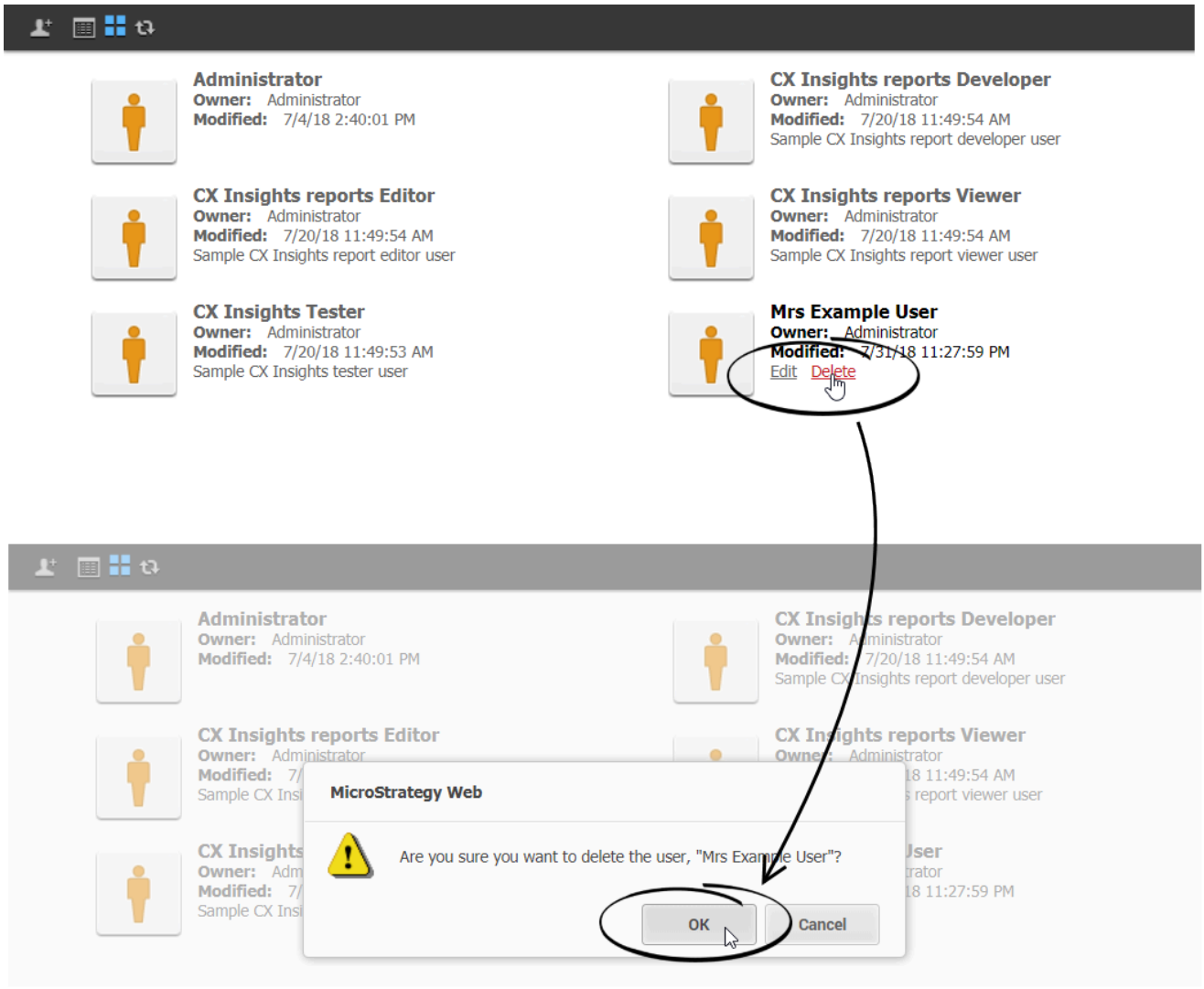
Edit Users And Groups privilege.

1. In your web browser, open the MicroStrategy Web Administrator page:

   ```
   http://<hostname>:<port>/MicroStrategy/servlet/mstrServerAdmin
   ```

2. On the page that appears, select your server.
3. On the MicroStrategy Web Administrator login screen, enter your user name and current password, and click **Login**. The **Tools** page opens.
4. Click **User Manager**.
5. On the menu, click **Create New User**.
6. Specify user information as appropriate, on each tab in the editor. If you need more information about any field, see the MicroStrategy Web Administrator Help. Be sure to:
   1. Include a **Login Name**, **Full Name**, **Password**, **Confirm Password** and other selections in accordance with your password policies (on the **General** tab).
   2. Assign at least one **Group** (on the **Groups** tab). By default, all users are also members of the group **Everyone**, but you must assign at least one group, or the new user account will not be editable by other administrators.
7. Click **OK**.
8. To verify that the user was created, open one of the groups to which you added the user (or open the group **Everyone**).

Deleting a user



Use the following steps to delete a user account.

To edit another user's account, you must log in as a member of a group that has the **Create And Edit Users And Groups** privilege.

1. In your web browser, open the MicroStrategy Web Administrator page:

   `http://<hostname>:<port>/MicroStrategy/servlet/mstrServerAdmin`

2. On the page that appears, select your server.
3. On the MicroStrategy Web Administrator login screen, enter your user name and current password, and click **Login**. The **Tools** page opens.
4. Click **User Manager**.

5. Open a group of which the user is a member, for example **Everyone**.
6. Hover over the user you plan to delete, and click **Delete**.
7. Click **OK**.

> **Important**
>
> Beginning with Genesys CX Insights release 9.0.013, the default user accounts (Developer, Editor, Viewer), are disabled by default. A new container management variable, `GCXI_USERS_ENABLED=false|true`, is added, which you can use to enable the default accounts.

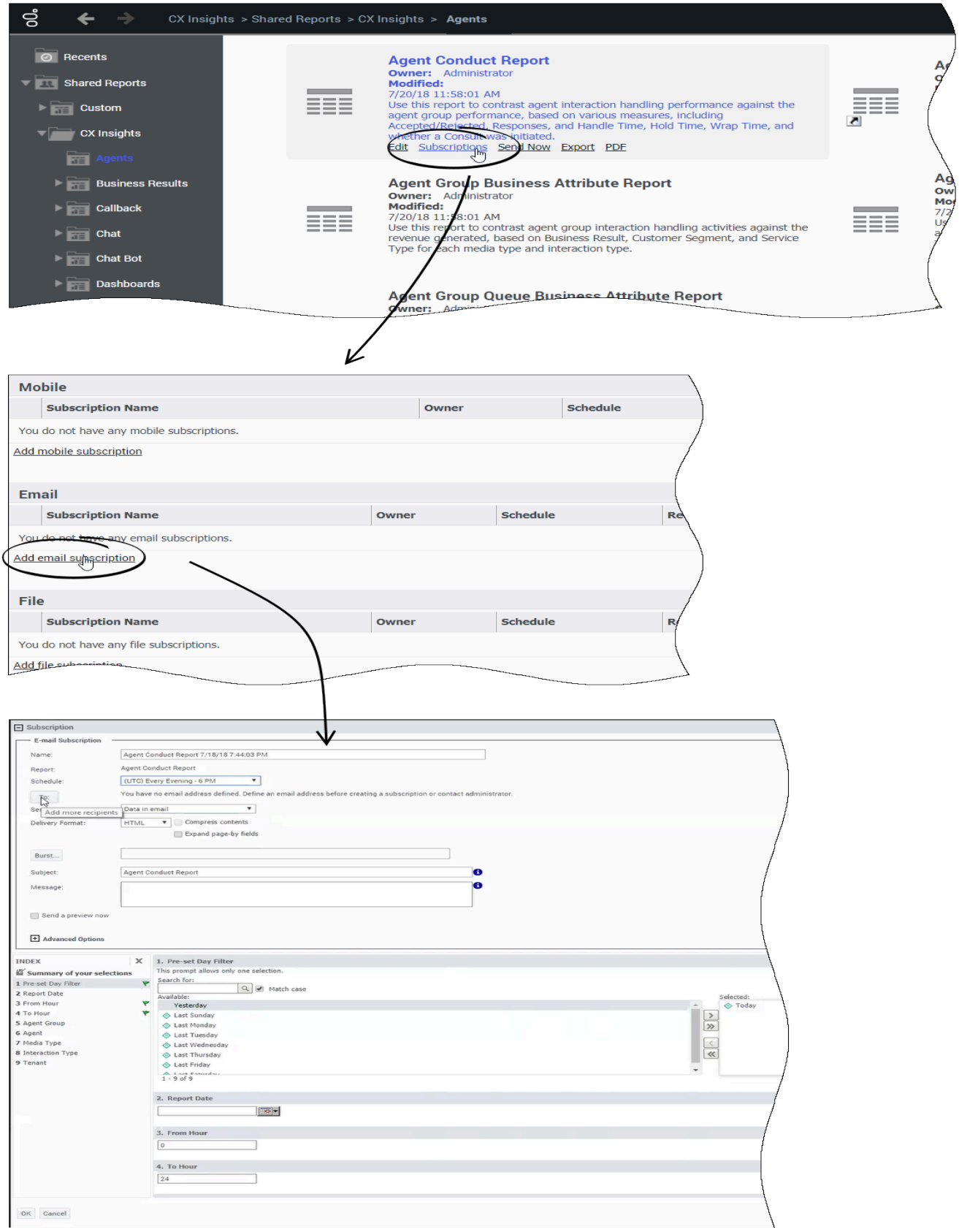
How do I schedule reports to run on a regular basis?

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Report scheduling.

You may find that it is helpful to run the same report regularly--every day, or every week, for example. Using Genesys CX Insights *Subscriptions*, you can easily schedule the system to deliver one or more reports, in the format you prefer, to the destination of your choice, such as your email, printer, or mobile device.

This page describes how to schedule the delivery of a report by email.

Scheduling the delivery of a report


CX Insights > Shared Reports > CX Insights > Agents
Recents
Shared Reports
Custom
CX Insights
Agents
Business Results
Callback
Chat
Chat Bot
Dashboards
Agent Conduct Report
Owner: Administrator
Modified:
7/20/18 11:58:01 AM
Use this report to contrast agent interaction handling performance against the agent group performance, based on various measures, including Accepted/Rejected, Responses, and Handle Time, Hold Time, Wrap Time, and whether a Consult was initiated.
Edit Subscriptions Send Now Export PDF
Agent Group Business Attribute Report
Owner: Administrator
Modified:
7/20/18 11:58:01 AM
Use this report to contrast agent group interaction handling activities against the revenue generated, based on Business Result, Customer Segment, and Service Type for each media type and interaction type.
Agent Group Queue Business Attribute Report
Mobile
Subscription Name
Owner
Schedule
You do not have any mobile subscriptions.
Add mobile subscription
Email
Subscription Name
Owner
Schedule
You do not have any email subscriptions.
Add email subscription
File
Subscription Name
Owner
Schedule
You do not have any file subscriptions.
Add file subscription
Subscription
E-mail Subscription
Name:
Agent Conduct Report 7/18/18 7:44:03 PM
Report:
Agent Conduct Report
Schedule:
(UTC) Every Evening - 6 PM
You have no email address defined. Define an email address before creating a subscription or contact administrator.
Add more recipients
Data in email
Delivery Format:
HTML
Compress contents
Expand page-by fields
Burst...
Subject:
Agent Conduct Report
Message:
Send a preview now
Advanced Options
INDEX
Summary of your selections
1 Pre-set Day Filter
2 Report Date
3 From Hour
4 To Hour
5 Agent Group
6 Agent
7 Media Type
8 Interaction Type
9 Tenant
1. Pre-set Day Filter
This prompt allows only one selection.
Search for:
Match case
Available:
Yesterday
Last Sunday
Last Monday
Last Tuesday
Last Wednesday
Last Thursday
Last Friday
1 - 9 of 9
Selected:
Today
2. Report Date
3. From Hour
0
4. To Hour
24
OK
Cancel

Genesys CX Insights enables you to subscribe to a report so that it runs on a schedule you specify. You can configure Genesys CX Insights to save the report in your history list, so you can view it at your convenience, or you can configure Genesys CX Insights to automatically email the report to you.

1. Open the **Shared Reports** folder, and navigate to the report you want to schedule.
2. Hover over the report name/description, and several options appear below the description. Click **Subscriptions**.
3. On the **My Subscriptions** page, scroll down and select the type of subscription you want to create. Genesys CX Insights supports either of the following selections:
   - **Add History List subscription** to save the report in your history, so you can access it later.
   - **Add email subscription** to automatically send the report to a configured email address. The email subscription details appear.

For example, to configure an email subscription:

1. Click **Add email subscription**.
2. Select a value from the **Schedule** list to control how often the report is generated, and from the **To** list to indicate the email address to which the report is sent.
3. Select values for the various prompts as required.
4. Optionally, select **Send a preview now** to test the subscription (if you do so, an email will arrive in as little as a few minutes).
5. Optionally, add additional users in the **To** list, to send the same report to more than one user.
6. Click **OK**.

> ### Tip
> In some scenarios, email messages that notify users of scheduled reports can contain unexpected characters in the email subject line. If you encounter this issue, see the Known Issue GCXI-2327

Subscriptions can be sent only to enabled users. If a subscription is created for a user who is, or later becomes, disabled, MicroStrategy does not send the report to that user, though it continues to send it to other subscribers.

If you have created subscriptions for a user, and wish to transfer those subscriptions to another user (without having to recreate them from scratch), contact Genesys Customer Care for assistance.

Workforce Management

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

Genesys Workforce Management (WFM) provides a sophisticated set of management tools so you can better manage your contact center workforce while still delivering top service to your customers. It allows you to easily manage your agents, schedules, and forecasts, and provides real-time monitoring of contact center performance and agent-adherence.

Your agents can request time off or specify working hours, bid on proposed schedules, and even trade their schedules with other agents. All while maintaining optimal staffing levels.

WFM makes it easy for you to manage agents of all skill levels, in contact centers that are using single media formats (like voice calls) or a variety of media formats (such as voice calls and emails).

Current documentation set for Workforce Management

Workforce Management Help—Overview and instruction on how to use Workforce Management 8.5.1.

Workforce Management ETL Database Reference—This document describes WFM's Extract, Transform, and Load (ETL) database schema 8.5.1.

> **Important**
> If you are using Workforce Management 8.5.2, refer to the Workforce Management Web for Supervisors Help and the Workforce Management ETL Database Reference.

Workforce Management Product Alerts

- **Google Chrome version 45 ends support for Java Plug-ins**—WFM Web will no longer work with Chrome v45 and later, due to Chrome's end of support for Java Plug-ins. If you are running WFM Web on Chrome v45 or later, Genesys recommends you use either Internet Explorer or Firefox.
- **Limitation for WFM Web in Google Chrome Stable Channel update version 42**—By default, this browser version blocks the Java plug-in with no pop-up or other indication.

Workforce Management supports Chrome, which means, in WFM Web, attempts to invoke Java applet subsystems prompts a message (after a long delay) that Java is not detected.

Resolution:
Starting with Chrome version 42, an additional configuration step is required to continue using NPAPI plug-ins.

To enable NPAPI in Chrome Version 42 and later

1. In your URL bar, enter: `chrome://flags/#enable-npapi`.
2. Click the **Enable** link for the **Enable NPAPI** configuration option.
3. Click the **Relaunch** button that appears at the bottom of the **Configuration** page.

Official source from Oracle: www.java.com

Administrator Tasks

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

This section contains information associated with Workforce Management administration.

Find the information you need by clicking the topic links below or by using the Table of Contents in the left-side navigation bar.

Daily operations

There are many tasks that, as a Workforce Administrator, you will need to perform on a daily basis. These topics provide information about setting up and performing these daily operations in your environment.

Forecasting

Scheduling

Monitoring Performance and Adherence

Reporting

Managing user security

Managing the Calendar

Creating time-off bidding offers

Using task sequencing

Setting up e-mail notifications

Reporting and Metrics

Find information about WFM reports, and metrics.

Scheduling meals & breaks

Enabling wait-lists

Tracking overtime

Schedule Summary View and Reporting

Contact Center Performance Report

Agent Adherence Report

Primers

Find information about how to use WFM Primers to optimize the efficiency of your contact center.

Time Off

Overlays

Multi-Forecasting

Scheduling

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

This topic provides detailed information about Workforce Management (WFM) Scheduling that will help you to plan and create schedules for your workforce.

Scheduling Overview

Workforce Management (WFM) uses the published Master Forecast to create agent schedules that comply with user-defined business constraints. Or you can create “empty” schedules to which you can then assign agents. Schedule constraints include available personnel with required skills, staffing requirements, employment contracts, business policies, and agent preferences.

The staffing requirements act as a target for schedule generation. An optimized schedule ensures the least amount of over- and understaffing while still meeting contractual obligations. WFM uses each agent’s individual skills, contracted working rules, and calendar items as guides to help identify when each agent can work, and what he or she will work on.

WFM aids compliance with regional working rules by helping to apply the following aspects of Contract rules:

- User-defined weekend days
- Schedule synchronization based on specific days of the week
- Maximum number of consecutive weekends an agent may work

You can schedule agents to be available to perform multiple types of work at once or you can schedule them to work on specific types of work for periods of time within their day. You can also combine these, to create schedules in which some periods are set aside for specific types of work while at other times agents perform any work that arrives for which they are qualified.

Once you finalize your schedule, you can publish it to the Master Schedule, where it immediately becomes available for agents to view through WFM Web for Agents. Agents may then trade their schedules as needed, if the schedule trade complies with trading rules and is either auto-approved or is approved by a qualified supervisor.

Maximum Agents by Length of Schedule Period

You can build schedules for up to 5,000 agents and 6 weeks. Memory requirements are decreased, and contiguous memory is not necessary for scheduling. If you have 1.5 GB of virtual memory available, you can now build large schedules (5,000 agents). Generally, for schedules with 2,000 or fewer agents, 600 MB of virtual memory is enough.

> ### Important
> Consider the preceding limits to be rough estimates; scheduling duration varies depending on your configuration. There is no way to provide a general estimate for schedule build time, based on just a few simple parameters such as the number of schedule weeks and the number of agents. The maximum schedule size must be determined uniquely by each user, based on specific performance requirements.

Back to Top

Profile Scheduling

Although agent-based scheduling offers a multitude of advantages, in some cases you must build schedules without agents assigned to them. To do so, you create a schedule composed of empty schedule slots that are appropriate for the contract types or agent skill sets you currently have, or for which you anticipate hiring. WFM offers several methods for creating blank schedules to which you can assign agents:

- **Scheduling Using Profiles**—Profiles are based on contracts and include a skill set. They are used to represent a typical kind of agent or a proposed new agent classification. For example, you can create a new flexible full-time profile to enable planners to evaluate the adoption of a 4-day, 10-hours-per-day work week. Each profile has a skill set with assigned skill levels. Scheduler uses either a user-specified number of each profile type or a blend of profiles based on the current staff, to create blank schedules to which you can assign qualified agents.
- **Mixed Scheduling**—You can build schedules using a combination of profiles and actual agents. This can enable planners to create additional optimized schedules for expected new hires or for outsourced agents to use.
- **Schedule Bidding**—Supervisors create an optimal set of schedules with no agent names, authorize a set of agents to participate in the bidding process, and open the schedules for bidding. Agents review the schedules on which they are qualified to work, and bid by numbering the most desirable and least desirable schedules 1, 2, 3, and so on. The Supervisor can then have WFM assign the schedules to the agents automatically, based on the agents' bids as well as their seniority and/or "rank".

Back to Top

Automated Schedule Bidding

Automated Schedule Bidding allows supervisors to create schedules with no agent names associated, and then distribute them to agents via the Web. The agents can view, filter, and sort these schedules,

and bid on their favorite schedules over a preconfigured period of time. An automated assignment engine then assigns schedules to agents, based on their bids and their seniority and/or rank. When possible, preplanned Calendar items such as granted time off, days off, and exceptions are integrated into agent schedules when the schedules are published to the Master.

This new and powerful feature helps contact centers to comply with union regulations requiring that agents be assigned their desired schedules based on their seniority or rank. It also enhances supervisor productivity by automating the process. Even in non-unionized contact centers, automated schedule bidding improves agent satisfaction by giving agents more control over their future schedules.

Back to Top

Skills-Based Scheduling

You can build schedules based on primary and secondary skills. WFM defines activity/skill levels as follows:

- Primary activity—Is assigned to the agent directly and *not marked* as secondary. If activity assignment is based on skills and skill levels, the primary activity is one, for which none of the agent's skills are marked as secondary.
- Secondary activity—Is assigned to the agent directly and *marked* as secondary. If activity assignment is based on skills and skill levels, the secondary activity is one, for which at least one of agent's skills is marked as secondary.

> **Important**
>
> WFM Builder API 8.5.1 or later must be installed to enable full implementation of this feature. If you installed an older WFM Builder API or you have disabled the primary/secondary activities feature, scheduling behavior is the same as in earlier versions.

Algorithm for Skills-Based Scheduling

Agents are typically assigned a list of activities to work on for each day of the schedule scenario. When primary/secondary skills-based scheduling is enabled, WFM assigns two lists of activities: primary activities and secondary activities, and uses the following algorithm for daily distribution:

1. Minimum required primary activity
2. Minimum required all activities
3. Regular required primary activity
4. Regular required all activities

Here are some other things to consider when using skills-based scheduling:

- For timesteps, when no activities are selected during daily distribution or if activities are removed during swapping, the primary activities have a higher weighting during selection.

- While tasks are being optimized, the primary activities have a higher weighting during activity selection.
- Task sequences, activity sets, and scheduled shift items might negatively impact primary/secondary activity selection. Be sure to check every timestep individually to determine if any agents are working on a secondary activity, instead of a primary activity.

Multi-Site Planning

Using a familiar tree structure, you can configure WFM's objects to correspond exactly to your Enterprise organization. For centralized, multi-site contact centers, WFM enables forecasting and building of schedules for work activities spanning all sites.

For decentralized, multi-site contact centers, WFM supports two main models.

- For multi-site contact centers that distribute calls based on percentage allocation, WFM enables you to forecast interaction volumes centrally and distribute the workload to each site for further planning efforts. Each site can set parameters such as service objectives and staffing requirements, and can build schedules.
- For multi-site contact centers that are virtualized and distribute calls based on agent availability, skill set, and so on, WFM enables you to forecast staffing centrally and then split the staffing requirements to each site. Schedules may then be built for each site. By building staffing requirements centrally, WFM can account for the efficiencies of scale that are seen in a true virtual contact center environment.

The browser-based capabilities provided by WFM ensure that in any multi-site environment users across the enterprise can participate in the planning process.

Also see Multi Forecasting Primer.

Back to Top

Agent Preferences

The Scheduler can optionally consider agent preferences when building schedules. Agents can enter preferences for shifts, days off, availability, and time off using WFM Web for Agents. Supervisors can enter agent preferences in WFM Web for Supervisors and, with the appropriate security permissions, can grant or reject preferences. If a supervisor grants a preference, the calendar algorithm considers that agent's preference when building the schedule, along with various other criteria such as seniority.

Preference Fulfillment and Schedule Optimization

Contact center administrators can also specify whether preference fulfillment or schedule optimization is the more important goal. This adds another layer of control over preference scheduling.

Flexible Shifts

The method WFM uses to create shifts enables you to configure flexible shift durations and start and end times. Additionally, WFM schedules use flexible break and meal parameters.

In a sense a WFM shift is an abstraction, representing countless possible working times. This is true even if the shift is configured to produce very regular, fixed, agent schedules. This is in contrast to the conventional notion of a shift with a mandatory fixed weekly start time, fixed duration, and set breaks.

A single WFM shift can incorporate hundreds of possible start times and durations as long as they fall within the parameters of the associated contract. However, through synchronicity constraints and use of more-rigid shift configuration settings, you can fix agent start times and workday durations. This combination of flexibility and structure makes the WFM shift a tremendously powerful scheduling mechanism. In fact, in some cases, you can configure an entire contact center using only a few WFM shifts.

Back to Top

Example of a WFM Shift:

Consider a contact center with a standard full-time shift of 8 hours a day, 5 days a week, and an alternative full-time shift of 10 hours a day, 4 days a week. You can schedule both types of agents using a single shift with a flexible duration of 8 to 10 hours per day. In either case, the agents are contracted to receive 40 hours work each week and to work 4 or 5 days. You can configure WFM to guarantee that specific agents work 4 days a week and others 5 days, or let the WFM Scheduler determine how many agents of each full-time type to use to provide the most effective schedule.

Task Sequences

WFM task-based scheduling enables you to configure sequences of work activities to be used in shifts. These task sequences guarantee that a specific period of time is spent on a specified activity or set of work activities.

Using task sequences, multimedia contact centers can generate agent-friendly schedules that build in extended periods of time set aside for handling specific tasks. Agents are thus able to focus on a single media or skill set, enabling them to complete their tasks more effectively, without the confusing effects of frequently switching media. Contact center planners can ensure that task time is equitably distributed among all qualified agents. And WFM can optimize the assignment of task times based on forecast staffing requirements.

For example, you can guarantee that all appropriately skilled agents receive exactly 2 hours of outbound work for every shift, or you can allow WFM to determine how much outbound work to distribute to each agent. You can configure Genesys Routing to use WFM schedule information as input for routing decisions. In this way, you can use task-based scheduling to provide a closed-loop routing system that complements an agent-based approach to contact center management.

For more information about task sequences, see Configuring WFM Task Sequences.

Back to Top

Schedule Trading

WFM schedule trading enables agents to trade schedules among themselves. They can do so either through a trade with a specified agent or through a trade open to any qualified agents within their community.

Contact center planners no longer need to spend an excessive amount of time managing and processing agent schedule-trade requests. Agents feel that they have flexibility when they need to change their usual schedule and that they have more proactive control over the times they work. In some cases, schedule trades can be approved without supervisor intervention, enabling managers to focus on trades that may affect service levels or violate company policies.

Marked Time

Use marked time to distinguish any periods of time that are not otherwise tracked and reported on in an existing WFM category. For example, you can create a marked-time type for a particular project. Or you can use marked time to identify overtime periods that you want to appear in a report.

You configure marked-time types using WFM Web. You can specify marked time in WFM Web for Supervisors and view periods of marked time in its Intra-Day schedule view. You can report on marked time using the Schedule Marked Time Report and the Schedule Marked Time Totals Report.

Intra-Day Scheduling

The WFM Web for Supervisors Intra-Day schedule views enable you to make real-time adjustments to schedule scenarios or to the Master Schedule. You can insert exceptions, edit or change shift start and end times, assign meetings, enter time off for an agent who has suddenly gone home ill, change the activities agents are working on, or make other changes to the schedule to improve contact center performance and to make the schedule reflect actual contact center circumstances.

You can make changes one at a time or use one of the Schedule wizards to make changes to multiple agents' schedules at once.

Back to Top

Intra-Day Schedule Re-Optimization

When you build a schedule in WFM, the scheduling algorithm minimizes the over and under-staffing of agents against the forecasted staffing requirements, while meeting the configured working rules. Since schedules can be built several weeks in advance, a variety of circumstances can cause the schedule to become suboptimal by the time a particular schedule day arrives.

Here are some examples:

- Contact center management re-forecast volumes and staffing for the day.
- Agents called in sick or were granted time off.

- Existing agent schedules were manually adjusted.
- Additional agents were added into the schedule.
- Meetings or other types of exceptions were added to the schedule.

It does not make sense to re-optimize schedule items for days or hours that have already passed. For most contact centers, it is also not practical to re-optimize the current hour. Any changes to meals, breaks, and/or work activities might be difficult to communicate to the affected agents. For these reasons, a re-optimization wizard allows you to select the date, start time, affected agents, and the set of schedule items to be re-optimized.

For example, you have the option to re-optimize the placement of:

1. Breaks only
2. Meals and breaks only
3. Activity sets/task sequences/activities only
4. Breaks/meals/activity sets/task sequences/activities without affecting shift start/end times
5. Breaks/meals/activity sets/task sequences/activities, and shift start/end times

Re-optimization provides some flexibility if you do not wish to change certain shift items or work activities because it might be difficult for your agents to adjust to those changes. For example, if agents use their meal breaks to go out of the office and go to appointments, you might not want to change these times once they have been published. Similarly, you can decide whether shift durations should be allowed to change or not. In some contact centers, this can be done to offer additional work hours to certain agents. In other contact centers, this is not a desired practice.

Additionally, you can choose to exclude from re-optimization any agents whose schedules have already been manually edited. You might have already spent time manually adjusting shift items or work activities for an agent (for example, you moved meals or breaks based on a particular request from an agent) and you don’t want to lose those changes.

Back to Top

Pending Schedule Changes

Users who do not have the Approve Changes security permission enabled can make only pending changes to the Master Schedule. Pending changes do not affect the official version of the Master Schedule.

A user with the Approve Changes security permission enabled must commit pending changes before they are incorporated into the official schedule. Discarded changes are *rolled* back.

Alternatively, such a user can go to the Master Schedule Changes Approval module (invisible to users without the Approve Changes security permission enabled). There, she or he can review the pending changes to the Master Schedule made by any user, and approve or reject them.

You can also enter pending changes into a schedule scenario. Such pending changes are visible only to the user who entered them. You can later review your pending changes, and either commit them or roll them back. Once committed, the changes are visible to all users with access to the schedule

scenario.

> **Important**
> If a scenario with pending changes is published to the Master Schedule, the pending changes are not included.

Schedule State Group Totals View

This view provides intra-day totals of the number of agents in each schedule state group (Meetings, Lunch, Breaks, and more).

It helps managers and supervisors understand how many agents are scheduled for each type of activity during a particular time period, and provides a snapshot view of productive vs. nonproductive time on an intra-day basis.

Back to Top

Forecasting

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

Use the Workforce Management (WFM) Forecasting tool to predict contact-center workload and staffing requirements, based on historical data or user-defined templates. WFM provides multiple methods of forecasting the workload and staffing requirements for work activities.

Start by creating one or multiple forecast scenarios. Creating multiple scenarios enables you to see the effects of changes to forecasting parameters, such as service objectives and predicted interaction volume. When you have determined the most satisfactory forecast, publish it, making it the Master Forecast on which schedule scenarios, and eventually the Master Schedule, are based.

If you choose to, you can derive workload forecasts from historical information that is either collected automatically by WFM from the Genesys system or imported from `.csv` files using the WFM Web. You can also create workload and staffing forecasts as reusable templates. Once you have generated a workload prediction, WFM determines the staffing requirements needed to service the workload, taking into account any applicable service objectives.

Using Historical Data

WFM automatically collects historical data from Stat Server for all work activities handled by the Genesys platform encompassing all media, contact segments, and service types. Using Genesys Stat Server, rather than automatic call distribution (ACD) reports, provides you with far greater flexibility in defining and gathering statistics that provide an appropriate measure of contact center performance over time.

WFM analyzes interaction volumes and average handling time (AHT) in order to predict future trends for each work activity. This data enables WFM to build accurate forecasts for the anticipated workload, and to calculate the staffing required to meet that workload.

Using the WFM API, you can also develop a custom application that will enable Interaction Volume and AHT data from a third-party system to be imported directly into the WFM database. This is useful if you want to use WFM to forecast and schedule a type of work that is not being routed by Genesys.

Back to Top

Using Forecasting Events

WFM can track events that might affect interaction volume. A sales promotion or marketing campaign, for example, might cause a predictable peak in interaction volume. These types of events are entered in WFM Web for Supervisors and used by the advanced WFM algorithms. If an event recurs, the forecasting algorithms learn the impact of that event and account for its impact in future forecasts.

Setting Service Objectives

With WFM forecasting, you can set specific service objectives. You can also adjust these objectives and then rebuild the forecast, which provides a detailed "what-if" analysis of the potential impact of staffing or service-objective changes. WFM forecasting uses parameters to determine effects of different service objective settings, such as:

- Interaction volumes
- Average handle time (AHT)
- Average speed of answer (ASA)
- Desired percentage of interactions handled within a target time (service level)
- Occupancy
- Maximum percentage of abandoned interactions

Back to Top

Flexible Forecasting

WFM supports an unlimited number of forecasting scenarios, enabling you to create multiple forecasts and evaluate how changes in the parameters or the forecasting method that you use, affect expected service objectives. Resource planners can then easily create reliable forecasts, fine-tuning the results in tabular and graphical data views. You can also save forecast workforce data as templates for use in subsequent forecast building.

WFM offers several different forecasting methods of varying complexity:

- **Template-Based**—Good for work activities with little historical information or for activities with very predictable interaction traffic.
- **Expert Average Engine**—Good for work activities that have a reasonable amount of historical data or those that fluctuate more dramatically because of unknown factors.
- **Universal Modeling Engine**—Good for work activities with more than one year of historical data and accurate forecasting event information.
- **Copy Historical Data**—Good for work activities when you have some historical data, but not enough to use the Expert Average Engine or the Universal Modeling Engine. You can combine the historical data with overlap templates, which fill in gaps in the historical data.

- **Use Value**—Good for work activities if your site activity load is very regular. Applies a specific interaction volume or AHT to each time interval in the scenario.

Deferred-Work Forecasting

WFM is designed to consider work activities that can be deferred, such as e-mail, as inherently different from *immediate* work, such as a phone call. WFM uses a proprietary algorithm designed to distribute the backlog of interactions that can be deferred across the day in order to satisfy your service goal, which is expressed in minutes, hours, or days.

Spreading out the deferred work enables you to avoid spikes in workload forecasts when a contact center opens for the day, or during brief periods of high volume.

Back to Top

Multi-Skill Support

A multi-skilled contact center presents an opportunity for increased productivity.

An agent might be idle in a single-skill environment, because she cannot answer calls that are queuing for a particular activity/skill that she possesses—because the schedule prevents her from using that skill.

In a multi-skilled environment, the agent can use her additional skills to answer calls. A multi-skilled agent is qualified to work on multiple activities, and therefore can perform different types of work during a shift.

In a multi-skill environment, an agent can be available for multiple activities during any timestep. The agent can be scheduled to work on an activity for only part of a timestep, and only the fraction of the time period during which she works is counted.

Because of this, the value for staffing can be expressed as a fraction. For details, see Multi Forecasting Primer.

Back to Top

Monitoring Performance and Adherence

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

Workforce Management (WFM) provides the tools, described in this topic, to help you can monitor the workforce performance and adherence in your environment. You can generate Performance and Adherence data, enabling you to analyze it and spot trends that might be developing over time.

Performance

The Performance modules compare the forecast and schedule to what is actually happening in the contact center. WFM shows intra-day statistics, such as interaction volume, average handling time (AHT), agents logged in, service level, average speed of answer (ASA), and abandons, and compares them to the planned values.

Intra-day contact-center performance data is displayed in an informative and easy-to-read format, enabling efficient performance monitoring and quick response to unanticipated interaction flow or agent-staffing situations.

WFM also provides a “what-if” calculator as an aid to decision making. You can enter new values for staffing, interaction volume, and/or other performance statistics into the `What-If` window. The what-if calculator then supplies the results to be expected if the values change as you project.

See also, Contact Center Performance Report Metrics.

Adherence

Workforce Management provides real-time agent-adherence data, which compares the current agent status to the scheduled status. WFM can track agent adherence per time interval on a single channel or across multiple channels simultaneously.

Agents who are not adhering to their schedules (within user-defined thresholds) are highlighted in yellow if they are nonadherent or in red if they are severely nonadherent. WFM also displays the amount of time, in minutes, that the agent’s current status has differed from the scheduled status. This running total is continually updated.

Multi-Channel Adherence

To enable tracking of multi-channel adherence in WFM Web, assign a media channel to a Schedule State Group; WFM then calculates adherence by comparing real-time states to the scheduled states for that channel. See the following example:

Use Case 1: Multi-Channel Adherence
A site in the contact center has voice and e-mail related activities, and agents can make outbound calls, which are scheduled as exceptions. The schedule state groups can be configured as follows:

Schedule state group	Channel	Schedule state	Real-time state
Inbound calls	Voice/unspecified	All (immediate) voice-related activities	`WaitForNextCall`, `CallRinging`, `AfterCallWork`, `CallInbound`
Outbound calls	Voice/unspecified	Exception types that represent outbound call work	`WaitForNextCall`, `CallRinging`, `AfterCallWork`, `CallOutbound`
E-mail	`<channel_name>` (for example, `email`)	Deferred, e-mail related activities	`WaitForNextCall`, `CallRinging`, `AfterCallWork`, `CallInbound`
Overhead	None (no channel)	Breaks, meals, time off, exceptions, etc.	`NotReadyForTheNextCall`, `LoggedOut`

If WFM is tracking multi-channel adherence (new in WFM 8.5), and there are no Schedule State Groups with a channel name under the Site, WFM calculates adherence as it would for a single agent real-time state, and Stat Server aggregates the statistics as it did prior to 8.5. However, if there is at least one Schedule State Group with a configured channel name, WFM tracks multiple channels. For each Site, WFM tracks as many different channels as there are distinct channel names for all Schedule State Groups under the Site.

For information about how to configure multi-channel adherence, see Setting Up Media Channels for Schedule State Groups.

Important

After updating to 8.5, if any site in your environment supports multiple media channels and you have configured Schedule State Groups to a setting other than None, the Agent Adherence Report includes data that differs from the data in WFM 8.1.3 in two ways:

1. For any given set of criteria (site/timezone/agent/date), the data in the 8.5 report is different than the data in the 8.1.3 report, because the adherence rules change when you add multi-channel settings to Schedule State Groups, which means the adherence

percentages also changes.

2. The Schedule State, Agent State, Start Time, End Time columns are in reverse order. In 8.5, the report displays the columns in this order: Start Time, End Time, Schedule State, Agent State.

How WFM Calculates Adherence

Workforce Management (WFM) calculates adherence for single channel or multi-channel environments. For sites, in which there are schedule state groups configured for a single channel, WFM compares the agent's real-time state to the current schedule state. For sites, in which there are schedule state groups configured for multiple channels, WFM compares the agent's real-time state and reason code on each channel, with the scheduled states for that same channel during each specified time interval. If at least one scheduled state for that channel can be mapped to the channel real-time state, according to its adherence rules, the agent is considered adherent.

Single-Channel Adherence Calculation

Single channel agent adherence is calculated as follows:

1. WFM maps the agent's real-time state plus the reason code. If there is more than one reason code, there is more than one state + reason code mapping. If there is no reason code, WFM uses only the state for mapping. For example, if the agent has real-time state `WaitingForNextCall` with reason codes `r1` and `r2`, for adherence purposes, WFM maps `WaitingForNextCall + r1` and `WaitingForNextCall + r2`.
2. WFM then finds all schedule state groups that are adherent to at least one agent real-time state from step 1. A list of schedule state groups is compiled that maps to the state, based on the configuration of the schedule state groups.
3. WFM obtains all scheduled states from the current agent schedule and maps them to the schedule state groups.
4. WFM collects all schedule state groups from step 3.
5. WFM Intersects the sets of schedule state groups from steps 2 and 4. If the intersection is not empty, the agent is adherent.

Multi-Channel Adherence Calculation

Multi-channel agent adherence is calculated as follows:

1. Similar to step 1 in Single-Channel Adherence Calculation, WFM maps the agent real-time state + reason code. However, in addition to the aggregated agent state, WFM also adds separate real-time states for each channel configured on the site. (Agents can sometimes have no state on certain channels.) If reason codes are used, WFM could map multiple state + reason code pairs for each

channel, plus the aggregated state.

2. Similar to step 2 in Single-Channel Adherence Calculation, WFM maps schedule state groups adherent to the aggregated state. However, in addition, WFM finds a separate set of schedule state groups for each channel. WFM considers only the schedule state groups that are specifically assigned to a particular channel for adherence with the states on that channel. WFM considers the schedule state groups without a channel for adherence with the aggregated agent state.
3. WFM obtains all scheduled states from the current agent schedule and maps them to the schedule state groups.
4. WFM collects all schedule state groups from step 3.
5. WFM Intersects the sets of schedule state groups from steps 2 and 4 separately for each channel. If both sets are empty or the intersection is not empty, WFM considers the agent is adherent to the channel. For the aggregated agent status, WFM assumes adherence, when either the pair in step 4 is empty or step 2 and step 4 intersect. WFM considers the agent adherent, if he/she is adherent on all channels and adherent to the aggregated status.

The multi-channel algorithm also comes to a boolean conclusion; that is, the agent is either adherent or non-adherent. However, to be adherent the agent must be adherent on every channel, on which he/she is scheduled or, for which he/she receives a real-time state. Also, if the agent is scheduled on non-channel-related states, he/she must also be adherent to those states. See the following example:

Use Case 2: Multi-Channel Adherence
This use case is based on the schedule state group configuration in Use Case 1 Multi-Channel Adherence.

Summary	Real-time state	Scheduled states	Adherence
Agent is working on voice only, but scheduled for e-mail and voice	Agent state: `CallInbound` DN e-mail: `NotReady` DN 2323: `CallInbound`	E-mail activity, voice activity	Not adherent
Agent is working on e-mail and voice, but scheduled for e-mail only.	Agent state: `CallInbound` DN e-mail: `WaitForNextCall` DN 2323: `CallInbound`	E-mail activity	Not adherent
Agent is on a break.	Agent state: `NotReady` DN e-mail: `NotReady` DN 2323: `NotReady`	Break	Adherent

Calculation of Agent Headcount

WFM calculates the agent head count for activities in this way: If the activity belongs to a channel-related schedule state group, the agent is counted for the activity, only if the he/she is compliant with the adherence rules for that channel. Also, if an agent is non adherent overall, but adherent for a channel, the agent is added to the headcount for the activities for that channel.

Using Reason Codes

WFM enables you to enter reason (aux) codes when you configure agent-adherence rules. The reason codes are linked to Genesys `Agent States` and add additional details to the state information. The

Genesys state + reason code combination is mapped to WFM Scheduled State Groups and is displayed in Adherence views.

> **Tip**
>
> When you filter on user-defined reason codes in the Adherence Filter dialog, the reason code that you specify must not contain any spaces within or at the end of the key value.

For example, an agent might signal that she is in a NotReady state. By adding a reason code, she can specify that she is doing after-call work or answering email. This detailed information then appears in the WFM Web Adherence Details view and agent-adherence reports.

To use reason codes, your switch must support them. See your T-Server documentation to find out whether your switch can include reason codes when it sends Genesys TEvents.

See also, Agent Adherence Report Metrics and End Notes

Reporting

WFM Web for Supervisors provides access to a variety of reports that are designed to present key contact-center data in a flexible and accessible format. The report types are:

- Configuration Reports—Contain information about work activity configuration.
- Policies Reports—Contain information about agents, contracts, shifts, and rotating patterns.
- Calendar Reports—Contain information about time off and agent calendar items.
- Forecast Reports—Display forecast interaction volumes, AHT, and staffing requirements in tabular and graph formats.
- Schedule Reports—Display schedule data for agents, activities, teams, sites, multi-site activities, and business units at various granularities. Also present budget information and schedule validation warnings and errors.
- Performance Reports—Contain various types of contact-center performance statistics in detailed and summary formats.
- Adherence Reports—Contain agent-adherence information for agents, teams, sites, business units, and the enterprise.
- Audit Reports—Contain information that enables you to audit a history of changes made within the Calendar subsystem and a history of changes made to the Master Schedule.

For more information about Workforce Management report data and other metrics, see WFM Metrics.

User Security

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

The Workforce Management (WFM) user security modules **Configuration** > **Roles** and **Configuration** > **Users** in WFM Web for Supervisors, enable you to fine-tune the precise access each user has to WFM modules, objects, and functions. For example, you can:

- Limit certain users so that they can view only certain sites or teams.
- Limit certain users so that they can read the schedule but not change it.
- Limit access to reports.
- Limit access to WFM configuration settings modules, such as **Contracts** and **Time Off Rules**.

Pending Schedule Changes

User security enables you to control who can make changes to schedule scenarios and to the Master Schedule. Users might be able to enter changes to the Master Schedule, but unable to commit or approve changes. Such changes are in **pending** status. An authorized user can then review the changes, and either commit/approve them or roll back/delete the changes.

This enables contact center managers to provide Master Schedule access to certain users who might not ordinarily have access. For example, supervisors who manage teams of agents, but who don’t normally have any scheduling responsibility, can enter team meetings or other exceptions into the schedule. Workforce-scheduling professionals can then review these to ensure that coverage is not adversely affected.

Calendar Management

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

Workforce Management's (WFM) unique agent-based scheduling approach enables robust agent-calendar management prior to scheduling. WFM can incorporate known obligations into agent schedules to ensure that agents can keep appointments and request adjusted shifts or working hours while WFM maximizes contact-center efficiency. By more accurately planning for known obligations, WFM can take the guesswork out of forecasting for staffing overheads, leading to more efficient use of the agent pool.

Planning and Scheduling Meetings

The Meeting Planner provides great flexibility when planning meetings. You select the meeting participants, define the range of time in which the meeting should occur, and set the duration of the meeting. You can configure recurring meetings, specifying either the number of occurrences or the start and end dates of the meeting series and the interval (weekly, for example).

The Meeting Scheduler builds the meeting into the work schedules of the participants during the scheduling process, finding the optimal times for agents' shifts and the meeting at the same time.

The meeting is included as an exception in all attendees' schedules. WFM displays the meeting exception in the Schedule views using the meeting short name, so you can find it easily when looking at schedules. If a sufficient percentage of participating agents is unavailable, then the meeting is not scheduled, and you receive a warning.

Additional Functionality

Use the Meeting Scheduler...

- To insert meetings directly into multiple agent schedules as an exception after building the schedule.
- To create optimally-scheduled meetings within an existing schedule.

Use the Meeting Planner...

- To configure meetings that are pre-planned, such as recurring team meetings.

Play ‘what if’...

- To add a meeting to a schedule that has already been built, and WFM will insert the meeting into the most optimal time slot, based on the list of participants.

Supervisors can use this feature to better determine the optimal meeting times that otherwise required manual calculation or guesswork.

Time Off

You can use the WFM Web's **Time Off Types** module to configure multiple types of time off. Doing so enables you to set different characteristics for each type, define different rules for the accumulation of accrued time off and distribution of awarded time off, and fine-tune your record-keeping using the Time Off Report.

Time off can be accrued or awarded. The settings for these differ slightly, but both are configured in the WFM Web’s **Time-Off Rules** module. You can configure separate time-off rules for each time-off type, or assign several time-off types to the same time-off rule, meaning all of the time-off types assigned to this rule share the time-off balance.

You can also associate multiple time-off rules with one time-off type. For example, you might want to have different accrual rules for agents with more seniority than for those who were recently hired. By assigning the appropriate time-off rule to each agent who receives time off of that type, you can determine the rate at which each agent accrues the time off.

Use the **Calendar** module in WFM Web for Supervisors to set time-off limits. You can set time-off limits for an activity, for a team, or for an entire site. You can also set different time-off limits for a specified period. For example, you might want to further limit time off because of special circumstances.

You can also set different time-off limits for various periods during the day. For example, you might permit more time off in the evening than in the busier morning.

Agents can view their balances for each time-off type and request time off in WFM Web for Agents. Agents can request both full-day and part-day time off. Supervisors can enter time-off requests into the WFM Web for Supervisors **Calendar** module.

Requested time off can be manually approved by the supervisor or automatically approved by WFM Web, based on agent time-off balances and the limits set on the number of agents with time off per activity, team, or site.

When the Scheduler runs, all time off that has been granted is scheduled. Additional time off can be scheduled, depending on whether it meets time-off limits and scheduling optimization constraints. Once time off is scheduled, agents can no longer edit or remove the time-off assignment using WFM Web for Agents.

Time Off Wait-List

When a time-off request is made, but time off limits have already been reached, if the agent asks for the request to be wait-listed, the request remains in the WFM Calendar in a **Preferred** status, rather

than being declined. Supervisors can view this *wait list* in the Calendar, sorting the time off requests by timestamp, and selectively grant time off requests.

Supervisors can grant agent time off for future periods, if the time off limits are raised, or if other agents cancel their existing requests. This improves supervisor productivity by no longer requiring them to track these requests with a paper-based system.

Your supervisor can enable automatic approval (also known as *auto-granting*), which eliminates the need for that supervisor to approve your legitimate time-off requests.

Time Off Within Bidding Periods

You can enable Time-Off Bidding by configuring bidding periods in WFM Web and associating them with sites within your enterprise. Agents within the site can submit multiple time off requests concurrently and WFM processes them within the specified bidding period. This configuration ensures that agent requests are all granted, declined, or wait-listed on the same processing date and eliminates the possibility of some requests for a vacation period being granted, while others are not. For details, see Time-Off Bidding.

Schedule Exceptions

Exceptions are additions to a schedule that are not work, but which must be taken into account to allocate agent time correctly. Examples of exceptions include meetings, training, and special projects.

Exception Types

You create exception types based on the needs of your contact center. These types can be extremely flexible and you can link them to other WFM scheduling features. For example, you can specify that some exception types are used in meeting planning, and that some can be converted to a day off, if necessary.

Exceptions can be full-day or part-day. You can assign multiple part-day exceptions, assuming they do not overlap or otherwise violate internal WFM consistency checks.

Payback Exceptions

Payback exceptions are unpaid part-day work intervals that are inserted into schedules to make up for missed time due to lateness or personal appointments. Supervisors and agents can insert these exceptions and specify payback options to recoup missed time.

Payback exceptions might not be applicable or required in all sites, business units, or contact centers. To enable this functionality, contact your Genesys representative.

For information about how supervisors and agents can use payback exceptions, see Insert Exceptions with Payback in the *Workforce Management Web for Supervisors Help* and Inserting Payback Exceptions in the *Workforce Management Agent Help*.

Exception Memos

You can add Exception memos in Schedules, Calendar items, and Meetings to provide comments or additional information about Exceptions types. Exception memos can contain up to 256 characters.

Memos in Calendar Items

When you create Exception memos in Calendar items, they are visible in the Schedule views after they are scheduled, and preserved if transferred between Schedule scenarios, from Scenario to Master Schedule, or from Master to Schedule Scenario. If you edit the memo in the Calendar, the schedule reflects the change without having to be rebuilt or republished.

If this same memo is in the schedule (rather than in Calendar items), it is not saved in the Calendar. Instead, WFM creates a new memo that is related to this scheduled Exception. This newly-created memo is not related to the Calendar item. Therefore, changing the original memo in the Calendar item does not affect this instance of the scheduled Exception—it is a separate memo.

> **Tip**
> You can create an Exception memo in the schedule, even if you have created an Exception without a memo in Calendar items. Again, the scheduled Exception memo is not synchronized with the Calendar item Exception and will not reflect any modifications made to Exception memo in the schedule. See Memos in Schedule Views.

If you delete an Exception and its memo in the Calendar, it will no longer be visible in the schedule. However, you can enter a new memo for the Exception that is not related to the Calendar item. If there are multiple instances of an Exception in multiple scenarios and/or the Master Schedule and you delete the related Calendar item, all instances become separate Exceptions.

When you change a memo in a scheduled Exception the change is reflected in the schedule audit and history. If the settings for the Exception type allow, scheduled Exception memos are swapped and traded with Exceptions. If a calendar-related memo is swapped or traded, WFM copies it to the schedule and disassociated it from the calendar.

Exception memos are displayed:

- In the **Details** bar at the bottom of the **Schedule Intra-Day** and **Agent-Extended** views.
- Next to the scheduled **Exception Type Name** in the **Individual Schedule** and **Team Schedule** reports.
- In the schedule details (and printed schedule details) for the Web for Agents > **My Schedule** view, in a manner similar to the reports.

Memos in Schedule Views

While creating new Exceptions in the Schedule Scenario or Master Schedule that are not associated with a Calendar item, you can also add and edit an Exception memo. You can edit Exception memos

that have been inserted directly into the schedule and initially created without memos or those that became schedule-only Exceptions after the related Calendar item was deleted. (See Memos in Calendar Items.) However, each Exception is edited separately, even if they were inserted using the **Insert Multiple** wizard.

> **Important**
>
> Due to the size limitations for an entire scheduling day, the memo might not fit into the schedule if there are multiple exceptions with long memos in the same agent day.

Exception memos are published or extracted to and from the Master Schedule along with Exceptions, and are transferred when you create scenarios, based on data from another scenario or data taken from the Master Schedule. They are also committed (together with Exceptions) from the agent's schedule to the the Master Schedule. However, unlike Exception memos created in Calendar items, after publishing or creating schedules, the link between Exceptions is not maintained and editing one instance of a memo does not change other instances.

After the schedules are published, the memos appear in the **Details** bar at the bottom of the Schedule > Intra-Day and Schedule > Agent Extended views. You might not see the entire memo, depending on the length of the memo or number of characters. However, when editing the memo, the entire memo text is displayed.

Memos in Meeting Exceptions

Entering memos in Meeting Exceptions is not quite the same as adding Exception memos in the Calendar or Schedule views. In the meeting **Properties** settings, the **Meeting Name** field can include a short memo, further describing the type or purpose of the meeting. For example, `exception_name, meeting_name (exception_memo)`. Together with the **Meeting Name** and **Exception Type** this memo is displayed in published schedules, just as exception memos are for Calendar items and Schedule views.

Exceptions and Preferences Hierarchy

Exceptions and preferences are ranked in a hierarchy. This means that, if multiple exceptions and preferences are assigned for an agent on a single day, Calendar Management analyzes the assignments and immediately selects the highest-priority exception for assignment, noting the others as declined.

However, declined exceptions and preferences are stored in the WFM database, in case there are changes to calendar information later. If, for example, a training session is canceled, an agent’s previously overridden day-off preference might then change status and be available for scheduling.

Time-Off Bidding

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

When you enable time-off bidding, WFM processes agent requests for time off that are within a configured bidding period on the same process date and time. This ensures that all requests that were made concurrently within this period are all granted, declined, or wait-listed on the same processing date, and eliminates the possibility that some requests for a vacation period are granted, while others are not. WFM grants these requests, based on seniority and/or rank. If there are available time slots, WFM grants any requests that were not previously granted, based on the order of submission.

Bidding Period

The bidding period is the time interval (in days) when agents can request time off. You determine and configure, edit, or delete bidding periods, based on your contact center requirements. However, you cannot change a bidding period that is being processed or has been processed. All requests within that period are processed on a specified date with a configured ranking, which determines how the requests are resolved.

Any changes you make to the bidding period can take effect immediately. For example, you might set the processing date to a date in the past when the actual bidding period, and the requests within that period, have not yet been processed. If items are flagged by a bidding period and you later delete the period before the processing date occurs, requests for that period are not processed unless you create a new bidding period that covers the dates of those items. If you enter items and then set the bidding period, items are processed if they are eligible for processing (if the preferred wait-list flag is present).

Each bidding period has a status that indicates when it is complete.

Request Processing

WFM batches all agent time off requests that fall within a bidding period, marks them as one request and, after all of the items pass validation, saves them to the WFM database. The initial request can contain any number of items. (Items entered in a previous version of WFM can contain only one item per request.) If items in a request fall within multiple bidding periods, WFM processes the request only once, on the latest processing date for all bidding periods.

You cannot modify single items in a request. If one item is modified, deleted, or recalled, the same action is performed on all items in the request.

Bid Processing

You can manually modify all items in a request before the processing date. Any request items that you manually grant are not automatically processed. WFM processes bidding items when the processing date conditions are met and the requests in a specified bidding period have not been processed.

During bid processing, any one of the following scenarios can occur:

- All items in request can be auto-granted (and possibly auto-published), based on the agent's seniority or rank.
- All items can be removed from the processing list and remain in **Preferred** status. They can become eligible for wait-list processing (on a first-come, first-served basis by the original submission date), if the following conditions occur:
  - Auto-grant does not apply for at least one item in the batch request.
  - The bidding period no longer applies (was deleted or edited) for at least one item in the request.
  - There are no time-off slots available to accommodate all batch request items on the bidding period processing date.

Time-Off Balances

Bidding time-off items display as regular time-off items and reduce the agent's time-off balance for the specific time-off type.

Wait-List Processing

Wait-listed items are not processed within bidding periods that are not yet processed. If there are no configured bidding periods or a bidding period for the wait-listed item date is already processed, the entire request is wait-listed and the `BatchRequest` configuration option in the `CalendarService` section of `WFM Server Application` is set to `true`. In previous WFM releases, items were wait-listed one by one.

WFM generates a warning when the time-off limit has been reached.

Time-Off Limits

Time-off limits can be changed during active the bidding period, but do not take effect until the processing date for that period. The bidding period implies that time-off limits are set to `0` and time-

off requests are not granted. If there is no configured bidding period for a certain date, time-off limits work as in previous WFM releases.

Using WFM Task Sequences

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

This topic presents information about how to create and use task sequences. It supplements the information in Shift Task Sequences.

Definitions

Within Genesys Workforce Management, a task sequence is a defined period of time during that agents can work only on one task or a specified set of tasks, called an activity set. You could also think of a task sequence as an *activity sequence*.

Activities

Activities (tasks) are work that is tracked and managed using Workforce Management. For example, a business might define the following activities in WFM Web for product A:

- Answering inbound calls
- Responding to e-mail
- Completing after-call work
- Performing scheduled callbacks
- Participating in chat sessions

These same activities might also be defined for products B and C.

You can use any activity set that is configured for a specific site and use the same activity set in any number of task sequences.

Back to Top

Purpose of Task Sequencing

Task sequences enable you to control how much the nature of an agent's work changes during part of a day. You can avoid asking agents to jump constantly from activity to activity—a situation that can result in agent confusion and fatigue and lower productivity. You do this by configuring task sequences. Once configured, you can assign a task sequence to any compatible shift in the same site.

Creating Task Sequences

You create task sequences in the Web for Supervisors **Policies** module, by selecting **Shifts > Task Sequences**. The **Task Sequence** pane is also where you configure task sequences for a shift, by selecting one of three options:

- **Do not use Task Sequences**
- **Task sequences must be used**
- **Task sequences can optionally be used**

If you want to use task sequences, create them by configuring the settings in the Task Sequence panes. See **Task Sequence** panes in Web for Supervisors below.

Also, see the procedure "Creating a Task Sequence" in *Workforce Management Web for Supervisors Help*, in the **Policies > Shifts > Shift Task Sequences** topic.

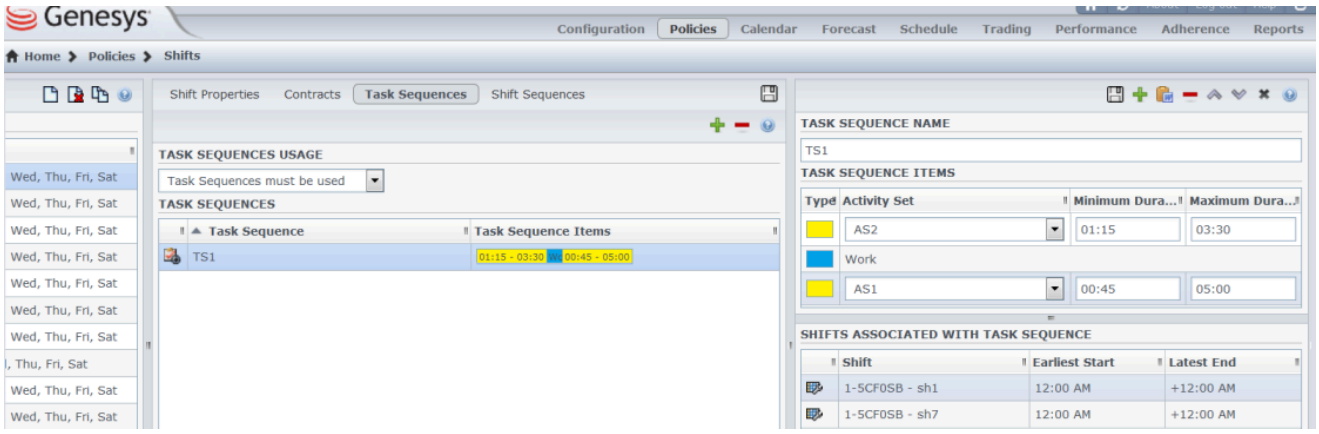


Figure: Task Sequence Settings

How Task Sequences Appear in WFM

After you configure task sequences and assign them to shifts (which are then assigned to agents), task sequences appear in the **Intra-Day** and **Agent-Extended** views. The task sequence appears as a blue area on the color-coded bar (unless you changed the color in the **Configuration** module in WFM Web Supervisors), where each type of schedule item is represented by a different color. The remainder of the color-coded bar (where no task sequence is scheduled) appears in a different color.

For example, the figure above shows a work item in blue and the remainder of the sequence in yellow.

Back to Top

Using E-mail Notifications in WFM

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

This topic provides information about how to use the Notifications module in Workforce Management (WFM) Web for Supervisors to configure e-mail notifications, by site, for the following types of events:

- Schedule trade status changes—This type of notification can be received by both agents and supervisors who are affected by a schedule trade proposal or response.
- Time off request status changes—This type of notification can be received by both agents and supervisors who are affected by a time off request.
- Schedule modifications—This type of notification can be received by the agent who is affected by the schedule change.

Schedule Trade Status Changes

WFM Daemon uses the following rules when sending schedule trade status change notifications:

- Both the proposing agent (creating the trade request) and responding agent (receiving the trade proposal) associated with the selected sites are notified when a trade status is **User-declined**, **User-approved**, **Auto-declined**, **Auto-approved**, or **Cancelled**.
- Supervisors who have appropriate security rights and are associated with the selected sites are notified when a trade status is **Pending**.
- Supervisors who have the appropriate security rights are notified when the **Schedule Trade** status changes from **Pending** to any other status.
- The responding agent is notified when the status of a trade proposal is **In Review** or **Open**.
- The proposing agent is notified when the response status of a trade proposal is **Accepted**, **In Review**, or **Cancelled**.

Time Off Request Status Changes

WFM Daemon uses the following rules when sending time off status change notifications:

- When a supervisor manually changes agent time off in the **Calendar** module, the affected agent is notified.

- When an agent requests time off (that is not **Auto-declined**), supervisors who have the appropriate security rights are notified.
- When automatic **Time-Off** request status changes occur, supervisors who have the appropriate security rights and affected agents are notified.
- When **Time-Off** balance-affecting changes occur, supervisors who have the appropriate security rights are notified.

For a complete description of the WFM security roles and privileges, see the Roles topic.

Schedule Modifications

When a supervisor changes one or more scheduled days, WFM sends a schedule modification notification to the affected agent in the configured site and to supervisors who have the appropriate security rights.

Procedures: How to Set Up E-mail Notifications in WFM

To set up e-mail notifications, complete all of the procedures in this section.

Configuring a Security Role

Purpose: To create a user security role to access e-mail notifications.

Start of Procedure

1. In WFM Web for Supervisors' **Configuration** module, select **Roles**.
2. In the Role pane, click **New**.
   A new security role appears in the Role pane with a default name.
3. In the Role Privileges pane, change the **Name** of the role (if desired).
4. In the list of Role Privileges, expand **Notifications** and check the boxes beside the privileges you want to assign to this role.

> **Important**
> Be sure to uncheck all privileges in the entire list that you do not want to assign to this role.

5. Click **Save**.

End of Procedure

For a complete description of all privileges, see Role Privileges.

Assigning a Security Role to a User

Purpose: To assign a security role to a user, enabling access e-mail notifications.

Start of Procedure

1. In WFM Web for Supervisors' **Configuration** module, select **Roles**.
2. At the top of the Role Privileges pane, click **Users**, and choose one of the following:
   - Click **Assign Users** to assign a WFM user to this role.
     A new pane opens, containing a list of WFM users.
   - Click **Import Genesys User** to assign a Genesys user to this role.
     A new pane opens, containing a list of Genesys users (in Genesys Administrator).

   > **Important**
   > New users are automatically granted access to all modules, objects, and sites.

3. Select the users you want to assign to this role and click **Apply**.
4. In the Users pane, click **Save Now**.

End of Procedure

Adding E-mail Addresses to the WFM Database

Purpose: To add Supervisor and Agent e-mail addresses to the WFM Database, so they can receive notifications.

> **Tip**
> To receive notifications, Supervisor and Agent e-mail addresses must be in the WFM Database.

Start of Procedure

1. In Genesys Adminstrator, enter these e-mail addresses into the **Supervisor** and **Agent person** objects.
   After WFM synchronization the e-mail address will be available in the WFM Database.

End of Procedure

Enabling Notifications in Web Supervisor

Purpose: To configure the Web Supervisor Application to enable notifications.

Start of Procedure

1. In WFM Web Supervisors' **Configuration** module, select **Notifications**.
   All three supported e-mail notification types are listed in the left-side navigation pane.
2. Select the one that you want to enable and in the right-side pane, click the **Targets** tab.
3. Check the sites for which you want to enable notifications. See the figure below.

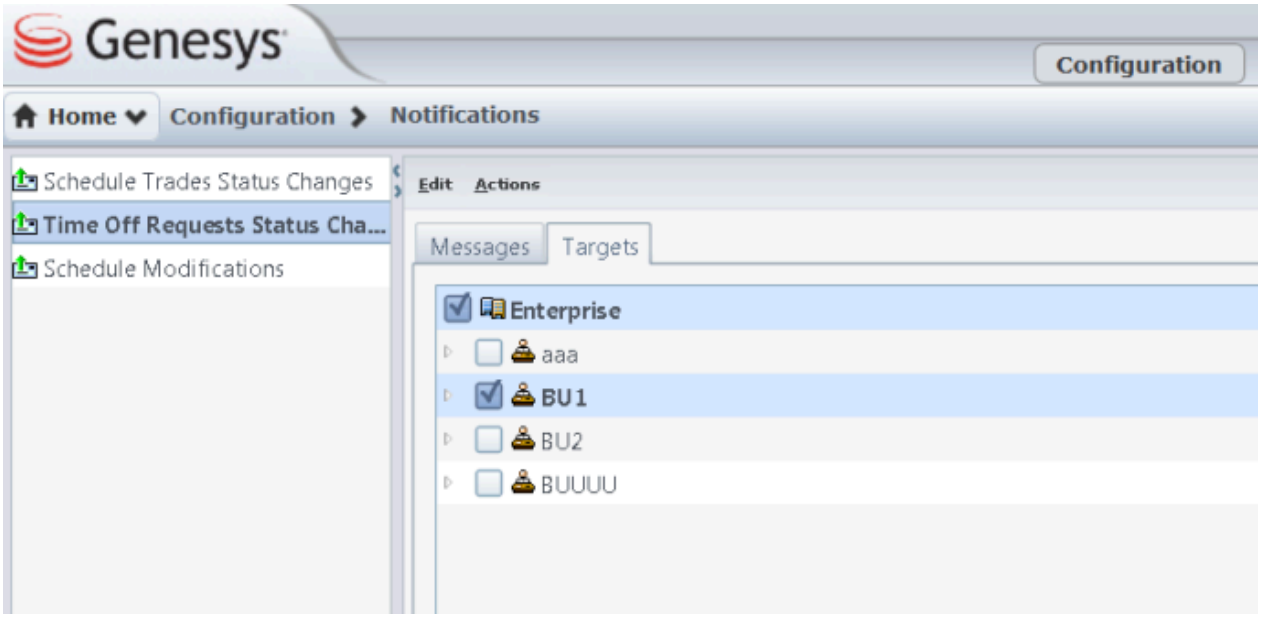


Figure: WFM Web Supervisor, with Notifications Selected

4. Use the **Messages** tab to configure the e-mail **Subject** and **Message** for all e-mail notifications of this type. You can change the text or keep the default.

End of Procedure

Scheduling Breaks and Meals in WFM

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

This topic describes how to use Workforce Management (WFM) to schedule meals and breaks in conjunction with Exceptions.

Pre-planned Breaks and Meals

Workforce Management enables you to pre-plan the breaks and meals (called *shift items*) that will be scheduled during a particular shift. You can define several parameters for these shift items, such as the *time window* during which they should be scheduled and whether they are paid or unpaid.

For example, if you set up a shift called **8-Hour Full-Time**, as part of the shift item configuration, you have specified that there should be a 15- minute paid break in the shift. The **Min Length from Shift Start** parameter is set to `2:00` (2 hours) and the **Max Length from Shift Start** parameter is set to `4:00`.

Additionally, your shift item configuration specifies that there should be an unpaid meal in the middle of the shift, with both the **Min Time Before This Meal** and **Min Time After This Meal** parameters set to `3:00`.

You have configured a rotating pattern for a particular agent that specifies that the agent should work the 8-hour full-time shift every day, starting at 8:00 a.m.

Due to the shift item configuration, when WFM builds a schedule scenario that includes this agent, it will try to schedule the break and meal in the following time windows:

- It will try to schedule the break in the time window between 10:00 a.m. and 12:00 p.m.
- It will try to schedule the meal in the time window between 11:00 a.m. and 1:30 p.m.

Sometimes the configured time windows for breaks and/or meals conflicts with planned exceptions, such as meetings, training sessions, or administrative time, that were entered through the WFM Calendar. In these cases, the behavior of the Scheduler varies, depending on the particular type of shift item and its properties.

Default Behavior of the Scheduler

This section describes the default behavior of the Scheduler in the following three scenarios.

When Unpaid Breaks Conflict with Exceptions

When the time window of an unpaid break is covered by a planned exception, the Scheduler *relaxes* the constraints of the break, in order to schedule it. That is, the time window is widened in both directions (if possible) so that the break can be scheduled adjacent to the exception—either immediately before the exception or immediately after the exception.

This relaxation of the break constraints occurs because unpaid breaks are considered mandatory by the Scheduler due to their effect on the paid time of the shift.

There will be instances when one or more unpaid break(s) cannot be scheduled, even though they are considered mandatory. For example, if a shift has a paid duration of 8 hours, and there is a granted exception in the Calendar that also has a paid duration of 8 hours, that would not leave any time remaining for the Scheduler to place an unpaid break. As a result, the unpaid break is not scheduled and a warning is generated when the scenario is built.

When Paid Breaks Conflict with Exceptions

Unlike unpaid breaks, by default, paid breaks are not considered mandatory. Therefore, if there is a conflict between a planned exception and a paid break—so that the time window of the paid break is covered by the exception—by default, the paid break is not scheduled when the scenario is built.

When Meals Conflict with Exceptions

Meals are considered a mandatory part of a shift, if the shift has a meal configured. If there is a conflict between a planned exception and a meal—so that the time window of the meal is covered by the planned time for the exception—one of two things happen when the scenario is built:

- WFM looks for, and finds, another shift that is compatible with the agent’s contract, and allows the exception to be scheduled, or
- When WFM resolves the conflicting items in the Calendar (prior to the schedule being built), it declines the exception unless it can find another shift that is compatible with the agent's contract that allows the exception to be scheduled. In this case, the exception is not scheduled and a warning is generated.

Changing the Behavior of the Scheduler

Configuration options are available that can change the default behavior of the Scheduler when breaks and/or meal time windows conflict with planned exceptions. These options are described in detail in this section.

Paid Breaks are Mandatory

This is an optional setting that controls whether or not a paid break is scheduled even when the time window of the break is covered by an exception. As described above, this is always the behavior of the Scheduler with unpaid breaks. However, if this setting is turned on, the same behavior occurs with paid breaks; if the time window of the paid break is covered by an exception, the paid break will be scheduled adjacent to the exception—either immediately before the exception or immediately after it.

There still might be times when some breaks cannot be scheduled, even if this setting is turned on, because there is not enough room in the shift to accommodate the exception and all the configured breaks. In this case, a warning will be generated when the scenario is built.

See below, some sample scenarios when WFM would not be able to schedule a break (paid or unpaid), regardless of whether the user defines this as being mandatory or not:

Examples... [+]

Example 1

There is a shift with an 8-hour duration but which is 7.5 paid hours. The user grants a paid exception that is 7.5 hours, right in the middle of the shift, leaving 15 minutes on either side of the exception in which to schedule any breaks. If there is a 30-minute unpaid break to schedule, it cannot be scheduled unless the user wants to allow breaks to be scheduled “during the exception” (see below for more information about that optional setting).

Example 2

There is an 8-hour shift from 8:00 a.m. to 4:00 p.m. The configuration of Break 1 (15-min) allows the break to be scheduled in a time window between 9:00 A.M -10:30 a.m. The configuration of Break 2 (15-min) allows the break to be scheduled within a time window between 2:00 p.m. - 4:00 p.m. The user grants an exception in the Calendar from 8:00 a.m. - 3:45 p.m. Unless the user wants to allow break(s) to be scheduled “during the exception,” one of the breaks cannot be scheduled, because there are only 15 minutes within the shift that is not already covered by the exception, and two 15-minute breaks to schedule.

Example 3

There is an 8-hour shift from 8:00 a.m. - 4:00 p.m. The user grants an exception in the Calendar from 8:00 a.m. - 11:45 a.m. and another exception from 12:00 p.m. - 4:00 p.m. This leaves only 15 minutes between the two exceptions in which to schedule any breaks. Unless the user wishes to allow break(s) to be scheduled “during the exception,” it is likely that one or more breaks would not be scheduled.

Example 4

Assume a shift from 8:00 a.m. - 1:00 p.m., one exception from 8:30 a.m. - 11:30 a.m., and two 1-hour breaks (the first one with configured window from 9:00 a.m. - 11:00 a.m., and the second one from 12:00 p.m. - 1:00 p.m.). Because the exception covers the first break, the break should be placed after the exception (because there is no room before it), from 11:30 a.m. - 12:30 p.m. Because of the scheduling of the first break, there is no room for the second break at all (but not because of the exception). In this case, one of the breaks would not be scheduled.

Suppress Break-Related Warnings

This is an optional setting to control whether schedule warnings that describe issues with break scheduling are hidden from the user. If you are scheduling a lot of long exceptions that you know will make it impossible for the Scheduler to fit in most of the breaks you have configured, you might want to check this setting so that the break-related warnings are suppressed. This allows you to focus on the other schedule warnings that you want to resolve.

Allow Breaks and Meals During Exception

For each **Exception Type**, this setting might be turned on. If this option is configured, then if a planned exception of this type is being scheduled and it covers the time window of a break, the Scheduler tries to schedule the break during the exception, preserving the original configured time window. The Scheduler might not always be able to accomplish this, so if it cannot schedule one or more breaks during the exception, it will next try to schedule them adjacent to the exception. This setting always affects the scheduling of unpaid breaks. This setting only affects the scheduling of paid breaks, if paid break scheduling is configured as mandatory.

This setting also controls whether the Scheduler will try to schedule meals during an exception, in cases when the configured time window of the meal is covered by the exception. However, if the Scheduler is unable to schedule the meal during the exception for some reason, it will not be scheduled adjacent to the exception as it will try to do with breaks.

It is important to note that when the user configures an exception type, such that break(s) and meal(s) could be scheduled during the exception, it does not mean that all of these shift items will be scheduled during the exception. For example, the user has configured a 15-minute break with a 5-minute start step. The break configuration permits the break to be scheduled somewhere between 8:45 a.m. and 10:15 a.m. There is an exception from 9:00 a.m. - 10:00 a.m.

The break could be scheduled in many possible places, including: `8:45 a.m. - 9:00 a.m.`
`9:00 a.m. - 9:15 a.m.`
`9:05 a.m. - 9:20 a.m.`
`9:10 a.m. - 9:25 a.m.`
`. . .`
`9:45 a.m. - 10:00 a.m.`

Also note that although the absolute start and end times of the exception will not be changed. For example, it is possible that the start of the exception could be covered by a break (both the break and the exception start at the same time).

Other Considerations

When there is no conflict between an exception and some break(s), but yet the exception makes it impossible for WFM to schedule the breaks according to all of their configured constraints, WFM continues its default software behavior, which is to relax the break constraints so that they can be scheduled.

Examples... [+]

Example 1

There is a 15-minute break that could be scheduled between 9:00 a.m. - 1:00 p.m., and a second 15-minute break that could be scheduled between 10:00 a.m. - 2:00 p.m. The user has configured that the minimum distance between these breaks must be 3 hours. The user has granted an exception that goes from 11:00 a.m. - 3:00 p.m. It is impossible to meet the minimum distance constraint and also schedule these two breaks within their configured time windows. Therefore, WFM could relax the break constraints, in order to meet the minimum distance constraint and one break would be scheduled prior to the exception, and the other break would be scheduled after the exception.

As described above, when relaxing break constraints to accommodate planned exceptions, WFM attempts to schedule the break immediately adjacent to the exception. However, it is not always possible to do this, and sometimes there will be a small duration of activity work scheduled between the break and the exception.

Example 2

The user has granted an exception from 12:00 p.m. - 1:05 p.m., and the configured time window for a particular 15-minute break specifies that the break must be scheduled somewhere between 1:15 p.m. - 2:15 p.m. Based on schedule coverage, WFM could place that break at 1:15 p.m., leaving just 10 minutes of activity work in between the exception and the break.

Example 3

The user has granted an exception from 12:00 p.m. - 2:00 p.m., and the configured time window for a particular 15-minute break specifies that the break must be scheduled somewhere between 1:00 p.m. - 2:16 p.m. WFM would only have between 2:00 p.m. - 2:16 p.m. in which to schedule the break. WFM could schedule the break from 2:01 p.m. - 2:16 p.m., leaving 1 minute of work between the exception and the break.

Also note that the features described in this section only address partial-day exceptions, not full-day exceptions. Therefore, if the user needs to schedule an exception that covers a worker's entire shift, they should consider using a full-day exception type.

Example 4

The user wants to grant an exception (type: meeting) after the Schedule has been built. In the Calendar, the user creates the `Calendar Item` and rebuilds the schedule. The meeting is reflected in the updated schedule and, in some cases, takes place during a paid break/meal or is adjacent to it.

> **Important**
>
> If the user attempts to schedule the meeting, by using the Meeting Planner (after the schedule is built), the meeting is not scheduled, nor is the warning messages suppressed—assuming that the system is configured in this way.

Hierarchy of Constraints

If breaks cannot be scheduled according to all of their configured constraints, WFM tries to satisfy the constraints in the following order:

1. Time window
2. Start step & start offset
3. Minimum distance between shift items
4. Maximum distance between shift items

Enabling Wait-lists

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

This topic contains lists and descriptions of the conditions and settings that make Workforce Management (WFM) wait-list functionality possible, and offers or refers to instructions for these tasks:

- Automatic Approval (sometimes called auto-granting)
- Auto-publishing
- Wait-listing
- Handling preferred time-off requests
- Making ungranted time-off requests *count* in a build
- Improved viewing time-off availability

If WFM cannot immediately grant a time-off request, it puts the request on a waiting list (called *wait-listing*). When all requirements are met, including configuration, WFM can grant the request and insert it into the Master Schedule automatically.

Overview

WFM considers all wait-listed time-off requests on a First-In, First-Out (FIFO) basis, using the date and time of the request. To determine if there are slots available for the wait-listed time-off request, the process first considers all time off instances that are **Granted/Granted** and **Scheduled** and have intersecting time steps with the wait-listed request for time off. If the **CalendarOverSchedule** option is set, then **Granted/Not Scheduled** time off instances are also taken into account before processing the wait-list.

Next, WFM considers all **Preferred** time-off requests that were submitted earlier than the wait-listed request that is currently being processed. Only **Preferred** time-off instances that count against the time-off limit according to their actual status are considered. Also, if an earlier **Preferred** request is not valid for time-off within the various limits, it is ignored.

You can configure the advance threshold for automated time-off wait-listing. You can also configure the minimum number of weeks in advance that a request can be made for each time-off rule; that minimum number is called the automatic approval threshold. If an ungranted time-off request in the

wait-list violates the automatic approval threshold, then WFM removes the request from the wait-list. That time-off request remains in the status **Preferred**, but it cannot remain wait-listed.

See Enabling automatic approval and setting the threshold below.

Events that might change the system's ability to grant time off, such as the agent's contract availability status, can affect the validity of any time-off request. Thus, wait-listing takes place on a schedule that is specified by a timeout option.

Enabling wait-listing

The following sections provide information about enabling wait-listing, and describes how auto-approvals, thresholds, and auto-publishing affect wait-listing.

Automatic approval of time-off requests

Wait-list automation is related to automatic approval functionality. The common workflow of wait-list automation is:

1. An agent's time-off request is automatically approved, but there are no time-off slots within the limits of the request.
2. The request is wait-listed until the appropriate time-off slots appear, and then automatically approved, if possible.
3. If automatic approval is enabled but the time-off request is beyond the automatic approval threshold, then the request cannot be wait-listed. It is instead saved with the status **Preferred**.

> **Important**
>
> Automatic approval must be enabled for wait-listing to work. When automatic approval is not enabled, supervisors manually grant time-off requests and requests are not wait-listed, even if requested by the agent.

When do Time-Off Slots Become Available?

Time-off slots become available when:

- A supervisor declines a time-off request
- An agent recalls a time-off request
- An agent or supervisor deletes a time-off request.

Time-off slots are taken (or unavailable) when a Granted or Preferred or Scheduled time-off request is entered.

Enabling automatic approval and setting the threshold

Automatic Approval must be enabled for wait-listing to work. To learn more, see Rules for Requests.

To enable and configure automatic approval, see the options:

- **Automatically approve time-off request if: Request is made by the following number of weeks in advance <*number*>**
- **Minimum requested time off is <*number*>** (hours and minutes).

Auto-publishing

When WFM automatically approves a time-off request, it can also insert it into the Master Schedule automatically, if auto-publishing is enabled.

Wait-listing

Wait-listing is a background function that is enabled by circumstances and configuration. The circumstances are: automatic approval must be enabled and an agent's time-off request must be valid (time off slots and agent credit must be available).

> ### Important
> The Auto-publishing and Wait-Listing features are not enabled by default. If you are interested in using these features in your environment, contact your Genesys Professional Services representative.

Notes

Why does a time-off request get wait-listed?

- The time-off request is valid (for example, not for time in the past).
- The time-off request is not for more time than the agent's time-off balance contains.
- The automatic approval option is enabled.

WFM will only wait-list those time-off requests that would have been automatically approved, if not for time-off limits. Also:

- If automatic approval is not set up in the accrual rule or the threshold has been passed, then that particular time-off cannot be wait-listed.
- The requesting agent should have sufficient time-off balance and there can be no other restrictions.
- If limits allow, the request will be granted and not wait-listed.

Treatment of Preferred Time-off Requests

Preferred time-off requests (having the status `Preferred`) are not considered during schedule building. When a master schedule is published, Preferred time-offs do not count against limits and agent balance anymore, because they are usually not scheduled.

Viewing Available Time-off Slots

Now Agents can see if the time off that they want is available before they request it. At the Time Off screen, just hover the cursor over any time slot, and you will see time off limits and wait list data about the time slot in a popup.

Use Cases and Notes

The following use cases discuss scenarios to compare availability and seniority, and provide some additional important rules to keep in mind.

When Availability Overrides Seniority

1. Request A asks for time slots that are not available (8AM-6PM). The time 8-9AM is not available, and so Request A is wait-listed.
2. Request B comes in later, but asks for time slots that are available (10AM-6PM).

The Result:
Request B is granted and Request A remains wait-listed, even though Request B came in later.

When Seniority Overrides Availability

1. Request A is wait-listed. The time slots for automatic approval of this request become available, but the wait-list timeout has not yet expired.So this request remains wait-listed.
2. Request B comes in later and asks for the same time slots as Request A.

The Result:

- If the time slots are available for both requests, Request B is honored immediately and Request A is honored after the wait-list timeout expires.
- If the time slots are available for only one request, Request B is wait-listed and after the wait-list timeout expires, Request A is granted first.

Rules to Remember

- When the wait-list process can grant a request, it changes the saved status of that request to granted (and removes the internal wait-list flag).

- If the auto-publish and wait-list functions are set up together, then granted requests are published to the master schedule.
- A wait-listed request can be granted only if: automatic approval is enabled, the time-off request is valid, there are no other restrictions.

WFM removes from the wait-list any time-off request that has expired (the time steps that it specifies are in the past) or belongs to an agent with no time-off balance.

Managing Overtime

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

Workforce Management (WFM) provides Overtime planning and tracking views for supervisors, into which they can enter overtime hours for a single agent or multiple agents by specifying the time interval and activity.

Overview

WFM automates the overtime insertion process in the following ways:

- Finds the appropriate shift definition for the extended shift
- Schedules break/meals on the overtime part of the shift
- Designates overtime by specifying a marked type

> **Important**
> Overtime functionality is available on the **Master Schedule** only. There is no overtime interface for Agents.

You can also enter overtime in the **Intra-Day**, **Agent Extended**, and **Weekly** views by inserting a work set with **Marked Time**. **Secondary** shifts and activities can be used to plan overtime.

Overtime Requirements View

In WFM Web, you can plan, set, and track how overtime requirements are met in the **Master Schedule > Overtime** view. To use this view, set the user security rights in **Configuration > Roles > Role Privileges > Access Overtime Requirement** or **Configuration > Users > Role Privileges > Access Overtime Requirement**.

Overtime data is displayed in a grid, which has an editable **Overtime Requirement** column and a

read-only **Overtime Scheduled** column. The **Overtime Scheduled** column is calculated according to the scheduled **Marked Time** (ensure the **Overtime** option is checked). The following optional columns contain information that justifies the need for overtime hours (see the figure below):

- Staffing—Calculated and required staffing.
- Difference—Schedule coverage showing the difference between calculated and required.
- Variance—Equals the coverage minus the calculated, and is the anticipated unplanned overhead.

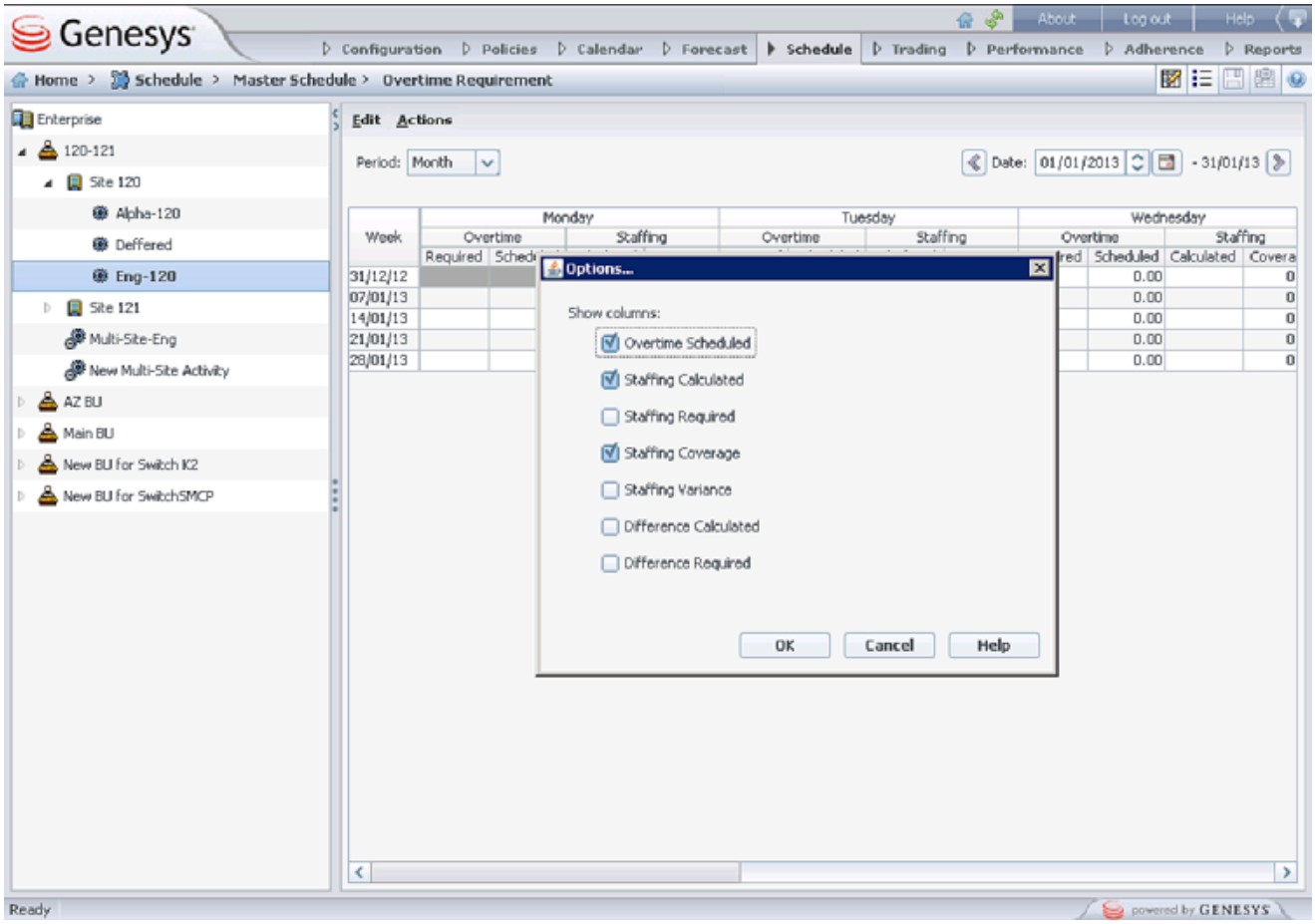


Figure: Overtime Requirement View with Monthly Granularity

Assigning Secondary Skill Sets for Overtime

Shifts are assigned to contacts in Web for Supervisors in the **Policies > Contracts** view (see the figure below) as either **Primary** or **Secondary**. **Secondary** assignments are used for overtime only and represent a broader set of constraints, while **Primary** assignments are used for other functionality.

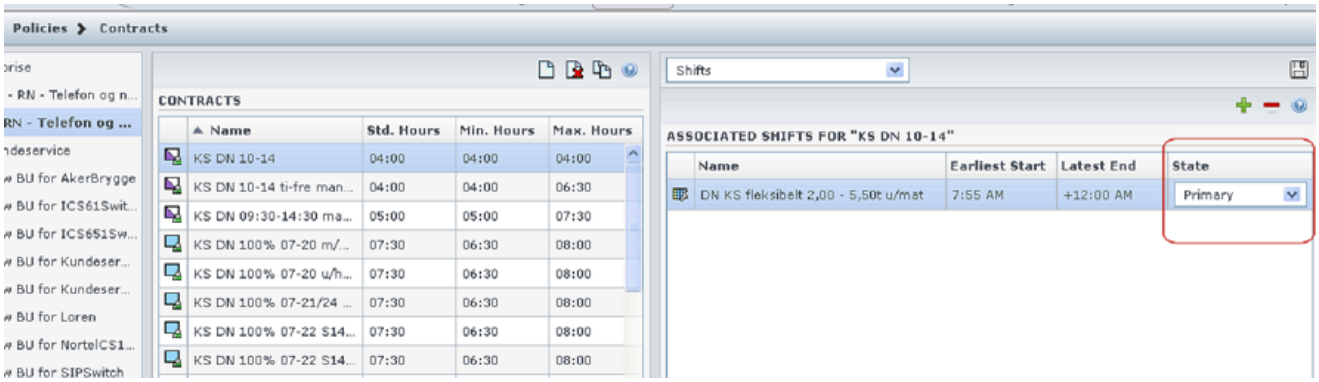


Figure: Web for Supervisors—Policies > Contracts > Shift Assignment

Secondary activities can be defined in one of two ways in Web for Supervisors in the **Configuration > Agents > Activities** view (see the figure below):

1. Assigning skills to the agent as **Secondary**
2. Assigning activities as **Secondary** with an effective date.

During overtime, agents can work on activities for which they are not normally scheduled. Therefore, entering overtime for an agent can change the agent's work day so that the usual shift definition no longer fits.

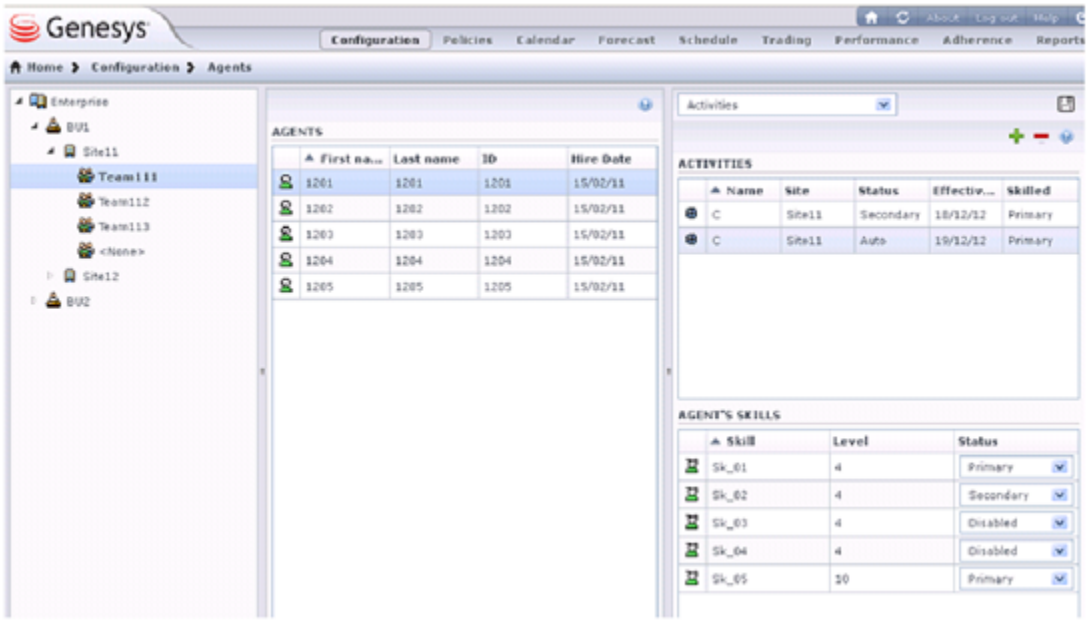


Figure: Web for Supervisors—Configuration > Agents > Activities View

Configuring Overtime for Individual Agents

The **Overtime** view is available in the **Master Schedule** only. However, you must publish a scenario with the inserted work sets before the information can be seen in the Overtime view.

You can schedule overtime for an agent by inserting a work set and using **Marked Time** to mark it as overtime. Use one the wizards in the following views:

- **Insert Wizard—Schedule > Intra-Day**, **Agent-Extended** views (**Master** and **Scenarios**)
- **Insert Multiple Wizard—Schedule > Intra-Day**, **Agent-Extended**, and **Weekly** views (**Master** and **Scenarios**)

Using the Insert Multiple Wizard to Set Overtime for an Agent

Purpose: To add a work set for an individual agent for overtime by using the **Insert Multiple Wizard** in WFM Web.

Start of Procedure

1. In the **Intra-Day**, **Agent Extended**, or **Weekly** view, select **Insert Multiple** from one of the following:
   - Actions toolbar
   - Actions menu
   - On the agent's schedule, Right-click and select the **Shortcut** menu (not in Weekly view)

   If you have unsaved changes, WFM Web prompts you to save them before proceeding.
2. In the **Insert Multiple Wizard**, select **Insert Work set**.
3. Create a new overtime work set by selecting **Marked Time** for the designated work type.

> ### Important
> You must select **Marked Time**; otherwise the resulting work set is not created as overtime (see the figure below). Also, inserting a work set over an existing shift can change the scheduled activities in the affected interval.

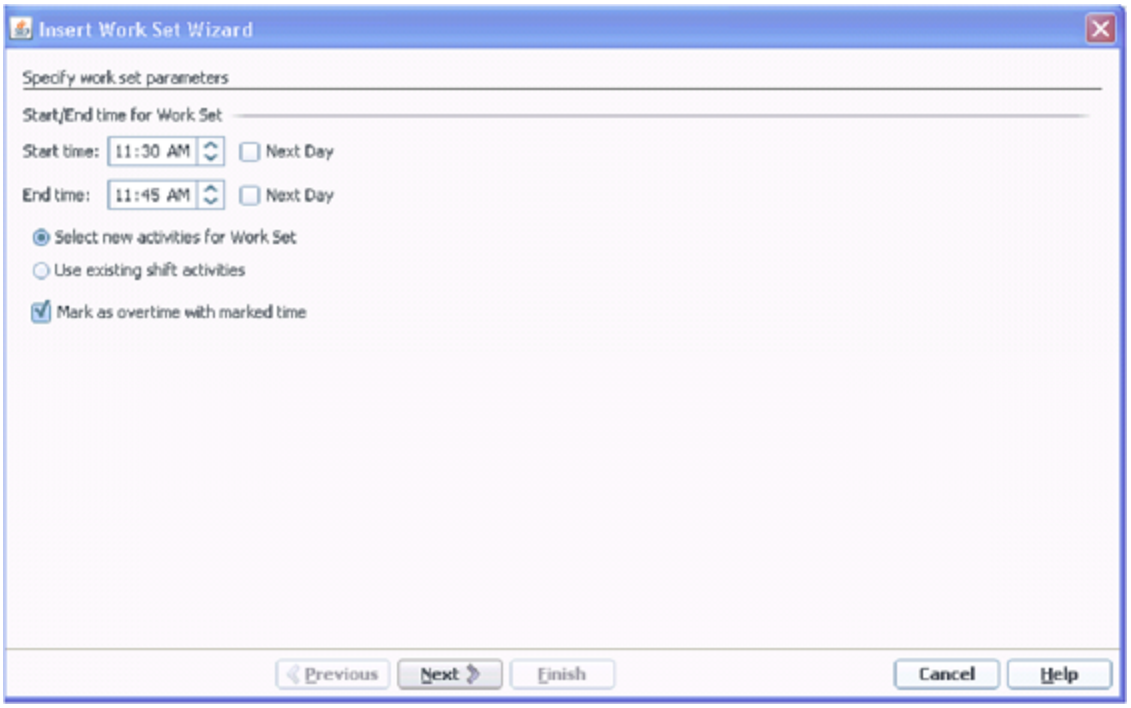


Figure: Insert Multiple Wizard—Inserting an Overtime Work Set **End of Procedure**

Tracking Scheduled Overtime Results

You can use the following views to track scheduled overtime results:

- **Schedule > Schedule Scenarios > Intra-Day** view in the **Performance Data** pane
- **Schedule > Master Schedule > Intra-Day** view in the **Performance Data** pane
- **Schedule > Master Schedule** > **Overtime** view

The **Performance Data** pane separates the scheduled overtime part of the coverage within the calculated staffing graph and distinguishes it from the overtime requirement. See the figure below.

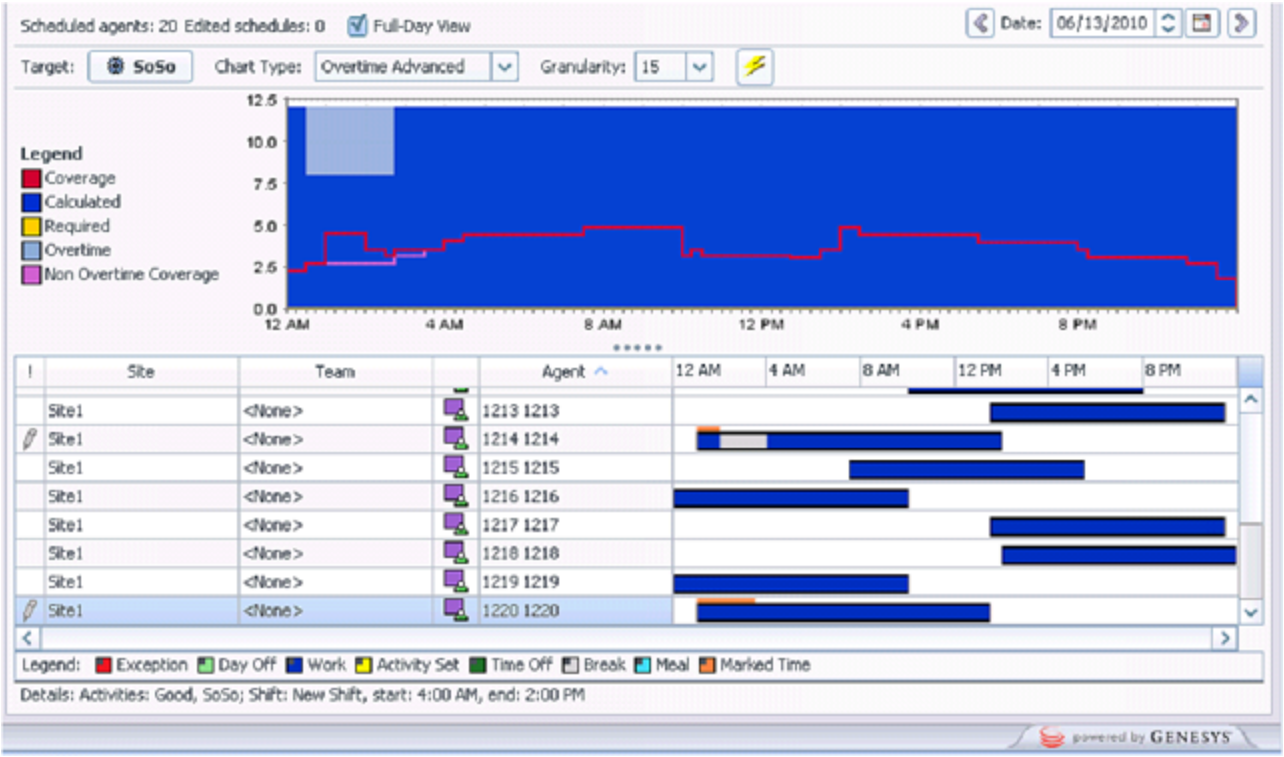


Figure: Calculated Staffing Graph—Full Day View

Overtime Bidding Offers

Supervisors with the appropriate role privileges can create Overtime Offers for a specific activity or multi-site activity in Web for Supervisors' **Overtime Bidding** view in the **Schedules** module. Offers are associated with the Master Schedule only and WFM uses the offer properties and other data in the automatic overtime scheduling process—the *other data* being the overtime slots chosen by agents when they are bidding on open offers.

Supervisors must have the Access Overtime Requirements and Overtime Bidding role privileges for the Schedule module. Then, they can track the overtime requirement that is currently scheduled, by checking the Marked Time that is denoted as "mark as overtime" in the agents' schedule for the activity or multi-site activity associated with the Overtime Offer.

Agents use the **Bidding** module in Web for Agents to see open schedules bidding scenarios or overtime offers, on which to bid. When they select an overtime offer, they can then choose the overtime slots they want added to their schedules. Agent see overtime offers only after the supervisor marks the offer as "open" and only if the supervisor has associated the agent and his/her site with the offer.

Continuous Processing of Overtime Offers

You can enable WFM to continuously process Overtime offers, which have FIFO selected for agent bid

ranking criteria. To enable this functionality, contact your Genesys representative.

For more information about Overtime bidding, see the *Workforce Management Web for Supervisors Help* and *Workforce Management Agent Help*.

For information about WFM Role Privileges see Assigning Role Privileges and Schedule Role Privileges topics in the *Workforce Management Web for Supervisors Help*.

WFM Metrics

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

Find everything you need to know about Workforce Management (WFM) metrics, by clicking on these topic links:

> **Important**
> Enabling the use of WFM statistics and metrics in your environment might require assistance from Genesys Professional Services. For more information, contact Genesys Customer Care.

> **Tip**
> A period in these metrics refers to the specific granularity of the report being run. For *Intra-day* granularity a period is 15 minutes; for *Daily* granularity a period is 1 day; and so on.

Schedule Summary View and Reporting

> **Important**
>
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

In this topic, find information about the Workforce Management Schedule Summary View and Report metrics. To go directly to a specific metric, click any link in the tables below.

• AHT – Difference • AHT – Forecasted • AHT – Scheduled	• ASA – Difference • ASA – Forecasted • ASA – Scheduled	• Budget – Difference • Budget – Forecasted • Budget – Scheduled	• Coverage – Difference • Coverage – Published • Coverage – Scheduled	• Difference – Calculated • Difference – Required
• Interaction Volume – Difference • Interaction Volume – Forecasted • Interaction Volume – Scheduled	• Number of Agents	• Occupancy – Difference • Occupancy – Forecasted • Occupancy – Scheduled	• Service Level – Difference • Service Level – Forecasted • Service Level – Scheduled	• Staffing – Calculated • Staffing – Required

> **Tip**
>
> A period in these metrics refers to the specific granularity of the report being run. For *Intra-day* granularity a period is 15 minutes; for *Daily* granularity a period is 1 day; and so on.

See Other Metrics Topics

- Contact Center Performance Report Metrics
- Agent Adherence Report Metrics
- Endnotes (for Metrics)

Number of Agents

Number of Agents is also called *Headcount* or *Agents in Seats*. If an agent is multi-skilled and is scheduled for more than one activity for a given period, he will actually count as 1 headcount for *each* of the activities for which he is scheduled. Therefore, in comparison with Coverage, Headcount can *double count* an agent if he is multi-skilled and is scheduled for more than one activity during a particular period.

For example, if an agent is scheduled for two activities in a particular 15-minute timestep, he might count as 0.5 towards the Coverage of each of those activities, but he would count as 1 Headcount towards each of those activities. Unlike Coverage, where an agent could count towards partial coverage if he's scheduled for something other than activity work for part of a 15-minute timestep, for Headcount it is *all or nothing*. This means that as long as an agent is scheduled for at least 1 minute of work on an activity during a given 15-minute timestep, he will count as 1 Headcount towards that activity.

Back to Top

How the Total / Average is Calculated

The value for Headcount in the totals/averages row at the top of this view (or bottom of the report) is a simple average of all the values for all the timesteps of the selected time period (which can be: Intra-day, Daily, nWeeks, or Monthly).

Service Level – Scheduled

The Service Level that you should achieve on this activity, with the number of agents currently scheduled for this activity. Due to agent rounding, this value may differ from the original service level objective that was stated when the staffing forecast was built.

For example, WFM might forecast a staffing requirement of 12 agents to meet a service level objective of 80% of interactions answered within 20 seconds. But a Service Level Percentage Forecast may report a higher number, such as 83.48%. This is because 12 was the minimum number of agents required to meet the 80% service level objective but, with that number of agents, the contact center can be expected to achieve a slightly better service level than 80%. With one less agent (11 agents), the contact center would not be expected to achieve the 80% service level.

How the Total / Average is Calculated

A weighted average, calculated across the open hours:

$$\text{AVG SL Scheduled} = \sum(\textit{Scheduled } SL_i * \textit{Forecasted } IV_i) \,/\, \sum(\textit{Forecasted } IV_i)$$

Where:

$\textit{Scheduled } SL_i$ = Calculated Service Level based on the number of scheduled agents for $\textit{timestep}_i$
$\textit{Forecasted } IV_i$ = Forecasted Interaction Volume for $\textit{timestep}_i$
$\textit{timestep}_i$ = timestep number over the open hours

Back to Top

Service Level—Forecasted

The Service Level Percentage objective that you should be able to achieve when staffing with the number of agents from the staffing forecast (also known as Budget – Difference).

How the Total / Average is Calculated

A weighted average, weighted by Forecasted IV:

$$\text{AVG SL Forecasted} = \Sigma\ (\textit{Forecasted } SL_i * \textit{Forecasted } IV_i) \ / \ \Sigma\ (\textit{Forecasted } IV_i)$$

Where:

$\textit{Forecasted } SL_i$= Forecasted Service Level for $\textit{timestep}_i$
$\textit{Forecasted } IV_i$= Forecasted Interaction Volume for $\textit{timestep}_i$
$\textit{timestep}_i$= timestep number over the open hours

Service Level – Difference

Service Level – Scheduled minus Service Level – Forecasted.

Interaction Volume – Scheduled

The number of interactions that can be handled based on the schedule coverage. Calculated by using the inverse of the WFM's staffing forecast algorithm.

WFM uses a *modified Erlang* algorithm to derive Calculated Staffing based on the IV, the AHT, and service objectives stated when building the forecast. Therefore, to calculate the scheduled interaction volume, WFM uses that formula *in reverse*.

Back to Top

How the Total / Average is Calculated

The sum is across the entire report time range.

Interaction Volume – Forecasted

The interaction volume taken from the Master Forecast.

How the Total / Average is Calculated

The sum across the entire report time range.

Interaction Volume – Difference

Interaction Volume – Scheduled minus Interaction Volume – Forecasted.

AHT – Scheduled

The Average Handling Time (AHT) per interaction that you should achieve, based on the schedule coverage. Calculated by using the *inverse* of the WFM's staffing forecast algorithm.

WFM uses a *modified Erlang* algorithm to derive Calculated Staffing, based on the IV, AHT, and service objectives stated when building the forecast. Therefore, to calculated the scheduled AHT, it uses that formula *in reverse*.

Back to Top

How the Total / Average is Calculated

A weighted average, weighted by Forecasted IV:

$$\text{AVG AHT Scheduled} = \Sigma\ (\textit{Scheduled AHT}_i * \textit{Forecasted IV}_i) / \Sigma\ (\textit{Forecasted IV}_i)$$

Where:

$\textit{Scheduled AHT}_i$= Scheduled Average Handling Time for $\textit{timestep}_i$
$\textit{Forecasted IV}_i$= Forecasted Interaction Volume for $\textit{timestep}_i$
$\textit{timestep}_i$= timestep number over the open hours

AHT – Forecasted

Average Handling Time of interactions, taken from the Master Forecast.

How the Total / Average is Calculated

A weighted average, weighted by Forecasted IV:

$$\text{AVG AHT Forecasted} = \Sigma\ (\textit{Forecasted AHT}_i * \textit{Forecasted IV}_i) / \Sigma\ (\textit{Forecasted IV}_i)$$

Where:

$\textit{Forecasted AHT}_i$= Forecasted Average Handling Time for $\textit{timestep}_i$
$\textit{Forecasted IV}_i$= Forecasted Interaction Volume for $\textit{timestep}_i$

$timestep_i$= timestep number over the open hours

AHT – Difference

AHT – Scheduled minus AHT – Forecasted.

Back to Top

Budget – Scheduled

The budget for this schedule based on the number of agents from the Coverage – Scheduled column. The calculation is based on a full-time equivalent's hourly wage, as well as on the Planned Overhead % and Unplanned Overhead % (which were specified when the staffing forecast was built).

For a 15-minute timestep, the formula is:

(Coverage/4)((100/(100-Planned Overhead))*(100/100-Unplanned Overhead)* Hourly Wage)*

The unit of measure is in whatever monetary unit was used when the Hourly Wage was specified while building the forecast.

How the Total / Average is Calculated

This figure is in the sum across the entire report time range.

Budget – Forecasted

The budget for the schedule based on the number of agents from the "Staffing – Calculated" column, based on a full-time equivalent's hourly wage, as well as the Planned Overhead % and Unplanned Overhead % (specified when the staffing forecast was built).

For a 15-minute timestep, the formula is:

(Staffing/4)((100/(100-Planned Overhead))*(100/100-Unplanned Overhead)* Hourly Wage)*

The unit of measure is in whatever monetary unit was used when the Hourly Wage was specified when building the forecast.

How the Total / Average is Calculated

This figure is in the sum across the entire report time range.

Back to Top

Budget – Difference

Budget – Scheduled minus Budget – Forecasted.

Staffing – Calculated

A value taken directly from the staffing forecast, for the particular timestep.

How the Total / Average is Calculated

The value in the column footer is calculated based on the same approach as what is described for Coverage – Scheduled.

Staffing – Required

A value taken directly from the staffing forecast, for the particular timestep. It will be populated only if you put some values in the optional `Required Staffing` column in your staffing forecast.

How the Total / Average is Calculated

The value in the column footer is calculated based on the same approach as what is described for Coverage – Scheduled.

Difference – Calculated

Coverage – Scheduled minus Staffing – Calculated.

Back to Top

How the Total / Average is Calculated

The value in the column footer is calculated based on the same approach as what is described for Coverage – Scheduled.

Difference – Required

Coverage – Scheduled minus Staffing – Required.

Coverage – Scheduled

The actual amount of time that an agent should count towards coverage of the work in this time period. If an agent is multi-skilled and is scheduled for multiple activities during a given period, he may count fractionally towards the coverage of each activity (for example, as 0.5 toward each of two activities for which he's scheduled).

If an agent has something other than activity work scheduled for part of a period, that will be subtracted from the amount of time he's counted towards the coverage of that activity work. For example, an agent who is scheduled for an activity for a given 15-minute timestep but has a break for the first 5 minutes of that timestep, would count as 0.67 towards the coverage of that activity for that period.

Back to Top

How the Total / Average is Calculated

This figure is in FTEs. For the calculation, see Endnote 1.

The value in the totals/averages row at the top of this view (or bottom of the report) is calculated as follows:

1. WFM calculates the sum of the agents which are covering this activity within each timestep during the day.
2. The value calculated in step 1 is multiplied by 15 minutes in order to get the total time of activity work.
3. The value calculated in step 2 is divided by the value set for `Paid Hours a Day`, which was entered while building staffing forecast for this activity.

Coverage – Published

The original values from the Coverage – Scheduled column the last time a schedule scenario was published to the Master Schedule.

How the Total / Average is Calculated

The value in the column footer is calculated based on the same approach as what is described for Coverage – Scheduled.

Coverage – Difference

Coverage – Scheduled minus Coverage – Published.

ASA – Scheduled

The Average Speed of Answer that you should achieve on this activity, with the number of agents currently scheduled for this activity.

The `totals/average` row for ASA Scheduled reports a weighted average, calculated across the open hours and weighted by Forecasted IV (exactly as for Service Level – Scheduled):

A weighted average, weighted by Forecasted IV:

$$\text{AVG ASA Scheduled} = S\ (Scheduled\ ASA_i * Forecasted\ IV_i) \ /\ S\ (Forecasted\ IV_i)$$

Where:

$Scheduled\ ASA_i$= Calculated Average Speed of Answer based on the number of scheduled agents for $timestep_i$
$Forecasted\ IV_i$= Forecasted Interaction Volume for $timestep_i$
$timestep_i$= timestep number over the open hours

Back to Top

ASA – Forecasted

The `totals/average` row for ASA Forecasted reports a weighted average, calculated across the open hours and weighted by Forecasted IV (exactly like for Service Level – Forecasted):

A weighted average, weighted by Forecasted IV:

$$\text{AVG ASA Forecasted} = S\ (Forecasted\ ASA_i * Forecasted\ IV_i) \ /\ S\ (Forecasted\ IV_i)$$

Where:

$Forecasted\ ASA_i$= Calculated Average Speed of Answer based on the number of scheduled agents for $timestep_i$
$Forecasted\ IV_i$= Forecasted Interaction Volume for $timestep_i$
$timestep_i$= timestep number over the open hours

ASA – Difference

ASA – Scheduled minus ASA – Forecasted.

Occupancy – Scheduled

The Occupancy that you *should achieve* on this activity, with the number of agents currently scheduled.

How the Total / Average is Calculated

The `totals/average` row for Occupancy reports a weighted average, calculated across the open hours and weighted by Forecasted IV (exactly as for Service Level – Scheduled:

A weighted average, calculated across the open hours and weighted by Forecasted IV:

$$\text{AVG Occupancy Scheduled} = S\ (Scheduled\ Occ_i * Forecasted\ IV_i) \ /\ S\ (Forecasted\ IV_i)$$

Where:

$Scheduled\ Occ_i$= Calculated Occupancy based on the number of scheduled agents for $timestep_i$
$Forecasted\ IV_i$= Forecasted Interaction Volume for $timestep_i$
$timestep_i$= timestep number over the open hours

Occupancy – Forecasted

The Occupancy objective that you *should be able to achieve* when staffing with the number of agents from the staffing forecast (Staffing – Calculated).

Back to Top

How the Total / Average is Calculated

The `totals/average` row for Occupancy reports a weighted average, calculated across the open hours and weighted by Forecasted IV (exactly as for Service Level – Forecasted):

A weighted average, weighted by Forecasted IV:

$$\text{AVG Occupancy Forecasted} = S\ (Forecasted\ Occ_i * Forecasted\ IV_i) \ /\ S\ (Forecasted\ IV_i)$$

Where:

$Forecasted\ Occ_i$= Forecasted Occupancy for $timestep_i$
$Forecasted\ IV_i$= Forecasted Interaction Volume for $timestep_i$
$timestep_i$= timestep number over the open hours

Occupancy – Difference

Occupancy – Scheduled minus Occupancy – Forecasted.

Back to Top

Contact Center Performance Report Metrics

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

In this topic, find information about the Workforce Management Contact Center Performance Report metrics. Click any link in the Contents to go directly to a specific metric.

> **Tip**
> A period in these metrics refers to the specific granularity of the report being run. For *Intra-day* granularity a period is 15 minutes; for *Daily* granularity a period is 1 day; and so on.

See Other Metrics Topics

- Schedule Summary Report Metrics
- Agent Adherence Report Metrics
- Endnotes (for Metrics)

> **Important**
>
> The *Difference* calculation is controlled by the WFM Web `Application` option `RevertDiffCalculation` in Management Framework's Configuration Manager.
>
> `false (default): Scheduled or Forecasted minus Actual`
> `true: Actual minus Scheduled or Forecasted`
>
> This document uses the default value.

Coverage – Scheduled

The actual amount of time that an agent should count toward coverage of the work in this time period. If an agent is multi-skilled and is scheduled for multiple activities during a given period, he may count fractional amounts of time toward the coverage of each activity (for example, 0.5 toward each of two activities for which he is scheduled).

If an agent has something other than activity work scheduled for part of a period, that will be subtracted from the amount of time counted toward the coverage of that activity work. For example, an agent who is scheduled for an activity for a given 15-minute timestep, but who has a break for the first five minutes of that timestep, would count as 0.67 toward the coverage of that activity for that period.

Back to Top

How the Total / Average Is Calculated

This figure is in FTEs. For the calculation, see Endnote 1.

The value in the totals/averages row at the top of this view (or at the bottom of the report) is calculated as follows:

1. WFM calculates the sum of the agents who are covering this activity within each timestep during the day.
2. The value calculated in Step 1 is multiplied by 15 minutes in order to get the total time of activity work.
3. The value calculated in Step 2 is divided by the value set for `Paid Hours a Day`, which was entered while building the staffing forecast for this activity.

Coverage – Optimal

The coverage that would have been required in order to meet the original service objectives, based on the actual IV and AHT.

Coverage – Difference

Coverage – Optimal minus Coverage – Scheduled.

Coverage – Percentage

$$\frac{\text{Coverage – Optimal}}{\text{Coverage – Optimal minus Coverage – Scheduled}}$$

in concept: Optimal / (Optimal – Scheduled)

Number of Agents – Scheduled

The number of agents scheduled for this period, also known as *headcount*.

Back to Top

How the Total / Average is Calculated

A simple average across the entire report time range. Thus, even if activity is only open for a portion of the day, if this report were run for a intra-day period, the average would be calculated over 96 timesteps.

Number of Agents – Actual

The number of agents who were actually logged in during this period, also known as *headcount*.

How the Total / Average is Calculated

A simple average of the number of time steps when agents were logged in.

$$\Sigma\ (Agents_i)\ /\ \text{Number of time steps}$$

Where

$Agents_i$ is the number of agents logged in (as reported by Stat Server) during $timestep_i$

Number of Agents – Difference

Number of Agents – Scheduled minus Number of Agents – Actual.

Number of Agents – Percentage of Difference

$$\frac{(\text{Number of Agents – Scheduled minus Number of Agents – Actual})}{\text{Number of Agents – Scheduled}}$$

in concept: (Scheduled – Actual) / Scheduled

Interaction Volume – Forecasted

The number of interactions forecasted for this period (taken directly from the Master Forecast).

Back to Top

How the Total / Average is Calculated

This Sum is spread across the entire report time range.

Interaction Volume – Actual

The number of interactions actually received. The exact nature of this metric will depend on what Stat Server statistic is configured for Interaction Volume.

For example, for voice interactions, normally this is based on Number of Calls Entered.

How the Total / Average is Calculated

The sum is of Interaction Volume for each time step within the report time range.

$\Sigma\ (IV_i)$

Where:

IV_i is the Interaction Volume recorded by Stat Server during $timestep_i$

Interaction Volume – Difference

Interaction Volume – Forecasted minus Interaction Volume – Actual.

Interaction Volume – Percentage

$$\frac{(\text{Interaction Volume – Forecasted minus Interaction Volume – Actual})}{}$$

Interaction Volume – Forecasted
in concept: (Forecasted - Actual) / Forecasted

Back to Top

AHT – Forecasted

Forecasted Average Handling Time for this period (taken directly from the Master Forecast)

How the Total / Average is Calculated

A weighted average, weighted by Forecasted IV:

$$\text{AVG AHT Forecasted} = \Sigma\ (\textit{Forecasted AHT}_i * \textit{Forecasted IV}_i) \ / \ \Sigma\ (\textit{Forecasted IV}_i)$$

Where:

$\textit{Forecasted AHT}_i$ = Forecasted Average Handling Time for $\textit{timestep}_i$
$\textit{Forecasted IV}_i$ = Forecasted Interaction Volume for $\textit{timestep}_i$
$\textit{timestep}_i$ = timestep number over the open hours

AHT – Actual

Actual Average Handling Time for calls handled during this period. This metric is based on what Stat Server statistics are configured for Total Handle Time and Number of Calls Handled.

How the Total / Average is Calculated

A weighted average, weighted by Number of Calls Handled:

$$\Sigma\ (\textit{AHT}_i * \textit{CallsHandled}_i) \ / \ \Sigma\ (\textit{CallsHandled}_i)$$

Where:

$\textit{AHT}_i$ = AHT recorded by Stat Server for $\textit{timestep}_i$
$\textit{CallsHandled}_i$ = Number of interactions handled during timestep_i as recorded by Stat Server.

AHT – Difference

AHT – Forecasted minus AHT – Actual.

AHT – Percentage

$$\frac{\text{(AHT – Forecasted minus AHT – Actual)}}{\text{AHT – Forecasted}}$$

in concept: (Forecasted - Actual) / Forecasted

Back to Top

Service Level – Scheduled

The Service Level that was scheduled to be achieved, based on the scheduled number of agents.

How the Total / Average is Calculated

A weighted average, weighted by Forecasted IV:

AVG SL Scheduled = Σ ($Scheduled\ SL_i$ * $Forecasted\ IV_i$) / Σ ($Forecasted\ IV_i$)

Where:

$Scheduled\ SL_i$ = Calculated Service Level based on the number of scheduled agents for $timestep_i$
$Forecasted\ IV_i$ = Forecasted Interaction Volume for $timestep_i$
$timestep_i$ = timestep number over the open hours

Service Level – Actual

The Service Level that was actually achieved. This metric is based on what Stat Server statistics are configured for Service Factor, Number of Calls Distributed, Average Speed of Answer, and Time Range.

How the Total / Average is Calculated

A weighted average, weighted by Number of Calls Distributed:

Σ (SFi * $CallsDistributed_i$) / Σ ($CallsDistributed_i$)

Where:

SF_i= Service Factor recorded by Stat Server for $timestep_i$

$CallsDistributed_i$= Number of calls distributed during $timestep_i$ as recorded by Stat Server. (This value does not appear in the report but is recorded by Stat Server. For more details, see Endnote 2.)

ASA – Scheduled

The Average Speed of Answer that was scheduled to be achieved, based on the scheduled number of agents. Calculated by using the *inverse* of the WFM's staffing forecast algorithm.

WFM uses a *modified Erlang* algorithm to derive Calculated Staffing, based on the IV, AHT, and service objectives such as ASA that were stated when building the forecast. Therefore, to calculated the Scheduled ASA it uses that formula *in reverse*.

How the Total / Average is Calculated

A weighted average, weighted by Forecasted IV:

$$\text{AVG ASA Scheduled} = \Sigma\ (\textit{Scheduled ASA}_i\ *\ \textit{Forecasted IV}_i)\ /\ \Sigma\ (\textit{Forecasted IV}_i)$$

Where:

$\textit{Scheduled ASA}_i$ = Calculated ASA based on the number of scheduled agents for $\textit{timestep}_i$
$\textit{Forecasted IV}_i$ = Forecasted Interaction Volume for $\textit{timestep}_i$
$\textit{timestep}_i$ = timestep number over the open hours

Back to Top

ASA – Actual

The Average Speed of Answer that was actually achieved. This metric is based on what Stat Server statistics are configured for ASA.

How the Total / Average is Calculated

A weighted average, weighted by Number of Calls Distributed:

$$\Sigma\ (\textit{ASA}_i\ \text{x}\ \textit{CallsDistributed}_i)\ /\ \Sigma\ (\textit{CallsDistributed}_i)$$

Where:

$\textit{ASA}_i$ = Average Speed of Answer for $\textit{timestep}_i$ as recorded by Stat Server
$\textit{CallsDistributed}_i$ = Number of calls distributed during $\textit{timestep}_i$ as recorded by Stat Server

Abandons Factor – Scheduled

The expected number of abandoned interactions based on the scheduled number of agents. Calculated by using the *inverse* of the WFM's staffing forecast algorithm.

WFM uses a *modified Erlang* algorithm to derive Calculated Staffing, based on the IV, AHT, and service objectives such as Abandonment percentage that were stated when building the forecast. Therefore, to calculate the Scheduled Abandons, WFM uses that formula *in reverse*.

How the Total / Average is Calculated

A weighted average, weighted by Forecasted IV:

$$\text{AVG AF Scheduled} = \Sigma\ (Scheduled\ AF_i * Forecasted\ IV_i) \ /\ \Sigma\ (Forecasted\ IV_i)$$

Where:

$Scheduled\ AF_i$ = Calculated Abandon Factor based on the number of scheduled agents for $timestep_i$
$Forecasted\ IV_i$ = Forecasted Interaction Volume for $timestep_i$
$timestep_i$ = timestep number over the open hours

Abandons Factor – Actual

The actual number of abandoned interactions during a specific period, based on which Stat Server statistics are configured for Abandoned Interactions.

How the Total / Average is Calculated

A weighted average, weighted by Number of Calls Distributed:

$$\Sigma\ (Abandons_i * CallsDistributed_i) \ /\ \Sigma\ (CallsDistributed_i)$$

Where:

$Abandons_i$ = Number of calls abandoned during $timestep_i$ as recorded by Stat Server
$CallsDistributed_i$ = Number of calls distributed during $timestep_i$, as recorded by Stat Server

Back to Top

Agent Adherence Report Metrics and End Notes

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

In this topic, find information about Workforce Management Agent Adherence Report metrics and Endnotes.

See Other Topics About WFM Metrics

- Schedule Summary Report Metrics
- Contact Center Performance Report Metrics

Agent Adherence Report

There is only one Adherence Report metric.

Percentage Adherence Per Day

The percentage of the day during which the agent was adherent to his or her scheduled state.

How the Percentage is Calculated

```
100 - ((NC+UNC)*100/(ST+UNC))
```

Where:

```
 NC = Noncompliant time
UNC = Noncompliant time outside of scheduled time
 ST = Scheduled time
```

Endnotes

This secton contains detailed descriptions and definitions for formula symbols that are used to calculate WFM metrics.

Endnote 1

The Formula for FTEs:

$$\text{FTEs} = \sum_{i=1}^{n} (staffing_i \div stepsHr \div paidHrs_i \times (100 \div (100 - {}_pOverheadHrs_i)))$$

Where:

```
        n = Number of time steps in a day (96)
        i = Current time step
 staffing = Calculated staffing requirements
  stepsHr = Time steps in one hour (4)
  paidHrs = Paid Hours in a Day as specified in staffing forecast
pOverhead = Planned Overhead percentage as specified in staffing forecast
```

Back to Top

Endnote 2

The `Total/Average` value of Service Level in the Contact Center Performance report is calculated as a weighted average, where for the weight coefficient we use the `TotalNumberCallsDistributed` configuration option.

That is, for an Activity when we define the Quality of Service Statistic type in the WFM Web, we are required to specify three statistics:

1. Service Factor
2. Distributed Interactions
3. Average Speed of Answer

(plus two time ranges required for Service Factor statistic calculation)

The Number of Distributed Interactions statistic (or TotalNumberCallsDistributed) plays the role of the weight coefficient in order to calculate the weighted average value of Service Level per:

Timestep	If we calculate Service Level for the same activity several objects (for example, across some Queues) this gives us the possibility to get an accurate result rather then a simple average.
Day	A weighted average gives a very accurate result in comparison with simple average.

So the calculation of Total Service Level per day is done by the formula:

$$\text{Service Level Total} = \frac{\text{SUM}\ (SL_i \times TNCD_i)}{\text{SUM}\ (TNCD_i)}$$

Where:

i = 1 ... 96 (intra-day 15 minute timesteps from 00:00 through 23:45)
SL_i = Service Level value for the $timestep_i$
$TNCD_i$ = TotalNumberCallsDistributed value for the $timestep_i$

Here is a simple example of the calculation, using this data:

Timestep	SL	TNCD	TNCD*x*SL
10:45 am	60.00	20	1200
11:00 am	90.00	1	90
11:15 am	20.00	150	3000
11:30 am	65.00	35	2275

If we assume that during the day we have only 4 timesteps of historical data, our Total Service Level will be:

$$\text{SL weighted} = \frac{60x20 + 90x1 + 20x150 + 65x35}{20+1+150+35} = \frac{6565}{206} = 31.87$$

Here an example of calculating a simple average:

$$\text{SL simple average} = \frac{60+90+20+65}{4} = 58.7$$

This example shows that a timestep with Service Level=90 where only one call has been processed should have minimal impact on the overall Service Level for the day, in comparison with a timestep where SL=20 and 150 calls have been processed. This is the reason the calculation is done in this way.

The user won't be able to see `TotalNumberCallsDistributed` in the report, as this value is stored in the WFM database for internal purposes only (in the table `WM_perf_activities` in the field `WM_distrib_calls`, for each activity and for each timestep).

Also keep in mind that `TotalNumberCallsDistributed` is not the same as Interaction Volume, since in general Interaction Volume is configured as `TotalNumberCallsEntered` (`Answered + Abandoned`), while `TotalNumberCallsDistributed` is the number of calls which are being distributed from the queue.

Back to Top

WFM Primers

> **Important**
>
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

Workforce Management (WFM) primers provide detailed information about certain WFM functions and features and suggest ways in which you might use them in your deployment to optimize the efficiency of your contact center.

Primers contain suggestions and example, which may or may not be applicable to your environment, but the information is provided to help you to make the best choice, based on these recommended uses and applications.

Primers are provided for these important WFM features:

- Multi-Forecasting
- Time Off
- Overlays

Time Off Primer

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

Use the information in this topic to learn how WFM processes time off and how to configure it to suit the needs of your workforce and contact center.

This topic contains the following sections:

Time-Off Types and Time-Off Rules

You must understand the basics of the time off features in WFM to use them correctly and efficiently. Before you can use time off, you must configure Time-Off types and Time-Off rules.

Time-Off Types

Probably the simpler of the two objects, because each time off type is simply a container. You add time off to, or remove time off from, each container. You can create and arbitrarily name an unlimited number of Time-Off types.

You create Time-Off types for each individual site. You can use time off to track an agent request or maintain a balance of requests on a regular basis. Time off types include vacation, personal time off, flexible time off, paid sick days, floating holidays, and more.

Time-Off Rules

You can configure a Time-Off rule for a one or multiple Time-Off types. When you configure multiple Time-Off types for the same rule, the Time-Off balance is calculated and accrued for all Time-Off types associated with that rule. This configuration enables multiple Time-Off types to share the same Time-Off balance.

Time-off rules define:

- The rate at which time off is accumulated for an agent, and how and when an agent can request time off.
- Whether or not a Time-Off request will be manually approved, or auto-approved by the WFM system.

You assign Time-Off type/Time-Off rule combinations to agents by using the **Configuration > Agents > Time Off** or **Policies > Time-Off Rules > Assignments** pane in WFM Web for Supervisors.

> ## Important
> Time-Off types are configured on a per site basis. Therefore, if an agent has outstanding Time-Off requests and is moved to a different site, the requests are hidden, because WFM tracks them only at the initial site. The default Time-Off type Vacation is the only exception. Vacation is considered valid for all sites in the system, and Time-Off requests for vacation are retained in the system even after the agent moves from site-to-site. In addition, WFM recalculates the agent's Time-Off balance(s) when the agent moves from one site to another, based on the change in Time-Off rules. Moving an agent to a different site does not affect his/her Time-Off status.

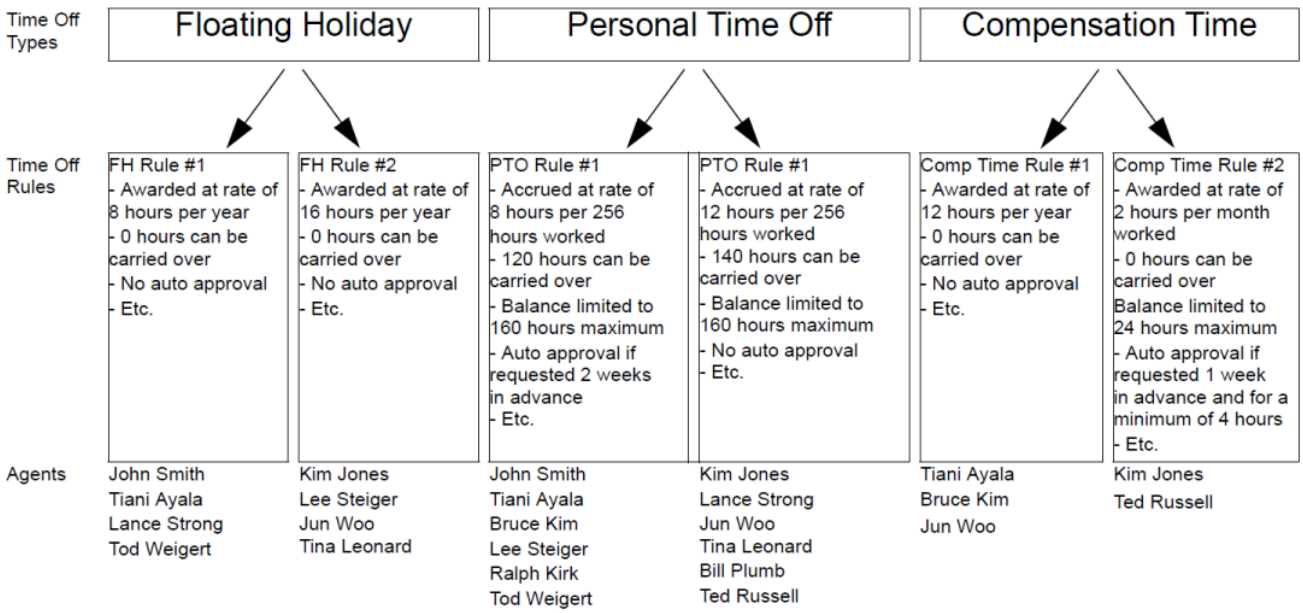


Figure: Relationship Between Time-Off Types, Time-Off Rules, and the Agents Assigned to Them

Notes About the Figure Above

- Agents are assigned to Time-Off rules, not directly to Time-Off types.
- The association of Time-Off rules to time off types is many-to-one, because for a single Time-Off type (for example, personal time off), different agents might be given this type of time off at different rates—perhaps based on seniority. For example, you might want requests from a certain type of agent

for paid sick days to be auto-granted by the system, but for another type of agent you might want the same requests to be manually reviewed.

- When you create a Time-Off rule, you define whether it is a rule of the type award (a fixed number of hours) or accrual (a number of hours that accumulates during the year). In the figure below, all Floating Holiday time offs are awarded and all Personal Time Offs are accrued. Compensation Time can be either awarded or accrued, depending on an Agent's assigned Time-Off rule.

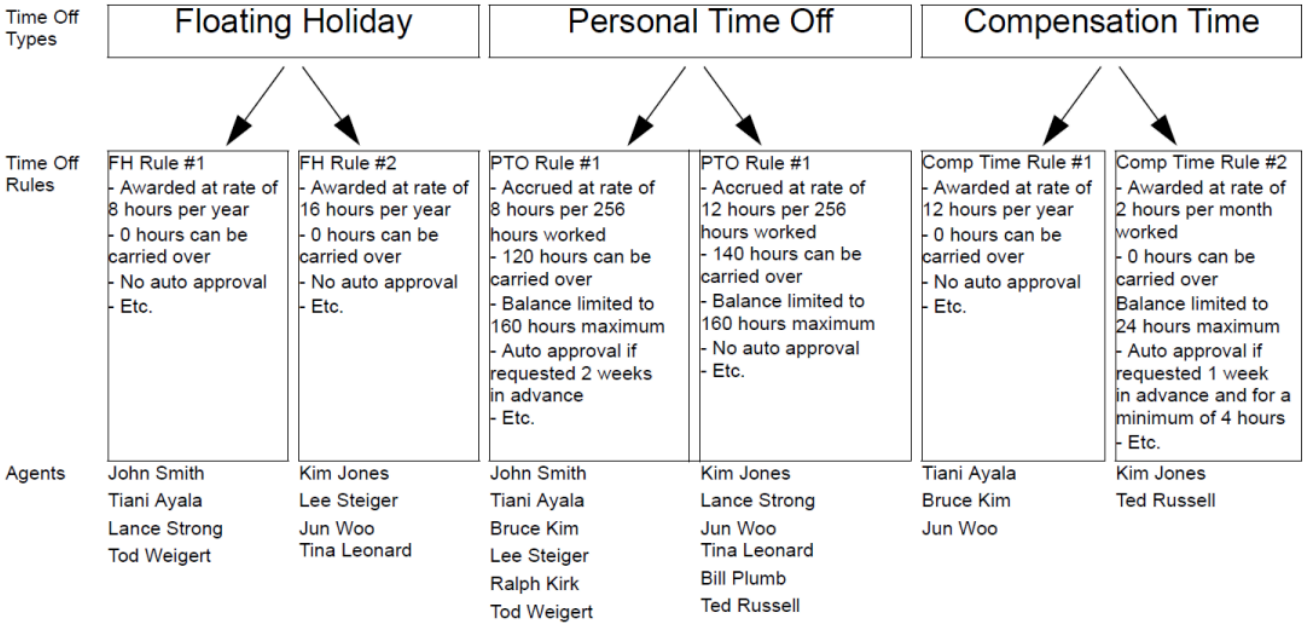


Figure: Awarded versus Accrued Time Off

Back to Top

Notes About the Figure Above

- An agent can be assigned to mulitiple Time-Off rules.
  Example 1: Jun Woo has three Time-Off rules assigned to him. One rule defines how he will be awarded a Floating Holiday each year and when he can use that time off. The second rule defines the rate at which he will accrue Personal Time Off, how much balance he can have, and when he can use it. The third rule defines how he will be awarded Compensation Time each year.
- Not all agents need to be assigned to a rule for each Time-Off type.
  Example 2: John Smith has a Floating Holiday award rule and a Personal Time-Off accrual rule. However, he has no Time-Off rule defining how he can accrue or be awarded Compensation Time. Therefore, he will have no balance of this time off type and will not be able to request it.

Exception Used as Time Off

When you configure an exception type in WFM Web, you can use the option **Exception is used as Time Off** to designate the exception as time off. This is a legacy feature from earlier versions of WFM, which supported only one type of time off. WFM now supports an unlimited number of Time-Off types, which means Agents and Supervisors can request both full-day and partial-day time off. Therefore, Genesys recommends that you *do not* use exceptions to represent time off.

When Time-Off Types No Longer Apply

A combo box in the upper-left corner of an agent's **Time-Off Planner** displays all the Time-Off types that are configured for that agent's site. Some of these might not relevant for the agent. (Agents are assigned to Time-Off rules, which in turn are associated with Time-Off types.)

An agent can create, edit, delete, or recall time off requests only for those types correspond to a **Time-Off Rule** that is assigned to that agent. Therefore, an agent might be able to view **Time-Off Types** corresponding to **Time-Off Rules** are no longer assigned to him or her. The agent can see these "old" Time-Off types, but he or she cannot interact with them. They appear below the **Others** legend in the combo box.

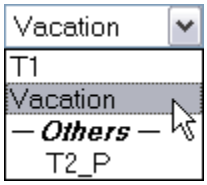


Figure: Time-Off Types Combo Box

Back to Top

Time-Off Types that the agent can use appear above the **Others** legend.

WFM enables an Agent to select a **Time-Off Type** from below the **Others** legend in the combo box; however, the agent cannot perform any functions with it (such as requesting time off or viewing his or her time off balance).

Time-Off Limits

Before giving your Agents the ability to request time off, you should configure **Time-Off Limits**. To do so, use the **Calendar** module in WFM Web Supervisor.

If you have the correct permissions, you can configure the maximum number of Agents who can take time off for each 15-minute interval of each calendar day.

For example, you could configure that from midnight to 8 a.m. on March 3, 2007, 5 Agents can be allowed to take time off, whereas from 8 a.m. to 4 p.m., only 3 Agents can be take time off. You can set Time-Off limits either for an absolute number of Agents, or for a maximum percentage of Agents.

Additionally, you can set Time-Off limits at three levels: Site, Team, and Activity.

> ### Important
> You cannot set different Time-Off limits for different Time-Off types. A single set of Time-Off limits governs the total amount of time off of all types that can be granted for a specified time of day, for a specified date.

Time-Off Request and Approval Process

There are two different processes for scheduling time off for agents. The process that is used depends on whether the user is planning for a future schedule period (see Planning for Time Off in the Future) or working within a schedule period that is already published to the Master Schedule (see Planning for Time Off for the Current Schedule Period).

Planning for Time Off in the Future

For future time periods (schedule days that are not yet published to the Master Schedule), the process works as shown in the below.

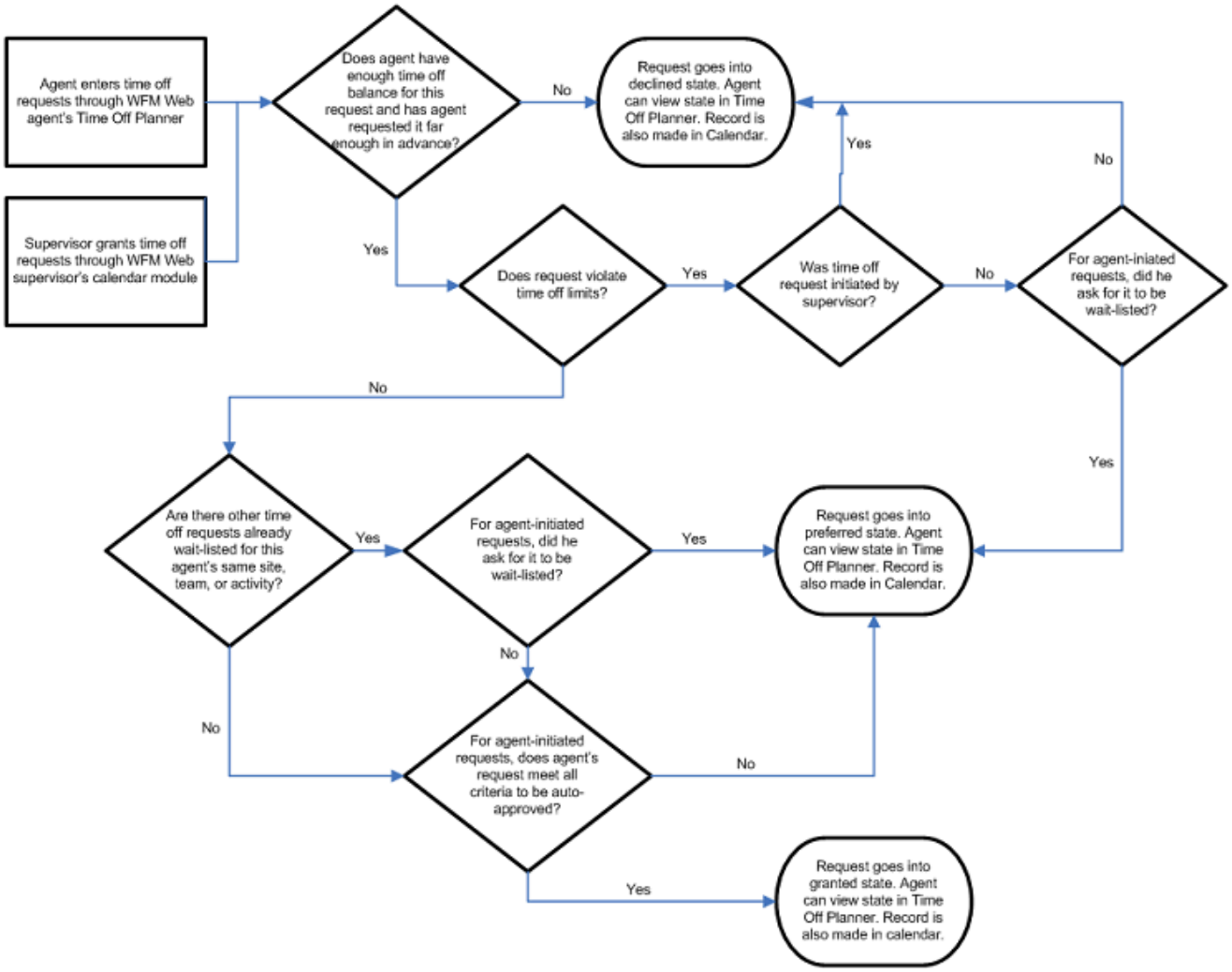


Figure: Future Time Periods Process

Back to Top

The Time-Off Planner (in the WFM Web Agent application) and the **Calendar** (in the 'WFM Web Supervisor application) are really two different ways to input the same information. The **Time-Off**

Planner is the Agent's method of entering Calendar items. It allows the Agent to enter time off, but that is all.

The Supervisor's **Calendar** module has much more power: the Supervisor can enter all types of **Calendar** items, including shifts, working hours, exceptions, days off, and time off.

Both of these schedule-building input tools have the same result: the Time-Off items are recorded in the **WFM Calendar**.

Granted Time Off versus Preferred Time Off

WFM considers only Time-Off items with the status granted when building a schedule scenario. It does not include Time-Off items with the status **preferred**.

> **Important**
>
> WFM enables a Supervisor to consider Agent preferences when building schedules. These preferences include: day off, availability, and shift preferences; but not Time-Off preferences.

Time Off Items in the Calendar Hierarchy

When you enter multiple types of Calendar items for the same Agent on the same day, WFM uses its internal hierarchy to resolve their status.

- **Example 1: Full-Day Time Off versus Rotating Schedule Day In**—An Agent is assigned a Rotating Pattern for the week starting September 17, 2006, and for the date of September 19 his Rotating Pattern assignment states he should be on a *Day In*. But a supervisor grants a full-day time off for September 19. Now, the agent has two conflicting Calendar items for the same day: a working day according to the Rotating Pattern assignment, and a full-day time off according to the Calendar. The Calendar hierarchy specifies that the Full-Day Time-Off item should be granted and this should cause the Rotating Pattern assignment for that day should be declined. The statuses are reflected in the Calendar and the supervisor can read the reasons there. Some items in the Calendar do have a higher priority that Full-Day Time-Off items.
- **Example 2: Full-Day Exception versus Full-Day Time Off**—A full-day exception is granted for an agent in the Calendar for the same day as a full-day time off. The Calendar hierarchy specifies that the full-day exception should take priority and the Full-Day Time-Off item should be declined. The hierarchy of the WFM Calendar is described in the *Workforce Management Web for Supervisors Help* in the topic `Calendar Items Overview`.

Back to Top

Using the Calendar to Understand the Status of Time-Off Items

You can view the *Actual Status* of any item in the Calendar; a status of **Granted**, **Preferred**, or **Declined** confirms that no schedule has been published for this Agent for this date. In a built and published schedule, the *Actual Status* of an item will be either **Scheduled** or **Not Scheduled**. If a Calendar item was not scheduled, you can view a Reason field which will describe why it was not scheduled (in the **Reason** column).

Planning for Time Off for the Current Schedule Period

To enter agent Time-Off items for days that are already published to the Master Schedule, use either of these methods:

1. Enter time off directly into the **Master Schedule**.

> **Important**
>
> The agent can view this Time-Off entry in his schedule by logging in to the **WFM Web for Agents Application**. He/she will also see a change to his time-off balance in the **WFM Web for Agents' Time Off Planner** and he/she can see that time off for this day is scheduled, when he views his **Time-Off Planner**. However, no entry for this time off will be made in the **WFM Calendar**, because the **Calendar** is a planning tool, and is meant for entering items such as exceptions, time off, days off, etc., that are to be considered when building a schedule scenario.

2. Enter the granted Time-Off request into the **WFM Calendar**.
3. Click **Update Schedule**.

WFM has now automatically updated the **Master Schedule** with the agent's **Granted Time Off**.

Using **Update Schedule** from within the **Calendar** updates the **Master Schedule** from the **Calendar** for the agent's Time-Off request, which streamlines the review and approval process and eliminates the need to manually rebuild the schedule.

> **Important**
>
> From within the **Calendar** module, users can update the **Master Schedule** with a granted **Time-Off** items for any agent on any date for which the agent has a schedule published, and the **Time-Off** request is compatible with the agent's existing schedule. For more details, see the *WFM Web for Supervisors Help* topics **Calendar Items Overview** and **Update Schedule Options**.

Agent Time-Off Planner

Agents request time off and see the status of these requests through **WFM Web for Agents Time-Off Planner**. If you do not want your agents to have access to the **Time-Off Planner**, you can disable this through the **Time-Off Planner Enabled** setting. In Configuration Manager, for your **WFM Web Application** object, open **Options > AgentTimeOff**, and set the variable **AllowTimeOffPlanner** to `false`.

Time-Off Balance in Agent’s Time-Off Planner

The Time-Off Planner displays the balances for an agent's time off, in this way: the agent selects a Time-Off type and clicks on a date in the yearly planning calendar to view balances for that type, up

to that date.

The agent can also view the types of Time-Off hours prior to the selected date: already granted, preferred, declined and scheduled, bonus, advance, and carried-over.

When an agent's Time-Off rule is changed, his time off is recalculated, based on the new rules.

See some examples.. [+]

- If a Time-Off rule is assigned to an agent and you change any of the rule's properties.
- If a Time-Off rule is assigned to an agent and you use the `Time Off` tab under `Configuration` > `Organization` > `Agent properties` to change the rule.
- If a Time-Off rule is assigned to an agent and you use the `Assignments` tab under `Policies` > `Time-Off Rules` > `Properties` to remove the assignment.

In each case, the user is asked to approve the calculation before it is performed.

> ### Warning
> Moving to a different site can affect an agent's Time-Off balance. An agent who moves to a different site might appear to lose a portion of her existing Time-Off balance, because WFM tracks that time off only at the old site. To preserve the Time-Off hours that the agent earned at the previous site, her supervisor must enter these hours as a bonus to the new `Agent Rule` under the new site. For details, see the *WFM Web for Supervisors Help* topic `Agent Time Off` > `Bonus Time Off Hours`.

Back to Top

Recalling Time Off Requests

An agent can recall a Time-Off request that is in preferred or granted status, as long as the Time-Off item has not been scheduled. Use the **WFM Web for Agents Time-Off Planner**; see the topic "Requesting Time Off".

After an agent's Time-Off request has been granted, scheduled, and published to the Master Schedule, the agent cannot recall the time off.

Only the supervisor can do that, by manually changing the agent's schedule for that date, through the Master Schedule. And only the supervisor can schedule a different activity for the agent for that date, in lieu of the time off.

Wait Listed Time Off Requests

When an agent makes a Time-Off request, he can optionally ask for it to be wait-listed. (See step 7 in the *Workforce Management Web for Agents Help* topic "Requesting Time Off": *If you want your request to be wait-listed, click the Wait-list check box*.)

> **Important**
>
> Wait-listing means that if a Time-Off request is denied because the Time-Off limits have been reached, the request stays in a preferred status in case an opening becomes available. The request could eventually be granted by a supervisor, although this is not guaranteed.

In detail, this means that if the Time-Off request would be declined because the Time-Off limits have already been reached, wait listing gives it a preferred status. If some Time-Off slots become available for certain dates due to cancellations, or if the Supervisor decides to raise the Time-Off limits for a certain date, the Supervisor user can review all of these preferred Time-Off items in the Calendar and grant some of them. There is a field in the Calendar which displays the date and time that the Agent or Supervisor submitted each Time-Off item, to help the supervisor manage the wait list. Agents' Time-Off requests are not pulled automatically from this wait list; these requests must be manually granted by the supervisor.

Viewing the Status of Time Off Requests

Agents can view the status of any Time-Off request, for any calendar date, with some limitations. If an agent requests time off for a date on a schedule that is already been published to the Master Schedule:

- This request will not be scheduled.
- The Agent can view the reason it was not scheduled, in the Reason column.
- The request, with the same status and reason, will appear in the WFM Calendar, where Supervisors can see it.

However, the supervisor will not know to look for the request unless the agent tells him to look for it. Therefore, if an agent sees Time-Off requests with status **Not Scheduled** in the **Time-Off Planner**, she should alert her supervisor.

The supervisor can then do one of two things:

1. Grant the time off through the WFM Calendar and rebuild/republish the agent's schedule for the affected date(s).
2. Manually add the time off to the Master Schedule.

The Time-Off Planner gives agents a limited view into their Time-Off status in the WFM Calendar. Therefore, when a supervisor adds or deletes time off in the WFM Calendar, he must rebuild and republish the schedule for the affected date(s).

Back to Top

Overlays Primer

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

Workforce Management Overlays are events designed to indicate the abnormalities in historical data or anticipated ones in the future. Abnormalities, such as fluctuation of interaction volumes, AHT (other than the usual, seasonal fluctuation), or intra-week and intra-day trends. If events occur that highlight an abnormality, of the type that happens multiple times or can happen in again in the future, you can arrange these overlay events (or *Overlays*) into overlay groups. Some examples are billing, promotion, or catalog drops.

This Overlays topic includes the following sections:

Impact of Overlay Events on Prediction Data

Overlays impact prediction data directly when a specific overlay event occurs over a prediction interval.

Two types of overlays exist, based on how the impact on prediction data is calculated:

- **Multiplicative**—Increase (or decrease) every step of predicted data by a percentage. The percentage is defined by overlay impact distribution, multiplied by the *strength* of the event. The duration of the interval is affected by event changes.
- **Overriding**—Re-distribute (but dos not change) the volume of an event's interval. The total volume does not change, but it might be moved from one event-step to another.

Multiplicative Overlays

This type of overlay factor has existed in WFM since early versions and was called Factor. Each step (daily or hourly) of prediction data covered by the event is adjusted by a certain percentage, which is also multiplied by event strength.

Overriding Overlays

This type of overlay is designed to keep the predicted total of the affected period and adjust the distribution of the volumes within that period. WFM applies the events of this overlay type as the last step of the prediction. It applies the seasonal components (intra-day, daily, or yearly) and multiplicative overlays before the overriding overlays are calculated.

The overriding overlay distributes the volumes according to the weight (or the percentage) of each event-step (daily or hourly). It adjusts the volume of each event-step, so the event-step receives its portion of the total of the entire event period, according to its weight.

For example, an overriding overlay has 3 event-steps with the following weights: 20, 30, 50. If the predicted total of the entire interval is 1000, then the first event step receives 200, the second 300, and the last one 500.

> **Important**
>
> The initial predicted total of the event-step is not considered when the impact of this overlay type is calculated. The weight of the event-step is determined by the overlay itself, and the predicted total of the entire interval that is affected by the event, is considered.

When the event-step (daily or hourly) is calculated, its total is distributed proportionally to 15-minute timesteps to a volume of each timestep before the event was applied. So, the intra-day or intra-hour pattern is preserved.

The overriding overlay event type cannot be calculated when it overlaps with other events of the same type, even if both events belong to different overlays. However, it can overlap with events of multiplicative overlays.

Event Impact Distribution

There are three ways to determine the impact of each event-step:

1. By using start-end.
2. By keeping the entire detailed distribution.
3. By performing the calculation for each event-step during the prediction.

These three methods of determining impact distribution are applicable to both multiplicative and overriding overlay type. WFM specifies how each overlay's impact distribution will be determined and applies it to all of the overlay's events.

Start-end

WFM determines the overlay's impact distribution by specifying start and end impact values. It gradually changes the impact by the same amount every event-step from start value to end value. For example, if the start value is 100 and the end value is 200 and it is a daily overlay with a length of 6 days, then the impact on the first day is 100, on the second day it is 120, then 140, 160, 180 and 200 on the last day.

Distribution

The overlay saves the impact of each event-step separately. You can either determine the impact by precalculating it, by entering it manually, or by using a combination of both methods.

Always Calculating

The overlay impact is always calculated during the prediction. To ensure the calculation is successful, the historical period must have one or more events for the same overlay. The impact of the overlay is determined by the prediction algorithm according to the historical data, and it is then used in the prediction.

Impact of Overlay Events on Historical Data

Events within any overlay type can have a *ignore historical data* flag. If this flag is set for an event, the historical data interval data covered by the event is not used in the calculation of either volume prediction or impact of overlays (see Calculating an Overlay's Impact).

If an event does not have the ignore historical data flag set, the the data covered by the event is considered for prediction.

> **Important**
>
> In 7.6.1, WFM does not do perform any additional processing of historical data that is affected by the event. WFM simply uses it or ignores it.

Calculating an Overlay's Impact

You can determine the impact of overlay by analyzing the historical data and using a prediction algorithm. The algorithm analyzes the period of historical data, which contains one or more overlay events to be calculated.

You can pre-calculated Overlays before starting volume forecasting or during volume forecasting (see, Event Impact Distribution). If the historical data and method used are the same, the results will be identical.

Impact of Multiplicative Overlays

WFM calculates Multiplicative overlays by separating seasonal component (yearly, daily, intra-day) from the event impact for each event of the overlay in the given historical data. The impact is divided by the *strength* of the event and then averaged.

When the impact of an event is applied to the prediction interval, it is multiplied by the *strength* of that event.

Impact of Overriding Overlays

The percentage of each event-step in the total of the whole event period is calculated for each event and then averaged. For example, a historical period has two daily overlay events that are 3 days long. The 3 days of the first event are: 150, 200, 150 (that is 30%. 40%, 30% of the total respectively) and the 3 days of the second event are: 150, 150, 200. In this example, the overlay is calculated as 30%, 35%, 35%.

Multi-Forecasting Primer

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

This topic provides information about the Genesys approach to multi-skill forecasting, where agents with multiple skills can increase the center's efficiency by performing multiple tasks within a single timestep.

Multi-Skill Forecasting

A multi-skilled contact center presents an opportunity for increased productivity.

An agent might be idle in a single-skill environment because she cannot answer calls that are queuing for an activity/skill which she may posses—but a skill that the schedule prevents her from using.

In a multi-skilled environment, she can use her additional skills to answer those calls.

> **Important**
> A *high-load environment* does not present much opportunity for increased efficiency, because the agents have very little idle time. However, in an *overstaffed environment*, agents have more idle time and can use their multiple skills to increase efficiency.

How WFM Supports Multi-Skilled Agents

A multi-skilled agent is qualified to work on multiple activities, and therefore may perform different types of work during a shift.

In a multi-skill environment, an agent might be available for multiple activities during any timestep. That agent can be scheduled to work for an activity for only part of a time interval, and only the fraction of the time period during which she or he works is counted.

Because of this, the value for staffing can be expressed as a fraction. Consider the following example

of a 15-minute timestep:

Example: An agent is scheduled to work on Activity A for 10 minutes and for 5 minutes on Activity B. She is counted as 2/3 (or .667) of an agent for Activity A, and as 1/3 (or .333) of an agent for Activity B.

Enabling Multi-Skill Support

Display procedure..[+]

To enable multi-skill support, follow these steps:

1. Open Genesys Administrator.
2. Open the **WFM Server Application**.
3. From the **Options** tab, open the section **ScheduleService**.
4. Create a new option named `SplitMS` and set the value to `true`. (See figure below.)

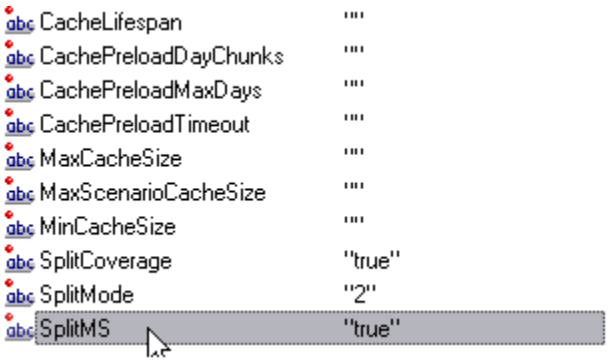


Figure: Enable Multi-Skill Support

Calculating Multi-Skill Equivalents

Consider the comparison of Single Skill Equivalents (SSE) to Multi-Skill Equivalents (MSE) in the figure below.

The Multi-Skilled forecasting algorithm takes into account how many agents (with their various skill sets) could be available to work on each Activity, as well as how the occupancy of an average agent would be divided among this Activity and the other Activities on which the agent could work.

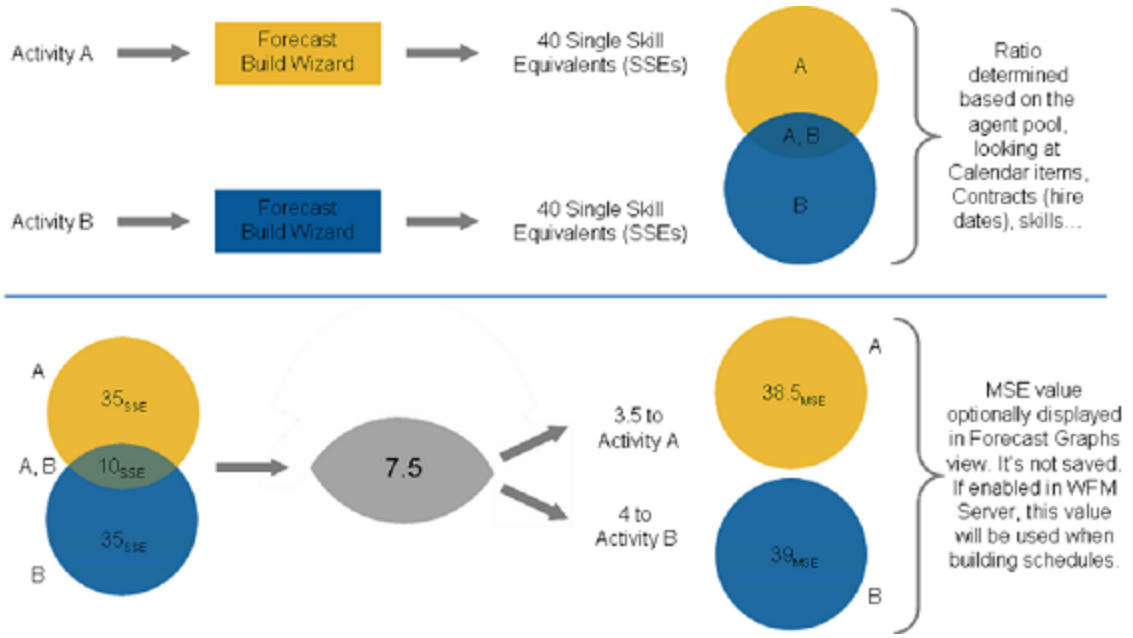


Figure: Comparing Multi-Skill and Single-Skill Equivalents

When building a schedule, WFM can optionally use the staffing forecast in Multi-Skilled Equivalents (MSEs) while taking into account agents that the schedule is being built for, as well as agents for whom schedules have already been built.

If the MSE option is set, in the **Performance** views/reports and **Schedule** views/reports, coverage for an Activity is based on MSEs, calculated from actual agent schedules. See the figure below, which depicts multi-skill gains.

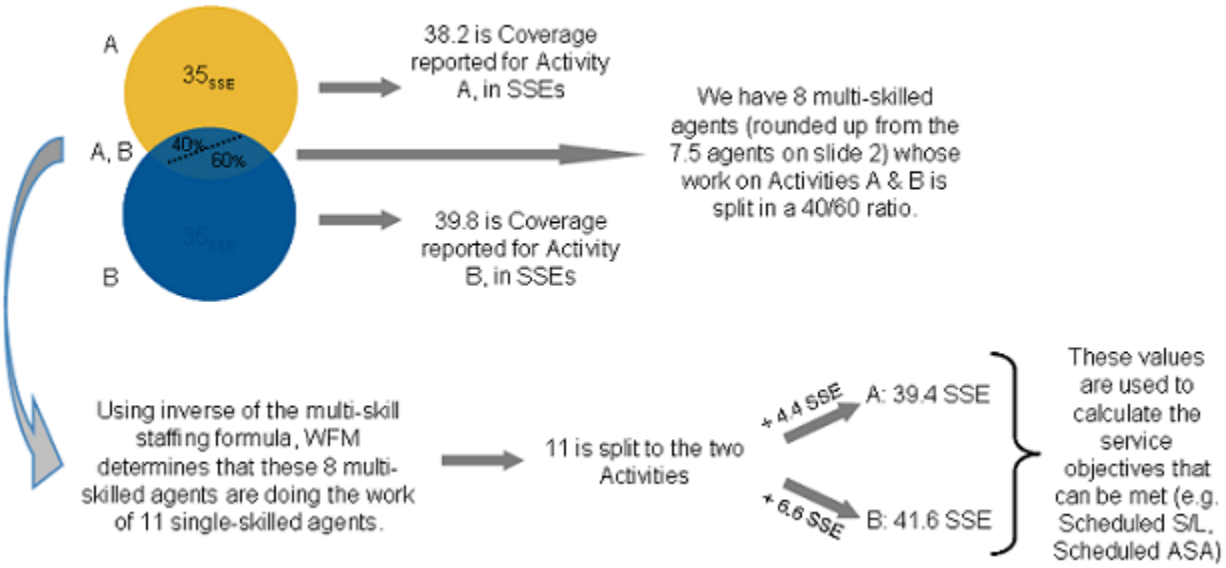


Figure: Multi-Skill Gains

Workforce Management ETL Database Reference

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

The Workforce Management (WFM) Extract, Transform, and Load (ETL) database schema enables Genesys CX Insights and other third-party reporting applications to easily create reports that incorporate Genesys WFM data. Once configured, this functionality can obtain Schedule, Adherence, and Performance information from WFM and store it into a documented relational database schema.

The ETL schema can co-exist with the main operational WFM database, be a standalone database, or part of any other database. WFM provides the SQL script to create the database schema, but does not specify which physical tablespace, user, or database on which to create it. The script is included in WFM Database Utility (DBU) IP, but is not executed automatically by the DBU.

Related Topics

Intended audience

This reference guide is intended for:

- Reporting and business analysts who want to leverage the data that is contained in Genesys WFM, Data Mart, Info Mart and other third party applications to produce reports for business users.
- IT administrators who want to gain an understanding of the components that enable WFM.

This reference assumes that the reader understands of the following:

- Relational database concepts.
- Structured Query Language (SQL) for querying and mining data.
- Genesys WFM configuration and its data sources.
- Data warehouse concepts—including working with star schemas, dimensions, aggregates, and measures.
- Extraction, Transformation, and Loading (ETL) concepts

Overview

The Workforce Management (WFM) ETL Database schema contains Dimension, Fact, Service and Control, and Referred Info Mart tables. Dimension tables correspond to the WFM Organization, Configuration, and Policy objects. The Dimension tables provide sorting, grouping, and filtering capabilities for reports. The Fact tables contain Adherence, Performance, and Schedule information and can be sorted, grouped, and filtered by dimensions.

This reference contains detailed descriptions of these tables (summarized in this topic), and examples of queries that can be run on the table data.

Dimension tables

There are three types of Dimension tables.

- **General Dimension tables include:**
  - `WFM_BU`—Business unit descriptive information.
  - `WFM_SITE`—Site descriptive information.
  - `WFM_TEAM`—Team descriptive information.
  - `WFM_AGENT`—Agent descriptive information.
  - `WFM_ACTIVITY_TYPE`—Activity types.
  - `WFM_ACTIVITY`—Activity descriptive information.
- **Schedule Dimension tables include:**
  - `WFM_SSG_TYPE`—Schedule State Group types.
  - `WFM_SSG`—Schedule State Group descriptive information.
  - `WFM_STATE_TYPE`—Schedule state types.
  - `WFM_STATE`—Schedule state descriptive information.
- **Performance Dimension tables include:**
  - `WFM_PERF_ITEM`—Performance statistics.

See all Dimension tables in detail.

Fact tables

The Fact tables provide the following data:

- Agent/team/site adherence totals aggregates for the calendar day and for the 15-minute interval.
- Agent/team/site schedule totals aggregates for the schedule day and for the 15-minute interval.

- Agent schedule states.
- Schedule state and Schedule State Group (SSG) duration aggregates for the 15-minute interval.
- Numerous activity (single-site, multi-site, Activity Group) and site performance statistics aggregates for the calendar day and for the 15-minute interval.

There are three types of Fact tables.

- **Adherence Fact tables inlcude:**
  - WFM_ADH_AGENT_DAY—Aggregate of the agent adherence information for 24-hour days.
  - WFM_ADH_AGENT_TIMESTEP—Aggregate of the agent adherence for 15-minute intervals.
- **Schedule Fact tables include:**
  - WFM_SCH_AGENT_DAY—Agent schedule day information.
  - WFM_SCH_AGENT_TIMESTEP—Aggregate of agent's schedule totals for 15-minute intervals.
  - WFM_SCH_AGENT_STATE—Agent Schedule state information.
  - WFM_SCH_AGENT_STATE_TIMESTEP—Aggregate of schedule state duration for 15-minute intervals.
- **Performance Fact tables include:**
  - WFM_PERF_ITEM_DAY—Performance statistics in 24-hour day granularity
  - WFM_PERF_ITEM_TIMESTEP—Performance statistics in 15-minute granularity

See all Fact tables in detail.

Service and Control table

There is one Service and Control table:

- WM_DB_VERSION—Internal version table.

Referred Genesys Info Mart tables

There are six referred Info Mart tables:

- CTL_AUDIT_LOG
- DATE_TIME
- TIME_ZONE
- GIDB_GC_TENANT
- GIDB_GC_SWITCH
- GIDB_GC_AGENT

Query examples

There are three categories of ETL query examples:

- Adherence queries
- Schedule queries
- Performance statistics queries

See all query examples in detail.

Abbreviations of database terms

The Workforce Management (WFM) ETL Database Reference uses abbreviations throughout all topics to provide detailed information about and within the tables, including a concise listing of primary and foreign keys, default field values, and mandatory fields for each table. The field and index abbreviations are described here:

Field characterizations

- **P**—Primary key
- **M**—Mandatory field
- **F**—Foreign key
- **DV**—Default value

Index characterizations

- **C**—Cluster
- **U**—Unique

Dimension tables

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

This topic describes the Dimension tables in the Workforce Management (WFM) ETL Database schema.

Related Topics

For a description of the abbreviations used in these tables, see Abbreviations for ETL Database Terms.

WFM_BU

This table contains business unit descriptive information.

Column	Data type	P	M	F	DV
WFM_BU_KEY	int	X	X		
WFM_BU_NAME	varchar(255)		X		
WFM_TIMESTAMP	numeric(19)		X		
TIME_ZONE_KEY	int			X	
ACTIVE_FLAG	int		X		
TENANT_KEY	int			X	
CREATE_AUDIT_KEY	numeric(19)			X	
UPDATE_AUDIT_KEY	numeric(19)			X	
PURGE_FLAG	int				

Description of Columns

- **WFM_BU_KEY**—The primary key for this table.
- **WFM_BU_NAME**—The name of Business Unit (BU).
- **WFM_TIMESTAMP**—An internal timestamp value.

- **TIME_ZONE_KEY**—The surrogate key used to join the TIME_ZONE dimension to the fact tables. It specifies the time zone of the Business Unit.
- **ACTIVE_FLAG**—Indicates whether the Business Unit is currently active: 0 = No, 1 = Yes.
- **TENANT_KEY**—The surrogate key used to join the TENANT dimension to the fact tables.
- **CREATE_AUDIT_KEY**—The surrogate key used to join to the CTL_AUDIT_LOG control table. The key specifies the lineage for data creation. This value is useful for aggregation, Enterprise Application Integration (EAI), and ETL tools (that is, applications that need to identify newly added data).
- **UPDATE_AUDIT_KEY**—The surrogate key used to join to the CTL_AUDIT_LOG control table. The key specifies the lineage for data update. This value is useful for aggregation, Enterprise Application Integration (EAI), and ETL tools (that is, applications that need to identify recently modified data).
- **PURGE_FLAG**—This field is reserved.

WFM_SITE

This table contains site descriptive information.

Column	Data type	P	M	F	DV
WFM_SITE_KEY	int		X		
WFM_BU_KEY	int		X	X	
WFM_SITE_NAME	varchar(255)		X		
WFM_TIMESTAMP	numeric(19)		X		
SWITCH_KEY	int			X	
TIME_ZONE_KEY	int			X	
ACTIVE_FLAG	int		X		
TENANT_KEY	int			X	
CREATE_AUDIT_KEY	numeric(19)			X	
UPDATE_AUDIT_KEY	numeric(19)			X	
PURGE_FLAG	int				

Description of Columns

- **WFM_SITE_KEY**—The primary key for this table.
- **WFM_BU_KEY**—The surrogate key used to join the WFM_BU dimension to the fact tables. It specifies the Business Unit of the Site.
- **WFM_SITE_NAME**—The name of the Site.
- **WFM_TIMESTAMP**—An internal timestamp value.
- **SWITCH_KEY**—The surrogate key used to join the GIDB_GC_SWITCH dimension to the fact tables. It specifies the switch associated with the Site.
- **TIME_ZONE_KEY**—The surrogate key used to join the TIME_ZONE dimension to the fact tables. It specifies the time zone of the Site.

- **ACTIVE_FLAG**—Indicates whether the Site is currently active: 0 = No, 1 = Yes.
- **TENANT_KEY**—The surrogate key used to join the TENANT dimension to the fact tables.
- **CREATE_AUDIT_KEY**—The surrogate key used to join to the CTL_AUDIT_LOG control table. The key specifies the lineage for data creation. This value is useful for aggregation, Enterprise Application Integration (EAI), and ETL tools (that is, applications that need to identify newly added data).
- **UPDATE_AUDIT_KEY**—The surrogate key used to join to the CTL_AUDIT_LOG control table. The key specifies the lineage for data update. This value is useful for aggregation, Enterprise Application Integration (EAI), and ETL tools (that is, applications that need to identify recently modified data).
- **PURGE_FLAG**—This field is reserved.

WFM_TEAM

This table contains team descriptive information.

Column	Data type	P	M	F	DV
WFM_TEAM_KEY	int	X	X		
WFM_TEAM_NAME	varchar(255)		X		
WFM_SITE_KEY	int		X	X	
WFM_TIMESTAMP	numeric(19)		X		
ACTIVE_FLAG	int		X		
TENANT_KEY	int			X	
CREATE_AUDIT_KEY	numeric(19)			X	
UPDATE_AUDIT_KEY	numeric(19)			X	
PURGE_FLAG	int				

Description of Columns

- **WFM_TEAM_KEY**—The primary key for this table.
- **WFM_TEAM_NAME**—The name of the Team.
- **WFM_SITE_KEY**—The surrogate key used to join the WFM_SITE dimension to the fact tables. It specifies the Site of the Team.
- **WFM_TIMESTAMP**—An internal timestamp value.
- **ACTIVE_FLAG**—Indicates whether the Team is currently active: 0 = No, 1 = Yes.
- **TENANT_KEY**—The surrogate key used to join the TENANT dimension to the fact tables.
- **CREATE_AUDIT_KEY**—The surrogate key used to join to the CTL_AUDIT_LOG control table. The key specifies the lineage for data creation. This value is useful for aggregation, Enterprise Application Integration (EAI), and ETL tools (that is, applications that need to identify newly added data).
- **UPDATE_AUDIT_KEY**—The surrogate key used to join to the CTL_AUDIT_LOG control table. The key specifies the lineage for data update. This value is useful for aggregation, Enterprise Application Integration (EAI), and ETL tools (that is, applications that need to identify recently modified data).

- **PURGE_FLAG**—This field is reserved.

WFM_AGENT

This table contains agent descriptive information.

Column	Data type	P	M	F	DV
WFM_AGENT_KEY	int	X	X		
WFM_BU_KEY	int			X	
WFM_SITE_KEY	int			X	
WFM_TEAM_KEY	int			X	
EMPLOYEE_ID	varchar(64)		X		
FIRST_NAME	varchar(64)		X		
LAST_NAME	varchar(64)		X		
HIRE_DATE	date		X		
TERMINATION_DATE	date				
WFM_TIMESTAMP	numeric(19)		X		
AGENT_KEY	int			X	
ACTIVE_FLAG	int		X		
TENANT_KEY	int			X	
CREATE_AUDIT_KEY	numeric(19)			X	
UPDATE_AUDIT_KEY	numeric(19)			X	
PURGE_FLAG	int				

Description of Columns

- **WFM_AGENT_KEY**—The primary key for this table.
- **WFM_BU_KEY**—The surrogate key used to join the WFM_BU dimension to the fact tables. It specifies the the Agent's business unit.
- **WFM_SITE_KEY**—The surrogate key used to join the WFM_SITE dimension to the fact tables. It specifies the Agent's site.
- **WFM_TEAM_KEY**—The surrogate key used to join the WFM_TEAM dimension to the fact tables. It specifies the Agent's team. It is NULL if Agent does not belong to any team.
- **EMPLOYEE_ID**—The Agent's employee ID.
- **FIRST_NAME**—The Agent's first name.
- **LAST_NAME**—The Agent's last name.
- **HIRE_DATE**—The Agent's hire date.
- **TERMINATION_DATE**—The Agent's termination date.

- **WFM_TIMESTAMP**—An internal timestamp value.
- **AGENT_KEY**—The surrogate key used to join the GIDB_GC_AGENT dimension to the fact tables.
- **ACTIVE_FLAG**—Indicates whether the agent is currently active: 0 = No, 1 = Yes.
- **TENANT_KEY**—The surrogate key used to join the TENANT dimension to the fact tables.
- **CREATE_AUDIT_KEY**—The surrogate key used to join to the CTL_AUDIT_LOG control table. The key specifies the lineage for data creation. This value is useful for aggregation, Enterprise Application Integration (EAI), and ETL tools (that is, applications that need to identify newly added data).
- **UPDATE_AUDIT_KEY**—The surrogate key used to join to the CTL_AUDIT_LOG control table. The key specifies the lineage for data update. This value is useful for aggregation, Enterprise Application Integration (EAI), and ETL tools (that is, applications that need to identify recently modified data).
- **PURGE_FLAG**—This field is reserved.

WFM_ACTIVITY_TYPE

This table contains activity types and descriptions.

Column	Data type	P	M	F	DV
WFM_ACTIVITY_TYPE_KEY	int	X	X		
WFM_ACTIVITY_TYPE_NAME	varchar(64)		X		

Description of Columns

- **WFM_ACTIVITY_TYPE_KEY**—The Activity type ID.
- **WFM_ACTIVITY_TYPE_NAME**—The Activity type name. The table below contains valid values.

ID	Name
0	'Immediate'
2	'Fixed Staffing'
4	'Deferred'
10	'Activity Group'

WFM_ACTIVITY

This table contains activity descriptive information.

Column	Data type	P	M	F	DV
WFM_ACTIVITY_KEY	int	X	X		
WFM_BU_KEY	int		X	X	
WFM_SITE_KEY	int			X	

Column	Data type	P	M	F	DV
WFM_MSA_KEY	int			X	
WFM_ACTIVITY_NAME	varchar(255)		X		
WFM_ACTIVITY_SHORT_NAME	varchar(5)		X		
WFM_ACTIVITY_TYPE_KEY	int		X	X	
WFM_TIMESTAMP	numeric(19)		X		
ACTIVE_FLAG	int		X		
TENANT_KEY	int			X	
CREATE_AUDIT_KEY	numeric(19)			X	
UPDATE_AUDIT_KEY	numeric(19)			X	
PURGE_FLAG	int				

Description of Columns

- **WFM_ACTIVITY_KEY**—The primary key for this table.
- **WFM_BU_KEY**—The surrogate key used to join the WFM_BU dimension to the fact tables. It specifies the Business Unit of the Activity.
- **WFM_SITE_KEY**—The surrogate key used to join the WFM_SITE dimension to the fact tables. It specifies the Site of the Activity. It is NULL if Activity is Multi-Site Activity (MSA) or Activity Group (AG).
- **WFM_MSA_KEY**—The surrogate key used to join the parent Multi-Site Activity to the child Activity. It is NULL for Multi-Site Activity and Activity Group or if Activity does not belong to any Multi-Site Activity.
- **WFM_ACTIVITY_NAME**—The name of the Activity.
- **WFM_ACTIVITY_SHORT_NAME**—The short name of the Activity.
- **WFM_ACTIVITY_TYPE_KEY**—The surrogate key used to join the WFM_ACTIVITY_TYPE dimension. It specifies the type of the Activity.
- **WFM_TIMESTAMP**—An internal timestamp value.
- **ACTIVE_FLAG**—Indicates whether the Activity is currently active: 0 = No, 1 = Yes.
- **TENANT_KEY**—The surrogate key used to join the TENANT dimension to the fact tables.
- **CREATE_AUDIT_KEY**—The surrogate key used to join to the CTL_AUDIT_LOG control table. The key specifies the lineage for data creation. This value is useful for aggregation, Enterprise Application Integration (EAI), and ETL tools (that is, applications that need to identify newly added data).
- **UPDATE_AUDIT_KEY**—The surrogate key used to join to the CTL_AUDIT_LOG control table. The key specifies the lineage for data update. This value is useful for aggregation, Enterprise Application Integration (EAI), and ETL tools (that is, applications that need to identify recently modified data).
- **PURGE_FLAG**—This field is reserved.

WFM_SSG_TYPE

This table contains schedule state group types and descriptions.

Column	Data type	P	M	F	DV
WFM_SSG_TYPE_KEY	int	X	X		
WFM_SSG_TYPE_NAME	varchar(64)		X		

Description of Columns

- **WFM_SSG_TYPE_KEY**—The Schedule State Group type ID.
- **WFM_SSG_TYPE_NAME**—The Schedule State Group type name. The table below contains valid values.

ID	Name
1	'Working Overhead'
2	'Non-Working Overhead'
3	'Actual Work'

WFM_SSG

This table contains schedule state group descriptive information.

Column	Data type	P	M	F	DV
WFM_SSG_KEY	int	X	X		
WFM_SITE_KEY	int		X	X	
WFM_SSG_NAME	varchar(255)		X		
WFM_SSG_TYPE_KEY	int		X	X	
WFM_SSG_WEIGHT	int		X		
WFM_TIMESTAMP	numeric(19)		X		
ACTIVE_FLAG	int		X		
TENANT_KEY	int			X	
CREATE_AUDIT_KEY	numeric(19)			X	
UPDATE_AUDIT_KEY	numeric(19)				
PURGE_FLAG	int				

Description of Columns

- **WFM_SSG_KEY**—The primary key for this table.
- **WFM_SITE_KEY**—*The surrogate key used to join the WFM_SITE dimension to the fact tables. It specifies the site of the Schedule State Group (SSG).
- **WFM_SSG_NAME**—The name of the Schedule State Group.
- **WFM_SSG_TYPE_KEY**—The surrogate key used to join the WFM_SSG_TYPE dimension. It specifies the type of the Schedule State Group.

- **WFM_SSG_WEIGHT**—The superficial weight value of Schedule State Group used for grouping.
- **WFM_TIMESTAMP**—An internal timestamp value.
- **ACTIVE_FLAG**—Indicates whether the Schedule State Group is currently active: 0 = No, 1 = Yes.
- **TENANT_KEY**—The surrogate key used to join the TENANT dimension to the fact tables.
- **CREATE_AUDIT_KEY**—The surrogate key used to join to the CTL_AUDIT_LOG control table. The key specifies the lineage for data creation. This value is useful for aggregation, enterprise application integration (EAI), and ETL tools (that is, applications that need to identify newly added data).
- **UPDATE_AUDIT_KEY**—The surrogate key used to join to the CTL_AUDIT_LOG control table. The key specifies the lineage for data update. This value is useful for aggregation, Enterprise Application Integration (EAI), and ETL tools (that is, applications that need to identify recently modified data).
- **PURGE_FLAG**—This field is reserved.

WFM_STATE_TYPE

This table contains schedule state types and descriptions.

Column	Data type	P	M	F	DV
WFM_STATE_TYPE_KEY	KEY	X	X		
WFM_STATE_TYPE_NAME	varchar(64)		X		

Description of Columns

- **WFM_STATE_TYPE_KEY**—The State type ID.
- **WFM_STATE_TYPE_NAME**—The State type name. The table below contains valid values.

ID	Name
0	'None'
1	'Day Off'
2	'Time Off'
3	'Exception'
4	'Break'
5	'Meal'
6	'Activity'
7	'Activity Set'
8	'Shift'
9	'Marked Time'

WFM_STATE

This table contains schedule state descriptive information.

Column	Data type	P	M	F	DV
WFM_STATE_KEY	int	X	X		
WFM_SITE_KEY	int		X	X	
WFM_SSG_KEY	int			X	
WFM_STATE_TYPE_KEY	int		X	X	
WFM_STATE_ID	int		X		
WFM_STATE_NAME	varchar(255)		X		
WFM_STATE_SHORT_NAME	varchar(6)		X		
TENANT_KEY	int			X	
CREATE_AUDIT_KEY	numeric (19)			X	
UPDATE_AUDIT_KEY	numeric(19)			X	
PURGE_FLAG	int				

Description of Columns

- **WFM_STATE_KEY**—The primary key for this table.
- **WFM_SITE_KEY**—The surrogate key used to join the WFM_SITE dimension to the fact tables. It specifies the Site of the Schedule State.
- **WFM_SSG_KEY**—The surrogate key used to join the WFM_SSG dimension to the fact tables. It specifies the SSG of the Schedule State.
- **WFM_STATE_TYPE_KEY**—The surrogate key used to join the WFM_STATE_TYPE dimension. It specifies the type of the Schedule State.
- **WFM_STATE_ID**—The ID of Schedule State corresponding to the type of Schedule State. The ID is unique within the context of Schedule State type.
- **WFM_STATE_NAME**—The name of the Schedule State.
- **WFM_STATE_SHORT_NAME**—The short name of the Schedule State.
- **TENANT_KEY**—The surrogate key used to join the TENANT dimension to the fact tables.
- **CREATE_AUDIT_KEY**—The surrogate key used to join to the CTL_AUDIT_LOG control table. The key specifies the lineage for data creation. This value is useful for aggregation, Enterprise Application Integration (EAI), and ETL tools (that is, applications that need to identify newly added data).
- **UPDATE_AUDIT_KEY**—The surrogate key used to join to the CTL_AUDIT_LOG control table. The key specifies the lineage for data update. This value is useful for aggregation, Enterprise Application Integration (EAI), and ETL tools (that is, applications that need to identify recently modified data).
- **PURGE_FLAG**—This field is reserved.

WFM_PERF_ITEM

This table contains performance items (statistics) and descriptions.

Column	Data type	P	M	F	DV
WFM_PERF_ITEM_KEY	int	X	X		
WFM_PERF_ITEM_CODE	varchar(64)		X		
WFM_PERF_ITEM_DESCRIPTION	varchar(256)		X		

Description of Columns

- **WFM_PERF_ITEM_KEY**—The Performance item ID.
- **WFM_PERF_ITEM_CODE**—The code of Performance item (statistic). The table below contains WFM Performance statistics.
- **WFM_PERF_ITEM_DESCRIPTION**—The description of the Performance item.

Performance Statistics

ID	Code	Description
12	FRC_CALC_STAFFING	Total Calculated Staffing Difference (between Optimal number of agents for Forecast workload and Calculated Staffing)
16	FRC_REQ_STAFFING	Total Required Staffing
38	FRC_CALC_SERVICE_PCT	Weighted average of (Achieved) Calculated Service Level % (weighted on Forecast Interaction Volume)
15	FRC_REQ_SERVICE_PCT	Weighted average of (Achieved) Required Service Level % (weighted on Forecast Interaction Volume) for Activity of type Deferred
40	FRC_CALC_ASA	Weighted average of (Achieved) Calculated Average Speed of Answer (weighted on Forecast Interaction Volume)
14	FRC_REQ_ASA	Weighted average of Required Average Speed of Answer (weighted on Forecast Interaction Volume)
39	FRC_CALC_ABANDONED_IV_PCT	Weighted average of (Achieved) Calculated Abandoned Interaction Volume % (weighted on Forecast Interaction Volume)

ID	Code	Description
18	FRC_REQ_ABANDONED_IV_PCT	Weighted average of Required Abandoned Interaction Volume % (weighted on Forecast Interaction Volume)
41	FRC_CALC_MAX_OCCUPANCY_PCT	Weighted average of (Achieved) Calculated Maximum Occupancy % (weighted on Forecast Interaction Volume)
77	FRC_REQ_MAX_OCCUPANCY_PCT	Weighted average of Required Maximum Occupancy % (weighted on Forecast Interaction Volume)
10	FRC_IV	Total of Forecast Interaction Volume
50	FRC_CALC_FTE	Total of Calculated Full-time Equivalent
51	FRC_REQ_FTE	Total of Required Full-time Equivalent
56	FRC_CALC_MAN_HOURS	Total of Calculated Man Hours
57	FRC_REQ_MAN_HOURS	Total of Required Man Hours
21	SCH_COVERAGE	Total of Scheduled Coverage
24	SCH_SERVICE_PCT	Weighted average of Scheduled Service Level % (weighted on Forecast Interaction Volume)
19	SCH_ASA	Weighted average of Scheduled Average Speed of Answer (weighted on Forecast Interaction Volume)
22	SCH_ABANDONED_IV_PCT	Weighted average of Scheduled Abandoned Interaction Volume % (weighted on Forecast Interaction Volume)
23	SCH_MAX_OCCUPANCY_PCT	Weighted average of Scheduled Maximum Occupancy % (weighted on Forecast Interaction Volume)
49	SCH_FTE	Total of Scheduled Full-time Equivalent
55	SCH_MAN_HOURS	Total of Scheduled Man Hours
6	ACT_STAFFING	Total of Actual Staffing Difference (between Optimal number of agents for Actual workload and Scheduled Coverage)
59	ACT_COVERAGE	Total of Actual Coverage (agent minutes divided by timestep)
3	ACT_SERVICE_PCT	Weighted average of Actual Service Level % (weighted on

ID	Code	Description
		Actual Distributed Interaction Volume) for Activity of type Deferred
5	ACT_ASA	Weighted average of Actual Average Speed of Answer (weighted on Actual Interaction Volume)
4	ACT_ABANDONED_IV_PCT	Total of Actual Abandoned Interaction Volume %
1	ACT_IV	Total of Actual Interaction Volume
62	ACT_ABANDONED_IV	Total of Actual Abandoned Interaction Volume %
8	ACT_DISTRIBUTED_IV	Total of Actual Distributed Interaction Volume
9	ACT_HANDLED_IV	Total of Actual Handled Interaction Volume
60	ACT_FTE	Total of Actual Full-time Equivalent
61	ACT_MAN_HOURS	Total of Actual Man Hours
20	SCH_HEADCOUNT	Total of Scheduled Headcount
2	ACT_AHT	Weighted average of Actual Handle Time (weighted on Actual Handled Interaction Volume)
78	ACT_SIMPLE_AHT	Simple average of Actual Handle Time
11	FRC_AHT	Weighted average of Forecast Handle Time (weighted on Forecast Interaction Volume)
58	FRC_SIMPLE_AHT	Simple average of Forecast Average Handle Time
70	SCH_AHT	Weighted average of Scheduled Average Handle Time (weighted on Forecast Interaction Volume)
69	SCH_IV	Total of Scheduled Interaction Volume

Fact tables

> **Important**
>
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

This topic describes the Fact tables in the Workforce Management (WFM) ETL Database schema.

Related Topics

For a description of the abbreviations used in these tables, see Abbreviations for ETL Database Terms.

WFM_ADH_AGENT_DAY

This table contains a 24-hour day aggregate of agent adherence information.

Column	Data type	P	M	F	DV
WFM_ADH_AGENT_DAY_KEY	numeric(19)	✔	✔		
WFM_AGENT_KEY	int		✔	✔	
WFM_SITE_KEY	int		✔	✔	
WFM_TEAM_KEY	int			✔	
WFM_DATE	date		✔		
WFM_NON_ADHERENCE_DURATION	int		✔		
WFM_OUT_SCH_NON_ADH_DURATION	int		✔		
WFM_SCHEDULE_DURATION	int		✔		
WFM_ACTUAL_WORK_DURATION	int		✔		
WFM_ADHERENCE_PERC	float		✔		
WFM_CONFORMANCE_PERC	float		✔		
WFM_TIMESTAMP	numeric(19)		✔		
DATE_TIME_DAY_KEY	int		✔	✔	
TENANT_KEY	int			✔	
CREATE_AUDIT_KEY	numeric(19)			✔	
UPDATE_AUDIT_KEY	numeric(19)			✔	

Column	Data type	P	M	F	DV
PURGE_FLAG	int				

Description of Columns

- **WFM_ADH_AGENT_DAY_KEY**—The primary key for this table.
- **WFM_AGENT_KEY**—The surrogate key used to join the WFM_AGENT dimension to the fact tables. It specifies the Agent of the Agent Adherence Day.
- **WFM_SITE_KEY**—The surrogate key used to join the WFM_SITE dimension to the fact tables. It specifies the Site of the Agent Adherence Day.
- **WFM_TEAM_KEY**—The surrogate key used to join the WFM_TEAM dimension to the fact tables. It specifies the historical Team of the Agent at the time of adherence date specified in WM_DATE column. It is NULL if the Agent was not under any team at that time.
- **WFM_DATE**—The date of the Agent Adherence Day in the Agent's Site time zone.
- **WFM_NON_ADHERENCE_DURATION**—The Agent’s total non-adherence time in seconds for the day.
- **WFM_OUT_SCH_NON_ADH_DURATION**—The Agent’s total out of schedule non-adherence time in seconds for the day.
- **WFM_SCHEDULE_DURATION**—The Agent’s total schedule time plus Agent’s total out of schedule non-adherence time for the day in seconds.
- **WFM_ACTUAL_WORK_DURATION**—The Agent’s total work time (logged in time) in seconds for the day.
- **WFM_ADHERENCE_PERC**—The Agent’s adherence percentage for the day. The adherence percentage is calculated using the following formula:}
  WFM_ADHERENCE_PERC = 100.0 - (100.0 * WFM_NON_ADHERENCE_DURATION) / WFM_SCHEDULE_DURATION
- **WFM_CONFORMANCE_PERC**—The Agent’s conformance percentage for the day, calculated by using the following formula:
  WFM_CONFORMANCE_PERC = (100.0 * WFM_ACTUAL_WORK_DURATION) / WFM_SCHEDULE_DURATION
- **WFM_TIMESTAMP**—An internal timestamp value.
- **DATE_TIME_DAY_KEY**—Identifies the start of a day interval in which the fact began and is equal to the UTC-equivalent time value, at which the day interval started. The value is the number of seconds that have elapsed since midnight on January 1, 1970, not counting leap seconds (also known as UNIX time). Use this value as a key to join the Fact tables to any configured DATE_TIME dimension to group the facts that are related to the same interval and/or convert day interval start to an appropriate time zone.
- **TENANT_KEY**—The surrogate key used to join the TENANT dimension to the Fact tables.
- **CREATE_AUDIT_KEY**—The surrogate key used to join to the CTL_AUDIT_LOG control table. The key specifies the lineage for data creation. This value is useful for aggregation, Enterprise Application Integration (EAI), and ETL tools (that is, applications that need to identify newly added data).
- **UPDATE_AUDIT_KEY**—The surrogate key used to join to the CTL_AUDIT_LOG control table. The key specifies the lineage for data update. This value is useful for aggregation, Enterprise Application Integration (EAI), and ETL tools (that is, applications that need to identify recently modified data).
- **PURGE_FLAG**—This field is reserved.

WFM_ADH_AGENT_TIMESTEP

This table contains a 24-hour day aggregate of agent adherence information.

Column	Data type	P	M	F	DV
WFM_ADH_AGENT_DAY_KEY	numeric(19)	✔	✔	✔	
WFM_TIME_STEP	datetime	✔	✔		
WFM_NON_ADHERENCE_DURATION	int		✔		
WFM_OUT_SCH_NON_ADH_DURATION	int		✔		
WFM_SCHEDULE_DURATION	int		✔		
WFM_ACTUAL_WORK_DURATION	int		✔		
DATE_TIME_KEY	int		✔	✔	
TENANT_KEY	int			✔	
CREATE_AUDIT_KEY	numeric(19)			✔	
UPDATE_AUDIT_KEY	numeric(19)			✔	
PURGE_FLAG	int				

Description of Columns

- **WFM_ADH_AGENT_DAY_KEY**—The surrogate key used to join parent WFM_ADH_AGENT_DAY record containing the Agent, Site and Team, as well as the corresponding calendar day information.
- **WFM_TIME_STEP**—The start date/time of 15-minute interval in the Agent's Site time zone.
- **WFM_NON_ADHERENCE_DURATION**—The Agent's total non-adherence time in seconds for the 15-minute interval.
- **WFM_OUT_SCH_NON_ADH_DURATION**—The Agent's total out of schedule non-adherence time in seconds for the 15-minute interval.
- **WFM_SCHEDULE_DURATION**—The Agent's total schedule time in seconds for the 15-minute interval.
- **WFM_ACTUAL_WORK_DURATION**—The Agent's total work time (logged in time) in seconds for the 15-minute interval.
- **DATE_TIME_KEY**—Identifies the start of a 15-minute interval, in which the fact began and is equal to the UTC-equivalent time, at which the interval started. The value is the number of seconds that have elapsed since midnight on January 1, 1970, not counting leap seconds (also known as UNIX time). Use this value as a key to join the Fact tables to any configured DATE_TIME dimension to group the facts that are related to the same interval and/or convert interval start to an appropriate time zone.
- **TENANT_KEY**—The surrogate key used to join the TENANT dimension to the Fact tables.
- **CREATE_AUDIT_KEY**—The surrogate key used to join to the CTL_AUDIT_LOG control table. The key specifies the lineage for data creation. This value is useful for aggregation, Enterprise Application Integration (EAI), and ETL tools (that is, applications that need to identify newly added data).
- **UPDATE_AUDIT_KEY**—The surrogate key used to join to the CTL_AUDIT_LOG control table. The key specifies the lineage for data update. This value is useful for aggregation, Enterprise Application Integration (EAI), and ETL tools (that is, applications that need to identify recently modified data).
- **PURGE_FLAG**—This field is reserved.

WFM_SCH_AGENT_DAY

This table contains the Agent's shift-day schedule information. The shift-day starts anywhere within the corresponding 24-hour calendar day, but it can end on the next calendar day, if the scheduled shift is an overnight shift.

Column	Data type	P	M	F	DV
WFM_SCH_AGENT_DAY_KEY	numeric(19)	✔	✔		
WFM_AGENT_KEY	int		✔	✔	
WFM_SITE_KEY	int		✔	✔	
WFM_TEAM_KEY	int			✔	
WFM_DATE	date		✔		
WFM_DAY_START	datetime		✔		
WFM_DAY_END	datetime		✔		
WFM_STATE_KEY	numeric(19)		✔	✔	
WFM_FULL_DAY	int		✔		
WFM_SCHEDULE_DURATION	real		✔		
WFM_WORK_DURATION	float		✔		
WFM_PAID_DURATION	float		✔		
WFM_OVERTIME_DURATION	real		✔		
WFM_TIMESTAMP	numeric(19)		✔		
START_DATE_TIME_KEY	int		✔	✔	
END_DATE_TIME_KEY	int		✔	✔	
START_TS	int		✔		
END_TS	int		✔		
TENANT_KEY	int			✔	
CREATE_AUDIT_KEY	numeric(19)			✔	
UPDATE_AUDIT_KEY	numeric(19)			✔	
PURGE_FLAG	int				

Description of Columns

- **WFM_SCH_AGENT_DAY_KEY**—The primary key for this table.
- **WFM_AGENT_KEY**—The surrogate key used to join the WFM_AGENT dimension to the fact tables. It specifies the Agent of the schedule day.
- **WFM_SITE_KEY**—The surrogate key used to join the WFM_SITE dimension to the fact tables. It specifies the Site of the schedule day.
- **WFM_TEAM_KEY**—The surrogate key used to join the WFM_TEAM dimension to the fact tables. It specifies the historical Team of the Agent at the time of schedule date specified in WM_DATE column. It is NULL if the Agent was not in any team at that time.

- **WFM_DATE**—The date of Agent Adherence Day in the Agent's Site time zone.
- **WFM_DAY_START**—The start date/time of schedule day in the Agent's Site time zone. It is a start time of the first (the earliest) schedule state within the schedule day.
- **WFM_DAY_END**—The end date/time of schedule day in the Agent's Site time zone. It is a end time of the last (the latest) schedule state within the schedule day.
- **WFM_STATE_KEY**—The surrogate key used to join the WFM_STATE dimension to the Fact tables. It specifies the full-day schedule state corresponding to the schedule day.
- **WFM_FULL_DAY**—Indicates whether the schedule is full-day or not: 0 = No, 1 = Yes. The full-day schedule day is one that has no specific start/end time defined (for example, Day-Off).
- **WFM_SCHEDULE_DURATION**—The total schedule time, in minutes, for the schedule day.
- **WFM_WORK_DURATION**—The total scheduled work on activities time, in minutes, for the schedule day.
- **WFM_PAID_DURATION**—The total scheduled paid time, in minutes, for the schedule day.
- **WFM_OVERTIME_DURATION**—The total scheduled overtime, in minutes, for the schedule day.
- **WFM_TIMESTAMP**—An internal timestamp value.
- **START_DATE_TIME_KEY**—Identifies the start of a 15-minute interval, in which the fact began. Use this value as a key to join the Fact tables to any configured DATE_TIME dimension to group the facts that are related to the same interval and/or convert the START_TS timestamp to an appropriate time zone.
- **END_DATE_TIME_KEY**—Identifies the start of a 15-minute interval, in which the fact ended. Use this value as a key to join the Fact tables to any configured DATE_TIME dimension to group the facts that are related to the same interval and/or convert the END_TS timestamp to an appropriate time zone.
- **START_TS**—The date and time, at which the fact began, as a Coordinated Universal Time (UTC) value—the number of seconds that have elapsed since midnight on January 1, 1970, not counting leap seconds (also known as UNIX time).
- **END_TS**—The date and time, at which the fact ended, as a Coordinated Universal Time (UTC) value—the number of seconds that have elapsed since midnight on January 1, 1970, not counting leap seconds (also known as UNIX time).
- **TENANT_KEY**—The surrogate key used to join the TENANT dimension to the Fact tables.
- **CREATE_AUDIT_KEY**—The surrogate key used to join to the CTL_AUDIT_LOG control table. The key specifies the lineage for data creation. This value is useful for aggregation, Enterprise Application Integration (EAI), and ETL tools (that is, applications that need to identify newly added data).
- **UPDATE_AUDIT_KEY**—The surrogate key used to join to the CTL_AUDIT_LOG control table. The key specifies the lineage for data update. This value is useful for aggregation, Enterprise Application Integration (EAI), and ETL tools (that is, applications that need to identify recently modified data).
- **PURGE_FLAG**—This field is reserved.

WFM_SCH_AGENT_TIMESTEP

This table contains a 15-minute interval aggregate of agent’s schedule information.

Column	Data type	P	M	F	DV
WFM_SCH_AGENT_DAY_KEY	Numeric(19)	✔	✔	✔	

Column	Data type	P	M	F	DV
WFM_TIME_STEP	datetime	✔	✔		
WFM_SCHEDULE_DURATION	float		✔		
WFM_WORK_DURATION	float		✔		
WFM_PAID_DURATION	float		✔		
WFM_OVERTIME_DURATION	float		✔		
DATE_TIME_KEY	int		✔	✔	
TENANT_KEY	int			✔	
CREATE_AUDIT_KEY	numeric(19)			✔	
UPDATE_AUDIT_KEY	numeric(19)			✔	
PURGE_FLAG	int				

Description of Columns

- **WFM_SCH_AGENT_DAY_KEY**—The surrogate key used to join parent WFM_SCH_AGENT_DAY record containing the Agent, Site and Team, as well as corresponding schedule day information.
- **WFM_TIME_STEP**—The start date/time of the 15-minute interval in the Agent's Site time zone.
- **WFM_SCHEDULE_DURATION**—The total schedule time, in minutes, for the 15-minute interval.
- **WFM_WORK_DURATION**—The total scheduled work on activities time, in minutes, for the 15-minute interval.
- **WFM_PAID_DURATION**—The total scheduled paid time, in minutes, for the 15-minute interval.
- **WFM_OVERTIME_DURATION**—The total scheduled overtime, in minutes, for the 15-minute interval.
- **DATE_TIME_KEY**—Identifies the start of a 15-minute interval, in which the fact began and is equal to the UTC-equivalent time, at which the interval started. The value is the number of seconds that have elapsed since midnight on January 1, 1970, not counting leap seconds (also known as UNIX time). Use this value as a key to join the Fact tables to any configured DATE_TIME dimension to group the facts that are related to the same interval and/or convert interval start to an appropriate time zone.
- **TENANT_KEY**—The surrogate key used to join the TENANT dimension to the Fact tables.
- **CREATE_AUDIT_KEY**—The surrogate key used to join to the CTL_AUDIT_LOG control table. The key specifies the lineage for data creation. This value is useful for aggregation, Enterprise Application Integration (EAI), and ETL tools (that is, applications that need to identify newly added data).
- **UPDATE_AUDIT_KEY**—The surrogate key used to join to the CTL_AUDIT_LOG control table. The key specifies the lineage for data update. This value is useful for aggregation, Enterprise Application Integration (EAI), and ETL tools (that is, applications that need to identify recently modified data).
- **PURGE_FLAG**—This field is reserved.

WFM_SCH_AGENT_STATE

This table contains agent’s schedule state information.

Column	Data type	P	M	F	DV
WFM_SCH_AGENT_DAY_KEY	numeric(19)	✔	✔	✔	
WFM_STATE_KEY	numeric(19)	✔	✔	✔	
WFM_STATE_START	datetime	✔	✔		
WFM_STATE_END	datetime		✔		
WFM_STATE_DURATION	float		✔		
WFM_PAID_DURATION	float		✔		
WFM_FULL_DAY	int		✔		
START_DATE_TIME_KEY	int		✔	✔	
END_DATE_TIME_KEY	int		✔	✔	
START_TS	int		✔		
END_TS	int		✔		
TENANT_KEY	int			✔	
CREATE_AUDIT_KEY	numeric(19)			✔	
UPDATE_AUDIT_KEY	numeric(19)			✔	
PURGE_FLAG	int				

Description of Columns

- **WFM_SCH_AGENT_DAY_KEY**—The surrogate key used to join the parent WFM_SCH_AGENT_DAY record containing the Agent, Site and Team, as well as corresponding schedule day information.
- **WFM_STATE_KEY**—The surrogate key used to join the WFM_STATE dimension to the Fact tables. It specifies the schedule state of the agent schedule state.
- **WFM_STATE_START**—The start date/time of the Agent schedule state in the Agent's Site time zone.
- **WFM_STATE_END**—The end date/time of the Agent schedule state in the Agent's Site time zone.
- **WFM_FULL_DAY**—Indicates whether the Agent schedule state is full-day or not: `0` = No, `1` = Yes. The full-day schedule state is one that has no specific start/end time defined (for example, Day-Off).
- **WFM_STATE_DURATION**—The schedule state duration in minutes.
- **WFM_PAID_DURATION**—The schedule state paid duration in minutes.
- **START_DATE_TIME_KEY**—Identifies the start of a 15-minute interval, in which the fact began. Use this value as a key to join the Fact tables to any configured DATE_TIME dimension to group the facts that are related to the same interval and/or convert the START_TS timestamp to an appropriate time zone.
- **END_DATE_TIME_KEY**—Identifies the start of a 15-minute interval, in which the fact ended. Use this value as a key to join the Fact tables to any configured DATE_TIME dimension to group the facts that are related to the same interval and/or convert the END_TS timestamp to an appropriate time zone.
- **START_TS**—The date and time, at which the fact began, as a Coordinated Universal Time (UTC) value—the number of seconds that have elapsed since midnight on January 1, 1970, not counting leap seconds (also known as UNIX time).
- **END_TS**—The date and time, at which the fact ended, as a Coordinated Universal Time (UTC) value—the number of seconds that have elapsed since midnight on January 1, 1970, not counting leap seconds (also known as UNIX time).

- **TENANT_KEY**—The surrogate key used to join the TENANT dimension to the Fact tables.
- **CREATE_AUDIT_KEY**—The surrogate key used to join to the CTL_AUDIT_LOG control table. The key specifies the lineage for data creation. This value is useful for aggregation, Enterprise Application Integration (EAI), and ETL tools (that is, applications that need to identify newly added data).
- **UPDATE_AUDIT_KEY**—The surrogate key used to join to the CTL_AUDIT_LOG control table. The key specifies the lineage for data update. This value is useful for aggregation, Enterprise Application Integration (EAI), and ETL tools (that is, applications that need to identify recently modified data).
- **PURGE_FLAG**—This field is reserved.

WFM_SCH_AGENT_STATE_TIMESTEP

This table contains a 15-minute interval aggregate of schedule state duration information.

Column	Data type	P	M	F	DV
WFM_SCH_AGENT_DAY_KEY	numeric(19)	✔	✔	✔	
WFM_STATE_KEY	numeric(19)	✔	✔	✔	
WFM_TIME_STEP	datetime	✔	✔		
WFM_STATE_DURATION	float		✔		
DATE_TIME_KEY	int		✔	✔	
TENANT_KEY	int			✔	
CREATE_AUDIT_KEY	numeric(19)			✔	
UPDATE_AUDIT_KEY	numeric(19)			✔	
PURGE_FLAG	int				

Description of Columns

- **WFM_SCH_AGENT_DAY_KEY**—The surrogate key used to join the parent WFM_SCH_AGENT_DAY record containing Agent, Site and Team, as well as corresponding schedule day information.
- **WFM_STATE_KEY**—The surrogate key used to join the WFM_STATE dimension to the Fact tables. It specifies the schedule state of the 15-minute interval aggregate.
- **WFM_TIME_STEP**—The start date/time of the 15-minute interval in the Agent's Site time zone.
- **WFM_STATE_DURATION**—The total schedule state time in minutes for the 15-minute interval.
- **DATE_TIME_KEY**—Identifies the start of a 15-minute interval, in which the fact began and is equal to the UTC-equivalent time, at which the interval started. The value is the number of seconds that have elapsed since midnight on January 1, 1970, not counting leap seconds (also known as UNIX time). Use this value as a key to join the Fact tables to any configured DATE_TIME dimension to group the facts that are related to the same interval and/or convert interval start to an appropriate time zone.
- **TENANT_KEY**—The surrogate key used to join the TENANT dimension to the Fact tables.
- **CREATE_AUDIT_KEY**—The surrogate key used to join to the CTL_AUDIT_LOG control table. The key specifies the lineage for data creation. This value is useful for aggregation, Enterprise Application Integration (EAI), and ETL tools (that is, applications that need to identify newly added data).

- **UPDATE_AUDIT_KEY**—The surrogate key used to join to the CTL_AUDIT_LOG control table. The key specifies the lineage for data update. This value is useful for aggregation, Enterprise Application Integration (EAI), and ETL tools (that is, applications that need to identify recently modified data).
- **PURGE_FLAG**—This field is reserved.

WFM_PERF_ITEM_DAY

This table contains a 24-hour calendar day aggregate of the activity and/or the site performance statistics.

Column	Data type	P	M	F	DV
WFM_PERF_ITEM_DAY_KEY	numeric(19)	✔	✔		
WFM_ACTIVITY_KEY	int			✔	
WFM_SITE_KEY	int			✔	
WFM_DATE	date		✔		
WFM_PERF_ITEM_KEY	int		✔	✔	
WFM_PERF_ITEM_VALUE	float		✔		
WFM_TIMESTAMP	numeric(19)		✔		
DATE_TIME_DAY_KEY	int		✔	✔	
TENANT_KEY	int			✔	
CREATE_AUDIT_KEY	numeric(19)			✔	
UPDATE_AUDIT_KEY	numeric(19)			✔	
PURGE_FLAG	int				

Description of Columns

- **WFM_PERF_ITEM_DAY_KEY**—The primary key for this table
- **WFM_ACTIVITY_KEY**—The surrogate key used to join the WFM_ACTIVITY dimension to the Fact tables. It specifies the Activity (Single-Site or Multi-Site or Activity Group) of the performance statistic aggregate for the day. It is NULL for the Site statistic aggregate.
- **WFM_SITE_KEY**—The surrogate key used to join the WFM_SITE dimension to the fact tables. It specifies the Site of the performance statistic aggregate for the day. It is NULL for the Activity statistic aggregate.
- **WFM_DATE**—The date of performance statistic day aggregate in time zone of the Activity or the Site. Single-Site Activity uses the Site time zone, while Multi-Site Activity and Activity Group use the Business Unit time zone.
- **WFM_PERF_ITEM_KEY**—The surrogate key used to join the WFM_PERF_ITEM dimension to the Fact tables. It specifies the performance statistic type of the day aggregate. See the list of available statistics in the description of the WFM_PERF_ITEM dimension.
- **WFM_PERF_ITEM_VALUE**—The value of the Activity or Site performance statistic aggregate for the day.
- **WFM_TIMESTAMP**—An internal timestamp value.

- **DATE_TIME_DAY_KEY**—Identifies the start of a day interval, in which the fact began and is equal to the UTC-equivalent time value, at which the day interval started. The value is the number of seconds that have elapsed since midnight on January 1, 1970, not counting leap seconds (also known as UNIX time). Use this value as a key to join the Fact tables to any configured DATE_TIME dimension to group the facts that are related to the same interval and/or convert day interval start to an appropriate time zone.
- **TENANT_KEY**—The surrogate key used to join the TENANT dimension to the Fact tables.
- **CREATE_AUDIT_KEY**—The surrogate key used to join to the CTL_AUDIT_LOG control table. The key specifies the lineage for data creation. This value is useful for aggregation, Enterprise Application Integration (EAI), and ETL tools (that is, applications that need to identify newly added data).
- **UPDATE_AUDIT_KEY**—The surrogate key used to join to the CTL_AUDIT_LOG control table. The key specifies the lineage for data update. This value is useful for aggregation, Enterprise Application Integration (EAI), and ETL tools (that is, applications that need to identify recently modified data).
- **PURGE_FLAG**—This field is reserved.

WFM_PERF_ITEM_TIMESTEP

This table contains a 15-minute interval aggregate of the activity and/or the site performance statistics.

Column	Data type	P	M	F	DV
WFM_PERF_ITEM_DAY_KEY	numeric(19)	✔	✔	✔	
WFM_TIME_STEP	datetime	✔	✔		
WFM_PERF_ITEM_VALUE	float		✔		
DATE_TIME_KEY	int		✔	✔	
TENANT_KEY	int			✔	
CREATE_AUDIT_KEY	numeric(19)			✔	
UPDATE_AUDIT_KEY	numeric(19)			✔	
PURGE_FLAG	int				

Description of Columns

- **WFM_PERF_ITEM_DAY_KEY**—The surrogate key used to join the parent WFM_PERF_ITEM_DAY record containing the Activity or Site Performance Statistic, as well as corresponding calendar day information.
- **WFM_TIME_STEP**—The start date/time of 15-minute interval in time zone of the Activity or Site. Single-Site Activity uses the Site time zone, while Multi-Site Activity and Activity Group use the Business Unit time zone.
- **WFM_PERF_ITEM_VALUE**—The value of the Activity or the Site performance statistic aggregate for the 15-minute interval.
- **DATE_TIME_KEY**—Identifies the start of a 15-minute interval, in which the fact began and is equal to the UTC-equivalent time, at which the interval started. The value is the number of seconds that have elapsed since midnight on January 1, 1970, not counting leap seconds (also known as UNIX time). Use this value as a key to join the Fact tables to any configured DATE_TIME dimension to group the facts that are related to the same interval and/or convert interval start to an appropriate time zone.

- **TENANT_KEY**—The surrogate key used to join the TENANT dimension to the Fact tables.
- **CREATE_AUDIT_KEY**—The surrogate key used to join to the CTL_AUDIT_LOG control table. The key specifies the lineage for data creation. This value is useful for aggregation, Enterprise Application Integration (EAI), and ETL tools (that is, applications that need to identify newly added data).
- **UPDATE_AUDIT_KEY**—The surrogate key used to join to the CTL_AUDIT_LOG control table. The key specifies the lineage for data update. This value is useful for aggregation, Enterprise Application Integration (EAI), and ETL tools (that is, applications that need to identify recently modified data).
- **PURGE_FLAG**—This field is reserved.

Query examples

> **Important**
>
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

This topic provides examples for the three types of queries that are used in the Workforce Management (WFM) ETL Database schema.

Related Topics

Adherence queries

Agent Adherence Totals Query (Daily Granularity)	**Show query.. [+]**
	
	`SELECT WFM_SITE.WFM_SITE_NAME,`
	`WFM_TEAM.WFM_TEAM_NAME,`
	`WFM_AGENT.FIRST_NAME,`
	`WFM_AGENT.LAST_NAME,`
	`WFM_ADH_AGENT_DAY.WFM_DATE,`
	`WFM_ADH_AGENT_DAY.WFM_SCHEDULE_DURATION,`
	`WFM_ADH_AGENT_DAY.WFM_NON_ADHERENCE_DURATION,`
	`WFM_ADH_AGENT_DAY.WFM_OUT_SCH_NON_ADH_DURATION,`
	`WFM_ADH_AGENT_DAY.WFM_ADHERENCE_PERC`
	`FROM WFM_ADH_AGENT_DAY`
	`JOIN WFM_SITE ON (WFM_SITE.WFM_SITE_KEY =`
	`WFM_ADH_AGENT_DAY.WFM_SITE_KEY)`
	`JOIN WFM_AGENT ON (WFM_AGENT.WFM_AGENT_KEY =`
	`WFM_ADH_AGENT_DAY.WFM_AGENT_KEY)`
	`LEFT JOIN WFM_TEAM ON (WFM_TEAM.WFM_TEAM_KEY =`
	`WFM_AGENT.WFM_TEAM_KEY)`
	`WHERE`
	`WFM_ADH_AGENT_DAY.WFM_DATE >= ? AND`
	`WFM_ADH_AGENT_DAY.WFM_DATE <= ?`
	`ORDER BY`
	`WFM_SITE.WFM_SITE_NAME,`
	`WFM_TEAM.WFM_TEAM_NAME,`
	`WFM_ADH_AGENT_DAY.WFM_DATE,`
	`WFM_AGENT.FIRST_NAME,`
	`WFM_AGENT.LAST_NAME`

Team Adherence Totals Query (Daily Granularity)	**Show query.. [+]**
	
	`SELECT`
	`WFM_BU.WFM_BU_NAME,`
	`WFM_SITE.WFM_SITE_NAME,`
	`WFM_ADH_AGENT_DAY.WFM_DATE,`
	`WFM_TEAM.WFM_TEAM_NAME,`

```
SUM(WFM_ADH_AGENT_DAY.WFM_SCHEDULE_DURATION),
SUM(WFM_ADH_AGENT_DAY.WFM_NON_ADHERENCE_DURATION),
SUM(WFM_ADH_AGENT_DAY.WFM_OUT_SCH_NON_ADH_DURATION)
FROM WFM_ADH_AGENT_DAY
JOIN WFM_SITE ON (WFM_SITE.WFM_SITE_KEY =
WFM_ADH_AGENT_DAY.WFM_SITE_KEY)
JOIN WFM_BU ON (WFM_BU.WFM_BU_KEY =
WFM_SITE.WFM_BU_KEY)
LEFT JOIN WFM_TEAM ON (WFM_TEAM.WFM_TEAM_KEY =
WFM_ADH_AGENT_DAY.WFM_TEAM_KEY)
WHERE
WFM_ADH_AGENT_DAY.WFM_DATE >= ? AND
WFM_ADH_AGENT_DAY.WFM_DATE <= ?
GROUP BY
WFM_BU.WFM_BU_NAME,
WFM_SITE.WFM_SITE_NAME,
WFM_ADH_AGENT_DAY.WFM_DATE,
WFM_TEAM.WFM_TEAM_NAME
HAVING SUM(WFM_ADH_AGENT_DAY.WFM_SCHEDULE_DURATION) >
0
ORDER BY
WFM_BU.WFM_BU_NAME,
WFM_SITE.WFM_SITE_NAME,
WFM_ADH_AGENT_DAY.WFM_DATE,
WFM_TEAM.WFM_TEAM_NAME
```

Schedule Queries

Schedule States Query

Show query.. [+]

```
SELECT
WFM_SITE.WFM_SITE_NAME,
WFM_TEAM.WFM_TEAM_NAME,
WFM_AGENT.EMPLOYEE_ID,
WFM_AGENT.FIRST_NAME,
WFM_AGENT.LAST_NAME,
WFM_SCH_AGENT_DAY.WFM_DATE,
WFM_STATE.WFM_STATE_NAME,
WFM_SSG.WFM_SSG_NAME,
WFM_SCH_AGENT_STATE.WFM_FULL_DAY,
WFM_SCH_AGENT_STATE.WFM_STATE_START,
WFM_SCH_AGENT_STATE.WFM_STATE_END,
WFM_SCH_AGENT_STATE.WFM_STATE_DURATION,
WFM_SCH_AGENT_STATE.WFM_PAID_DURATION
FROM WFM_SCH_AGENT_STATE
JOIN WFM_SCH_AGENT_DAY ON
(WFM_SCH_AGENT_DAY.WFM_SCH_AGENT_DAY_KEY =
WFM_SCH_AGENT_STATE.WFM_SCH_AGENT_DAY_KEY)
JOIN WFM_AGENT ON (WFM_AGENT.AGENT_KEY =
WFM_SCH_AGENT_DAY.WFM_AGENT_KEY)
JOIN WFM_SITE ON (WFM_SITE.WFM_SITE_KEY =
WFM_SCH_AGENT_DAY.WFM_SITE_KEY)
LEFT JOIN WFM_TEAM ON (WFM_TEAM.WFM_TEAM_KEY =
WFM_AGENT.WFM_TEAM_KEY)
JOIN WFM_STATE ON (WFM_STATE.WFM_STATE_KEY =
WFM_SCH_AGENT_STATE.WFM_STATE_KEY)
JOIN WFM_STATE_TYPE ON
(WFM_STATE_TYPE.WFM_STATE_TYPE_KEY =
WFM_STATE.WFM_STATE_TYPE_KEY)
LEFT JOIN WFM_SSG ON (WFM_SSG.WFM_SSG_KEY =
WFM_STATE.WFM_SSG_KEY)
WHERE
WFM_STATE_TYPE.WFM_STATE_TYPE_NAME NOT IN ('Shift',
'Activity Set', 'Marked Time') AND
```

	WFM_SCH_AGENT_STATE.WFM_STATE_END >= ? AND WFM_SCH_AGENT_STATE.WFM_STATE_START < ? **ORDER BY** WFM_SITE.WFM_SITE_NAME, WFM_TEAM.WFM_TEAM_NAME, WFM_AGENT.EMPLOYEE_ID, WFM_SCH_AGENT_DAY.WFM_DATE, WFM_SCH_AGENT_STATE.WFM_STATE_START, WFM_STATE.WFM_STATE_KEY
Agent Schedule State Totals Query	**Show query.. [+]** **SELECT** WFM_SITE.WFM_SITE_NAME, WFM_TEAM.WFM_TEAM_NAME, WFM_AGENT.FIRST_NAME, WFM_AGENT.LAST_NAME, WFM_SCH_AGENT_DAY.WFM_DATE, **SUM**(WFM_SCH_AGENT_STATE_TIMESTEP.WFM_STATE_DURATION) **FROM** WFM_SCH_AGENT_STATE_TIMESTEP JOIN WFM_SCH_AGENT_DAY **ON** (WFM_SCH_AGENT_DAY.WFM_SCH_AGENT_DAY_KEY = WFM_SCH_AGENT_STATE_TIMESTEP.WFM_SCH_AGENT_DAY_KEY) JOIN WFM_AGENT **ON** (WFM_AGENT.AGENT_KEY = WFM_SCH_AGENT_DAY.WFM_AGENT_KEY) JOIN WFM_SITE **ON** (WFM_SITE.WFM_SITE_KEY = WFM_SCH_AGENT_DAY.WFM_SITE_KEY) LEFT JOIN WFM_TEAM **ON** (WFM_TEAM.WFM_TEAM_KEY = WFM_AGENT.WFM_TEAM_KEY) JOIN WFM_STATE **ON** (WFM_STATE.WFM_STATE_KEY = WFM_SCH_AGENT_STATE_TIMESTEP.WFM_STATE_KEY) JOIN WFM_SSG **ON** (WFM_SSG.WFM_SSG_KEY = WFM_STATE.WFM_SSG_KEY) **WHERE** WFM_SCH_AGENT_DAY.WFM_DATE >= ? AND WFM_SCH_AGENT_DAY.WFM_DATE <= ? **GROUP BY** WFM_SITE.WFM_SITE_NAME, WFM_TEAM.WFM_TEAM_NAME, WFM_AGENT.FIRST_NAME, WFM_AGENT.LAST_NAME, WFM_SCH_AGENT_DAY.WFM_DATE **ORDER BY** WFM_SITE.WFM_SITE_NAME, WFM_TEAM.WFM_TEAM_NAME, WFM_AGENT.FIRST_NAME, WFM_AGENT.LAST_NAME, WFM_SCH_AGENT_DAY.WFM_DATE
Team Schedule State Totals Query	**Show query.. [+]** **SELECT** WFM_BU.WFM_BU_NAME, WFM_SITE.WFM_SITE_NAME, WFM_TEAM.WFM_TEAM_NAME, WFM_SCH_AGENT_DAY.WFM_DATE, **SUM**(WFM_SCH_AGENT_STATE_TIMESTEP.WFM_STATE_DURATION) **FROM** WFM_SCH_AGENT_STATE_TIMESTEP JOIN WFM_SCH_AGENT_DAY **ON** (WFM_SCH_AGENT_DAY.WFM_SCH_AGENT_DAY_KEY = WFM_SCH_AGENT_STATE_TIMESTEP.WFM_SCH_AGENT_DAY_KEY) JOIN WFM_AGENT **ON** (WFM_AGENT.AGENT_KEY = WFM_SCH_AGENT_DAY.WFM_AGENT_KEY) JOIN WFM_SITE **ON** (WFM_SITE.WFM_SITE_KEY = WFM_SCH_AGENT_DAY.WFM_SITE_KEY) JOIN WFM_BU **ON** (WFM_BU.WFM_BU_KEY = WFM_SITE.WFM_BU_KEY) LEFT JOIN WFM_TEAM **ON** (WFM_TEAM.WFM_TEAM_KEY =

	WFM_SCH_AGENT_DAY.WFM_TEAM_KEY) JOIN WFM_STATE **ON** (WFM_STATE.WFM_STATE_KEY = WFM_SCH_AGENT_STATE_TIMESTEP.WFM_STATE_KEY) JOIN WFM_SSG **ON** (WFM_SSG.WFM_SSG_KEY = WFM_STATE.WFM_SSG_KEY) **WHERE** WFM_SCH_AGENT_DAY.WFM_DATE >= ? AND WFM_SCH_AGENT_DAY.WFM_DATE <= ? **GROUP BY** WFM_BU.WFM_BU_NAME, WFM_SITE.WFM_SITE_NAME, WFM_TEAM.WFM_TEAM_NAME, WFM_SCH_AGENT_DAY.WFM_DATE **ORDER BY** WFM_BU.WFM_BU_NAME, WFM_SITE.WFM_SITE_NAME, WFM_TEAM.WFM_TEAM_NAME, WFM_SCH_AGENT_DAY.WFM_DATE
Schedule Marked Time Report Query	**Show query.. [+]** **SELECT** WFM_SITE.WFM_SITE_NAME, WFM_TEAM.WFM_TEAM_NAME, WFM_AGENT.FIRST_NAME, WFM_AGENT.LAST_NAME, WFM_SCH_AGENT_DAY.WFM_DATE, WFM_STATE.WFM_STATE_NAME, WFM_SCH_AGENT_STATE.WFM_STATE_START, WFM_SCH_AGENT_STATE.WFM_STATE_END, WFM_SCH_AGENT_STATE.WFM_STATE_DURATION, WFM_SCH_AGENT_STATE.WFM_PAID_DURATION **FROM** WFM_SCH_AGENT_STATE JOIN WFM_SCH_AGENT_DAY **ON** (WFM_SCH_AGENT_DAY.WFM_SCH_AGENT_DAY_KEY = WFM_SCH_AGENT_STATE.WFM_SCH_AGENT_DAY_KEY) JOIN WFM_AGENT **ON** (WFM_AGENT.AGENT_KEY = WFM_SCH_AGENT_DAY.WFM_AGENT_KEY) JOIN WFM_SITE **ON** (WFM_SITE.WFM_SITE_KEY = WFM_SCH_AGENT_DAY.WFM_SITE_KEY) LEFT JOIN WFM_TEAM **ON** (WFM_TEAM.WFM_TEAM_KEY = WFM_AGENT.WFM_TEAM_KEY) JOIN WFM_STATE **ON** (WFM_STATE.WFM_STATE_KEY = WFM_SCH_AGENT_STATE.WFM_STATE_KEY) JOIN WFM_STATE_TYPE **ON** (WFM_STATE_TYPE.WFM_STATE_TYPE_KEY = WFM_STATE.WFM_STATE_TYPE_KEY) **WHERE** WFM_STATE_TYPE.WFM_STATE_TYPE_NAME IN ('Marked Time') AND WFM_SCH_AGENT_STATE.WFM_STATE_END >= ? AND WFM_SCH_AGENT_STATE.WFM_STATE_START < ? **ORDER BY** WFM_SITE.WFM_SITE_NAME, WFM_TEAM.WFM_TEAM_NAME, WFM_AGENT.FIRST_NAME, WFM_AGENT.LAST_NAME, WFM_SCH_AGENT_DAY.WFM_DATE, WFM_SCH_AGENT_STATE.WFM_STATE_START, WFM_STATE.WFM_STATE_KEY
Schedule Marked Time Totals Query (Daily Granularity)	**Show query.. [+]** **SELECT** WFM_SITE.WFM_SITE_NAME, WFM_TEAM.WFM_TEAM_NAME, WFM_AGENT.FIRST_NAME, WFM_AGENT.LAST_NAME,

	WFM_SCH_AGENT_DAY.WFM_DATE, **SUM**(WFM_SCH_AGENT_STATE.WFM_STATE_DURATION), **SUM**(WFM_SCH_AGENT_STATE.WFM_PAID_DURATION) **FROM** WFM_SCH_AGENT_STATE JOIN WFM_SCH_AGENT_DAY **ON** (WFM_SCH_AGENT_DAY.WFM_SCH_AGENT_DAY_KEY = WFM_SCH_AGENT_STATE.WFM_SCH_AGENT_DAY_KEY) JOIN WFM_AGENT **ON** (WFM_AGENT.AGENT_KEY = WFM_SCH_AGENT_DAY.WFM_AGENT_KEY) JOIN WFM_SITE **ON** (WFM_SITE.WFM_SITE_KEY = WFM_SCH_AGENT_DAY.WFM_SITE_KEY) LEFT JOIN WFM_TEAM **ON** (WFM_TEAM.WFM_TEAM_KEY = WFM_AGENT.WFM_TEAM_KEY) JOIN WFM_STATE **ON** (WFM_STATE.WFM_STATE_KEY = WFM_SCH_AGENT_STATE.WFM_STATE_KEY) JOIN WFM_STATE_TYPE **ON** (WFM_STATE_TYPE.WFM_STATE_TYPE_KEY = WFM_STATE.WFM_STATE_TYPE_KEY) **WHERE** WFM_STATE_TYPE.WFM_STATE_TYPE_NAME IN ('Marked Time') AND WFM_SCH_AGENT_STATE.WFM_STATE_END >= ? AND WFM_SCH_AGENT_STATE.WFM_STATE_START < ? **GROUP BY** WFM_SITE.WFM_SITE_NAME, WFM_TEAM.WFM_TEAM_NAME, WFM_AGENT.FIRST_NAME, WFM_AGENT.LAST_NAME, WFM_SCH_AGENT_DAY.WFM_DATE **ORDER BY** WFM_SITE.WFM_SITE_NAME, WFM_TEAM.WFM_TEAM_NAME, WFM_AGENT.FIRST_NAME, WFM_AGENT.LAST_NAME, WFM_SCH_AGENT_DAY.WFM_DATE
Schedule Marked Time Totals Query (Time Step Granularity)	**Show query.. [+]** **SELECT** WFM_SITE.WFM_SITE_NAME, WFM_TEAM.WFM_TEAM_NAME, WFM_AGENT.FIRST_NAME, WFM_AGENT.LAST_NAME, WFM_SCH_AGENT_STATE_TIMESTEP.WFM_TIME_STEP, WFM_SCH_AGENT_STATE_TIMESTEP.WFM_STATE_DURATION **FROM** WFM_SCH_AGENT_STATE_TIMESTEP JOIN WFM_SCH_AGENT_DAY **ON** (WFM_SCH_AGENT_DAY.WFM_SCH_AGENT_DAY_KEY = WFM_SCH_AGENT_STATE_TIMESTEP.WFM_SCH_AGENT_DAY_KEY) JOIN WFM_AGENT **ON** (WFM_AGENT.AGENT_KEY = WFM_SCH_AGENT_DAY.WFM_AGENT_KEY) JOIN WFM_SITE **ON** (WFM_SITE.WFM_SITE_KEY = WFM_SCH_AGENT_DAY.WFM_SITE_KEY) LEFT JOIN WFM_TEAM **ON** (WFM_TEAM.WFM_TEAM_KEY = WFM_AGENT.WFM_TEAM_KEY) JOIN WFM_STATE **ON** (WFM_STATE.WFM_STATE_KEY = WFM_SCH_AGENT_STATE_TIMESTEP.WFM_STATE_KEY) JOIN WFM_STATE_TYPE **ON** (WFM_STATE_TYPE.WFM_STATE_TYPE_KEY = WFM_STATE.WFM_STATE_TYPE_KEY) **WHERE** WFM_STATE_TYPE.WFM_STATE_TYPE_NAME IN ('Marked Time') AND WFM_SCH_AGENT_STATE_TIMESTEP.WFM_TIME_STEP >= ? AND WFM_SCH_AGENT_STATE_TIMESTEP.WFM_TIME_STEP < ? **ORDER BY** WFM_SITE.WFM_SITE_NAME, WFM_TEAM.WFM_TEAM_NAME, WFM_AGENT.FIRST_NAME,

	WFM_AGENT.LAST_NAME, WFM_SCH_AGENT_STATE_TIMESTEP.WFM_TIME_STEP
Weekly Schedule Report Query	**Show query.. [+]** **SELECT** WFM_SITE.WFM_SITE_NAME, WFM_TEAM.WFM_TEAM_NAME, WFM_AGENT.EMPLOYEE_ID, WFM_AGENT.FIRST_NAME, WFM_AGENT.LAST_NAME, WFM_SCH_AGENT_DAY.WFM_DATE, WFM_STATE.WFM_STATE_NAME, WFM_SCH_AGENT_DAY.WFM_FULL_DAY, WFM_SCH_AGENT_DAY.WFM_DAY_START, WFM_SCH_AGENT_DAY.WFM_DAY_END, **SUM**(WFM_SCH_AGENT_DAY.WFM_SCHEDULE_DURATION) **AS** SCHEDULE_DURATION, **SUM**(WFM_SCH_AGENT_DAY.WFM_PAID_DURATION) **AS** PAID_DURATION, **SUM**(WFM_SCH_AGENT_DAY.WFM_WORK_DURATION) **AS** WORK_DURATION, **SUM**(WFM_SCH_AGENT_DAY.WFM_OVERTIME_DURATION) **AS** OVERTIME_DURATION **FROM** WFM_SCH_AGENT_DAY JOIN WFM_STATE **ON** (WFM_STATE.WFM_STATE_KEY = WFM_SCH_AGENT_DAY.WFM_STATE_KEY) JOIN WFM_SITE **ON** (WFM_SITE.WFM_SITE_KEY = WFM_SCH_AGENT_DAY.WFM_SITE_KEY) JOIN WFM_AGENT **ON** (WFM_AGENT.WFM_AGENT_KEY = WFM_SCH_AGENT_DAY.WFM_AGENT_KEY) LEFT JOIN WFM_TEAM **ON** (WFM_TEAM.WFM_TEAM_KEY = WFM_AGENT.WFM_TEAM_KEY) **WHERE** WFM_SCH_AGENT_DAY.WFM_DATE >= ? AND WFM_SCH_AGENT_DAY.WFM_DATE <= ? **GROUP BY** WFM_SITE.WFM_SITE_NAME, WFM_TEAM.WFM_TEAM_NAME, WFM_AGENT.EMPLOYEE_ID, WFM_AGENT.FIRST_NAME, WFM_AGENT.LAST_NAME, WFM_SCH_AGENT_DAY.WFM_DATE, WFM_STATE.WFM_STATE_NAME, WFM_SCH_AGENT_DAY.WFM_DAY_START, WFM_SCH_AGENT_DAY.WFM_DAY_END, WFM_SCH_AGENT_DAY.WFM_FULL_DAY **ORDER BY** WFM_SITE.WFM_SITE_NAME, WFM_TEAM.WFM_TEAM_NAME, WFM_AGENT.EMPLOYEE_ID, WFM_AGENT.FIRST_NAME, WFM_AGENT.LAST_NAME, WFM_SCH_AGENT_DAY.WFM_DATE
Schedule State Group (SSG) Totals Query	**Show query.. [+]** **SELECT** WFM_BU.WFM_BU_NAME, WFM_SITE.WFM_SITE_NAME, WFM_SCH_AGENT_STATE_TIMESTEP.WFM_TIME_STEP, WFM_SSG.WFM_SSG_NAME, **SUM**(WFM_SCH_AGENT_STATE_TIMESTEP.WFM_STATE_DURATION) / 15 **AS** WFM_SSG_TOTAL, WFM_SSG.WFM_SSG_WEIGHT **FROM** WFM_SCH_AGENT_STATE_TIMESTEP JOIN WFM_SCH_AGENT_DAY **ON** (WFM_SCH_AGENT_DAY.WFM_SCH_AGENT_DAY_KEY =

	WFM_SCH_AGENT_STATE_TIMESTEP.WFM_SCH_AGENT_DAY_KEY) JOIN WFM_AGENT **ON** (WFM_AGENT.AGENT_KEY = WFM_SCH_AGENT_DAY.WFM_AGENT_KEY) JOIN WFM_SITE **ON** (WFM_SITE.WFM_SITE_KEY = WFM_SCH_AGENT_DAY.WFM_SITE_KEY) JOIN WFM_BU **ON** (WFM_BU.WFM_BU_KEY = WFM_SITE.WFM_BU_KEY) JOIN WFM_STATE **ON** (WFM_STATE.WFM_STATE_KEY = WFM_SCH_AGENT_STATE_TIMESTEP.WFM_STATE_KEY) JOIN WFM_SSG **ON** (WFM_SSG.WFM_SSG_KEY = WFM_STATE.WFM_SSG_KEY) **WHERE** WFM_SCH_AGENT_STATE_TIMESTEP.WFM_TIME_STEP >= '11/14/ 2013' AND WFM_SCH_AGENT_STATE_TIMESTEP.WFM_TIME_STEP < '11/15/2013' AND WFM_SITE.WFM_SITE_NAME = 'Sched Pot 4' **GROUP BY** WFM_BU.WFM_BU_NAME, WFM_SITE.WFM_SITE_NAME, WFM_SCH_AGENT_STATE_TIMESTEP.WFM_TIME_STEP, WFM_SSG.WFM_SSG_NAME, WFM_SSG.WFM_SSG_WEIGHT **ORDER BY** WFM_SCH_AGENT_STATE_TIMESTEP.WFM_TIME_STEP, WFM_SSG.WFM_SSG_WEIGHT
Activity Schedule Coverage Query	**Show query.. [+]** **SELECT** WFM_BU.WFM_BU_NAME, WFM_SITE.WFM_SITE_NAME, WFM_ACTIVITY.WFM_ACTIVITY_NAME, WFM_SCH_AGENT_STATE_TIMESTEP.WFM_TIME_STEP, **SUM**(WFM_SCH_AGENT_STATE_TIMESTEP.WFM_STATE_DURATION) / 15 **AS** WFM_ACTIVITY_COVERAGE **FROM** WFM_SCH_AGENT_STATE_TIMESTEP JOIN WFM_SCH_AGENT_DAY **ON** (WFM_SCH_AGENT_DAY.WFM_SCH_AGENT_DAY_KEY = WFM_SCH_AGENT_STATE_TIMESTEP.WFM_SCH_AGENT_DAY_KEY) JOIN WFM_AGENT **ON** (WFM_AGENT.AGENT_KEY = WFM_SCH_AGENT_DAY.WFM_AGENT_KEY) JOIN WFM_SITE **ON** (WFM_SITE.WFM_SITE_KEY = WFM_SCH_AGENT_DAY.WFM_SITE_KEY) JOIN WFM_BU **ON** (WFM_BU.WFM_BU_KEY = WFM_SITE.WFM_BU_KEY) JOIN WFM_STATE **ON** (WFM_STATE.WFM_STATE_KEY = WFM_SCH_AGENT_STATE_TIMESTEP.WFM_STATE_KEY) JOIN WFM_STATE_TYPE **ON** (WFM_STATE_TYPE.WFM_STATE_TYPE_KEY = WFM_STATE.WFM_STATE_TYPE_KEY) JOIN WFM_ACTIVITY **ON** (WFM_ACTIVITY.WFM_ACTIVITY_KEY = WFM_STATE.WFM_STATE_ID AND WFM_STATE_TYPE.WFM_STATE_TYPE_NAME = 'Activity') **WHERE** WFM_SCH_AGENT_STATE_TIMESTEP.WFM_TIME_STEP >= '11/14/ 2013' AND WFM_SCH_AGENT_STATE_TIMESTEP.WFM_TIME_STEP < '11/15/2013' AND WFM_SITE.WFM_SITE_NAME = 'Sched Pot 4' AND WFM_ACTIVITY.WFM_ACTIVITY_NAME = 'Broadband Priority Care' **GROUP BY** WFM_BU.WFM_BU_NAME, WFM_SITE.WFM_SITE_NAME, WFM_ACTIVITY.WFM_ACTIVITY_NAME, WFM_SCH_AGENT_STATE_TIMESTEP.WFM_TIME_STEP ORDER BY WFM_SCH_AGENT_STATE_TIMESTEP.WFM_TIME_STEP, WFM_ACTIVITY.WFM_ACTIVITY_NAME

Performance Statistics Queries

Schedule Daily Summary for Activity	**Show query.. [+]** **SELECT** WFM_SITE.WFM_SITE_NAME, WFM_ACTIVITY.WFM_ACTIVITY_NAME, WFM_PERF_ITEM_DAY.WFM_DATE, WFM_PERF_ITEM.WFM_PERF_ITEM_CODE, **SUM**(WFM_PERF_ITEM_DAY.WFM_PERF_ITEM_VALUE) **FROM** WFM_PERF_ITEM_DAY JOIN WFM_ACTIVITY **ON** (WFM_ACTIVITY.WFM_ACTIVITY_KEY = WFM_PERF_ITEM_DAY.WFM_ACTIVITY_KEY) JOIN WFM_SITE **ON** (WFM_SITE.WFM_SITE_KEY = WFM_ACTIVITY.WFM_SITE_KEY) JOIN WFM_PERF_ITEM **ON** (WFM_PERF_ITEM.WFM_PERF_ITEM_KEY = WFM_PERF_ITEM_DAY.WFM_PERF_ITEM_KEY) **WHERE** WFM_PERF_ITEM_DAY.WFM_DATE >= ? AND WFM_PERF_ITEM_DAY.WFM_DATE <= ? AND WFM_PERF_ITEM.WFM_PERF_ITEM_CODE IN ('SCH_HEADCOUNT', 'SCH_SERVICE_PCT', 'FRC_CALC_SERVICE_PCT', 'SCH_IV', 'FRC_IV', 'SCH_AHT', 'FRC_AHT', 'FRC_CALC_MAN_HOURS', 'FRC_REQ_MAN_HOURS', 'SCH_MAN_HOURS', 'SCH_ASA', 'FRC_CALC_ASA', 'SCH_MAX_OCCUPANCY_PCT', 'FRC_CALC_MAX_OCCUPANCY_PCT') **GROUP BY** WFM_SITE.WFM_SITE_NAME, WFM_ACTIVITY.WFM_ACTIVITY_NAME, WFM_PERF_ITEM_DAY.WFM_DATE, WFM_PERF_ITEM.WFM_PERF_ITEM_CODE **ORDER BY** WFM_SITE.WFM_SITE_NAME, WFM_ACTIVITY.WFM_ACTIVITY_NAME, WFM_PERF_ITEM_DAY.WFM_DATE, WFM_PERF_ITEM.WFM_PERF_ITEM_CODE
Schedule Daily Summary for Multi-Site Activity (MSA)	**Show query.. [+]** **SELECT** WFM_BU.WFM_BU_NAME, WFM_ACTIVITY.WFM_ACTIVITY_NAME, WFM_PERF_ITEM_DAY.WFM_DATE, WFM_PERF_ITEM.WFM_PERF_ITEM_CODE, **SUM**(WFM_PERF_ITEM_DAY.WFM_PERF_ITEM_VALUE) **FROM** WFM_PERF_ITEM_DAY JOIN WFM_ACTIVITY **ON** (WFM_ACTIVITY.WFM_ACTIVITY_KEY = WFM_PERF_ITEM_DAY.WFM_ACTIVITY_KEY AND WFM_ACTIVITY.WFM_SITE_KEY IS NULL AND WFM_ACTIVITY.WFM_ACTIVITY_TYPE_KEY <> 10) JOIN WFM_BU **ON** (WFM_BU.WFM_BU_KEY = WFM_ACTIVITY.WFM_BU_KEY) JOIN WFM_PERF_ITEM **ON** (WFM_PERF_ITEM.WFM_PERF_ITEM_KEY = WFM_PERF_ITEM_DAY.WFM_PERF_ITEM_KEY) **WHERE** WFM_PERF_ITEM_DAY.WFM_DATE >= ? AND WFM_PERF_ITEM_DAY.WFM_DATE <= ? AND WFM_PERF_ITEM.WFM_PERF_ITEM_CODE IN ('SCH_HEADCOUNT', 'SCH_SERVICE_PCT', 'FRC_CALC_SERVICE_PCT', 'SCH_IV', 'FRC_IV', 'SCH_AHT', 'FRC_AHT', 'FRC_CALC_FTE', 'FRC_REQ_FTE', 'SCH_FTE', 'SCH_ASA', 'FRC_CALC_ASA', 'SCH_MAX_OCCUPANCY_PCT', 'FRC_CALC_MAX_OCCUPANCY_PCT') **GROUP BY** WFM_BU.WFM_BU_NAME, WFM_ACTIVITY.WFM_ACTIVITY_NAME, WFM_PERF_ITEM_DAY.WFM_DATE, WFM_PERF_ITEM.WFM_PERF_ITEM_CODE **ORDER BY**

	WFM_BU.WFM_BU_NAME, WFM_ACTIVITY.WFM_ACTIVITY_NAME, WFM_PERF_ITEM_DAY.WFM_DATE, WFM_PERF_ITEM.WFM_PERF_ITEM_CODE
Schedule Daily Summary for Activity Group (AG)	**Show query.. [+]** **SELECT** WFM_BU.WFM_BU_NAME, WFM_ACTIVITY.WFM_ACTIVITY_NAME, WFM_PERF_ITEM_DAY.WFM_DATE, WFM_PERF_ITEM.WFM_PERF_ITEM_CODE, **SUM**(WFM_PERF_ITEM_DAY.WFM_PERF_ITEM_VALUE) **FROM** WFM_PERF_ITEM_DAY JOIN WFM_ACTIVITY **ON** (WFM_ACTIVITY.WFM_ACTIVITY_KEY = WFM_PERF_ITEM_DAY.WFM_ACTIVITY_KEY AND WFM_ACTIVITY.WFM_SITE_KEY IS NULL AND WFM_ACTIVITY.WFM_ACTIVITY_TYPE_KEY = 10) JOIN WFM_BU **ON** (WFM_BU.WFM_BU_KEY = WFM_ACTIVITY.WFM_BU_KEY) JOIN WFM_PERF_ITEM **ON** (WFM_PERF_ITEM.WFM_PERF_ITEM_KEY = WFM_PERF_ITEM_DAY.WFM_PERF_ITEM_KEY) **WHERE** WFM_PERF_ITEM_DAY.WFM_DATE >= ? AND WFM_PERF_ITEM_DAY.WFM_DATE <= ? AND WFM_PERF_ITEM.WFM_PERF_ITEM_CODE IN ('SCH_HEADCOUNT', 'SCH_SERVICE_PCT', 'FRC_CALC_SERVICE_PCT', 'SCH_IV', 'FRC_IV', 'SCH_AHT', 'FRC_AHT', 'FRC_CALC_MAN_HOURS', 'FRC_REQ_MAN_HOURS', 'SCH_MAN_HOURS', 'SCH_ASA', 'FRC_CALC_ASA', 'SCH_MAX_OCCUPANCY_PCT', 'FRC_CALC_MAX_OCCUPANCY_PCT') **GROUP BY** WFM_BU.WFM_BU_NAME, WFM_ACTIVITY.WFM_ACTIVITY_NAME, WFM_PERF_ITEM_DAY.WFM_DATE, WFM_PERF_ITEM.WFM_PERF_ITEM_CODE **ORDER BY** WFM_BU.WFM_BU_NAME, WFM_ACTIVITY.WFM_ACTIVITY_NAME, WFM_PERF_ITEM_DAY.WFM_DATE, WFM_PERF_ITEM.WFM_PERF_ITEM_CODE
Schedule Daily Summary for Site	**Show query.. [+]** **SELECT** WFM_BU.WFM_BU_NAME, WFM_SITE.WFM_SITE_NAME, WFM_PERF_ITEM_DAY.WFM_DATE, WFM_PERF_ITEM.WFM_PERF_ITEM_CODE, **SUM**(WFM_PERF_ITEM_DAY.WFM_PERF_ITEM_VALUE) **FROM** WFM_PERF_ITEM_DAY JOIN WFM_SITE **ON** (WFM_SITE.WFM_SITE_KEY = WFM_PERF_ITEM_DAY.WFM_SITE_KEY) JOIN WFM_BU **ON** (WFM_BU.WFM_BU_KEY = WFM_SITE.WFM_BU_KEY) JOIN WFM_PERF_ITEM **ON** (WFM_PERF_ITEM.WFM_PERF_ITEM_KEY = WFM_PERF_ITEM_DAY.WFM_PERF_ITEM_KEY) **WHERE** WFM_PERF_ITEM_DAY.WFM_DATE >= ? AND WFM_PERF_ITEM_DAY.WFM_DATE <= ? AND WFM_PERF_ITEM.WFM_PERF_ITEM_CODE IN ('SCH_HEADCOUNT', 'SCH_SERVICE_PCT', 'FRC_CALC_SERVICE_PCT', 'SCH_IV', 'FRC_IV', 'SCH_AHT', 'FRC_AHT', 'FRC_CALC_FTE', 'FRC_REQ_FTE', 'SCH_FTE', 'SCH_ASA', 'FRC_CALC_ASA', 'SCH_MAX_OCCUPANCY_PCT', 'FRC_CALC_MAX_OCCUPANCY_PCT') **GROUP BY** WFM_BU.WFM_BU_NAME, WFM_SITE.WFM_SITE_NAME, WFM_PERF_ITEM_DAY.WFM_DATE,

	WFM_PERF_ITEM.WFM_PERF_ITEM_CODE **ORDER BY** WFM_BU.WFM_BU_NAME, WFM_SITE.WFM_SITE_NAME, WFM_PERF_ITEM_DAY.WFM_DATE, WFM_PERF_ITEM.WFM_PERF_ITEM_CODE
Schedule Intraday Summary for Activity	**Show query.. [+]** **SELECT** WFM_SITE.WFM_SITE_NAME, WFM_ACTIVITY.WFM_ACTIVITY_NAME, WFM_PERF_ITEM_TIMESTEP.WFM_TIME_STEP, WFM_PERF_ITEM.WFM_PERF_ITEM_CODE, **SUM**(WFM_PERF_ITEM_TIMESTEP.WFM_PERF_ITEM_VALUE) **FROM** WFM_PERF_ITEM_TIMESTEP JOIN WFM_PERF_ITEM_DAY **ON** (WFM_PERF_ITEM_DAY.WFM_PERF_ITEM_DAY_KEY = WFM_PERF_ITEM_TIMESTEP.WFM_PERF_ITEM_DAY_KEY) JOIN WFM_ACTIVITY **ON** (WFM_ACTIVITY.WFM_ACTIVITY_KEY = WFM_PERF_ITEM_DAY.WFM_ACTIVITY_KEY) JOIN WFM_SITE **ON** (WFM_SITE.WFM_SITE_KEY = WFM_ACTIVITY.WFM_SITE_KEY) JOIN WFM_PERF_ITEM **ON** (WFM_PERF_ITEM.WFM_PERF_ITEM_KEY = WFM_PERF_ITEM_DAY.WFM_PERF_ITEM_KEY) **WHERE** WFM_PERF_ITEM_DAY.WFM_DATE = ? AND WFM_PERF_ITEM.WFM_PERF_ITEM_CODE IN ('SCH_HEADCOUNT', 'SCH_SERVICE_PCT', 'FRC_CALC_SERVICE_PCT', 'SCH_IV', 'FRC_IV', 'SCH_AHT', 'FRC_AHT', 'FRC_CALC_STAFFING', 'FRC_REQ_STAFFING', 'SCH_COVERAGE', 'SCH_ASA', 'FRC_CALC_ASA', 'SCH_MAX_OCCUPANCY_PCT', 'FRC_CALC_MAX_OCCUPANCY_PCT') **GROUP BY** WFM_SITE.WFM_SITE_NAME, WFM_ACTIVITY.WFM_ACTIVITY_NAME, WFM_PERF_ITEM_TIMESTEP.WFM_TIME_STEP, WFM_PERF_ITEM.WFM_PERF_ITEM_CODE **ORDER BY** WFM_PERF_ITEM_TIMESTEP.WFM_TIME_STEP, WFM_SITE.WFM_SITE_NAME, WFM_ACTIVITY.WFM_ACTIVITY_NAME, WFM_PERF_ITEM.WFM_PERF_ITEM_CODE
Schedule Intraday Summary for Activity	**Show query.. [+]** **SELECT** WFM_SITE.WFM_SITE_NAME, WFM_ACTIVITY.WFM_ACTIVITY_NAME, WFM_PERF_ITEM_TIMESTEP.WFM_TIME_STEP, WFM_PERF_ITEM.WFM_PERF_ITEM_CODE, **SUM**(WFM_PERF_ITEM_TIMESTEP.WFM_PERF_ITEM_VALUE) **FROM** WFM_PERF_ITEM_TIMESTEP JOIN WFM_PERF_ITEM_DAY ON (WFM_PERF_ITEM_DAY.WFM_PERF_ITEM_DAY_KEY = WFM_PERF_ITEM_TIMESTEP.WFM_PERF_ITEM_DAY_KEY) JOIN WFM_ACTIVITY ON (WFM_ACTIVITY.WFM_ACTIVITY_KEY = WFM_PERF_ITEM_DAY.WFM_ACTIVITY_KEY) JOIN WFM_SITE ON (WFM_SITE.WFM_SITE_KEY = WFM_ACTIVITY.WFM_SITE_KEY) JOIN WFM_PERF_ITEM ON (WFM_PERF_ITEM.WFM_PERF_ITEM_KEY = WFM_PERF_ITEM_DAY.WFM_PERF_ITEM_KEY) **WHERE** WFM_PERF_ITEM_DAY.WFM_DATE = ? AND WFM_PERF_ITEM.WFM_PERF_ITEM_CODE IN ('SCH_HEADCOUNT', 'SCH_SERVICE_PCT', 'FRC_CALC_SERVICE_PCT', 'SCH_IV', 'FRC_IV', 'SCH_AHT', 'FRC_AHT', 'FRC_CALC_STAFFING', 'FRC_REQ_STAFFING',

	'SCH_COVERAGE', 'SCH_ASA', 'FRC_CALC_ASA', 'SCH_MAX_OCCUPANCY_PCT', 'FRC_CALC_MAX_OCCUPANCY_PCT') **GROUP BY** WFM_SITE.WFM_SITE_NAME, WFM_ACTIVITY.WFM_ACTIVITY_NAME, WFM_PERF_ITEM_TIMESTEP.WFM_TIME_STEP, WFM_PERF_ITEM.WFM_PERF_ITEM_CODE **ORDER BY** WFM_PERF_ITEM_TIMESTEP.WFM_TIME_STEP, WFM_SITE.WFM_SITE_NAME, WFM_ACTIVITY.WFM_ACTIVITY_NAME, WFM_PERF_ITEM.WFM_PERF_ITEM_CODE
Schedule Intraday Summary for Multi-Site Activity (MSA)	**Show query.. [+]** **SELECT** WFM_BU.WFM_BU_NAME, WFM_ACTIVITY.WFM_ACTIVITY_NAME, WFM_PERF_ITEM_TIMESTEP.WFM_TIME_STEP, WFM_PERF_ITEM.WFM_PERF_ITEM_CODE, **SUM**(WFM_PERF_ITEM_TIMESTEP.WFM_PERF_ITEM_VALUE) **FROM** WFM_PERF_ITEM_TIMESTEP JOIN WFM_PERF_ITEM_DAY **ON** (WFM_PERF_ITEM_DAY.WFM_PERF_ITEM_DAY_KEY = WFM_PERF_ITEM_TIMESTEP.WFM_PERF_ITEM_DAY_KEY) JOIN WFM_ACTIVITY **ON** (WFM_ACTIVITY.WFM_ACTIVITY_KEY = WFM_PERF_ITEM_DAY.WFM_ACTIVITY_KEY AND WFM_ACTIVITY.WFM_SITE_KEY IS NULL AND WFM_ACTIVITY.WFM_ACTIVITY_TYPE_KEY <> 10) JOIN WFM_BU **ON** (WFM_BU.WFM_BU_KEY = WFM_ACTIVITY.WFM_BU_KEY) JOIN WFM_PERF_ITEM **ON** (WFM_PERF_ITEM.WFM_PERF_ITEM_KEY = WFM_PERF_ITEM_DAY.WFM_PERF_ITEM_KEY) **WHERE** WFM_PERF_ITEM_DAY.WFM_DATE = ? AND WFM_PERF_ITEM.WFM_PERF_ITEM_CODE IN ('SCH_HEADCOUNT', 'SCH_SERVICE_PCT', 'FRC_CALC_SERVICE_PCT', 'SCH_IV', 'FRC_IV', 'SCH_AHT', 'FRC_AHT', 'FRC_CALC_STAFFING', 'FRC_REQ_STAFFING', 'SCH_COVERAGE', 'SCH_ASA', 'FRC_CALC_ASA', 'SCH_MAX_OCCUPANCY_PCT', 'FRC_CALC_MAX_OCCUPANCY_PCT') **GROUP BY** WFM_BU.WFM_BU_NAME, WFM_ACTIVITY.WFM_ACTIVITY_NAME, WFM_PERF_ITEM_TIMESTEP.WFM_TIME_STEP, WFM_PERF_ITEM.WFM_PERF_ITEM_CODE **ORDER BY** WFM_PERF_ITEM_TIMESTEP.WFM_TIME_STEP, WFM_BU.WFM_BU_NAME, WFM_ACTIVITY.WFM_ACTIVITY_NAME, WFM_PERF_ITEM.WFM_PERF_ITEM_CODE
Schedule Intraday Summary for Activity Group (AG)	**Show query.. [+]** **SELECT** WFM_BU.WFM_BU_NAME, WFM_ACTIVITY.WFM_ACTIVITY_NAME, WFM_PERF_ITEM_TIMESTEP.WFM_TIME_STEP, WFM_PERF_ITEM.WFM_PERF_ITEM_CODE, **SUM**(WFM_PERF_ITEM_TIMESTEP.WFM_PERF_ITEM_VALUE) **FROM** WFM_PERF_ITEM_TIMESTEP JOIN WFM_PERF_ITEM_DAY **ON** (WFM_PERF_ITEM_DAY.WFM_PERF_ITEM_DAY_KEY = WFM_PERF_ITEM_TIMESTEP.WFM_PERF_ITEM_DAY_KEY) JOIN WFM_ACTIVITY **ON** (WFM_ACTIVITY.WFM_ACTIVITY_KEY = WFM_PERF_ITEM_DAY.WFM_ACTIVITY_KEY AND WFM_ACTIVITY.WFM_SITE_KEY IS NULL AND WFM_ACTIVITY.WFM_ACTIVITY_TYPE_KEY = 10)

	JOIN WFM_BU **ON** (WFM_BU.WFM_BU_KEY = WFM_ACTIVITY.WFM_BU_KEY) JOIN WFM_PERF_ITEM **ON** (WFM_PERF_ITEM.WFM_PERF_ITEM_KEY = WFM_PERF_ITEM_DAY.WFM_PERF_ITEM_KEY) **WHERE** WFM_PERF_ITEM_DAY.WFM_DATE = ? AND WFM_PERF_ITEM.WFM_PERF_ITEM_CODE IN ('SCH_HEADCOUNT', 'SCH_SERVICE_PCT', 'FRC_CALC_SERVICE_PCT', 'SCH_IV', 'FRC_IV', 'SCH_AHT', 'FRC_AHT', 'FRC_CALC_STAFFING', 'FRC_REQ_STAFFING', 'SCH_COVERAGE', 'SCH_ASA', 'FRC_CALC_ASA', 'SCH_MAX_OCCUPANCY_PCT', 'FRC_CALC_MAX_OCCUPANCY_PCT') **GROUP BY** WFM_BU.WFM_BU_NAME, WFM_ACTIVITY.WFM_ACTIVITY_NAME, WFM_PERF_ITEM_TIMESTEP.WFM_TIME_STEP, WFM_PERF_ITEM.WFM_PERF_ITEM_CODE **ORDER BY** WFM_PERF_ITEM_TIMESTEP.WFM_TIME_STEP, WFM_BU.WFM_BU_NAME, WFM_ACTIVITY.WFM_ACTIVITY_NAME, WFM_PERF_ITEM.WFM_PERF_ITEM_CODE
Schedule Intraday Summary for Site	**Show query.. [+]** **SELECT** WFM_BU.WFM_BU_NAME, WFM_SITE.WFM_SITE_NAME, WFM_PERF_ITEM_TIMESTEP.WFM_TIME_STEP, WFM_PERF_ITEM.WFM_PERF_ITEM_CODE, **SUM**(WFM_PERF_ITEM_TIMESTEP.WFM_PERF_ITEM_VALUE) **FROM** WFM_PERF_ITEM_TIMESTEP JOIN WFM_PERF_ITEM_DAY **ON** (WFM_PERF_ITEM_DAY.WFM_PERF_ITEM_DAY_KEY = WFM_PERF_ITEM_TIMESTEP.WFM_PERF_ITEM_DAY_KEY) JOIN WFM_SITE **ON** (WFM_SITE.WFM_SITE_KEY = WFM_PERF_ITEM_DAY.WFM_SITE_KEY) JOIN WFM_BU **ON** (WFM_BU.WFM_BU_KEY = WFM_SITE.WFM_BU_KEY) JOIN WFM_PERF_ITEM **ON** (WFM_PERF_ITEM.WFM_PERF_ITEM_KEY = WFM_PERF_ITEM_DAY.WFM_PERF_ITEM_KEY) **WHERE** WFM_PERF_ITEM_DAY.WFM_DATE = ? AND WFM_PERF_ITEM.WFM_PERF_ITEM_CODE IN ('SCH_HEADCOUNT', 'SCH_SERVICE_PCT', 'FRC_CALC_SERVICE_PCT', 'SCH_IV', 'FRC_IV', 'SCH_AHT', 'FRC_AHT', 'FRC_CALC_STAFFING', 'FRC_REQ_STAFFING', 'SCH_COVERAGE', 'SCH_ASA', 'FRC_CALC_ASA', 'SCH_MAX_OCCUPANCY_PCT', 'FRC_CALC_MAX_OCCUPANCY_PCT') **GROUP BY** WFM_BU.WFM_BU_NAME, WFM_SITE.WFM_SITE_NAME, WFM_PERF_ITEM_TIMESTEP.WFM_TIME_STEP, WFM_PERF_ITEM.WFM_PERF_ITEM_CODE **ORDER BY** WFM_PERF_ITEM_TIMESTEP.WFM_TIME_STEP, WFM_BU.WFM_BU_NAME, WFM_SITE.WFM_SITE_NAME, WFM_PERF_ITEM.WFM_PERF_ITEM_CODE
Contact Center Performance Report for Activity	**Show query.. [+]** **SELECT** WFM_TIME_STEP, WFM_SITE.WFM_SITE_NAME, WFM_ACTIVITY.WFM_ACTIVITY_NAME, WFM_PERF_ITEM.WFM_PERF_ITEM_CODE, **SUM**(WFM_PERF_ITEM_TIMESTEP.WFM_PERF_ITEM_VALUE)

```
FROM WFM_PERF_ITEM_TIMESTEP
JOIN WFM_PERF_ITEM_DAY ON
(WFM_PERF_ITEM_DAY.WFM_PERF_ITEM_DAY_KEY =
WFM_PERF_ITEM_TIMESTEP.WFM_PERF_ITEM_DAY_KEY)
JOIN WFM_ACTIVITY ON (WFM_ACTIVITY.WFM_ACTIVITY_KEY =
WFM_PERF_ITEM_DAY.WFM_ACTIVITY_KEY)
JOIN WFM_SITE ON (WFM_SITE.WFM_SITE_KEY =
WFM_ACTIVITY.WFM_SITE_KEY)
JOIN WFM_PERF_ITEM ON (WFM_PERF_ITEM.WFM_PERF_ITEM_KEY
= WFM_PERF_ITEM_DAY.WFM_PERF_ITEM_KEY)
WHERE
WFM_TIME_STEP >= ? AND WFM_TIME_STEP < ? AND
WFM_PERF_ITEM.WFM_PERF_ITEM_CODE IN ('ACT_IV',
'ACT_ABANDONED_IV_PCT', 'ACT_AHT', 'ACT_ASA',
'ACT_IV', 'ACT_SERVICE_PCT')
GROUP BY
WFM_SITE.WFM_SITE_NAME,
WFM_ACTIVITY.WFM_ACTIVITY_NAME,
WFM_PERF_ITEM.WFM_PERF_ITEM_CODE,
WFM_PERF_ITEM_TIMESTEP.WFM_TIME_STEP
ORDER BY
WFM_SITE.WFM_SITE_NAME,
WFM_ACTIVITY.WFM_ACTIVITY_NAME,
WFM_PERF_ITEM_TIMESTEP.WFM_TIME_STEP,
WFM_PERF_ITEM.WFM_PERF_ITEM_CODE
```

Recording, Quality Management and Speech Analytics

> **Important**
>
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Recording, Quality Management, and Speech Analytics.

The Genesys Recording, Quality Management and Speech Analytics solution leverages recorded customer interactions for review and analysis of critical business issues.

Designed to provide valuable voice interaction information in an intuitive and easy to use user interface, the solution allows you to uncovers valuable insights about workforce performance and the customer experience that may be hidden within the agent-customer interactions your organization records. It provides insight into the cause and effect relationships that influence business issues and contact center performance.

In addition, with the optional Quality Management (QM) add-on product, specific agent training requirements, compliance breaches or customer satisfaction issues can be assessed on a regular basis to improve agent performance and customer service delivery.

With the Speech Analytics add-on, you can analyze 100% of recordings to uncover why customers are contacting your company, what are their topics of conversation, why multiple contacts are needed to resolve specific issues, what processes cause customer frustration and whether your agents are providing an appropriate level of service.

> **Important**
>
> SpeechMiner UI provides a single user interface (UI) across different products within the Genesys Workforce Optimization suite in Genesys Engage cloud, including Interaction Recording, Quality Management and Speech Analytics, each product is sold separately. Interaction Recording is a pre-requisite, however, Quality Management and Speech Analytics can be added based on the specific needs of your business.
>
> Screen recording is available only for voice interactions. Screen recording of chat and email interactions is not available.
>
> The Genesys Recording, QM and Speech Analytics solution does not include:
>
> - Text Analytics
> - Access to SMART for Topics/Category creation

- Export of transcripts or Analytics data
- Distinct retention periods for Analytics and QM data; the retention period for all data is tied to the retention period of the underlying call recording

For additional information about the Genesys recording solution refer to:

Recording, Quality Management and Speech Analytics (SpeechMiner UI): The Recording, QM and Speech Analytics solution evaluates recorded customer interactions for data about what is happening in your organization, and uncovers the cause and effect relationships that influence business issues and contact center performance.

Recording Certificate Management: The Genesys recording solution enables proper management of public and private keys used to encrypt voice and screen recordings.

Access control for Recording users: The Genesys recording solution applies access control to recordings.

Deploying the Screen Recording Service: The Screen Recording Service (SR Service) enables the Agent to capture what is happening on the screen at the time of an active voice interaction.

Recording Cloud Backup Service: The Recording Cloud Backup Service (RCBS) allows you to make a backup copy of your Genesys Interaction Recording voice files (some or all) prior to their automated deletion as per the Cloud retention policy.

Recording certificate management

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Recording certificate management.

The Genesys recording solution requires proper management of public and private keys used to encrypt voice and screen recordings. The public key is stored in a certificate file and is used to encrypt a unique session key that is then used to encrypt each media file. The public key must be provisioned for voice recordings and for screen recordings. The private key is stored securely on a protected server and is used to help decrypt each media file (voice or screen) for playback.

This document describes the process for generating and installing the public and private keys.

> **Important**
> It is your responsibility to store your private keys and certificates, including the expired ones. Genesys will not be able to re-apply any of your keys or certificates in the event of a catastrophic Genesys site failure. In this case, you will need to re-apply any previously created keys. Therefore these keys must be kept somewhere safe and reliable for future use. Furthermore, if you will be taking advantage of the Recording Cloud Backup Service utility, then, in order to listen to those recordings once they have been moved to your own site, you will need to have the keys and certificates available. Please contact your Genesys Professional if you have any questions about this very important note

Certificate requirements

Before you can encrypt certificates for voice and screen recordings, you must generate the following keys and certificates:

- Generate a recording private key in `.pem` format.
  2048-bit RSA (or higher). Align encryption strength requirements with your IT Security.
- Generate a self-signed recording certificate (also known as a public key) in `.pem X.509 RSA` format.
  The certificate validity period will determine when the next certificate should be generated for renewal.
  Note: It is the customer's responsibility to track this, install a new key and certificate prior to expiry and contact Genesys to help provision the new key and certificate before the expiration date.

Generating a self-signed certificate key pair using OpenSSL

> **Important**
>
> To generate a self-signed certificate key pair using OpenSSL you must have access to a Linux/Windows system with OpenSSL installed.

The following OpenSSL commands generate a private key and then use the private key to generate a self-signed certificate:

```
openssl genrsa -out tenant.key 2048

openssl req -new -x509 -key tenant.key -out tenant.pem -days validity_period -subj
"/CN=common_name/C=Country/ST=State/O=Organization"
```

Example:

```
openssl genrsa -out tenant.key 2048

openssl req -new -x509 -key tenant.key -out tenant.pem -days 3650 -subj "/CN=Genesys
Recording/C=US/ST=California/O=Genesys"
```

Refer to the following table for DN field descriptions and sample values:

DN Field	Description	Example
Common Name (CN)	The name of your recording solution.	Genesys Recording
Country (C)	A two letter country code.	US
State or Province (ST)	The full state or province where your organization is legally located.	California
Organization (O)	The exact legal name of your organization. Do not abbreviate your organization name.	Genesys

As a result of this command, the following two files are created:

- tenant.key— the private key (PEM format) that is used to decrypt the recordings. It must be kept safe and should not be shared.
- tenant.pem— a self-signed recording certificate and the public key.

Uploading/installing certificates to encrypt your voice recordings

> **Important**
> The following steps describe how to configure encryption for voice recordings and should be performed by an administrator.

Upload recording certificates

1. Verify that you have Administrator privileges.

   > **Important**
   > The Platform Administration section of the Genesys Portal is the tool that should be used to manage recording certificates (public keys), and private keys.

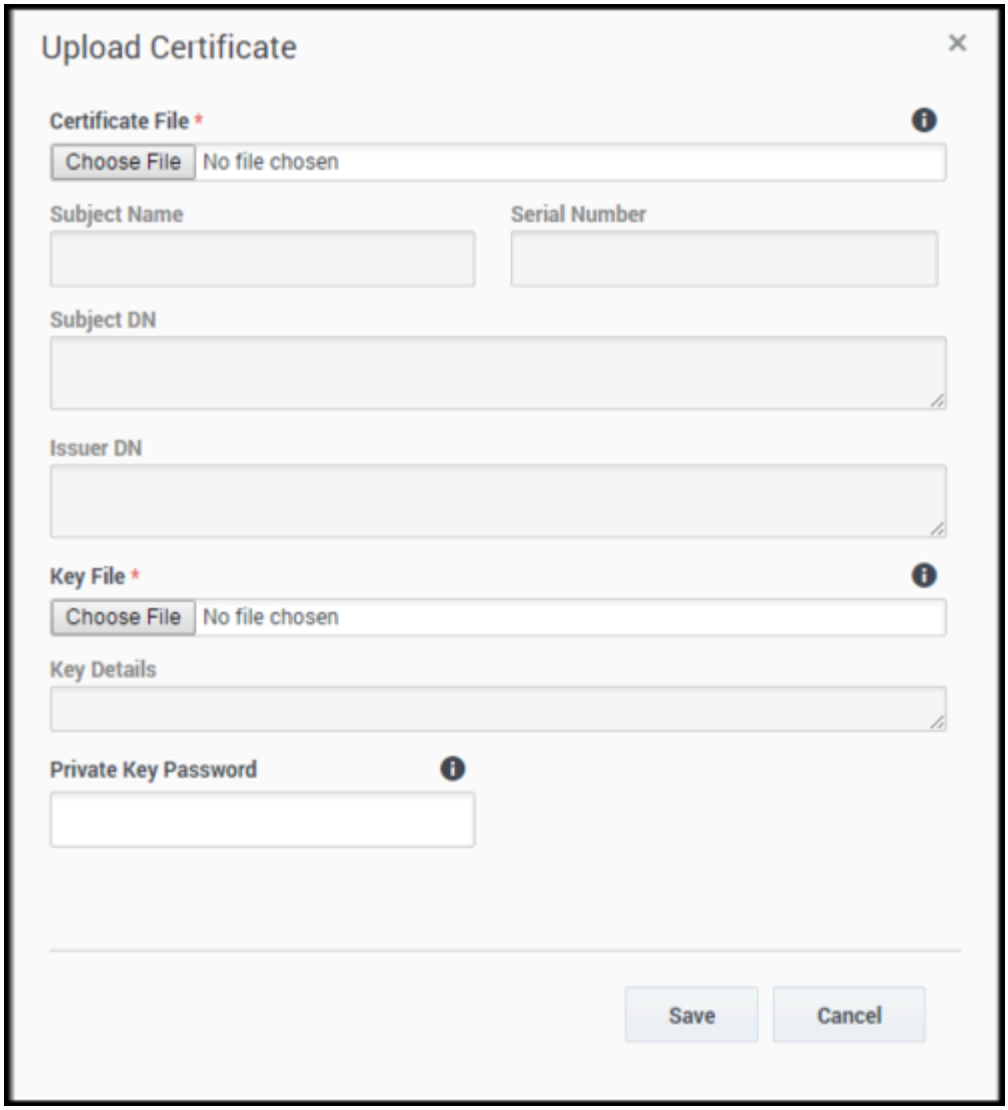
2. Select **Administration > Certificates**.

   The **Recording Certificates** screen displays the list of defined Recording Certificates. To refresh the list at any time, click  .

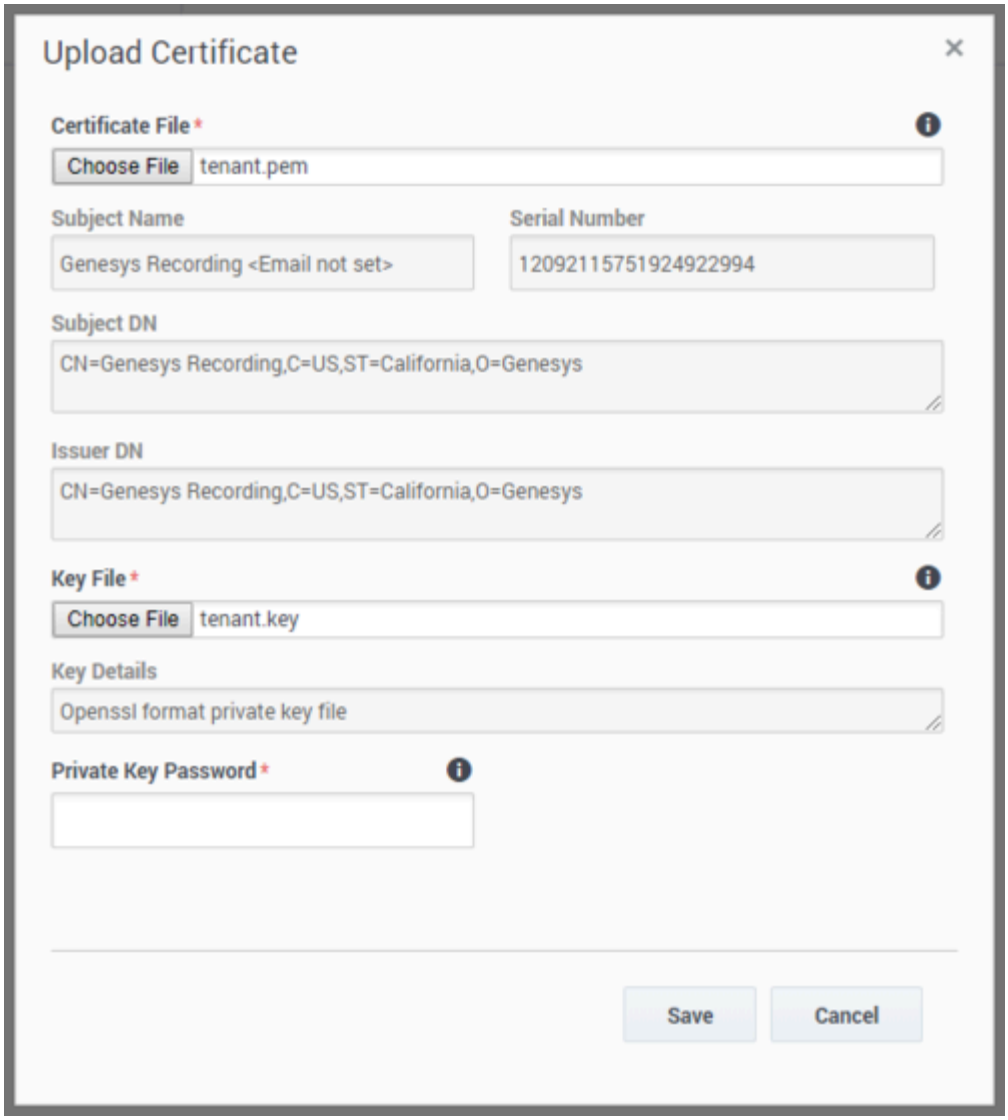
   

   > **Tip**
   > Click a recording certificate in the list to display its details.

3. Click **Upload**.



4. In the **Upload Certificate** panel, under **Certificate File**, click **Choose File**.
5. Select the recording certificate file (PEM file).
   The **Subject Name**, **Serial Number**, **Subject DN**, and **Issuer DN** fields automatically populate.
6. In the **Key File** section, click **Choose File**.
7. Select the private key file.



8. Leave the **Private Key Password** field empty.
9. Click **Save**. Both public and private keys are stored in a secure keystore file dedicated to your tenant.
10. After uploading the self-signed recording certificate you must contact your Genesys Professional and ask them to have the certificate assigned to your IVR profile. You will be asked for a copy of the Self-Signed Recording Certificate. You do not need to provide them your private key.

Important

- If you upload and/or delete recording certificates in one Platform Administration session, these changes are not reflected in another Platform Administration session. You must log out and log in again to the second Platform Administration session.

- In the Certificate Administration section, there is an option to Delete certificates. Do not delete any certificates without first discussing this with your Genesys Professional, since there may be adverse side-effects of doing this (for example, not being able to playback recordings). Even if a certificate is expired, it will need to remain in the system so that older recordings can be played back.

Provisioning certificates for screen recording

Perform the following steps to configure encryption for screen recordings, only after completing the Uploading/installing certificates to encrypt your voice recordings procedure. If you have not purchased screen recording services, you may skip this step.

Screen Recording Certificates list

The Screen Recording Certificates page enables you to manage the certificates for screen recording encryption.

- Assign new certificates
- Remove certificates

Assigning Screen Recording Certificates

To assign a new certificate:

1. In the header, go to **Administration > Screen Recording Certificates**.
2. On the **Screen Recording Certificates** panel, click **Add**.
3. From the **Select Certificate** window, perform one of the following actions:
   - Select the check box next to the appropriate certificate, and click **Add**.
   - Click **Cancel** to discard any changes.
4. Perform one of the following actions:
   - Click the **Save** button to accept the changes.

- Click the **Cancel** button to discard the changes.

Removing Screen Recording Certificates

To remove a Recording Certificate, perform the following actions:

1. In the header, go to **Administration > Screen Recording Certificates**.
2. On the **Screen Recording Certificates** panel, select the check box next to the certificate that you want to remove.
3. Click **Remove**.
4. Perform one of the following actions:
   - Click the **Save** button to accept the changes.
   - Click the **Cancel** button to discard the changes.

Important

- If you remove a certificate from the Screen Recording Certificates, you will turn off encryption, and screen recordings will no longer be encrypted. Do not remove any certificates without first discussing this with your Genesys Professional.
- The modifications to encryption settings described in this document only affect future recordings and do not change the encrypted status of old recordings.

Access control for Recording users

> **Important**
>
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Access control for Recording users.

Each recording file is considered an object that is subject to access control at the user level. When a recording file is generated, the access control for the recording file is set based on the following criteria:

1. Access control is set based on the agent that was recorded. Agents are organized as an agent hierarchy; for example, the hierarchy can be a reporting structure in an organization.

   Note: With IVR recording, there is no associated agent for the specific segment of the call, since IVR is not a user.

2. Access control is set based on partitions. Partitions are set as a specific attached data in a call, and the attached data is typically set by a routing strategy.

To search and playback a recording file that is subject to access control, the user accessing the Recording application must be assigned to the appropriate Access Groups to access the recordings. If the user accessing the application is an agent, they are granted implicit playback access to their own recordings.

Agent hierarchy

The agent hierarchy shows how the agents are organized in the hierarchy, and the hierarchy is represented as a field configured within Agent Setup. For more information, see the description of the Recording option in How do I add a single agent.

The following example shows the agent hierarchy with four agents:

- /
  - Anthony
    - John
      - Agent1
      - Agent2
    - Paul

- Agent3
- Agent4

`Agent1` and `Agent2` are on John's team. John reports to Anthony.

To represent this structure, the following fields are configured in each agent:

Agent	agent_hierarchy
Agent1	/Anthony/John
Agent2	/Anthony/John
Agent3	/Anthony/Paul
Agent4	/Anthony/Paul

> **Important**
> When there are agents specified in the path, the path must contain the username for those agents. For example, for the hierarchy /Anthony and /Anthony/John, Anthony and John must match the usernames for Anthony and John.

If a user wants to listen to recordings handled by `Agent1`, the user needs to be granted access to either the `Anthony`, or the `John` Access Group. If a user is granted access to the `Anthony` Access Group, that user has access to recordings from all four agents, because all four agents are within Anthony's hierarchy.

Partitions

Partitions are arbitrary names that allows a contact center to partition recordings based on business rules. For example, partitions can be business groups such as sales, support, marketing, etc. To set one or more partitions to a recording, attach data to the call with the `GRECORD_PARTITIONS` key with a comma-separated list of partition names.

For example, if the `GRECORD_PARTITIONS` key is set to `/sales,/support`, the recording belongs to the `/sales` partition as well as the `/support` partition.

To access any recording belonging to a partition, the user must be assigned to an Access Group with the same name. For example, if `user1` is assigned to the `/sales` Access Group, `user1` can search and playback any recordings within the `/sales` partition.

Access groups

All access groups for recording purposes must be created within the **Recording** folder. The access group "/" grants access to all recordings. For more information on configuring access groups, see Access Groups.

Deploying the Screen Recording Service

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Deploying the Screen Recording Service.

Genesys Interaction Recording (GIR) requires that a Screen Recording Service (SRS) be installed on each Agent's desktop to enable the Agent to capture what is happening on the screen at the time of an active interaction.

The procedures on this page show how to download, install, configure and test the Screen Recording Service.

> **Important**
> In a Genesys Engage cloud deployment, screen recording is available only for voice interactions. Screen recording of chat and email interactions is not available in a Genesys Engage cloud deployment.

Prerequisites

The following list provides you with the requirements you need to successfully deploy the Screen Recording Service (SRS):

- Before you can install and use the SR Service on your desktop, you must have the following information ready at hand. Your IT department or Genesys Professional can help you get this information.
  - Access to Agent Desktop
  - The software (minimum version 8.5.302.10)
  - Agent Desktop V9 is supported starting with SRS version 8.5.355.61
- Verify that the client machine meets the following minimum specifications:
  - Pentium Dual Core CPU
  - 2 GB RAM (800 MB available for the SR Service)
  - A minimum of 5 GB of available space (in total) for the SR Service installation and working space.

- If you are running Bria 4 on Windows 7, you must enable Windows Aero. If you do not enable Windows Aero, the Screen Recording Service may fail to capture the Bria 4 application.
- Verify the client machine is synchronized with an NTP server — for example, www.time.gov.

Installation considerations

After verifying that your system meets the basic prerequisites, you should consider the following:

- The recommended installation procedure will install the Screen Recording Service's self-signed PFX certificates to the root certificates store.

- When required use one of the following options to query the SR Service version:
  - Run the following command line `wmic datafile where name='C:\\<Installation Directory>\\GenesysServiceHandler.exe`.
  - Open the web browser and navigate to https://127.0.0.1/version if the SR Service is deployed with HTTPS enabled or http://127.0.0.1:8080/version if the SR Service is running as HTTP.
- Proxy support for outbound connections from SRS can be enabled either with or without authentication support.
  - The parameters used to configure the SRS Proxy are available in Advanced configuration for the Screen Recording Service.
- When a proxy is used it may interfere with the SR Service operation. The SR Service runs as an HTTP server and relies on an incoming socket connection to correctly identify the agent's windows session. If the HTTP requests are forwarded by a proxy, the SR Service may not be able to correctly identify the user session in a multi-user environment. With a single user, the SR Service will rollback to the currently active windows session.
  When a proxy is used it is recommended that localhost (127.0.0.1) connections be excluded from the proxy settings.
  When the proxy is an internal system service (like an Antivirus\Firewall), it is recommended that the SRS related processes (SrsProcess.exe and GenesysServiceHandler.exe) be added to the security software exception\white list.

- The Screen Recording Service can be used by a Citrix client. The following Citrix configurations are supported:
  - Citrix XenApp 6.x running under Windows Server 2008 R2 or Windows Server 2012 R2
  - Citrix XenApp 7.x or Citrix XenDesktop 7.x running under Windows Server 2008 R2 or Windows Server 2012 R2
- In a Citrix environment (for Genesys SR Service 8.5.230.23 and later), SRS only supports a single session per remote PC (Session Sharing is not supported).
- In a Citrix environment (for Genesys SR Service 8.5.370.85 and later), SRS can be configured to work with Citrix's Virtual Loopback feature.
  - Configure the `authenticationHost` parameter so that SRS uses a loopback IP address that is outside of the range being used by the Citrix Virtual Loopback Feature. See Advanced Configuration for the Screen Recording Service for more details on how to configure the

`authenticationHost` parameter.

- If the IPv4 SRS `authenticationHost` parameter is configured to something other than 127.0.0.1, then use that IP address instead of 127.0.0.1 in the above URLs. See Advanced Configuration for the Screen Recording Service for more details.
- If the IPv4 SRS `authenticationHost` parameter is configured to something other than 127.0.0.1, and SRS is configured to use HTTPS, then use that IP address when creating self-signed certificates. See Creating Self-Signed Certificates to support IP Loopback Addresses other than 127.0.0.1 for more details.
- The SR Service can be used in a VMware Horizon environment. The following VMware Horizon configuration is supported:
  - VMware Horizon 7 running under Windows Server 2012 R2

Screen Recording Service - operating systems

The Screen Recording Service is supported on the following operating systems in a non-Citrix mode:

- Windows 7 (32 and 64 bit)
- Windows 8, 8.1 (32 and 64 bit)
- Windows 10 (32 and 64 bit)

The Screen Recording Service is supported on the following operating systems for Citrix support:

- Windows Server 2008 R2
- Windows Server 2012 R2

The Screen Recording Service is supported on the following operating system for VMware Horizon support:

- Windows Server 2012 R2

Recommended screen resolutions

Genesys has tested the Screen Recording Service under the following recommended screen resolutions. If you use the Screen Recording Service on a computer with different screen resolution than listed above, you should do a field validation of the Screen Recording Service in your setup to ensure that it is working properly. If there you encounter unexpected results, Genesys recommends that you set your screen resolution to one of the recommended and tested resolutions listed below.

> **Warning**
> If a field validation has been completed against an earlier version using a non-

> supported resolution, there is no guarantee that resolution will continue to work on upgrades to new releases. Only supported resolutions are continually tested against each new version.

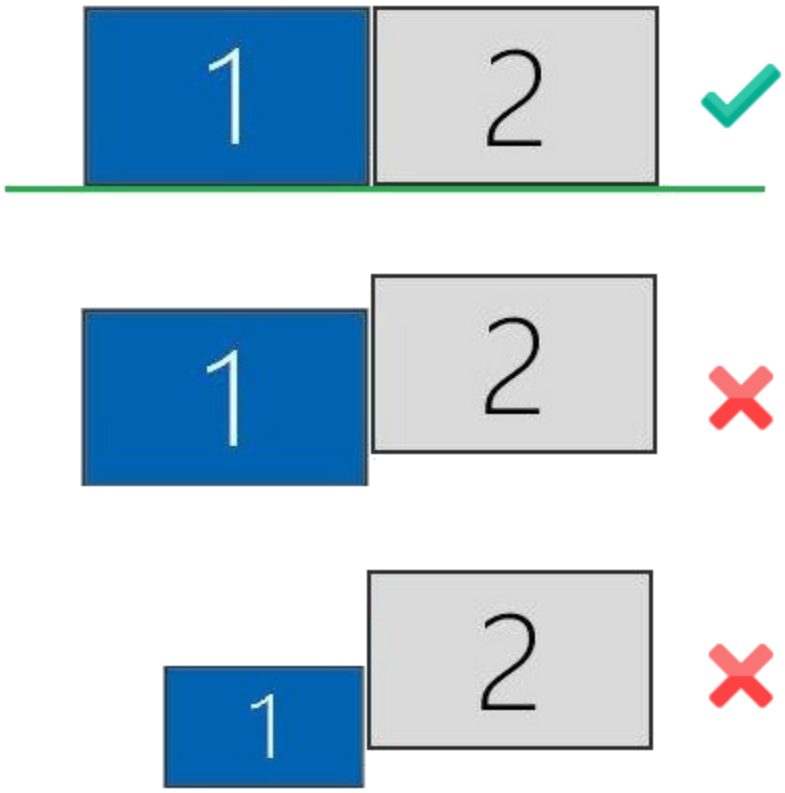
Single Monitor:

- 1024 x 768
- 1280 x 720
- 1600 x 1200
- 1920 x 1080

Dual Monitor:

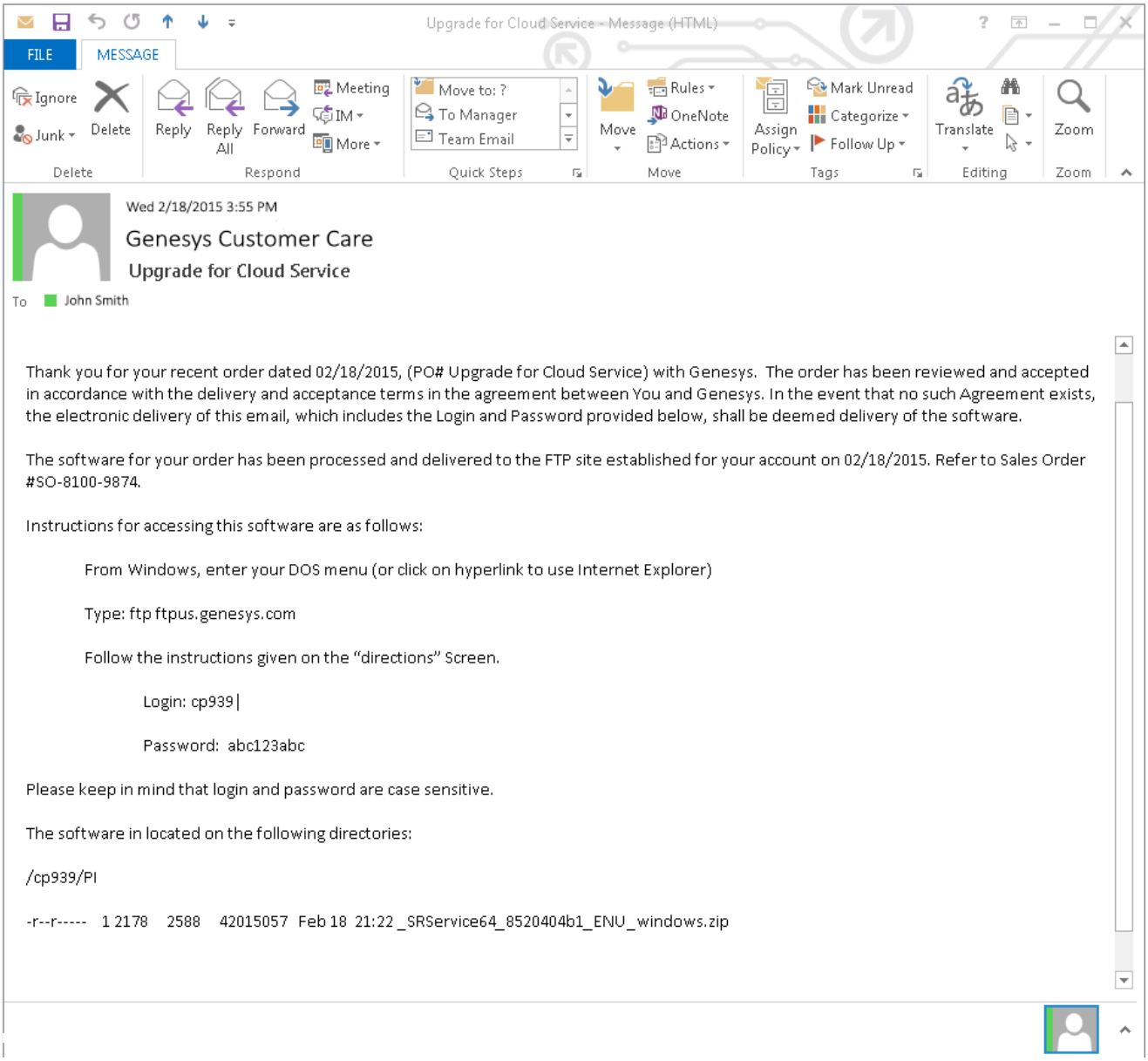
- Side-by-side 1024 x 768 + 1024 x 768
- Side-by-side 1280 x 720 + 1280 x 720
- Side-by-side 1600 x 1200 + 1600 x 1200
- Side-by-side 1920 x 1080 + 1920 x 1080

When using dual monitors, set both displays to the same resolution and arrange them side-by-side (*not* offset) in your display settings, as shown here:



Using dual monitors in a non-recommended configuration can result in errors.

Get your software



Find the email you received from Genesys with the details about your software (it will look similar to the example above), and using your favorite FTP client—for example, Filezilla, connect with the credentials listed in the email.

Download the zipped file to a temporary folder on your computer.

Installing your software for the first time

There are two ways to install the SR Service:

- Installation Wizard - for version 8.5.3 and higher
- Command Prompt

> **Important**
>
> - To install the SR Service you must have Administrator privileges.
> - Firefox users must close the browser before installing the SR Service. If Firefox is open while the SR Service is being installed, restart the browser after the installation is completed.

Installing the SR Service for the first time with the installation wizard

This type of installation procedure is for version 8.5.3 and higher. Ensure you follow the specified steps so that the SR Service will work in a Genesys Engage cloud deployment.

1. Locate the setup.exe and double-click its icon. The installation wizard is activated.
2. Select the Standard option and click **Next**.
3. Select **Use an existing configuration file** (optional) to copy the configuration of one machine, to all other installations of the SR Service on different machines in the same deployment. In the **Location** field, enter the location of the existing configuration file and click **Next**.
4. Verify that the location in the **Destination Folder**, is the correct location (that is, the location where the SR Service will be installed) for the SR Service. If it is not the correct location, enter the correct location and click **Next**.
5. Click **Install**, to complete the first time installation.

Installing the SR Service for the first time with the command prompt

1. Open a command prompt, and type cd to change directories to the installation folder.
2. At the prompt, enter the following command and press **Enter**:

```
setup.exe /s /z"-s '<C:\genesys_silent.ini>' -sl '<setup log file name>' -t '<setup
 wizard log file name>'"
```

Important

- Set the configured `genesys_silent.ini` file path in the command line. Use the absolute path for the input file parameters.
  For example, run `setup.exe /s /z"-s'c:\genesys_silent.ini' -sl 'c:\setup.log' -t'c:\setup_wizard.log'"`
- The `genesys_silent.ini` file must be configured when using command line silent installation and an unused parameter must be commented out in the `genesys_silent.ini` file. The standard **genesys_silent.ini** file is included with the installation package.
  - The **genesys_silent.ini** file provides all possible configuration parameters along with a description of each.
  - The file lists all the parameters with placeholders.
  - Verify that the unused configuration parameters are either deleted or commented.
  - Verify that the configuration file contains at least the following parameters:
    ```
    [SRServer]
    InstallationType=Standard
    [IPCommon]
    InstallPath=<Absolute path where the SR Service needs to be installed>
    [MaintMode]
    Mode=FirstInstall
    ```
  - For additional security options, consult a Genesys Professional.
  - During the installation process, the antivirus program may block the installation when the installation process detects that the antivirus program is attempting to make system changes. In this scenario, the user will have to unblock the installation program to continue the installation.

Verify the installation

Use Windows Explorer to locate the directory where you installed the software. For example, `C:\Program Files (x86)\GCTI\Genesys SR Service \Logs\GSR`. Once you see the folder is there, restart your computer to confirm that the service starts automatically.

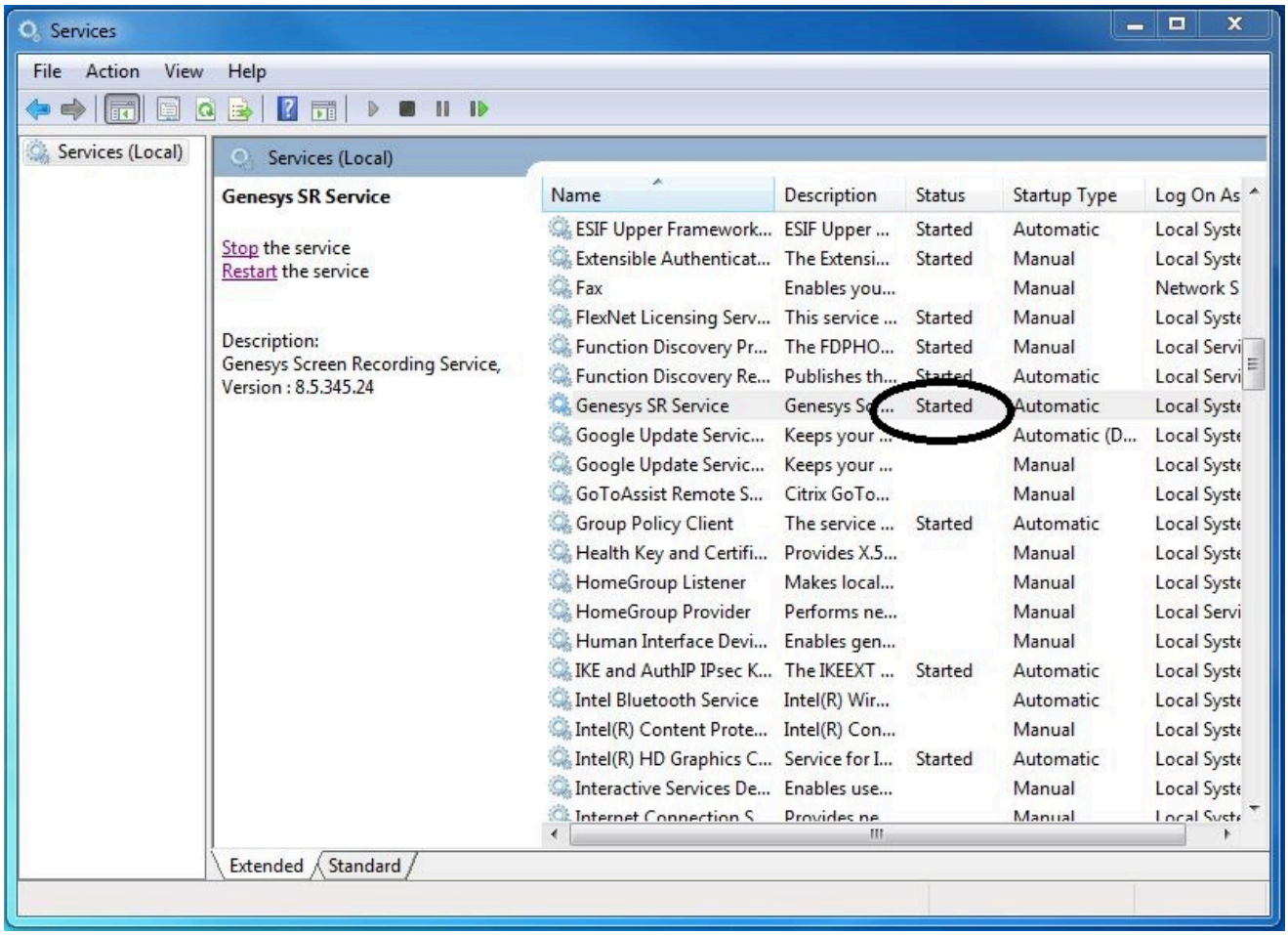
To verify the version installed, browse to https://127.0.0.1/version or http://127.0.0.1:8080/version.

Test the service and validate the installation

After installation, use Windows Services to confirm that the Genesys SR Service is ‘Started’. Check the startup log file as follows:

1. Open the `C:\Program Files (x86)\GCTI\Genesys SR Service \Logs\GSR.log` file, and make sure that something similar to the following lines are included (with the version reflecting the version you have just installed):
   `ServiceHandler: Running Version:8.5.230.23, IP:135.39.66.17, OS:win32`
2. Make sure that the C:\Program Files (x86)\GCTI\Genesys SR Service \Logs\GSR.log file contains no errors or exceptions.
3. Use the agent desktop to login as an agent that has been configured to have their voice interactions recorded. When the **`recordingWhen`** parameter is not set to `off`, the screens will also be recorded when the Screen Recording Service is running. Once logged-in as an agent, request an inbound call to that agent, or use the agent desktop to initiate an outbound call (For example, to a cell phone). Keep the interaction active for 10-20 seconds, and then disconnect the call. Proceed with step 4 to review the log file.
4. After the test, review the C:\Program Files (x86)\GCTI\Genesys SR Service \Logs\GSR.log for the following line: `Uploader: Upload of file <file-name-of-media> was successful`.

> **Tip**
>
> If your installation is unsuccessful, contact your Genesys Professional.



Advanced Installation Procedures

Creating Self-Signed Certificates to support IP Loopback Addresses other than 127.0.0.1

SRS can be configured so that its Authentication Server uses Loopback IP Addresses other than 127.0.0.1. The HTTPS Certificates that are created by default only work if SRS is configured to use the Loopback IP Address 127.0.0.1. To use SRS with Loopback Addresses besides 127.0.0.1 and HTTPS, new HTTPS Certificates must be created specifically for the Loopback IP Address that SRS is using.

To create self-signed certificates with Loopback addresses other than 127.0.0.1, following installation, perform the following:

1. Open a command window as an Administrator.
2. Navigate to the **<install_dir>\Certificates\Self-Signed** directory.

3. Run **uninstall_certificates.bat** to remove the existing certificates.
4. Run **create_certificates.bat** and pass a value for the **IPV4_HOST** parameter. Below is an example to create certificates for 127.1.1.2:

   ```
   create_certificates.bat -IPV4_HOST 127.1.1.2
   ```

5. Run **install_certificates.bat** to install the new certificates. This installs the new self-signed certificates to the Windows trusted certificates store.
6. Configure SRS to use the newly created certificates. Please see the **authenticationCertificate** option in Advanced Configuration for the Screen Recording Service for more details.
7. Restart the Genesys SR Service Windows service.

Upgrading the Screen Recording Service

Screen Recording Service can be upgraded manually or automatically. Both types of upgrades assume a functional existing deployment of Screen Recording Service. If the functionality of the existing deployment is in question, it is recommended to look for and stop the service, delete the previous installation folder and proceed as though this is the first time deploying the software. Contact your Genesys Professional if you are not sure if the software is working.

Manual upgrade from any version to 8.5.302.10

1. Create a backup copy of the `C:\Genesys\SRC` directory and name the backup directory `C:\Genesys\SRC.backup.`
2. Unzip your new software in a temporary directory (for example, C:\temp).
3. Update the **`.ini`** file. Access the temporary directory and type the following command in a command prompt window:

   ```
   setup.exe /s /z"-s '<genesys_silent.ini>' -sl '<setup log file name>' -t '<setup wizard log file name>'"
   ```

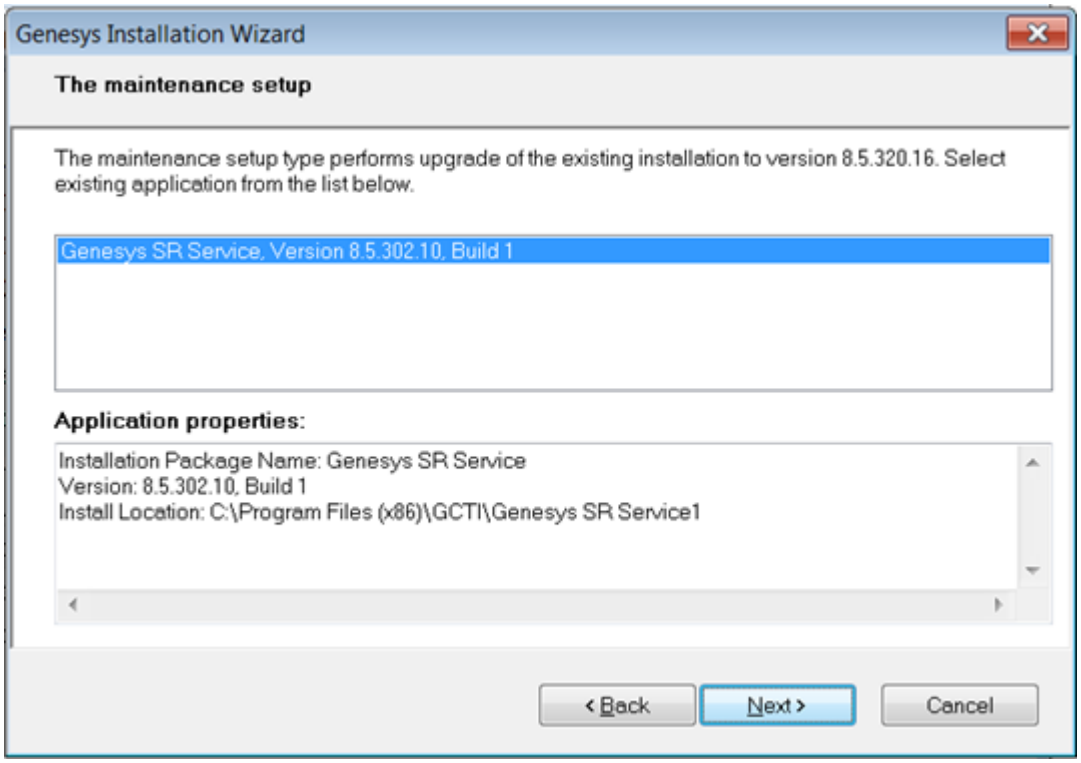
4. Validate the upgrade using the steps in the Test the Service and Validate the Installation section above.

Manual Upgrade

> ### Important
>
> - The following steps must be performed by a System Administrator.
> - Before you upgrade to a newer Screen Recording Service version, check with your Genesys Professional about compatibility with your system.

1. Copy the new SR Service software to a temporary directory.

2. Run the **setup.exe**. As shown in the following image, the setup process automatically detects the existing SR Service installation and selects it for upgradation.



3. Click **Next** and follow the instructions provided in the Installing the SR Service for the first time with the installation wizard section above.
4. Validate the upgrade using the steps in the Test the Service and Validate the Installation section.

Upgrading while using HTTPS with an IP address other than 127.0.0.1

When SR Service is upgraded, the self-signed HTTPS certificates are removed and new ones are generated and installed. The newly generated HTTPS certificates will be for the IP address 127.0.0.1. If the IPv4 SRS `authenticationHost` parameter (see Advanced Configuration for the Screen Recording Service for more details about the `authenticationHost` parameter) is configured to something other than 127.0.0.1, then the HTTPS certificates will not work.

To continue using HTTPS with an IP address other than 127.0.0.1, new HTTPS certificates must be generated. Follow the instructions in Creating Self-Signed Certificates to support Virtual Loopback Addresses to create and install new HTTPS certificates.

Advanced configuration for the Screen Recording Service

These parameters can be configured locally in the **config.json** file present in the SRS installation directory. All the configuration parameter values should be in JSON notation, for example: for

example: {"name":"parameterName","value":"parameterValue"}.

> **Important**
> Screen Recording Service does not support the use of System Proxies configured via PAC (Proxy Auto-Configuration) files.

Name	Mandatory	Description	Default value
authenticationHost	N	The IPv4 Address that the Authentication Server will bind to when SRS starts if SRS is configured to use IPv4. The parameter value must be an IPv4 address within 127.0.0.0/8. The IP addresses 127.0.0.0 and 127.255.255.255 are not allowed.	127.0.0.1
dummyRecordingDuration	N	When enabled, launches a VLC process for the specified duration during the start of the service. The parameter is disabled when a negative value is given. This parameter can be set to a value less than 60 (seconds). **Warning** Only configure this parameter if instructed by Genesys.	-1
preLoadVlc	N	Decides whether to load VLC process in advance after agent logs in. Valid values are `true` and `false`. **Warning** Only configure this parameter if instructed by Genesys.	false
proxyServerHost	N	The hostname of the proxy server.	Empty
proxyServerPort	N	The server port of the proxy server.	Empty
proxyServerUsername	N	The username to connect to the proxy	Empty

Name	Mandatory	Description	Default value
		server.	
proxyServerPassword	N	The password to connect to the proxy server.	Empty
useSystemProxy	N	If this value is `true`, the Screen Recording Service uses the Windows System Proxy settings.	false

Important

If specified, the proxy server parameters take precedence over the **useSystemProxy** parameter.

Rollback to a previous version

To rollback to a previous version of the Screen Recording Service:

Important

- The SR Service only supports a manual rollback.
- Recordings captured but not uploaded will need to be manually moved to the upload folder of the active SRS directory after the rollback is complete.

1. In the Task Manager, verify that **Genesys SR Service** is stopped. If it has not been stopped, stop it now.
2. Copy the current `C:\Program Files (x86)\GCTI\Genesys SR Service` directory to a different folder. (For example: C:\Program Files (x86)\GCTI\Genesys SR Service.<date>). This directory contains recordings that have not yet been uploaded; it may be needed for subsequent troubleshooting purposes.
3. Uninstall the existing SR Service installation.
4. Install the previous SR Service version.
5. Restart your computer or start the Genesys SR Service Windows service.
6. Validate the rollback using the steps in the Verify the Installation section above.

Uninstalling the Screen Recording Service

1. Open the **Start** menu and select **Control Panel**.
2. Click **Programs and Features**.
3. In the **Name** column, select the **Screen Recording Service** entry (for example, Genesys SR Service 8.5.xxx.yy), right click and select **Uninstall**.

   The Screen Recording Service is uninstalled.

Recording Cloud Backup Service

> **Important**
>
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Recording Cloud Backup Service.

The Recording Cloud Backup Service (RCBS) allows you to make a backup copy of your Genesys Interaction Recording voice and/or screen recording files prior to their automated deletion as per the Cloud retention policy. Once installed, you can securely download the encrypted voice and screen recording files from Amazon S3 and their respective metadata files from Genesys Engage and store them on your machines.

RCBS can be installed on local machines or on AWS EC2 instances. The recording file can then be decrypted and used as desired, for example, for compliance.

> **Important**
>
> - Unless backed up, all recordings will be deleted when maximum retention date is reached.
> - RCBS will only work with encrypted recordings. Therefore, ensure encryption is enabled.
> - RCBS does not support MPLS.

Prerequisites

Before you can install and use the Recording Cloud Backup Service on a machine, verify that you have the following prerequisites. Your IT department or your Genesys professional can help you get this information.

- Windows Server 2008/2012 64-bit or Red Hat Enterprise Linux AS 6 Operating System with admin privileges.
- 4 GB RAM, minimum 20 GB hard drive (the amount of space required depends on the number of recordings to be downloaded).
- The Recording Cloud Backup Service software (minimum version 8.5.2xx.xx).
- The target directory or shared folder in your environment to download the recording files to—for example, `C:/target_directory` (this is for the **targetDir** parameter).

- The private key you used to initially configure recording file encryption, so that the recording files can be decrypted (this is for decrypting the downloaded files).
- The name of your Platform Administration tenant administrator account – (this is for the `GWS_USERNAME` environment variable). Usernames should be in the format `username@customer_tenant.com`. For more information, refer Creating a user.
- The password for your Platform Administration tenant administrator account – (this is for the `GWS_PASSWORD` environment variable). For more information, refer Creating a user.
- Java 8 is the current supported version.

Security

The recording files are encrypted throughout the media lifecycle. After the recording files are created, they are encrypted and stored in Amazon S3. RCBS securely transfers the encrypted recordings from S3 to a machine by using the HTTPS internet protocol. The recordings can be decrypted only on that machine.

Getting Started

The following sections explain how to request RCBS functionality and install the software on Windows and Linux environments.

Requesting RCBS functionality

To request RCBS functionality, create a Salesforce case to request delivery of the software. Customer Care will provide an FTP download link to the software, and they will be in touch to request:

- The public IP ranges for the network where the RCBS client software will be installed and from where access to recordings will be established.
  - If a proxy is used, then the public IP range of the proxy will be requested instead.
- If RCBS is planned to be deployed on an AWS EC2 Instance, then additional information will be requested:
  - The AWS Region where RCBS is planned to be deployed.
  - Whether or not you wish to use a VPC Endpoint to connect to Genesys S3 Storage.
    - If you choose to use a VPC Endpoint, Customer Care will provide more information.
- A public PGP key so the Genesys Operations team can securely transfer the S3 storage access credentials to you, which are needed by the RCBS to access the recording storage location.

Once the software has been delivered, Genesys will provide you with the following information:

- The Interaction Recording Web Services URL to access the recording metadata—for example, `https://example.com/api/v2` (this is for the **gwsUriPrefix** parameter).
- The access ID for the S3 storage, used to gain access to the recordings – (this is for the

AWS_ACCESS_KEY_ID environment variable).

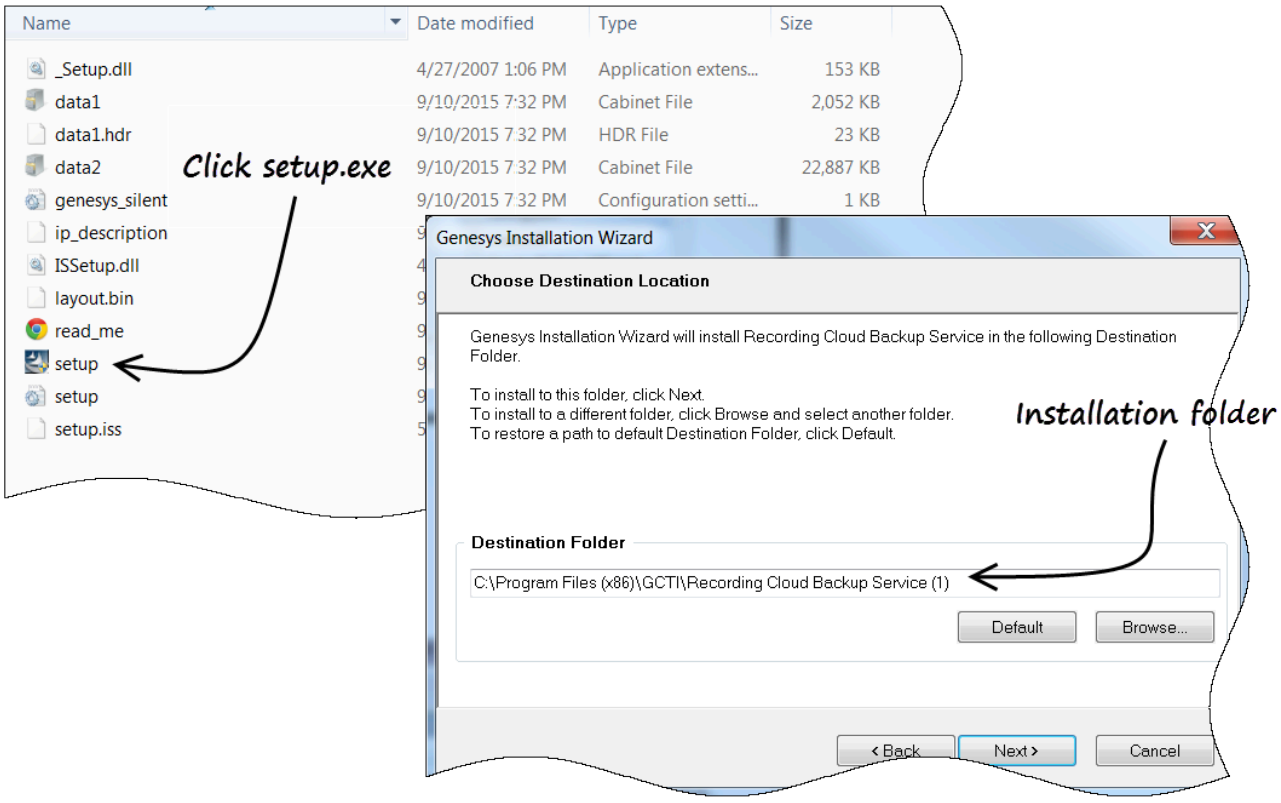
- The secret access key for the S3 storage, used to gain access to the recordings – (this is for the AWS_SECRET_ACCESS_KEY environment variable).

Creating a user

Refer to Creating a new user from scratch to create a user for RCBS. Perform steps 1 and 2, clear the Agent check box, skip the remaining steps, and save to create a user for RCBS. The user that is created is GWS_USERNAME environment variable and the password for this user is GWS_PASSWORD environment variable.

> **Important**
>
> If you plan to run RCBS in verification mode, an extra provisioning step is required for your user. Please contact Genesys to get the required provisioning.

Installing on Windows

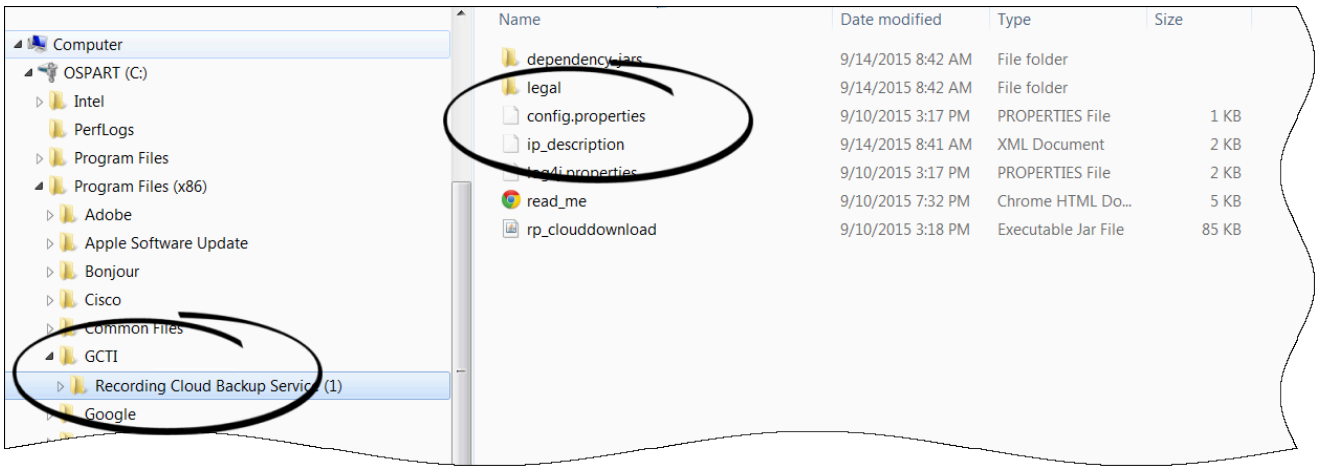


Locate your software in the installation directory, and click **setup.exe** to start the Genesys

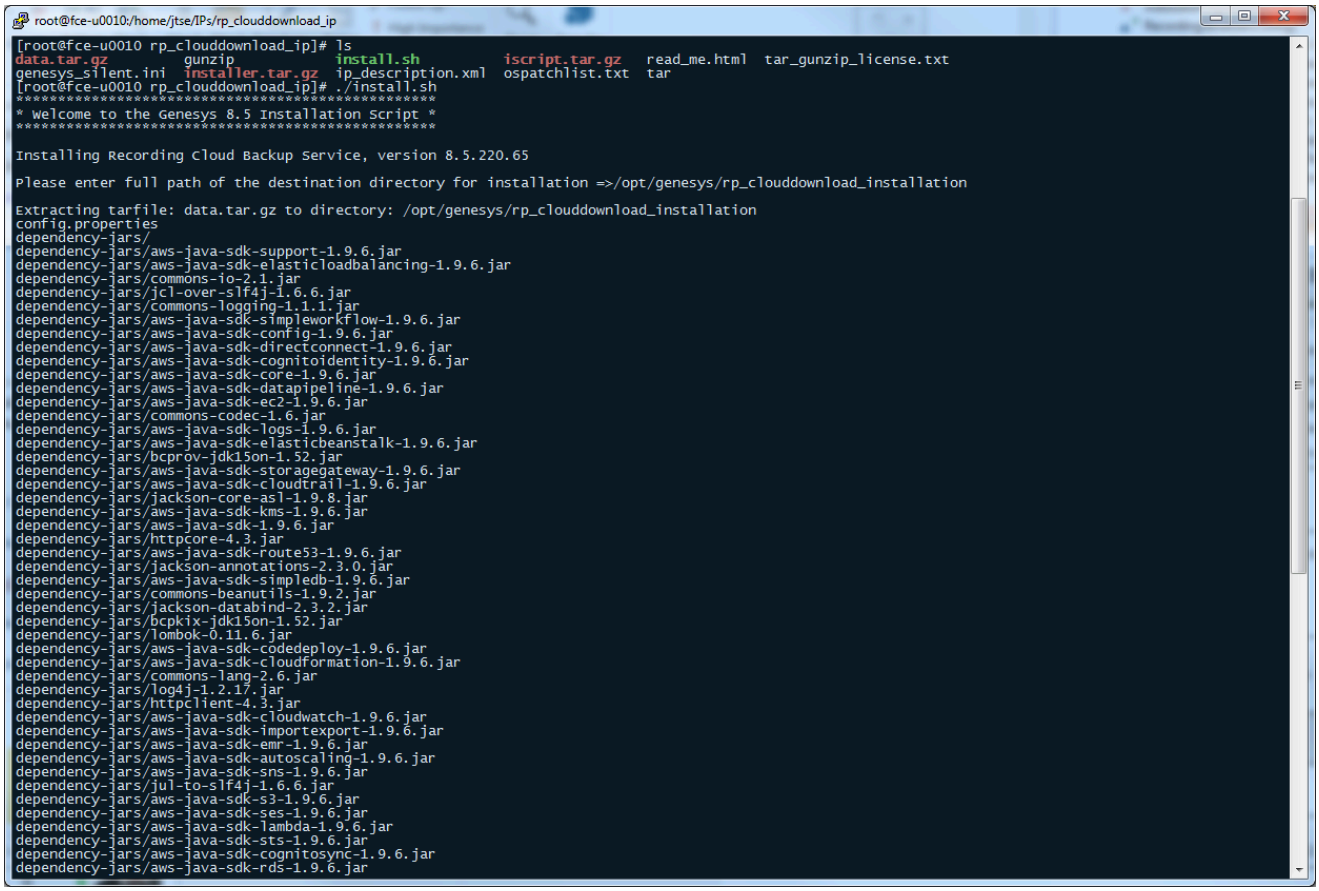
Installation Wizard.

Follow through the wizard until finished making sure that you make note of the installation directory.

Check the installation directory and verify that the **config.properties** file is available.



Installing on Linux



The **glibc.i686** package is required to install RCBS. To install **glibc.i686**, run the following command: `yum install glibc.i686`

In the installation directory, at the prompt, type `./install.sh`.

Let the script install your software.

Check the installation directory and verify that the **config.properties** file is available.

Configuration and setup

The following sections explain the configuration properties and environment variables to set for proper functioning of RCBS.

Configuration properties

The following properties must be modified to successfully retrieve recording files from Amazon S3.

Locate your **config.properties** file, usually found in the installation directory, edit the file with a text editor, and set the following parameters:

Parameter Name	Description	Example Value
gwsUriPrefix	The URL prefix of Interaction Recording Web Services where the metadata for the recording files is retrieved from. This is a mandatory parameter and will be provided by Genesys.	`https://example.com/api/v2`
maxAge	All recordings newer than the specified **maxAge** value, in days, are downloaded. You can specify any integer greater than or equal to 0 (0 is any age). The default value is 2, which means that all recordings from the last 2 days will be retrieved. If recordings have already been downloaded, they will not be downloaded again. **Note:** If recordings are moved from their downloaded folder (**targetDir**), they will be downloaded again when RCBS is run. To ensure recordings are not downloaded more than once, only move recordings from this folder once **maxAge** days have passed since RCBS was last run.	2
targetDir	The directory where the recordings are downloaded to. This directory can be anywhere on the system as long as the account running the software has permission to write to the directory. Ensure that the required space is available to download the desired number of recordings. **Note:** • On both Windows and Linux, you must use the directory separator "/" (forward slash) instead of "\" (backslash). • RCBS supports the use of a UNC path for targetDir. For example, targetDir = //server_name/path.	`<Installation Directory>/target`

Specify the following parameters only if the machine running RCBS cannot connect directly to Amazon S3 or the Interaction Recording Web Services address.

Parameter Name	Description	Example Value
awsProxyHost	Indicates the proxy host address to be used for Amazon Web Services. Specify only the host name or IP address.	`10.0.1.31`
awsProxyPort	Indicates the proxy port to be used for the corresponding **awsProxyHost** parameter to connect to Amazon Web Services.	`8080`
gwsProxyHost	Indicates the proxy host address to be used for Interaction Recording Web Services. The format is `http://proxyaddress`.	`http://10.0.1.31`
gwsProxyPort	Indicates the proxy port to be used for the corresponding **gwsProxyHost** parameter to connect to Interaction Recording Web Services.	`8080`

Environment variables

Once configured, before running the Recording Cloud Backup Service from the command line, the following environment variables must be set based on those provided earlier:

- GWS_USERNAME

- GWS_PASSWORD
- AWS_ACCESS_KEY_ID
- AWS_SECRET_ACCESS_KEY

On Windows

To set environment variables, select **System** from the **Control Panel** (change category view (**View by**) to **Small icons**), click **Advanced system settings**, and then click **Environment Variables**. Under User variables for `<your_user>` or System variables, add as the following:

Variable	Value
GWS_USERNAME	username@customer_tenant.com
GWS_PASSWORD	`your_password`
AWS_ACCESS_KEY_ID	`your_aws_access_key_id`
AWS_SECRET_ACCESS_KEY	`your_aws_secret_access_key`

On Linux

Create the **rcbs.sh** file under the `/etc/profile.d/` directory. The file should contain the following:

```
#!/bin/bash
export GWS_USERNAME=username@customer_tenant.com
export GWS_PASSWORD=your_password
export AWS_ACCESS_KEY_ID=your_aws_access_key_id
export AWS_SECRET_ACCESS_KEY=your_aws_secret_access_key
```

Provide execute permission and using the **source** command this file will be used for setting the environment variable as follows:

```
[root@rcbsmachine ~]# cd /etc/profile.d/
[root@rcbsmachine profile.d]# chmod +x rcbs.sh
[root@rcbsmachine profile.d]# source rcbs.sh
```

Launching the Recording Cloud Backup Service

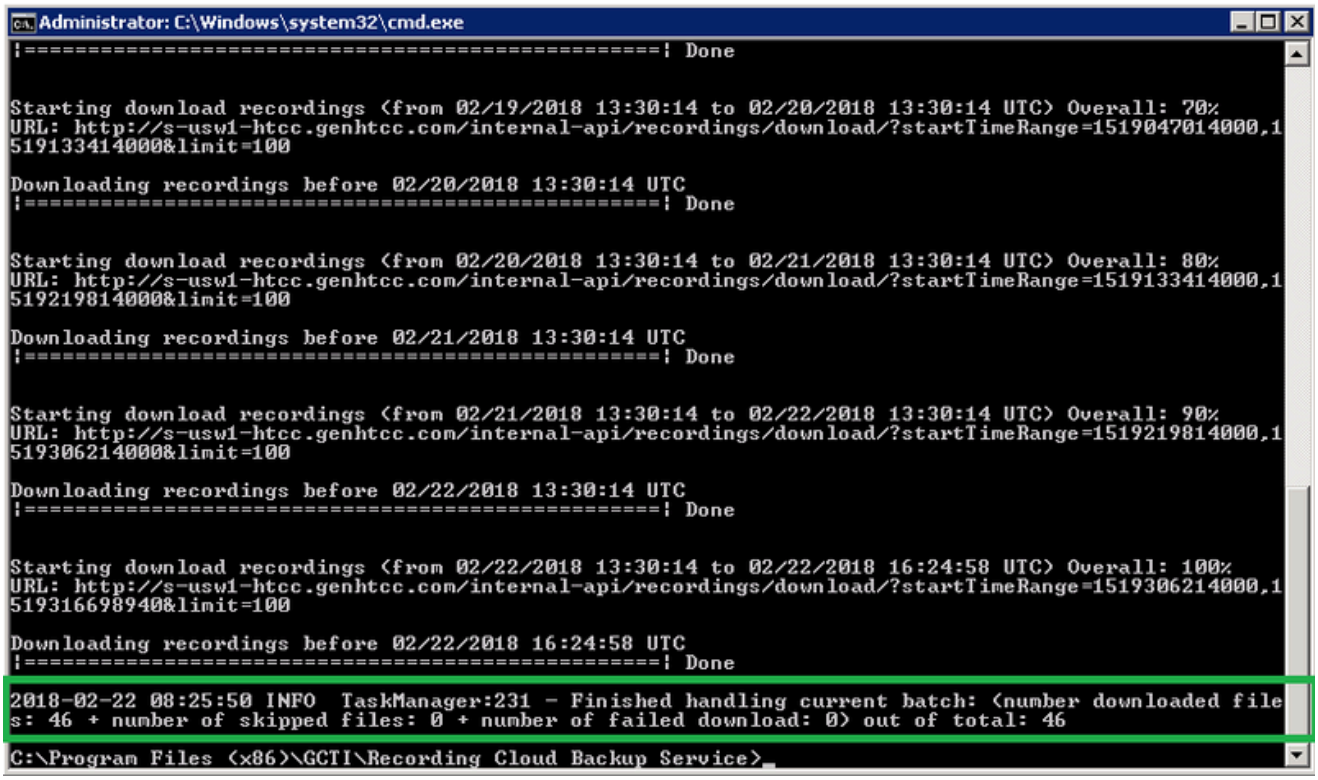
After the config.properties file has been modified and the environment variables are set, access the RCBS installation directory and type the following command line to start RCBS:

```
java –jar rp_clouddownload.jar –config config.properties
```

> **Important**
>
> Do not copy and paste the command from this document. Instead, manually type the command.

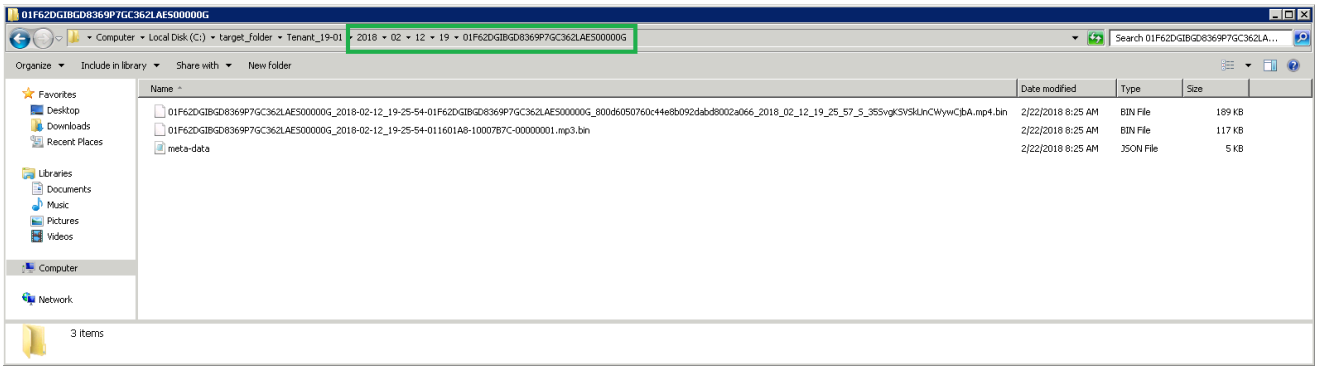
You can view the progress of the download process in percentage in console window. Download process will be completed once the progress reaches 100% with the message as shown in the

following image.



The tool exits when the backup is complete. Check your **targetDir** to ensure that the expected recordings have been downloaded.

In the below example, 2018 is the year, 02 is the month, 12 is the date, 19 is the hour, and 01F62DGIBGD8369P7GC362LAES00000G is the recording folder. Recordings are grouped at the hour as the lowest level. Each recording folder has encrypted voice and screen recording files (if applicable) along with a metadata file in JSON format.



Running RCBS in verification mode

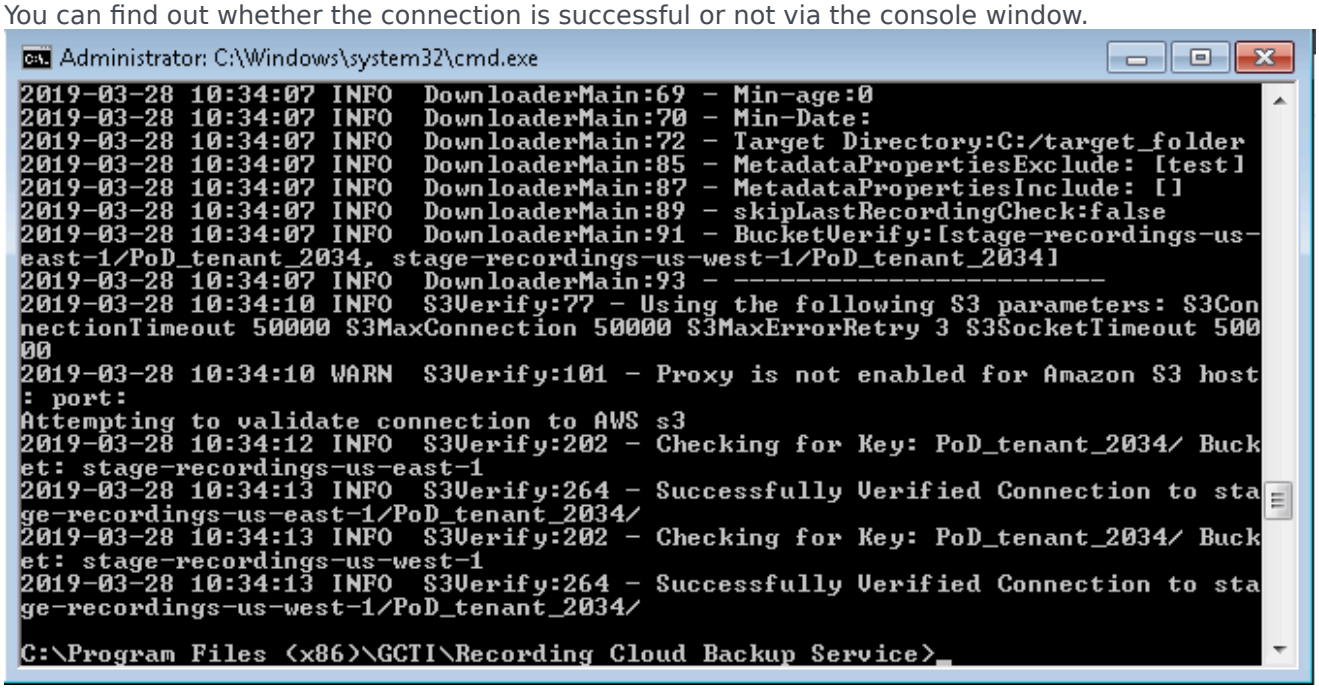
After the config.properties file has been modified and the environment variables are set, access the

RCBS installation directory and type the following command line to start RCBS to verify your connection to download recordings from AWS S3:

```
java –jar rp_clouddownload.jar –config config.properties -verify s3
```

> **Important**
>
> - This function is only available in RCBS 8.5.298.58 and later versions.
> - If you plan to run RCBS in verification mode, an extra provisioning step is required for your user. Please contact Genesys to get the required provisioning.

You can find out whether the connection is successful or not via the console window.



Scheduling backup

The following sections explain how to schedule a backup.

How to schedule a Windows task

For information on how to schedule or manage your tasks in Windows, see the Windows documentation. Do not forget to set your environment variables.

How to create a Linux cronjob

You can set up a recurring backup by using cronjob (crontab -e). The following example illustrates how to use "crontab -e" to configure an appropriate cronjob on Linux:

```
AWS_ACCESS_KEY_ID=<access_id>
AWS_SECRET_ACCESS_KEY=<access_key>
GWS_PASSWORD=<gws_password>
GWS_USERNAME=<gws_username>
30 4,10,16,22 * * * (cd <installation_folder>; java -jar rp_clouddownload.jar -config
config.properties)
```

Replace the above <access_id>, <access_key>, <gws_password>, <gws_username>, <installation_folder> with the actual values, and the job will be executed 4 times daily at 4:30, 10:30, 16:30 and 22:30.

> **Important**
>
> - Genesys strongly recommends that you create backup copies several weeks prior to the expected deletion date.
> - Ensure your local machine has enough space for the scheduled backup.

Decrypting the downloaded files

You will use OpenSSL to decrypt your recording files. You can download the software by following the instructions here.

When working with Windows, the OpenSSL binaries can be downloaded from: OpenSSL Binaries Distribution.

Each recording folder contains the encrypted recording files and the respective recording metadata files (in json format).

To decrypt the downloaded files that are in encrypted format, use the following OpenSSL commands:

Windows:

```
openssl smime -decrypt –inform DER -in <encrypted.file.bin> -inkey <private_key_file> –out
<outputfile>
```

Linux:

```
openssl cms –decrypt –inform DER -in <encrypted.file.bin> -binary -inkey <private_key_file>
–out <outputfile>
```

where:
<encrypted.file.bin> is the file to be decrypted

<private_key_file> is the private key you used to initially configure recording file encryption, so that the recording files can be decrypted
<outputfile> is the file that would be written after decryption

Storage

Ensure that the required space is available to download the desired number of recordings. Genesys recommends that you decrypt the recording files to a different destination than the encrypted files so that the original encrypted source file is not modified or overwritten by the decrypted file.

Advanced configuration

If you are an advanced user, you can change the behavior of the Recording Cloud Backup Service by changing the values of the parameters in the **config.properties** file.

Configuring download period

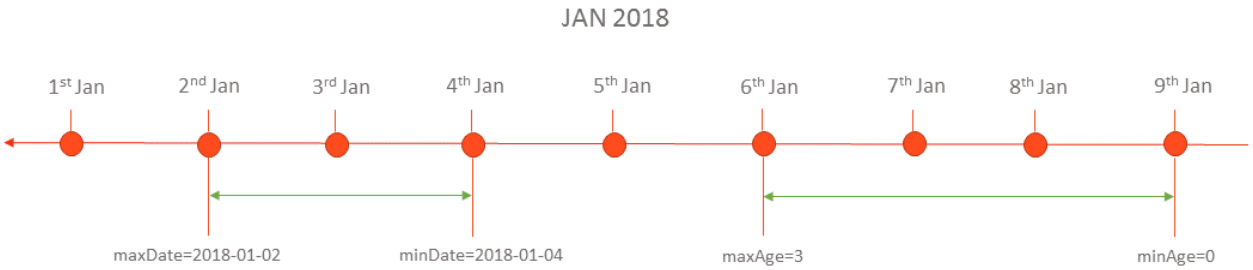
Use the following parameters to set the download period.

Parameter Name	Description	Example Value
minAge	All recordings older than the specified **minAge** value, in days, are included in the download period. You can specify any integer greater than or equal to 0. The default is 0, which means that the download period includes recordings older than the current time. The **minAge** value must be less than the **maxAge** value. ISO 8601 format can also be used with respect to current time. Refer to ISO 8601 format for more information. If you specify a minimum age of PT30M, all recordings that are older than half an hour ago are included in processing. If you specify a minimum age of P30D, all recordings older than 30 days are included in processing.	0
maxAge	All recordings newer than the specified **maxAge** value, in days, are included in download period. You can specify any integer greater than or equal to 0 (0 is any age). The default value is 2, which means that the period includes recordings from the last 2 days. The **maxAge** value must be greater than the **minAge** value. ISO 8601 format can also be used with respect to current time. Refer to ISO 8601 format for more information. If you specify a maximum age of PT30M, all recordings that are newer than half an hour are included in processing. If you specify a maximum age of P30D, all recordings newer than 30 days are included in processing.	2
minDate	The absolute date, in the YYYY-MM-DD format (in GMT), to include recordings older than the specified date. When specified, this value would be used as the **minAge**. Alternatively, the epoch time value can be specified instead of the YYYY-MM-DD format. Refer to UNIX Epoch time format for more information. This parameter is optional.	2015-07-31 Or 1438300800000
maxDate	The absolute date, in the YYYY-MM-DD format (in GMT), to include recordings newer than the specified date. When specified, this value would be used as the **maxAge**, and override the local storage's last recording endtime's value (last_recording_endtime.txt). Alternatively, the	2015-07-31 Or 1438300800000

Parameter Name	Description	Example Value
	epoch time value can be specified instead of the YYYY-MM-DD format. Refer to UNIX Epoch time format for more information. This parameter is optional.	

For repetitive scheduled download of recordings, use the **minAge** and **maxAge** parameters. RCBS will download recordings for the configured duration with respect to current time. For example, if the current date is January 09, 2018 and if you want to download recordings of three days with respect to current time, then set **minAge=0** and **maxAge=3** as shown in the following image. Recordings between 06 January and 09 January will be downloaded.

For one-time download of recordings between two dates (GMT), use the **minDate** and **maxDate** parameters. RCBS will download recordings within the configured period. To perform a one-time download, Genesys recommends that you create a copy of the **config.properties** file, delete or rename the **last_recording_endtime.txt** file and make changes to the **minDate** and **maxDate** parameters. When you execute RCBS, use the following command with the name of the copy of the configuration file (for example, new_config.properties): `java -jar rp_clouddownload.jar -config new_config.properties`. To download recordings between a date range, for example, between 02 January to 04 January, set the parameters such as the following: **minDate=2018-01-04** and **maxDate=2018-01-02**.



Download period always includes the recordings newer than the last download period as specified in the **last_recording_endtime.txt** file.
The **last_recording_endtime.txt** file is updated after download of recordings for configured period has completed successfully. The next time when the download tool starts, it checks to see if the **last_recording_endtime.txt** file is older than the specified **maxAge** parameter. If it is, the tool uses the value from the **last_recording_endtime.txt** instead of the configured **maxAge** value. For example, the download tool is scheduled to run daily with **maxAge** set to 2 days. If the server was offline for three days, it is replaced with the **last_recording_endtime.txt** file check, and the tool downloads all the recordings that were missed.

UNIX Epoch time format

RCBS supports UNIX Epoch time format in milliseconds for certain parameters. It is a 13 digit integer value. You can convert the date and time to 13 digit integer value by using tools such as Epoch Converter.

ISO 8601 format

RCBS supports ISO 8601 format `P[n]Y[n]M[n]DT[n]H[n]M[n]S` for certain parameters. The description of the format is as follows:

- P - Mandatory prefix to identify that the configuration is in ISO format.
- [n] - Integer value which is specific for the suffix followed by it
- [n]Y - Number of Years. Example - 2Y means 2 years
- [n]M - Number of Months
- [n]D - Number of Days
- T - Mandatory prefix to identify the following content is time
- [n]H - Number of Hours
- [n]M - Number of Minutes
- [n]S - Number of Seconds

Examples of valid values:

- P1Y2DT4H30M - Indicates 1 Year + 2 Days + 4 Hours + 30 Minutes
- P6MT12H - Indicates 6 Months + 12 Hours

Configuring media

Use the following parameters to set the media configurations.

Parameter Name	Description	Example Value
mediaTypePattern	Indicates the file types that will be downloaded by RCBS in a regular expression. The default value is audio\/mp3|video\/mp4. This value will download both audio and video files. To download only MP3, set this value to audio\/mp3. To download only MP4, set this value to video\/mp4. This parameter is optional.	audio\/mp3|video\/mp4
recordingFolderFormat	The directory structure for storing the recordings. Default value is in the yyyy/MM/dd/HH format which means the top level folder is year, subfolder is month, then date, then hour. Note the directory separator "/" (forward slash) must be used instead of "\" (backslash) on both Windows and Linux.	yyyy/MM/dd/HH
encryptionFormat	The cipher to use if the encryption is performed by the download tool. The supported values are AES-128 or AES-256. The default value is AES-128. **Note:** If AES-256 is used, the JCE unlimited Strength Jurisdiction Policy File must be installed.	AES-128
metadataOnly	Indicates whether to download the recording files along with the metadata. If set to true, the metadata is downloaded without the recordings. You do not need S3 credentials when using this option. Default value is false.	false
usePayloadSigning	Configures whether payload signing is used during the file transfer from Amazon S3. Disabling payload signing improves performance. Default value is false. To enable, set to true.	false

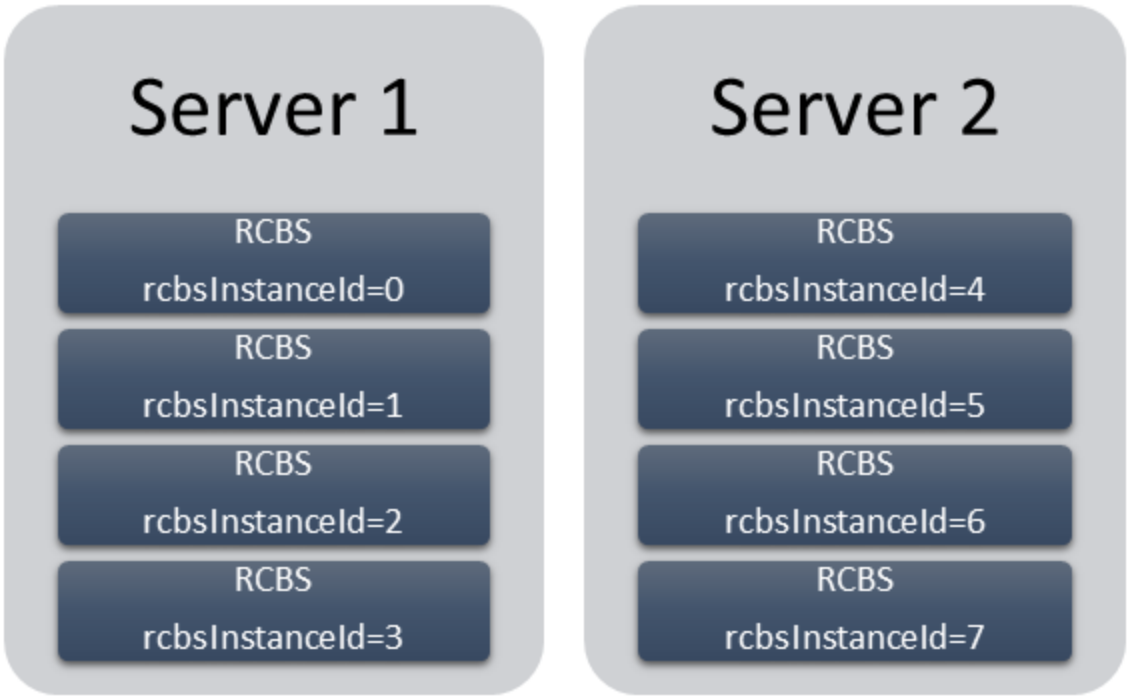
Configuring multiple instances

Multiple instances of RCBS can be used to increase the download rate of the recordings for the configured download period. Multiple instances can be used in the same machine or different machines based on network bandwidth. Each instance will process the same download period and split the download process based on hashing of the recording ID. A particular recording will be downloaded by only one instance and all other instances will skip that recording. All instances should be running properly to download all the recordings in the configured download period.

The **minAge** and **maxAge** of the separate instances must be the same. The point of multiple instances is that each instance is assigned a different subset of recordings to download to spread the load. Changing the **minAge** and **maxAge** means each instance will download a separate chunk of a different time period.

Note: If running multiple instances of RCBS on the same machine, each RCBS must be started from a different installation directory, and **targetDir** for each instance must point to a different output folder. RBCS has a built-in protection mechanism to prevent multiple instances from writing to the same directory; the second instance will terminate immediately if they share the same path.

Parameter Name	Description	Example Value
totalRcbsInstances	The total number of RCBS instances deployed.This parameter must be used with the **rcbsInstanceId** configuration parameter. This parameter is optional.	4
rcbsInstanceId	Indicates the current RCBS instance ID that shares the overall load. **rcbsInstanceId** starts at **0**, up to **totalRcbsInstances** minus **1**. For example, if the download load was distributed across four instances of a running RCBS process, then **totalRcbsInstances** should be set to 4. For each RCBS configuration, assign **rcbsInstanceId** to **0** for the first RCBS instance, **rcbsInstanceId** to **1** for the second RCBS instance, **rcbsInstanceId** to **2** for the third RCBS instance, and **rcbsInstanceId** to **3** for the fourth RCBS instance. All RCBS instances should have the same **minAge** and **maxAge** configuration values.	0



In the above image, eight instances of RCBS spread between two servers. Here, the value of **totalRcbsInstances** is 8 in all the instances.

Configuring URIs (optional)

The following parameters are optional. Do not set the URI path for these parameters unless instructed by Genesys.

Parameter Name	Description	Example Value
gwsRecordingsUri	The path to the recording API.	`/recordings`
gwsSettingsUri	The path to the settings API.	`/me/settings/rcbs`

Recording metadata

Metadata is organized by records and can be used for finding specific calls from a larger downloaded group of recordings (for example, by searching for a particular string of text, perhaps the 'callerPhoneNumber'). A record represents a single call interaction which may contain multiple calls and recording segments. A metadata record is uniquely identified (per switch) by a CallUUID (GUID).

The metadata record is stored in JSON format and contains three main sections within the top level object.

- The interaction level attributes (the top level object's attributes)

- The mediaFiles list—A list of media files connected to the call interaction
- The eventHistory list—A list of call events including attached data events and agent left and join events.

[+] Show the properties and examples

Metadata properties

Property	Description
Id	The CallUUID for the recording interaction.
callRecordingId	The call recording identifier. This attribute is in screen recording metadata only.
callerPhoneNumber	The caller's phone number.
dialedPhoneNumber	The dialed phone number.
startTime	The start time of the call.
stopTime	The end time of the call.
screenRecording	Indicates whether or not the call recording has one or more associated screen recordings. This attribute is in call recording metadata only.
region	The region of the call.
mediaFiles	A list of media file records. See the mediaFile properties.
eventHistory	The events attached to the call. See the eventHistory properties.

mediaFile properties

The following table describes the mediaFile properties.

Property	Data Type	Description	Required
startTime	datetime	Specifies the start time of the media file.	Yes
stopTime	datetime	Specifies the stop time of the media file. If MCP fails, this value will be the same as the `startTime`.	Yes
mediaID	string	Specifies the media file name for the media file that is used by clients to refer to the same media file. MCP ensures that this value is globally unique.	Yes
type	string	Specifies the MIME type of the media file.	Yes
duration	time	Specifies the time	No

Property	Data Type	Description	Required
		duration of the media file.	
size	number	Specifies the size, in bytes, of the media file.	No
tenant	string	Specifies the tenant that the recording belongs to.	Yes
ivrprofile	string	Specifies the IVR Profile name that serviced the recording.	Yes
parameters	object—The properties are parameters.	Specifies the list of additional metadata information provided by SIP Server and the client applications. The properties are: • username • sipsAppName • ani • dnis • dateTime • connid • agentId • id • record	Yes
masks	array of objects—Each object contains the time and type property.	Specifies the time stamps of the pause/ resume periods if the recording is masked by a client application.	No
certAlias	array of strings	Specifies a list of aliases to the encryption certificates if the media file is encrypted.	No
partitions	array of strings	Specifies a list of partition names for the media file.	Yes
accessgroups	array of strings	Specifies the access groups identified agent associated with the recording.	Yes
channels	number	Specifies whether the recording audio is	Yes

Property	Data Type	Description	Required
		capture in mono (1) or stereo (2).	

eventHistory properties

The following table describes the eventHistory properties.

Property	Data Type	Description	Required
occurredAt	datetime	Specifies the start time of the event.	Yes
calluuid	string	Specifies the call UUID that the event belongs to.	Yes
event	string	Specifies the event type: • `Joined` • `Left` • `data`	Yes
contact	object	Specifies the the contact information of the caller who joined or left the recording if the event is `Joined` or `Left`.	No
data	object	The attached data included in the recording if the event is `data`.	No

Metadata format

The following code snippet illustrates the metadata format:

```
{
  "id" : "011AP643CSAPR4FKQGQE31TAES0001TH",
  "callerPhoneNumber" : "8522001",
  "dialedPhoneNumber" : "+14160000001",
  "startTime" : "2015-09-01T16:42:59.000+0000",
  "stopTime" : "2015-09-01T16:43:36.000+0000",
  "mediaFiles" : [ {
    "startTime" : "2015-09-01T16:42:59.000+0000",
    "stopTime" : "2015-09-01T16:43:10.000+0000",
    "callUUID" : "011AP643CSAPR4FKQGQE31TAES0001TH",
    "mediaId" :
"011AP643CSAPR4FKQGQE31TAES0001TH_2015-09-01_16-42-58-006B015E-10003E55-00000001.mp3",
    "type" : "audio/mp3",
    "duration" : "11595",
    "tenant" : "Environment",
    "ivrprofile" : "CallRecProfile",
```

```
    "size" : "184896",
    "parameters" : {
      "id" : "011AP643CSAPR4FKQGQE31TAES0001TH_2015-09-01_16-42-58",
      "dnis" : "+14160000001",
      "username" : "screen1@genesys.com",
      "connId" : "006a0269722cd7b1",
      "callUuid" : "011AP643CSAPR4FKQGQE31TAES0001TH",
      "ani" : "8522001",
      "dateTime" : "2015-09-01T16:42:58Z",
      "recordDN" : "+14160000001",
      "agentId" : "+14160000001",
      "sipsAppName" : "SIP_Server"
    },
    "certAlias" : [ "rcs_Tenant_21-86_GIR_Team:214:CN=gir_21-86,
E=arun_sundar.n@genesys.com,OU=gir_gcloud,O=Genesys,
L=Chennai,ST=TN,C=IN:2" ],
    "partitions" : [ ],
    "accessgroups" : [ "/" ],
    "channels" : 2
  }, {
    "startTime" : "2015-09-01T16:43:11.000+0000",
    "stopTime" : "2015-09-01T16:43:36.000+0000",
    "callUUID" : "011AP643CSAPR4FKQGQE31TAES0001TH",
    "mediaId" :
"011AP643CSAPR4FKQGQE31TAES0001TH_2015-09-01_16-43-11-006B015E-10003E57-00000001.mp3",
    "type" : "audio/mp3",
    "duration" : "25426",
    "tenant" : "Environment",
    "ivrprofile" : "CallRecProfile",
    "size" : "406080",
    "parameters" : {
      "id" : "011AP643CSAPR4FKQGQE31TAES0001TH_2015-09-01_16-43-11",
      "dnis" : "+14160000001",
      "username" : "screen3@genesys.com",
      "connId" : "006a0269722cd7b1",
      "callUuid" : "011AP643CSAPR4FKQGQE31TAES0001TH",
      "ani" : "8522001",
      "dateTime" : "2015-09-01T16:43:11Z",
      "agentId" : "+14160000003",
      "recordDN" : "+14160000003",
      "sipsAppName" : "SIP_Server"
    },
    "certAlias" : [ "rcs_Tenant_21-86_GIR_Team:214:CN=gir_21-86,
E=arun_sundar.n@genesys.com,OU=gir_gcloud,O=Genesys,
L=Chennai,ST=TN,C=IN:2" ],
    "partitions" : [ ],
    "accessgroups" : [ "/" ],
    "channels" : 2
  } ],
  "eventHistory" : [ {
    "occurredAt" : "2015-09-01T16:42:59.000+0000",
    "eventId" : "2015-09-01 16:42:59.167_011AP643CSAPR4FKQGQE31TAES0001TH",
    "event" : "Data",
    "calluuid" : "011AP643CSAPR4FKQGQE31TAES0001TH",
    "data" : {
      "added" : {
        "GSIP_RECORD" : "ON",
        "GSIP_REC_FN" : "011AP643CSAPR4FKQGQE31TAES0001TH_2015-09-01_16-42-58"
      }
    }
  }, {
    "occurredAt" : "2015-09-01T16:43:10.000+0000",
    "eventId" : "2015-09-01 16:43:10.513_011AP643CSAPR4FKQGQE31TAES0001TH",
```

```
    "event" : "Data",
    "calluuid" : "011AP643CSAPR4FKQGQE31TAES0001TH",
    "data" : {
      "deleted" : {
        "GSIP_RECORD" : "ON"
      }
    }
  }, {
    "occurredAt" : "2015-09-01T16:43:11.000+0000",
    "eventId" : "2015-09-01 16:43:11.357_011AP643CSAPR4FKQGQE31TAES0001TH",
    "event" : "Data",
    "calluuid" : "011AP643CSAPR4FKQGQE31TAES0001TH",
    "data" : {
      "added" : {
        "GSIP_RECORD" : "ON"
      },
      "updated" : {
        "GSIP_REC_FN" : "011AP643CSAPR4FKQGQE31TAES0001TH_2015-09-01_16-43-11"
      }
    }
  }, {
    "occurredAt" : "2015-09-01T16:43:36.000+0000",
    "eventId" : "2015-09-01 16:43:36.653_011AP643CSAPR4FKQGQE31TAES0001TH",
    "event" : "Data",
    "calluuid" : "011AP643CSAPR4FKQGQE31TAES0001TH",
    "data" : {
      "deleted" : {
        "GSIP_RECORD" : "ON"
      }
    }
  }, {
    "occurredAt" : "2015-09-01T16:42:58.000+0000",
    "event" : "Joined",
    "calluuid" : "011AP643CSAPR4FKQGQE31TAES0001TH",
    "contact" : {
      "type" : "User",
      "phoneNumber" : "+14160000001",
      "userName" : "screen1@genesys.com",
      "firstName" : "Screen",
      "lastName" : "GIR"
    }
  }, {
    "occurredAt" : "2015-09-01T16:43:11.000+0000",
    "event" : "Joined",
    "calluuid" : "011AP643CSAPR4FKQGQE31TAES0001TH",
    "contact" : {
      "type" : "User",
      "phoneNumber" : "+14160000003",
      "userName" : "screen3@genesys.com",
      "firstName" : "Screen3",
      "lastName" : "GIR"
    }
  }, {
    "occurredAt" : "2015-09-01T16:42:58.000+0000",
    "event" : "Joined",
    "calluuid" : "011AP643CSAPR4FKQGQE31TAES0001TH",
    "contact" : {
      "type" : "External",
      "phoneNumber" : "8522001"
    }
  }, {
    "occurredAt" : "2015-09-01T16:43:10.000+0000",
    "event" : "Left",
```

```
    "calluuid" : "011AP643CSAPR4FKQGQE31TAES0001TH",
    "contact" : {
      "type" : "User",
      "phoneNumber" : "+14160000001",
      "userName" : "screen1@genesys.com",
      "firstName" : "Screen",
      "lastName" : "GIR"
    }
  }, {
    "occurredAt" : "2015-09-01T16:43:36.000+0000",
    "event" : "Left",
    "calluuid" : "011AP643CSAPR4FKQGQE31TAES0001TH",
    "contact" : {
      "type" : "User",
      "phoneNumber" : "+14160000003",
      "userName" : "screen3@genesys.com",
      "firstName" : "Screen3",
      "lastName" : "GIR"
    }
  }, {
    "occurredAt" : "2015-09-01T16:43:36.000+0000",
    "event" : "Left",
    "calluuid" : "011AP643CSAPR4FKQGQE31TAES0001TH",
    "contact" : {
      "type" : "External",
      "phoneNumber" : "8522001"
    }
  } ],
  "callType" : "Inbound",
  "screenRecording" : false,
  "region" : "region1"
}
```


Disk usage estimation

RCBS downloads the voice and screen recording files from the Genesys Interaction Recording system and stores the files on the local machine, thereby occupying the disk space. This section explains how to estimate the amount of disk space that will be used.

Estimating disk space required to store downloaded voice recordings

The disk space required to store voice recordings can be estimated as follows:

- **Estimated size of a metadata file**: The size of a metadata file for each voice recording has an upper bound of 1 MB. You can use that value to estimate how much space the metadata files will use.
- **Estimated size of a voice recording file**: The size of a voice recording file can be estimated by using the average call duration (in seconds) and the recording bitrate (in kbps, the default value is 32 kbps).

Total disk usage for a day

The estimated disk usage per day (in MB) can be calculated in one of the following ways:

- `<number of recordings per day> * (<estimated disk usage of a recording file> + <estimated disk usage of a metadata file>)`

- `<number of recordings per day> * [<average call duration> * ((<recording bitrate> / 8 / 1024) + 1)]`

Estimating disk space required to store downloaded screen recordings

The disk space required to store screen recordings can be estimated as follows:

- **Estimated size of a metadata file**: The size of a metadata file for each screen recording has an upper bound of 8 KB. You can use that value to estimate how much space the metadata files will use.
- **Estimated size of a screen recording file**: The size of a Screen Recording file can be estimated by using the average call duration (in seconds) and the screen recording bitrate (in kbps), which is the sum of the screen recording bitrate and voice recording bitrate because the RCBS downloads the muxed screen recording files. The default value for the total bitrate is 256 kbps.

Total disk usage for a day

The estimated disk usage per day (in MB) can be calculated in one of the following ways:

- `<number of recordings per day> * (<estimated disk usage of a recording file> + <estimated disk usage of a metadata file>)`
- `<number of recordings per day> * [<average call duration> * ((<recording bitrate> / 8 / 1024) + 1)]`

IVR Administration

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

Interactive Voice Response (IVR) assists in resolving your customers' issues when they call into your company. Your customer and the IVR interact with one another to drill-down on what the customer wants to do and, ideally, the IVR resolves the issue without having to transfer the call to an agent.

IVR offerings

IVR consists of a computer and an application or call-flow. "Computer" refers to the Genesys Voice Platform (GVP) that executes or interprets the work, which is defined in an application or call-flow.

IVR includes two product offerings — the Enterprise IVR and GVP PaaS. Both the Enterprise IVR (entirely Genesys managed) and GVP PaaS (Customer or Genesys managed) offerings are available for Genesys Engage cloud.

Enterprise IVR

Enterprise IVR (EIVR) is an Inbound Voice offering wherein both the computer (GVP) and the application or call-flow are installed in the Genesys Engage cloud and managed by Genesys. The customer and Genesys work closely together to create and test a suitable IVR call-flow.

GVP PaaS

GVP PaaS is a different Inbound Voice offering wherein the application or call-flow is installed on the customer premises and the incoming telco connection is terminated at the computer (GVP) in the Genesys Engage cloud. Customers can manage the application or call-flow entirely by themselves (or, with Genesys assistance), and rely on Genesys to manage the complex infrastructure and high-availability associated with the telco connection.

GVP reporting

IVR is associated with GVP Reporting. GVP has a number of individual components that perform different tasks during a call.

GVP reports give you a diagnostic look at your contact center's performance by monitoring how the

components perform — by themselves and with each other.

The goal is to improve your contact center's efficiency. For more information, see GVP Reporting.

Enterprise IVR

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

What is Enterprise IVR?

Enterprise IVR (EIVR) is an Inbound Voice offering wherein both the computer (GVP) and the application or call-flow are installed in the Genesys Engage cloud and managed by Genesys. The customer and Genesys work closely together to create and test a suitable IVR call-flow.

GVP PaaS is a different Inbound Voice offering wherein the application or call-flow is installed on the customer premises and the incoming telco connection is terminated at the computer (GVP) in the Genesys Engage cloud. Customers can manage the application or call-flow entirely by themselves (or, with Genesys assistance), and rely on Genesys to manage the complex infrastructure and high-availability associated with the telco connection.

With the EIVR solution, you might need to do these admin tasks:

- Manage DID Groups with IVR Administration
- Provision IVR Profiles yourself

EIVR is comprised of various features designed to run your contact center effectively:

- Personalities
- Audio Resources
- Audio Resource Files
- Parameters
- Parameter Group Templates
- Parameter Groups

Manage DID Groups with IVR Administration

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

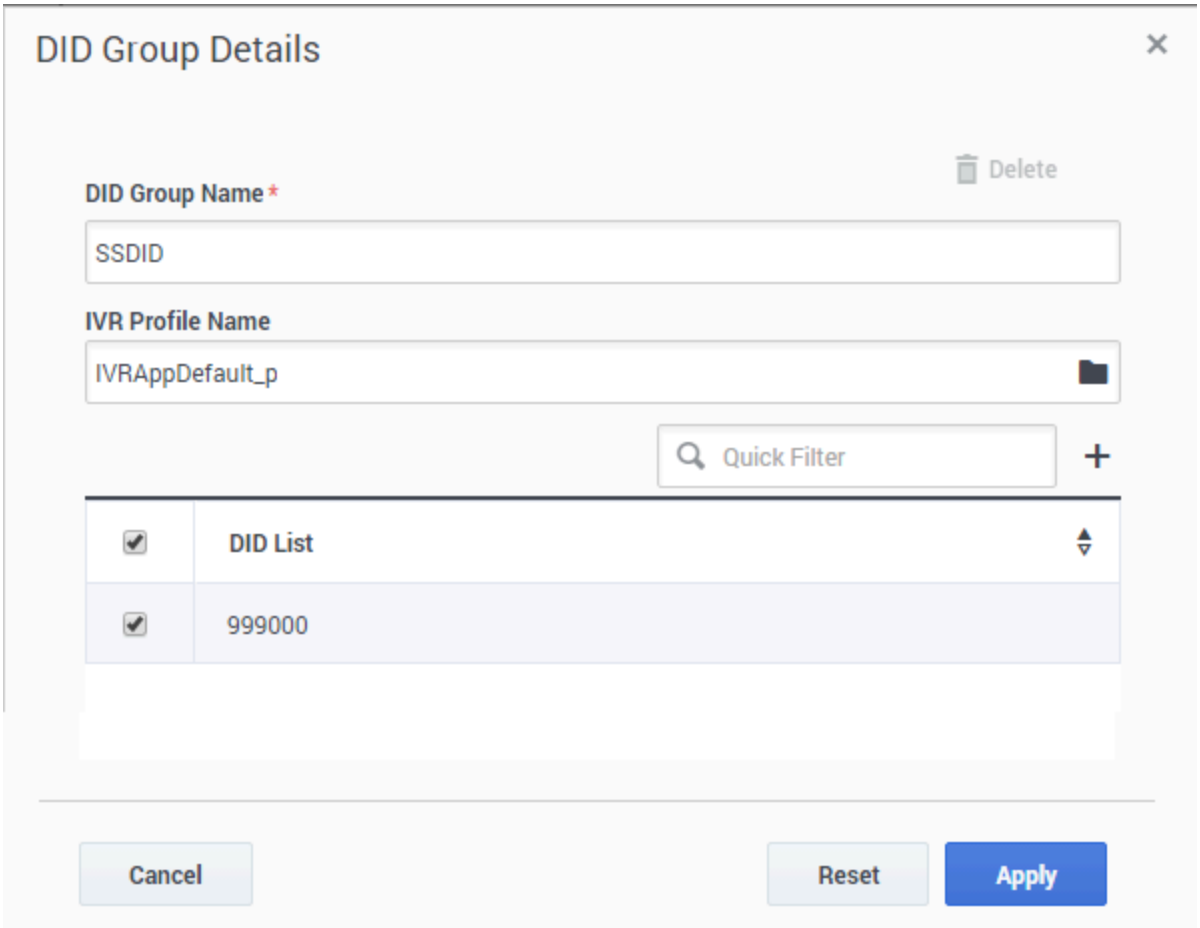
All procedures begin in the DID Groups Main Page.

To get there:

1. Log into IVR Administration as a tenant user.
2. Select DID Groups from the Administration menu.

At the end of each procedure, click **Apply** to save changes.

Create a new DID Group

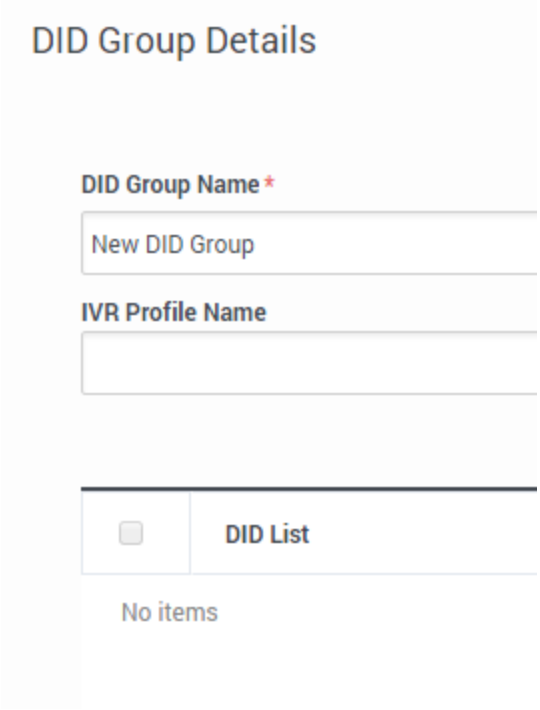


1. Click **New**.
2. Configure the DID Group details dialog:
   - Enter an unused DID Group Name and an existing IVR Profile Name.
   - Check the selection box of each DID in the DID List field, to include that DID in the group you are creating.

     Optional Quick Filter field: enter text to limit the number of DIDs displayed, or click plus (+) to query for additional DIDs not displayed.

When you leave this dialog, the DID Group that you created is now listed in the DID Groups directory.

Modify an existing DID Group



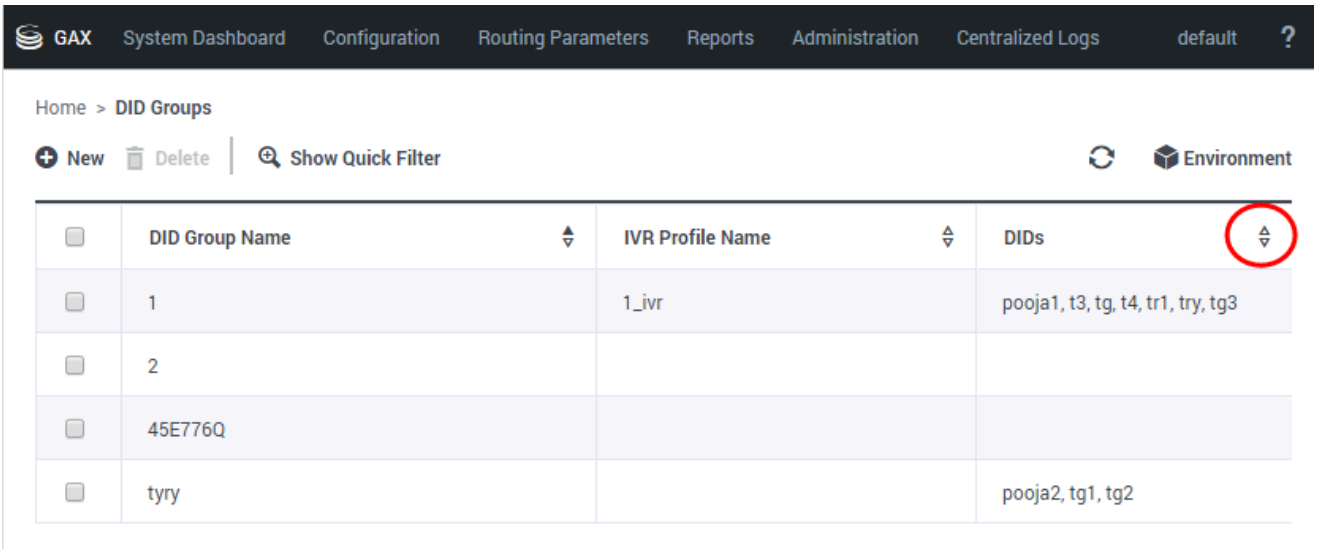
Remove items from the group

1. Click on the name of an existing DID Group to open its DID Group Details dialog.
2. Uncheck individual items to remove them from the DID group.

Provision conflicting entries

- DID group provisioning succeeds when you remove the conflicting entries in overlapping ranges (example: 100-200 and 150-250).

Sort an existing DID Group



1. Click on the name of an existing DID Group to open its DID Group Details dialog.
2. Click the arrowhead (a triangle) at the top right of the DID List field to sort it in the direction of that the arrowhead points (up=descending order, down=ascending order).

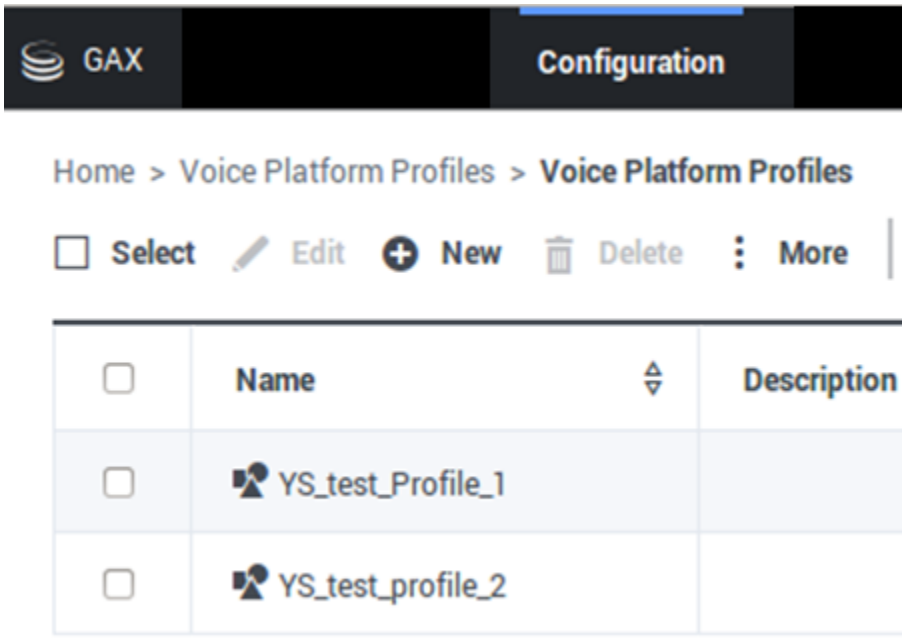
The list of DIDs in the group is sorted. You can re-sort the list, but you cannot unsort it.

Provision IVR Profiles yourself

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

You can use IVR Administration to provision Voice Platform Profiles (IVR profiles) and map Direct Inward Dialed (DID) Groups to these IVR profiles.

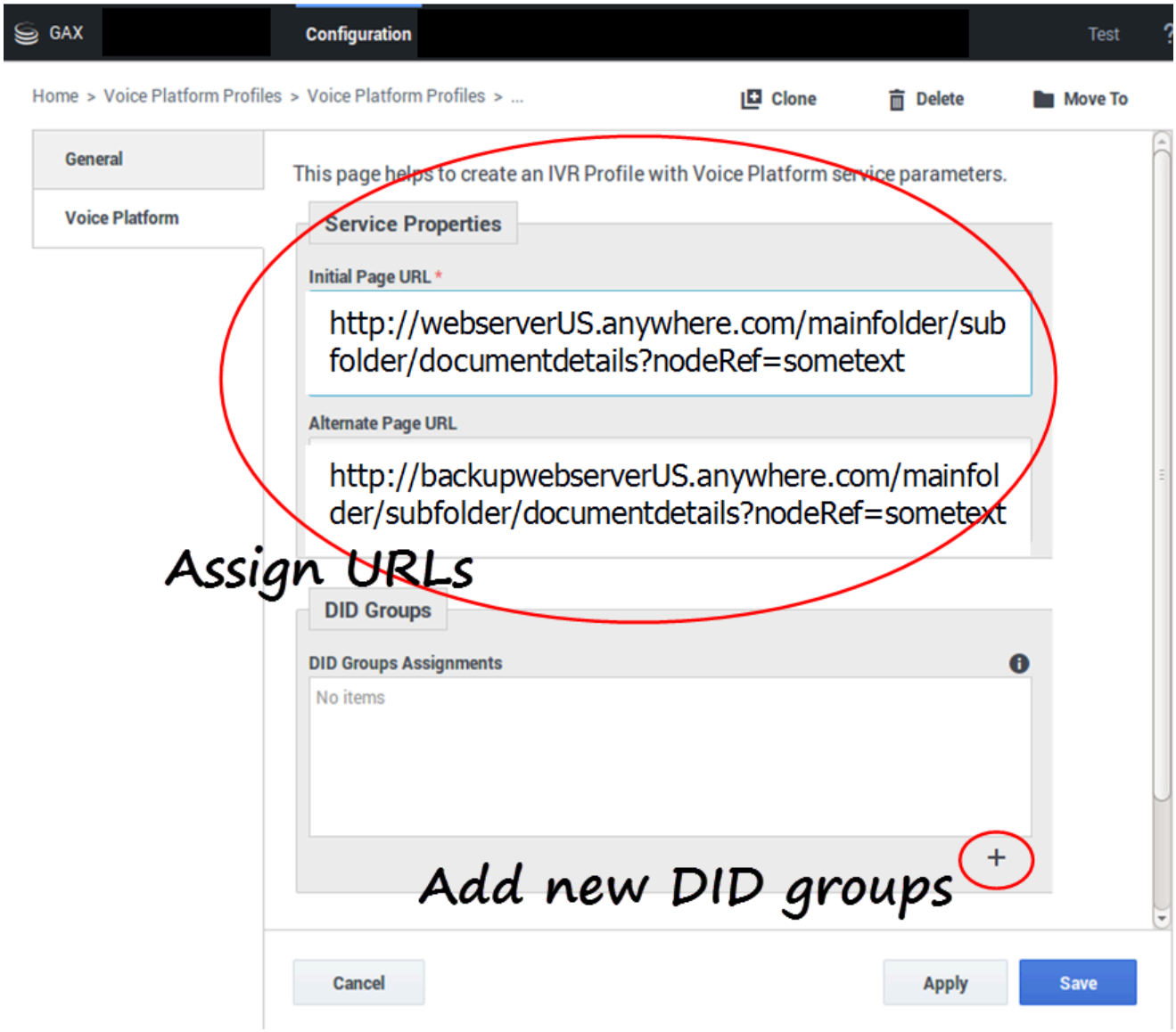
How to provision IVR profiles



Log in as a tenant and select the Voice Platform Profile icon on the Main Menu page.

- The Voice Platform Profiles Main Page lists all existing IVR profiles that belong to you (the logged-in tenant).
- The Edit, New, Delete, and More commands (above the list) act on the IVR profile(s) that you select.

How to create new IVR profiles



Creating a new IVR profile requires completing two forms.
Click **New** and be ready to enter information that the forms need:

On the Service Properties page:

- Initial page URL (only http, https and file protocols supported)
- Alternate Page URL (only http, https and file protocols supported)
- DID Groups
  - Add to the list—click the **+** button (plus) near the bottom right.

  - Remove from the list—roll over the item with the cursor and click the **X**.
- **Save** lets you name the profile and moves you to the IVR profile listing page.

On the Available DID Groups page:

- Select the DID groups to assign and click **OK**. Your selections appear in the DID Groups Assignments list.
- **Save** also returns you to the VP Profiles Main Page, which now includes your newly created / newly assigned DID groups in the list.

Personalities

> **Important**
>
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

A Personality is a set of attributes and Audio Resource Files used for human interactions in your contact center.

For example, you might have a personality named `Jim`. This personality might have several Audio Resource Files that are voice recordings from one person, such as:

- A greeting that is played when customers call the contact center.
- A listing of menu options that is played after the greeting.
- Conditional messages to be played on holidays.

You can have multiple Personalities in your contact center. For example, you might use the `Jim` Personality for English-speaking callers, and you might use a Personality named `Juanita` for Spanish-speaking callers.

The Personality List screen displays a list of all Personalities available. Click a Personality in the list to see its properties displayed in a panel that opens to the right.

You can filter the contents of this list in two ways:

- Type the name or partial name of an object in the `Quick Filter` field.
- You can sort the items in the list by clicking a column head. Clicking a column head a second time reverses the sort order.

Audio Resources

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

An Audio Resource is a grouping that consists of one or more Audio Resource Files. Each Audio Resource File consists of one audio file that is associated to one Personality.

Audio Resources are referenced by routing strategies and played back to customers via a Media Server.

Click an Audio Resource in the list to display its properties in the <audio resource name> panel that is displayed to the right of the Audio Resources screen.

You can filter the contents of this list in two ways:

- Type the name or partial name of an object in the `Quick Filter` field.
- You can sort the items in the list by clicking a column head. Clicking a column head a second time reverses the sort order.

Modifying

Modifying an Audio Resource

- In the header, go to `Operations > Audio Resources > Audio Resources`.
- Select the Audio Resource that you want to modify.
- Modify the properties of the Audio Resource, as required. (**Note:** Only the Name and Descriptions properties can be modified.)
- Click Save.

Audio Resource Files

> **Important**
>
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

Audio Resource Files are contained in Audio Resources, and consist of an audio file and a unique Personality. You create and manage Audio Resource Files by using the <audio resource name> panel that is displayed to the right of the Audio Resources screen after you have selected an audio resource.

Audio Resource Files are created when an audio file—such as a WAV (.wav) file—and a Personality are combined and assigned to an Audio Resource. Therefore, all composite audio files, Personalities, and Audio Resources, must have been created before you can create the Audio Resource Files.

To view Audio Resource Files, complete the following steps:

- In the header, go to `Operations > Audio Resources > Audio Resources`.
- In the `Audio Resource List` screen, select an audio resource.
- Click the `Files` button or select `Files` from the `Related` menu. The `Audio Resource Files List` panel is displayed.

The `Audio Resource Files List` panel displays a list of Audio Resource Files that are already associated with the selected Audio Resource. For each Audio Resource File, the name of its underlying audio file, Personality, size, ID, and status are displayed.

To display the properties of the audio file, click an audio file in the list. The properties of the associated Audio Resource File is diplayed in the <audio file name> panel that will be displayed to the right of the `Audio Resource Files List` panel.

You can filter the contents of this list in two ways:

- Type the name or partial name of an object in the `Quick Filter` field.
- You can sort the items in the list by clicking a column head. Clicking a column head a second time reverses the sort order.

Properties

Viewing the Properties of an Audio Resource File

Properties of Audio Resource Files

Property	Mandatory	Description
Name	No	The name of this Audio Resource File.
Language	No	The language that is attributed to this Audio Resource File by the Personality
Personality	Yes	The personality that is assigned to this Audio Resource File.
Size	No	The size of the Audio Resource File, in bytes. This field appears only after the Audio Resource File is created, and it cannot be changed.
Status	Yes	Information about the availability of the Audio Resource File.

Parameters

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

Parameters are used to customize Routing Strategies. For example, you can use a `Schedule` parameter to define the opening and closing hours of your contact center. Other parameters allow you to perform tasks such as modify call flows or insert holiday greetings on defined days.

Parameters are grouped by Genesys into a collection of parameters known as a Parameter Group Template. Genesys deploys the Parameter Group Template to you, and then you can customize the values of the parameters in the template. These parameters are then read by a Routing Strategy and incorporated into the call flow.

The Parameters screen displays a list of all parameters that are available to you. Click a parameter in the list to see its properties displayed in a panel that opens to the right.

You can filter the contents of this list in two ways:

- Type the name or partial name of an object in the `Quick Filter` field.
- You can sort the items in the list by clicking a column head. Clicking a column head a second time reverses the sort order.

Note: To modify parameters within a parameter group, see Parameter Groups.

Parameter Group Templates

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

Parameter Group Templates are sets of Operational Parameters that can be deployed to one or more tenants, and are defined by the Service Provider. A parameter can be grouped into more than one Group Template. You can also group sets of parameters into sections within a Parameter Group Template to enable you to create Parameter Group Sections within your parameter groups.

Overview

Overview

This screen displays a list of all defined Parameter Group Templates for which you have the required role privileges to view. To refresh the list at any time, click Refresh. Click a Parameter Group Template in the list, and its properties are displayed to the right of the list.

You can filter the contents of this list in several ways:

- Type the name or partial name of the Parameter Group Template, In Use, or Tenant in the Quick Filter field.
- Click the Tenant Filter button (the icon with the circle and horizontal bar) to open the Tenant filter panel. In this panel, click the checkbox(es) beside the tenants that you want to select. Use the Quick Filter field in this panel to filter the tenant list.
- You can sort the Parameter Group Templates in the list by clicking on a column head. Clicking a column head a second time reverses the sort order.

Properties

Parameter Group Template Properties

Properties of Parameter Group Templates

Property	Description
Name	The name of the Parameter Group Template. It must be unique in the system.
Description	Optional text describing the Parameter Group Template or providing additional information.
Parameters	A list of Operational Parameters that have been added to the Parameter Group Template. The Section(s) to which this Parameter Group Template belongs is also displayed here. **Note**: Sections can be deleted. Deleting a section only deletes the section grouping, and does not delete the set of parameters previously within that section.
Used	(Read-only) Used to deploy a parameter group.

Parameter Groups

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

Parameter Groups are sets of Operational Parameters that are associated with a Routing Strategy. They are deployed as Parameter Group Templates by the Service Provider. The administrator then assigns values to the Operational Parameters in the Parameter Group. When the URS application executes a Routing Strategy, the values of the Operational Parameters in the associated Parameter Group are incorporated into the call flow.

Overview

Parameter Groups Overview

This screen displays a list of all Parameter Groups for which the logged-in user is associated, and for which you have the required role privileges to view. To refresh the list at any time, click Refresh.

Filter

Filter Parameter Groups

You can filter the contents of this list in several ways:

- Type the name or partial name of the Parameter Group, Parameter Group Template, or Flag in the Quick Filter field.
- Click the `Filter` button (the icon with the circle and horizontal bar) to open the filter panel. In this panel, click the checkbox(es) beside the tenants that you want to select. Use the Quick Filter field in this panel to filter the list.

- You can sort the Parameter Groups in the list by clicking on a column head. Clicking a column head a second time reverses the sort order.

GVP PaaS

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

Genesys Voice Platform (**GVP**) is a software-only, standards-based voice portal that provides cost-effective customer interactions, 24x7, for businesses using voice, video, the web, and the cloud.

Functioning beyond traditional interactive voice response (IVR) systems, GVP provides touch-tone access to applications and incorporates optional speech recognition technology and video for conversational exchanges, better to identify and resolve customer requests.

Related Topics

GVP PaaS (Platform as a Service) is an Inbound Voice offering wherein the application or call-flow is installed on the customer premises and the incoming telco connection is terminated at the computer (GVP) in the Genesys Engage cloud. Customers can manage the application or call-flow entirely by themselves (or, with Genesys assistance), and rely on Genesys to manage the complex infrastructure and high-availability associated with the telco connection.

Once your application or call-flow is deployed to your on-premises application servers, create an IVR profile and a Direct Inward Dialing (DID) group, and link them together before calls are made into the PaaS IVR. For steps on how to do this, see the **Related Topics** section above.

GVP PaaS also includes the following features:

- Historical Call Browser: GVP PaaS provides tenant-based access to the GVP Historical Call Browser on the Production platform. For more information, see Call Browser Report Filters.
- Call Dashboard: GVP PaaS provides per-tenant access to the GVP Call Dashboard. For information, see Dashboard Report Filters.
- Operational Reports: GVP PaaS provides per-tenant access to GVP Operational Reports (peak usage, arrivals). For more information, see Operational Report Filters.

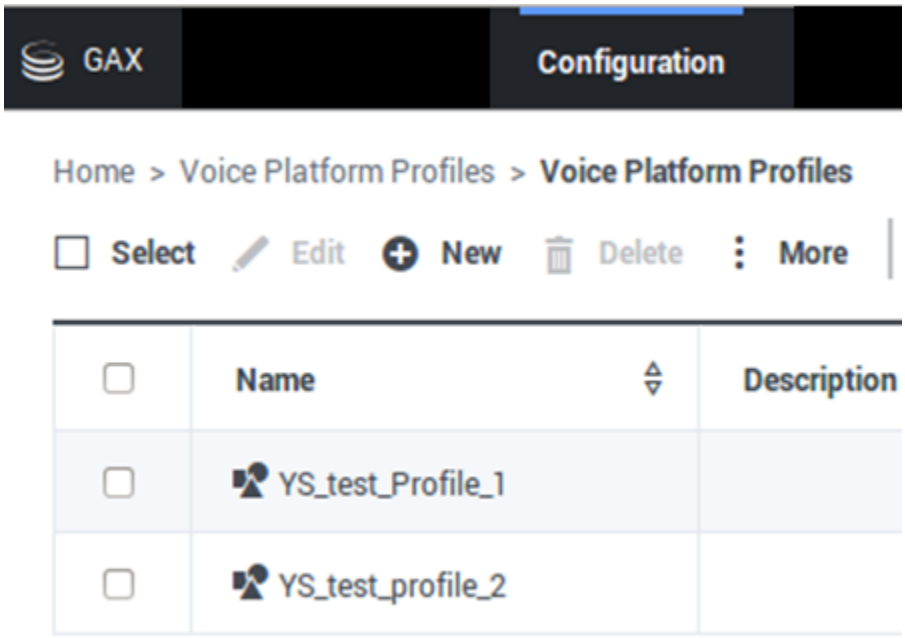
Provisioning IVR profiles yourself

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

You can use Platform Administration to provision IVR profiles and Map Direct Inward Dialing (DID) Groups to IVR profiles. You don't have to ask a manager or a system administrator.

Related Topics

How to provision IVR profiles

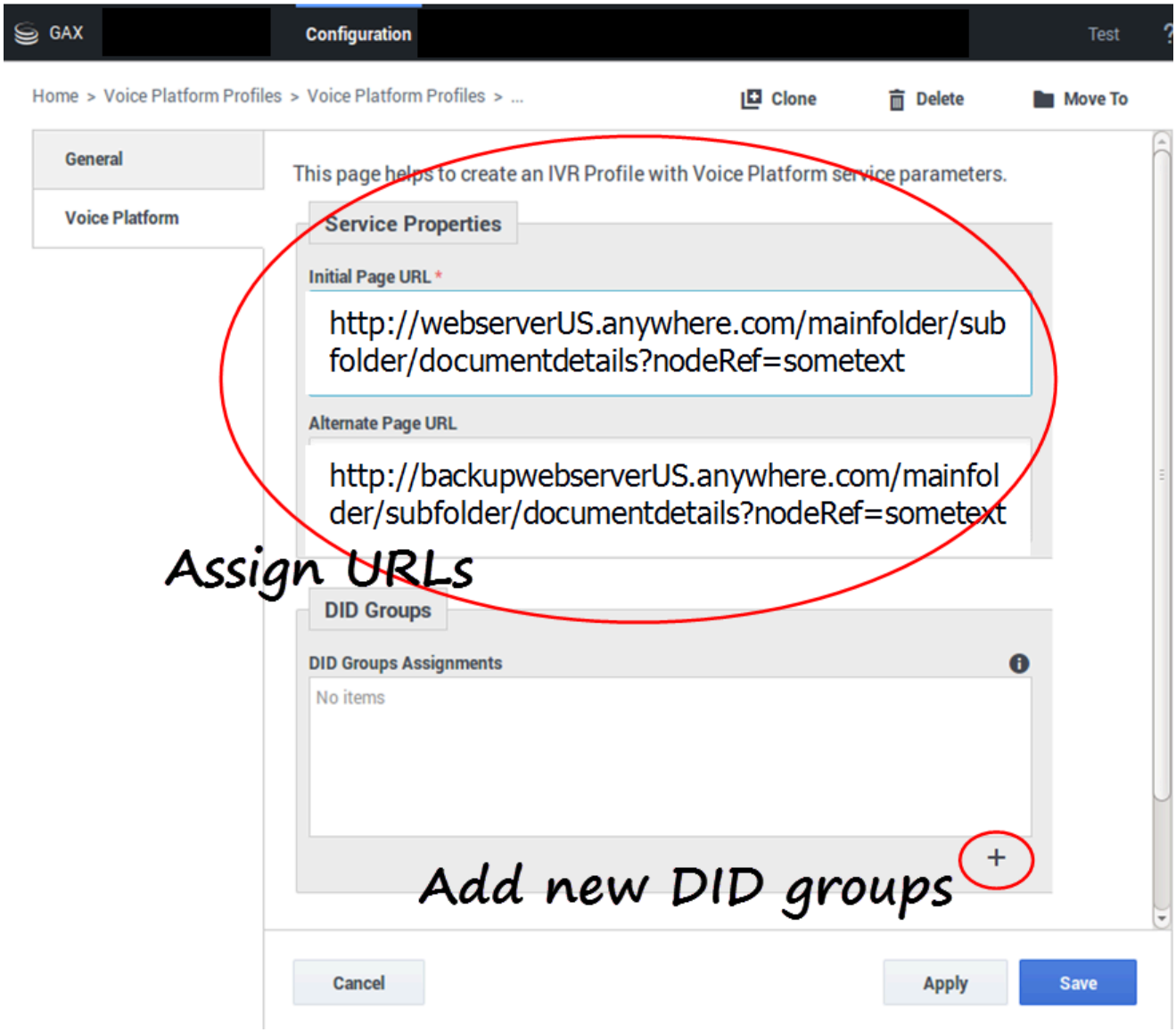


Log in as a tenant and select the Voice Platform Profile icon on the Main Menu page.

- The Voice Platform Profiles Main Page lists all existing IVR profiles that belong to you (the logged-in tenant).

- The Edit, New, Delete, and More commands (above the list) act on the IVR profile(s) that you select.

How to create new IVR profiles



Creating a new IVR profile requires completing two forms.
Click **New** and be ready to enter information that the forms need:

On the Service Properties Page

- Initial page URL (only http, https and file protocols supported)
- Alternate Page URL (only http, https and file protocols supported)

- DID Groups
  - Add to the list—click the **+** button (plus) near the bottom right. An Available DID Groups dialog will pop up.
  - Remove from the list—roll over the item with the cursor and click the **X**.
- **Save** also moves you to the IVR profile listing page.

On the Available DID Groups Page Popup Dialog

- Select the DID groups to assign and click **OK**. Your selections appear in the DID Groups Assignments list.
- **Save** also returns you to the VP Profiles Main Page, which now includes your newly created / newly assigned DID groups in the list.

Creating and managing DID Groups

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

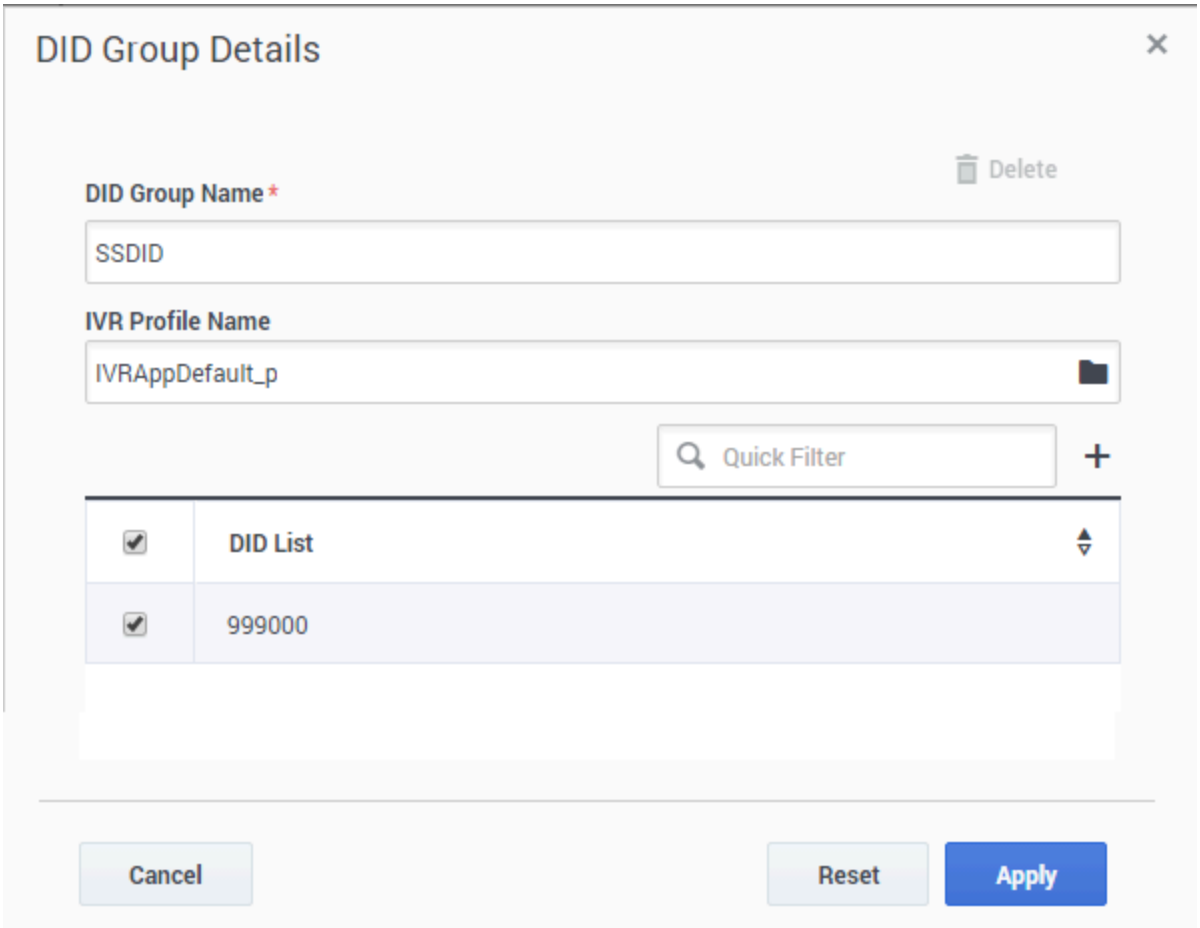
All procedures begin in the DID Groups Main Page. To get there:

1. Log into Platform Administration as a tenant user.
2. Select DID Groups from the Administration menu.

At the end of each procedure, click **Apply** to save changes.

Related Topics

Creating a new DID Group

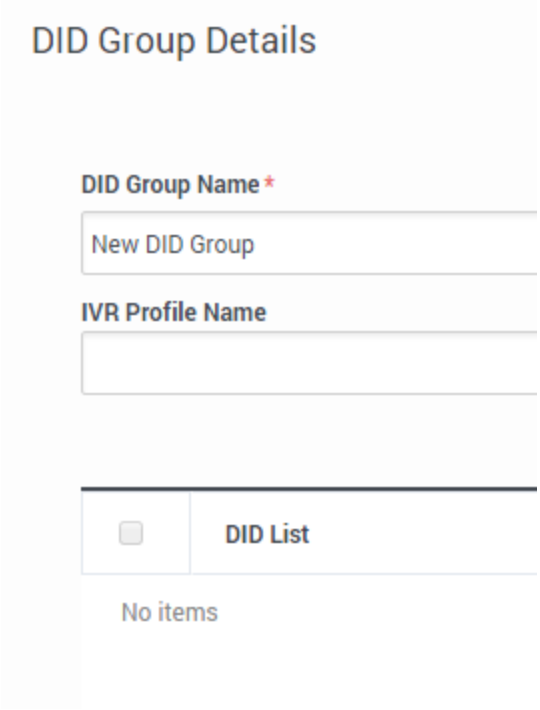


1. Click **New**.
2. Configure the DID Group Details dialog:
   - Enter an unused DID Group Name and an existing IVR Profile Name.
   - Check the selection box of each DIDs in the DID List field, to include those DIDs in the group you are creating.

     Optional Quick filter field: enter text to limit the number of DIDs displayed, or click plus (+) to query for additional DIDs not displayed.

When you leave this dialog, the DID Group that you created is now listed in the DID Groups directory.

Modifying an existing DID Group



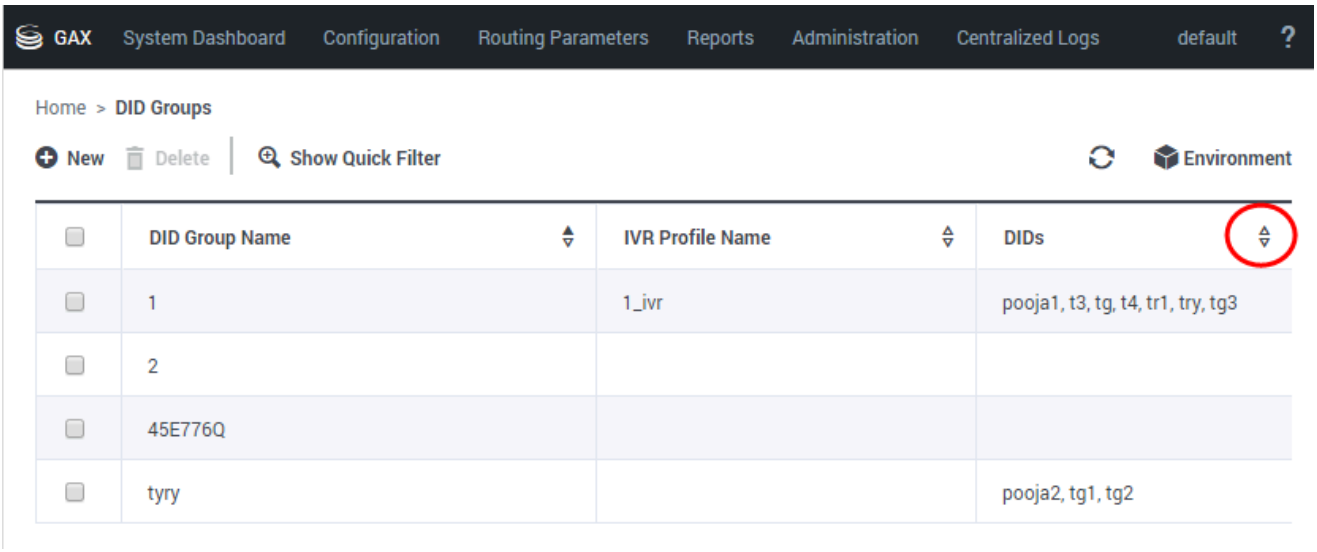
Removing items from the group

1. Click on the name of an existing DID Group to open its DID Group Details dialog.
2. Uncheck individual items to remove them from the DID group.

Provisioning conflicting entries

- DID group provisioning succeeds when you remove the conflicting entries in overlapping ranges (example: 100-200 and 150-250).

Sorting an existing DID Group



1. Click on the name of an existing DID Group to open its DID Group Details dialog.
2. Click the arrowhead (a triangle) at the top right of the DID List field to sort it in the direction of that the arrowhead points (up=descending order, down=ascending order).

The list of DIDs in the group is sorted. You can re-sort the list, but you cannot unsort it.

Outbound

> **Important**
> This is legacy Outbound content. For the latest Outbound functionality, refer to CX Contact.

This brief Introduction provides an overview of Outbound functionality and is intended to help administrators and supervisors become familiar with key concepts before accessing the full documentation.

Related Topics

Welcome

The Outbound applications enable administrators and supervisors to administer and manage Outbound campaigns. Through an easy-to-use, intuitive web-based user interface, users can quickly and easily create and manage comprehensive outbound contact strategies, manage multiple contact lists, and create and apply compliance rules.

About this document

This document provides information about managing Outbound campaigns, from the perspective of an enterprise manaeger or an account manager.

Select the Enterprise Manager documentation as you use your Enterprise account to configure compliance rules and create campaign templates that can be used by the other accounts in your enterprise.

Select the Account Manager documentation as you use your User accounts to configure and manage your calling campaigns.

Outbound Applications	Getting Started
Enterprise Manager	Outbound Solutions

Account Manager

Outbound Business Scenario

Set Up Outbound Routing

Load a Sub-campaign and Make a Call

Articles and Resources

Multi-Channel Campaigns

Outbound Dialing Modes Explained

Salesforce Adapter for Outbound

Latest Videos

Link to video Link to video

Outbound Solutions

The Outbound solution is a fully customizable Cloud solution that enables you to set up, run, and fine-tune your Outbound campaigns.

Through a series of Genesys Engage cloud products, you can define your routing application, set up a campaign and sub-campaign, apply compliance rules, and monitor and assess your campaign through real-time and historical reports.

Related Topics

And you can do all of this in four stages:

Stage 1—Configuring Outbound routing
Stage 2—Setting up a campaign
Stage 3—Dialing and call handling
Stage 4—Monitoring a campaign

Stage 1—Configuring Outbound routing

Genesys will configure Outbound routing for you! Here’s what we’ll do:

We’ll create and configure the Routing Point DN, the Virtual Queue DN, and the agent groups using Platform Administrator. Then, we’ll head to Designer to tie it all together.

If you’re new to Designer, here’s all you need to know: It’s a powerful yet easy-to-use web-based tool for developing IVR and Routing applications that run on the Genesys platform.

And here’s what we’ll do with it: Basically, we’ll create the Outbound routing application so that you can route calls to agents. We’ll drag and drop blocks from a menu (we call it a Palette) into one of the four phases that define your IVR (we call it the Application Flow), and then we'll choose from a few routing options. We'll publish it and then hand it off to you.

After that, you can pop into the applications at any time to tweak the settings. Here are a few instructions:

- How to set up Outbound Routing
- How to create and modify DNs
- How to set up agents
- How to add the Route Call Block in Designer

Want to learn more about Designer or Platform Administration?

- Designer help

- Platform Administration help

Stage 2—Setting up a campaign

OK, now it’s time to head to Engage to set up your campaign. Let’s cover a few basics first.

First – the difference between a campaign and a sub-campaign.

Think of a campaign as the foundation of a house – the starting point. Every inch of the house is built in reference to it and depends on it in order to stand above ground.

And now think of a sub-campaign as the house.

Simple as that: a campaign is the foundation that the sub-campaign depends on. The campaign holds the core information – like compliance rules, strategies, templates, etc. – that the sub-campaign needs in order to run.

It's also important to know that if you're running a sales campaign with more than 15 agents assigned to the campaign, you need to run it in ASM mode with Predictive dialing. This is the only way you can meet the 2-second connect standard enforced by regulatory bodies such as Ofcom and FTC. In other words, when the system detects a voice on the line, it will bridge the call to an agent within two seconds. We’ll talk a bit more about that later on.

Campaigns

Now, let’s set up a campaign. Basically, your Engage tasks are broken down into three components:

1. Contacts and contact lists
2. Compliance
3. Campaign configuration

Contacts and contact lists

Think of a contact list as a phone book. Page by page, it displays the contact information of everyone in a particular group or community. That contact information could contain a name, phone number, email address, home address, or an ID number.

When you pick up a phone book and say “I want to call every person in this book,” those people become part of your Outbound sub-campaign. And that’s exactly what happens when you select a contact list within Engage.

Before you tell the system to start calling those people, however, you can do a few things with your list:

- Create or import a contact list and contacts—create or import a new list or append to an existing one
- Edit or delete a list—except for any list named ‘Unassigned’
- Export or Download a list—as a CSV file

- Filter a list—specify the name, description, and selection rules and definitions
- Generate a report of a list— specify output options such as FTP destination, file name format, specification file, and compression and security options.
- Edit contacts—the changes affect new sub-campaigns only, not those currently running or scheduled to run
- Delete contacts—permanently removes them from the list

After you've imported a list, you can go to the **Lists** tab on the main menu of the **Campaigns** page and do any of the following:

- Apply list rules—to split a list into smaller lists by quantity, percentage, field, or custom criteria
- Import specification files—to tell the system how to structure your data
- Perform a contact search—to search for a single contact, by name, from within the contact lists

Learn more about contacts and contact lists here

Compliance

Now choose the people you'll exclude from your calling campaigns – both running and future campaigns. You do this for two reasons: to adhere to industry standards and regulations set out by FTC and OFCOM, and to honor your customers' requests.

Choose **Compliance** from the main menu of the **Campaigns** page and you'll see two options: **Compliance Tools** and **Contact Suppression Lists**.

Compliance tools
This is where you'll make the exclusions based on geographical rules. The **Do Not Contact Location Rule** enables you to suppress contact attempts to listed locations using specified channels.

Contact suppression lists
Think of it as another phone book, but consisting of people you should never call.

Here's what you can do with a suppression list:

- Create a new suppression list – specify the list name, channels (Voice, Text, Email), expiry date (if any), and the type of list (Device or Client ID).
- Import a suppression list—specify a file format, a destination for your contacts (create new list or append to an existing one), and the suppression list attributes you specify when you create a new list.
- Add contacts to a suppression list—enter the list of client IDs or devices, depending on the list type.
- Append contacts to an existing suppression list—choose Append to List from the options
- Clear contacts from a suppression list—you clear the contacts but the list attributes—such as Type, Channels and Expiration—remain visible
- Edit or delete a suppression list
- Export a suppression list, to Desktop or to FTP—a CSV text file is retrieved from the database that contains all records

More on compliance tools:

- Enterprise Managers click here
- Account Managers click here

More on contact suppression lists:

- Enterprise Managers click here
- Account Managers click here

Campaign configuration

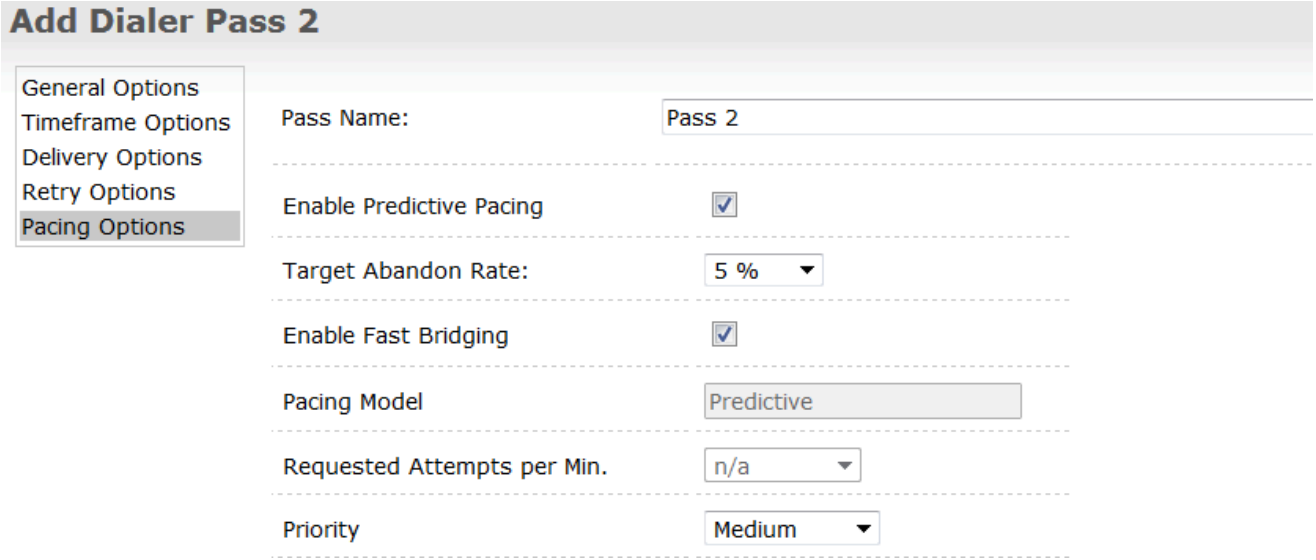
Campaign configuration is easier than it sounds. Basically, you'll go through a series of options and choose the ones that work best for your campaign.

The starting point is the Campaigns page – you'll land there when you log into the Account Manager. It's where you'll find details about the status of your campaigns, and it's where you'll set up a new campaign and create and run a new sub-campaign.

The **Campaigns** tab on the main menu includes links to the pages where you'll define some of the important campaign settings:

Click the **Campaigns** option from the menu to define some of the foundational stuff:

- Create the campaign template—enter the name, description, time frame, start and end date (if applicable) and the On Demand Strategy.
- Define the campaign strategy—select a pre-defined one or add a new one.
- Select Outbound Options—specify the script, agent group, contact order, timezone source, and constrain pass times.
- Select filtering and suppression options—specify phone number options, list exclusion options, contact suppression lists, and Do Not Contact location rules.
- Select pattern options—specify the escalation type and execution order.
- Choose a channel— select the channel/s you want to use: Voice, Dialer, Preview, Text, or Email. Note: if you choose a Text or Email channel,, you must specify a valid text or email script. See Scripts for more information.
- Select pass options—specify general, time frame, delivery, retry, and pacing options, where applicable.



This is where you'll specify the dialing mode and switching mode. For example, if you're running a sales campaign and using an automatic dialer, and you have more than 15 agents assigned to the campaign, you should enable Predictive dialing in ASM mode:

1. Click the **Outbound** tab
2. In the **Pattern Options** section select the **Dialer** pass option
3. Select **Pacing Options**
4. Check the box next to **Enable predictive pacing**
5. Check the box next to **Enable fast bridging**

Remember, doing this is the only way you can meet the 2-second connect standard required by FTC and OFCOM regulations (you'll learn how that happens in the next stage). Go here to learn more about Outbound dialing modes.

Now you can move onto these options – all available from the **Campaigns** tab on the main menu:

- Add, edit, change, or delete Caller ID numbers
- Set AutoManage Rules—set alert issues, contact center issues, text messaging or Email issues.
- View or disable a contact order—choose from a list of previously defined contact order strategies
- Copy enterprise campaigns—do this if you want the campaign to match or be similar to an existing campaign
- Strategies—edit an existing strategy or add a new one

Sub-campaigns

OK, you've done all you need to do to lay your foundation – or set up your campaign. Now go ahead and create and send an Outbound sub-campaign by doing this:

- Specify the sub-campaign information—the campaign it belongs to, contact list, list name, script, and agent group
- Choose from the available options—filtering and suppression options, contact order, timezone source, and an email address for summary reports

Need to know more?

- Learn more about campaigns
- Enterprise Manager help
- Account Manager help
- Scripts for SMS and Email

Stage 3—Dialing and bridging to an agent

Dialing

Once your campaign is running in ASM mode with Predictive dialing, our platform can intelligently bridge a call to an agent within two seconds of detecting a voice. This is how it happens:

First, it checks the routing application before it dials out. It wants to know what agent group is assigned to the campaign and it wants to check all other parameters that specify how it should route the call once it detects a voice.

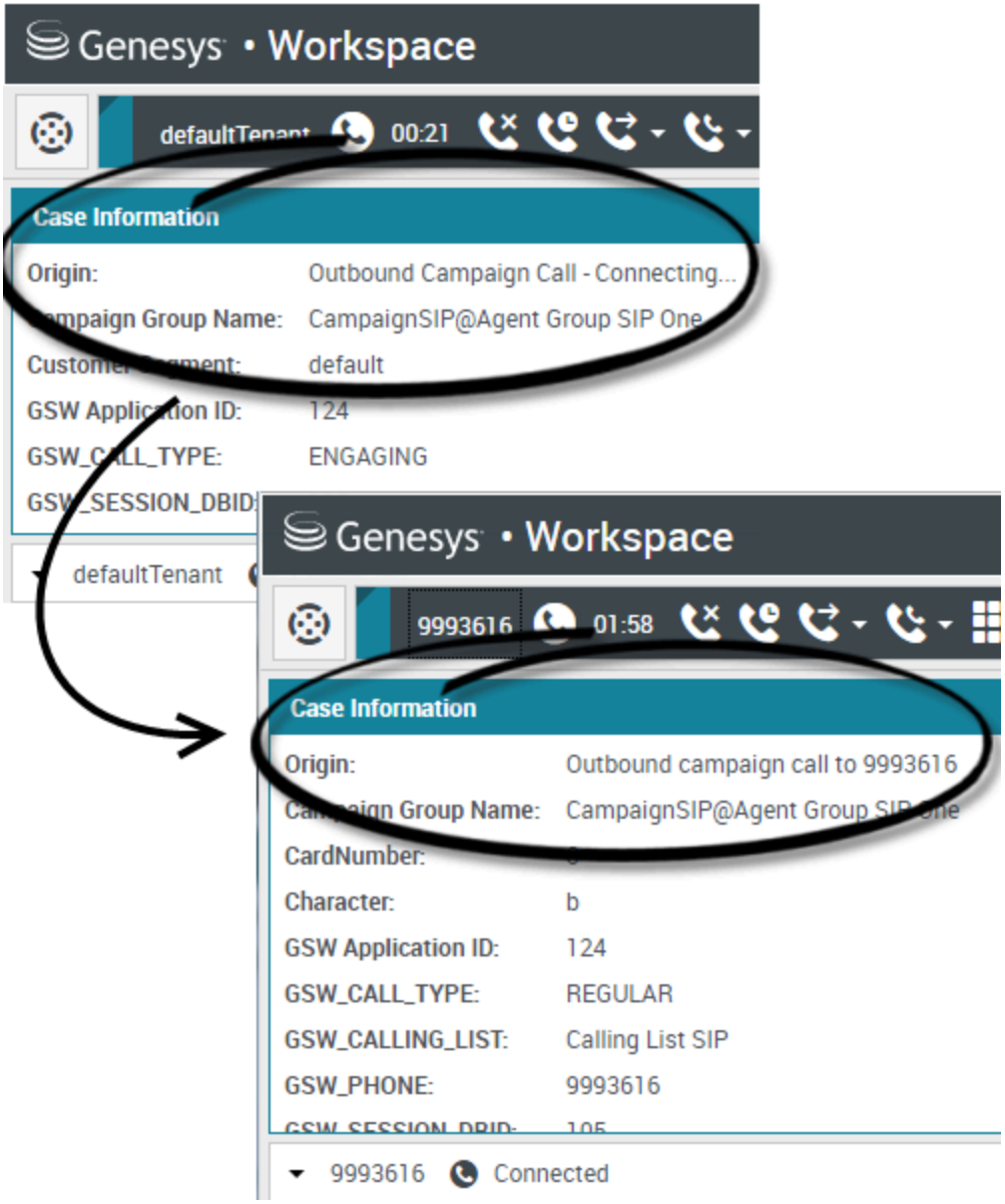
And then, it reserves an agent. By doing this, it commits the agent to a call once the agent completes all other queued-up calls.

Then it starts dialing.

Call handling

And now here’s where your agents come in.

Once a new campaign starts, the assigned agents get a notification via Agent Desktop. When an agent is Ready and reserved the phone rings, and the agent waits while the system dials a number. When the system detects a voice, the reserved agent hears a beep and is then connected to the customer.



In the meantime, in Agent Desktop, the agent sees a pop-up notification, and in the call **Case Information** areas, the **Origin** of the call is **Outbound Campaign Call - Connecting...**

As soon as the customer answers the phone and the call is connected, the **Origin** of the call changes to Outbound campaign call to **<name or number of contact>**. The agent might see information about the call in the **Case Information** area – such as campaign call information, the call status, and call actions.

Note: In some environments, an agent might not be connected to the Outbound call until after the contact has answered the phone. In this case, the agent will not see that the **Origin** of the call is **Outbound Campaign Call - Connecting....**; instead it will immediately show the name and or number of the contact.

Learn more here:

- Agents and Outbound campaigns
- Agent Desktop help
- Outbound Dialing Modes

Stage 4—Monitoring a campaign

You've set up and run your campaign and sub-campaigns. You now need to monitor the status of an ongoing campaign or look at the results of a campaign that has ended. This is where Genesys reporting tools come in.

Real-time reports

To monitor the status of an ongoing campaign in real-time, you have two options: the Engage dashboard and Genesys Pulse. Use the Engage Dashboard for a snapshot of call results, and use Genesys Pulse to generate in-depth reports about Agent activity.

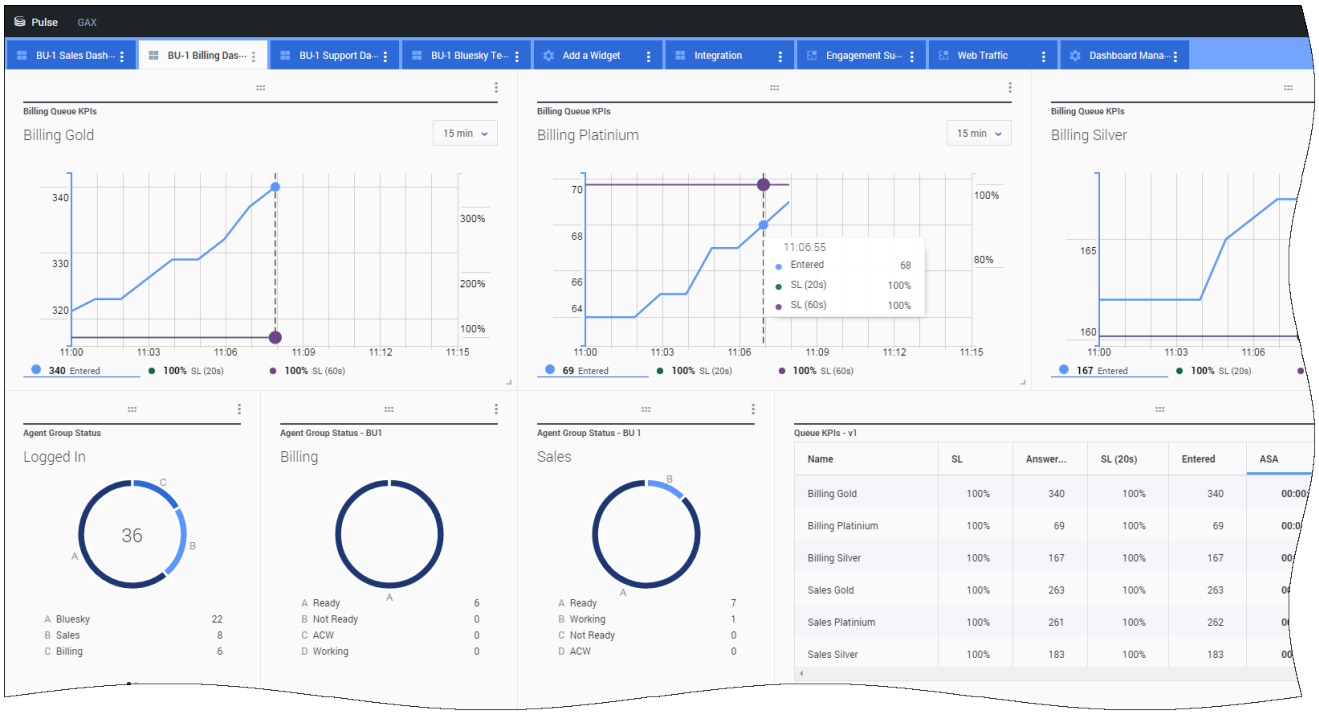
Engage dashboard

Go to the Campaigns page in Engage and you'll see that results of each call are updated in real-time on the dashboard:

- **Filtered**—The number of contacts filtered from the list.
- **Delivered**—The number of successful contact attempts.
- **Retrying or Failed**— The number of contacts being retried or with a current failure status.
- **Not Attempted** —The number of contacts remaining to be attempted.
- **Done**—The total number of contact attempts made.
- **Remain**—The number of contact attempts remaining to be made.

Genesys Pulse

Now go to Genesys Pulse (Real-Time Reporting) if you need more than just a snapshot of attempted calls. For your Outbound campaigns, refer to Genesys Pulse when you need a close look at Agent activity. If you're not familiar with the application, here's what it is, briefly:



It’s a plug-in application that provides real-time reports of a contact center’s statistics. You have a dashboard where you’ll see widgets that display the data you specified in the form of a grid, KPI, line chart, or list.

Basically, you'll add a report widget to your dashboard, choose a template or define your own, select objects and statistics, and specify default settings – like the name, refresh rate, and type of widget. And then you can save and download your report as a CSV file.

There’s also something called an IFRAME widget – this takes content from an external URL and displays it on your Genesys Pulse dashboard.

For more detailed instructions, go here:

Real-time Reporting help

Historical reports

Now you want to retrieve statistics of a campaign that has ended. OK, you have two options:

- Engage
- Genesys CX Insights

Engage

Through the Reports tab in Account Manager, you can run account, campaign, and sub-campaign summary and detail reports.

- **Account reports** provide a record of what happened across all campaigns in your account for the selected time period.
- **Campaign reports** provide a record of what happened over an entire campaign or for a range of dates within a campaign. The reports include combined information from all sub-campaigns that were run under the campaign. You can generate the reports for an entire campaign, a specified time period, or as an incremental report.
- **Sub-campaign reports** provide a record of what happened for a single sub-campaign. You can generate the reports for an entire sub-campaign or for the previous day's sub-campaign activity.

For more information, including a description of each available report, visit the Reports section of the documentation.

Genesys CX Insights

Genesys CX Insights uses data stored in Genesys Info Mart, and presents it in readable reports. There are several Outbound Engagement Reports:

Campaign Callbacks Summary report — displays a summary of information about callback activity, including the total number of callbacks processed by the contact center, broken down into the total number scheduled, missed, and completed for each day of the reporting period. Personal callbacks are distinguished from nonpersonal ones. The report's design internally filters the dataset to return Outbound voice-only interactions. Use this report to understand the frequency with which Callback was used in your Outbound campaigns, and the overall Callback success rates.

Campaign Summary report— summarizes key metrics, such as Accepted and Not Accepted, that illustrate the disposition of contact attempts associated with Outbound campaigns. The report also examines call-progress detection (CPD) efficiency. The report internally filters the dataset to return Outbound voice-only interactions. Use this report to understand the disposition of Outbound campaign contact dialing attempts; whether calls connected, were dropped, or failed (together with the reason for failure).

Contact List Effectiveness report—provides detailed information about the number of contact attempts that were generated by an Outbound campaign, the ratio of attempts that resulted in the detection of a special information tone (SIT), and a breakdown of the call results of those SIT-detected attempts for the selected calling list. The report internally filters the dataset to return Outbound voice only interactions. Use this report to determine which calling lists are working efficiently, and which need to be adjusted. The report contrasts, for each list, the number of outbound call attempts to the number of times the call failed to connect (a SIT tone was detected).

Agent Outbound Campaign report—provides total and average durations of call-handling activities (including Handle Time, Wrap Time, Preview Time, Engage Time, and Hold Time) for agents who participate in outbound campaigns. Use this report in conjunction with the reports in the Outbound Campaign folder to understand agent performance in your outbound campaigns, by reviewing total and average durations of call handling activities (including Handle Time, Wrap Time, Preview Time, Engage Time, and Hold Time) for each agent.
For more information, see:

- Report descriptions
- Get Started with Genesys CX Insights

Outbound Business Scenario

Now that you're familiar with the products and features that make up Outbound for Cloud, let's see how it *actually* works in a real-life business scenario.

Related Topics

Golden Gate Communications is a contact center that makes calls on behalf of G-Tel, a nation-wide telephone company in the United States. G-Tel wants to sell more long distance calling plans to existing customers and wants Golden Gate Communications to run an Outbound campaign to reach approximately 1,500 customers.

Harry is the administrator for Golden Gate Communications and has no experience with Genesys products or solutions. He'll handle most of the work, but the success of the campaign will depend on other key players. Here's how it will work out:

Stage 1: Configuring Outbound routing

- Key Player: Alex, a Genesys Administrator
- Genesys Engage cloud Products: Platform Administration, Designer

Stage 2: Setting up a campaign and sub-campaign

- Key Player: Harry, the Golden Gate Communications Administrator
- Genesys Engage cloud Products: Engage

Stage 3: Dialing and call handling

- Key Player: Martha, Sales Agent for Golden Gate Communications
- Genesys Engage cloud Products: Agent Desktop

Stage 4: Monitoring a campaign

- Key Player: Lucy, Manager at Golden Gate Communications
- Genesys Engage cloud Products: Engage, Genesys Pulse, Interactive Insights

Let's help Harry figure this out. Go to Stage 1

Stage 1: Configuring Outbound routing

- Monitoring

Given that Harry is new to Genesys Engage cloud products, Alex, a Genesys Administrator, will take care of the first step – routing configuration.

Using **Platform Administrator** Alex will create and configure the Routing Point DN, the Virtual Queue DN, and the Agent Group, and then assign agents to the group.

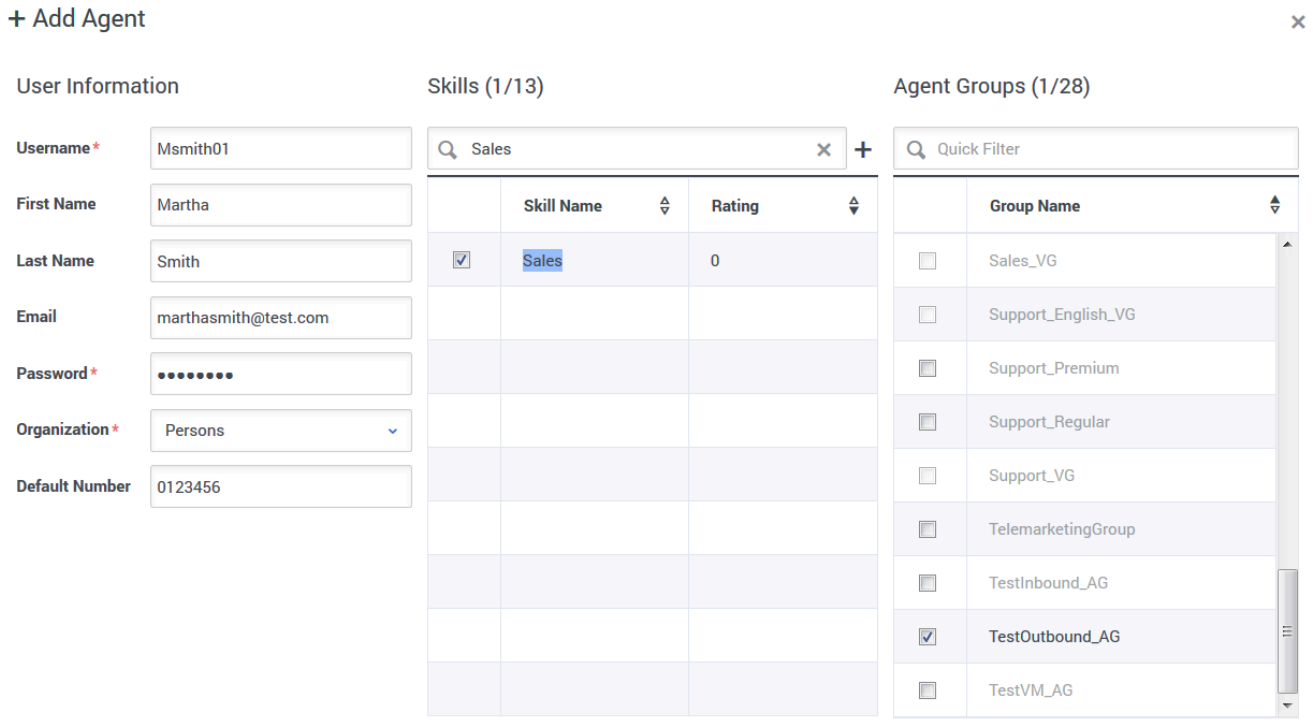
Then he'll build the IVR using **Designer.** He'll drag and drop blocks from the Palette into one of the four Application Flow phases that define the IVR. Then he'll hand it off to Harry.

Scenario: A week later, the contact center hired five new agents and asked Harry to add them and assign them to the Outbound campaign.

Solution: Even though Alex assigned the agents to an Agent Group in **Platform Administrator**, Harry can sign into the application and add or remove agents as needed.

[+] Instructions

Add Agents



From the Agents screen in Platform Administration:

1. Create the Agent account.
2. Assign one or more skills to the agent.
3. Add the new agent to an Agent Group.

Want to learn more?

- How to set up Outbound Routing
- How to add the Route Call Block in Designer
- Designer help
- Platform Administration help
- How to create a DN

Stage 2: Setting up a campaign and sub-campaign

- Monitoring

Now Harry will configure his campaign in **Engage.** To make it easy for him, we'll break his work down into four tasks with sub-tasks:

- Contacts and contact lists
- Compliance
- Campaigns
- Sub-campaigns

Contacts and contact lists

Import contacts

Scenario: Harry has a spreadsheet containing the names of 1,500 customers that will be contacted. He's busy and doesn't have the time to add each customer manually into **Engage**.

Solution: Rather than enter each contact's information directly into **Engage**, Harry can import the spreadsheet that contains the contact information of each customer. He'll do this in two steps:

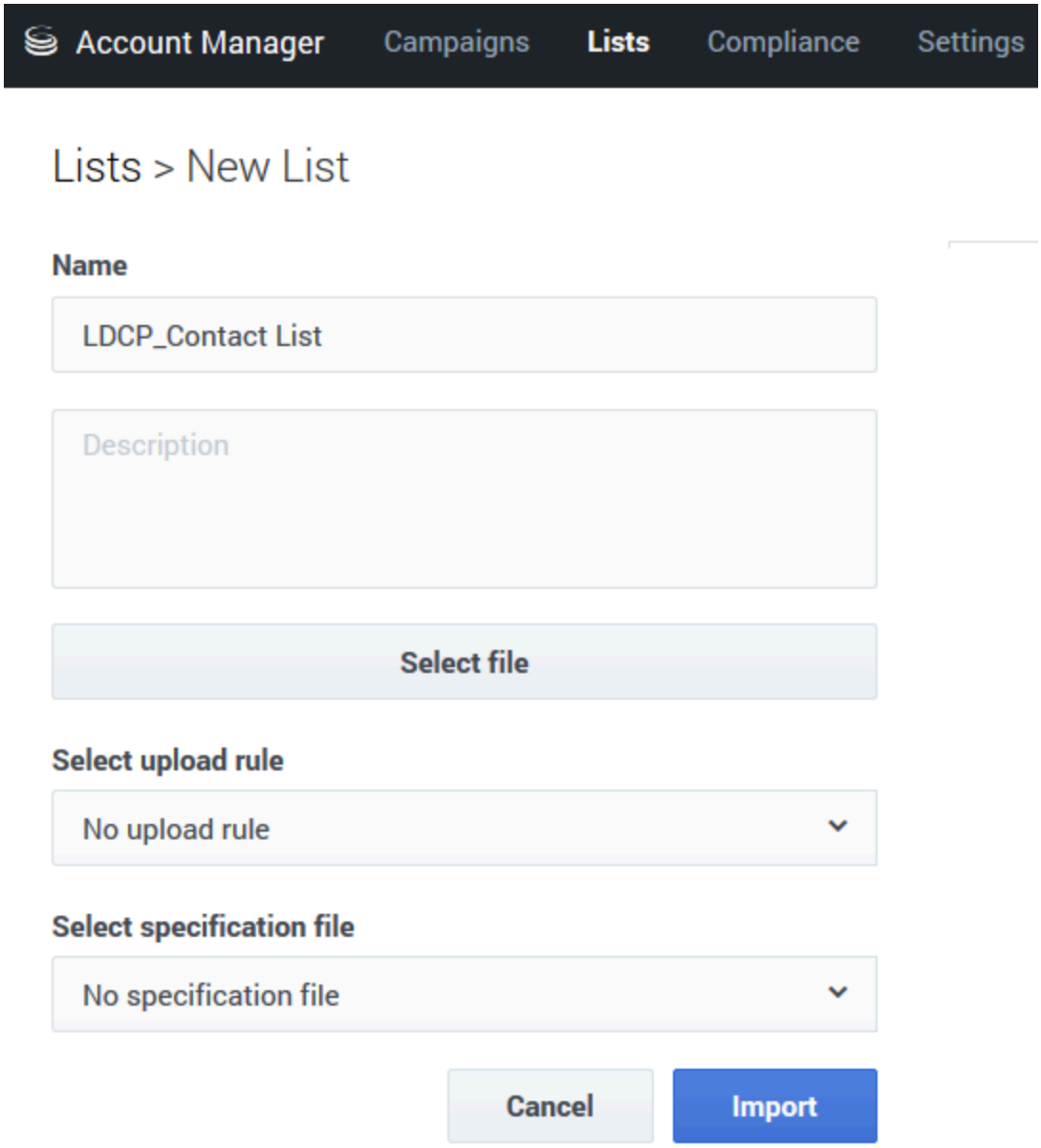
[+] Step 1 Instructions: Prepare data for import

First, he'll need to prepare the data for import so that it maps correctly in to the appropriate list fields in **Engage.**

1. Remove all unnecessary fields. Harry keeps only **FirstName**, **LastName**, **ClientID**, and Device1 **and removes all other fields.**
2. Include a header with the field names that match those in the messaging system contact list so that the imported data is mapped directly into the appropriate contact list field.
3. Save the spreadsheet as a CSV file.

[+] Step 2 Instructions: Import the contact list

Import contacts



Now, in **Account Manager**, Harry will import his contact list.

1. From the menu bar on the **Lists** tabbed page, click **Lists.**
2. From the **Lists** screen, select **New-->List.**
3. Type the name of the contact list (**LDCP_ContactList** in Harry's case) in the **Name** field.
4. Select **Browse** and select the file to be imported.
5. Harry doesn't want to split his list into smaller lists, so he'll select **No upload rule.**

6. Because Harry set up the header row in his spreadsheet, he'll select **No specification file**.
7. Click **Import.**

Once Harry imports his list in **Account Manager**, he can retrieve the list for this campaign and for future campaigns.

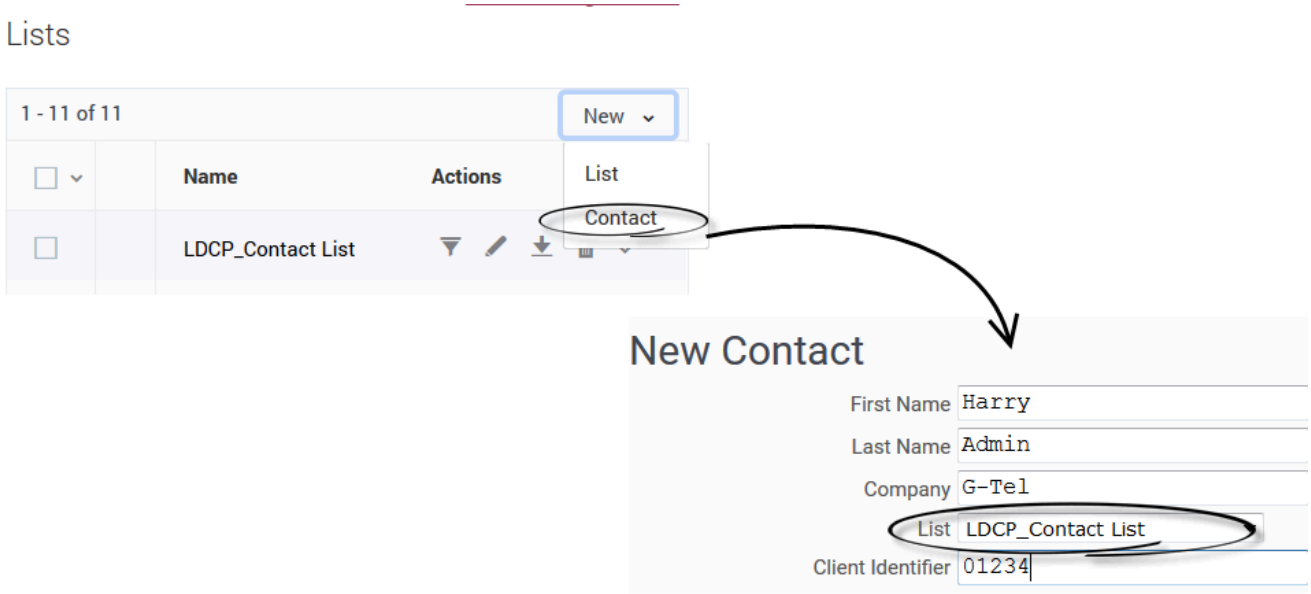
Add contacts to an existing list

Scenario: After Harry imported his spreadsheet, his account representative from G-Tel called with the contact information of seven customers they wanted added to the campaign. Harry confirmed that these customers were not listed on the spreadsheet he just finished importing.

Solution: Harry can manually add each of the seven customers to the imported contact list.

[+] Instructions

Add new contacts



1. On the **Lists** tabbed page, click **New** and then **Contact.** The New Contact page appears
2. Enter the contact information as required.
3. Select the list (**LDCP_ContactList** in Harry's case) to specify where the new contact will go.
4. Click **Save and Add New** to add the contact to the list and to continue adding the remaining six contacts.

Now that Harry has imported the contact list and added his additional seven contacts, he can edit, delete, export, filter, download, or generate a report of the list or edit or delete individual contacts.

He can also split his list into smaller lists using the Apply list rules function, and he can import specification files and perform a contact search. He's happy with the list and moves on.

Compliance

Now it's time for Harry to select the people he'll exclude from the campaign. When he clicks on the **Compliance** tab on the **Campaigns** page, he'll have two options: Compliance Tools and Contact Suppression Lists.

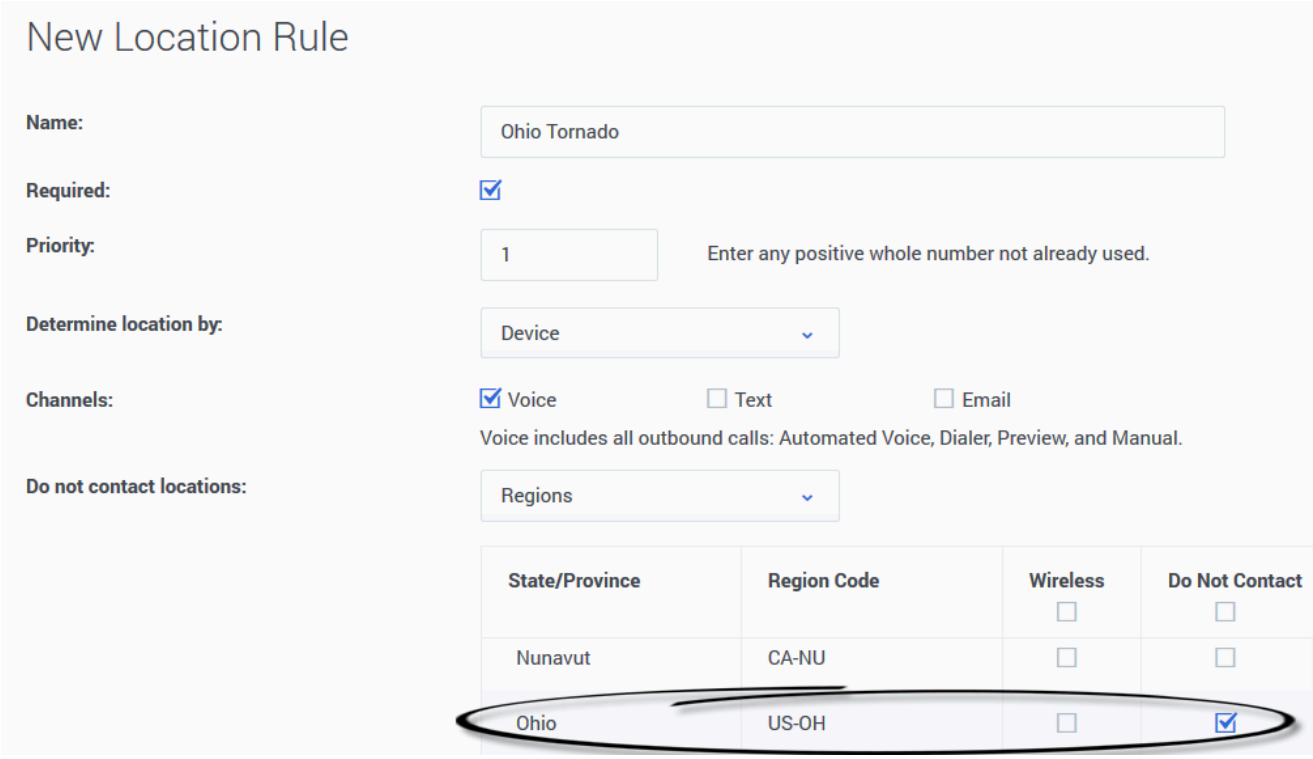
Compliance Tools

Scenario: Harry has also been told that Ohio is under a state of emergency as a result of widespread damage from a tornado. He wants to remove all Ohio customers from the calling list.

Solution: Harry can use the **Do Not Contact Location** rule to stop the system from calling Ohio.

[+] Instructions

Do Not Contact Location Rule



1. On the **Campaigns** tabbed page, click **Compliance** and select **Compliance Tools**.

2. Click **Location Rule.** The **Do Not Contact Location Rules** screen appears.
3. Click **New.**
4. Name the rule **Ohio Tornado**
5. Check the **Required** box.
6. Indicate the Priority of the rule by entering **1**.
7. From the **Determine Location By** drop-down list, select **Device** because the location will be determined by the area code of the device.
8. In the **Using Channel(s):** section, select **Voice.**
9. From the **Do not contact location:** drop-down, select **Regions** find **Ohio** and check the check the associated **Do Not Contact** box.

Contact Suppression Lists

Scenario: Harry has a spreadsheet containing the names of 50 people who have requested to be removed from all future calling campaigns.

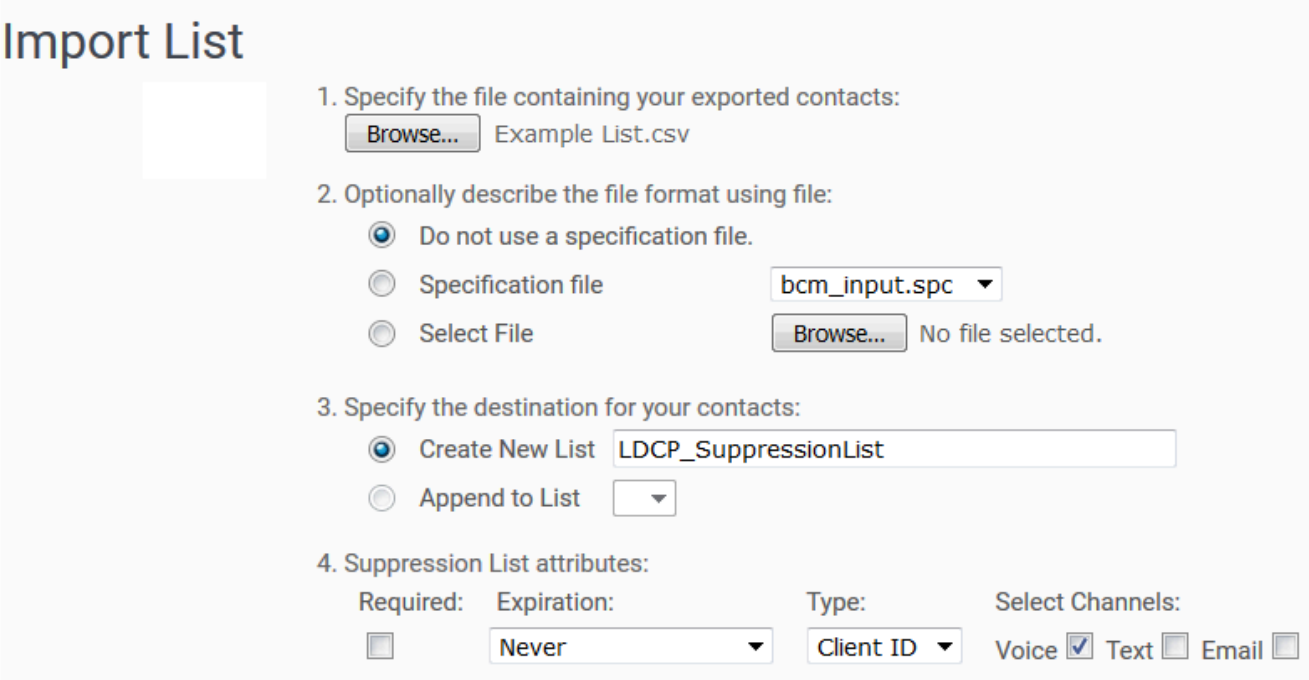
Solution: Harry will use the **Contact Suppression Lists** function to exclude those 50 people from the campaign. He'll do this in two steps:

[+] Step 1 Instructions: Prepare data for import

1. Remove all unnecessary fields. Harry keeps only **FirstName**, **LastName**, **ClientID**, and Device1 **and removes all other fields.**
2. Include a header with the field names that match those in the messaging system contact list so that the imported data is mapped directly into the appropriate contact list field.
3. Save the spreadsheet as a CSV file.

[+] Step 2 Instructions: Import the suppression list

Import Suppression List



1. Select **Compliance** -->**Contact Suppression List.**
2. Click **Add -> Import file.** The Import List page appears
3. Select **Browse** and select the file to be imported.
4. Select **Do not use specification file** because the header row in the spreadsheet was defined and matches the structure in the messaging system.
5. Select **Create New List** and type **LDCP_SuppressionList** in the **Name** field.
6. Select **Required:**, meaning that the suppression list is required for all campaigns in the account.
7. From the Expires: list, select **Never,** meaning the device or Client ID is never removed from the suppression list.
8. From the Type: list, select **Client ID.**
9. In the Select Channels field, select **Voice**.
10. Click **Next** to see a mapping page that shows all the fields found in the file. Confirm that the data is mapped appropriately and click **Next** again. A confirmation page will show the number of records available to be imported.
11. Click **Complete Import**.

If he wants to, Harry can now select from the following suppression list options: edit, delete, export a list, or add, edit or delete individual contacts. He'll leave it alone for now.

Campaigns

Harry can now configure his campaign using **Engage.**

Scenario: Because this is a sales campaign, Harry must abide by the FCC rule that says the contact center must connect a call to an agent within 2 seconds of detecting a voice.

Solution: In order to meet the 2-second connect standard, Harry will run his campaign in **ASM** mode, often referred to as **Fast Bridging**. And because Harry has only two weeks to reach 1,500 customers, with 25 agents assigned to the campaign, he chooses the fastest dialing mode, **Predictive**.

[+] Campaign details

The starting point is the **Campaigns** page in **Account Manager**. He'll set up a new campaign and create and run a new sub-campaign.

Select **New-->Campaign** from the top of the **Campaigns** page, he'll provide basic details about his campaign.

- Campaign Name: **Long Distance Calling Plan**
- Campaign Description: **Outbound campaign to sell more long distance calling plans to existing customers.**
- Start Date and End Date: **July 1 to July 15**

[+] Outbound options

Now, in the **Outbound** tab, he'll specify **Default Single Pass** as his strategy name and he'll use the following settings for his **Predictive** dialing campaign:

- Script: **dialer_script**
- Agent Group: Select **TelemarketingGroup*. Alex set this up in*** *Platform Administrator*'.
- Timezone source: **Device**
- Constrain Pass Time: Since Harry's campaign will run seven days a week for the campaign period from 9AM-9PM, he will select 9:00 in **Never start passes before:** field and he'll select 21:00 in **Always end passes by:** field. He'll select his timezone America/New York.

[+] Filtering options

In this section Harry will only need to add the contact suppression list he uploaded earlier. The compliance rules he defined in **Enterprise Manager** should appear in this section.

[+] Pattern options

Escalation Type: Select **None** because escalation is not required in this campaign. Execution Order: **Sequential**, meaning that supplemental passes to a contact will not be executed until at least one attempt on every contact in a list has been made in a previous pass.

[+] Define a pass

In the **Pattern Options** section, Harry will select **Dialer Pass** and then ensure the settings are as follows:

General options:

- Pass Name: **Pass 1**
- Accept Phone Numbers: **Landline** and **Wireless**

If Harry later added a second dialer pass, he would configure the added options as follows:

- Accept Contacts From: **All Passes**
- Last Voice Messaging result was: Select the following: **Not Attempted, Hang up on Machine, Not Connected, Busy, Undelivered Machine, Bad Device, No Answer**
- Last Text Messaging result was: **Not Attempted**
- Last Email Messaging result was: **Not Attempted**
- Last Script result was: Leave blank

Timeframe options

- Days to Wait: **0**
- Allow Attempts on These Days: Select all
- Start time: **9:00 Local**
- End Time: **21:00**

Delivery options

- Bridge Calls to Agent when: **Live Party Detected**
- Caller ID Number: **Account Default**
- Deliver to answering machines: leave unchecked

Retry options

- Retry Busy Calls: **0 times with 5 minutes between attempts**
- Retry Attempts: **0 times with 30 minutes between attempts**
- When Call Result is: leave options unchecked

Pacing options

- Enable Predictive Pacing: Check
- Target Abandon Rate: **5%**
- Enable Fast Bridging: Check

- Priority: **Background**

Remember, enabling **Predictive Pacing** and **Fast Bridging** is the only way to meet the 2-second connect rule set by FCC regulations.

Add a default Caller ID Number

Scenario: The Caller ID number that will display on a contact's phone is currently defaulted to the phone number associated with Harry's account, but Harry wants to specify a different phone number.

Solution: Harry can change the default Caller ID number to any specified phone number.

[+] Instructions

On the **Campaigns** tab, click **Caller ID Numbers.**

1. Click **New** to add a new number. The New Caller ID Number page appears.
2. Enter the Caller ID number **1 800 555 3232**
3. Enter **LDCP_July2016**
4. Enter description **Caller ID for LDCP_July2016**
5. Select **Set as default**
6. Click **Save Caller ID**

Once Harry creates the **Caller ID Number**, he can change it (by using the **Reassign** function), edit the name of description fields, or delete the number.

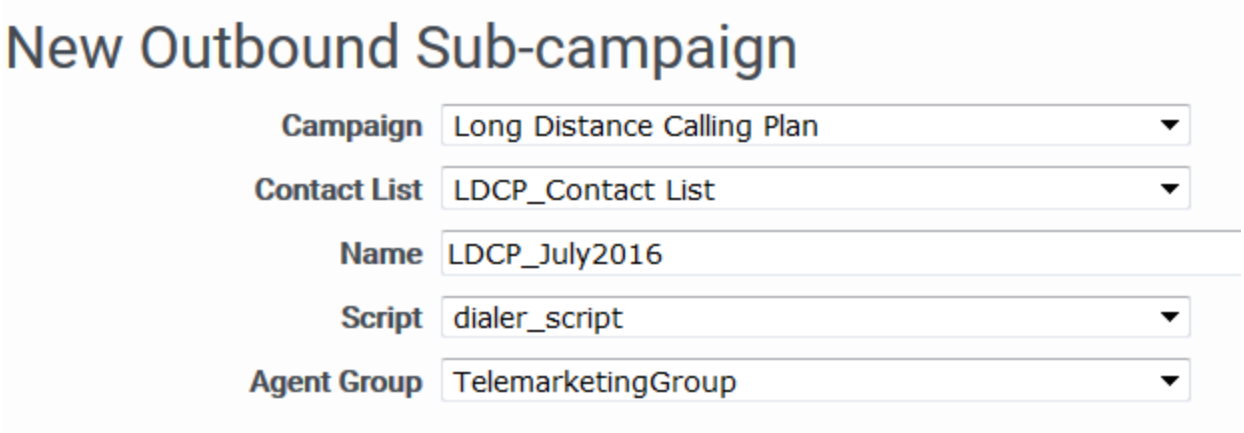
Sub-campaigns

Scenario: Harry is now ready to create and send his sub-campaign. He doesn't have time to go through and specify the same options he specified when he configured the campaign.

Solution: Harry can apply his campaign settings to his sub-campaign.

[+] Instructions

Create sub-campaign from campaign



On the **Campaigns** page, under **Actions**, he'll select **New Outbound** from the Sub-campaign section of the drop-down menu. Then he'll specify the sub-campaign information as follows:

1. Select the campaign he labelled **Long Distance Calling Plan** from the **Campaign** menu.
2. Select the contact list he named **LDCP_ContactList** from the **Contact List** menu.
3. In the **Name** field, enter **LDCP_July2016**. This description will appear on the **Campaigns** list, under the sub-campaign name.
4. In the **Script** field, select **dialer_script.**
5. In **Agent Group**, select **TelemarketingGroup.**

Want to learn more?

- Enterprise Manager help
- Account Manager help
- Compliance (Account Manager)
- Contact Suppression Lists (Account Manager)
- Contacts and contact lists
- Campaigns
- Outbound dialing modes

Stage 3: Dialing and call handling

- Monitoring

Now it's time to start reaching out to customers. First, an automatic dialer will make predictions about agent availability and call results and then begin dialing numbers. Once it detects a voice, it will hand the call off to an available agent.

Dialing

Once Harry's campaign is running in **ASM** mode with **Predictive** dialing, the system can bridge a call to an agent within 2 seconds of detecting a voice.

First, it checks the routing strategy before it dials out. It's looking for the agent group assigned to the campaign and it's checking all routing parameters that specify how it should route the call once it detects a voice. And then, it reserves an agent. The agent commits to that call once all other queued-up calls are cleared.

Then it starts dialing.

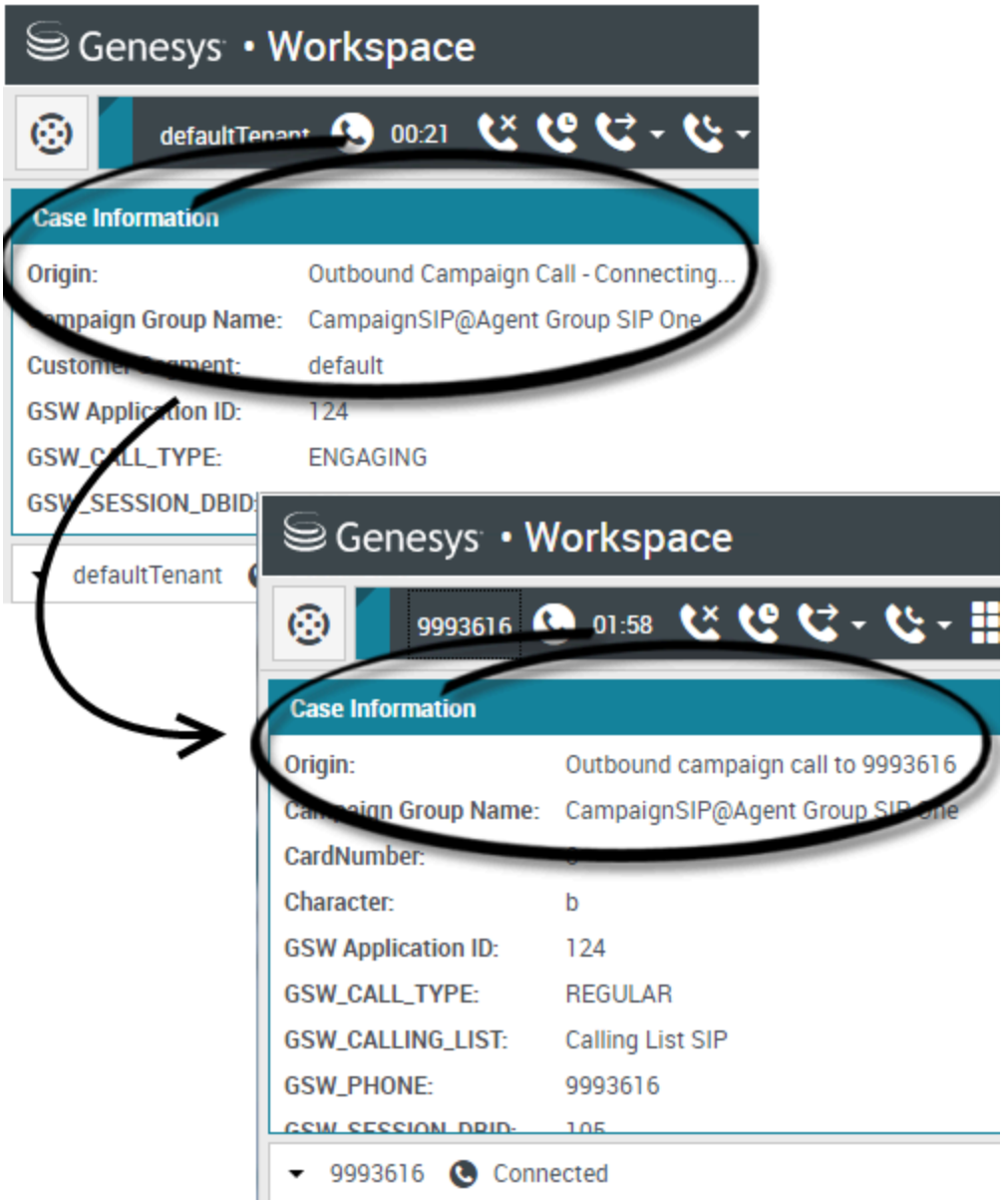
Call handling

Scenario: Agent Martha logs in to **Agent Desktop** just before her shift begins and sees a notification that she has been added to the **Long Distance Calling Plan** campaign.

Solution: Martha can join the campaign and start connecting with customers.

[+] Instructions

Handling an Outbound call



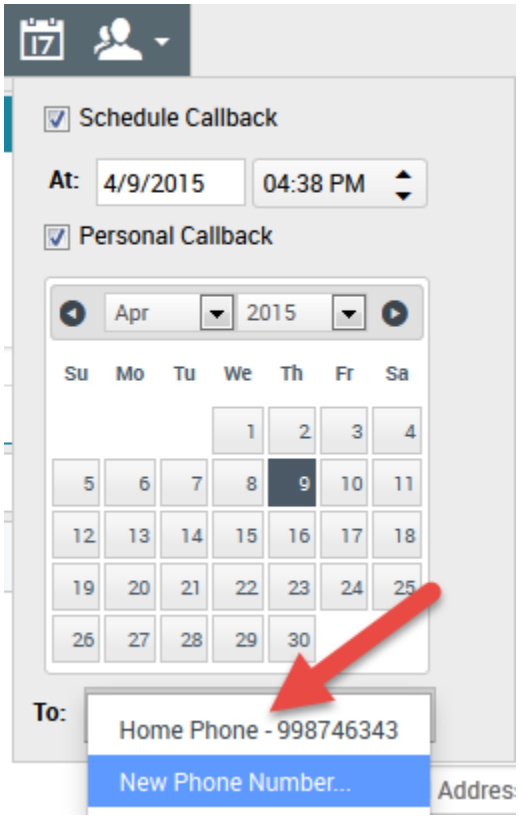
1. In **Agent Desktop**, click **OK** to join the campaign. When the agent is **Ready** and reserved, the phone rings. and the agent waits while the system dials a number. When the system detects a voice, the reserved agent hears a beep and is then connected to the customer.
2. In the meantime, in **Agent Desktop**, the agent sees a pop-up notification, and in the call **Case Information** areas, the **Origin** of the call is **Outbound Campaign Call - Connecting...**
3. As soon as the customer answers the phone and the call is connected, the **Origin** of the call changes to **Outbound campaign call to <name or number of contact>**. The agent might see information about the call in the **Case Information** area – such as campaign call information, the call status, and call actions.

Scenario: Customer Sally is very interested in the new long distance plan that Martha is offering. In the middle of their conversation, Sally's doorbell rings, and she needs to hang up from the call to tend to her guest. She asks Martha to call her back at 7PM to discuss the long distance plan further.

Solution: With Sally on the phone, Martha can use the **Schedule Callback** function.

[+] Instructions

Scheduling a callback



1. Open the calendar and do the following:
    1. Check **Schedule Callback**
    2. Set the date and time for the callback: **July 6, 7PM** in Martha's case.
    3. Check **Personal Callback** because Sally wants Martha, not another agent, to call her back.
    4. In the **To** field, select Sally's Home Phone number from the drop-down.
2. End the call.

Want to learn more?

- Agents and Outbound campaigns
- Agent Desktop help
- Outbound Dialing Modes

Stage 4: Monitoring a campaign

- Monitoring

Harry has set up and run his campaign and sub-campaign. Now it needs to be monitored. This is where Lucy, the manager of Golden Gate Communications, comes in. Lucy is interested in two things: real-time reports of the campaign while it's running and historical reports of the campaign once it ends.

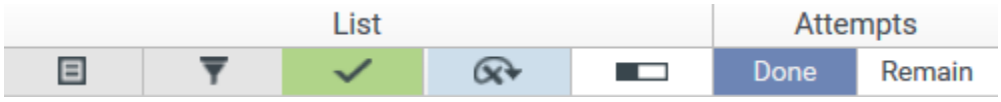
Real-time reports (Engage)

Scenario: Lucy wants to ensure her agents are meeting the campaign goal, which is to reach 120 customers each day of the campaign. To assess if her agents are on track, she wants to keep track of call results each day of the campaign.

Solution: To monitor the status of an ongoing campaign in real-time, Lucy can use the **Engage** dashboard for a snapshot of call results.

[+] Instructions

Engage reports



1. Go to **Account Manager**, and on the **Campaigns** page, the dashboard will show real-time results of the campaign, as follows:

- List columns
  - **Filtered**—The number of contacts filtered from the list.
  - **Delivered**—The number of successful contact attempts.
  - **Retrying or Failed**—The number of contacts being retried or with a current failure status.
  - **Not Attempted**—The number of contact attempts remaining.
- Attempts columns
  - **Done**—The total number of contact attempts made.
  - **Remain**—The number of contact attempts remaining to be made.

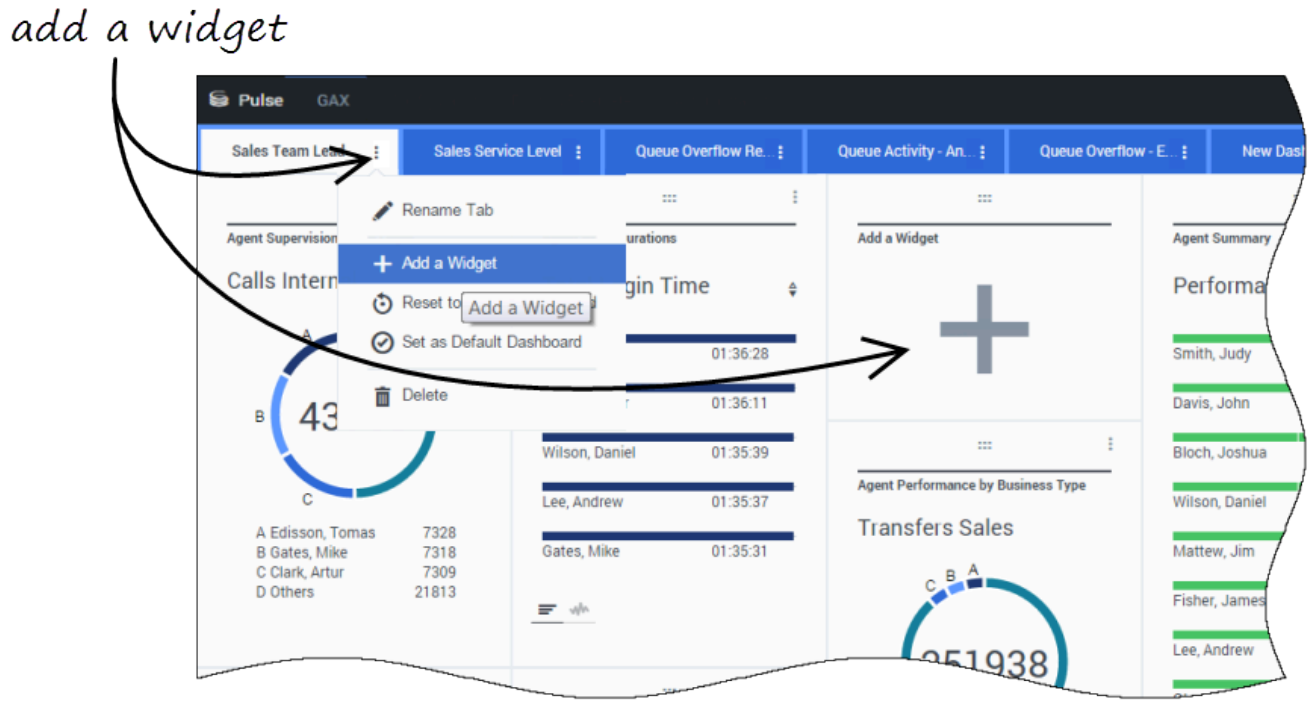
Real-time reports (Genesys Pulse)

Scenario: Lucy has just hired five new agents, including Martha, and wants to track the number of customers they're talking to each day of the campaign.

Solution: Lucy can use **Genesys Pulse** to view agent activity. Lucy will sign in to **Genesys Pulse** and go to her dashboard. She'll add a **report widget** to her dashboard, select the **template and objects or statistics**, and specify **default display settings** – like the name, refresh rate, and type of widget.

[+] Add a report widget

Add a report widget



There are two options for adding a report widget:

1. Click the more icon in the right corner of the dashboard and click **Add a Widget**.
2. Click the **Add a Widget** icon on the dashboard. **Genesys Pulse** opens a widget wizard to help you add a report

[+] Add a template and add objects or statistics

The template wizard guides you through the process of creating, changing, and deleting templates. To open the template wizard:

1. Click **Add Widget**
2. Select **New Template**. The **Choose a Widget** screen displays an alphabetical list of **Genesys Pulse** templates.

To create a **Genesys Pulse** template, you must add or configure:

- One or more object types.
- One or more statistics.
- One widget type with specific options to display the information.

On the **Details** screen:

1. Define the name of the new template
2. Describe the scope of this template
3. Select one or more object types from the selection based on what you might want to monitor.

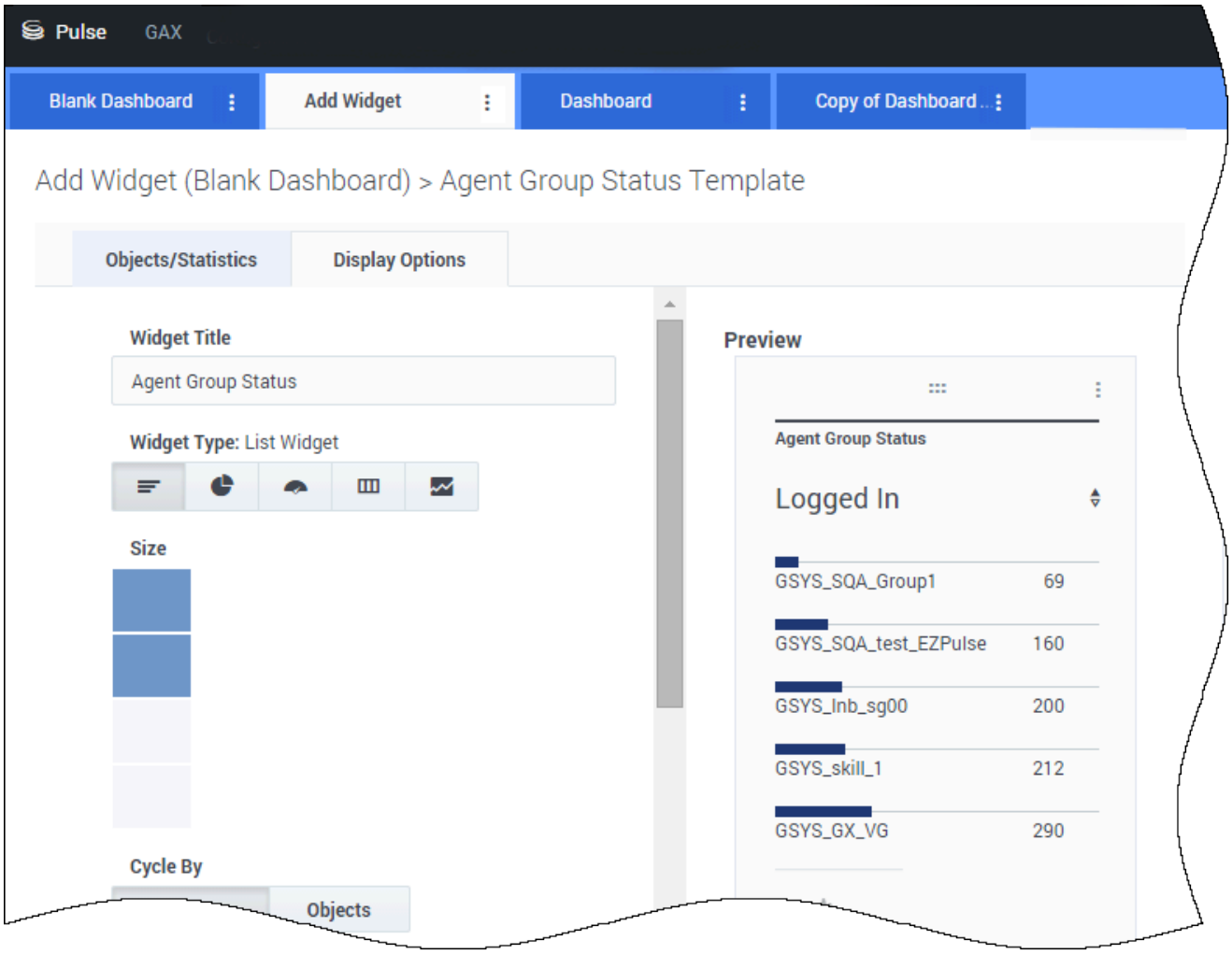
Genesys Pulse allows you to select objects that are compatible with your template. For example:

- The **Agent KPI** template includes the **Agent**, **Agent Group**, **Place**, and **Place Group** objects.
- The **Agent Group Status** template includes the **Agent Group** and **Place Group** objects.

Select the statistics and properties to include in your template. **Genesys Pulse** statistics are described in detail in the templates.xls file. You must add at least one non-string statistic.

[+]Select default display settings

Display options



The final step before validating your template is to define what should be the default display of your widget on the main dashboard. This setting is the one displayed to **Genesys Pulse** users, but they can then modify the widget options on their own dashboard. Name the widget title

1. Select the **Widget Type** to display.
2. Select the Widget refresh rate.
3. Select options associated with the visualization (for example, thresholds and size).
4. If needed, select the statistics for alerts and define the alert values (from 1 to 3).

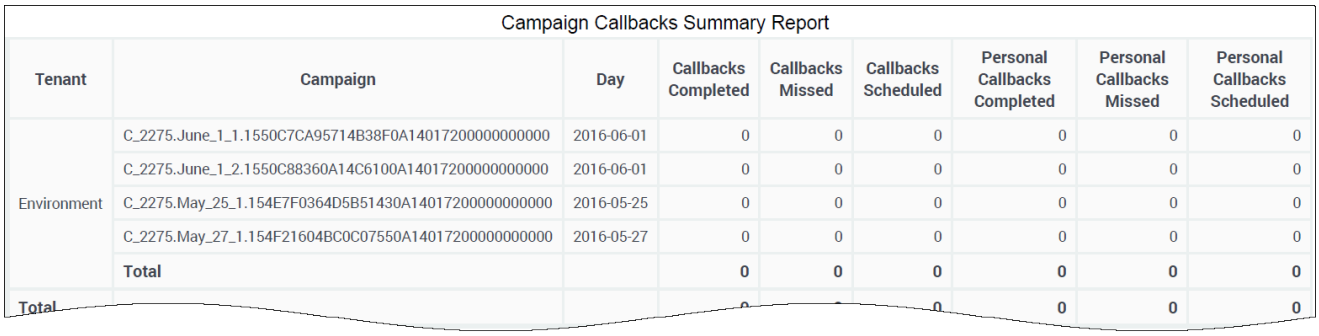
The agents' activity will display in the form of a grid, KPI, line chart, or list. Lucy can save and download the report data as a CSV file.

Historical reports

Scenario: The campaign has ended and G-Tel has asked Lucy for the results so that they can determine if the campaign was worth the investment.

Solution: Lucy can use either **Engage** to run an account, campaign, or sub-campaign summary or detail report, as described here, or she can use **Genesys CX Insights** to run Outbound Engagement reports, as described below.

Campaign Callbacks Summary report

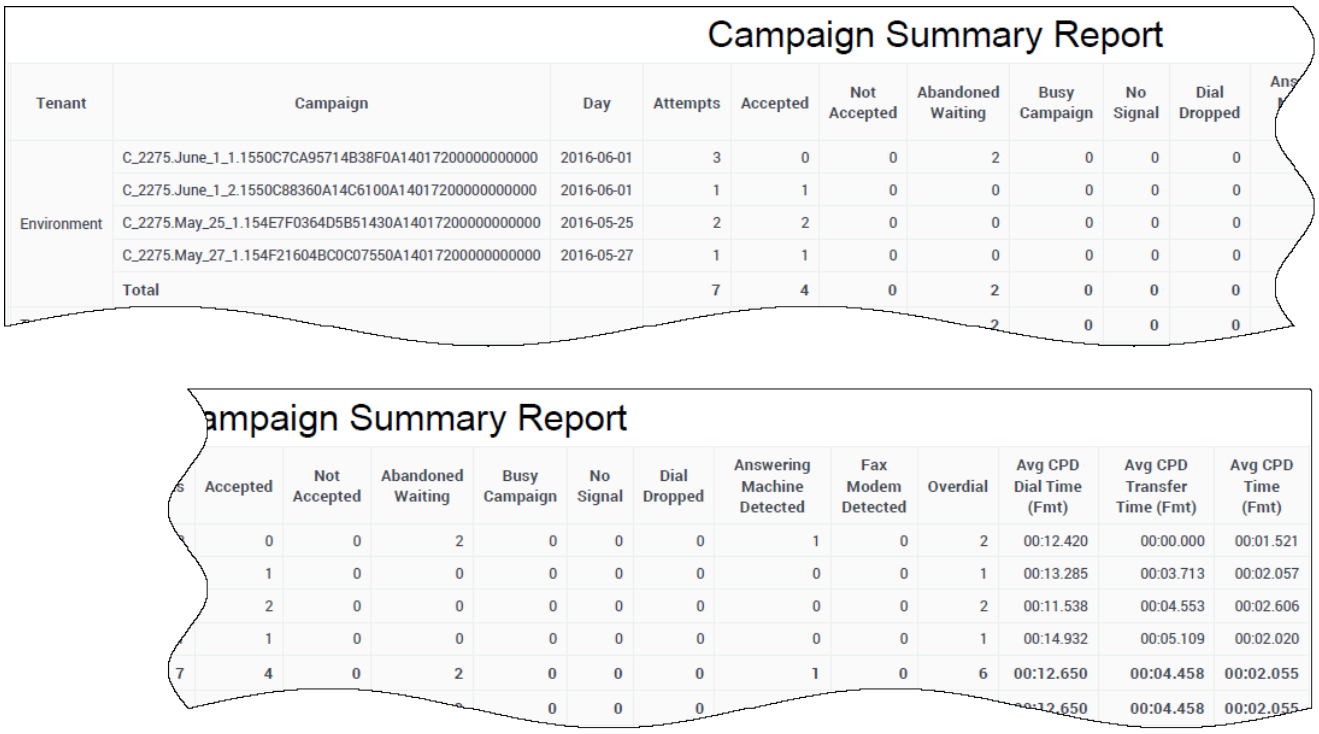


Campaign Callbacks Summary Report

Tenant	Campaign	Day	Callbacks Completed	Callbacks Missed	Callbacks Scheduled	Personal Callbacks Completed	Personal Callbacks Missed	Personal Callbacks Scheduled
Environment	C_2275.June_1_1.1550C7CA95714B38F0A14017200000000000	2016-06-01	0	0	0	0	0	0
	C_2275.June_1_2.1550C88360A14C6100A14017200000000000	2016-06-01	0	0	0	0	0	0
	C_2275.May_25_1.154E7F0364D5B51430A14017200000000000	2016-05-25	0	0	0	0	0	0
	C_2275.May_27_1.154F21604BC0C07550A14017200000000000	2016-05-27	0	0	0	0	0	0
	Total		**0**	**0**	**0**	**0**	**0**	**0**
Total					**0**	**0**	**0**	**0**

The Campaign Callbacks Summary report summarizes the total number of callbacks processed by the contact center, breaking them down into the total number scheduled, missed, and completed for each day of the reporting period and distinguishing personal callbacks from non-personal ones.

Campaign Summary report



Campaign Summary Report

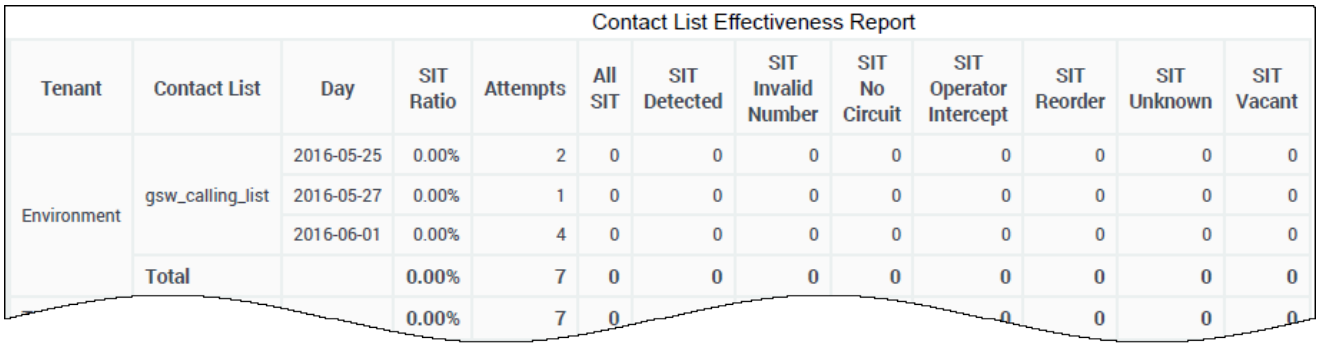
Tenant	Campaign	Day	Attempts	Accepted	Not Accepted	Abandoned Waiting	Busy Campaign	No Signal	Dial Dropped	Ans
Environment	C_2275.June_1_1.1550C7CA95714B38F0A14017200000000000	2016-06-01	3	0	0	2	0	0	0	
	C_2275.June_1_2.1550C88360A14C6100A14017200000000000	2016-06-01	1	1	0	0	0	0	0	
	C_2275.May_25_1.154E7F0364D5B51430A14017200000000000	2016-05-25	2	2	0	0	0	0	0	
	C_2275.May_27_1.154F21604BC0C07550A14017200000000000	2016-05-27	1	1	0	0	0	0	0	
	Total		7	4	0	2	0	0	0	
						2	0	0	0	

ampaign Summary Report

s	Accepted	Not Accepted	Abandoned Waiting	Busy Campaign	No Signal	Dial Dropped	Answering Machine Detected	Fax Modem Detected	Overdial	Avg CPD Dial Time (Fmt)	Avg CPD Transfer Time (Fmt)	Avg CPD Time (Fmt)
	0	0	2	0	0	0	1	0	2	00:12.420	00:00.000	00:01.521
	1	0	0	0	0	0	0	0	1	00:13.285	00:03.713	00:02.057
	2	0	0	0	0	0	0	0	2	00:11.538	00:04.553	00:02.606
	1	0	0	0	0	0	0	0	1	00:14.932	00:05.109	00:02.020
7	4	0	2	0	0	0	1	0	6	00:12.650	00:04.458	00:02.055
				0	0	0				00:12.650	00:04.458	00:02.055

The Campaign Summary report provides several key measures generated by Outbound campaigns that summarize the disposition of contact attempts to reach customers over the reporting interval. The Summary tab, in bar chart format, focuses on two of those key measures: Accepted and Not Accepted.

Contact List Effectiveness report



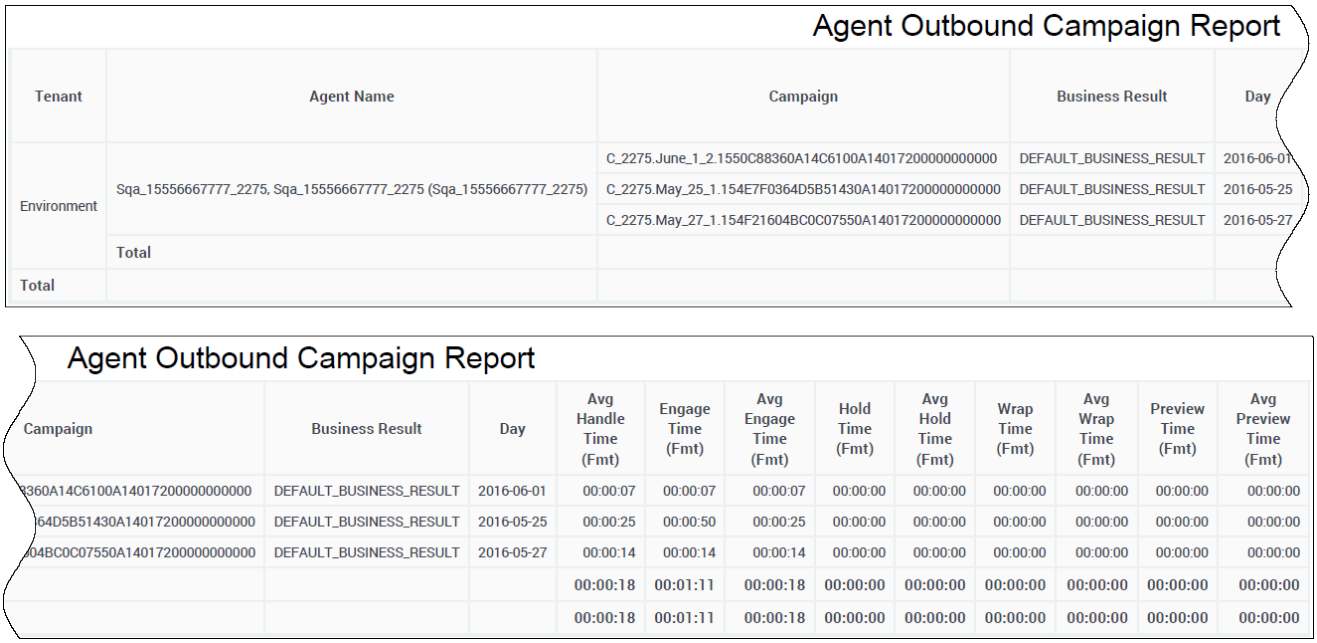
Contact List Effectiveness Report

Tenant	Contact List	Day	SIT Ratio	Attempts	All SIT	SIT Detected	SIT Invalid Number	SIT No Circuit	SIT Operator Intercept	SIT Reorder	SIT Unknown	SIT Vacant
Environment	gsw_calling_list	2016-05-25	0.00%	2	0	0	0	0	0	0	0	0
		2016-05-27	0.00%	1	0	0	0	0	0	0	0	0
		2016-06-01	0.00%	4	0	0	0	0	0	0	0	0
	Total		0.00%	7	0	0	0	0	0	0	0	0
			0.00%	7	0				0	0	0	0

The Contact List Effectiveness report provides the number of contact attempts that were generated by an Outbound campaign, the ratio of attempts that resulted in the detection of a special information tone (SIT), and a breakdown of the call results of those SIT-detected attempts for the

selected calling list. The Summary tab of this report plots, in bar chart format, the SIT ratio for each calling list.

Agent Outbound Campaign report



Agent Outbound Campaign Report

Tenant	Agent Name	Campaign	Business Result	Day
Environment	Sqa_15556667777_2275, Sqa_15556667777_2275 (Sqa_15556667777_2275)	C_2275.June_1_2.1550C88360A14C6100A14017200000000000	DEFAULT_BUSINESS_RESULT	2016-06-01
		C_2275.May_25_1.154E7F0364D5B51430A14017200000000000	DEFAULT_BUSINESS_RESULT	2016-05-25
		C_2275.May_27_1.154F21604BC0C07550A14017200000000000	DEFAULT_BUSINESS_RESULT	2016-05-27
	Total			
Total				

Agent Outbound Campaign Report

Campaign	Business Result	Day	Avg Handle Time (Fmt)	Engage Time (Fmt)	Avg Engage Time (Fmt)	Hold Time (Fmt)	Avg Hold Time (Fmt)	Wrap Time (Fmt)	Avg Wrap Time (Fmt)	Preview Time (Fmt)	Avg Preview Time (Fmt)
2360A14C6100A14017200000000000	DEFAULT_BUSINESS_RESULT	2016-06-01	00:00:07	00:00:07	00:00:07	00:00:00	00:00:00	00:00:00	00:00:00	00:00:00	00:00:00
64D5B51430A14017200000000000	DEFAULT_BUSINESS_RESULT	2016-05-25	00:00:25	00:00:50	00:00:25	00:00:00	00:00:00	00:00:00	00:00:00	00:00:00	00:00:00
04BC0C07550A14017200000000000	DEFAULT_BUSINESS_RESULT	2016-05-27	00:00:14	00:00:14	00:00:14	00:00:00	00:00:00	00:00:00	00:00:00	00:00:00	00:00:00
			00:00:18	00:01:11	00:00:18	00:00:00	00:00:00	00:00:00	00:00:00	00:00:00	00:00:00
			00:00:18	00:01:11	00:00:18	00:00:00	00:00:00	00:00:00	00:00:00	00:00:00	00:00:00

The Agent Outbound Campaign report provides total and average duration of call handling activities for agents who participate in Outbound campaigns.

Want to learn more?

- Genesys CX Insights Outbound Contact reports and Agent Outbound Campaign Report
- Get Started with Genesys CX Insights Historical Reporting
- Get Started with Pulse Real-time Reporting

Outbound Dialing Modes

When you configure your Outbound campaign, you'll need to choose a dialing mode that best suits your campaign. The dialing mode you choose will depend on the type of campaign, the number of agents assigned to the campaign, and legal requirements. Read the article below or watch the videos to learn about the dialing modes and Outbound IVR modes available in Genesys Engage cloud.

Related Topics

The first video provides an overview of the dialing modes available in Genesys Engage cloud, while the second video walks you through four unique business scenarios and identifies the best dialing mode in each scenario and campaign.

Link to video Link to video

Dialing Modes

The table gives a brief overview of each dialing mode. You can find more details, including instructions for enabling, in the sections below the table.

Dialing Mode	Description
Predictive (Automatic)	• The fastest dialing mode. • The dialer uses an algorithm to predict agent availability and call results so that it can safely dial multiple phone numbers for each available agent. • The constant rate of dialing results in reduced agent idle time. • Can run in either ASM (fast bridging) mode or Transfer mode. • To minimize the risk of exceeding the target abandoned rate, use only in high-volume campaigns with at least 15 agents.
Progressive (Automatic)	• The dialer places only one call for every one available agent. • Lower risk of exceeding the target abandoned rate. • Can run in either ASM (fast bridging) mode or Transfer mode.

Dialing Mode	Description
	• Use in low-volume, high-value campaigns with fewer than 15 agents in the group.
Preview (Manual)	• An agent requests (pulls) records from the system, previews the record, and then decides whether to dial the call. • Use in high-value campaigns, where individual ownership of accounts is the highest priority.

Predictive

Predictive is an automatic dialing mode, which means the dialer automatically dials phone numbers and then bridges the answered calls to agents. This is the fastest among all the dialing modes because the automatic dialer uses an algorithm to predict agent availability and call results so that it can safely dial multiple phone numbers for each available agent.

Because of this constant rate of dialing and connecting, agent idle time is reduced, and the dialer can run through the contact list much quicker than it could in any other mode.

This constant rate of dialing can also increase the risk of exceeding the target abandoned rate - defined as the percentage of dropped, or abandoned, calls resulting from the system making more calls than there are available agents - so you should use Predictive only in campaigns with at least 15 agents to ensure there are always enough agents available to accept the bridged calls. If you have fewer than 15 agents, you should consider Progressive.

Progressive

Progressive is also an automatic dialing mode, meaning the dialer automatically dials phone numbers and bridges the answered calls to agents. In Progressive mode, the dialer places only one call for every one available agent. For example, if six agents are available, the system dials six phone numbers.

This guarantees that there’s an available agent for every contact who picks up the phone, so there’s no risk of exceeding the 3% allowable call abandoned rate set out by federal regulations.

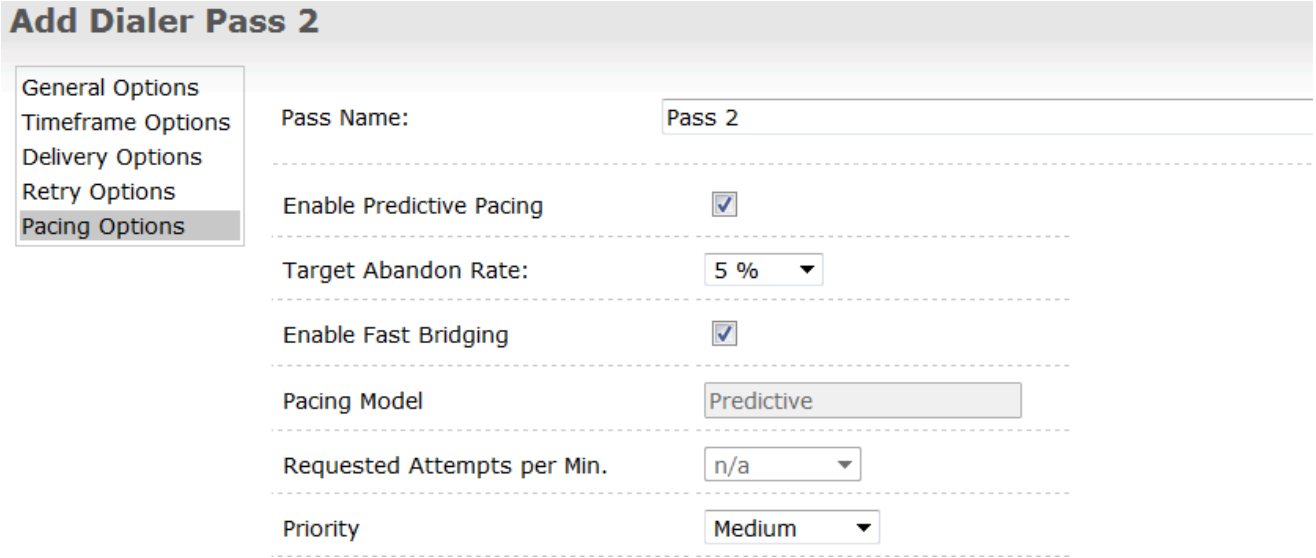
Use Progressive if you have fewer than 15 agents assigned to a campaign.

Switching Modes: ASM and Transfer

When a campaign using an automatic dialer runs in Active Switching Matrix (ASM) mode, the system does two things before it dials out: it looks at the routing parameters so that it knows how to route

the call when it detects a voice on the line, and it reserves an agent up front. By doing this work up front, it can safely bridge the call to an agent within two seconds after detecting a voice on the line, as required by federal regulations.

When a campaign runs in Transfer mode, the system waits until it has a contact on the line before it checks the routing parameters and looks for an agent. Because this process will always take longer than two seconds, it's not recommended in sales or telemarketing campaigns that are bound by federal regulations.

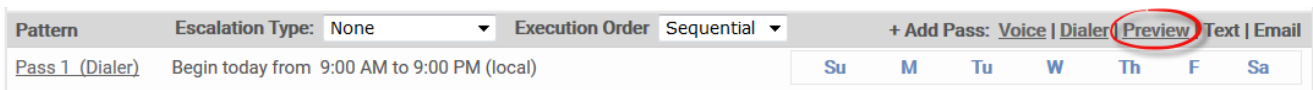


To enable Progressive or Predictive dialing:

1. On the **Outbound** tab, go to **Pattern Options** and select **Dialer** Pass
2. Select **Pacing Options**
3. Check the box next to **Enable Predictive Pacing** to enable Predictive dialing or leave it unchecked to enable Progressive dialing
4. Check the box next to **Enable Fast Bridging** to run the campaign in ASM mode. If you don't check the box next to **Enable Fast Bridging**, the campaign will run in Transfer mode.

Preview

Preview dialing is a manual dialing mode, where an agent requests one or several records from the system, previews each record, and dials the contacts manually.



To enable Preview dialing:

1. Go to the **Outbound** tab
2. In **Pattern Options** select **Preview** pass

> **Important**
> If you select the **Preview** pass, the system will not honor any pacing options.

Outbound IVR

Outbound IVR dialing enables you to play personalized and interactive automated messages to your contacts. All processing of these outbound calls is handled by Designer.

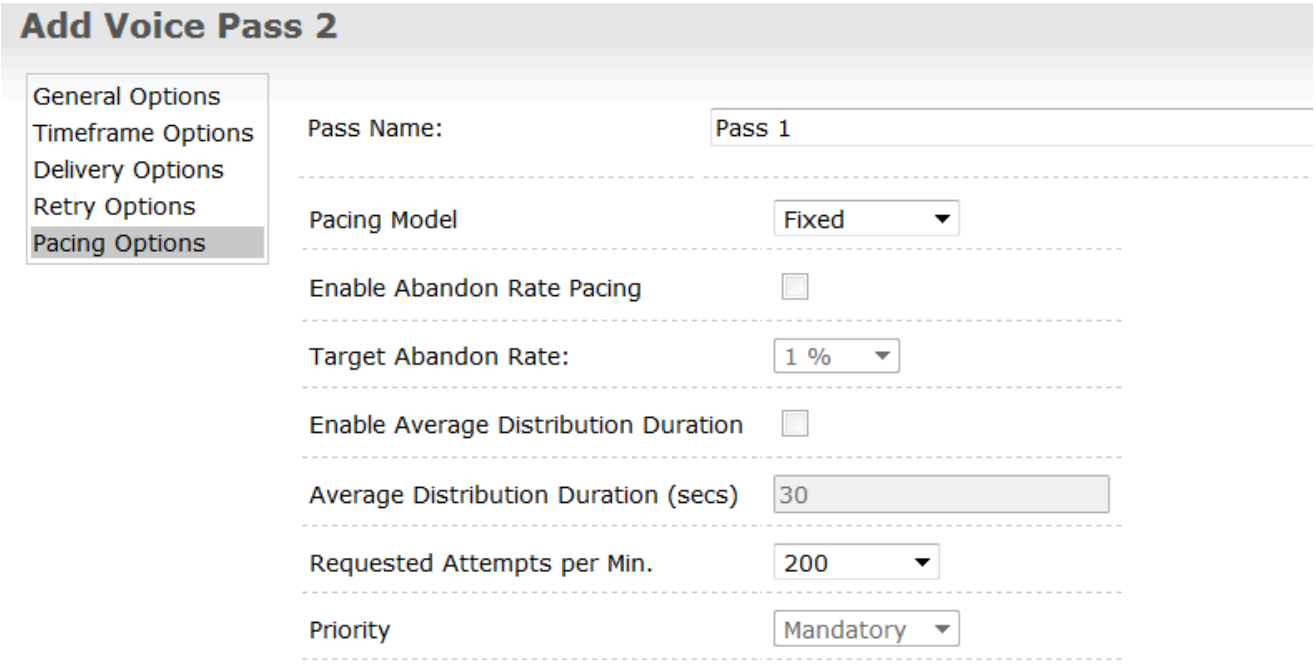
The table gives a brief overview of each Outbound IVR mode. You can find more details, including instructions for enabling, in the sections below the table.

IVR Mode	Description
Power GVP	• Messages can be personalized and interactive (*Press 1 to take a survey*, for example). • Agents are not involved in these campaigns.
Predictive GVP	• Calls can be connected to an agent. (*Press 0 to speak to a customer service representative*, for example). • Can run in either ASM mode or Transfer mode • Pacing is based on either the Target Abandon Rate or the Average Distribution Duration.

Power GVP

With **Power GVP**, messages can be personalized for each contact and they can be interactive (For example, Press 1 to take the survey). Agents are not involved in these campaigns, so contacts are never given the option to speak to one. When the message ends or when the contact completes an interaction, the call ends.

Power GVP uses a Fixed pacing model, which enables you to specify a number of contact attempts per minute.



To enable Power GVP:

1. On the **Outbound** tab, go to **Pattern Options** and select **Voice** Pass
2. In **Pacing Options**, specify the following:

- Pass Name
- Pacing Model: Fixed
- Requested Attempts per Min: Specify a number
- Priority: Mandatory

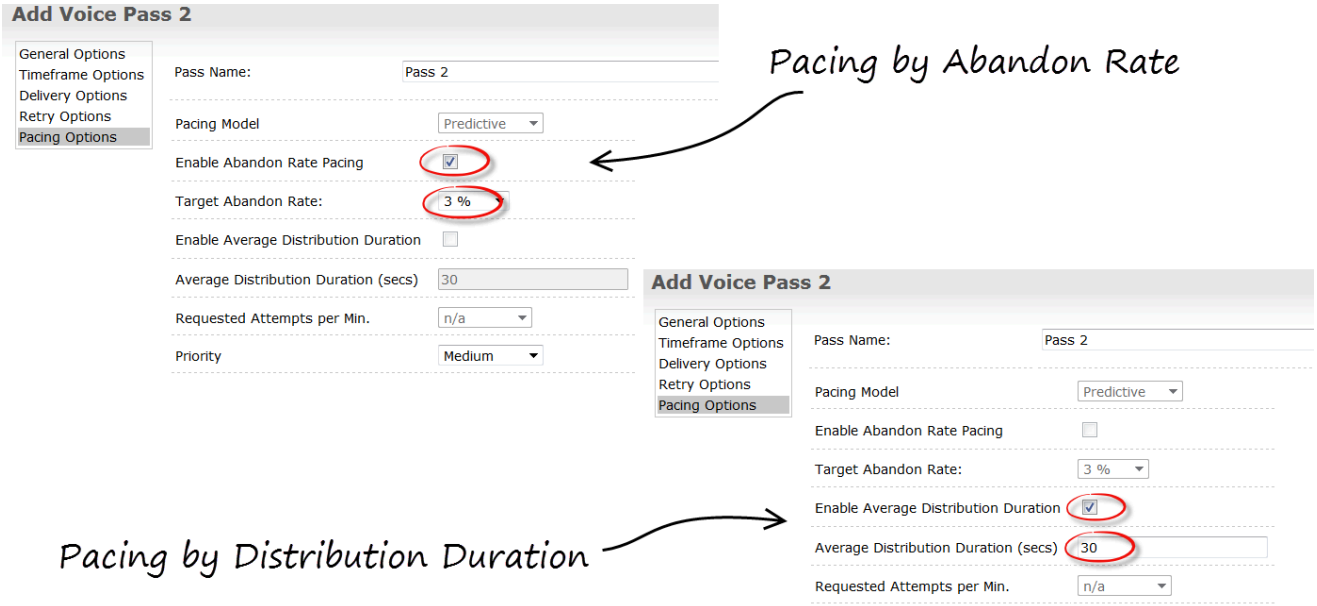
Predictive GVP

Predictive GVP is similar to Power GVP, but with this mode, a contact is given the option to speak to an agent (For example, *Press 1 to speak to an agent*).

Predictive GVP can run in either ASM mode or Transfer mode. and you can select pacing options based on either the Target Abandon Rate or the Average Distribution Duration.

Use Account Manager to enable the dialing mode and use Designer to set the routing parameters.

> **Important**
>
> For both Outbound IVR dialing modes, you must create an Agent Group in Platform Administrator and select that Agent Group when creating the sub-campaign in Account Manager.



To enable Predictive GVP based on Target Abandon Rate, specify the following in **Pacing Options**:

- Pass Name:
- Pacing Model: Predictive
- Check Abandon Rate Pacing
- Target Abandon Rate: 3%
- Priority: Medium

To enable Predictive GVP based on Average Distribution Duration, specify the following in **Pacing Options**:

- Pass Name:
- Pacing Model: Predictive
- Click Average Distribution Duration
- Average Distribution Duration (secs): 30 (for example)
- Priority: Medium

Multi-channel Campaigns

Leverage both automated and agent-assisted outbound channels to proactively reach customers. With Genesys Engage cloud, your agents can make calls – manually or via an automatic dialer – or you can send automated messages via text, email, or outbound IVR.

You can initiate outreach in one channel and then expand to another channel if no response is received, and you can blend Interactions by leveraging more than one channel as part of a single customer dialog, including strategies such as scheduling a service call by phone, then sending a text-based reminder one day before the scheduled appointment.

Here's how you can set up a multi-channel campaign using Outbound for Genesys Engage cloud.

Channels

First, let's take a look at the available channels.

Channel	Description	Example
Voice	Outbound IVR and automated alert messaging (Power GVP and Predictive GVP).	• Power GVP: Send customer satisfaction surveys that ask customers to answer questions using their keypad. • Predictive GVP: Send automated payment reminders. Give customers the option to speak to an agent when the message finishes playing.
Dialer	Automatic dialer places calls (Progressive and Predictive).	• Run a telemarketing campaign to sell more data plans to existing customers.
Preview	Agents review customer records and then manually launch the call.	• High-risk collection campaigns that require careful review of customer records before making contact.
Text	One-way and interactive SMS text messaging.	• One-way: Send order confirmations. • Two-way: Send appointment

Channel	Description	Example
		reminders and ask customers to respond with Y or N.
Email	Personalized and dynamic email messaging.	• Send order confirmations. Customers can click the embedded link to view or cancel their order.

Voice

Use the Voice channel to run an Outbound IVR campaign that sends automated messages and alerts to your contacts. You can run an Outbound IVR campaign in Power GVP mode, with no agents assigned to the campaign, or in Predictive GVP mode, where contacts are given the option to speak to an agent.

Dialer

Use the Dialer channel to run a campaign using a dialer that automatically dials and bridges calls to agents.

This channel enables you to run a campaign using Predictive dialing, which paces calls based on anticipated agent availability, or Progressive dialing, which places a call only when an agent is available. In addition, you can run a campaign in ASM mode, also called fast bridging, to optimize bridging time.

Learn more about the dialing modes here

Preview

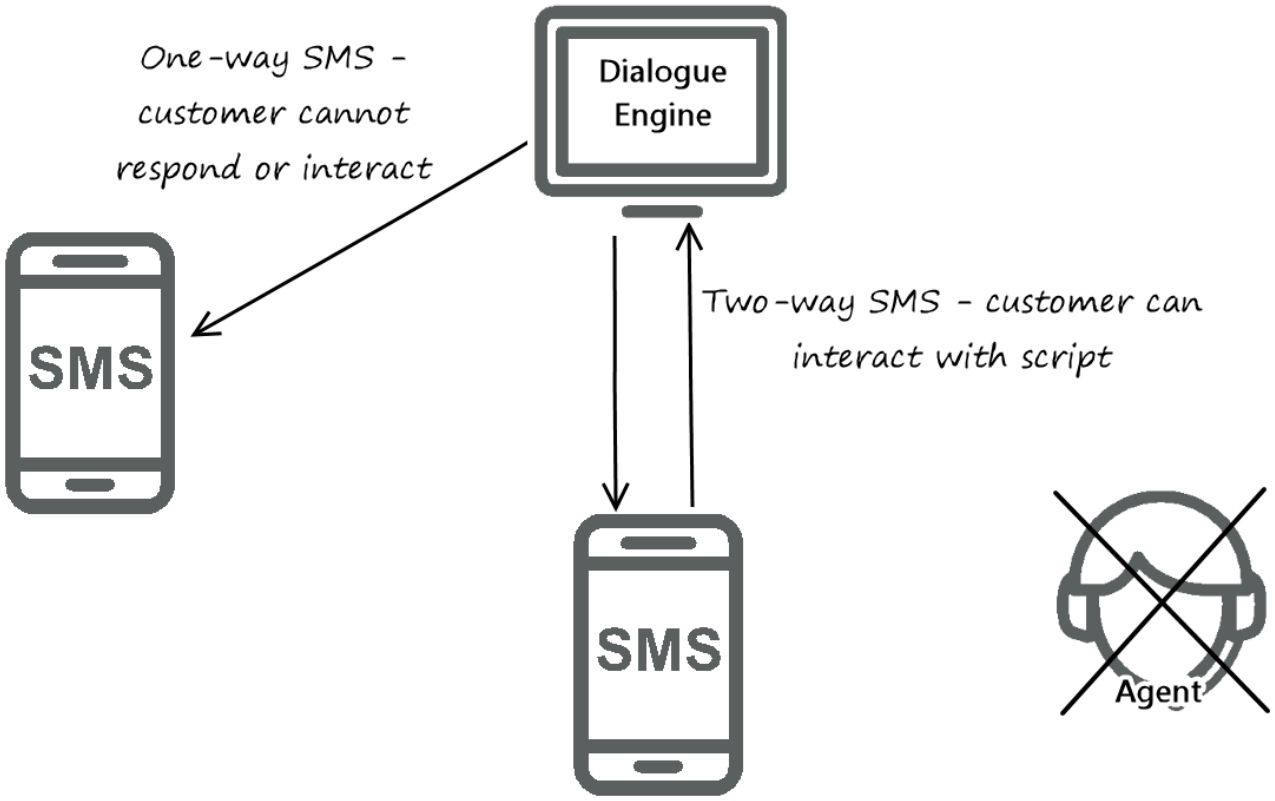
Use the Preview channel if you don’t want to use an automatic dialer. Instead, agents preview each record and then manually launch the call.

Text

Use the Text channel to send one-way or two-way text messages to your contacts. With one-way messaging, you simply broadcast a message to every person on your contact list. In this scenario, contacts don’t respond to the scripted message. For example, a wireless provider could send payment reminders to customers who are more than 30 days behind on their bill. Customers are not asked to respond to the text.

If you’re running a campaign that requires a customer's response – for example a questionnaire or appointment confirmation – use two-way SMS. For example, a clothing retailer could invite customers to an exclusive in-store sales event and ask them to respond to the text with Y or N.

With this option, the script engages a dialog engine that looks for predefined keywords or short codes in the customer's response. Interactions continue to take place between the script and the customer only; an agent is never involved.



> **Important**
> MMS is not currently supported.

Email

Use the Email channel to send personalized and customized messages, including embedded links. With email, one-way broadcast is the only option. While custom REPLY TO addresses for direct responses are supported, auto-response (dialog engine) functionality is not supported.

For example, a hotel chain can email customers to announce an upcoming sale on its website. A customer can reply to the email but a dialog engine will not be engaged, and therefore interactions will not take place. The reply email will instead go to a specified mailbox.

Scripts for Text and Email

Unlike the Voice channel, which uses Designer for script and outbound IVR functionality, the Text and Email channels use Engage scripts.

A script is a file that combines the text, images and business rules, or set of instructions, sent out to a customer via a text message or Email.

Scripts can be very simple - involving only one-way communication with no interactive capabilities - or in the case of SMS, they can involve a dialog engine that enables interaction between the script and the customer.

Your account representative creates the script for you and can also edit the script and basic script attributes, such as response groups, message variables, and Email image resources.

Configure multi-channel campaigns

You can run a multi-channel, or multi-pass, campaign if you want to reach out to your customer using more than one channel or device. For example, in the first pass, you could use the Voice channel to send an automated payment reminder to your customers, via an outbound IVR. In the second pass, you could use the Text channel to send a personalized reminder message to the customers you didn’t reach in the first pass. In the third pass, you could use the Preview channel to have agents manually call the customers you didn’t reach in the first two passes.

To configure a multi-channel pass:

1. Specify a script
2. Select the call pattern options
3. Define each voice pass

Specify a script

If you plan to send a text or email, you need to first specify the script that contains the message content. This script was created or uploaded on the Scripts tab.

To specify a script:

1. On the Edit Campaign page, click the Outbound tab.
2. In Outbound Options,select the script from the Script menu.

If your campaign won't include an Text or Email channel, you do not need to specify a valid script. Voice scripts are created in Designer.

Select the Pattern options

Define the call pattern by selecting the escalation type and execution order In the Pattern section of

the Outbound tab.

Escalation Types are as follows:

- **None** - Select this option when escalation is not required.
- **Contact Based** - This option enables you to attempt multiple contact methods on each pass. Contact based escalation requires an input specification file that lists the order of escalation (from the Order field) and the conditions defining the escalation (from the Conditions field), and then a strategy is set to use those conditions. Whatever is listed in the input specification file as a condition must also be selected in the Retry Options for the pass.

Execution Order options are as follows:

- **Sequential** - Only one pass runs at a time. The pass with the earlier start time will run first. If passes have the same start time, the pass listed first in the Pattern section of the Outbound tab will run first.
- **Parallel** - Multiple passes are executed at the same time. For example, assume Pass1 is scheduled to dial from 9:00 AM to 11:00 AM (local time zone) and Pass2 is scheduled to dial from 10:30 AM to 12:00 PM (also local time zone). Any contact attempts made during Pass1 can be moved immediately to Pass2 and a second contact attempt made when Pass2 begins at 10:30 AM.

Define the passes

Now you can set up a multi-channel campaign by adding one pass at a time. A pass is basically a contact attempt that applies pre-defined treatments.

To add a pass, go to the Outbound tab of your campaign, and in the Pattern section, select the following:

- **+ Add Pass**- Choose a channel: Voice, Dialer, Preview, Text, or Email

Now apply the rules and conditions for each pass.

- Voice Pass Options
- Dialer Pass Options
- Preview Pass Options
- Text and Email Pass Options

Configure the standard campaign filters, business rules, contact rules, and suppression rules as you normally would for a single-channel campaign.

Edit or delete a pass

Once you've defined a pass, a summary of the pass will appear in the Pattern section of the Outbound tab. To edit the pass, click the pass name to open the list of pass options. When you've finished editing, click **Save Pass**. The changes take effect immediately.

To delete a pass, click the corresponding X that appears next to the pass name and summary.

Need more information?

- About Campaigns and Campaign Configuration
- Outbound Dialing Modes Explained
- Scripts

Enterprise Manager

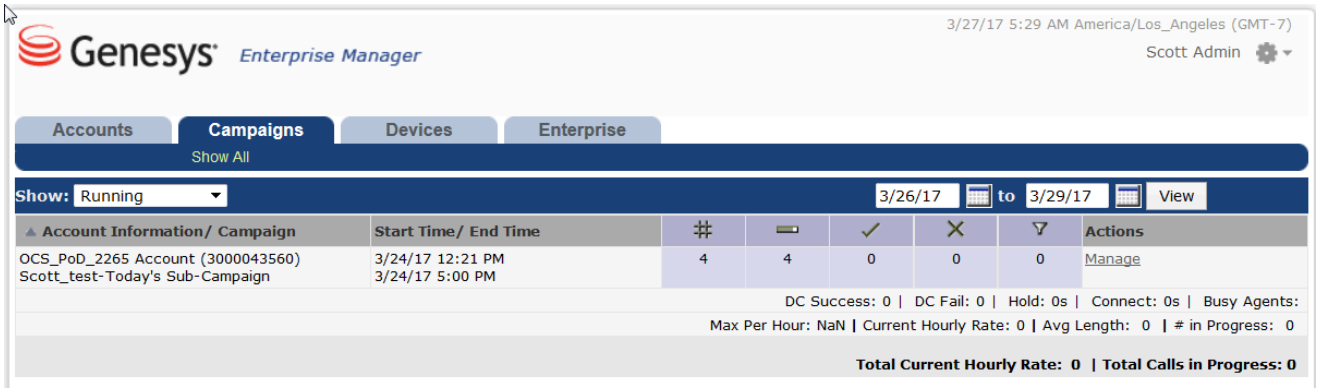
Use Enterprise Manager to create one or more individual accounts in which messaging campaigns are run. With the enterprise account, you can apply global settings across all individual accounts at once.

You create individual accounts in the enterprise for each user in the enterprise. For example, a typical enterprise contains one or more individual accounts with Corporate plan privileges. If needed, the enterprise may also contain individual accounts with defined roles, such as Account Manager, Reporting or Call Center roles, in order to perform specific tasks.

What can I do in Enterprise Manager?

Before you get started, get to know the applications and what's involved in setting up and managing a campaign:

- Outbound Solutions Guide
- Outbound Business Scenario
- Set Up Outbound Routing
- Load a Sub-campaign and Make a Call



Now you can go directly to Enterprise Manager Help to learn about these topics:

- Campaigns and Sub-campaigns
- Enterprise Accounts
- Devices
- Plans
- Users

- Enterprise Settings

Still looking for answers?

Check out these individual topics:

- Search for users and accounts in the enterprise.
- Sign in to any individual account in the enterprise, including the Enterprise Account.
- Create and edit campaign strategies, which you can make available to all individual accounts.
- Import input and output specification files.
- Import an enterprise-wide contact suppression list.
- Set enterprise-wide compliance rules.
- Manage sub-campaigns for all individual accounts in the enterprise.
- Track actions taken against one or more individual accounts in the enterprise.
- Define enterprise-wide security settings.

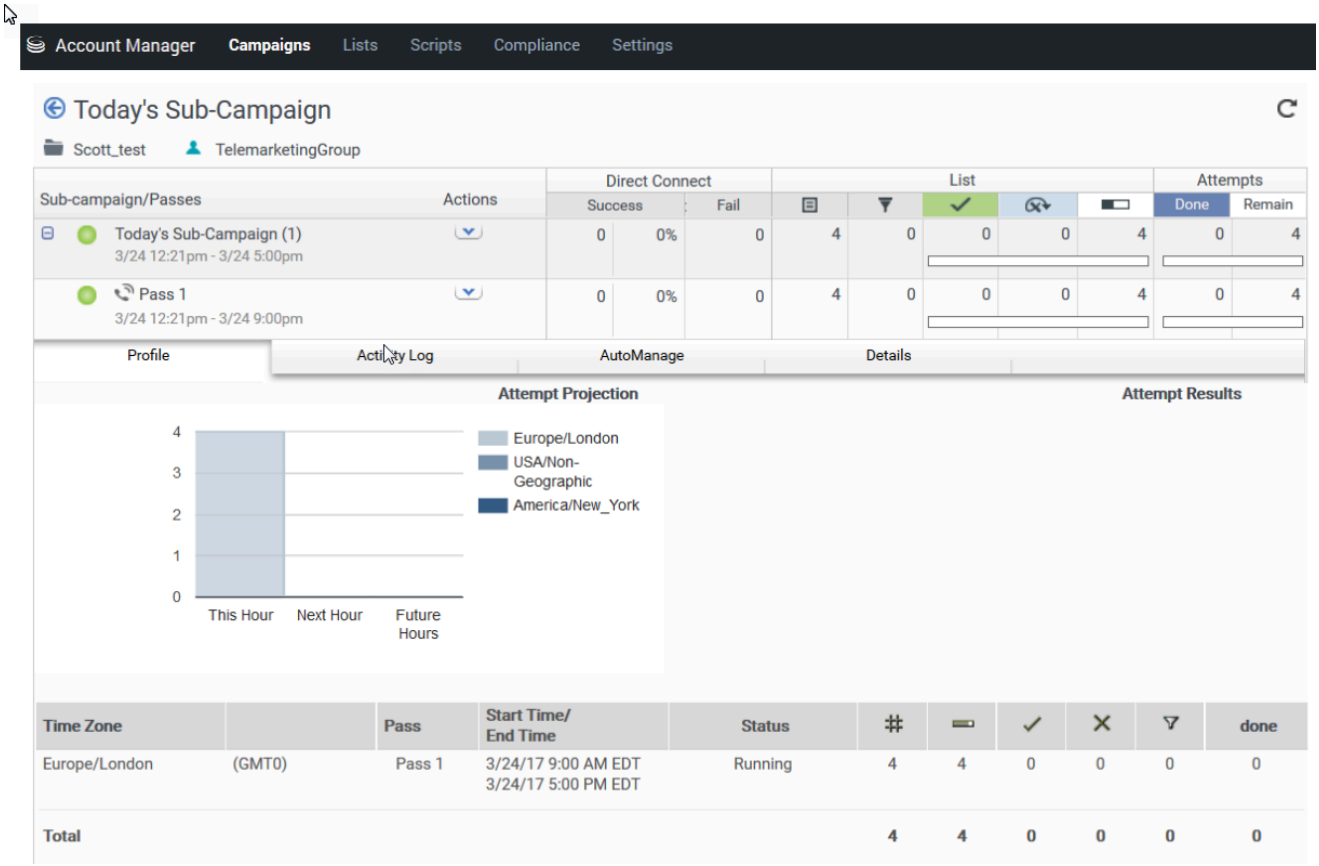
Account Manager

The Account Manager provides access to an individual account, and its associated campaigns, within the enterprise.

What can I do in Account Manager?

In the Account Manager user interface, you can configure and manage your campaigns and sub-campaigns. Here are a few guides to get you started:

- Outbound Solutions Guide
- Outbound Business Scenario
- Set Up Outbound Routing
- Load a Sub-campaign and Make a Call



Now you can go directly to Account Manager Help to learn about these topics:

- Campaigns and Sub-campaigns
- Contact Lists
- Compliance Tools and Suppression Lists
- Scripts
- Settings

Still looking for answers?

Check out these individual topics:

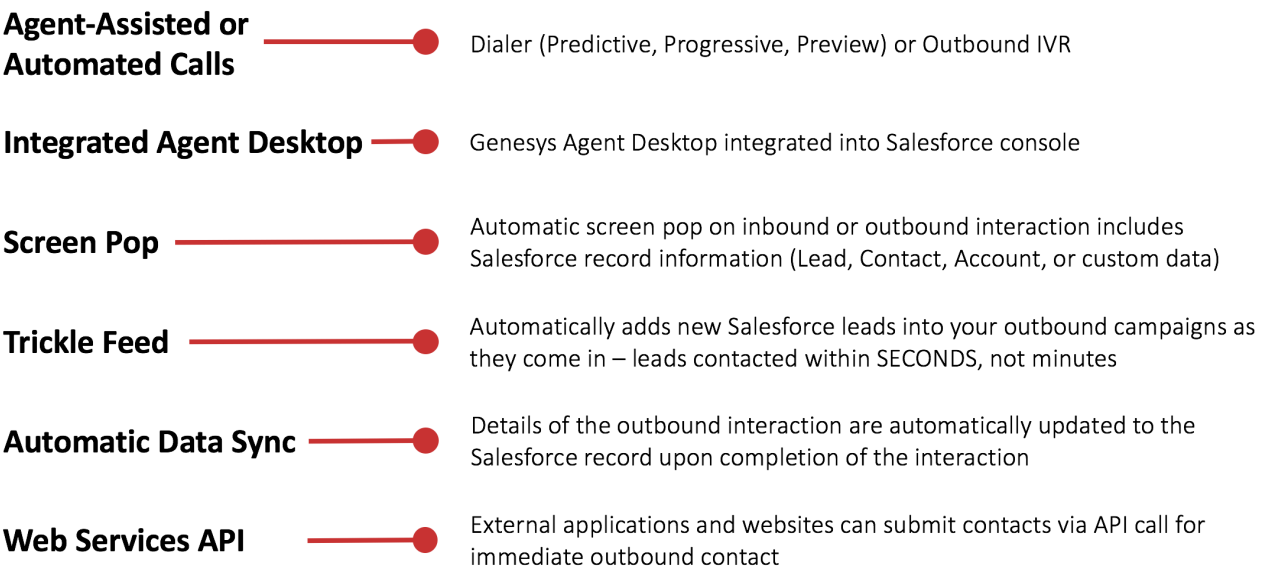
- Run a multi-channel campaign to include both agent-assisted and automated communications.
- Create a campaign strategy to define default campaign settings.
- Configure a dialing mode.
- Create and upload contact lists, using the Account Manager user interface or the Salesforce Adapter for Outbound.

- Suppress contacts using compliance tools and contact suppression lists.
- Create a script for text and email campaigns.
- Define AutoManage Rules and monitor active issues and alerts.
- View a log of all system events for the account or enterprise.

Salesforce Adapter for Outbound

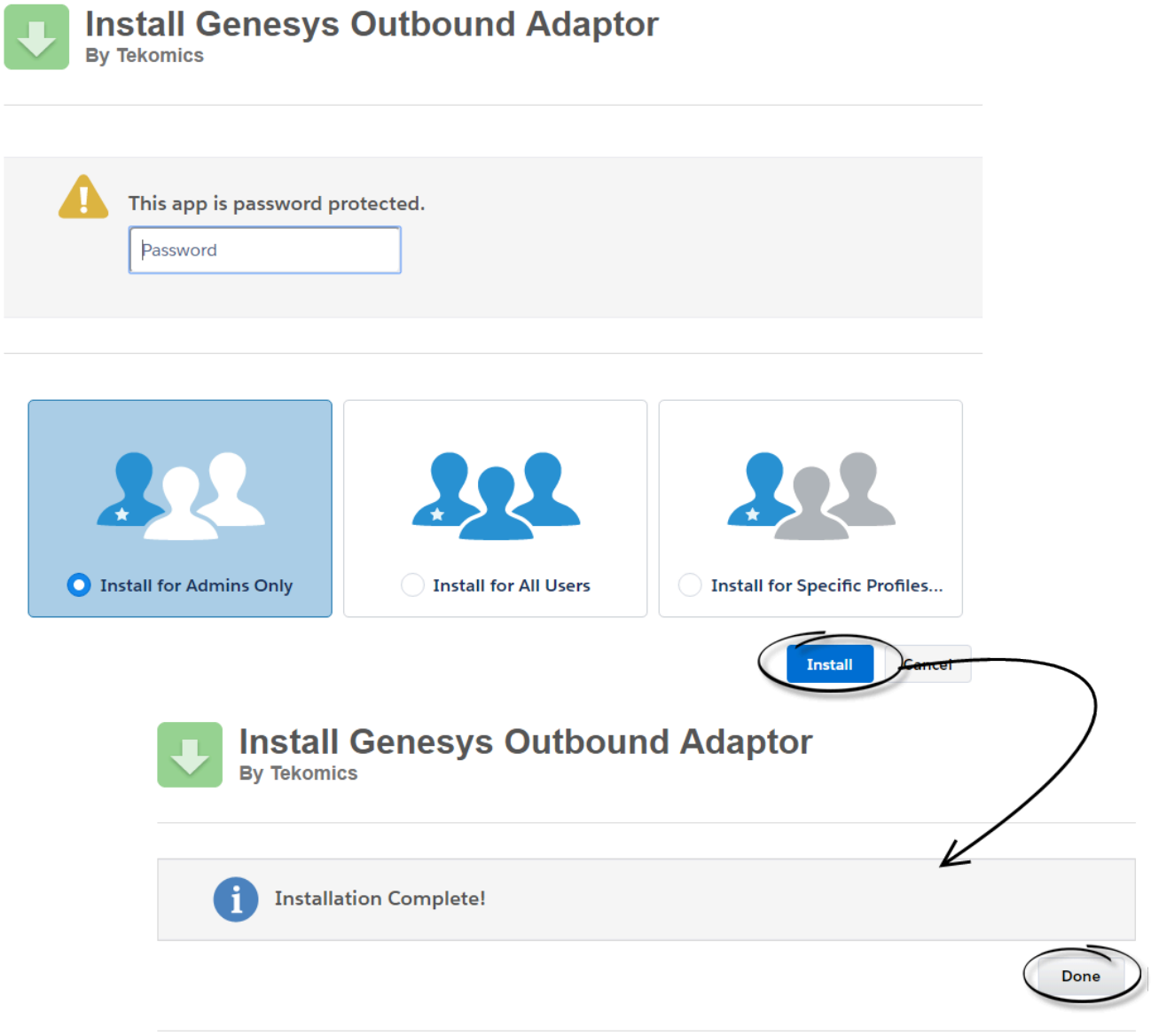
Welcome to the *Salesforce Adapter for Outbound*. This document provides information about installing and configuring the Engage SFDC Adapter for Outbound on salesforce.com, as well as some quick how-to information on exporting your campaigns.

The Engage SFDC Adapter for Outbound is a plugin that allows customers to export contacts to the Engage platform, easily. Here are some of its key features:



Installing and Uninstalling

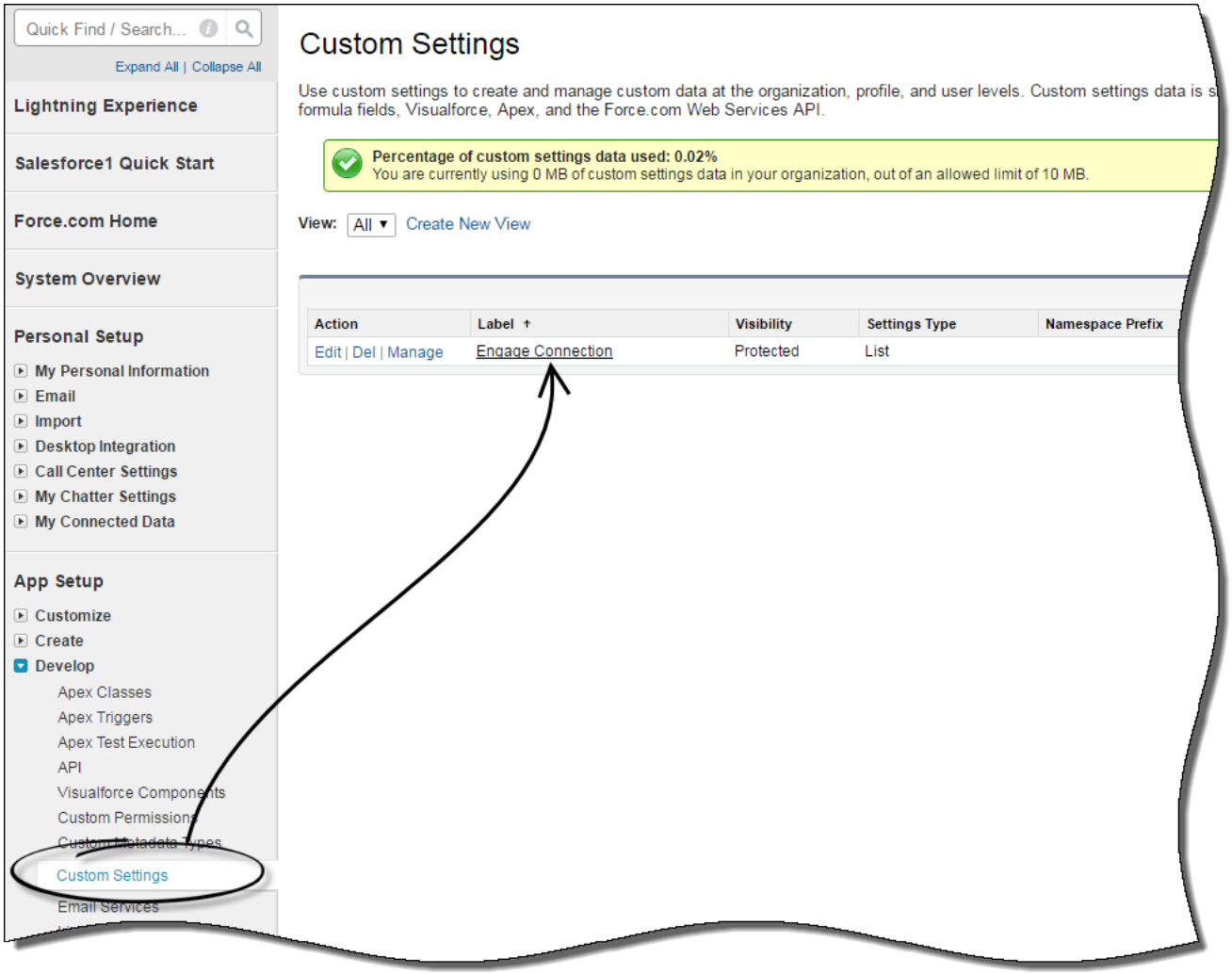
How to install the Engage SFDC Adapter for Outbound



Installing the Engage SalesforceIntegration package in your Salesforce environment is easy. With just a few clicks the adapter is ready to use. Once you've logged into salesforce.com with your user name and password, simply click the following link to begin the installation:

Genesys Outbound Adapter (password is "*Genesys1*")
Choose your security level to begin the download. As you follow the prompts at the bottom of the page, you will be asked to approve third-party access in order to complete the installation. Once the package has been installed successfully, you'll see the **Installation Complete** message displayed. Now you just need to verify the configuration settings are installed and that there is a remote site record. These next sections will help you with those tasks.

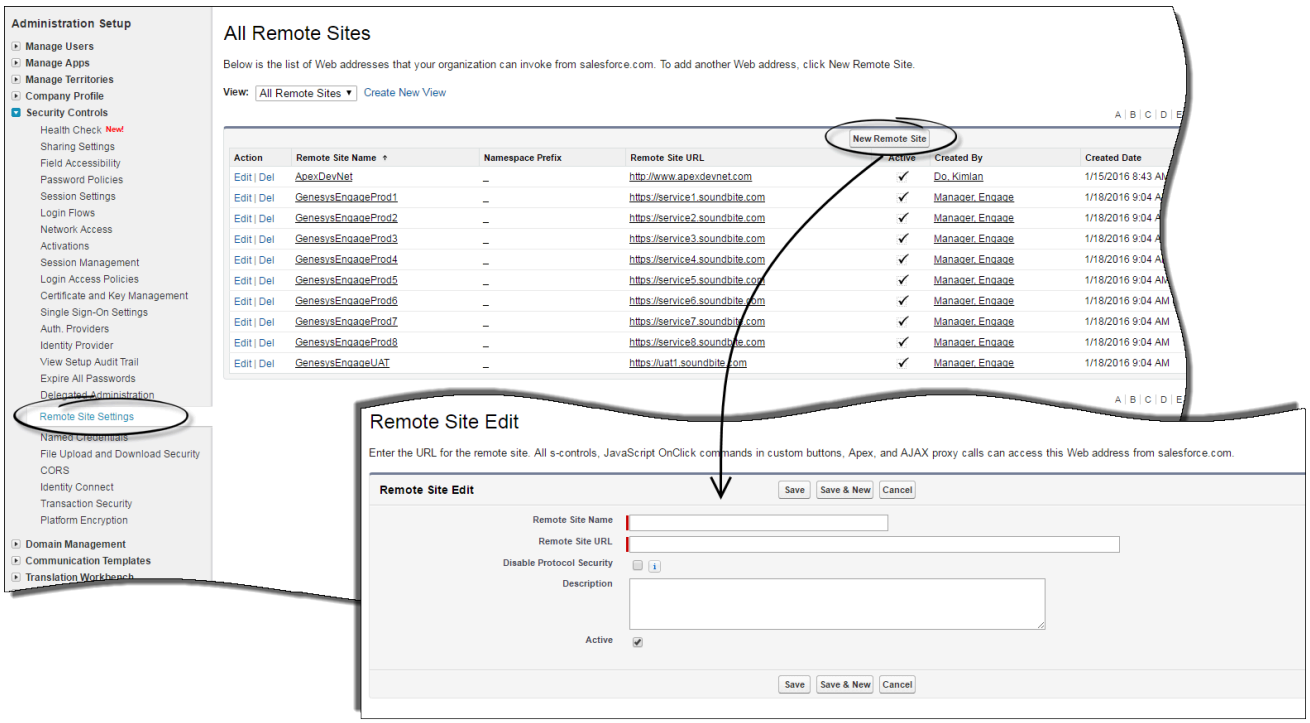
How to verify the configuration settings are installed



To check that your configuration settings were installed properly, navigate to **Custom Settings** found under **Develop** in the **Setup** area. Here you should see the **Engage Connection** record which stores the configuration settings for Engage.

One more step and you're done. Next, verify that there is a remote site record created.

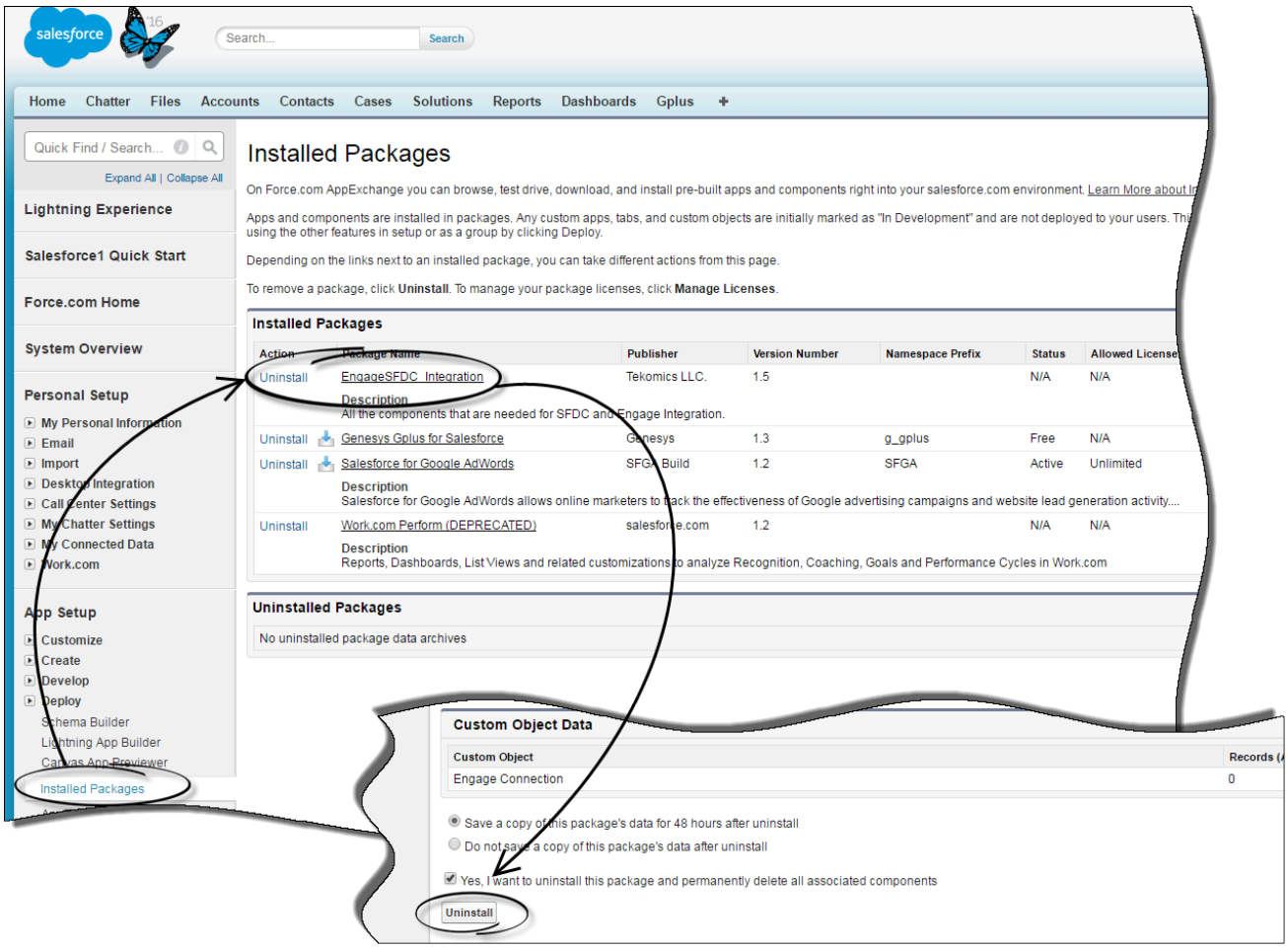
How to verify a remote site has been created



To check that there is a remote site created for Engage, you'll need to navigate to the **Remote Site Settings** option, found under **Security Controls** in the **Setup** area. Click on the **New Remote Site** button, add in your remote site details in the new window that opens and **Save** your changes.

Congratulations! Now that you've completed all the above steps, you're ready get started with your Engage SFDC Adapter for Outbound!

Uninstalling



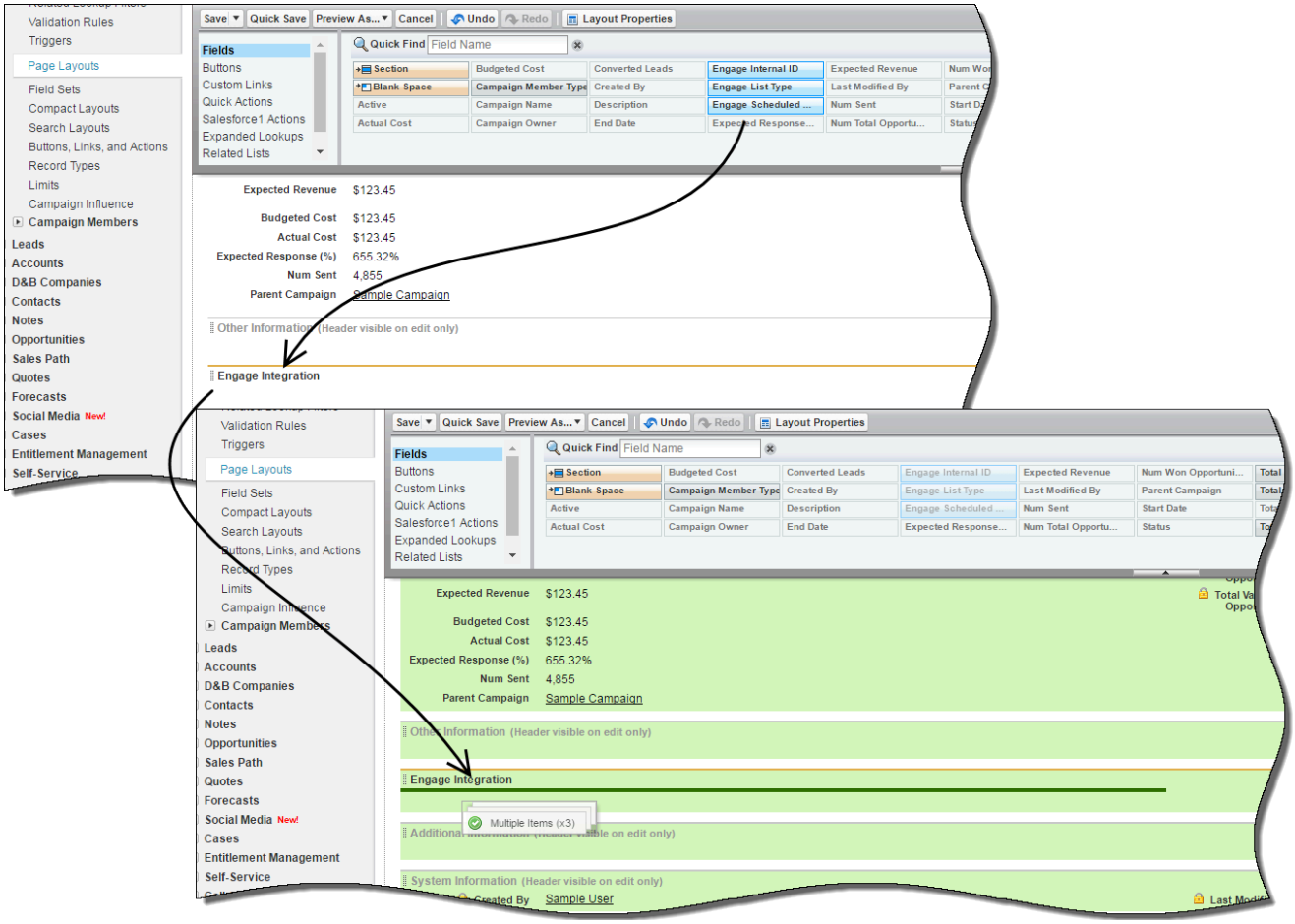
Should you wish to uninstall the adapter, you only need navigate to your **Installed Packages**, found under **Deploy** in your **Setup** menu. Simply click on **Uninstall** next to the adapter name and scroll to the bottom of the page where you'll find the removal options. Choose whether or not you wish save a copy of the data for 48 hours after the uninstall, check the check box next to the removal agreement and then click **Uninstall**. You're automatically brought back to the Installed Packages window where you'll now see the adapter name under the Uninstalled Packages section.

Configuration features

Page Layouts

Drag and drop makes configuring page layouts a breeze. Below we demonstrate how you can make your campaign layout work for you. It all starts from the **Page Layouts** menu found under **Campaigns** in **Customize**. From here you can click **Edit** next to your campaign name to perform the following configurations:

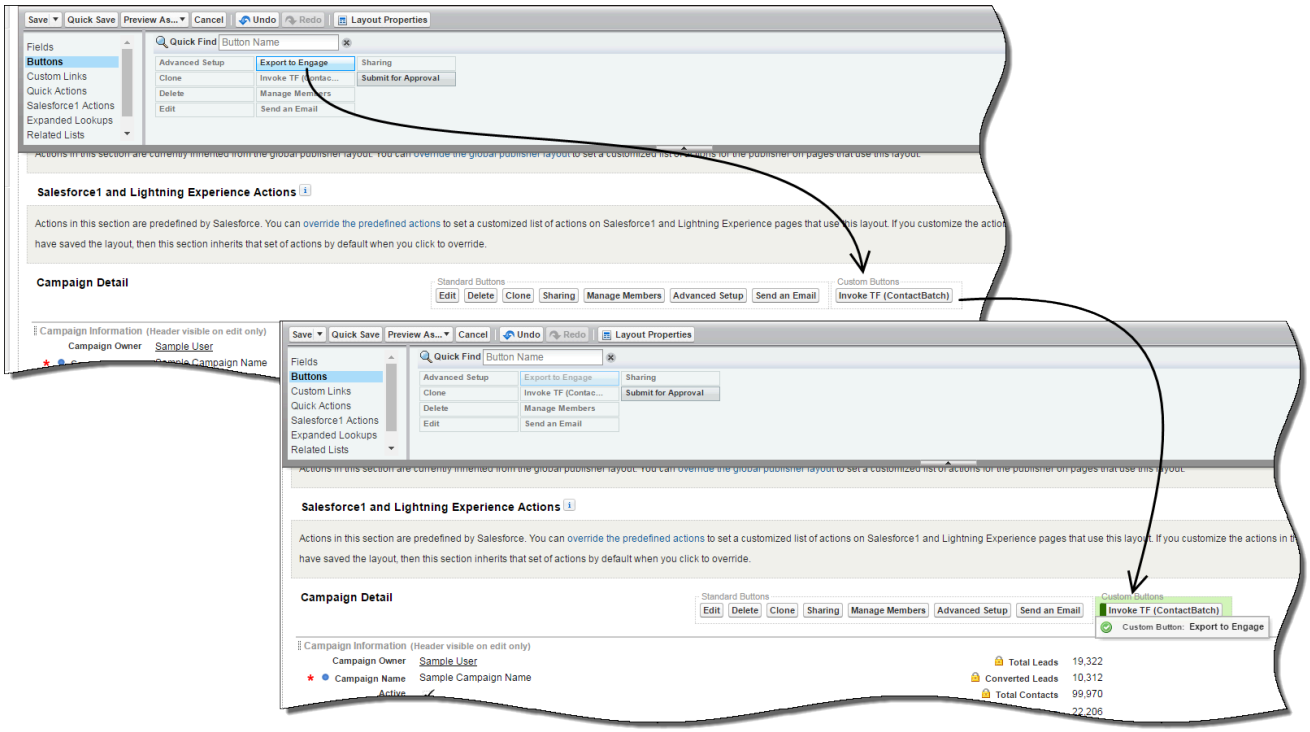
Custom Fields



To add new **Fields** to your campaign layout, simply highlight the fields from the top menu and drag them down to the main screen. In this case, we're dragging the **Engage Internal ID**, **Engage List**

Type, and the **Engage Scheduled Sync** down to the **Engage Integration** section. Once you're done, click the **Save** button in the top menu to save your changes.

Custom Buttons

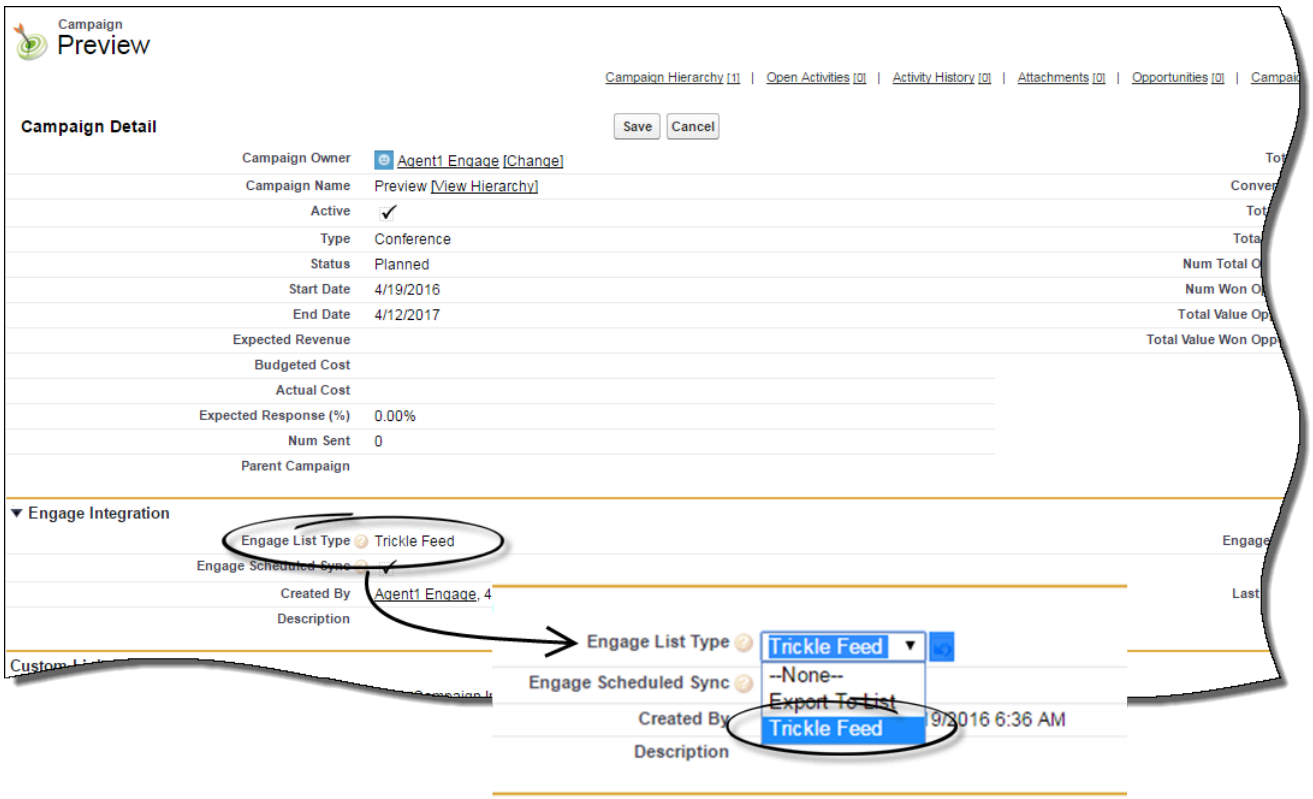


To add new **Buttons** to your campaign layout, highlight the button from the top menu and drag it down into the **Standard Buttons** or **Custom Buttons** menu. Once you're done, click the **Save** button in the top menu to save your changes.

Moving Sections

Each section also has the ability to be moved, making the campaign layout truly your space. To move a section you only need drag it to the new area and drop it in. Notice how the green bar and the green check mark appear as you begin to drag the heading? This is how you know you're completing the action correctly. Once you're done, click the **Save** button in the top menu to save your changes.

Configuring Trickle Feeds

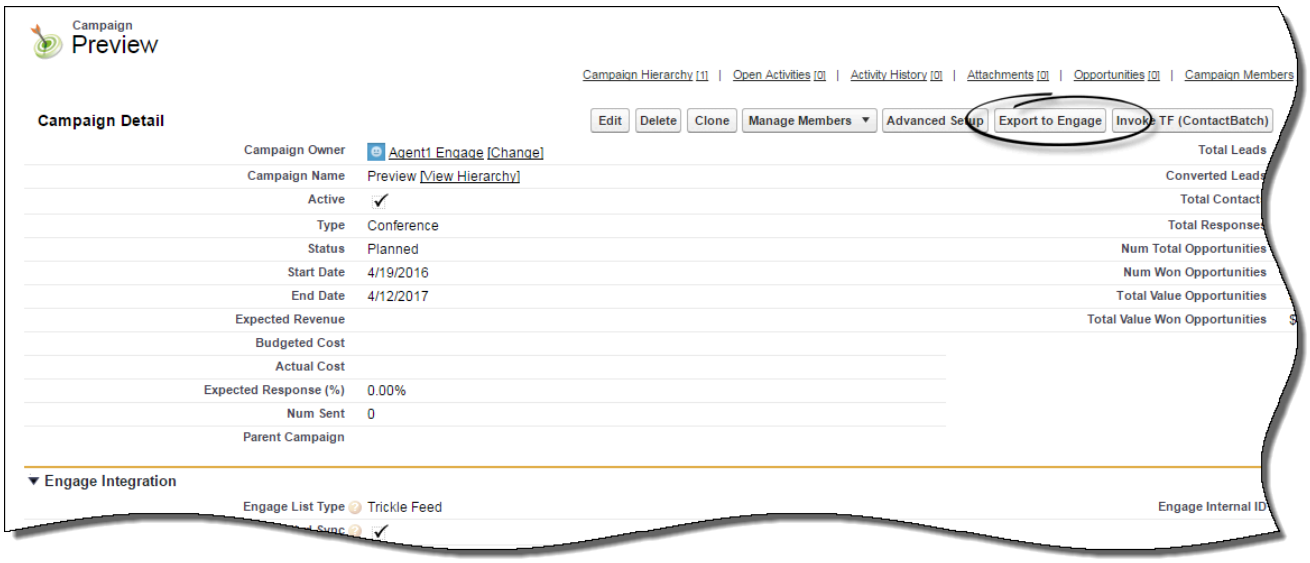


Important

Before configuring Trickle Feeds, you must first associate a Strategy with a Campaign that has the same name as the campaign in Salesforce.

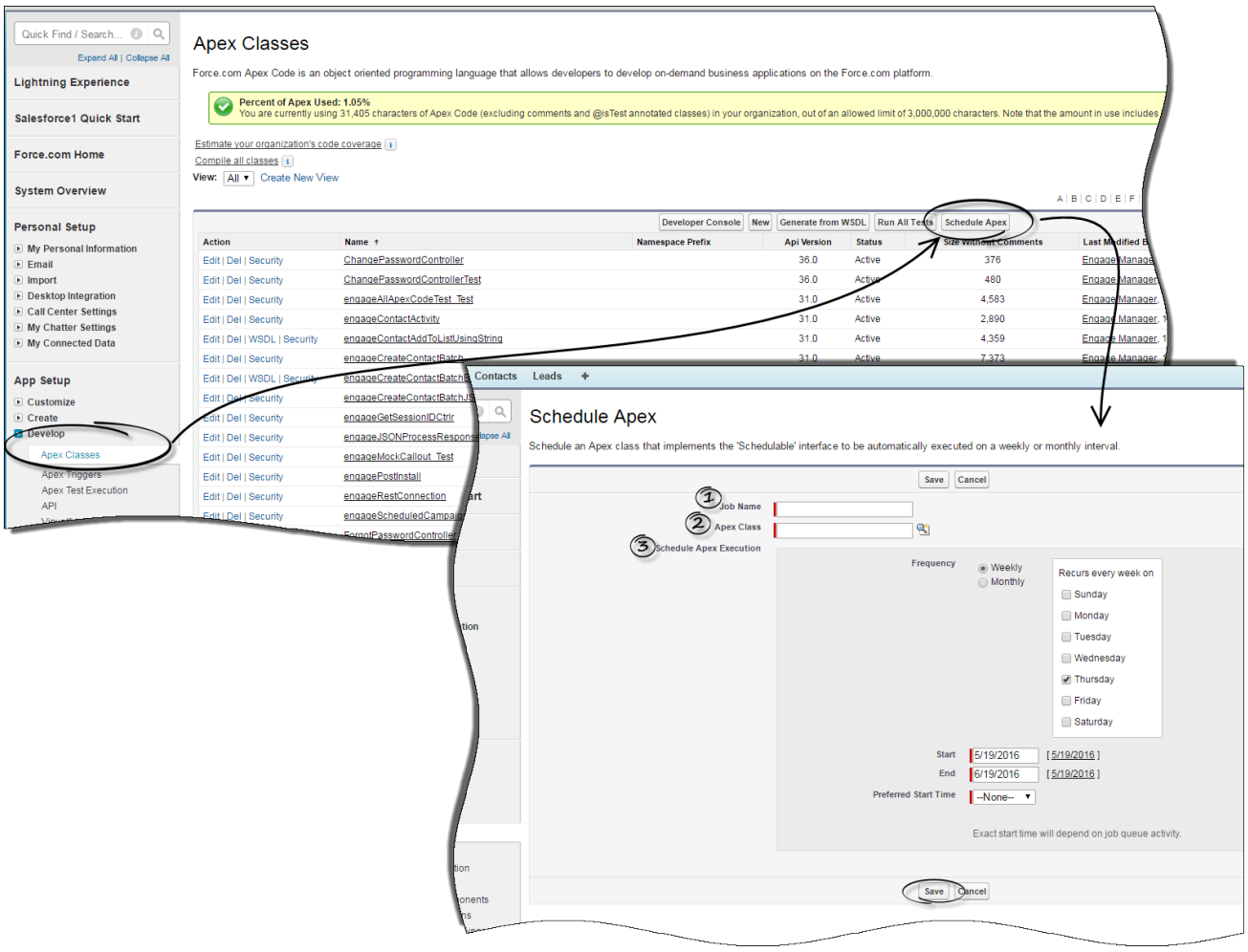
You can configure a campaign so that new contacts are added to the corresponding Engage sub-campaign on an on-demand basis, rather than a single batch upload. You do this by clicking on the **Campaigns** tab and choosing **Trickle Feed** from the **Engage List Type** in your campaign record. Now, as you add campaign members to the campaign, they are transferred to your sub-campaign. Note that Salesforce sends this data using asynchronous calls so there might be a few seconds delay in the transfer.

Exporting



Now that you have your Engage SFDC Adapter for Outbound installed and configured, exporting your campaign information is easily done by clicking on the **Export to Engage** button from the campaign details view. Within moments you should have a message appear indicating that your export was successful. To verify that all details have been exported, simply log in to your Outbound account and check for the campaign name under your **Lists** tab.

Scheduling campaign exports



You can use the adapter to schedule an export. Use the **Apex Classes** found under the **Develop** section in the **Setup** menu. Once you click on the **Schedule Apex** button from the menu, another window opens asking you for the following information:
1- *Job Name:* this is a name you choose for your job
2- *Apex Class:* click on the Apex Class Lookup icon and choose **engageScheduledCampaignSync**
3- *Schedule Apex Execution:* these settings are up to you
Once you've completed all the sections, click the **Save** button at the bottom of the page and your export is now scheduled.

On Demand Strategies for Trickle Feed

On Demand Strategies are used to manage and control outbound trickle feed dialing activity within a campaign. This allows users to set the dialing mode and calling parameters for Salesforce records uploaded into your Outbound account for immediate contact.

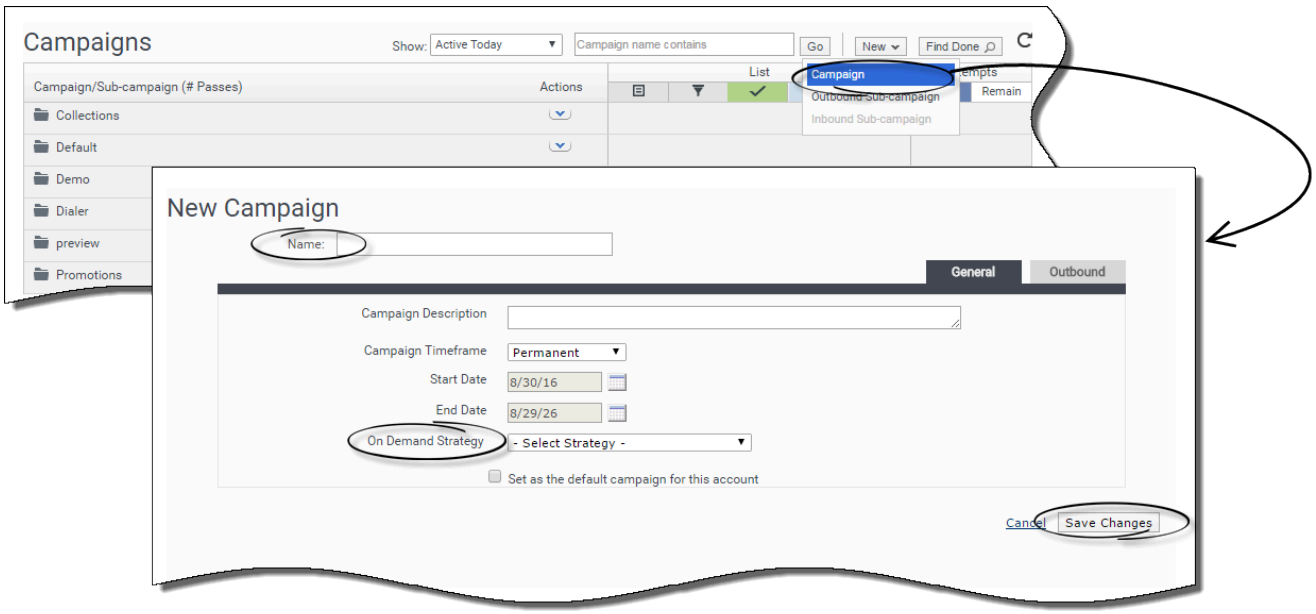
In order to configure a trickle feed, you must create an On Demand Strategy and then associate that Strategy to a Campaign with the same name as the Salesforce campaign:

Creating an On Demand Strategy

Login to your Outbound account and select **Strategies** from the **Campaigns** tab, then select **New** to

create the strategy.
In the main window, give the strategy a name, and select the script and target agent group. Then define the channel (dial mode) and any calling parameters you require.

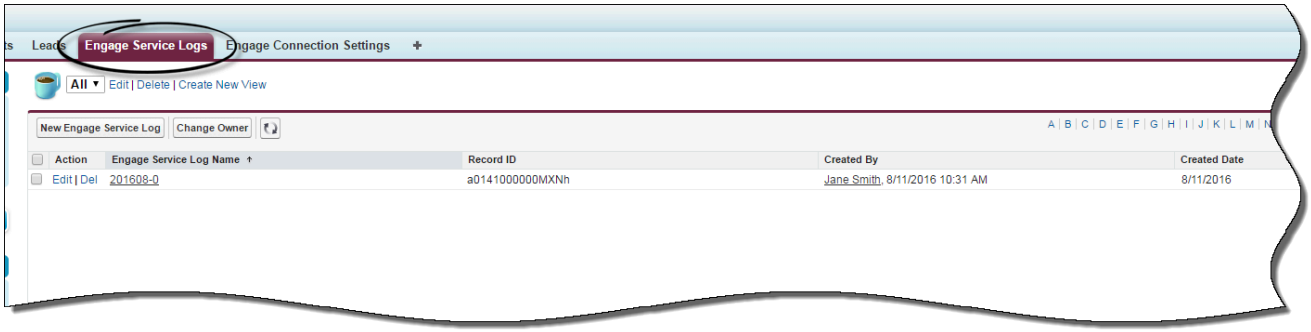
Associating a Strategy with a Campaign to invoke Trickle Feed



From the **Campaign** tab, select **New** > **Campaign**.
In the main window, ensure the campaign name is identical to the campaign name used within Salesforce then select the On Demand Strategy you just created as the On Demand Strategy for this campaign. You can also continue to setup campaign level strategies for standard dialing activity to lists exported from Salesforce.
Once the campaign is configured and saved, you can invoke the Trickle Feed function from within Salesforce. This is done during the Salesforce campaign setup. Simply select Trickle Feed as the contact method when creating the campaign.

Service Logs

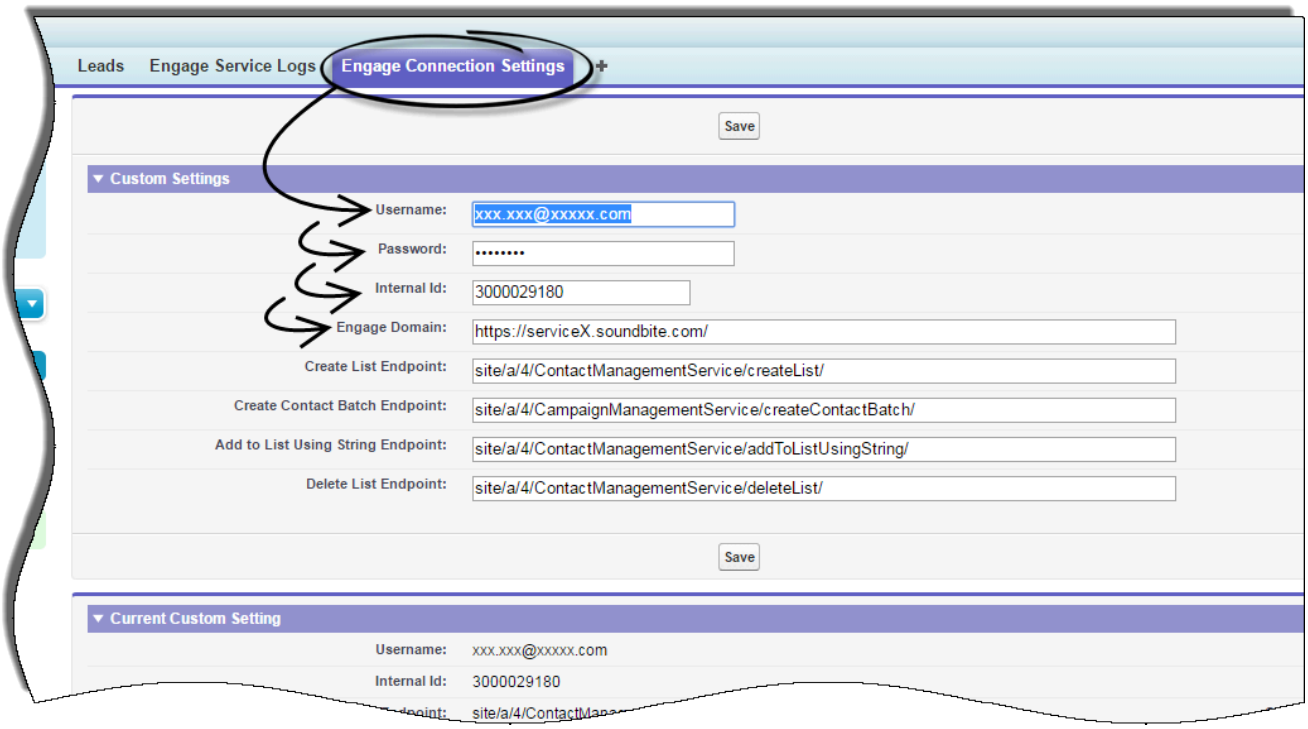
Service Logs are where all Salesforce API exchanges with your Outbound account are recorded and stored. Each interaction consists of a Request and Acknowledgment. These API exchanges are stored as a service log every time this communication is invoked.



From the Engage Service Log tab at salesforce.com, you can view, edit, and delete your recent Engage service logs as well as create new service logs manually.

Connection Settings

The Engage Connection Settings tab in salesforce.com is where you set your login credentials for your Outbound account so that Salesforce can securely connect and communicate with it.

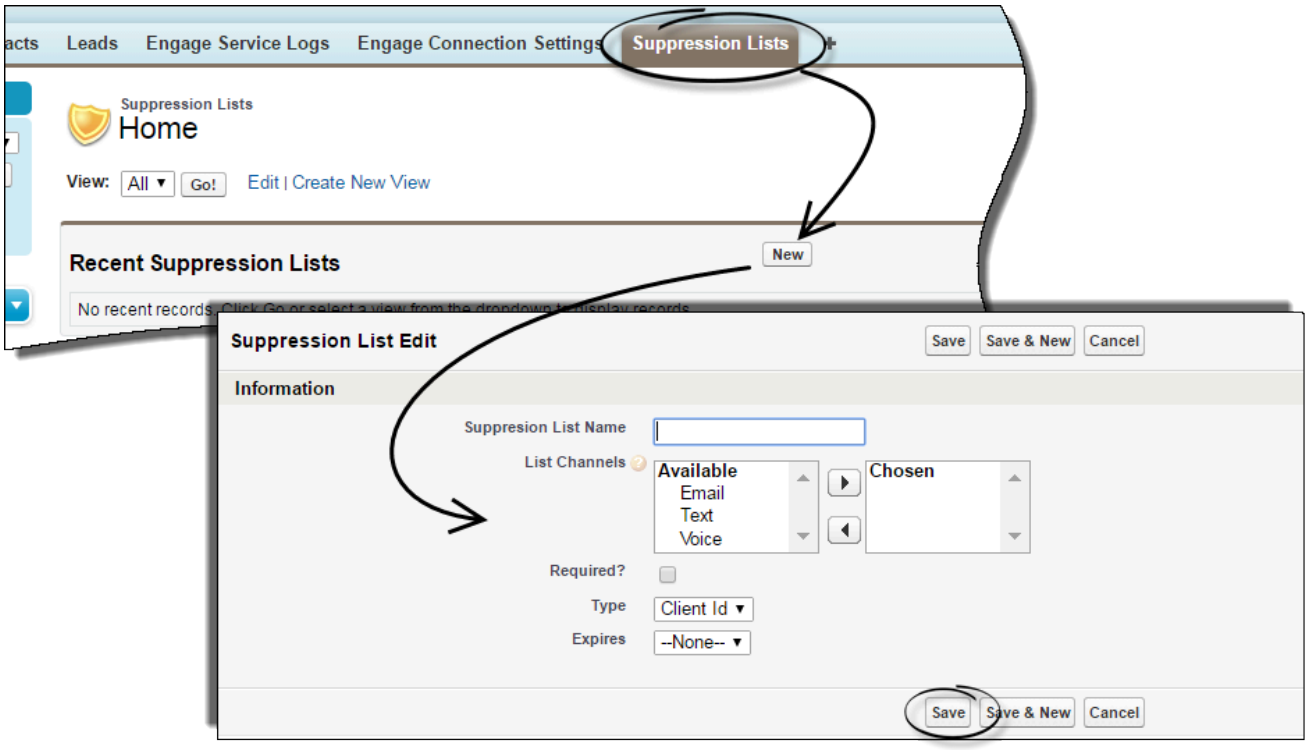


When creating custom settings, important fields to include are **Username**, **Password**, **Internal ID**, and the **Engage Domain**. **Note:** In the Engage Domain field, the address includes "ServiceX" where "X" is a number (for example, "Service3").

Suppression Lists

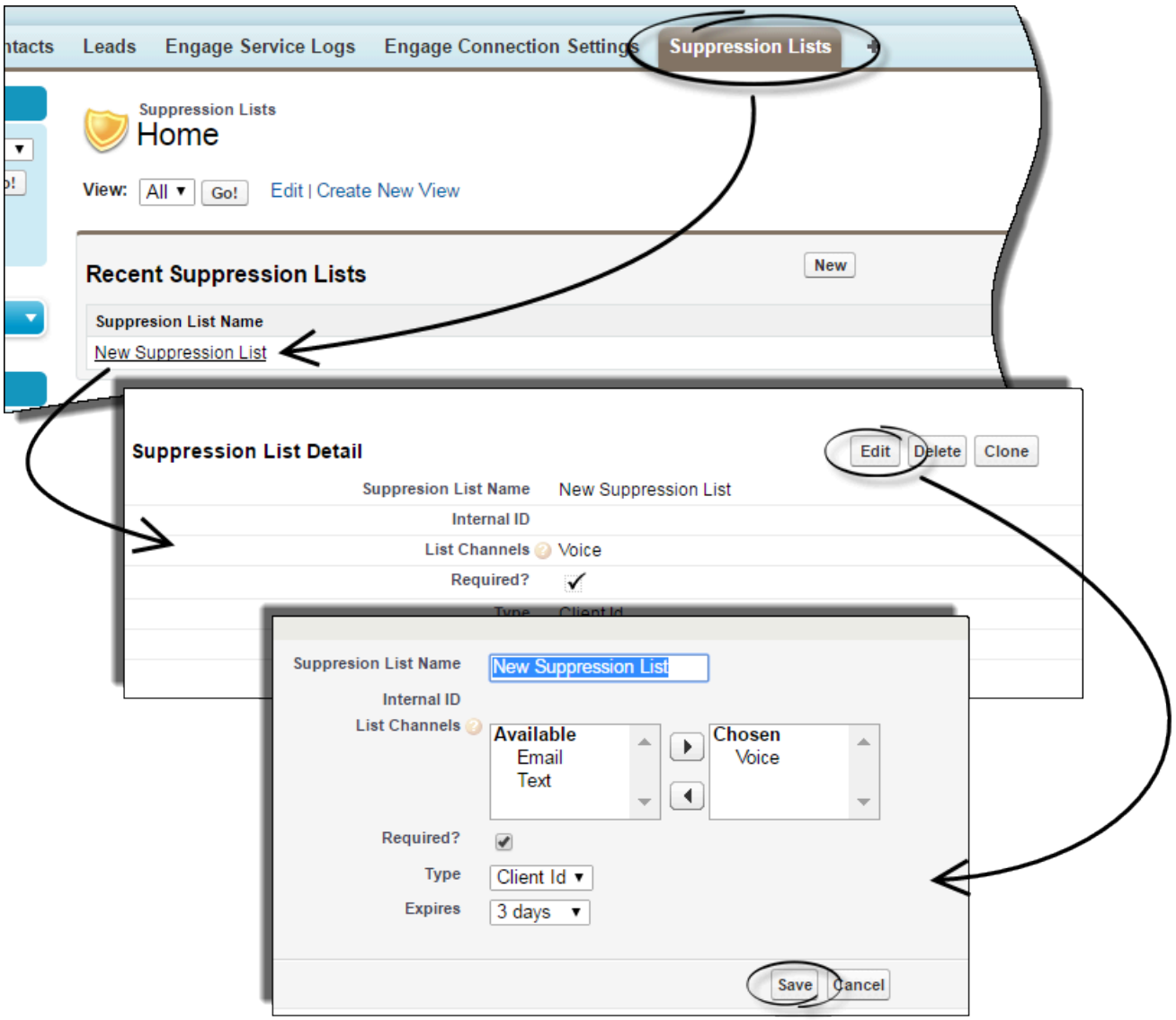
Suppression lists are used to either partially or permanently exclude contacts from any future communications from your company.

Creating a new suppression list



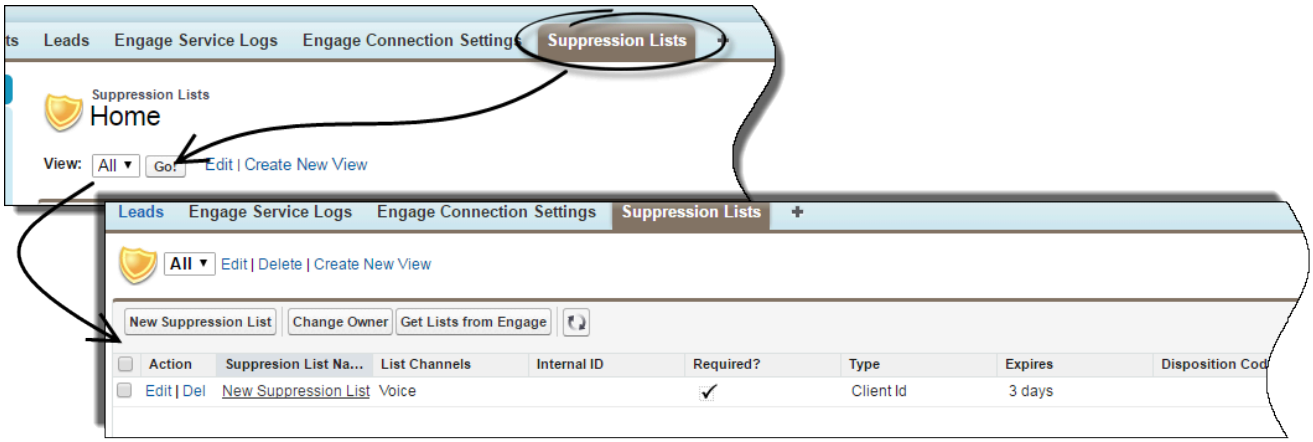
From the **Suppression Lists** tab, click **New** and fill in the appropriate fields for your new suppression list. Pay special attention to the **List Channels** and **Expires** fields. The **List Channels** field identifies which channels you wish to include in this suppression list by moving one or more channels from the **Available** field to the **Chosen** field. The expiry date can be anywhere from **never** to **90 days**. All other fields are self-explanatory.

Editing suppression lists



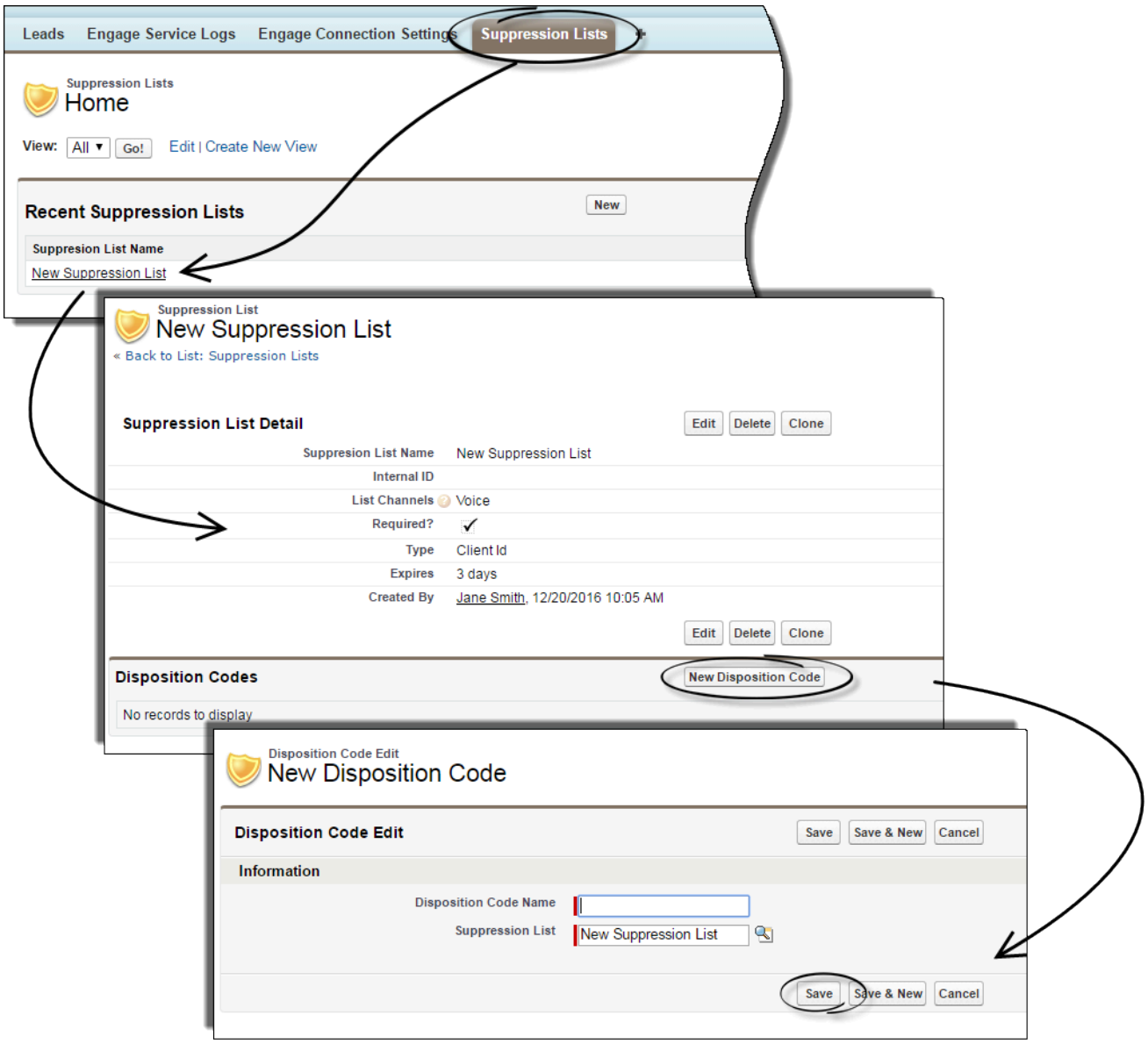
Perhaps you wish to either shorten or extend the duration of a suppression list, or maybe you've decided that you want to make this suppression list *required* in your active campaigns. Editing your suppression list is how you would do that. Simply click **Edit** from the **Suppression List Detail** screen and edit the fields to your preference.

Retrieving current suppression lists



To see a list of all your suppression lists synchronized with Salesforce, from the **Suppression Lists Home** screen make sure the **View** is set to **All** and click **Go!**. If necessary, click on the refresh button  to refresh the view.

Assigning disposition codes to a suppression list



Disposition codes are used by agents to track call outcomes, and these outcomes are found in the **Call Status** field under **Activity History**. The disposition code options for your campaigns are set up by your Administrator.
You can assign certain disposition codes to trigger a suppression event in Engage. To identify a disposition code that will cause a customer to be added to a suppression list, click **New Disposition Code** from the **Suppression List Detail** view, then define the **Disposition Code Name**. The name of the disposition code needs to exactly match the existing disposition code as set up by your Administrator. You can add one or more disposition codes for association to any of your suppression lists.

How to set up Outbound routing

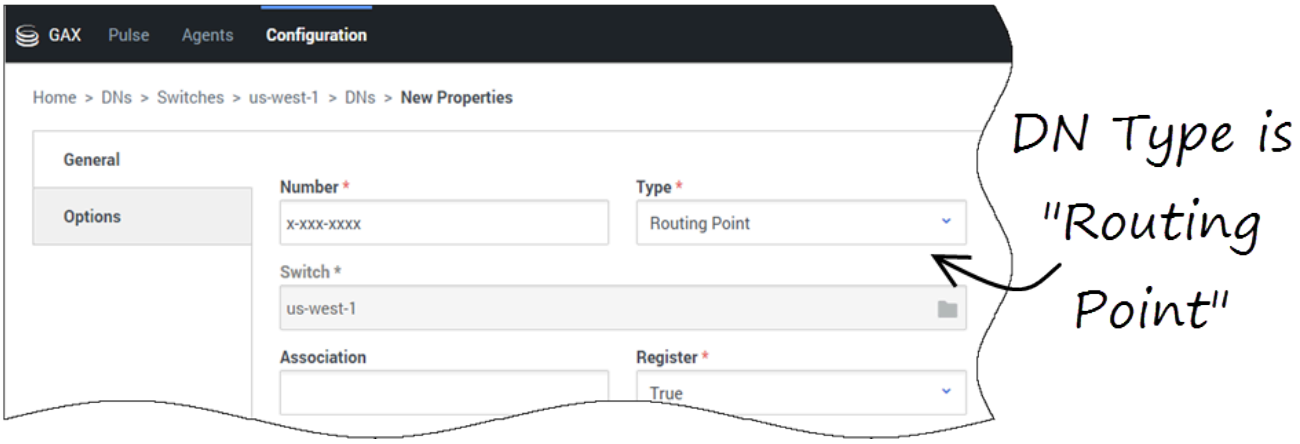
This page shows an example of how Outbound routing could be set up. This example assumes that the site already has a trunk group for Outbound and that the person doing the configuration has access to Platform Administration and Designer.

Note: Keep in mind that this is just one possible method and that your site might require a different approach.

Basically, this type of setup involves the following steps:

- Create a Routing Point DN on the primary SIP Server.
- Create a Virtual Queue DN on the statistics server.
- Create a new Agent Group for outbound agents and assign them to it.
- Create a new Designer application that ties it all together.

Create a new Routing Point DN

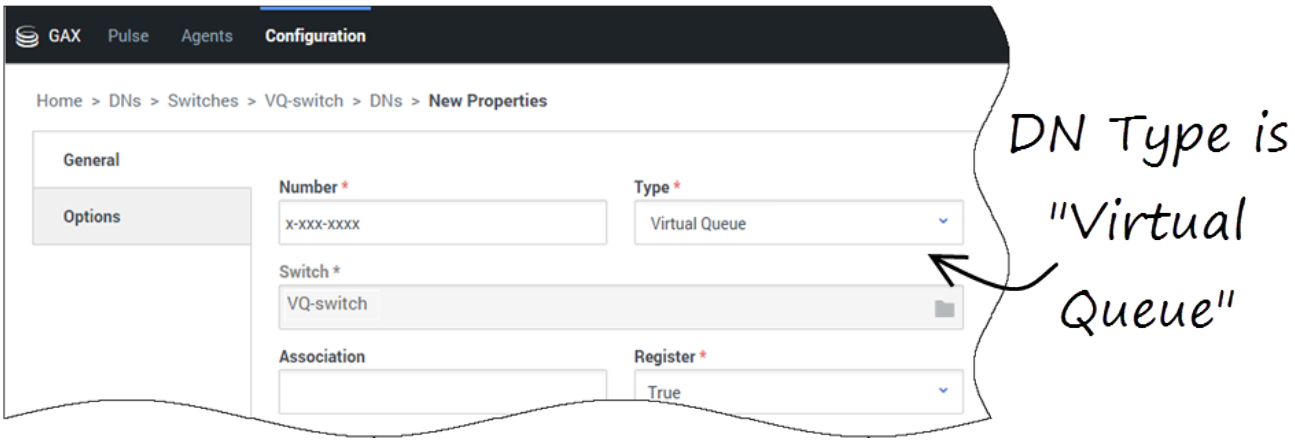


In Platform Administration, go to **Switching** and select **DNs**.

On the primary SIP Server switch, browse to the folder where you want to add the new DN.

Add a new DN and choose **Routing Point** as the **Type**.

Create a new Virtual Queue DN

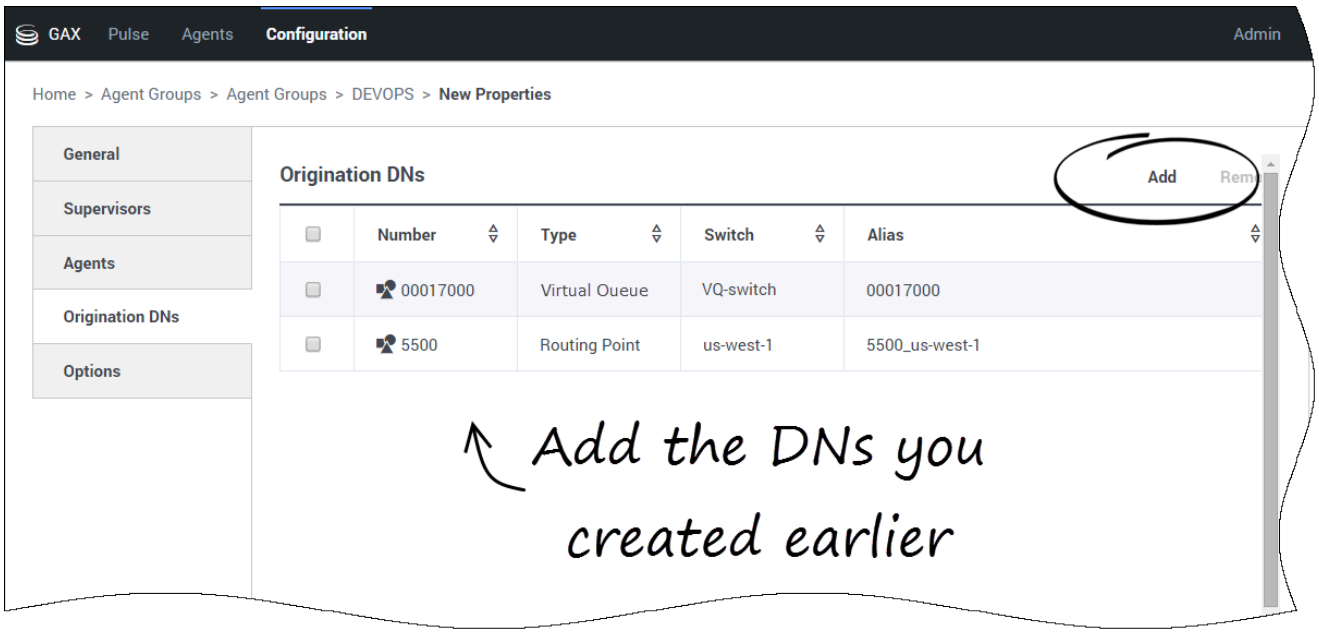


In Platform Administration, go to **Switching** and select **DNs**.

On the statistics switch, browse to the folder where you want to add the new DN.

Add a new DN and choose **Virtual Queue** as the **Type**.

Create an Agent Group for Outbound agents



Under **Origination DNs**, add the **Routing Point** and **Virtual Queue** you created earlier.

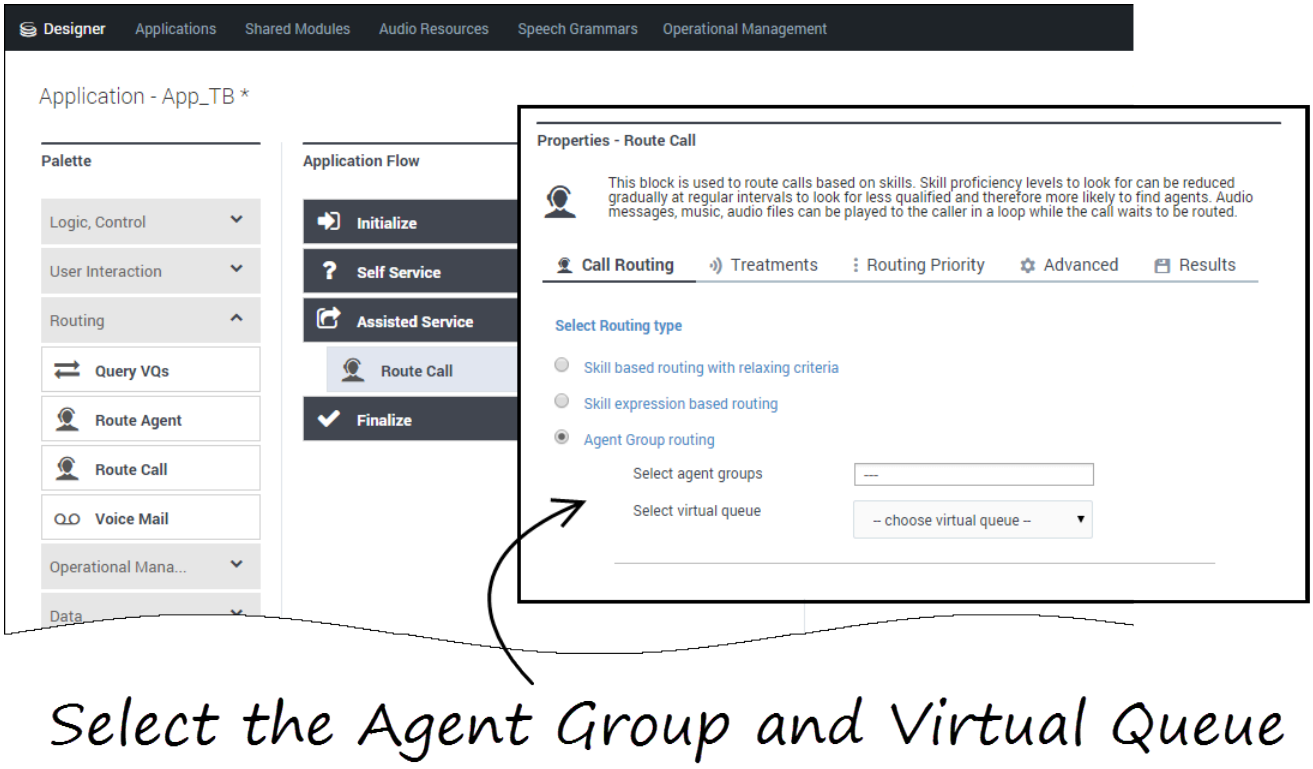
Next, go to **Options**, and add a new option for the Outbound account time zone:

- **Section** = default
- **Key** = tz
- **Value** = (enter the time zone of the OCS account)

Now you can assign your Outbound agents to the new Agent Group. Go into the Agent Group properties, select the **Agents** tab, and choose the agents you want to add.

Note: New Agent Groups are automatically synchronized with the OCS account, so you'll be able to see them when creating campaigns.

Create an application for Outbound routing



In Designer, create a new application to use for Outbound routing. (Need help with this? See the Designer help.)

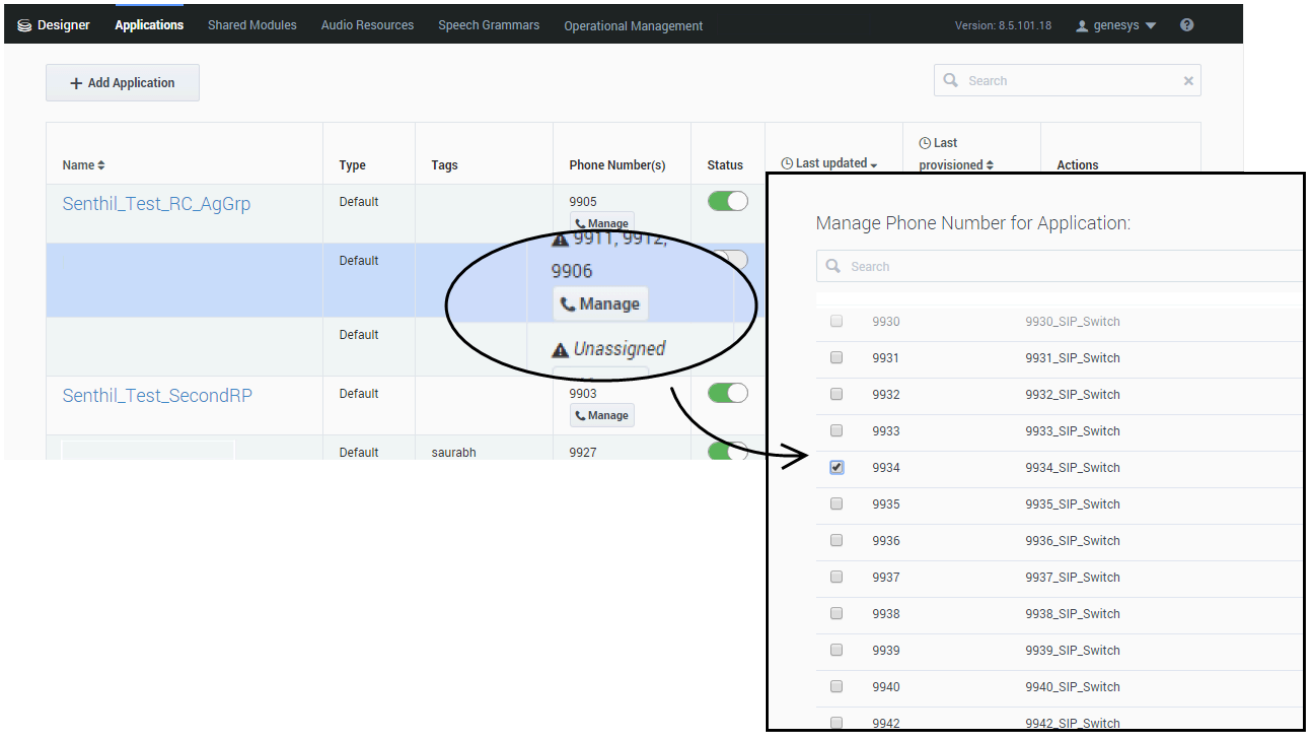
From the Palette, expand the **Routing** section and add a **Route Call** block to the Assisted Service phase of the Application Flow.

In the application properties, under **Call Routing**, select **Agent group routing** and choose the Agent Group and Virtual Queue you created earlier.

Next, under **Treatments**, add a **Treatment for Busy** to your application.

When you are done, save and publish the new application.

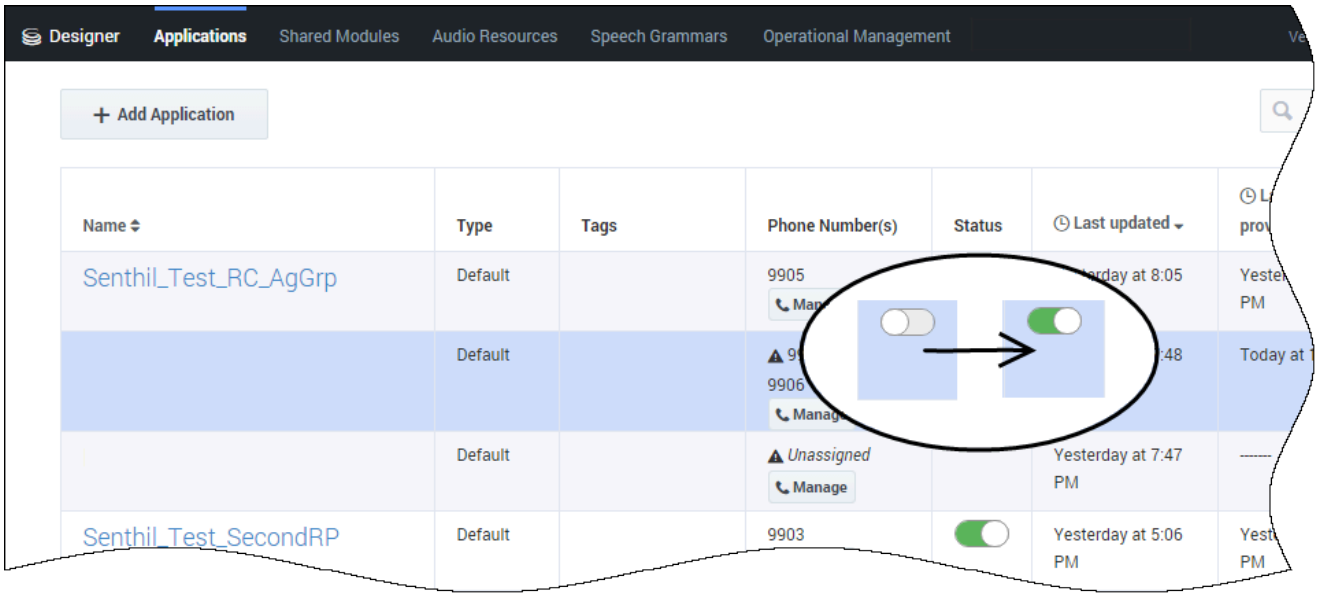
Assign the Routing Point to the new application



In Designer, find the new application in the main list and select **Manage**.

From the list of **Phone Numbers**, select the Routing Point DN you configured earlier.

Enable the application



Change the application **Status** by moving the slider to green (enabled).

Cloud Data Download Service

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

Cloud Data Download Service (CDDS) enables you to securely export and download your contact center data.

The exported data files are encrypted with your public encryption key and stored for 30 days, during which time you can download the files and decrypt them using your private encryption key.

After 30 days, the data files are deleted.

What types of data are exported?

Cloud Data Download Service supports the export and download of Contact History data. When you set up a data export job, you'll select which type of data you want to export:

- **Interaction** data contains details about interactions that were processed during the time period specified in the job settings, such as voice calls, chats, and emails.
- **Contact** data contains details about each customer, such as their name, customer ID, email address, and the date that their contact information was created or modified.

> **Important**
> Info Mart exports are not supported.

See the Export page for information about downloading, decrypting, and viewing the exported files. For information about the schemas used for Contact History export files, see the Contact History Export Reference page.

Requirements

CDDS requires public/private keys for the encryption and decryption of data files and a valid X.509 RSA-compliant certificate. Before you can export data, you must have a valid encryption certificate specified in your user preferences.

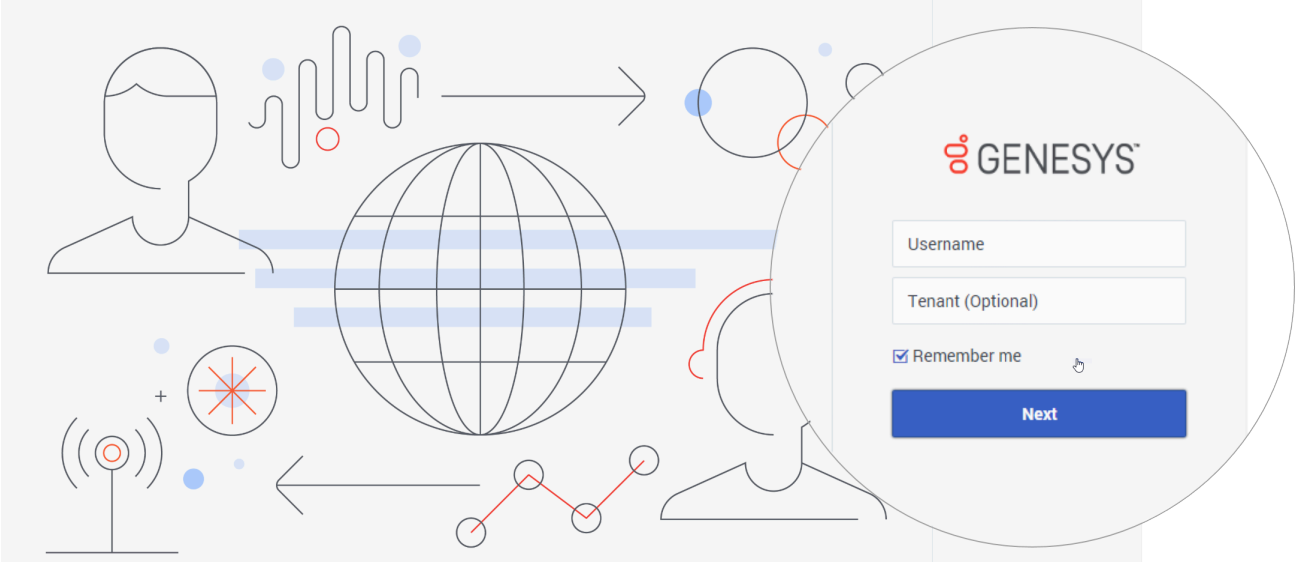
To learn how to generate an encryption certificate you can use with CDDS, see Certificate requirements.

General Data Protection Regulation (GDPR) compliance

In general, Genesys support for GDPR compliance is based on default configuration settings and typical application usage. Cloud Data Download Service, like other underlying components within Genesys Engage cloud, does not store sensitive information beyond 30 days. Users who download their data using this tool are responsible for GDPR compliance in regards to any data that they have downloaded.

For more information, see Genesys Engage cloud Support for GDPR.

Logging in



To log in to CDDS, enter your **Username**. You can also specify the **Tenant** name, but this is optional.

If you want your details to be saved for the next time you log in, select **Remember me**.

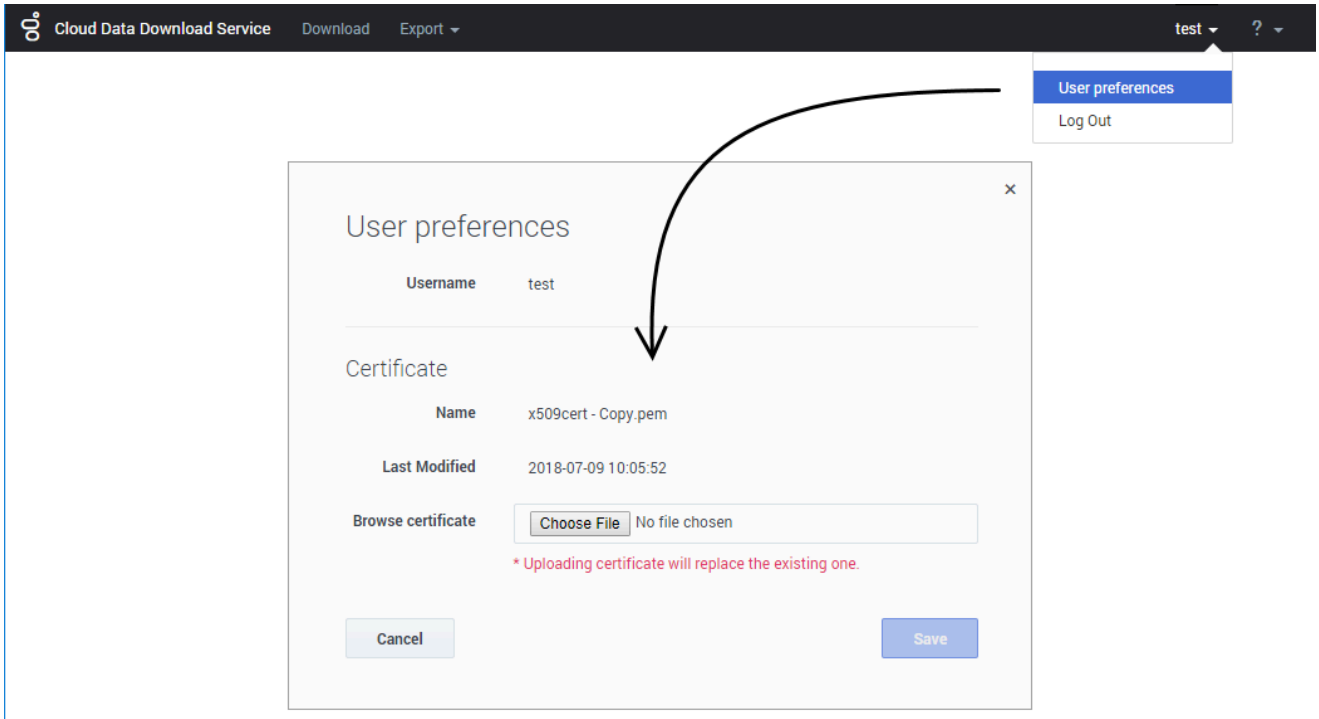
Click **Next**.

Enter your **password** and click **Sign in**.

> **Important**
> If you don't have the permissions required to access CDDS, you'll be taken to an **Access Denied** page. If you feel this is in error, ask your administrator to check your

permissions.

Importing an encryption certificate



Before you can export data, a valid encryption certificate must be specified in your user preferences. To learn how to generate an encryption certificate, see Certificate requirements.

To add (or change) your encryption certificate, click your user name and select **User preferences**. Choose the certificate file you want to use and save your changes.

Important

Careful! When you upload a certificate, it replaces any certificate that was previously specified.

Logging out

To log out, click your user name to open the drop-down and select **Log Out**.

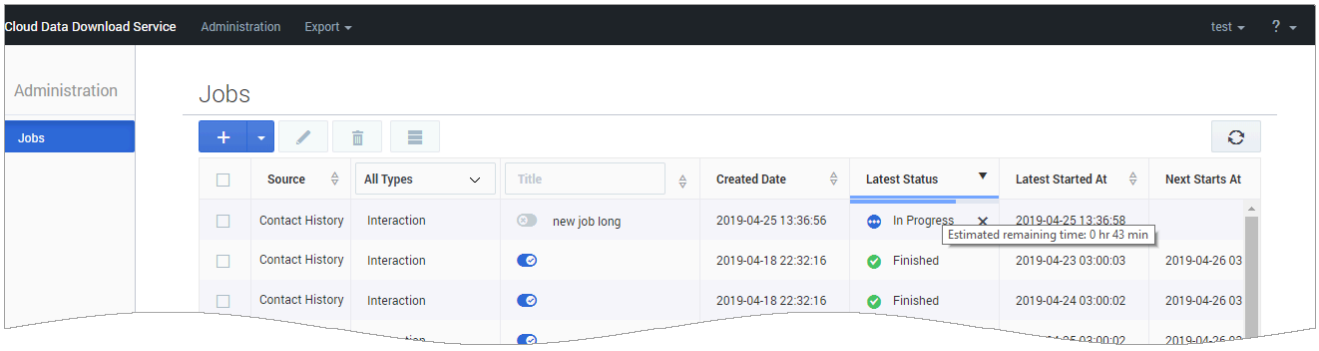
Administration

> **Important**
>
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

From the CDDS **Administration** menu, you can access the **Jobs** page. The **Jobs** page displays a list of your data export jobs. From here, you can create a new data export job or monitor and manage existing ones.

> **Tip**
>
> You can view full details for all active and completed jobs on the History page, which you can access from the Export menu.

The page provides details about each job, such as data source, job type, latest status, and date when each job last ran. If a job is still running, the **Latest Status** column features a status bar to indicate the progress of the active data export job. If you hover your mouse over the bar, it displays the estimated time remaining until the job completes.



Adding a new job

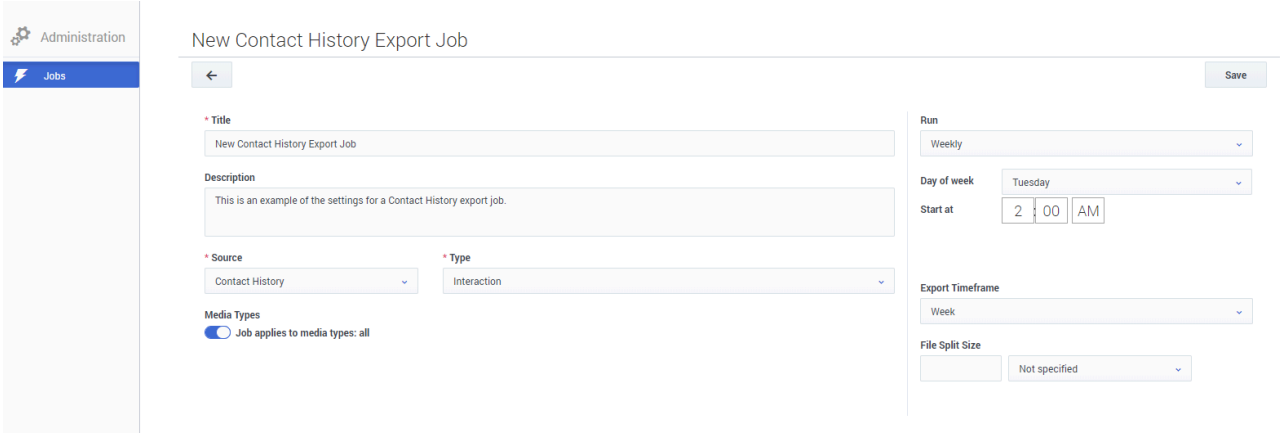
To create a new job, click the **Add new job** button:

This opens the job editor page, where you can specify the settings for the new job. Mandatory fields are marked with an asterisk (*).

> **Important**
>
> Info Mart exports are not supported. Do not select Info Mart as the data source for your job.

You must give the job a **Title**. You can also add a **Description**, but this is optional.

After you select **Contact History** as the data **Source**, you can select either **Interaction** or **Contact** as the data **Type**.



By default, **Media Types** is enabled for **all**. If you turn the slider off, you must select at least one media type (for example, **Chat**). You can add multiple media types.

Use the **Run** setting to specify how often to run the job. You can then choose to run the job immediately or schedule it to start at a specific date and time.

> **Important**
>
> - If you select **Hourly**, the start time is calculated based on 00:00 UTC (Coordinated Universal Time) and not the local timezone. For example, let's say you want the job to run every 4 hours. The first run of the job will be at 00:00 UTC, the next at 04:00 UTC, then 08:00 UTC, and so on.
> - Also, note that CDDS uses Coordinated Universal Time (UTC) to calculate the start time for scheduled jobs and not the local timezone. As a result, job start times are not automatically adjusted to align with changes to the local timezone, such as for Daylight

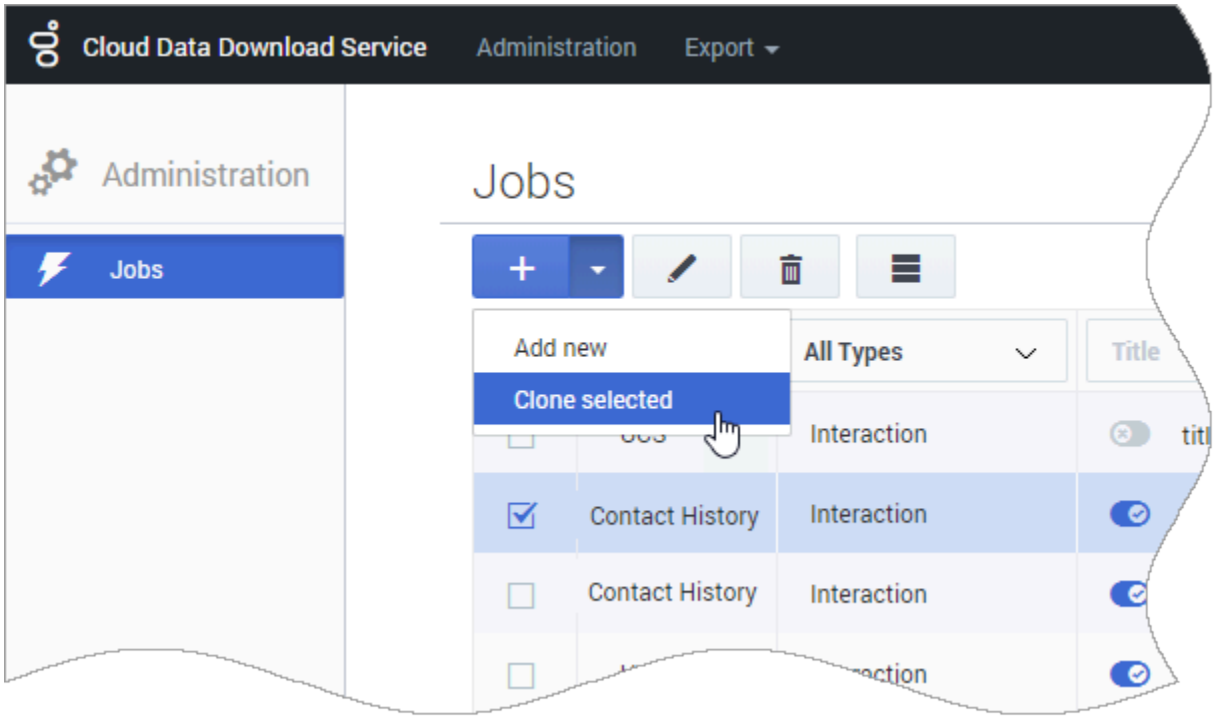
> Savings Time. After a local timezone change, review the start times for your scheduled jobs and adjust accordingly.

The **Export Timeframe** setting lets you export all data or just the data for a specific time period. If you select **Fixed interval**, you can then specify the **Start** and **End** dates for the export.

By default, the data is exported as a single file. If you want to break the exported data into a set of smaller files, use the **File Split Size** setting to specify the size of each file.

When you are finished setting up the job, click **Save**.

Cloning a job



If you want to create a new job that is based on the settings of an existing job, you can clone the job.

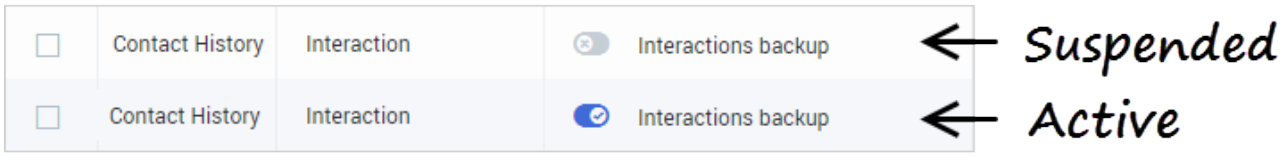
Select the job you want to clone and choose **Clone selected** from the dropdown menu. This opens the job settings editor.

You can keep the existing job settings, or make any desired changes. For example, you might want to give the job a different **Title** or adjust the range of the **Export Start/End** dates.

When you are finished, click **Save**.

Suspending (and resuming) a job

Use the slider to change the state of a job to **suspended** or **active**.



Important

If you create a one-time data export job (i.e. a job you intend to only run once) and submit it to run immediately, it is automatically added in a suspended state and can't be resumed. If you create and submit a one-time job with a delayed (i.e. scheduled) start time, it automatically changes to a suspended state when the job starts.

Editing a job

Select the job you want to edit and click the pencil icon:

When you are finished making changes, click **Save**.

Important

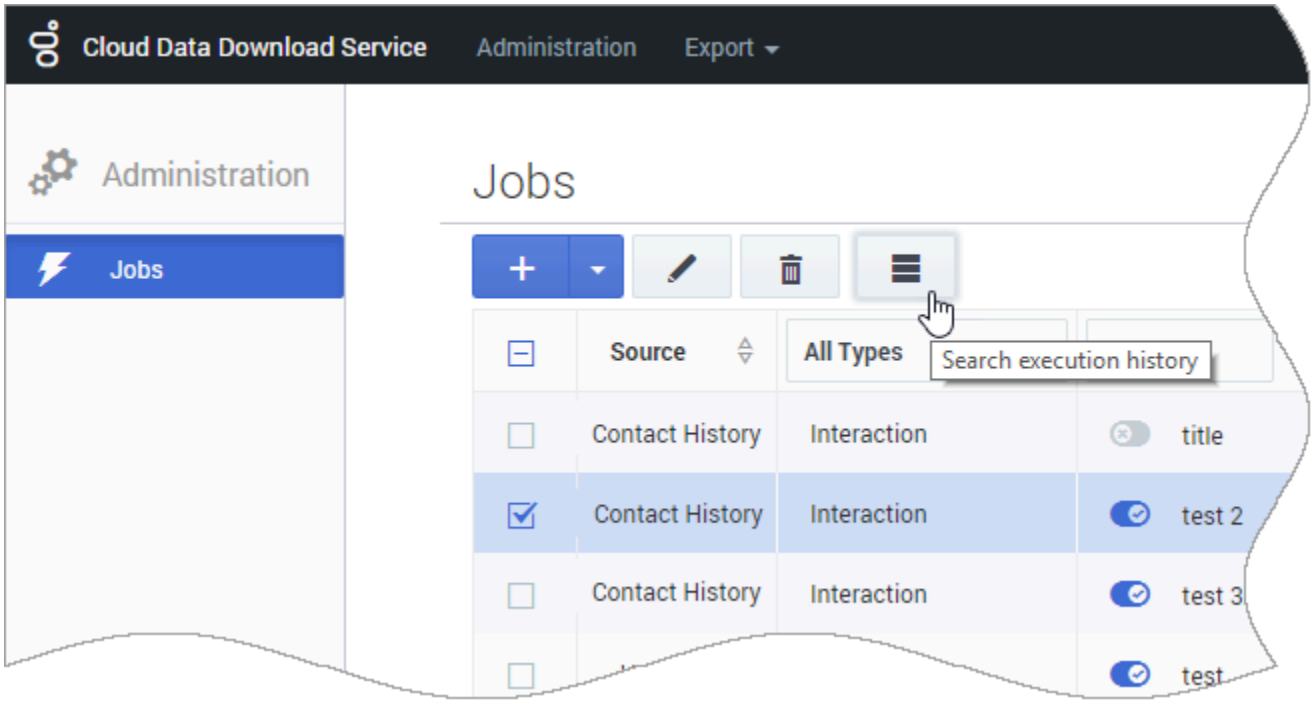
If the data export job is only going to be run once, you won't be able to edit it after it is submitted.

Deleting a job

Select the job you want to delete and click the trash icon:

> **Important**
> You can't delete a job that is still in progress.

Viewing the job history



If you select a job and click the collapsed (or "hamburger") button, it takes you to the History page, where the results are filtered to only show the history of the selected job.

Export

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

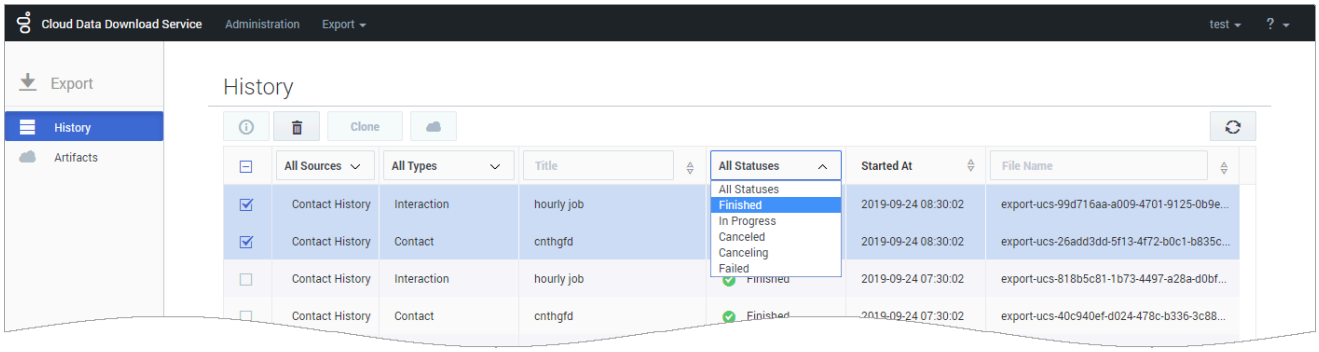
From the CDDS **Export** menu, you can access the job History and Artifacts pages.

History page

The **History** page shows a list of active and completed export jobs, including details such as the data source, job type, current status, start time, and the resulting file name. You can use the column headers to select, sort, or filter the files in the list.

For jobs with a status of **In Progress**, a progress bar indicates that the data export is currently active. If you hover your mouse over the bar, it will show you the estimated remaining time for the job to complete.

From this list, you can select a job and use the toolbar buttons to view additional details about the job, delete the job, clone that particular instance of the job, or locate the related file(s) on the **Artifacts** page.



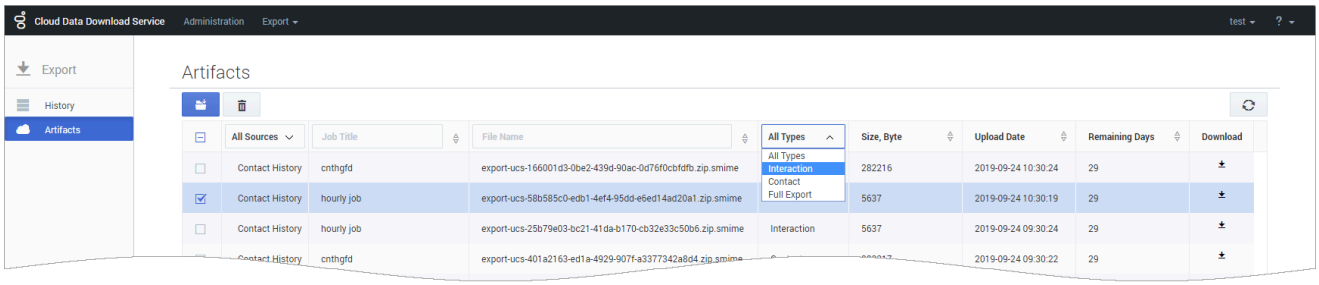
Artifacts page

The **Artifacts** page shows the exported data files that are available for download. You can see

additional details about each file, such as the data type, when it was created, and how many days are remaining in the 30-day storage limit. You can use the column headers to select, sort, or filter the files in the list.

From this page, you can select a file(s) and then use the buttons to **download** or **delete** the selected file(s). You can select up to a maximum of 100 files, but note that each download opens in a separate browser window. Depending on your browser settings, you might need to disable any pop-up blockers.

Alternatively, you can also download a single file by clicking its download icon ( ). This sends the file directly to your specified download location.



> **Important**
>
> Files are stored for a maximum of 30 days. When the **Remaining Days** value reaches zero (0), the file is removed from storage and no longer available for download.

Downloading split files

If a job was set up using the **File Split Size** option, you must download all of the files associated with that job before extracting the data.

You can tell if a file belongs to a split job by the filename — it will indicate that the file is part of a numbered set. For example, **export-2b710a2c-0e7f-4918-983c-c02d108eaeb0.z12.smime** indicates that this file is part 12 of a compressed data file.

When you have downloaded all of the split files for a job, you just need to initiate the extraction for one of the files. The extraction utility will automatically locate and extract the other pieces that belong to the data file.

Decrypting downloaded data files

To obtain exported data files, you must have a valid encryption certificate specified in your CDDS user preferences. During the generation of data files, the original files are zip-compressed and packaged in an S/MIME message with an enveloped-data content type that is encrypted using the public key of the certificate you provided. Once you download the data files, you can decrypt them by using the private key associated with the certificate.

You can use OpenSSL to decrypt your data export files. You can download the software by following the instructions here. If you are working with Windows, the OpenSSL binaries can be downloaded here.

To decrypt a downloaded data file, use the following OpenSSL command:

```
openssl smime -decrypt -binary -inkey <private_key_file>.key -in <encrypted_file>.zip.smime
-out <output_file>.zip
```

where:
<encrypted_file> is the file to be decrypted
<private_key_file> is the private key you used to initially configure file encryption, so that the exported files can be decrypted
<output_file> is the file that would be written after decryption

After the file is decrypted, you can extract the contents of the zip file. Typically, the extracted files and folders are arranged in a hierarchical structure.

For more information about the formats and schemas used for data export files, see the Contact History Export Reference page.

If you want to decrypt your data files using a Java program, you can leverage code from this repository.

Contact History Export Reference

> **Important**
> This content may not be the latest Genesys Engage cloud content. To find the latest content, go to Genesys Engage cloud for Administrators.

This page contains information about how the data files for Contact History export jobs are formatted and structured.

The exported data files are formatted in JSON (JavaScript Object Notation) Lines format. In this type of format, each line in the file corresponds to one interaction or contacts attribute, represented as a JSON object with keys and values.

For example, an entry in a contact export might look like this:

```
{"ContactId":"00001a57JGQ00A8TZSA","tenantid":101,"createddate":"2015-08-05 08:31:20.78"}
```

The way that the data fields and values are organized in each export file is controlled by a *schema*. The schemas used by CDDS are based on **JSON Schema** (draft 7). You can learn more about JSON Schema by visiting their website.

The following sections provide the schemas used for each type of export job and include some examples of exported data.

> **Important**
> The examples provided for each export file have been formatted for easier viewing. In an actual export file, the data is displayed as a single-line, with each key-value pair separated by a comma.

Interactions

Interaction data contains details about interactions that were processed during the time period specified in the job settings, such as voice calls, chats, and emails. When CDDS exports this data, all of the data records for each interaction type are merged into a single export file:

<EntityType>-<jobId>.json

The exported interaction data includes several details about the interaction, such as when it took

place, the contact ID, disposition code, and any notes that the agent might have made during the interaction. You can see an example of exported interaction data here.

In addition to data values, interaction data may also include attached files (such as attachments in email interactions). The contents of these file attachments are exported in a separate file that uses a document schema:

<EntityType>-Attachment-*<jobId>*.json

You can associate interactions with their file attachments by cross-referencing the **id** field in the interaction export with the **entityid** field in the related document export file.

> **Important**
>
> Schema files are created only if there is corresponding data being exported. For example, if there are no documents being exported, no document schema is generated.

Interaction schema

[+] Click to show


```
{
  "definitions": {},
  "$schema": "http://json-schema.org/draft-07/schema#",
  "$id": "http://genesys.com/gdps/schema/9.0.001.07/interaction.json",
  "type": "object",
  "title": "Interaction schema",
  "required": [
    "IXNID"
  ],
  "properties": {
    "IXNID": {
      "$id": "#/properties/IXNID",
      "type": "string",
      "title": "The IXNID Schema",
      "default": "16 characters",
      "examples": [
        {
          "example of interaction Id": "17PKMJG4K82TRGG1"
        },
        {
          "Length": "16 characters"
        }
      ]
    },
    "STATUS": {
      "$id": "#/properties/STATUS",
      "type": "integer",
      "title": "The STATUS Schema",
      "default": "null",
      "examples": [
        {
          "values": "0: New, 1: Pending, 2: InProcess, 3: Stopped"
```

```
        },
        {
          "definition": "Interaction state available through a set of enums"
        }
      ]
    },
    "ENTITYTYPEID": {
      "$id": "#/properties/ENTITYTYPEID",
      "type": "integer",
      "title": "The ENTITYTYPEID Schema",
      "default": "null",
      "examples": [
        {
          "values": "0: EmailIn, 1: EmailOut, 2: Chat, 3: PhoneCall, 5: Callback, 6:
CoBrowse, 7: Interaction"
        },
        {
          "definition": "Interaction type available through a set of enums"
        }
      ]
    },
    "MEDIATYPEID": {
      "$id": "#/properties/MEDIATYPEID",
      "type": "string",
      "title": "The MEDIATYPEID Schema",
      "default": "",
      "examples": []
    },
    "TYPEID": {
      "$id": "#/properties/TYPEID",
      "type": "string",
      "title": "The TYPEID Schema",
      "default": "",
      "examples": []
    },
    "SUBTYPEID": {
      "$id": "#/properties/SUBTYPEID",
      "type": "string",
      "title": "The SUBTYPEID Schema",
      "default": "",
      "examples": []
    },
    "EXTERNALID": {
      "$id": "#/properties/EXTERNALID",
      "type": "string",
      "title": "The EXTERNALID Schema",
      "default": "",
      "examples": []
    },
    "OWNERID": {
      "$id": "#/properties/OWNERID",
      "type": "integer",
      "title": "The OWNERID Schema",
      "default": "",
      "examples": []
    },
    "CONTACTID": {
      "$id": "#/properties/CONTACTID",
      "type": "string",
      "title": "The CONTACTID Schema",
      "default": "",
      "examples": []
    },
```

```
"PARENTID": {
  "$id": "#/properties/PARENTID",
  "type": "string",
  "title": "The PARENTID Schema",
  "default": "",
  "examples": []
},
"STARTDATE": {
  "$id": "#/properties/STARTDATE",
  "type": "string",
  "title": "The STARTDATE Schema",
  "default": "Date and time of creation",
  "examples": [
    {
      "StartDate": "2010-04-30T12:03:30.133Z"
    },
    {
      "definition": "Date and time in ISO 8601 standard when interaction starts"
    }
  ]
},
"MODIFIEDDATE": {
  "$id": "#/properties/MODIFIEDDATE",
  "type": "string",
  "title": "The MODIFIEDDATE Schema",
  "default": "null",
  "examples": [
    {
      "ModifiedDate": "2010-04-30T12:03:30.133Z"
    },
    {
      "definition": "Date and time in ISO 8601 standard when interaction is updated"
    }
  ]
},
"ENDDATE": {
  "$id": "#/properties/ENDDATE",
  "type": "string",
  "title": "The ENDDATE Schema",
  "default": "null",
  "examples": [
    {
      "EndDate": "2010-04-30T12:03:30.133Z"
    },
    {
      "definition": "Date and time in ISO 8601 when interaction is stopped"
    }
  ]
},
"THREADID": {
  "$id": "#/properties/THREADID",
  "type": "string",
  "title": "The THREADID Schema",
  "default": "",
  "examples": []
},
"CATEGORYID": {
  "$id": "#/properties/CATEGORYID",
  "type": "string",
  "title": "The CATEGORYID Schema",
  "default": "",
  "examples": []
},
```

```
"TIMESHIFT": {
  "$id": "#/properties/TIMESHIFT",
  "type": "integer",
  "title": "The TIMESHIFT Schema",
  "default": "",
  "examples": []
},
"SUBJECT": {
  "$id": "#/properties/SUBJECT",
  "type": "string",
  "title": "The SUBJECT Schema",
  "default": "",
  "examples": []
},
"TEXT": {
  "$id": "#/properties/TEXT",
  "type": "string",
  "title": "The TEXT Schema",
  "default": "",
  "examples": []
},
"STRUCTUREDTEXT": {
  "$id": "#/properties/STRUCTUREDTEXT",
  "type": "string",
  "title": "The STRUCTUREDTEXT Schema",
  "default": "",
  "examples": []
},
"STRUCTTEXTMIMETYPE": {
  "$id": "#/properties/STRUCTTEXTMIMETYPE",
  "type": "string",
  "title": "The STRUCTTEXTMIMETYPE Schema",
  "default": "",
  "examples": []
},
"THECOMMENT": {
  "$id": "#/properties/THECOMMENT",
  "type": "string",
  "title": "The THECOMMENT Schema",
  "default": "",
  "examples": []
},
"TENANTID": {
  "$id": "#/properties/TENANTID",
  "type": "integer",
  "title": "The TENANTID Schema",
  "default": "",
  "examples": []
},
"SUBTENANTID": {
  "$id": "#/properties/SUBTENANTID",
  "type": "integer",
  "title": "The SUBTENANTID Schema",
  "default": "",
  "examples": []
},
"THREADHASH": {
  "$id": "#/properties/THREADHASH",
  "type": "integer",
  "title": "The THREADHASH Schema",
  "default": "",
  "examples": []
},
```

```
    "CANBEPARENT": {
      "$id": "#/properties/CANBEPARENT",
      "type": "integer",
      "title": "The CANBEPARENT Schema",
      "default": "",
      "examples": []
    },
    "CREATORAPPID": {
      "$id": "#/properties/CREATORAPPID",
      "type": "integer",
      "title": "The CREATORAPPID Schema",
      "default": "",
      "examples": []
    },
    "QUEUENAME": {
      "$id": "#/properties/QUEUENAME",
      "type": "string",
      "title": "The QUEUENAME Schema",
      "default": "",
      "examples": []
    },
    "ALLATTRIBUTES": {
      "$id": "#/properties/ALLATTRIBUTES",
      "type": "object",
      "title": "The ALLATTRIBUTES Schema",
      "default": "null",
      "examples": [
        {
          "example of AllAttributes data":
"\"AllAttributes\":{\"LastName\":\"WALLACE\",\"RStrategyDBID\":\"252\",\"RVQDBID\":\",\"ServiceObjective\":0,\"
Scheduled\"}}\""
        },
        {
          "definition": "Userdata key/value pairs"
        },
        {
          "format": "{\"<key>\": <data>} where key is a string and data either a string, an
integer or another valid json object. Format allows nested structures."
        }
      ]
    },
    "STRATTRIBUTE1": {
      "$id": "#/properties/STRATTRIBUTE1",
      "type": "string",
      "title": "The STRATTRIBUTE1 Schema",
      "default": "",
      "examples": []
    },
    "STRATTRIBUTE2": {
      "$id": "#/properties/STRATTRIBUTE2",
      "type": "string",
      "title": "The STRATTRIBUTE2 Schema",
      "default": "",
      "examples": []
    },
    "STRATTRIBUTE3": {
      "$id": "#/properties/STRATTRIBUTE3",
      "type": "string",
      "title": "The STRATTRIBUTE3 Schema",
      "default": "",
      "examples": []
    },
    "STRATTRIBUTE4": {
```

```
    "$id": "#/properties/STRATTRIBUTE4",
    "type": "string",
    "title": "The STRATTRIBUTE4 Schema",
    "default": "",
    "examples": []
  },
  "STRATTRIBUTE5": {
    "$id": "#/properties/STRATTRIBUTE5",
    "type": "string",
    "title": "The STRATTRIBUTE5 Schema",
    "default": "",
    "examples": []
  },
  "STRATTRIBUTE6": {
    "$id": "#/properties/STRATTRIBUTE6",
    "type": "string",
    "title": "The STRATTRIBUTE6 Schema",
    "default": "",
    "examples": []
  },
  "STRATTRIBUTE7": {
    "$id": "#/properties/STRATTRIBUTE7",
    "type": "string",
    "title": "The STRATTRIBUTE7 Schema",
    "default": "",
    "examples": []
  },
  "STRATTRIBUTE8": {
    "$id": "#/properties/STRATTRIBUTE8",
    "type": "string",
    "title": "The STRATTRIBUTE8 Schema",
    "default": "",
    "examples": []
  },
  "STRATTRIBUTE9": {
    "$id": "#/properties/STRATTRIBUTE9",
    "type": "string",
    "title": "The STRATTRIBUTE9 Schema",
    "default": "",
    "examples": []
  },
  "STRATTRIBUTE10": {
    "$id": "#/properties/STRATTRIBUTE10",
    "type": "string",
    "title": "The STRATTRIBUTE10 Schema",
    "default": "",
    "examples": []
  },
  "INTATTRIBUTE1": {
    "$id": "#/properties/INTATTRIBUTE1",
    "type": "integer",
    "title": "The INTATTRIBUTE1 Schema",
    "default": "",
    "examples": []
  },
  "INTATTRIBUTE2": {
    "$id": "#/properties/INTATTRIBUTE2",
    "type": "integer",
    "title": "The INTATTRIBUTE2 Schema",
    "default": "",
    "examples": []
  },
  "INTATTRIBUTE3": {
```

```
    "$id": "#/properties/INTATTRIBUTE3",
    "type": "integer",
    "title": "The INTATTRIBUTE3 Schema",
    "default": "",
    "examples": []
  },
  "INTATTRIBUTE4": {
    "$id": "#/properties/INTATTRIBUTE4",
    "type": "integer",
    "title": "The INTATTRIBUTE4 Schema",
    "default": "",
    "examples": []
  },
  "INTATTRIBUTE5": {
    "$id": "#/properties/INTATTRIBUTE5",
    "type": "integer",
    "title": "The INTATTRIBUTE5 Schema",
    "default": "",
    "examples": []
  },
  "ISSPAM": {
    "$id": "#/properties/ISSPAM",
    "type": "integer",
    "title": "The ISSPAM Schema",
    "default": "",
    "examples": []
  },
  "WEBSAFEEMAILSTATUS": {
    "$id": "#/properties/WEBSAFEEMAILSTATUS",
    "type": "string",
    "title": "The WEBSAFEEMAILSTATUS Schema",
    "default": "",
    "examples": []
  },
  "ISCATEGORYAPPROVED": {
    "$id": "#/properties/ISCATEGORYAPPROVED",
    "type": "integer",
    "title": "The ISCATEGORYAPPROVED Schema",
    "default": "",
    "examples": []
  },
  "STOPPEDREASON": {
    "$id": "#/properties/STOPPEDREASON",
    "type": "string",
    "title": "The STOPPEDREASON Schema",
    "default": "",
    "examples": []
  },
  "LANG": {
    "$id": "#/properties/LANG",
    "type": "string",
    "title": "The LANG Schema",
    "default": "",
    "examples": []
  },
  "FROMADDRESS": {
    "$id": "#/properties/FROMADDRESS",
    "type": "string",
    "title": "The FROMADDRESS Schema",
    "default": "",
    "examples": []
  },
  "FROMPERSONAL": {
```

```
    "$id": "#/properties/FROMPERSONAL",
    "type": "string",
    "title": "The FROMPERSONAL Schema",
    "default": "",
    "examples": []
  },
  "REPLYTOADDRESS": {
    "$id": "#/properties/REPLYTOADDRESS",
    "type": "string",
    "title": "The REPLYTOADDRESS Schema",
    "default": "",
    "examples": []
  },
  "TOADDRESSES": {
    "$id": "#/properties/TOADDRESSES",
    "type": "string",
    "title": "The TOADDRESSES Schema",
    "default": "",
    "examples": []
  },
  "CCADDRESSES": {
    "$id": "#/properties/CCADDRESSES",
    "type": "string",
    "title": "The CCADDRESSES Schema",
    "default": "",
    "examples": []
  },
  "BCCADDRESSES": {
    "$id": "#/properties/BCCADDRESSES",
    "type": "string",
    "title": "The BCCADDRESSES Schema",
    "default": "",
    "examples": []
  },
  "SENTDATE": {
    "$id": "#/properties/SENTDATE",
    "type": "string",
    "title": "The SENTDATE Schema",
    "default": "null",
    "examples": [
      {
        "SentDate": "2010-04-30T12:03:30.133Z"
      },
      {
        "definition": "Date and time in ISO 8601 standard"
      }
    ]
  },
  "MAILBOX": {
    "$id": "#/properties/MAILBOX",
    "type": "string",
    "title": "The MAILBOX Schema",
    "default": "",
    "examples": []
  },
  "WHICHRULEMATCHED": {
    "$id": "#/properties/WHICHRULEMATCHED",
    "type": "string",
    "title": "The WHICHRULEMATCHED Schema",
    "default": "",
    "examples": []
  },
  "EMAILOUTID": {
```

```
    "$id": "#/properties/EMAILOUTID",
    "type": "string",
    "title": "The EMAILOUTID Schema",
    "default": "",
    "examples": []
  },
  "HEADER": {
    "$id": "#/properties/HEADER",
    "type": "string",
    "title": "The HEADER Schema",
    "default": "null",
    "examples": [
      {
        "definition": "Encoded mail header in Binary64 format"
      }
    ]
  },
  "FromAddressEmailOut": {
    "$id": "#/properties/FromAddressEmailOut",
    "type": "string",
    "title": "The FromAddressEmailOut Schema",
    "default": "",
    "examples": []
  },
  "FromPersonalEmailOut": {
    "$id": "#/properties/FromPersonalEmailOut",
    "type": "string",
    "title": "The FromPersonalEmailOut Schema",
    "default": "",
    "examples": []
  },
  "ReplyToAddressEmailOut": {
    "$id": "#/properties/ReplyToAddressEmailOut",
    "type": "string",
    "title": "The ReplyToAddressEmailOut Schema",
    "default": "",
    "examples": []
  },
  "ToAddressesEmailOut": {
    "$id": "#/properties/ToAddressesEmailOut",
    "type": "string",
    "title": "The ToAddressesEmailOut Schema",
    "default": "",
    "examples": []
  },
  "CcAddressesEmailOut": {
    "$id": "#/properties/CcAddressesEmailOut",
    "type": "string",
    "title": "The CcAddressesEmailOut Schema",
    "default": "",
    "examples": []
  },
  "BccAddressesEmailOut": {
    "$id": "#/properties/BccAddressesEmailOut",
    "type": "string",
    "title": "The BccAddressesEmailOut Schema",
    "default": "",
    "examples": []
  },
  "SentDateEmailOut": {
    "$id": "#/properties/SentDateEmailOut",
    "type": "string",
    "title": "The SentDateEmailOut Schema",
```

```
    "default": "null",
    "examples": [
      {
        "SentDateEmailOut": "2010-04-30T12:03:30.133Z"
      },
      {
        "definition": "Date and time in ISO 8601 standard"
      }
    ]
  },
  "FORWARD": {
    "$id": "#/properties/FORWARD",
    "type": "integer",
    "title": "The FORWARD Schema",
    "default": "",
    "examples": []
  },
  "REFERENCEID": {
    "$id": "#/properties/REFERENCEID",
    "type": "string",
    "title": "The REFERENCEID Schema",
    "default": "",
    "examples": []
  },
  "REVIEWERID": {
    "$id": "#/properties/REVIEWERID",
    "type": "integer",
    "title": "The REVIEWERID Schema",
    "default": "",
    "examples": []
  },
  "STANDARDRESPONSEID": {
    "$id": "#/properties/STANDARDRESPONSEID",
    "type": "string",
    "title": "The STANDARDRESPONSEID Schema",
    "default": "",
    "examples": []
  },
  "ESTABLISHEDDATE": {
    "$id": "#/properties/ESTABLISHEDDATE",
    "type": "string",
    "title": "The ESTABLISHEDDATE Schema",
    "default": "null",
    "examples": [
      {
        "EstablishedDate": "2010-04-30T12:03:30.133Z"
      },
      {
        "definition": "Date and time in ISO 8601 standard"
      }
    ]
  },
  "RELEASEDDATE": {
    "$id": "#/properties/RELEASEDDATE",
    "type": "string",
    "title": "The RELEASEDDATE Schema",
    "default": "null",
    "examples": [
      {
        "definition": "Date and time in ISO 8601 standard"
      },
      {
        "ReleasedDate": "2010-04-30T12:03:30.133Z"
```

```
      }
    ]
  },
  "DURATION": {
    "$id": "#/properties/DURATION",
    "type": "integer",
    "title": "The DURATION Schema",
    "default": "",
    "examples": []
  },
  "OUTCOME": {
    "$id": "#/properties/OUTCOME",
    "type": "string",
    "title": "The OUTCOME Schema",
    "default": "",
    "examples": []
  },
  "PHONENUMBER": {
    "$id": "#/properties/PHONENUMBER",
    "type": "string",
    "title": "The PHONENUMBER Schema",
    "default": "",
    "examples": []
  },
  "TCONNECTIONID": {
    "$id": "#/properties/TCONNECTIONID",
    "type": "string",
    "title": "The TCONNECTIONID Schema",
    "default": "",
    "examples": []
  },
  "CALLBACKSTATUS": {
    "$id": "#/properties/CALLBACKSTATUS",
    "type": "integer",
    "title": "The CALLBACKSTATUS Schema",
    "default": "",
    "examples": []
  },
  "DETAILEDDESCRIPTION": {
    "$id": "#/properties/DETAILEDDESCRIPTION",
    "type": "string",
    "title": "The DETAILEDDESCRIPTION Schema",
    "default": "",
    "examples": []
  },
  "CUSTOMDATA": {
    "$id": "#/properties/CUSTOMDATA",
    "type": "string",
    "title": "The CUSTOMDATA Schema",
    "default": "",
    "examples": []
  },
  "DESIREDRESPONSETYPE": {
    "$id": "#/properties/DESIREDRESPONSETYPE",
    "type": "integer",
    "title": "The DESIREDRESPONSETYPE Schema",
    "default": "",
    "examples": []
  },
  "STARTTIME": {
    "$id": "#/properties/STARTTIME",
    "type": "string",
    "title": "The STARTTIME Schema",
```

```
    "default": "null",
    "examples": [
      {
        "StartTime": "2010-04-30T12:03:30.133Z"
      },
      {
        "definition": "Date and time in ISO 8601 standard"
      }
    ]
  },
  "ENDTIME": {
    "$id": "#/properties/ENDTIME",
    "type": "string",
    "title": "The ENDTIME Schema",
    "default": "null",
    "examples": [
      {
        "EndTime": "2010-04-30T12:03:30.133Z"
      },
      {
        "definition": "Date and time in ISO 8601 standard"
      }
    ]
  },
  "CUSTOMERNUMBER": {
    "$id": "#/properties/CUSTOMERNUMBER",
    "type": "string",
    "title": "The CUSTOMERNUMBER Schema",
    "default": "",
    "examples": []
  },
  "ATTEMPTS": {
    "$id": "#/properties/ATTEMPTS",
    "type": "integer",
    "title": "The ATTEMPTS Schema",
    "default": "",
    "examples": []
  },
  "DN": {
    "$id": "#/properties/DN",
    "type": "string",
    "title": "The DN Schema",
    "default": "",
    "examples": []
  },
  "LOCATION": {
    "$id": "#/properties/LOCATION",
    "type": "string",
    "title": "The LOCATION Schema",
    "default": "",
    "examples": []
  },
  "CALLBACKSERVERID": {
    "$id": "#/properties/CALLBACKSERVERID",
    "type": "string",
    "title": "The CALLBACKSERVERID Schema",
    "default": "",
    "examples": []
  },
  "CALLRESULT": {
    "$id": "#/properties/CALLRESULT",
    "type": "integer",
    "title": "The CALLRESULT Schema",
```

```
      "default": "",
      "examples": []
    },
    "THETYPE": {
      "$id": "#/properties/THETYPE",
      "type": "integer",
      "title": "The THETYPE Schema",
      "default": "",
      "examples": []
    },
    "MIMETYPE": {
      "$id": "#/properties/MIMETYPE",
      "type": "string",
      "title": "The MIMETYPE Schema",
      "default": "",
      "examples": []
    },
    "THESIZE": {
      "$id": "#/properties/THESIZE",
      "type": "integer",
      "title": "The THESIZE Schema",
      "default": "",
      "examples": []
    },
    "CONTENT": {
      "$id": "#/properties/CONTENT",
      "type": "string",
      "title": "The CONTENT Schema",
      "default": "null",
      "examples": [
        {
          "definition": "Encoded mail content in Binary64 format"
        }
      ]
    },
    "SUBSCRIBERID": {
      "$id": "#/properties/SUBSCRIBERID",
      "type": "string",
      "title": "The SUBSCRIBERID Schema",
      "default": "null",
      "examples": [
        {
          "definition": "Name of the business line, when defined"
        }
      ]
    }
  }
}
```

Example of an Interaction export

[+] Click to show

```
{
  "STARTDATE": "2012-03-30T10:29:50.407Z",
  "MIMETYPE": "message/rfc822",
  "HEADER":
"UmV0dXJuLVBhdGg6IENsaWVudDFAY3VzdC5kZXYNClJlY2VpdmVkOiBmcm9tIGJzZ2VueHA0Y2lpdyAoWzE3Mi4yNS4xNTcuNzddKQ0KCWJ5IG
  "ENTITYTYPEID": 0,
  "CONTENT":
"UmV0dXJuLVBhdGg6IENsaWVudDFAY3VzdC5kZXYNClJlY2VpdmVkOiBmcm9tIGJzZ2VueHA0Y2lpdyAoWzE3Mi4yNS4xNTcuNzddKQ0KCWJ5IG
  "SUBTYPEID": "InboundNew",
```

```
"CONTACTID": "0000Xa6VJTWV000M",
"OWNERID": 126,
"MAILBOX": "\"Contact Center\" <main@mcr.dev>",
"CREATORAPPID": 171,
"MEDIATYPEID": "email",
"STATUS": 3,
"TENANTID": 101,
"IXNID": "0002Ha7QB2MX000R",
"FROMADDRESS": "Client1@cust.dev",
"CANBEPARENT": 1,
"TEXT": " 0 Test memory leak ",
"TOADDRESSES": "main@mcr.dev",
"ENDDATE": "2012-03-30T10:29:32.957Z",
"THREADHASH": 20772491,
"EXTERNALID": "<15655788.1333103294550.JavaMail.genesys@bsgen810ciiw>",
"ALLATTRIBUTES": {
  "CBR-actual_volume": "",
  "Ctg_0000Ra6J43AQ00AQ": "60",
  "RStrategyDBID": "252",
  "RVQDBID": "",
  "_ackDone": "1",
  "ServiceObjective": 0,
  "_ContainsAttachment": "false",
  "RTargetAgentSelected": "1004",
  "Ctg_0002Fa13WJWS0AU0": "50",
  "RVQID": "",
  "CustomerSegment": "default",
  "CBR-IT-path_DBIDs": "",
  "_AutoReplyCount": 0,
  "Subject": "Automatic tests via Sikuli - Check memory leak",
  "Mailbox": "\"Contact Center\" <main@mcr.dev>",
  "CtgId": "0002Fa13WJWS0AUH",
  "FirstName": "Client1",
  "ContactId": "0000Xa6VJTWV000M",
  "CBR-Interaction_cost": "",
  "FromPersonal": "",
  "RStrategyName": "email-routing-inbound",
  "RTargetRequested": "Agent Group 80001",
  "RTargetObjSelDBID": "116",
  "RTargetAgentGroup": "Agent Group 80001",
  "IW_CaseUid": "bd04cb63-3eb8-42b1-8dc8-893271de62bf",
  "IW_BundleUid": "497f696f-159a-4682-a8cd-30f78bc1b147",
  "RRequestedSkillCombination": "",
  "RTargetRuleSelected": "",
  "RTargetObjectSelected": "Agent Group 80001",
  "Header_Content-Type": "multipart/mixed; boundary=\"----=_Part_0_8026323.1333103294535\"",
  "RTargetTypeSelected": "2",
  "FromAddress": "Client1@cust.dev",
  "Header_Date": "Fri, 30 Mar 2012 12:28:14 +0200 (CEST)",
  "Header_Mime-Version": "1.0",
  "Header_Message-ID": "<15655788.1333103294550.JavaMail.genesys@bsgen810ciiw>",
  "RTenant": "defaultTenant",
  "ServiceType": "default",
  "To": "main@mcr.dev",
  "Origination_Source": "Email",
  "_AttachmentsSize": "0",
  "IWAttachedDataInformation": {
    "DispositionCodeValues.DC_Accepted": "Accepted",
    "CaseDataOrder": "",
    "CaseData.URL1": "URL1",
    "CaseData.Segment": "Segment",
    "CaseData.CallType": "Call Type",
    "DispositionCode.Key": "DispositionCode",
```

```
      "DispositionCodeValues.DC_Transferred": "Transferred",
      "Option.interaction.case-data.header-foreground-color": "#FF15428B",
      "Option.expression.url": "(?#Protocol)(?:(?:ht|f)tp(?:s?)\\:\\/\\/|~/|
/)?(?#Username:Password)(?:\\w+:\\w+@)?(?#Subdomains)(?:(?:[-\\w]+\\.)+(?#TopLevel
Domains)(?:com|org|net|gov|mil|biz|info|mobi|name|aero|jobs|museum|travel|[a-
z]{2}))(?#Port)(?::[\\d]{1,5})?(?#Directories)(?:(?:(?:/(?:[-\\w~!$+|.,=]|%[a-f\\d]{2})+)+|
/)+|\\?|#)?(?#Query)(?:(?:\\?(?:[-\\w~!$+|.,*:]|%[a-f\\d{2}])+=(?:[-\\w~!$+|.,*:=]|%[a-
f\\d]{2})*)(?:&(?:[-\\w~!$+|.,*:]|%[a-f\\d{2}])+=(?:[-\\w~!$+|.,*:=]|%[a-
f\\d]{2})*)*)*(?#Anchor)(?:#(?:[-\\w~!$+|.,*:=]|%[a-f\\d]{2})*)?",
      "CaseData.URL2": "URL2",
      "Option.interaction.case-data.frame-color": "#FFFFBA00",
      "DispositionCode.Label": "Disposition Code",
      "DispositionCodeValues.DC_Rejected": "Rejected",
      "CaseData.Priority": "Priority",
      "CaseData.Reason": "Reason"
    },
    "PegAGAgent Group 80001": 1,
    "RRequestedSkills": {},
    "RTargetPlaceSelected": "Place_1004",
    "Ctg_0000Ra6J43AQ00A4": "12",
    "CBR-contract_DBIDs": "",
    "CtgRelevancy": "84",
    "EmailAddress": "Client1@cust.dev"
  },
  "TIMESHIFT": 120,
  "SENTDATE": "2012-03-30T10:28:14.000Z",
  "THESIZE": 622,
  "SUBJECT": "Automatic tests via Sikuli - Check memory leak",
  "THREADID": "0002Ha7QB2MX000S",
  "TYPEID": "Inbound"
}
```


Document schema

[+] Click to show


```
{
  "definitions": {},
  "$schema": "http://json-schema.org/draft-07/schema#",
  "$id": "http://genesys.com/gdps/schema/9.0.001.07/document.json",
  "type": "object",
  "title": "Document schema",
  "required": [],
  "properties": {
    "ID": {
      "$id": "#/properties/ID",
      "type": "string",
      "title": "The ID Schema",
      "default": "",
      "examples": []
    },
    "MIMETYPE": {
      "$id": "#/properties/MIMETYPE",
      "type": "string",
      "title": "The MIMETYPE Schema",
      "default": "",
      "examples": []
    },
```

```
    "THENAME": {
      "$id": "#/properties/THENAME",
      "type": "string",
      "title": "The THENAME Schema",
      "default": "",
      "examples": []
    },
    "DESCRIPTION": {
      "$id": "#/properties/DESCRIPTION",
      "type": "string",
      "title": "The DESCRIPTION Schema",
      "default": "",
      "examples": []
    },
    "THESIZE": {
      "$id": "#/properties/THESIZE",
      "type": "integer",
      "title": "The THESIZE Schema",
      "default": "",
      "examples": []
    },
    "CONTENT": {
      "$id": "#/properties/CONTENT",
      "type": "string",
      "title": "The CONTENT Schema",
      "default": "null",
      "examples": [
        {
          "definition": "Encoded mail content in Binary64 format"
        }
      ]
    },
    "DOCUMENTID": {
      "$id": "#/properties/DOCUMENTID",
      "type": "string",
      "title": "The DOCUMENTID Schema",
      "default": "",
      "examples": []
    },
    "ENTITYID": {
      "$id": "#/properties/ENTITYID",
      "type": "string",
      "title": "The ENTITYID Schema",
      "default": "",
      "examples": []
    },
    "ENTITYTYPEID": {
      "$id": "#/properties/ENTITYTYPEID",
      "type": "integer",
      "title": "The ENTITYTYPEID Schema",
      "default": "null",
      "examples": [
        {
          "values": "0: EmailIn, 1: EmailOut, 2: Chat, 3: PhoneCall, 5: Callback, 6:
CoBrowse, 7: Interaction"
        },
        {
          "definition": "Interaction type available through a set of enums"
        }
      ]
    }
  }
}
```

Example of a Document export

[+] Click to show


```
{
  "ID": "0003Ra7VSTDP03H0",
  "MIMETYPE": "message/rfc822",
  "DESCRIPTION": "",
  "ENTITYTYPEID": 0,
  "CONTENT":
"UmV0dXJuLVBhdGg6IENsaWVudDFAY3VzdC5kZXYNClJlY2VpdmVkOiBmcm9tIGJzZ2VueHA0Y2lpdyAoWzE3Mi4yNS4xNTcuNzddKQ0KCWJ5IG
  "THENAME": "OriginalMsg.eml",
  "DOCUMENTID": "0003Ra7VSTDP03H0",
  "ENTITYID": "0003Ra7VSTDP03FU",
  "THESIZE": 628
}
```


AllAttributes field format

The **AllAttributes** field is part of the interactions export data and contains a special collection of key-value pairs. The field holds a list of objects in the following format, where **key** is a string identifier and **data** is a JSON type (such as string, integer, or other valid JSON object).

```
{ "<key>" : <data> }
```

Example of AllAttributes export

[+] Click to show


```
"ALLATTRIBUTES" : {
  "CBR-actual_volume": "",
  "Ctg_0000Ra6J43AQ00AQ": "60",
  "RStrategyDBID": "252",
  "RVQDBID": "",
  "_ackDone": "1",
  "ServiceObjective": 0,
  "_ContainsAttachment": "false",
  "RTargetAgentSelected": "1004",
  "Ctg_0002Fa13WJWS0AU0": "50",
  "RVQID": "",
  "CustomerSegment": "default",
  "CBR-IT-path_DBIDs": "",
  "_AutoReplyCount": 0,
  "Subject": "Automatic tests via Sikuli - Check memory leak",
  "Mailbox": "\"Contact Center\" <main@mcr.dev>",
  "CtgId": "0002Fa13WJWS0AUH",
  "FirstName": "Client1",
  "ContactId": "0000Xa6VJTWV000M",
  "CBR-Interaction_cost": "",
  "FromPersonal": "",
  "RStrategyName": "email-routing-inbound",
  "RTargetRequested": "Agent Group 80001",
  "RTargetObjSelDBID": "116",
  "RTargetAgentGroup": "Agent Group 80001",
  "IW_CaseUid": "bd04cb63-3eb8-42b1-8dc8-893271de62bf",
  "IW_BundleUid": "497f696f-159a-4682-a8cd-30f78bc1b147",
  "RRequestedSkillCombination": "",
  "RTargetRuleSelected": "",
  "RTargetObjectSelected": "Agent Group 80001",
```

```
  "Header_Content-Type": "multipart/mixed; boundary=\"----=_Part_0_8026323.1333103294535\"",
  "RTargetTypeSelected": "2",
  "FromAddress": "Client1@cust.dev",
  "Header_Date": "Fri, 30 Mar 2012 12:28:14 +0200 (CEST)",
  "Header_Mime-Version": "1.0",
  "Header_Message-ID": "<15655788.1333103294550.JavaMail.genesys@bsgen810ciiw>",
  "RTenant": "defaultTenant",
  "ServiceType": "default",
  "To": "main@mcr.dev",
  "Origination_Source": "Email",
  "_AttachmentsSize": "0",
  "IWAttachedDataInformation": {
    "DispositionCodeValues.DC_Accepted": "Accepted",
    "CaseDataOrder": "",
    "CaseData.URL1": "URL1",
    "CaseData.Segment": "Segment",
    "CaseData.CallType": "Call Type",
    "DispositionCode.Key": "DispositionCode",
    "DispositionCodeValues.DC_Transferred": "Transferred",
    "Option.interaction.case-data.header-foreground-color": "#FF15428B",
    "Option.expression.url": "(?#Protocol)(?:(?:ht|f)tp(?:s?)\\:\\/\\/|~/|
/)?(?#Username:Password)(?:\\w+:\\w+@)?(?#Subdomains)(?:(?:[-\\w]+\\.)+(?#TopLevel
Domains)(?:com|org|net|gov|mil|biz|info|mobi|name|aero|jobs|museum|travel|[a-
z]{2}))(?#Port)(?::[\\d]{1,5})?(?#Directories)(?:(?:(?:/(?:[-\\w~!$+|.,=]|%[a-f\\d]{2})+)+|
/)+|\\?|#)?(?#Query)(?:(?:\\?(?:[-\\w~!$+|.,*:]|%[a-f\\d{2}])+=(?:[-\\w~!$+|.,*:=]|%[a-
f\\d]{2})*)(?:&(?:[-\\w~!$+|.,*:]|%[a-f\\d{2}])+=(?:[-\\w~!$+|.,*:=]|%[a-
f\\d]{2})*)*)*(?#Anchor)(?:#(?:[-\\w~!$+|.,*:=]|%[a-f\\d]{2})*)?",
    "CaseData.URL2": "URL2",
    "Option.interaction.case-data.frame-color": "#FFFFBA00",
    "DispositionCode.Label": "Disposition Code",
    "DispositionCodeValues.DC_Rejected": "Rejected",
    "CaseData.Priority": "Priority",
    "CaseData.Reason": "Reason"
  },
  "PegAGAgent Group 80001": 1,
  "RRequestedSkills": {},
  "RTargetPlaceSelected": "Place_1004",
  "Ctg_0000Ra6J43AQ00A4": "12",
  "CBR-contract_DBIDs": "",
  "CtgRelevancy": "84",
  "EmailAddress": "Client1@cust.dev"
}
```


Contacts

Contact data export files contain details about each customer, such as their name, customer ID, email address, and the date that their contact information was created or modified.

Contact Attributes are associated with a particular contact and, like documents (i.e. attachments) for interactions, are exported in a separate file that uses a different schema.

Contact schema

[+] Click to show


```
{
  "definitions": {},
```

```
"$schema": "http://json-schema.org/draft-07/schema#",
"$id": "http://genesys.com/gdps/schema/9.0.001.07/contact.json",
"type": "object",
"title": "Contact schema",
"required": [
  "CONTACTID"
],
"properties": {
  "CONTACTID": {
    "$id": "#/properties/CONTACTID",
    "type": "string",
    "title": "The CONTACTID Schema",
    "default": "16 characters",
    "examples": [
      {
        "example of contactId": "18PKMJG3K82TRGG1"
      },
      {
        "Length": "16 characters"
      }
    ]
  },
  "TENANTID": {
    "$id": "#/properties/TENANTID",
    "type": "integer",
    "title": "The TENANTID Schema",
    "default": "",
    "examples": []
  },
  "SUBTENANTID": {
    "$id": "#/properties/SUBTENANTID",
    "type": "integer",
    "title": "The SUBTENANTID Schema",
    "default": "",
    "examples": []
  },
  "ISEXTERNALRESOURCE": {
    "$id": "#/properties/ISEXTERNALRESOURCE",
    "type": "integer",
    "title": "The ISEXTERNALRESOURCE Schema",
    "default": "",
    "examples": []
  },
  "STRATTRIBUTE1": {
    "$id": "#/properties/STRATTRIBUTE1",
    "type": "string",
    "title": "The STRATTRIBUTE1 Schema",
    "default": "",
    "examples": []
  },
  "STRATTRIBUTE2": {
    "$id": "#/properties/STRATTRIBUTE2",
    "type": "string",
    "title": "The STRATTRIBUTE2 Schema",
    "default": "",
    "examples": []
  },
  "STRATTRIBUTE3": {
    "$id": "#/properties/STRATTRIBUTE3",
    "type": "string",
    "title": "The STRATTRIBUTE3 Schema",
    "default": "",
    "examples": []
```

```
  },
  "STRATTRIBUTE4": {
    "$id": "#/properties/STRATTRIBUTE4",
    "type": "string",
    "title": "The STRATTRIBUTE4 Schema",
    "default": "",
    "examples": []
  },
  "STRATTRIBUTE5": {
    "$id": "#/properties/STRATTRIBUTE5",
    "type": "string",
    "title": "The STRATTRIBUTE5 Schema",
    "default": "",
    "examples": []
  },
  "STRATTRIBUTE6": {
    "$id": "#/properties/STRATTRIBUTE6",
    "type": "string",
    "title": "The STRATTRIBUTE6 Schema",
    "default": "",
    "examples": []
  },
  "STRATTRIBUTE7": {
    "$id": "#/properties/STRATTRIBUTE7",
    "type": "string",
    "title": "The STRATTRIBUTE7 Schema",
    "default": "",
    "examples": []
  },
  "STRATTRIBUTE8": {
    "$id": "#/properties/STRATTRIBUTE8",
    "type": "string",
    "title": "The STRATTRIBUTE8 Schema",
    "default": "",
    "examples": []
  },
  "STRATTRIBUTE9": {
    "$id": "#/properties/STRATTRIBUTE9",
    "type": "string",
    "title": "The STRATTRIBUTE9 Schema",
    "default": "",
    "examples": []
  },
  "STRATTRIBUTE10": {
    "$id": "#/properties/STRATTRIBUTE10",
    "type": "string",
    "title": "The STRATTRIBUTE10 Schema",
    "default": "",
    "examples": []
  },
  "DATEATTRIBUTE1": {
    "$id": "#/properties/DATEATTRIBUTE1",
    "type": "string",
    "format": "int64",
    "title": "The DATEATTRIBUTE1 Schema",
    "default": "",
    "examples": []
  },
  "DATEATTRIBUTE2": {
    "$id": "#/properties/DATEATTRIBUTE2",
    "type": "string",
    "format": "int64",
    "title": "The DATEATTRIBUTE2 Schema",
```

```
      "default": "",
      "examples": []
    },
    "DATEATTRIBUTE3": {
      "$id": "#/properties/DATEATTRIBUTE3",
      "type": "string",
      "format": "int64",
      "title": "The DATEATTRIBUTE3 Schema",
      "default": "",
      "examples": []
    },
    "CREATEDDATE": {
      "$id": "#/properties/CREATEDDATE",
      "type": "string",
      "title": "The CREATEDDATE Schema",
      "default": "null",
      "examples": [
        {
          "CreatedDate": "2010-04-30T12:03:30.133Z"
        },
        {
          "definition": "Date and time in ISO 8601 standard"
        }
      ]
    },
    "MODIFIEDDATE": {
      "$id": "#/properties/MODIFIEDDATE",
      "type": "string",
      "title": "The MODIFIEDDATE Schema",
      "default": "null",
      "examples": [
        {
          "ModifiedDate": "2010-04-30T12:03:30.133Z"
        },
        {
          "definition": "Date and time in ISO 8601 standard"
        }
      ]
    },
    "MERGEID": {
      "$id": "#/properties/MERGEID",
      "type": "string",
      "title": "The MERGEID Schema",
      "default": "",
      "examples": []
    },
    "SUBSCRIBERID": {
      "$id": "#/properties/SUBSCRIBERID",
      "type": "string",
      "title": "The SUBSCRIBERID Schema",
      "default": "",
      "examples": []
    }
  }
}
```


Example of a Contact export

[+] Click to show


```
{
  "CONTACTID": "00001a57JGQ00039",
```

```
  "STRATTRIBUTE4": "Joseph",
  "TENANTID": 101,
  "CREATEDDATE": "2009-08-05T08:30:25.467Z",
  "STRATTRIBUTE2": "A.Joseph@mycompany.com",
  "MODIFIEDDATE": "2009-08-05T08:30:25.513Z",
  "STRATTRIBUTE5": "A"
}
```


Contact Attributes schema

[+] Click to show


```
{
  "definitions": {},
  "$schema": "http://json-schema.org/draft-07/schema#",
  "$id": "http://genesys.com/gdps/schema/9.0.001.07/contactattribute.json",
  "type": "object",
  "title": "ContactAttribute schema",
  "required": [],
  "properties": {
    "ID": {
      "$id": "#/properties/ID",
      "type": "string",
      "title": "The ID Schema",
      "default": "",
      "examples": []
    },
    "CONTACTID": {
      "$id": "#/properties/CONTACTID",
      "type": "string",
      "title": "The CONTACTID Schema",
      "default": "",
      "examples": []
    },
    "ATTRIBUTEID": {
      "$id": "#/properties/ATTRIBUTEID",
      "type": "string",
      "title": "The ATTRIBUTEID Schema",
      "default": "",
      "examples": []
    },
    "ATTRIBUTENAME": {
      "$id": "#/properties/ATTRIBUTENAME",
      "type": "string",
      "title": "The ATTRIBUTENAME Schema",
      "default": "",
      "examples": []
    },
    "MIMETYPE": {
      "$id": "#/properties/MIMETYPE",
      "type": "string",
      "title": "The MIMETYPE Schema",
      "default": "",
      "examples": []
    },
    "STRVALUE": {
      "$id": "#/properties/STRVALUE",
      "type": "string",
      "title": "The STRVALUE Schema",
      "default": "",
      "examples": []
```

```
    },
    "STRVALUELOWERCASE": {
      "$id": "#/properties/STRVALUELOWERCASE",
      "type": "string",
      "title": "The STRVALUELOWERCASE Schema",
      "default": "",
      "examples": []
    },
    "BINVALUE": {
      "$id": "#/properties/BINVALUE",
      "type": "string",
      "title": "The BINVALUE Schema",
      "default": "null",
      "examples": [
        {
          "definition": "Binary content (png, jpg,...) encoded as base64"
        }
      ]
    },
    "DATEVALUE": {
      "$id": "#/properties/DATEVALUE",
      "type": "string",
      "title": "The DATEVALUE Schema",
      "default": "null",
      "examples": [
        {
          "DateValue": "2010-04-30T12:03:30.133Z"
        },
        {
          "definition": "Date and time in ISO 8601 standard"
        }
      ]
    },
    "DESCRIPTION": {
      "$id": "#/properties/DESCRIPTION",
      "type": "string",
      "title": "The DESCRIPTION Schema",
      "default": "",
      "examples": []
    },
    "ISPRIMARY": {
      "$id": "#/properties/ISPRIMARY",
      "type": "integer",
      "title": "The ISPRIMARY Schema",
      "default": "",
      "examples": []
    },
    "TENANTID": {
      "$id": "#/properties/TENANTID",
      "type": "integer",
      "title": "The TENANTID Schema",
      "default": "",
      "examples": []
    },
    "SUBTENANTID": {
      "$id": "#/properties/SUBTENANTID",
      "type": "integer",
      "title": "The SUBTENANTID Schema",
      "default": "",
      "examples": []
    }
  }
}
```

Example of a Contact Attributes export

[+] Click to show


```
{
  "ID": "00001a57JGQ0003A",
  "CONTACTID": "00001a57JGQ00039",
  "STRVALUELOWERCASE": "joseph",
  "ISPRIMARY": 1,
  "ATTRIBUTEID": "00001a57JGQ00008",
  "STRVALUE": "Joseph",
  "ATTRIBUTENAME": "LastName"
}
```