



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.

Agent Pacing Service Deployment Guide

Configuration

7/27/2024

Contents

- [1 About Configuration](#)
- [2 Pacing targets](#)
- [3 Virtual queue](#)
- [4 Agent groups](#)
 - [4.1 Configure a regular agent group](#)
 - [4.2 Configure a blended agent group](#)
- [5 Pacing service](#)

Configure your pacing targets and how the Agent Pacing Service works.

About Configuration

You must configure your pacing targets and your pacing service to make Agent Pacing Service work with your contact center.

Important

Pacing Service is a single-tenant application—it only has access to the tenant, objects, and transaction list of the tenant it is configured for. In a multi-tenant environment, you can configure a Pacing Service for each individual tenant.

Pacing targets

Each pacing target consists of:

- A media-specific virtual queue
- A set of one or more agent groups that are associated with this queue

Virtual queue

Although contact centers typically contain many virtual queues, only some of them are appropriate for monitoring by the pacing service. The pacing service only monitors virtual queues that contain:

- A **pacing** section in the options
- A **media** option in the **pacing** section. The value of this option must be the name of a single media type. You cannot specify a list of media types.

The **pacing** section can include the following additional options:

- **optimizationGoal**—Specifies the highest allowable percentage of proactively triggered interactions that can be closed by visitors prior to an agent joining the session.

- The value must be a float between **0** and **100**
- The default value is **3** The pacing service considers all suitable virtual queues, regardless of which Switch objects they are associated with.

Here is a sample virtual queue configuration that is accessible in Genesys Administrator Extension:

Name	Section	Key	Value
▼ pacing			
pacing \ optimizationGoal	pacing	optimizationGoal	3
pacing \ media	pacing	media	chat

Important

When configuring the virtual queue, you must set the value of **Alias** equal to the value of **Number**

Home > DNs > Switches > MultiMediaSwitch > DNs > Virtual Queue > Proactive Chat Properties

General

Number* Proactive Chat **Type*** Virtual Queue

Switch* MultiMediaSwitch

Association **Register*** True

Alias Proactive Chat **Route Type*** Default

DN Group

☒ Use Override **Override**

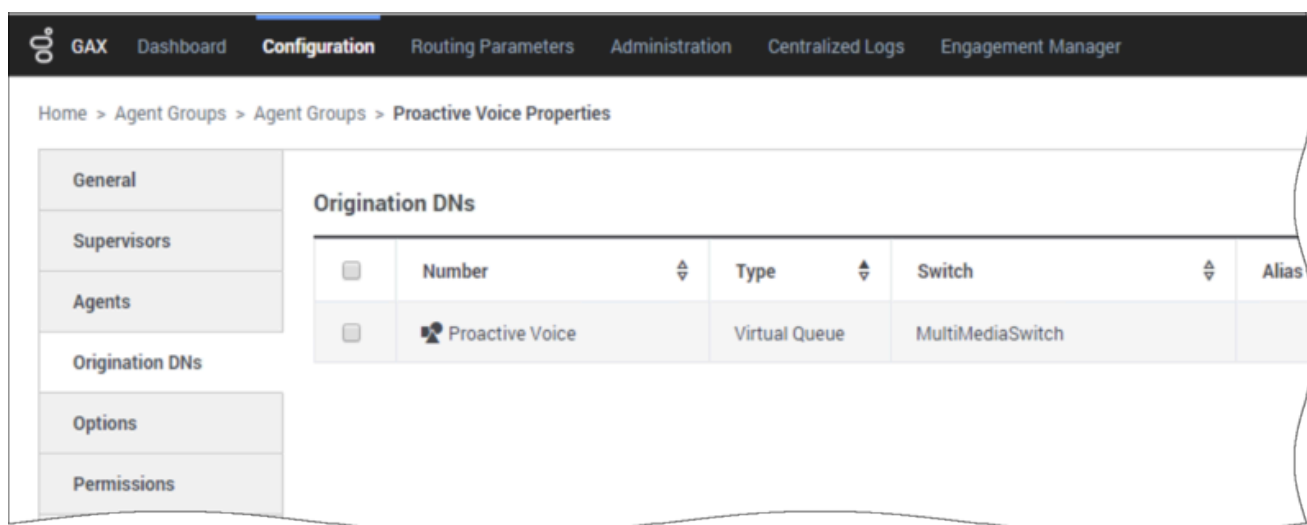
Agent groups

Agent Pacing Service only considers agent groups for processing as part of a pacing target if they include the relevant virtual queue in their list of Origination DNs. The media type specified in the virtual queue is used to limit the *Ready* agents in an agent group to those who can process that media type.

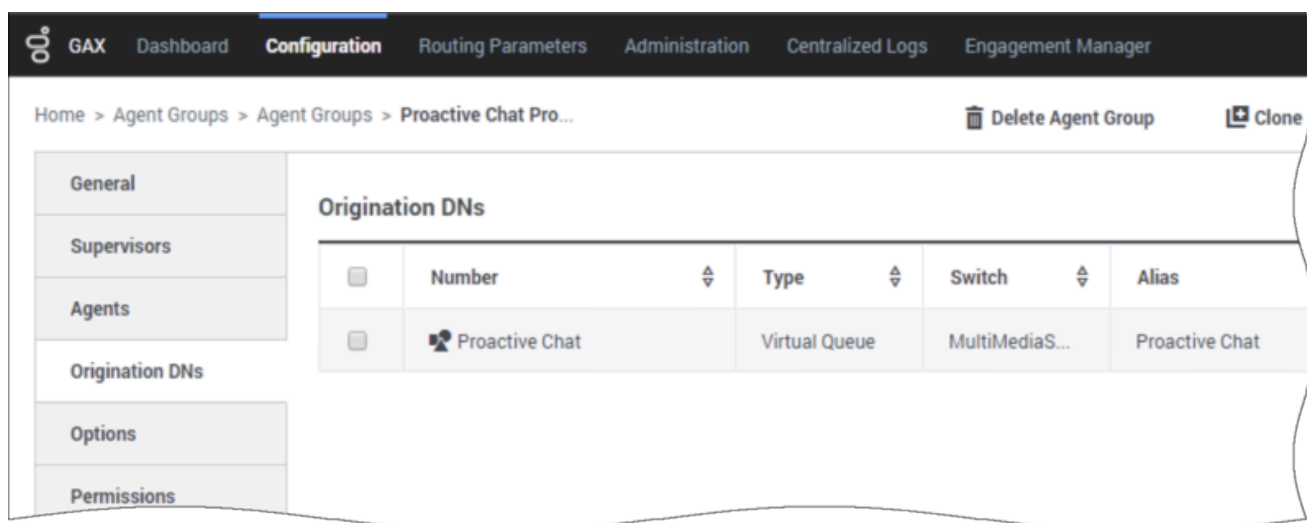
Agent groups can specify more than one virtual queue, and each virtual queue can specify a different media type. For example, **Proactive Chat** could specify **chat** and **Proactive Voice** could specify **voice**. This capability is important for agents in the agent group who have *blended* capabilities—that is, those who can work with several media types simultaneously—as they can be considered for pacing targets that use each of their available media types.

Configure a regular agent group

The following image shows an agent group called **Proactive Voice**. This agent group is associated with a voice-based virtual queue called **Proactive Voice**. The agent group and virtual queue form a pacing target called **Proactive Voice - voice**.



The next two images show two agent groups that are associated with the **Proactive Chat** virtual queue. The first one is **Proactive Chat** and the second is **Proactive Chat Sales**. These agent groups and the virtual queue form the **Proactive Chat - chat** pacing target.



Home > Agent Groups > Agent Groups > Proactive Chat Sales Properties

General

Name *
Proactive Chat Sales

Capacity Table

Quota Table

Cost Contract

Site

Script

Tenant

Configure a blended agent group

The next image shows a *blended* Agent Group—that is, an agent group that can handle more than one media type. The **Proactive Blended** agent group supports two virtual queues:

- **Proactive Chat**, which is part of the **Proactive Chat - chat** pacing target
- **Proactive Voice**, which is part of the **Proactive Voice - voice** pacing target

Home > Agent Groups > Agent Groups > Proactive Blended Properties

General

Origination DNs

<input type="checkbox"/>	Number	Type	Switch	Alias
<input type="checkbox"/>	Proactive Chat	Virtual Queue	MultiMediaS...	Proactive Chat
<input type="checkbox"/>	Proactive Voice	Virtual Queue	MultiMediaS...	

Pacing service

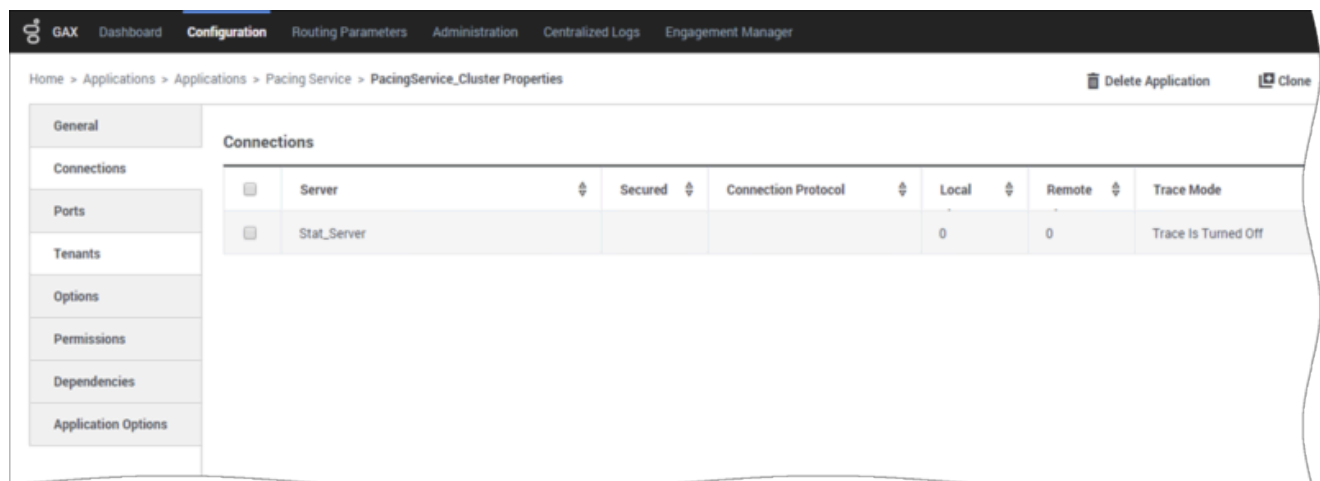
To install Agent Pacing Service, you need:

- A dedicated Genesys application with a type of **Application Cluster**
- A set of applications with a type of **Genesys Generic Server**, with one application for each instance of Pacing Service Server.

The Application Cluster application must include:

- Descriptions of all of the options (which are inherited by the nodes)
- Connections to the **Stat Server** and (optionally) **Message Server** applications.

The node-related applications must specify the hosts and ports for their server instances and for their connections to the cluster application. The following screenshot shows some of the setup for configuring a cluster and two nodes.



The screenshot shows the GAX Configuration interface. The top navigation bar includes links for GAX, Dashboard, Configuration, Routing Parameters, Administration, Centralized Logs, and Engagement Manager. The breadcrumb trail indicates the current path: Home > Applications > Applications > Pacing Service > PacingService_Cluster Properties. On the right side of the header, there are buttons for 'Delete Application' and 'Clone'.

The left sidebar contains a list of configuration categories: General, Connections, Ports, Tenants, Options, Permissions, Dependencies, and Application Options. The 'Connections' category is currently selected.

The main content area, titled 'Connections', displays a table with the following data:

	Server	Secured	Connection Protocol	Local	Remote	Trace Mode
<input type="checkbox"/>	Stat_Server			0	0	Trace is Turned Off

GAX Dashboard **Configuration** Routing Parameters Administration Centralized Logs Engagement Manager

Home > Applications > Applications > Pacing Service > PacingService_Node_1 Properties Delete

General
Connections
Ports
Tenants
Options
Permissions
Dependencies
Application Options

Ports

ID	Port	Connection	HA Sync	Listening Mode
default	9081	http		Secured

GAX Dashboard **Configuration** Routing Parameters Administration Centralized Logs Engagement Manager

Home > Applications > Applications > Pacing Service > PacingService_Node_2 Properties Delete Application Clone

General
Connections
Ports
Tenants
Options
Permissions
Dependencies
Application Options

Connections

Server	Secured	Connection Protocol	Local	Remote	Trace Mode
PacingService_Cluster	✓		0	0	Trace Is Turned Off

The cluster application specifies the following options:

- log Section—Behavior of the logging subsystem
- metrics Section—Metrics produced by the nodes
- pacing Section—Pacing-related configuration, including the optimization goal
- pacingEndpoint Section—Genesys Predictive Engagement endpoint configuration that specifies the URI path where Genesys Predictive Engagement listens to pacing REST requests. This path will be combined with the base URL specified in the transaction object `hybrid_integration`
- forward-proxy Section—Connection options for a forward proxy



Home > Applications > Applications > Pacing > Pacing_Service_900200_Cluster Properties

 Delete Application

General	Application Options Quick Filter				
Connections					
Ports					
Tenants					
Options					
Permissions					
Dependencies					
Application Options					
	<input type="checkbox"/>	Name	Section	Key	Value
	<input type="checkbox"/>	metrics \ messageServer.logFrequency	metrics	messageServer.logFrequency	10min
	<input type="checkbox"/>	metrics \ GcFrequency.threshold	metrics	GcFrequency.threshold	24
	<input type="checkbox"/>	metrics \ reporter.console.logFrequency	metrics	reporter.console.logFrequency	30min
	<input type="checkbox"/>	metrics \ reporter.messageServer.logFrequency	metrics	reporter.messageServer.logFrequency	30min
	<input type="checkbox"/>	metrics \ reporter.log.enabled	metrics	reporter.log.enabled	false
	<input type="checkbox"/>	metrics \ GcLatency.threshold	metrics	GcLatency.threshold	1000
	<input type="checkbox"/>	metrics \ reporter.messageServer.enabled	metrics	reporter.messageServer.enabled	true
	<input type="checkbox"/>	metrics \ reporter.log.logFrequency	metrics	reporter.log.logFrequency	30min
	<input type="checkbox"/>	metrics \ reporter.console.enabled	metrics	reporter.console.enabled	false
	<input type="checkbox"/>	▼ pacingEndpoint			
	<input type="checkbox"/>	pacingEndpoint \ targetEndpoint	pacingEndpoint	targetEndpoint	v2/journey/actiontargets/bulk