



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 Mobile Services Deployment Guide

Configuring Chat Version1

12/17/2025

Configuring Chat Version1

This page details the specific configuration steps required to use **Chat API Version 1** included with Genesys Mobile Services. This API includes two types of chat services:

- The basic [request-chat service](#).
- The ORS-Chat services described in the [Chat Immediate](#) and [Chat Delayed](#) scenarios.

For further details about the usage of this API, refer to the [Chat API Version1](#) page.

Prerequisites for Chat V1 Dependencies

Prerequisites: Before you can start, you must complete the previous installation steps which include:

- [Create](#) a GMS application object, which includes connections to the Solution Control Server and to the Chat Server.
- [Install GMS](#) in your system.
- Set the mandatory [GMS dependencies](#).

To use the Chat API Version 1, you must specify configuration options in the Application objects for the following objects:

- Genesys Mobile Services
- Chat Server

Important

The Chat Server must be installed and configured. Refer to the [eServices Deployment Guide](#), which include details on how to install a Chat Server when deploying eServices.

The following sections provide details about configuration changes required to use chat with your Genesys Mobile Services deployment. Procedures and illustrations on this page use Genesys Administrator, although the configuration can also take place using Configuration Manager.

Setting Chat Options in GMS Configuration

The following configuration options must be specified in your Genesys Mobile Services Application object:

1. In Genesys Administrator Extension, find **Configuration Manager > Environment > Applications**.
2. Locate and edit the Genesys Mobile Services Application object that you previously created and configured.
3. Under the *Application Options* tab, in the *chat* section, include the following mandatory **configuration options**:

chat_session_request_timeout

Section: chat

Default Value: 30000

Valid Values: Any positive integer

Changes Take Effect: Immediately upon notification.

Duration in milliseconds after which the chat interaction gets deleted.

default_chat_endpoint

Section: chat

Default Value: Environment:default

Valid Values: <tenant_name>:<chat_endpoint>

Changes Take Effect: Immediately upon notification.

This option is used for all chat services in order to define the queue (URS) where the chat session initiated by GMS will enter. The value of this option is the tenant name on which the service(s) will proceed, and the chat endpoint as defined in the ChatServer option. For example, the section endpoints for the tenant Environment in the chat options is written as endpoints:1. This section contains the endpoint options (for example, default=queue). The chat endpoint value to use this default endpoint in the Environment tenant is Environment:default.

Tip

You can supersede this option for each chat service using the `_chat_endpoint` option with the same `<tenant_name:chat_endpoint>` value.

_client_timeout

Section: chat

Default Value: 900

Valid Values: Any positive integer
Changes Take Effect: After restart

Client timeout in seconds for Cometd chat sessions.

If the client does not interact with the Chat service (refresh, send message, send event), GMS stops to poll the Chat server, and the Chat session is closed. This option applies only to chat sessions implemented using Cometd connections. For non-Cometd implementation, Chat server timeout parameters apply. The default value for this option is 15 minutes.

Single Tenant Support and Chat Service

For chat services ([Chat API Version1](#)), the default chat endpoint value in the `default_chat_endpoint` option can be used for all services. Or by service, you can customize the `_chat_endpoint` option. These options, `default_chat_endpoint` and `_chat_endpoint`, are composed of `<tenant name>:<chat endpoint name>`.

Chat Server Configuration in Primary/Backup mode

The Chat Server Application object being used by your Genesys Mobile Services deployment should have the following configuration updates:

- Add a connection to Interaction Server.
- Set a backup server and specify the redundancy type.

Configuring your Chat Server's Connections

The top screenshot shows the 'Configuration Manager' interface with the 'Connections' tab selected. The 'Connections' table lists the following connections:

	Server	Secured	Connection Protocol
☐	Message_Server		addp
☐	ContactServer		addp
☒	Interaction_Server		addp

The bottom screenshot shows the 'Ports' tab selected. The 'Ports' table lists the following ports:

	ID	Port	Connection	HA Sync
☐	default	4800		
☐	ESP	8788		
☒	webapi	8778		

1. In Genesys Administrator Extension, find **Configuration Manager > Environment > Applications**.
2. Locate and edit the Chat Server Application object associated with your Genesys Mobile Services deployment.
3. View the **Connections** tab and click **Add**.
4. Locate and select the Interaction Server Application object that you want to use.
5. Make sure to use either port with id webapi or port with chat-flex protocol of the Chat Server.

Repeat this procedure for each Chat Server associated with your Genesys Mobile Services application.

Setting Chat Server HA-Specific Options

Important

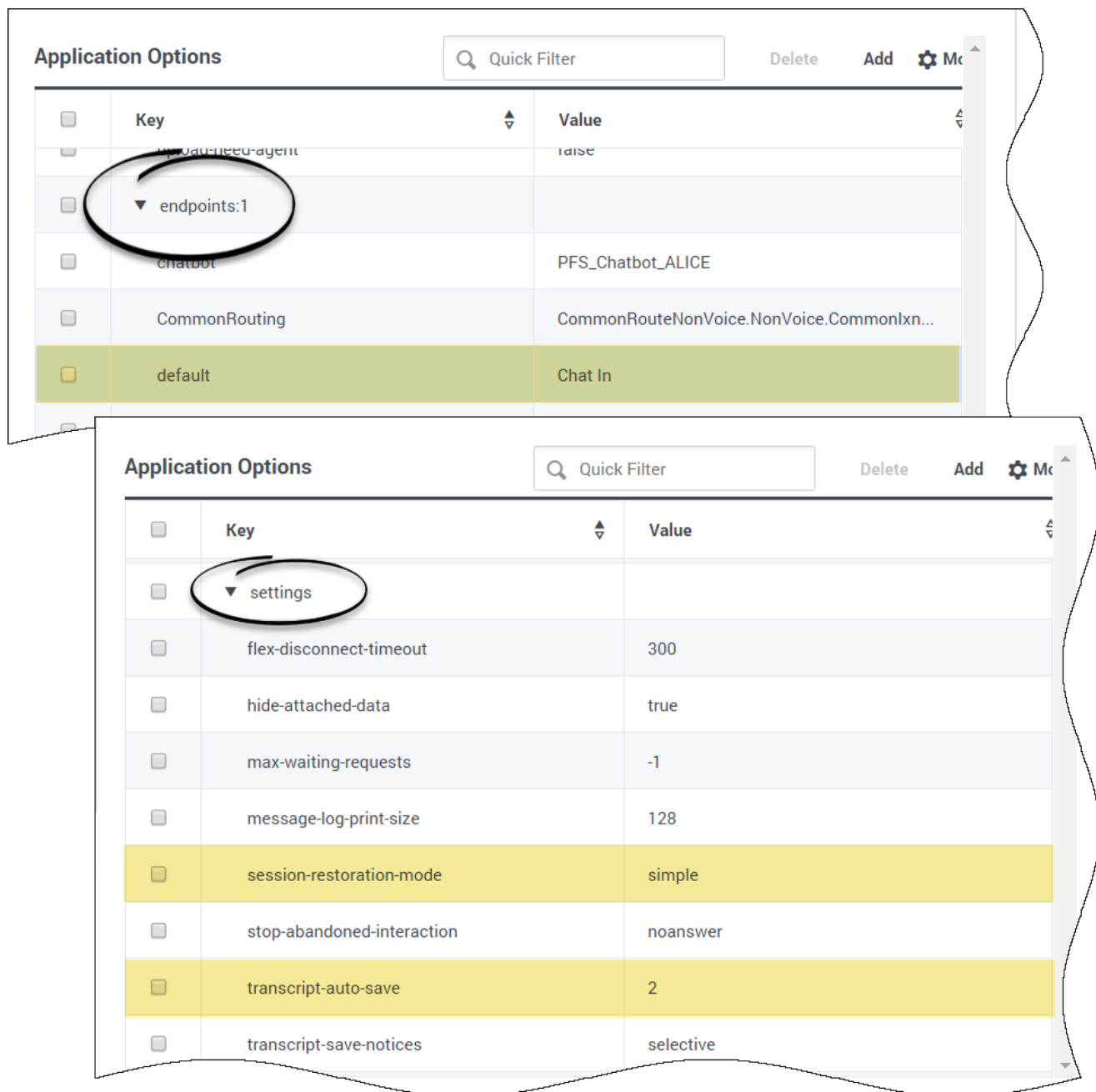
Enabling High Availability requires Chat Server **8.1.000.20 or higher**.

The screenshot shows the 'Chat_Server Properties' configuration page in the 'Configuration' section. The 'General' tab is selected and circled. The page includes a sidebar with options: General, Connections, Ports, Tenants, Options, Permissions, Dependencies, and Application Options. The main content area contains the following fields:

- ☒ Auto-Restart
- ☒ Primary
- Host*: 135.39.45.125
- Backup Server: Chat_Server_AA_Backup
- Redundancy Type: Warm Standby
- Timeout*: 10
- Attempts*: 1
- ☒ Login As System
- Login As Account: (empty field)
- Certificate: (empty field)
- Certificate Description: (empty field)

Buttons for Clone, Delete, and Move To are visible at the top right. A 'default' label is also present.

1. View the *Backup Server Info* section on the *General* tab.
2. Specify a *Backup Server* value.
3. Set the *Redundancy Type* to *Warm Standby*.



Under the **Application Options** tab, include the mandatory configuration options required for High Availability

- Section: **endpoints:1**
default = Chat In
- Section: **settings**
session-restoration-mode = simple
transcript-auto-save = 2

Chat Server Configuration in N+1 Mode

Home > Applications > Applications > **GMS_851_Single Properties** Clone Delete Move To default

General

Connections

Ports

Tenants

Options

Permissions

Dependencies

Application Options

Connections Add Remove

☐	Server	Secured	Connection Protocol	
☒	Solution_Control_Server			(
☐	Message_Server			(
☐	Stat_Server			(
☐	Orchestration_Server			(
☒	Chat_Server_1			(
☒	Chat_Server_2			(

Important

This configuration is recommended to support Load Balancing with your GMS application.

Prerequisites: Your Chat Servers must not be configured in Primary/Backup mode—That is, their configuration must not include a backup server.

Edit your GMS configuration and add all Chat Servers and a Solution Control Server to the list of connections.

Note: Make sure to use either port with id webapi or port with chat-flex protocol of Chat Server in the **Connections** tab of GMS.

Add an HTTP listening port to your Chat Server Configuration

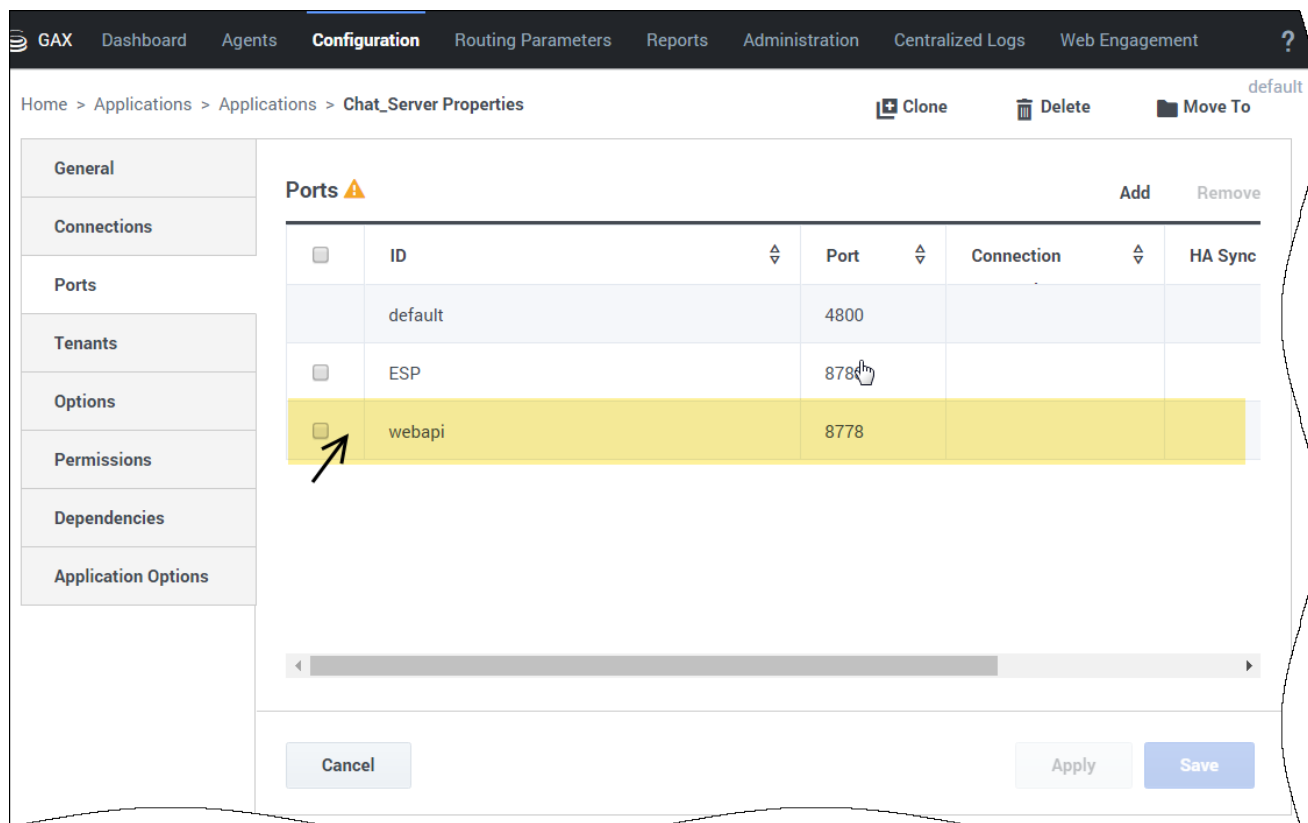
The screenshot shows the GAX Configuration interface for 'Chat_Server Properties'. The 'Ports' section is active, displaying a table with the following data:

ID	Port	Connection	HA Sync
default	4800		
ESP	878		
webapi	8778		

The 'webapi' row is highlighted in yellow, and a hand icon points to the 'webapi' ID cell. The interface includes a sidebar with navigation options like General, Connections, Ports, Tenants, Options, Permissions, Dependencies, and Application Options. At the top, there are tabs for GAX, Dashboard, Agents, Configuration, Routing Parameters, Reports, Administration, Centralized Logs, and Web Engagement. Below the table are 'Cancel', 'Apply', and 'Save' buttons.

Make sure to have an HTTP listening port (called `webapi` in our sample) in the configuration of each of your Chat Server.

Add an HTTP listening port to your Chat Server Configuration



Make sure to have an HTTP listening port (called `webapi` in our sample) in the configuration of each of your Chat Server.

Tune Chat Session Refresh

GMS nodes are responsible for refreshing chat sessions (that cover the refresh of the chat customer transcript using comet notification). However, a given node may sometimes be unable to refresh the previous chat sessions even if it can accept new ones. In that case, another GMS node will retrieve and refresh the suspended chat sessions. This is possible only if the Chat Server keeps these sessions alive for long enough.

To enable this mechanism, configure the `flex-disconnect-timeout` option in the **settings** section of the Chat Server application. Increase the value to **300** seconds so that the Chat Server keeps the chat sessions alive long enough for GMS to refresh. Then, GMS can again refresh uncompleted chat sessions after 60 seconds.

flex-disconnect-timeout

Section: settings

Default Value: 45

Valid Values: Any integer from 1-1728000

Changes Take Effect: Immediately

Modified: 8.5.301.06

Specifies timeout (in seconds) after which Chat Server disconnects an inactive web (flex) chat client. Inactivity is defined as the absence of protocol requests, not the absence of messages.

Next Steps

With basic configurations now complete, you can start loading and managing your services, using the GMS Service Management User Interface.

- [Service Management User Interface](#)

You can also configure additional, advanced settings that are outlined in the following section:

- [Configuration](#)