Bot Gateway Server Quick Start Guide

How Bot Gateway Server works

12/15/2025

Contents

How Bot Gateway Server works

Bot Gateway Server (BGS) is a Genesys component that is implemented as an eServices Digital Messaging Server (DMS) driver. BGS provides an integration platform for the deployment of chat bots that participate in chat sessions conducted by Chat Server. At a high level, the functional behavior of BGS is as follows:

- BGS is deployed as dedicated instances of DMS. Chat bot plugins (packaged as JAR files) are placed into a special subfolder of BGS and must be configured within the options of the DMS application.

- A workflow controls chat bots by sending an ESP request through Interaction Server. Workflows can start and stop the bot, as well as send data to the bot. For example, to connect a bot to chat session:

1. The Workflow sends ESP request **StartBot** to the BGS application.

2. BGS searches for the bot ID and, if found, requires the bot plugin to create a bot instance that serves this particular chat session.

3. After obtaining the bot instance, BGS connects the chat bot instance to the chat session as a user of type agent with a special attribute signifying "bot" style, and relays the notifications about chat session activity to the bot instance. The chat bot instance can use the BGS API to submit messages into the chat session, to exit chat session, and to update the userdata of the interaction.

4. At the end (either defined by a condition or caused by a **StopBot** ESP command) BGS removes the chat bot instance from the chat session and disposes the chat bot instance.

The following diagram explains how the chat bot is connected to the chat session:

How Bot Gateway Server works (diagram)

The following sequence diagram shows the interaction between the Customer, Chat Server, Bot Gateway Server, Chat Bot, Interaction Server, and Workflow:

- 1 start chat
- 2 submit interaction
- 3 Ok
- 4 process
- 5 start bot
- 6 invite
- 7 agree
- 8 connect
- 9 connected
- 10 started
- 11 Ok
- 12 started
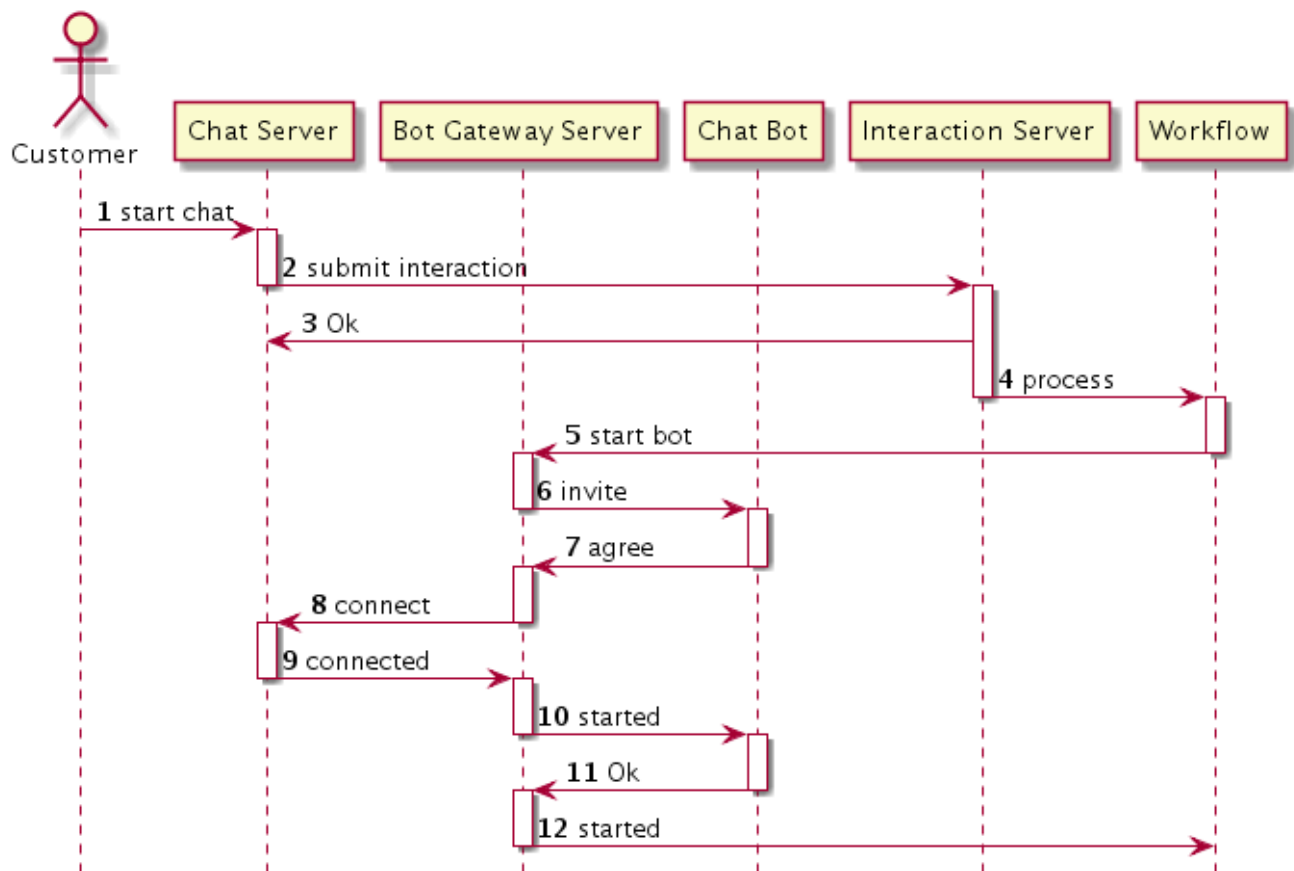
High Availability support

Beginning in version 9.0.006.04, Bot Gateway Server supports high availability mode which provides the following functionality:

- Automatic restart of the bot in case of a BGS switchover (either manual/planned or automatic due to failover). A special API method allows the bot to save and restore the bot context.

- Automatic reconnection to a chat session after losing the connection to Chat Server.

Workflow ESP API description

BGS ESP API (the control API) is intended to be used from a workflow (strategies of URS/ORS) to start/stop a bot and to send data to a bot. Besides using the API in a workflow, you can use the API from other applications like Custom Agent Desktop and other bots. The BGS installation package includes a sample URS and ORS workflow (see the description on how to test the bot on the Deployment page).

In order to send the command or request to BGS, the strategy must use the **External Service** object

and provide the following:

- **Application Type:** SocialMS
- **Application Name:** *The name of your DMS application where BGS is deployed*
- **Service name:** ChatBotPlatform
- **Method:** Choose one of the 4 ESP methods

How chat bots integrate with the workflow

Chat bots integrate with the workflow using the following two modes:

- Waiting mode: To use this mode, the ChatBotHoldup startup parameter must be set to true. In order to facilitate the workflow to hold the interaction until the bot finishes, a special workflow queue view with a scheduling condition must be used which is demonstrated in the workflow sample.

- Parallel mode: The workflow immediately routes the interaction to an agent after the bot starts. The bot works in parallel with the agent.

> ### Tip
> The Bot Gateway Server installation package (IP) provides a sample workflow that demonstrates both modes.

How chat bots finish execution

BGS stops the execution of a given chat bot instance and removes it under the following conditions:

Condition	BGS removes bot from chat session
When **StopBot** ESP request is received.	with after action **keep-alive**
When all participants (except bots, system, and external users) have left the chat session.	with after action **keep-alive**
When a bot instance explicitly requests (in Java API) to leave a chat session.	with after action **keep-alive**
When an agent joins a chat session (after the chat bot was already connected to the chat session). This condition applies only if the parameter **StopBotOnAgentArrival** has a value of true.	with after action **keep-alive**
When a customer leaves a chat session (after the chat bot was already connected to the chat session). This condition applies only if the parameter **StopBotOnCustomerLeft** has a value of true.	with after action **keep-alive**

Condition	BGS removes bot from chat session

When Chat Server removes the bot participant because:
 	
• Another participant explicitly requested to remove the bot
 	
• The chat session is ending due to the following conditions:
 	
   • Agent left the chat session with after-action **close** if no agents or **force close**
 	
   • Idle control monitoring.
 	
   • Interaction stopped when only customer was present in the chat session.
 	
   • Unrecoverable issue with UCS in HA (high-availability) mode.
	n/a

When BGS detects a disconnection from Chat Server, or a shut down of BGS has started.	n/a