



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.

Widgets Reference

API Events

API Events

Once you've registered your own plugin on the bus, you can subscribe and listen for published events. Below we'll quickly register a new plugin on the bus using the global bus object.

Important

The global bus object is a debug tool. When implementing Widgets on your own site, do not use the global bus object to register your custom plugins. Instead, see [Widgets Extensions](#) for more information about extending Genesys Widgets.

```
var oMyPlugin = window._genesys.widgets.bus.registerPlugin('MyPlugin');  
oMyPlugin.subscribe('WebChatService.ready', function(e){});
```

Name	Description	Data
ready	WebChatService is initialized and ready to accept commands.	n/a
restored	Chat session has been restored after page navigation or refresh.	n/a
restoreTimeout	Chat session restoration attempted was denied after user navigated away from originating website for longer than the time limit: default 60 seconds.	n/a
restoreFailed	Could not restore chat session after page navigation or refresh.	n/a
restoredOffline	Chat session was restored normally but chat server is offline. This means no messages can come through. When chat server is comes back online, 'chatServerBackOnline' is published.	n/a
messageReceived	A new message has been received from the server. Includes text messages, status messages, notices, and other message types.	n/a
error	An error occurred between the client and the server.	(AJAX Response)
started	Chat session has successfully	n/a

Name	Description	Data
	started.	
ended	Chat session has successfully ended.	n/a
agentTypingStarted	Agents has started typing a new message.	n/a
agentTypingStopped	Agent has stopped typing.	n/a
pollingStarted	Chat server automatic polling has started.	n/a
pollingStopped	Chat server automatic polling has stopped.	n/a
clientConnected	Indicates the user has been connected to the chat session.	{message: (object), agents: (object), numAgentsConnected: (number)}
clientDisconnected	Indicates the user has been disconnected form the chat session.	{message: (object), agents: (object), numAgentsConnected: (number)}
agentConnected	Indicates an agent has connected to the chat.	{message: (object), agents: (object), numAgentsConnected: (number)}
agentDisconnected	Indicates an agent has disconnected from the chat.	{message: (object), agents: (object), numAgentsConnected: (number)}
supervisorConnected	Indicates a supervisor has connected to the chat.	{message: (object), agents: (object), numAgentsConnected: (number)}
supervisorDisconnected	Indicates a supervisor has disconnected from the chat.	{message: (object), agents: (object), numAgentsConnected: (number)}
clientTypingStarted	The user has started typing. Sends an event to the agent.	n/a
clientTypingStopped	After a user stops typing, a countdown begins. When the countdown completes, the typing notification will clear for the agent.	n/a
disconnected	Cannot reach servers. No connection. Either the user is offline or the server is offline.	n/a
reconnected	Connection restored. This event is only published after 'disconnected'.	n/a
chatServerWentOffline	Chat server has gone offline but chat session has not ended. New messages are temporarily unavailable. This event is published only after the configuration option 'pollExceptionLimit' has been	n/a

Name	Description	Data
	exceeded. Default limit is 5 poll exceptions. 'restoredOffline' is an alternate to this event that is used only when the chat server is down while trying to restore your chat session. The reason for having two events is to allow for separate handling of both scenarios.	
chatServerBackOnline	Chat server had come back online after going offline. This will only be published after 'chatServerWentOffline'.	n/a
connectionPending	If there is a connection problem and WebChatService is trying to reconnect, this event will be published. Published before 'chatServerWentOffline'.	n/a
connectionRestored	Is published when the connection has been reestablished. Publishes at the same time as 'chatServerBackOnline'.	n/a