Widgets Reference

API Commands

12/12/2025

Contents

API Commands

Once you've registered your own plugin on the bus, you can call commands on other registered plugins. 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.command('SideBar.open');
```

configure

Internal use only. The main App plugin shares configuration settings to widgets using each widget's configure command. Sidebar widget has to be configured atleast with one channel. The configure command can also be called at runtime with new configuration, this will override the existing configuration showing new channels on the screens.

Example

```
oMyPlugin.command('SideBar.configure', {

        showOnStartup: false,
        position: 'left',
        expandOnHover: false,
        channels: [
                {
                        name: 'ChannelSelector',
                        clickCommand: 'ChannelSelector.open',
                        clickOptions: {},

                        //use your own static string or i18n query string for the below two
display properties. Example for i18n query string: '@i18n:sidebar.ChannelSelectorName' where
'sidebar' refers to plugin namespace and 'ChannelSelectorName' name refers to the property
key containing the actual text.

                        displayName: '@i18n:sidebar.ChannelSelectorName',
                        displayTitle: 'Get assistance from one of our agents right away', //
Your own static string
                        icon: 'agent',
                        onClick: function($, CXBus, Common) {
```

```
                              _genesys.widgets.bus.command('MyPlugin.open');
                    }
             }
             ...
      ]
}).done(function(e){

      // Sidebar configured successfully

}).fail(function(e){

      // Sidebar failed to configure properly
});
```

Options

Option	Type	Description
showOnStartup	boolean	Shows the sidebar on the screen when Widgets is launched.
position	string	Defines the position of sidebar on the screen.
expandOnHover	boolean	Enables the expand or contract behavior of sidebar.
channels	array	Array containing each channel configuration object. The order of channels are displayed based on the order defined here.
channels[index].name	string	Name of the channel. It can be found in the namespace section documentation of each Widget. Used to identify official channels vs custom channels. If a reserved name is used here, Sidebar will apply default values for that channel. To override the default values or when defining a new custom channel, use the below following properties.
channels[index].clickCommand	string	Change the default command that is triggered when clicked.
channels[index].clickOptions	object	Pass valid command options that are used in above click command execution.
channels[index].displayName	string or i18n query string	Change the default display name for this channel with your own static string or to achieve localization, use i18n query string. Syntax: @i18n:<plugin namespace>.<display key>.
channels[index].displayTitle	string or i18n query string	Change the default tooltip

Option	Type	Description
		content for this channel with your own static string or to achieve localization, use i18n query string. Syntax: @i18n:<plugin namespace>.<display key>.
channels[index].icon	string	Change the default Icon for this channel. For the list of Icon names see Included Icons.
channels[index].onClick	function	Define a custom onclick function, this overrides clickCommand and clickOptions.

Resolutions

Status	When	Returns
resolved	When configuration options are provided and set	n/a
rejected	When no configuration options are provided	'Invalid configuration. Please ensure at least one channel is configured.'

open

Opens the Sidebar UI. In Desktop, it opens as an actual SideBar and shows the configured channels where as in mobile it opens as a button at the bottom to start.

Example

```
oMyPlugin.command('SideBar.open');
```

Resolutions

Status	When	Returns
resolved	When sidebar is successfully opened	n/a
rejected	When sidebar is already opened	'Already opened'

close

Closes the Sidebar UI.

Example

```
oMyPlugin.command('SideBar.close');
```

Resolutions

Status	When	Returns
resolved	When sidebar is successfully closed	n/a
rejected	When sidebar is already closed	'already closed'

expand

To show more details about the channels, it slides out from the sides of the screen in desktop but expands to full screen in mobiles.

Example

```
oMyPlugin.command('SideBar.expand');
```

Resolutions

Status	When	Returns
resolved	When sidebar is successfully expanded	n/a
rejected	When sidebar is already expanded	'sidebar already expanded'

contract

Slides back showing only the channel buttons in desktop and sidebar launcher button in mobile.

Example

```
oMyPlugin.command('SideBar.contract');
```

Resolutions

Status	When	Returns
resolved	When sidebar is successfully contracted	n/a
rejected	When sidebar is already contracted	sidebar already contracted