Deployment Guide

Configuring Genesys Widgets

4/14/2025

Contents

Configuring Genesys Widgets

Genesys Widgets supports multiple widgets including Webchat, Web Engagement, and Co-browse. You can configure all widgets and services in the same configuration object. When you add new Genesys products and services, you must update your Genesys Widgets configuration to enable those widgets.

After you deploy the Genesys Widgets on your website, configure the CX Widget by defining the **global window._genesys** JavaScript object.

To include the JavaScript script, you can choose one of the following options:

- Place the script inline on your website; or

- Place it in a separate JavaScript file, and then include the file on your page.

The following example is a basic view of the global Genesys Widgets configuration object:

```
<script>

        if(!window._genesys)window._genesys = {};
        if(!window._gt)window._gt = [];

        window._genesys.widgets = {

                main: {},
                webchat: {},
                cobrowse: {},
                gwe:{}
        };

</script>

// include widgets.min.js after defining your configuration options
```

The following example is a populated Widget configuration that includes configuration options for Webchat, SendMessage, Web Engagement (GWE), and Co-browse (GCB):

> ### Important
> Your configuration options must be defined on the page **before** widgets.min.js is loaded. Failing to do so may result in errors.

> ### Tip
> Note that Web Engagement and Co-browse is optional.

```
<script>

      if(!window._genesys)window._genesys = {};
      if(!window._gt)window._gt = [];

      window._genesys.widgets = {
            main: {

                  theme: "dark",
                  lang: "en",
                  i18n: "http://HOST:PORT/path/to/languages/file.json",
                  customStylesheetID: "<ANY_ID_NAME>",
                  plugins: [
                        "cx-webchat",
                        "cx-webchat-service",
                        "cx-cobrowse",
                        "cx-gwe",
                        "cx-send-message",
                        "cx-send-message-service",
                        ],
                  mobileMode: 'auto',
                  mobileModeBreakpoint: 600
                  },
            webchat: {

                  apikey: "0123456789", // Used for Apigee service only
                  dataURL: "http://HOST:PORT/path/to/chat/service",
                  userData: {},
                  proactive: {

                        enabled: true,
                        idleTimer: 5,
                        cancelTimer: 30
                  },
                  chatButton: {

                        enabled: true,
                        template: false,
                        openDelay: 1000,
                        effectDuration: 300,
                        hideDuringInvite: true
                  }
```

```
            },
            cobrowse: {

                    src: "<COBROWSE_SERVER_URL>/cobrowse/js/gcb.min.js",
                    url: "<COBROWSE_SERVER_URL>/cobrowse"
            },
          gwe: {
                    httpEndpoint: "http://<GWE_SERVER_URL>:<PORT>",
                    httpsEndpoint: "https://<GWE_SERVER_URL>:<PORT>",
                    dslResource: "http://HOST:PORT/path/to/domain-model.xml"
            },
          sendmessage: {

                    SendMessageButton : {

                            enabled: true,
                            template: false,
                            openDelay: 1000,
                            effectDuration: 300
                    },
                    apikey: "0123456789", // Used for Apigee service only
                    dataURL: "http://HOST:PORT/path/to/sendmessage/service"
            },
    };

    (function(o){
    var f = function(){
            var d = o.location;
            o.aTags = o.aTags || [];
            for(var i=0;i<o.aTags.length;i++){
                    var oTag = o.aTags[i];
                    var fs = d.getElementsByTagName(oTag.type)[0], e;
                    if(d.getElementById(oTag.id)) return; e = d.createElement(oTag.type); e.id = oTag.id;
                    if(oTag.type == "script"){e.src = oTag.path;}
                    else{e.type = 'text/css';e.rel = 'stylesheet';e.href = oTag.path;}
                    if(fs){fs.parentNode.insertBefore(e, fs);}else{d.head.appendChild(e);}
            }},ol = window.onload;
            if(o.onload){typeof window.onload != "function"?window.onload=f:window.onload=function(){ol();f();}}else f();
})({location: document,
    onload: false,
    aTags: [{type:"script", id:"genesys-cx-widget-script", path:"http://www.yourhost.com/path/to/widgets.min.js"},
            {type:"link", id:"genesys-cx-widget-styles", path:"http://www.yourhost.com/path/to/widgets.min.css"}]});
```

```
</script>
```

Main Configuration

Genesys Widgets is a hub for multiple Genesys products and services. Some configuration options are set globally and therefore apply to all products and services running on the CX Widget platform. In the main application configuration you can configure options such as visual theme, language, and mobile support.

For detailed information on configuration options, see App Configuration Options.

Widget Configuration Options

- WebChat Configuration
- WebChatService Configuration
- SendMessage Configuration
- SendMessageService Configuration
- CoBrowse Configuration
- GWE Configuration
- Channel Selector Configuration

For a complete list of Widget configuration options, please see Genesys Widgets Reference.

Launcher

Launcher is a sample page which shows how widgets display on any host website.

Using a sample page has the following advantages:

- Viewing Genesys Widgets with your own configuration.
- Copying Configuration Script. Using the details you entered on the form, the configuration script is generated in the **Need Configuration Script** box. You can copy this script and use it on your website to launch widgets.

How to use Launcher

Webchat
Click the checkbox next to **Webchat** to enable it. Enter the mandatory field URL (marked with an asterisk) and click the **Launch** button. Other values are optional and self explanatory.

Co-browse
Click the checkbox next to the **Co-browse** heading to enable it. Enter required fields (marked with an

asterisk) and click the **Launch** button.

GWE
Click the checkbox next to the **GWE** heading to enable it. Enter required fields (marked with an asterisk) and click the **Launch** button.

Send Message
Click the checkbox next to the **Send Message** heading to enable it. Enter required fields (marked with an asterisk) and click the **Launch** button.

CallUs
Select the check box next to "Call Us" and provide the configuration data. Here, the **Edit/Use Sample Config** option is also provided and you can use this to prefill with the sample configuration and edit it for your own details. Ideally, call us is shown in the Live Assist widget or it can also be launched with bus command "CallUs.open".

ChannelSelector
Select the checkbox next to "Enable Sidebar with Live Assist (EWT)" to include ChannelSelector plugin. Enter the Stats URL field followed by Chat and CallUs Vitual Queue names to fetching Stats service Estimated Wait Time details. For the Live Assist widget to show Chat, SendMessage and CallUs channels, please make sure you select those plugins as well.

Callback
Select the checkbox next to "Callback" to include Callback and Calendar plugins. Enter the callback service provider URL field and other details as required, ensure "Enable Sidebar with Live Assist" is selected and click the "Launch" button.

Knowledge Center/Search
Select this checkbox to include Knowledge Center plugin. Ensure all the details are entered along with sidebar plugin enabled and click "Launch" button.

Important

Sidebar is not an officially released plugin; it is being used here here for the purpose of the launcher tool. Ideally, it should be launched with bus command "ChannelSelector.open".

Launcher tool (click to enlarge)