Widgets API Reference

CallUs

2/23/2026

Contents

- Developer

Learn how to display an overlay screen showing one or more phone numbers for customer service, as well as the hours that this service is available.

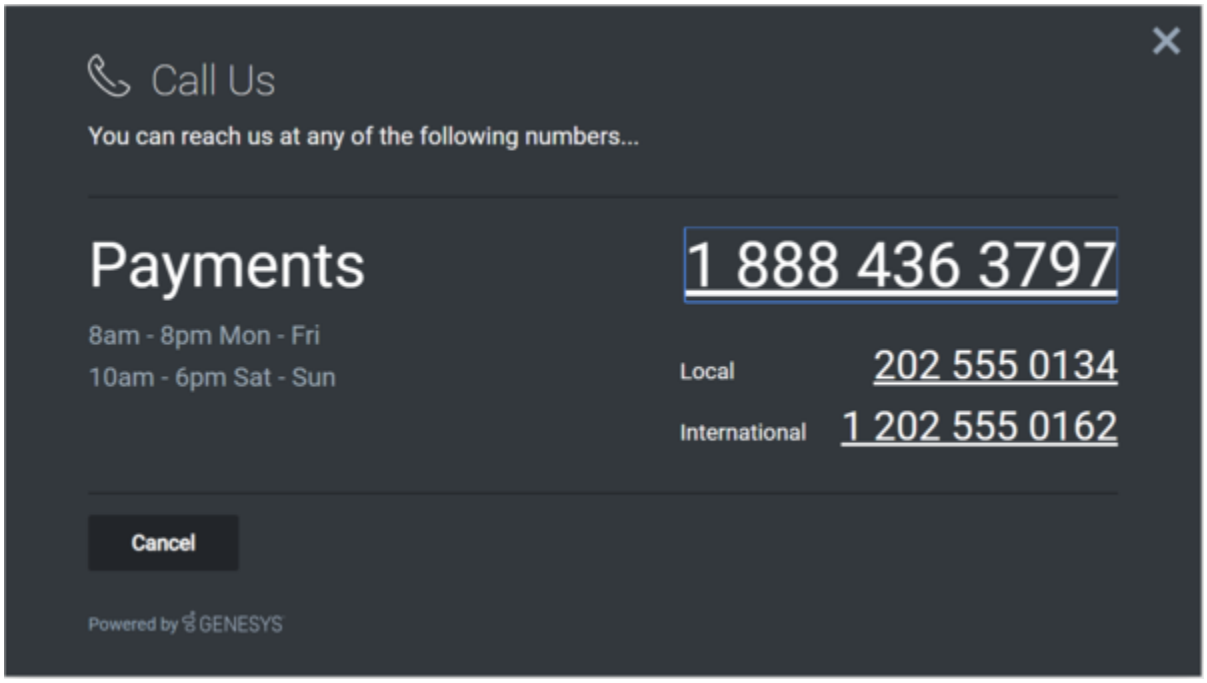
Related documentation:

-

Link to video

Overview

The CallUs Widget provides an overlay screen showing one or more phone numbers for customer service, as well as the hours that this service is available. The arrangement of numbers in this layout starts with a main phone number, which can be followed by alternative or additional phone numbers. Each number can be named, and there is no limit to the number of phone numbers you can include. If the list of numbers doesn't fit in the widget, the user can scroll down to see the rest.



Usage

Launch CallUs manually by using the following methods:

- Call the **CallUs.open** command
- Configure ChannelSelector to show CallUs as a channel
- Create your own custom button or link to open CallUs (using the "CallUs.open" command)

> **Important**
>
> By default a user has no way of launching the CallUs Widget. You must choose a suitable method for launching this widget.

Customization

You can customize and localize all of the text, titles, names, and numbers shown in the CallUs Widget by adding entries into your configuration and localization options. There are no formatting requirements. Text will appear as you entered it.

> **Important**
>
> If you do not configure the CallUs Widget it will appear as an empty overlay. You must configure this Widget before using it.

CallUs supports themes. You can create and register your own themes for Genesys Widgets.

Namespace

The CallUs plugin has the following namespaces tied up with each of the following types:

Type	Namespace
Configuration	callus
i18n—Localization	callus
CXBus—API commands & API events	CallUs
CSS	.cx-call-us

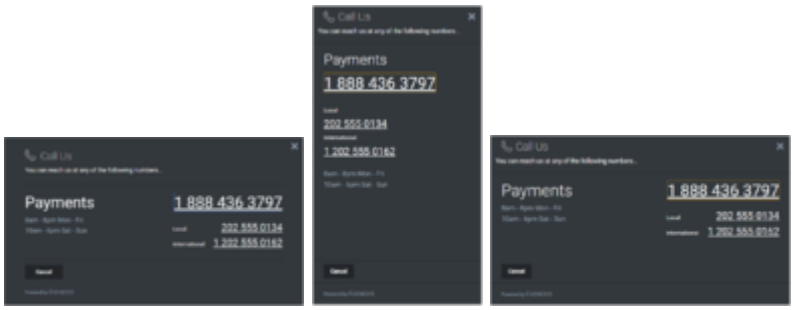
Mobile support

CallUs supports both desktop and mobile devices. Like all Genesys Widgets, there are two main modes: Desktop & Mobile. Desktop is employed for monitors, laptops, and tablets. Mobile is employed for smartphones. When a smartphone is detected, CallUs switches to special full-screen templates that are optimized for both portrait and landscape orientations.

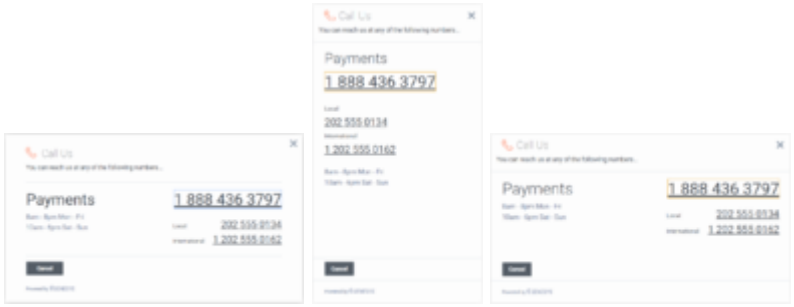
Switching between desktop and mobile mode is done automatically by default. You may configure Genesys Widgets to switch between Desktop and Mobile mode manually if necessary.

Screenshots

Dark theme



Light theme



Configuration

CallUs uses the **_genesys.widgets.callus** configuration property. You must specify all of the numbers and labels that appear in the CallUs UI.

Example

```
window._genesys.widgets.callus = callus: {

        contacts: [

                {
                        displayName: 'Payments',
                        i18n: 'Number001',
                        number: '1 202 555 0162'
                },
                {
                        displayName: 'Local',
                        i18n: 'Number002',
                        number: '202 555 0134'
                },
                {
```

```
                        displayName: 'International',
                        i18n: 'Number003',
                        number: '0647 555 0131'
                }
        ],

        hours: [

                '8am - 8pm Mon - Fri',
                '10am - 6pm Sat - Sun'
        ]
};
```

Options

Name	Type	Description	Default	Required

contacts	array	An array of objects that represent phone numbers and their labels. The first number in this list will display as the larger, main number. Phone labels can be set directly using the 'displayName' property or you can use String Names from your localization file by setting the String Name in the 'i18n' property. 'i18n' overrides 'displayName'.
		
		Example
		
		{
		
		"displayName": "Payments",
		"i18n": "Number001",
		
		"number": "1 202 555 0162"
		}
			[]	true

hours	array	Array of strings to show stacked in the business hours section. Strings here are freeform. See screenshots for ideas.	[]	

Localization

> **Important**
>
> For information on how to set up localization, please refer to Localize widgets and services.

Usage

Use the **callus** namespace when defining localization strings for the CallUs plugin in your i18n JSON file.

The following example shows how to define new strings for the **en** (English) language. You can use any language codes you wish; there is no standard format. When selecting the active language in your configuration, you must match one of the language codes defined in your i18n JSON file. Please note that you must only define a language code once in your i18n JSON file. Inside each language object you should define new strings for each widget.

Example i18n JSON

```
{
        "en": {
                "callus": {
                        "CallUsTitle": "Call Us",
                        "SubTitle": "You can reach us at any of the following NUMBERS...",
                        "CancelButtonText": "Cancel",
                        "AriaWindowLabel": "Call Us Window",
                        "AriaCallUsClose": "Call Us Close",
                        "AriaBusinessHours": "Business Hours",
                        "AriaCallUsPhoneApp": "Opens the phone application",
                        "AriaCancelButtonText": "Cancel"
                }
        }
}
```

API commands

Once you've registered your plugin on the bus, you can call commands on other registered plugins. Here's how to use the global bus object to register a new plugin on the bus.

> **Important**
>
> The global bus object is a debugging tool. When implementing Widgets on your own site, do not use the global bus object to register your custom plugins. Instead, see Genesys Widgets Extensions for more information about extending Genesys Widgets.

```
var oMyPlugin = window._genesys.widgets.bus.registerPlugin('MyPlugin');

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

open

Opens the CallUs UI.

Example

```
oMyPlugin.command('CallUs.open').done(function(e){

        // CallUs opened successfully

}).fail(function(e){

        // CallUs failed to open
});
```

Resolutions

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

close

Closes the CallUs UI.

Example

```
oMyPlugin.command('CallUs.close').done(function(e){

        // CallUs closed successfully

}).fail(function(e){

        // CallUs failed to close
});
```

Resolutions

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

configure

Internal use only. The main App plugin shares configuration settings to widgets using each widget's configure command. The configure command can only be called once at startup. Calling configure again after startup may result in unpredictable behavior.

Example

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

        contacts: [
                {
                        displayName: 'Payments',
                        i18n: 'Number001',
                        number: '1 888 436 3797'
                }
        ],
        hours: ['8am - 8pm Mon - Fri']

}).done(function(e){

        // CallUs configred successfully

}).fail(function(e){

        // CallUs failed to configure
});
```

Options

Option	Type	Description
contacts	Array	An array of objects that represent phone numbers and their labels. The first number in this list will display as the larger, main number.
hours	Array	Array of strings to show stacked in the business hours section. Strings here are freeform.

Resolutions

Status	When	Returns
resolved	CallUs configuration is provided	n/a
rejected	No configuration is provided	'Invalid Configuration'

API events

Once you've registered your plugin on the bus, you can subscribe to and listen for published events.

Here's how to use the global bus object to register a new plugin on the bus.

> **Important**
>
> The global bus object is a debugging tool. When implementing Widgets on your own site, do not use the global bus object to register your custom plugins. Instead, see Genesys Widgets Extensions for more information about extending Genesys Widgets.

```
var oMyPlugin = window._genesys.widgets.bus.registerPlugin('MyPlugin');

oMyPlugin.subscribe('CallUs.ready', function(e){});
```

Name	Description	Data
ready	CallUs is initialized and ready to accept commands	
opened	CallUs UI has been opened	
closed	CallUs UI has been closed	